

CA MICS® Resource Management

Analyzer Option for CICS Guide

Release 12.9



This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or 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 YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2014 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At <http://ca.com/support>, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at <http://ca.com/docs>.

Contents

Chapter 1: OVERVIEW	11
1.1 Primary Areas of Application.....	13
1.2 Major Features	15
1.3 Reporting and Inquiry Facilities.....	19
1.4 Files Overview	26
1.5 Product Prerequisites	28
1.6 Benefits	30
 Chapter 2: USAGE GUIDELINES	 31
2.1 Work File Compression	35
2.2 Data Source Background	36
2.2.1 Transaction vs. Task vs. Interaction	37
2.2.2 ISC and MRO Environment.....	39
2.2.3 CICS Response Time	45
2.2.4 CICS CPU Time	48
2.3 Data Analysis Guidelines	49
2.3.1 File Overview	50
2.3.2 Analyzing Performance	55
2.3.3 Analyzing Workload	58
2.3.4 Analyzing Uptime	62
2.4 Accounting and Chargeback Interface	65
2.5 Capacity Planner Interface	67
 Chapter 3: REPORTS	 69
3.1 MICF Inquiries	74
3.1.1 Service Analysis Inquiries	77
3.1.2 Workload Analysis Inquiries.....	173
3.1.3 Performance Analysis Inquiries.....	277
3.1.4 Availability Analysis Inquiries	292
3.1.5 Financial Analysis Inquiries.....	322
3.1.6 General Analysis Inquiries	370
3.1.7 CICS Transaction Gateway Inquiries.....	372
3.2 Changing Report Parameters	379
3.3 Running Reports in Batch	381
3.3.1 Production Report Interface Using MICF	381
3.3.2 JCL Requirements	382

Chapter 4: EXCEPTIONS

383

4.1 Exception Process Overview	385
4.2 Setting Exception Values	388
4.3 Detailed Exception Descriptions.....	389
04001: CICS Application Exceeds Avg Elapsed Time Standard	390
04002: CICS Application Exceeds Avg TCB/SRB CPU Limit	392
04004: CICS Application Exceeds Avg Wait Time Limit	394
04005: CICS Application Exceeds Avg Residency Time Limit.....	396
04006: CICS Application Exceeds Max Memory Limit	398
04007: CICS DL/I DMB Pool Wait	399
04008: CICS DB2 Pool Overflow	400
04009: CICS DB2 Wait Without Thread	401
04010: CICS Storage Violation Detected	402
04011: CICS Abnormal Condition Transaction	403
04012: CICS LSR String Wait	404
04013: CICS Max Task	405
04014: CICS Active Max Task	406
04015: CICS Short on Storage	407
04016: CICS VSAM String Wait.....	408
04017: CICS VSAM Buffer Wait	409
04018: CICS DL/I Thread Wait	410
04019: CICS Wait for Storage in DSA.....	411
04020: CICS Wait for Storage in EDSA.....	412
04021: CICS Wait for Temporary Storage (Main).....	413
04022: CICS Wait for File String	414
04023: CICS Wait for File Buffer	415
04024: CICS LSRPOOL Buffer Wait	416
04025: CICS Program Compressions	417
04026: CICS Storage Control Suspend.....	418
04027: CICS Temporary Storage PUT Wait	419
04028: CICS Main Storage Wait	420
04101: Prime Time CICS Throughput Degradation	421
04102: Hourly CICS System Resource Overload.....	423
04103: Hourly CICS Capacity Limit Objective Exceeded.....	425
04104: Hourly CICS Paging Rate Limit Exceeded.....	427
04105: Hourly CICS Short Service Objective Missed	428
04106: Hourly CICS Medium Service Objective Missed	430
04107: Hourly CICS Long Service Objective Missed	432
04108: Hourly CICS Total Service Objective Missed.....	434
04109: Hourly CICS Availability Objective Missed.....	436
04110: Hourly CICS Throughput Limit Objective Exceeded	437

04201: Hourly CICS User Resource Overload	438
04202: Prime Time CICS User Total Service Objective Missed.....	440

Chapter 5: FILES 443

5.1 Data Element Naming Conventions	445
5.2 CICS Information Area Files	446
5.2.1 CICS User Application Count File (CICCAC)	446
5.2.2 CICS Application Unit Activity File (CICCAU).....	451
5.2.3 CICS Dictionary File (CICCDC)	461
5.2.4 CICS Incident File (CICCIN).....	465
5.2.5 CICS File/DBD Activity File (CICCSF).....	469
5.2.6 CICS User Activity File (CICCSU).....	474
5.2.7 CICS System Activity File (CICCSY)	485
5.2.8 CICS MRO Activity File (CICCMR).....	498
5.3 CTG Information Area Files	503
5.3.1 Gateway Server Activity File (CTGGSA)	503

Chapter 6: DATA SOURCES 507

6.1 Data Source Descriptions	509
6.1.1 CICS Monitoring Facility (CMF).....	510
6.1.2 ASG-TMON for CICS TS for z/OS	514
6.1.3 OMEGAMON II for CICS.....	515
6.1.4 CICS Transaction Gateway SMF Type 111	516
6.2 File/Data Source Cross-Reference.....	517
6.3 Data Element/Data Source Cross-Reference.....	519
6.4 CA MICS and CA SMF Director Interface	548

Chapter 7: PARAMETERS 551

7.1 Environmental Considerations	552
7.2 Complex Level Parameters	553
7.2.1 Definition Statements (CICGENIN)	554
7.2.2 Define CICS Account Codes (CICACCT)	558
7.2.3 Code CICS Account Code Derivation Exit (CICACRT)	563
7.3 Unit Level Parameters	566
7.3.1 CICS Parameter Generation (CICPGEN).....	567
7.3.2 CICS Processing Options (CICOPS).....	569
7.3.3 INPUTRDR and INPUTCIC PARMS Members	649
7.3.4 Reserved for Future Use	653
7.3.5 CICS Application Unit Definition (CICAPU)	654
7.3.6 CICS Application Unit Derivation Routine (CICAURT).....	656

7.3.7 CICS Relative Longevity Routine (CICRLRT)	659
7.3.8 CICS Processing Thresholds (CICTHRSH)	664
7.3.9 CICS Multisystem Account Derivation Exit (CICMSAC)	668
7.3.10 Database Space Modeling (DBMODEL)	671

Chapter 8: INSTALLATION **675**

Chapter 9: PROCESSING **677**

9.1 Processing Considerations.....	677
9.1.1 Types of Data Processed	677
9.1.2 CMF Dictionary Processing.....	685
9.1.3 Dynamic Processing	687
9.2 Daily Update Processing Flow	691
9.2.1 Input CICS Data Dictionary	693
9.2.2 Input Raw CICS Data.....	694
9.2.3 Information Area Processing.....	706
9.2.4 Data Base Timespan Processing	707
9.2.5 File Aging.....	713
9.2.6 MICSLOG Operational Report	715
9.3 Utility Programs.....	717
9.3.1 Input List Program (CICILIST)	717
9.3.2 Data Dictionary Print Program (CICCDCPT)	719
9.3.3 CICS Standalone Processing Program (CICALONE)	720
9.3.4 CICS GDG Generation Program (CICGDGGN)	725

Chapter 10: MODIFICATION **727**

10.1 Checklists.....	727
10.1.1 Alter Derived Element Support	728
10.1.2 Add or Change Data Elements in a File	729
10.1.3 Add or Modify CICS Account Codes	731
10.1.4 Alter the Active Timespans for a File	733
10.1.5 Activate Archive for a File	744
10.1.6 Add a New File	747
10.1.7 Add or Delete a CICS Region in Daily Processing.....	749
10.1.8 Change the DDnames of Input Data.....	752
10.1.9 Activate the CICS File/DBD Activity File (CICCSF)	753
10.1.10 Change Retention Period of the CICCDC File	754
10.1.11 Change the Number of Work Files for DAY040.....	755
10.1.12 Process CMF User Clocks, Counters, and User Area	758
10.1.13 Activate the CICS MRO Activity File (CICCMR)	761

10.1.14 Convert to a Different Data Source	763
10.1.15 Process Additional Non-USER Areas in CMF	765
10.1.16 Enable Internal Step Restart	768
10.1.17 Implement Incremental Update.....	770
10.2 User Exits.....	773
10.2.1 Input Exits.....	776
10.2.2 Parameter-Related Exits	780
10.2.3 Output Exits.....	783

Appendix A: MESSAGES	793

Messages.....	793

Appendix B: DATA DICTIONARY	897

Chapter 1: OVERVIEW

The CA MICS Analyzer Option for CICS processes CICS monitoring data produced by IBM's CICS Monitoring Facility (CMF) and ASG-TMON. It also processes data that has the same format as these monitors, such as the SMF type 110 look-alike records produced by OMEGAMON. Finally, it processes SMF type 111 records generated by IBM's CICS Transaction Gateway to provide information useful for monitoring Gateway performance and throughput.

The CA MICS Analyzer Option for CICS offers a unified and easily understood approach to CICS management by providing the information you need for the systematic planning, control, and evaluation of CICS. It collects data at the system and user activity level and writes both detail and summarized information to the CA MICS database. CA MICS automatically integrates CICS information with other information that CA MICS maintains. This is possible because the CA MICS Analyzer Option for CICS stores information in a format that is common to all CA MICS components.

You can use the CA MICS Analyzer Option for CICS to track user-based problems, to perform detailed analyses of CICS use, and to display information for selected CICS users or user groups. It provides a set of standard management objective reports that you can tailor to meet the needs of all management levels, and a series of summary reports that provide daily CICS activity and resource usage summaries at the user level.

The CA MICS Analyzer Option for CICS is one of the many data integration applications of CA MICS. Like the other data integration applications, it processes raw data for analyzing and managing specific technologies.

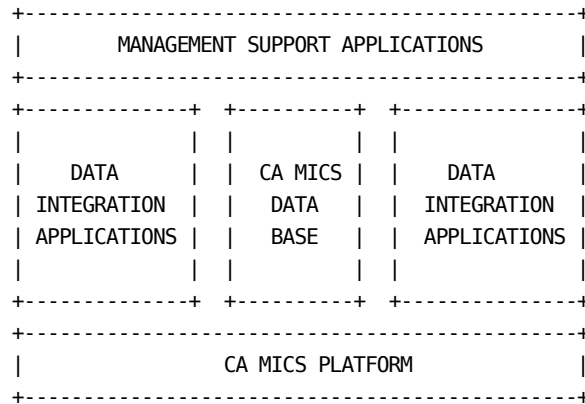


Figure 1-1. CA MICS

CA MICS is a comprehensive, flexible application system that applies standard management practices to the I/S organization. Designed with a formal architecture, CA MICS uses integrated applications analogous to the integrated financial applications that are now indispensable to corporate financial management.

This section contains the following topics:

- [1.1 Primary Areas of Application](#) (see page 13)
- [1.2 Major Features](#) (see page 15)
- [1.3 Reporting and Inquiry Facilities](#) (see page 19)
- [1.4 Files Overview](#) (see page 26)
- [1.5 Product Prerequisites](#) (see page 28)
- [1.6 Benefits](#) (see page 30)

1.1 Primary Areas of Application

The CA MICS Analyzer Option for CICS provides analysis and reporting capabilities that aid in CICS management in the following areas:

- o Performance Management - The CA MICS Analyzer Option for CICS provides multiple files that you can use to track all aspects of performance activity under CICS control. These files include information on the activity of a single transaction, multiple transactions that belong to a specific grouping (e.g., Payroll), a single CICS region, or multiple CICS regions on multiple system images. Using these files, a performance analyst can identify and correct specific problems, resulting in significant improvement in system performance.
- o Service Levels - The I/S manager can use CA MICS Analyzer Option for CICS data to group application response times in user-specified categories, either to assist in developing new service levels or to report on those already in place. This information is particularly useful in an MRO environment where overall service levels can be tracked across multiple CICS regions.
- o Capacity Planning - The CA MICS Analyzer Option for CICS provides a historical database that a capacity planner can use to predict long-term trends and to perform CICS consumption analyses. The data covers every aspect of CICS capacity planning from CPU consumption within transaction groups to analysis of multi-engine CPU requirements. Using this information, the analyst can make judgements concerning the need for capacity increases, multiple CICS regions for CPU and virtual storage constraint relief, or additional peripheral devices to support increased I/O rates.
- o Management Reporting - After establishing system and user service objectives such as response time and availability, management can use CA MICS Analyzer Option for CICS reports to compare actual service against these objectives. You can obtain this information on single-page management summaries that display the data by system, application unit, users, or groups of users.
- o CICS Network Management - The CA MICS Analyzer Option for CICS provides information on general CICS network activity and specific terminal usage through user-defined account codes. You can use this information to track network

operation, identify problems, evaluate performance, and plan for future requirements.

For CICS Transaction Gateway (CTG), the CA MICS Analyzer Option for CICS provides analysis and reporting capabilities that aid in CTG management in the following areas:

- o Performance Management - The CICS Transaction Gateway information area provides the Gateway Server Activity (CTGGSA) file that you can use to track the performance and response times for both the Gateway daemon and CICS as transactions flow through the CTG. Numerous metrics are available to help identify where specified configuration limits may be contributing to throughput degradation.
- o Capacity Planning - The CTGGSA file provides numerous metrics that a capacity planner can use to monitor trends in transaction rates, response times, and error conditions. The CTG is highly configurable, and by monitoring trends in key performance indicators, action can be taken to prevent service degradation.
- o Management Reporting - After establishing system objectives such as response time and availability, management can use reports to compare actual service against these objectives.

1.2 Major Features

The major features of the CA MICS Analyzer Option for CICS can be grouped into four categories:

- o Report facilities
- o Use of CICS data
- o General product flexibility
- o Integration capabilities

REPORT FACILITIES

The CA MICS Analyzer Option for CICS provides the following reports:

- o Management Reports are run on a daily, weekly, and/or monthly basis as part of the standard CA MICS processing. These reports summarize the activities of the I/S organization and are designed to help track CICS service and performance, monitor the operations configuration, and plan for future resource requirements.
- o Standard Analysis Reports provide concise information in the form of reports, graphs, charts, and plots. Typically, these reports are produced on an as-needed basis when the data is not in a form suitable for management or exception reports, or when a more in-depth analysis of the data is required.
- o Exception Analyzer Reports allow you to define, capture, and report on conditions that deviate from the expected norm.
- o Ad Hoc Reporting Facilities allow you to access information in the CA MICS database, either interactively or in batch, via the CA MICS Information Center Facility (MICF), a panel-oriented productivity tool. Another online tool, the CA MICS Workstation Facility (MMF), allows those familiar with the advanced analysis language provided as part of the SAS program product to use SAS either interactively or in batch mode.

USE OF CICS DATA

Another feature of the CA MICS Analyzer Option for CICS is its comprehensive handling of CICS data:

- o It consolidates data from multiple CICS regions, multiple monitoring data sources, and multiple data centers into an information database that you can use to develop a common method for reporting CICS activity.
- o It calculates and derives additional measurement statistics from the input data to further characterize CICS performance, service, and utilization. It groups and computes response measures and distributions based on the transaction classifications that you define.
- o It supports the CMF Data Dictionary facility to reduce sensitivity to IBM alteration of the CMF data format.
- o It consolidates usage and resource consumption data at both the system and user levels. The CA MICS Analyzer Option for CICS also consolidates terminal activity and internal response measurement data.
- o It supports a generic system identifier. This ensures that all data is representative of the processor on which CICS executes, even when you must migrate CICS regions between processors during planned or emergency situations.
- o It interprets encoded values in the CICS input data so that information is immediately usable in a logical form, without the need for further conversion or translation.
- o It supports CICS incident tracking, which records exceptional conditions in the detail data.
- o It supports user area, clocks, and counters.
- o It supports ASG-TMON file segments in the transaction data to enable you to track CICS files and database activities.
- o It supports user areas provided by OMEGAMON II for CICS to enable tracking of DB2, DL/I, and third party vendor database activities.

GENERAL PRODUCT FLEXIBILITY

The CA MICS Analyzer Option for CICS offers a number of flexibility features:

- o Usage guidelines for applying CA MICS Analyzer Option for CICS management information.
- o Cost center (organizational) classification that allows each organization to link individual user activities with the responsible organizational unit.
- o Application unit classification that allows an application to report transaction data. You can define this data by certain identifiers such as terminal, transaction code, or operator identifier, and then report on it using business applications, such as payroll.
- o Relative longevity classification to identify each transaction as short, medium, long, or conversational. The CA MICS Analyzer Option for CICS uses this classification to derive response measurement distributions to show the service provided for the above types of work.
- o Response limit specification to define the seven thresholds that are used for the response distribution calculations. The CA MICS Analyzer Option for CICS uses this definition, along with the relative longevity classification, to derive response statistics to report the percentage of work that was serviced within each response limit (for example, 93% of short transactions were serviced within three seconds).
- o Standard exits to tailor the product to your site's specific needs.
- o User-defined input data error tolerance to allow uninterrupted input processing. You can adjust tolerance levels either to accept the small number of errors that are normally found in input data or to accommodate a unique situation that may require a higher degree of input data error tolerance.
- o A data dictionary that describes the elements in the CA MICS CICS Information Area files and how those elements were derived.

INTEGRATION CAPABILITIES

The CA MICS Analyzer Option for CICS integrates its data into the CA MICS database, thus providing an interface with a

number of CA MICS components:

- o CA MICS Accounting and Chargeback Option. The CA MICS Analyzer Option for CICS provides data to CA MICS Accounting and Chargeback that supports accounting based on either consumed resources or transaction volume.
- o CA MICS Capacity Planner Option. The CA MICS Analyzer Option for CICS maintains a database of historical CICS usage and performance data that the CA MICS Capacity Planner can use to make growth/trend projections at both the system and user levels.

1.3 Reporting and Inquiry Facilities

The CA MICS Analyzer Option for CICS standard reports and online inquiries provide concise, comprehensive information to help you manage your CICS environment. It provides management reports, standard analysis reports, and exception reports that you can either use as delivered or tailor to your specific requirements. In addition, you may use MICF or the SAS language interface to create on-demand reports.

Each type of report and reporting facility is described below.

MANAGEMENT REPORTS

The management reports that are distributed with the CA MICS Analyzer Option for CICS provide a concise graphical or tabular representation of the data center's processing objectives and how well they have been met. You define the objectives through a series of parameters, then produce the reports during normal daily, weekly, or monthly CA MICS processing. You can activate or deactivate the reports for any given timespan (daily, weekly, or monthly).

The management reports described in the following chart are distributed with the CA MICS Analyzer Option for CICS.

Management Reports

Report	Function
Daily CICS Service Report - All Activity	Shows the CICS service provided for all activity and identifies the hours during the day when a CICS system fails to meet user-defined service objective.
Daily CICS Service Report - Medium Activity	Shows the CICS service provided for medium activity and identifies the hours during the day when a CICS system fails to meet user-defined service objective.
Daily CICS Service Report - Short Activity	Shows the CICS service provided for short activity and identifies the hours during the day when a CICS system fails to meet user-defined service objective.

Daily CICS Outage Report-Availability	Shows CICS availability to quantify the amount of downtime on an hourly basis.
Daily CICS Throughput Report	Shows CICS transaction volume on an hourly basis and identifies peak hour workload for each CICS system.
Weekly CICS Service Report - All Activity	Shows the CICS service provided for all activity during the previous seven-day period (excluding weekends) and identifies any hours during the day when a CICS system fails to meet the user-defined service objective.
Weekly CICS Service Report - Medium Activity	Shows the CICS service provided for medium activity during the previous seven-day period (excluding weekends) and identifies any hours during the day when a CICS system fails to meet the user-defined service objective.
Weekly CICS Service Report - Short Activity	Shows the CICS service provided for short activity during the previous seven-day period (excluding weekends) and identifies any hours during the day when a CICS system fails to meet the user-defined service objective.
Weekly CICS Outage Report-Availability	Shows CICS availability in the previous seven-day period (excluding weekends) to quantify the amount of downtime incurred on an hourly basis.
Weekly CICS Throughput Report	Shows CICS transaction volume on an hourly basis for the previous seven-day period (excluding weekends) and identifies peak hour workload for each CICS system.
Monthly CICS Service Report - All Activity	Shows the CICS service provided for all activity during the previous six months by zone.
Monthly CICS Service Report - Medium Activity	Shows the CICS service provided for medium activity during the previous six months by zone.
Monthly CICS Service Report - Short Activity	Shows the CICS service provided for short activity during the previous six months by zone.

Short Activity	months by zone.
Monthly CICS Outage Report-Availability	Shows CICS availability during the previous six months by zone.
Monthly CICS Throughput Report	Shows CICS transaction volume during the previous six months by zone.

STANDARD ANALYSIS REPORTS

Standard analysis reports are produced on an as-needed basis. They provide concise CICS activity information in the form of reports, graphs, charts, and plots. The standard analysis reports described in the following chart are distributed with the CA MICS Analyzer Option for CICS.

Report	Function
Daily Activity Overview Report	Provides a two-page report for each CICS system that CA MICS processes. The first page gives an Hourly Activity Profile and the second summarizes response time distributions for the day.
Application Unit: Summary Report	Provides the averages and totals of resources used for each user-defined application unit.
Daily User Activity Summary Report	Provides the averages and totals of resources used for each user-defined first-level accounting field.
File/DBD Activity Summary Report	Provides summarized information for each file used by CICS and quantifies the type of files, the types of requests made to the files, and any VSAM string waits that have occurred. This report is available for use only with input from ASG-TMON.

Standard analysis reports include a series of predefined MICF inquiries. These inquiries, like the reports listed above,

are designed to be run on an as-needed basis. You can also use them as templates for designing your own inquiries.

The MICF inquiries described in the following chart are distributed with the CA MICS Analyzer Option for CICS.

Inquiry	Function
Daily CICS System Overhead Analysis	Displays the percentage of CPU time spent in system overhead.
8 am - 5 pm CICS Service by Type	Displays prime time hourly CICS response time by transaction type and identifies any hours when CICS service falls below a user-defined management objective.
Daily CICS Service	Displays CICS response time and identifies the hours during the day when service falls below a user-defined management objective.
Weekly CICS Service	Displays the CICS response time trend in a seven-day period (excluding weekends) and identifies any hour between 8 a.m. and 5 p.m. when service fell below a user-defined management objective.
Monthly CICS Service	Displays the CICS response time trend in a six-month period and identifies zones in which service fell below a user-defined management objective.
Daily CICS Workload	Displays CICS transaction volumes on a daily basis and identifies peak hour workloads for each CICS system.
Weekly CICS Workload	Displays the CICS transaction volume trends in a seven-day period (excluding weekends) and identifies peak hour workloads between 8 a.m. and 5 p.m. for each CICS system.
Monthly CICS Workload	Displays the CICS transaction volume trends in a six-month period and identifies peak zone workloads for each CICS system.

Daily CICS Availability	Displays CICS availability by hour on a daily basis.
Weekly CICS Availability	Displays CICS availability between 8 a.m. and 5 p.m. over a seven-day period (excluding weekends)
Monthly CICS Availability	Displays CICS availability by zones over a six-month period.

EXCEPTION ANALYSIS REPORTS

The CA MICS Platform provides an Exception Analyzer that works with CA MICS Data Integration Applications to provide consolidated reporting of exception conditions across systems and data sources.

An "exception" is any condition that deviates from the expected norm, or any missed objective in system performance or service.

The CA MICS Analyzer Option for CICS is delivered with several exception tests, and you can easily add your own. The exception reports produce increasing levels of detail to address the needs of different audiences and, when combined with exceptions from other CA MICS applications, comprise a powerful diagnostic tool for your organization.

Report	Function
Exception Management Overview	Provides a concise, high-level summary of the exceptions reported for the previous day. The report lists, in terms of severity, the number of exceptions for each defined management area. Designed for use by upper management, it allows quick assessment of the operation's stability.
Management Area Exception	Provides an hourly summary of exceptions encountered for a specific management area (for example, performance or availability), and the number and severity of

each exception type that occurred during an hour. Typically, first-level management personnel who are responsible for defined management areas use this inquiry.

Severity Level
Exception

Provides an hourly summary of exceptions encountered for a given severity level, thus allowing quick assessment of the different types of exceptions in each level. It gives first-level managers, performance analysts, and system programmers an integrated report of the problems that may have affected the data center in any given hour.

Detail Exception

Provides a detailed list, in order of occurrence, of detected exceptions. You can selectively produce this report online using a number of different selection criteria, including a date and time range, severity level, management area, and so on. You can use it to obtain the necessary background detail for effective analysis of the reported exceptions.

AD HOC REPORTING FACILITIES

The specific types of information needed to manage a large I/S organization change daily. CA MICS addresses those needs with interactive capabilities that allow fast response to the most complex requests for information. The interactive capabilities are:

- o The CA MICS Information Center Facility (MICF) is a panel-oriented productivity tool that allows you to access information in the CA MICS database. When you define input, selection criteria, and report options, MICF fulfills your request by executing the program either in batch or interactive mode. Then, according to your specifications, MICF either prints the results of your request, displays the results at your terminal, or catalogs the results for later viewing.
- o The CA MICS Workstation Facility (MWF) provides an online environment in which SAS can be used both interactively or

in batch mode. CA MICS uses the SAS system for data management and as an advanced analysis language.

The Standard SAS Language is an advanced analysis language that enhances your staff's analytical capabilities and improves the quality of their decision making. CA MICS uses SAS, a powerful fourth generation language that provides easy data manipulation and statistical analysis. SAS also supports coding facilities for programmers who are conducting extensive analysis or designing new reports for I/S business applications.

1.4 Files Overview

The CA MICS Analyzer Option for CICS uses a number of different input records to build the CICS Information Area (CIC) which, in turn, maintains system and user measurement data on service, availability, load, and user activity. This section defines the use and content of each CA MICS file in the CICS Information Area.

- o System Activity File (CICCSY)

The CICCSY file quantifies total activity for the entire CICS system for all users and provides resource consumption, service, availability, and performance measures. This file is derived, at the DETAIL timespan, from the CMF global records (pre-CICS 3.1.1), the CICS statistics records (CICS 3.1.1 and higher), and ASG-TMON system records. At the DAYS and higher timespans, this file contains information from CMF and ASG-TMON.

- o User Activity File (CICCSU)

The CICCSU file quantifies the resources consumed at the user level using service, load, access, and performance measures that are provided based on user-defined account codes. This file is derived from CMF and ASG-TMON.

- o Application Unit Activity File (CICCAU)

The CICCAU file quantifies the resources consumed at the application level using service, load, access, and performance measures that are provided based on user-defined application unit identifiers. This file is derived from CMF and ASG-TMON.

- o User Application Count File (CICCAC)

The CICCAC file quantifies the number of transactions executed at the user and application unit levels. This file is derived from CMF and ASG-TMON.

- o Incident File (CICGIN)

The CICGIN file quantifies incidents that relate to the operation, performance, or integrity of the CICS system. This file is derived from CMF and ASG-TMON.

- o Dictionary File (CICCDC)

The CICCDC file contains data dictionary information from CMF performance class and pre-CICS 3.1.1 accounting class and exception class data. CICS 3.1.1 and higher releases only produces dictionary records for performance class data. CA MICS requires dictionary records in order to process the CMF data.

- o File/DBD Activity File (CICCSF)

The CICCSF file quantifies file activity. It is derived from the optional file segments of ASG-TMON transaction records.

- o MRO Activity File (CICCMR)

The CICMRO file quantifies MRO activity. It is derived from the optional MRO segments of ASG-TMON transaction records.

Figure 1-2 illustrates the data sources and files of the CICS Information Area.

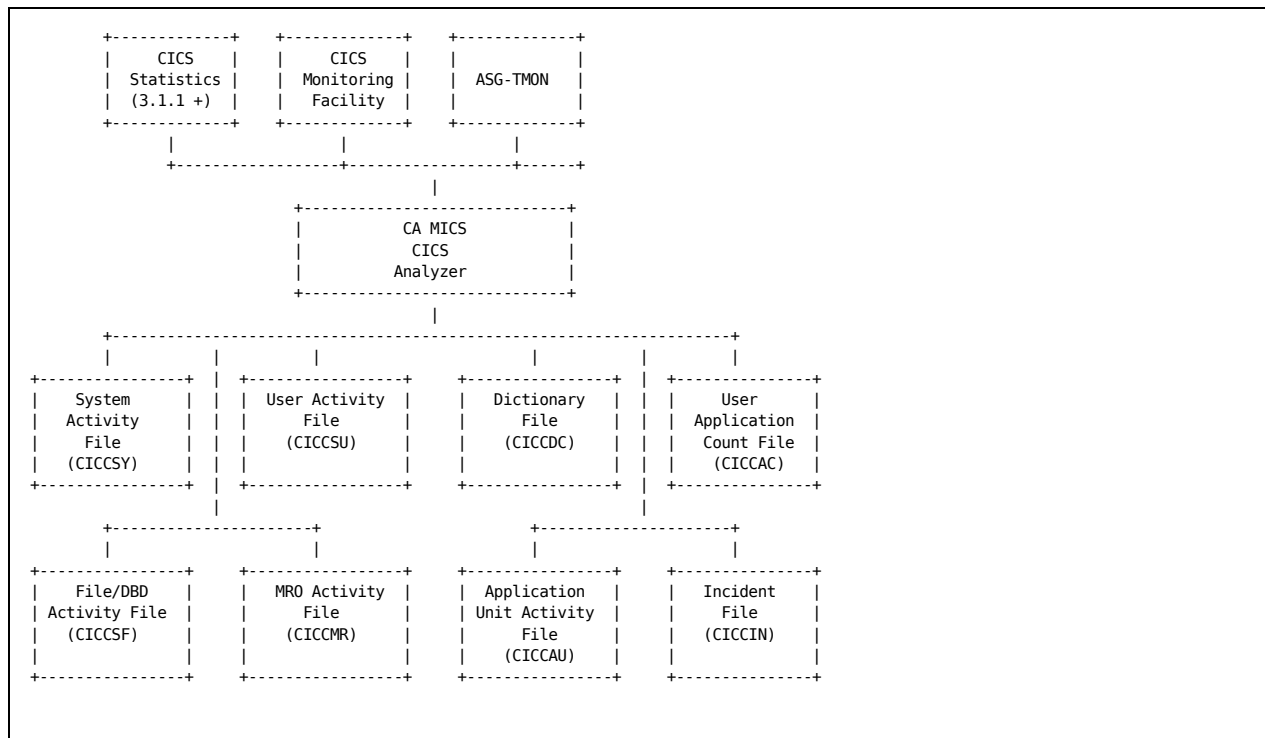


Figure 1-2. CICS Information Area Structure

1.5 Product Prerequisites

The CA MICS CICS Analyzer runs in MVS-based environments such as OS/390 and Z/OS. It supports the following data sources:

- o IBM's CICS Monitoring Facility (CMF), CTS 2.2 through CTS 5.1

CICS CMF monitoring data is written using the SMF 110 record type with the following subtypes:

- 0 - CICS journaling
- 1 - CICS monitoring
- 2 - CICS statistics
- 3 - Shared temporary storage queue server
- 4 - Coupling facility data table server statistics
- 5 - Named counter sequence number server statistics

The CA MICS Analyzer Option for CICS uses only the subtype 1 and 2 records currently. Subtypes 0, 3, 4, and 5 are not used.

Within the different subtypes, there are one or more classes of data that is produced, depending on the requirements. Following are the classes of data that are currently available within the CICS monitoring record (subtype 1):

- 1 - Dictionary data
- 2 - unused
- 3 - Performance data
- 4 - Exception data
- 5 - Transaction Resource data

The primary input is the subtype 1 (monitoring) record, class 1 dictionary data, and class 3 performance data. Processing for the class 4 exception data is optional.

- o CICS Statistics Records

For all supported releases, the CA MICS Analyzer Option for CICS processes selected CICS statistics that are written to SMF type 110, subtype 1, class 1 record.

For CTS 2.x and CTS 3.1, statistics record types 2, 6, 10, 30, 45, 48 and 60 are processed. For CTS 3.2 and CTS 4.1, statistics record types 6, 10, 14, 30, 45, 48, and 60 are processed. For CTS 4.2 statistics record types 6, 10, 29, 30, 45, 48, and 60 are processed. For CTS 5.1, statistics record types 6, 10, 29, 30, 45, 48, and 62 are processed.

- o ASG-TMON for CICS TS (TCE) through Release 3.3

The CA MICS Analyzer Option for CICS provides support for the Region Interval (TR) record in addition to Transaction Performance activity (TA) and Transaction Performance History interval (TI) records.

- o CMF type 110 look-alike records

CMF type 110 look-alike records produced by products such as OMEGAMON are also supported, when they are compatible with IBM's CMF data at a given CICS release level. Currently, the CA MICS Analyzer Option for CICS supports DBCTL for IMS usage reporting and OMEGAMON for general performance reporting only. If more data elements are introduced, then these elements must be defined within the \$CICGENIN member and an appropriate exit must be used to process the input data. For more information, see section 10.1.15.

1.6 Benefits

The CICS Analyzer contributes to the overall benefits that CA MICS provides. The CICS Analyzer contributes to these benefits by:

- o Improving Day-to-day I/S Management

It allows managers to track usage, identify system bottlenecks, and plan corrective actions to ensure smooth operations. It provides reliable information for setting service-level objectives.

- o Reducing Risk to the Enterprise

It allows I/S to define exceptional conditions and to assess their impact.

- o Controlling and/or Helping Reduce Costs

It allows you to equitably charge users for the amount of resources they use.

- o Improving Return on the I/S Investment

It allows managers to gauge the quality of service being provided.

- o Increasing Productivity

It allows less-experienced users to interpret report results and use database information. It reduces the task of coordinating information from multiple sources so you can report or analyze information at the enterprise rather than at the system level.

- o Improving Planning

It allows managers to anticipate problems before they affect the system, identifies areas for future CICS user expansion, and assists in projecting growth trends.

Chapter 2: USAGE GUIDELINES

This chapter describes guidelines for the interpretation and application of the data made available by the CA MICS Analyzer Option for CICS. It presents an overview of the scope and general areas of applications of the CA MICS Analyzer Option for CICS and a discussion of CICS data source concepts and terminology as they relate to the available data in the CA MICS database.

In addition, this chapter describes the interfaces between the CA MICS Analyzer Option for CICS and other CA MICS components.

About CICS

CICS is a major data-communication software system from IBM. As such, it provides an online information processing system, using terminals that can directly access data sets and databases. Its major functions provide communication interfaces among remote and local terminals and subsystems, concurrent processing of multiple online users, file servers that interface with IMS and DB2 database management systems, and the ability to communicate with other CICS and database systems, both within the same computer and across computers.

CICS provides its own facility, known as the CICS Monitoring Facility (CMF), to capture performance-related data during online processing for later batch analysis. CMF data is one of the data sources supported by the CA MICS Analyzer Option for CICS. In addition, third-party vendors have provided their own CICS monitors, which produce data that is also supported by the CA MICS Analyzer Option for CICS. One is ASG-TMON.

These CICS monitors capture detailed information at the transaction level so you can analyze the type of service CICS is providing its users. In addition, they capture system-wide information to provide an overall picture of CICS resource consumption and system performance. The CA MICS Analyzer Option for CICS provides you with the tool to analyze this vital information, which is used to assess the productivity and satisfaction of your CICS users and to monitor the effective use of your hardware resources.

About the CA MICS Analyzer Option for CICS

The CA MICS Analyzer Option for CICS helps you manage your CICS environment by:

- o Providing a common database to support the various CICS monitors.
- o Integrating system level data with transaction data in a common file to show total CICS activity.
- o Supplementing the CICS monitor data with derived and user-defined workload characterization data to increase usability of the data sources.
- o Producing management reports and summary reports to monitor various aspects of CICS performance, such as

system throughput and resource usage.

- o Providing exception analysis using CA MICS exception monitoring and reporting to alert you to problem conditions.
- o Providing history data for monitoring performance trends and planning for future growth.

Three broad categories of data are produced by the CA MICS Analyzer Option for CICS:

- o Workload and utilization data
- o Response/system performance data
- o Exception data

Workload and utilization data consists of transaction counts, CPU usage, file access, message counts, storage usage, calls to various CICS facilities such as journal control and temporary storage, and so on. The CA MICS Analyzer Option for CICS summarizes this information based on characterization keys that you define when the product is installed. The summarized information enables you to monitor, on a regular basis, the workload generated by a specific "user," such as a department, a terminal user, or an application. In addition, the CA MICS Analyzer Option for CICS provides historical data to help you identify workload trends and estimate future growth.

CICS response-time data provides the key indication of CICS performance. Ideally, this data should represent end-user response time, which measures from the time the terminal user presses the Enter key to the time CICS displays the response on the screen and unlocks the terminal keyboard. However, none of the CICS monitors supported by CA MICS provides end-user response time. As a result, the response-time measures produced by the CA MICS Analyzer Option for CICS represent the internal or host response time, which is calculated based on transaction start and end timestamps. The CA MICS Analyzer Option for CICS also calculates average response times for each type of transaction, including short, medium, long, and conversational. In addition, it produces response distributions by transaction type to show the percentage of transactions that completed within each of the response limits that you specify.

The system performance data identifies problem conditions that may degrade CICS performance, such as short-on-storage (SOS) conditions, frequent program compressions, maximum tasks conditions, and storage accounting area (SAA) errors. By monitoring this information on a regular basis, you can identify CICS system performance issues and address them accordingly.

Exception data alerts you to abnormal conditions and events that exceed site-specified limits, for example, transaction abends and excessive paging rates. This data enables you to quickly identify problem areas that need to be addressed.

This section contains the following topics:

[2.1 Work File Compression](#) (see page 35)

[2.2 Data Source Background](#) (see page 36)

[2.3 Data Analysis Guidelines](#) (see page 49)

[2.4 Accounting and Chargeback Interface](#) (see page 65)

[2.5 Capacity Planner Interface](#) (see page 67)

2.1 Work File Compression

The SAS system gives you the option to create variable-length or fixed-length observations in a SAS data set. Variable-length observations differ from fixed-length observations in that the former are usually smaller because the blank spaces used to pad fixed-length observations are removed.

You instruct SAS to create variable length observations by specifying the COMPRESS= option. SAS data set compression can be implemented for individual data sets or across the entire SAS system by specifying COMPRESS= on either a DATA statement (for the individual data set named on the DATA statement) or an OPTIONS statement (for the entire system).

For more information on the COMPRESS= option, see the SAS Institute documentation.

In deciding whether or not to implement compression for a SAS data set, CA's research indicates the following:

- o Compressed data sets generally require fewer IOs than uncompressed data sets.
- o Observations in a compressed data set cannot be accessed by observation number.
- o Compressed data sets use more TCB CPU time than uncompressed data sets. (TCB time is the amount of time spent executing application code.)

The CICS Analyzer supports compression for files in the CA MICS data base as well as the work files used during daily operational processing. If you choose to implement data compression for the CICS Analyzer's work files, elapsed times may be prolonged in the DAY040 step of the DAILY job.

2.2 Data Source Background

This section presents CICS background information to provide you with a framework for the application of the data produced by the CICS Analyzer. Only those CICS terminology and data source concepts that are relevant to the CA MICS CICS Analyzer are discussed.

The following topics are presented:

- 1 - Transaction vs. Task vs. Interaction
- 2 - MRO and ISC Environment
- 3 - CICS Response Time
- 4 - CICS CPU Time

2.2.1 Transaction vs. Task vs. Interaction

The unit of work performed under CICS, or any online system, is dependent on the perspective of the person defining the unit. The following definitions apply for the CA MICS CICS Analyzer:

- o Interaction

An interaction is one piece of a conversation between the terminal user and CICS. An interaction begins when the user enters a request to CICS from the terminal. It ends when CICS displays the results of the user's request on the terminal.

- o Transaction

The transaction is a unit of work that consists of zero, one, or more interactions between the user and CICS. Typically, a transaction results in a single reply being sent back to the user's terminal; it ends after the reply is issued and does not require a second input from the terminal. Other times, a transaction may involve several interactions between the terminal user and CICS, in which case the transaction is known as a "conversational" transaction. In some cases, a transaction is started without input from a terminal. Such transactions consist of zero interactions.

- o Task

A task is the CICS internal representation of a transaction. It begins when CICS associates a transaction with a control block called the Task Control Area (TCA) and ends when the TCA is no longer in existence. Depending on the types of processing, a transaction may require the initiation of one or more tasks.

CMF and ASG-TMON provide data at the transaction level. However, there are some differences between CMF and the others in recording data for conversational tasks.

For ASG-TMON, a conversational task automatically generates multiple transaction records. Thus, both conversational and nonconversational tasks are reported on the same basis; one record is written for each interaction.

For CMF, a conversational task can generate one or more

performance class transaction records depending upon the parameter that you specify. To get a separate CMF record for each interaction of the conversational task, you must specify the SIT parameter MNCONV=YES for CICS TS 1.2 and higher releases, or CONV=YES in the DFHMCT macro in pre-CICS TS releases. The default value is NO for all CICS releases, which means CMF produces one transaction record for the entire conversational task.

2.2.2 ISC and MRO Environment

In today's increasingly complex online transaction processing (OLTP) environments, CICS no longer exists as a single, isolated system. The methods by which CICS communicates with other mainframe, CICS, or IMS systems are intersystem communication (ISC) and multiregion operation (MRO). Collectively, ISC and MRO are known as CICS intercommunication facilities. This section provides some general information and terminology for these facilities, plus a look at how ISC and MRO affect the contents of the data sources processed by CA MICS.

ISC provides communication between systems that are in the same or different hosts via an SNA (System Network Architecture) access method such as ACF/VTAM (Advanced Communication Facility/Virtual Telecommunications Access Method), which provides communication protocols for the interconnected systems. ISC uses two types of SNA protocols, including Logical Unit Type 6 (LU 6.1) and Advanced Program-to-Program Communication (APPC, or LU 6.2).

MRO provides CICS-to-CICS communication independent of SNA access methods such as ACF/VTAM. In earlier CICS releases, MRO is used for CICS communication within the same host (single MVS image). Beginning with CICS/ESA 4.1 and MVS/ESA 5.1, MRO can be used between CICS systems in the same MVS system complex (sysplex), which comprises multiple MVS system images. The CICS internal support that enables MRO is called interregion communication (IRC), which can be implemented through a type 3 supervisory call (SVC), MVS cross-memory services, or the cross-system coupling facility (XCF) of MVS/ESA (5.1 and higher).

In a simple ISC/MRO configuration, the interconnected systems typically consist of a primary region and one or more secondary regions. The primary region owns the terminals to which users log on and is called the terminal-owning region (TOR). The secondary regions usually own the application programs and files and are called application-owning regions (AOR). When a secondary region owns the files and data needed to support the applications, it is known as a file-owning region (FOR).

The terms TOR, AOR, and FOR do not necessarily imply that the primary region must own all of the terminals or that the secondary regions must own all applications and files, although that is a possible configuration for many sites.

ISC and MRO facilitate the following types of processing:

- o Transaction routing - Allows terminal users in a given system to run transactions in any connected CICS system. Under transaction routing, the transaction is entered on the TOR and then routed by the CICS relay program to an AOR for execution.
- o Distributed transaction processing (DTP) - Provides a technique to distribute work over several transaction programs within a network. DTP is a form of synchronous processing in which the transaction is coded specifically for multiple regions to perform joint work.
- o Function shipping - Allows multiple CICS regions to share resources. Under function shipping, a transaction can execute within one CICS system and request a resource that resides in another system. The request is "shipped" to the other system for execution by a function called a "mirror transaction" (CSMx transaction codes). Once the request has been processed by the mirror transaction, it is shipped back to the system from which the request originated.
- o Asynchronous processing - A variation of function shipping, where the completion of the primary transaction is not dependent on the state of the secondary transaction it initiates.
- o External CICS interface (EXCI) - An application programming interface (API) that enables a non-CICS program to invoke a server program running in CICS. EXCI is available in CICS TS 1.2 and higher releases.

Figure 2-1 shows an example of CICS regions connected via MRO with one TOR, one AOR, and one FOR.

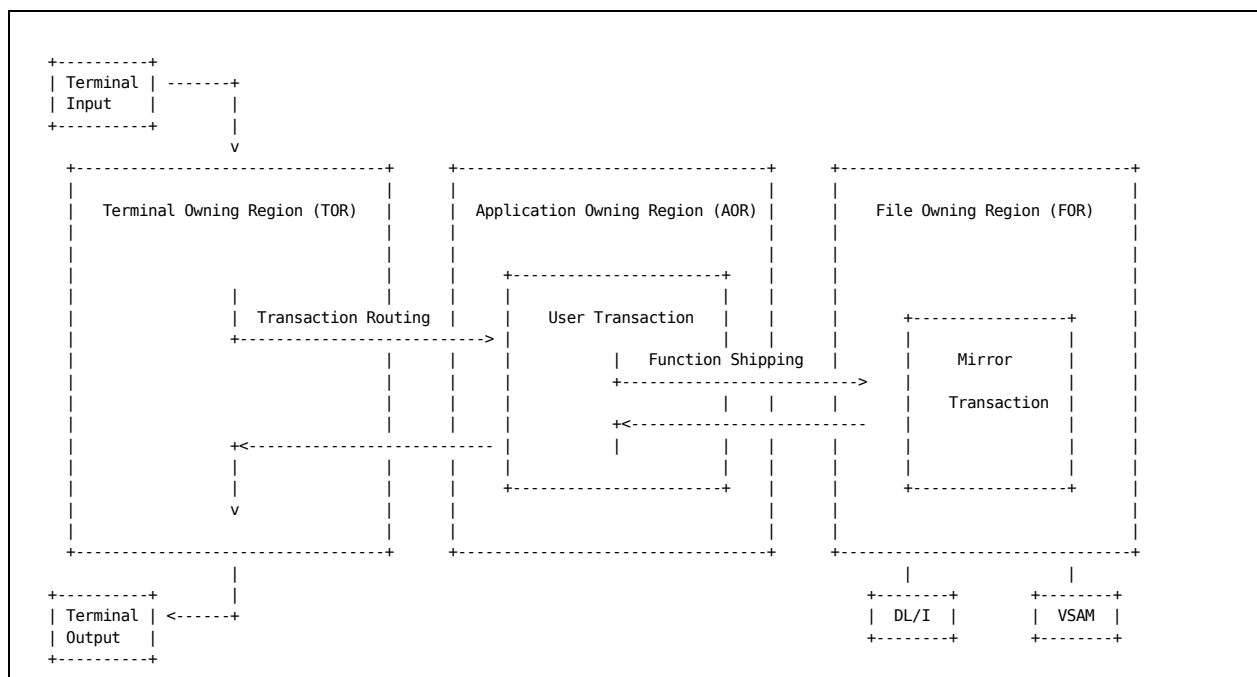


Figure 2-1. CICS Regions Connected Via MRO

For MRO and ISC environments, all of the CICS monitors supported by CA MICS generate a transaction record in each of the CICS regions where a piece of the transaction is executed. For example, under transaction routing, the TOR and AOR each write a record for their processing of the transaction. The TOR record contains terminal related information while the AOR record contains program resource usage information.

To identify and link the pieces of information written by each region back to the same transaction, CICS assigns a token to each transaction and passes the token to subsequent regions where the transaction is executed. This token, known as the unit of work ID (UOWID), is a time value derived from the initial task attach time. Along with the UOWID, CICS also passes the NETNAME (the name by which CICS is known to VTAM, or, in DL/I, the jobname.stepname.procname) of the originating system. Together, UOWID and NETNAME form a unique identifier by which related transaction records can be matched to provide a total view of a transaction's activities across CICS systems.

The CICS Analyzer processes transaction records from each CICS region individually; therefore, you will see multiple transaction records in your CA MICS data base for a single MRO/ISC transaction. The CICS Analyzer does not consolidate or collapse the related transaction records into a single observation in the CA MICS data base. However, for accounting purposes, you have the option to invoke a user exit during detail transaction record processing to pass accounting-related information from the TOR records, such as terminal ID, to the AOR and FOR records. This optional CA MICS user exit, called the multi-system accounting exit (CICMSAC), enables you to summarize transaction records that have identical UOWID and NETNAME combinations under the same CA MICS account codes. For more information on the CICMSAC user exit, refer to Section 7.3.9 of this guide.

Figure 2-2 provides an example of the CMF records showing the activity of a transaction as it is routed from the TOR to the AOR and then function-shipped between the AOR and the FOR. In the figure, examples 1 and 2 show the CMF records produced for the basic update transaction AUPD. Example 3 shows CMF records produced for the browse transaction ABRW.

In example 1, transaction AUPD is entered on terminal SR01. The task number for this transaction is 45. Notice the program name is DFHCRP, which is the CICS program that performs the transaction routing function in MRO/ISC. The

next line provides information for transaction AUPD, transaction number 38, on terminal SR01. However, the CICS system ID is SAOR. This is the result of transaction routing to the AOR. The program shown here, DFH\$AALL, is the actual application program name for the AUPD transaction. The last line in example 1 is the result of the DFH\$AALL program issuing a request for a remote file, which is controlled by the CICS system SFOR, a file-owning region. The request is function-shipped to SFOR and is processed by the mirror transaction CSMI in that region. Notice that this is the only entry in example 1 which registers file activity.

Example 2 shows the second interaction of the transaction APUD. It is within the second series of activity that the actual file update takes place (note the get and put file activity in the record written for the FOR).

Example 3 depicts the browse transaction ABRW being routed to the SAOR region, where a request is then function-shipped to the SFOR region. Of interest in this example is the initial ABRW running in conversational mode. CMF in this CICS system is recording each segment of the conversation in a separate transaction record. Note the TRANTYPE T for the first record and the TRANTYPE C for subsequent records, and that the task number remains the same for the subsequent records.

CICS MRO/ISC TRANSACTION RECORD DUMP														
CICS SYSTEM	NETNAME	UOWID	TASK ID	TRAN CODE	TERM	PROGRAM	TRAN TYPE	DURATION	TRAN CPU	FILE BROW	FILE ADD	FILE DEL	FILE GET	FILE PUT
EXAMPLE 1														
STOR	STOR	3577:58:58.68	45	AUPD	SR01	DFHCRP	T	00:00:00.349	0.018352	0	0	0	0	0
SAOR	STOR	3577:58:58.68	38	AUPD		DFH\$AALL	T	00:00:00.302	0.066880	0	0	0	0	0
SFOR	STOR	3577:58:58.68	29	CSMI		DFHMIR	T	00:00:00.036	0.014416	0	0	0	1	0
EXAMPLE 2														
STOR	STOR	3577:12:59.44	46	AUPD	SR01	DFHCRP	T	00:00:00.211	0.018464	0	0	0	0	0
SAOR	STOR	3577:12:59.44	39	AUPD		DFH\$AALL	T	00:00:00.164	0.052096	0	0	0	0	0
SFOR	STOR	3577:12:59.44	30	CSMI		DFHMIR	T	00:00:00.113	0.027840	0	0	0	1	1
EXAMPLE 3 (not all interactions are shown)														
STOR	STOR	3574:23:10.70	41	ABRW	SR01	DFHCRP	T	00:00:02.555	0.01909	0	0	0	0	0
SAOR	STOR	3574:23:10.70	34	ABRW		DFH\$ABRW	T	00:00:44.677	1.78850	0	0	0	0	0
SFOR	STOR	3574:23:10.70	27	CSMI		DFHMIR	T	00:00:44.363	1.30104	135	0	0	1	1
STOR	STOR	3574:23:10.70	41	ABRW	SR01	DFHCRP	C	00:00:01.191	0.01418	0	0	0	0	0
STOR	STOR	3574:23:10.70	41	ABRW	SR01	DFHCRP	C	00:00:01.196	0.01435	0	0	0	0	0
STOR	STOR	3574:23:10.70	41	ABRW	SR01	DFHCRP	C	00:00:01.188	0.01416	0	0	0	0	0
STOR	STOR	3574:23:10.70	41	ABRW	SR01	DFHCRP	C	00:00:01.194	0.01430	0	0	0	0	0
STOR	STOR	3574:23:10.70	41	ABRW	SR01	DFHCRP	C	00:00:01.288	0.01437	0	0	0	0	0
STOR	STOR	3574:23:10.70	41	ABRW	SR01	DFHCRP	C	00:00:01.191	0.01440	0	0	0	0	0
STOR	STOR	3574:23:10.70	41	ABRW	SR01	DFHCRP	C	00:00:01.194	0.01424	0	0	0	0	0

Figure 2-2. CICS MRO/ISC Activity Example

2.2.3 CICS Response Time

The ideal measure of response time is the end-to-end transaction response time, which starts when the terminal user enters a request and ends when the result of the request is displayed at the terminal. Unfortunately, this type of response measure requires tracking all events occurring at many different points of the communication network. Therefore, it is not available from any of the CICS monitors. Response time in terms of CICS is the internal transaction elapsed time.

MEASURING CICS RESPONSE TIME

The CICS internal transaction response time is the time between the CICS initiation of a task and the CICS termination of the task. This measure includes two primary components: dispatch time and wait time. Dispatch time includes CPU time and the time CICS is involuntarily interrupted while the task is dispatched. Wait time includes the time CICS is in a wait state, such as dispatch queue wait, I/O waits, and VSAM file string waits. Certain wait states cause the CICS task to be suspended, the most notable of which is the wait for terminal input.

The CA MICS Analyzer Option for CICS normally derives response time based on the transaction start and end timestamps as provided by the CICS monitors. However, for conversational tasks, it further excludes user think time from the transaction response time to prevent skewing of the internal response measure. The actual response derivation is described below.

MEASURING CONVERSATIONAL TRANSACTION RESPONSES

A conversational task, as described earlier in this chapter, involves multiple interactions between the terminal user and CICS. To provide a more accurate response time for this type of transaction, some CICS monitors subtract the time the transaction is suspended in CICS while waiting for terminal input from the transaction elapsed time. ASG-TMON for CICS TS (TCE) excludes terminal wait time for conversational tasks by generating a transaction record for each segment (a pair of terminal I/Os) of the conversation. The same is true for CMF when the MNCONV=YES parameter is specified in the SIT (at CICS/ESA 4.1 and higher), or the CONV=YES parameter is specified in the DFHMCT TYPE=RECORD macro (pre-CICS/ESA 4.1). Otherwise, CMF writes one transaction record for the entire

conversational task, in which case the elapsed time includes user think time.

To ensure the CA MICS response derivation accurately reflects the internal response time, the CA MICS Analyzer Option for CICS subtracts the minimum of either terminal control wait time or suspend time from transaction elapsed time.

$$\text{internal response} = \text{end} - \text{start} - \text{MIN}(\text{suspend}, \text{TC wait})$$

Terminal control wait time includes the time waiting for terminal input, but it also includes time waiting for other terminal control services that might not cause the task to be suspended. Suspend time includes time the task waited for terminal input and any other suspend-eligible reasons. One frequent cause of task suspension is having CICS operate at the active max task limit. For this reason, suspension has nothing to do with terminal control.

Figure 2-3 shows the flow of control for one CICS transaction, including waits for transaction-required resources during processing. An example of such a requirement is a CICS Terminal Control read operation.

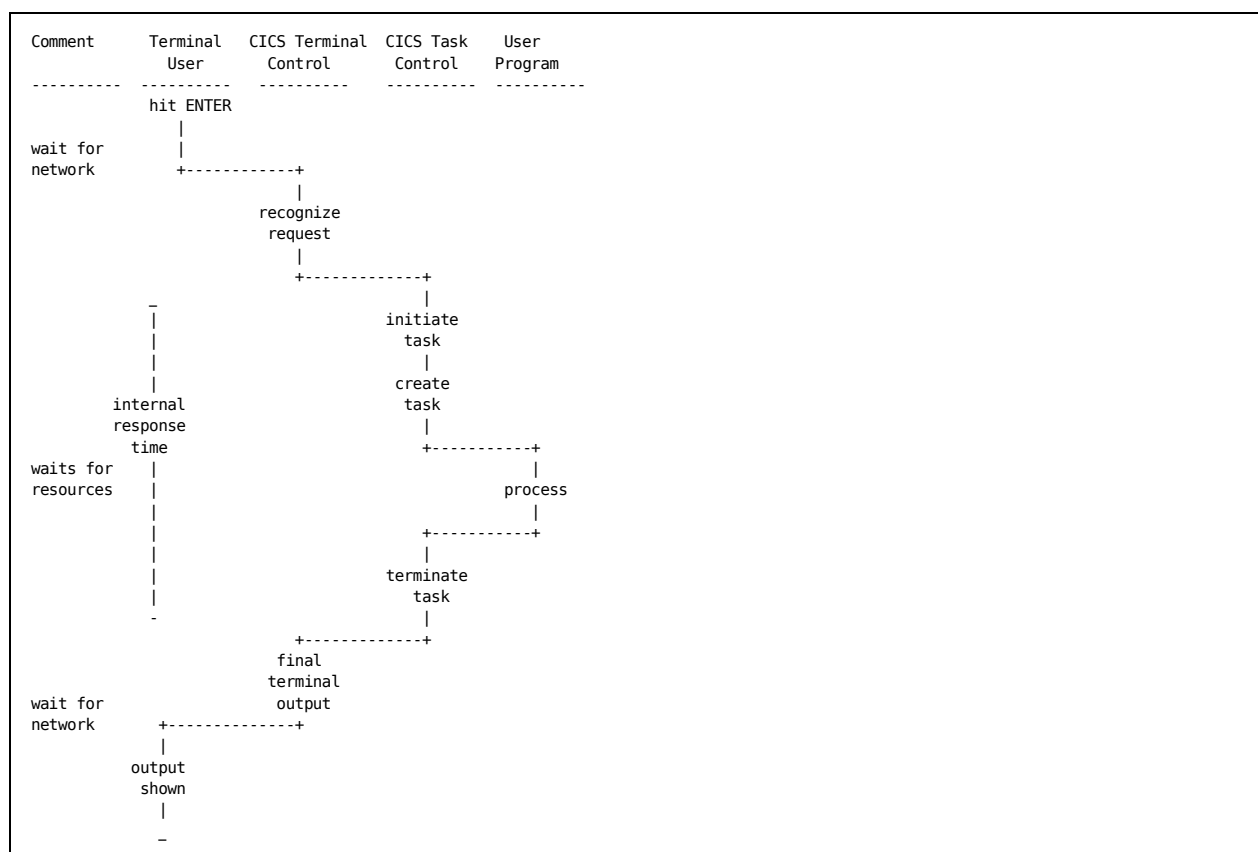


Figure 2-3. CICS Control Flow for a Transaction

2.2.4 CICS CPU Time

The CICS monitors supported by the CA MICS CICS Analyzer are event-driven rather than interval-driven. This means they collect data by intercepting control from CICS at many specific events during CICS processing. They do not sample CICS activity at fixed or random intervals and assume total system activity is some multiple of the observed activity. In effect, the monitors observe all CICS activity.

The supported monitors measure CPU time by calculating what amounts to the running total of CPU time used since CICS was started at each event. The difference between the current CPU time total and the total from the previous event is the amount of CPU time used since the last event. This amount is added into one of several accumulators (one for each division of CPU time measured), depending on the type of event the monitor just recognized. Most monitors calculate CPU time in two different ways: "dispatched" time and "real CPU" time.

DISPATCHED TIME AND CPU TIME CALCULATIONS

Dispatched time is the measure of CPU time that includes the amount of time CICS was involuntarily interrupted for any reason, such as page faults or interruption by tasks higher in the operating system's dispatching priority scheme.

CPU time is an accurate measure of the time during which CICS was in total control of the computer logic unit, executing instructions. It does not include the time CICS was involuntarily interrupted.

USING TWO MEASURES OF CPU TIME

The main reason for collecting both dispatched time and CPU time is that the ratio between the two can be useful. If, for example, the ratio of dispatched time to CPU time correlates with paging volume, CICS throughput degradations would likely be caused by paging activity. The correlation analyses can be performed with whatever subset of data is desired to narrow the search for the paging resource overload.

If the correlation of that ratio does not appear, or is of particularly low coefficient during certain periods of time, CICS response degradation would be caused by interference with other environmental factors within the operating system,

such as higher priority jobs.

2.3 Data Analysis Guidelines

This section generally describes the types of data available from the CA MICS CICS Analyzer and how it can be used to monitor CICS performance.

The following topics are presented:

- 1 - CICS Analyzer File Overview
- 2 - Analyzing Performance
- 3 - Analyzing Workload
- 4 - Analyzing Uptime

2.3.1 File Overview

The CA MICS Analyzer Option for CICS processes data from SMF type 110, ASG-TMON for CICS TS (TCE) records in the DAY040 step of the CA MICS DAILY job. From type 110 and ASG-TMON for CICS TS (TCE), it inputs transaction and system level records. In particular for CICS TS 2.1 and higher, it inputs the SMF type 110 Statistics records as well as the type 110 Performance and Exception records. The type 110 Performance records carry the transaction level data. Each data field in these records is read during the input processing step in DAY040, but not all of the fields are retained in the CA MICS database. Due to the large volume of CICS transaction records, many users prefer to keep summarized user activity files in the DAYS timespan instead of the DETAIL timespan. Any data fields, such as error flags, that may not be meaningful after file summarization are typically dropped from the CA MICS database.

The CA MICS Analyzer Option for CICS creates eight files in the CIC Information Area. A general description of each is given below. For detailed file descriptions and a list of data elements, see Chapter 5. To compare the possible data sources, see Chapter 6.

OVERALL ACTIVITY

- o CICCXY - CICS System Activity File

The CA MICS Analyzer Option for CICS provides overall system activity data in the CICS System Activity (CICCXY) file. At the DETAIL timespan, the CICCXY file provides system level information for each interval at which a system record is written by CMF or by ASG-TMON for CICS TS (TCE). For CICS releases 3.1.1 and higher, which do not produce CMF global performance records, the CA MICS Analyzer Option for CICS processes and populates the CICCXY file with data from selected SMF type 110 subtype 2 global statistics records, which are produced on an interval basis. At the DAYS timespan, the CA MICS Analyzer Option for CICS incorporates both system and user activity information in the CICCXY file to provide you with a total picture at the hourly basis. This is done by merging the CICCXY file with the CICS User Activity File (CICCSU) by hour.

By using the information provided by the CICCXY file in the DAYS timespan, you can monitor both system and user workload performance on a daily basis. The CICCXY file

size is relatively small compared to the CICCSU and CICS Application Unit Activity (CICCAU) files. This enables quick access through MICS for reporting purposes. In addition, the CA MICS Analyzer Option for CICS produces standard analysis and management reports using the CICCSY file. For a detailed description of the reports and MICS inquiries provided by the CA MICS Analyzer Option for CICS and the lists of data elements used by the reports, see Chapter 3.

USER ACTIVITY

- o CICCAC - CICS User Application Count File
- o CICCSU - CICS User Activity File

The CA MICS Analyzer Option for CICS summarizes CICS resource usage at the user level in several ways. Two files in particular are the CICS User Activity (CICCSU) file and the CICS User Application Count (CICCAC) file. These files are populated from the transaction records, which can be CMF (type 110 subtype 1 Performance Class 3) or ASG-TMON for CICS TS (TCE) TRANSACTION (TA) records.

The CICCSU and CICCAC files contain the identification information for CICS users, which are known to the CA MICS Analyzer Option for CICS as user account codes. You define account codes during installation of the CA MICS Analyzer Option for CICS, and they allow you to identify users based on USERID, OPERID, or any other method of identifying the user based on the input data. You can define up to nine levels of account codes (CICACT1- CICACT9), which are used as sequence elements during the CA MICS file summarization processing. Therefore, you should be careful when defining the accounting fields, as they will affect the sizes of the higher timespans of the CICCSU and CICCAC files in your CA MICS database.

The CICCSU file is used to generate summary reports for workload analysis. In addition, it is used for exception analysis to capture information about users with high resource usage.

APPLICATION ACTIVITY

- o CICCAU - CICS Application Unit Activity File

The CA MICS Analyzer Option for CICS provides CICS resource usage information at the application level in the CICS Application Unit Activity (CICCAU) file. This file is created from the transaction records produced by CMF type 110 subtype 1 Performance class 3 records and by ASG-TMON for CICS TS (TCE) transaction (TA) records.

The CICCAU file contains the Application Unit Identification (CICAPU) element that you can define during installation of the CA MICS Analyzer Option for CICS. CICAPU allows you to identify the CICS applications running at your site, and enables the CA MICS Analyzer Option for CICS to provide you with summarized resource usage information for these applications.

The CICCAU file is used to generate summary reports for workload analysis. In addition, it is used for exception analysis to capture any application with high resource usage.

CICS INCIDENTS

- o CICCIN - CICS Incident File

The CA MICS Analyzer Option for CICS records abnormal events, such as transaction ABENDs, in the CICS Incident (CICCIN) file, which is used for exception analysis. This file is created from the CMF type 110 subtype 1 Performance class 3 and Exception class 4 records and from ASG-TMON for CICS TS (TCE) Transaction (TA) records.

FILE ACCESS ACTIVITY

- o CICCFS - CICS File/DBD Activity File

The CA MICS Analyzer Option for CICS records access to individual files in the CICS File/DBD Activity (CICCFS) file. This file is created from the file segments in the ASG-TMON for CICS TS (TCE) Transaction (TA) records. It is not available from CMF data sources.

The CA MICS Analyzer Option for CICS creates summary reports using the CICCFS file to analyze your file activities.

MRO ACTIVITY

- o CICCMMR - CICS MRO Activity File

The CA MICS Analyzer Option for CICS records MRO interactions between a single transaction and a different system in the CICS MRO Activity (CICCMMR) file. This file is created from the MRO segments in the ASG-TMON for CICS TS (TCE) Transaction records from versions 2.0 and higher. It is not available for CMF.

The CA MICS Analyzer Option for CICS creates summary reports using the CICCMMR file to analyze your MRO activities.

DATA DICTIONARY PROCESSING

- o CICCDC - CICS Dictionary File

The CA MICS Analyzer Option for CICS uses type 110 data dictionary records, which are of subtype 1 and class 1, to populate the CICS Dictionary (CICCDC) file. Before the Performance records, which are of type 110 subtype 1 and class 3, for a particular region are processed, CA MICS must process a data dictionary record for that region, and this data dictionary record must be at the same CICS release level as that of the Performance records that follow it. The data dictionary records provide offset, length, and type information about each of the fields in the Performance records.

A data dictionary record is the first type 110 record that is produced when a CICS region is started. If a data dictionary record is needed but its region cannot be restarted just to obtain a new data dictionary record, one can be produced by using the IBM utility DFHMNDUP, which has been available since the release of CICS 3.1.1. A record that has been produced by DFHMNDUP should be concatenated to the front of the input for the next update run whether it is a DAILY or an incremental. Once a data dictionary record has been input to the CA MICS Analyzer Option for CICS once, care should be taken to make sure that it not necessarily appear in future update runs. Since the data dictionary information is retained in the CICCDC file, it is usually not necessary to keep an old data dictionary record in the input stream. Moreover, if a CICS version is changed, one should make sure that a new data dictionary record comes before the Performance records from the new CICS release.

Information on how to run the DFHMNDUP utility can be found in Section 2.7 (monitoring dictionary utility program) of the IBM CICS Operations and Utilities Guide.

Since the data dictionary values for each CICS region are retained in the CICCDC file, CA MICS will use the information from CICCDC when new data dictionary records do not appear in the input. Observations that carry data dictionary information for a particular region will be deleted from the CICCDC file after a specified number of days if no data from that region are processed by the CA MICS Analyzer Option for CICS for that number of days. The CICCCEX parameter in prefix.MICS.PARMS(CICTHRSH) is used to specify the number of days for this cutoff. It is described in Section 7.3.8 CICS Processing Thresholds (CICTHRSH).

Since the only purpose of the CICCDC file is to allow the CA MICS Analyzer Option for CICS to decode the input data properly, only one cycle of the CICCDC file is kept in the DETAIL timespan.

Data dictionary records and the CICCDC file are required to process only the type 110 subtype 1 Performance class 3 records or type 110 look-alike Performance records. No other type 110 records require data dictionary records. No ASG-TMON for CICS TS (TCE) records require data dictionary records.

2.3.2 Analyzing Performance

In terms of CICS, good performance is delivered when:

- o Short and consistent response time is provided to maintain user productivity and satisfaction.
- o The service level objectives stated by the users (explicitly or implicitly) have been met.
- o There is effective use of hardware resources (CPU, main storage, DASD, communication links) in providing the service required.

Therefore, the main focus of a CICS performance analysis is on the CICS response time measures. The CA MICS Analyzer Option for CICS provides you with summary and management reports to analyze response time. The summary reports are produced by submitting a batch job after the CA MICS DAILY update is completed. The management reports are produced automatically by the CA MICS operational jobs DAILY, WEEKLY, and MONTHLY. In addition, MICF inquiries are provided for you to analyze response time. You can modify the inquiries to suit your data center's needs. The MICF inquiries are executed on an ad hoc basis. See Chapter 3 of this guide for detailed descriptions and samples of the following reports:

- o Summary Reports
 - Daily Activity Overview Report - Response Time Distributions
 - Application Unit Summary Report
 - Daily User Activity Summary Report
- o Management Reports
 - Daily CICS Service Report - All Activity
 - Daily CICS Service Report - Medium Activity
 - Daily CICS Service Report - Short Activity
 - Weekly CICS Service Report - All Activity
 - Weekly CICS Service Report - Medium Activity
 - Weekly CICS Service Report - Short Activity
 - Monthly CICS Service Report - All Activity
 - Monthly CICS Service Report - Medium Activity
 - Monthly CICS Service Report - Short Activity
- o Graphic Inquiries

- CICCOT - 8AM - 5PM CICS Service by Type
- CICCOT2 - Daily CICS Service Report
- CICCOT2 - Weekly CICS Service Report
- CICCOT2 - Monthly CICS Service Report

o Printer Graphic Inquiries

- CICPD2 - Daily CICS Service Report
- CICPW2 - Weekly CICS Service Report
- CICPM2 - Monthly CICS Service Report

In addition to the reports provided, you can code SAS programs to analyze response time using the following data elements:

FILE	ELEMENT	DESCRIPTION
-----	-----	-----
CICCSY	CSYAVSTM	Avg Short Response Time
CICCSY	CSYAVMTM	Avg Medium Response Time
CICCSY	CSYAVLTM	Avg Long Response Time
CICCSY	CSYAVTTM	Avg Response Time All Functions
CICCSY	CSYSDST1- CSYSDST8	Count Short Response Within Limit n
CICCSY	CSYMDST1- CSYMDST8	Count Medium Response Within Limit n
CICCSY	CSYLDST1- CSYLDST8	Count Long Response Within Limit n
CICCSY	CSYTDST1- CSYTDST8	Count Total Response Within Limit n
CICCSY	CSYMXSTM	Max Short Response Time
CICCSY	CSYMXMTM	Max Medium Response Time
CICCSY	CSYMXLTM	Max Long Response Time
CICCSY	CSYMXTTM	Max Response Time All function

Note: The same set of data elements are also available from the CICCOT and CICCOT files. For example, to display the average short response time for a specific user, you can use the data element CSUAVSTM. For average short response time for an application unit, use the data element CAUAVSTM.

In addition to the CICS response time measures, you can also use the following data elements to examine problem conditions that may affect CICS performance:

FILE	ELEMENT	DESCRIPTION
-----	-----	-----

CICCSY	CSYPAGIN	Page Ins (pre CICS 3.1.1)
CICCSY	CSYMXTSK	Max Tasking Condition Detected
CICCSY	CSYSOS	Short on Storage Condition
CICCSY	CSYPCMD	Number of Program Compressions
CICCSY	CSYWTIM	CICS Dispatcher Wait Time
CICCSY	CSYWTCPU	CPU Across OPSYS Wait (ASG-TMON)
CICCSY	CSYMEMSZ	High-water Dynamic Area Memory
CICCSY	CSYDAMSC	Damaged Storage Chain
CICCSY	CSYUSRTM	User (application) Dispatched Time
CICCSY	CSYTCUTM	User TCB CPU Time
CICCSU	CSUABEND	Abnormal termination occurrences
CICCSU	CSUFCWAT	File Control Wait Time
CICCSU	CSUIRWAT	MRO/ISC Wait Time
CICCSU	CSUJCWAT	Journal Control Wait Time
CICCSU	CSUSUSTM	Time Transaction was on CICS Susp Chn
CICCSU	CSUTSWAT	Temporary Storage Wait Time
CICCSU	CSUTCWAT	Terminal Control Wait Time
CICCSU	CSUWAITM	Transaction Wait Time

Note: The set of CICCSU data elements are also available from the CICCAU file if the CICCAU file is active. For example, to obtain the number of abnormal termination occurrences for an application unit, use the data element CAUABEND.

2.3.3 Analyzing Workload

To plan for future growth, you must first analyze your current workload. This requires breaking down the total work into categories and developing a workload profile for each category. This profile contains:

- o transaction profile (number of functions)
- o transaction volume
- o resources required (real storage, DASD I/O, terminal I/O)

The CA MICS Analyzer Option for CICS breaks down the workload by summarizing detail transaction data by the application unit IDs that you define at product installation time. However, prior to using the CICS Application Unit Activity File (CICCAU) for your workload analysis, you should be aware of the factors that can affect the total transaction count in CA MICS.

TRANSACTION COUNTS

The CA MICS Analyzer Option for CICS provides counts of CICS transactions in the CICCXY, CICCXY, and CICCXY files for various transaction types. These include short, medium, long, and conversational transactions, as well as total number of transactions.

Normally, one count is added to each of the above counts for every detail transaction record input to the CA MICS Analyzer Option for CICS. The transaction records are written for each transaction execution. In an MRO/ISC environment, this translates to multiple records for a routed transaction, as each region produces a record for its execution of the transaction. The CA MICS Analyzer Option for CICS currently provides no facility to reduce the multiple records to a single "unit of work." For conversational tasks, a transaction record is usually written for each segment of the conversation, thus resulting in multiple records for the transaction. For CMF, multiple records are written for a conversational task when the parameter CONV=YES is specified in DFHMCT TYPE=RECORD macro.

There is an exception when a transaction record is not added to any of the above transaction counts. This occurs when you assign a TRANTYPE of "X" to a transaction using the CICS relative longevity routine (CICRLRT). This exit allows you to exclude certain transactions from the CA MICS response and

transaction distribution calculations, ensuring that the response calculations are not skewed by transactions that may have an abnormally long response time. Although the TRANTYPE "X" transactions are not included in the total transaction count, the CA MICS Analyzer Option for CICS does provide a separate set of "excessive" counters for you to track these transactions.

Note: The CPU time for the excessive transactions is included in the total CPU time calculation.

Given the above conditions, you should consider the following when analyzing the CICS transaction counts in the CICS files:

- o The data elements presented by CA MICS no longer represent the CICS task count, and should not be analyzed for max task or max task within class purposes.
- o The transaction count data elements indicate relative system load by transaction. An example of this is a transaction that executes conversationally all day. It may have very few CICS task counts, but it may also account for a large percentage of the total CICS resource use.
- o Total transaction count does not include "excessive" transactions as designated by your CICRLRT exit.

CICS FILE CONTROL CALLS

As part of the workload analysis, you are also estimating the number of DASD I/Os that are performed by each category of transactions. The CA MICS Analyzer Option for CICS provides file access counts for you to analyze DASD usage. However, CICS File Control calls do not correlate to the number of physical I/Os performed for each file, as CICS only records the number of logical requests made. If you have large VSAM buffers allocated above the 16Mb line, there can be virtually no physical I/Os performed for "get" type operations; however, CICS will continue to count the file control calls. On the other hand, a single insert operation to a VSAM file with little free space can cause a CI or CA split, which may result in many physical I/Os, but CICS will only show one file control call in this case. Therefore, you should take care not to use the number of file control calls as the physical I/O count.

REPORTS

The CA MICS Analyzer Option for CICS provides you with summary and management reports to analyze CICS workload. The summary reports are produced by submitting a batch job after the CA MICS daily update is completed. The management reports are produced automatically by the CA MICS operational jobs DAILY, WEEKLY, and MONTHLY. In addition, MICF inquiries are provided for you to analyze CICS workload. You can modify the inquiries to suit your data center's needs. The MICF inquiries are executed on an ad hoc basis. See Chapter 3 of this guide for detailed descriptions and samples of the following reports:

- o Summary Reports
 - Application Unit Summary Report
- o Management Objective Reports
 - Daily CICS Workload Report
 - Weekly CICS Workload Report
 - Monthly CICS Workload Report
- o Graphic Inquiries
 - CICC01 - Daily CICS Workload Report
 - CICC01 - Weekly CICS Workload Report
 - CICC01 - Monthly CICS Workload Report
- o Printer Graphic Inquiries
 - CICPD1 - Daily CICS Workload Report
 - CICPW1 - Weekly CICS Workload Report
 - CICPM1 - Monthly CICS Workload Report

In addition to the above reports, you can code SAS programs to retrieve workload-related data elements from the CA MICS database.

FILE	ELEMENT	DESCRIPTION
-----	-----	-----
CICCAU	CAUAVINC	Average Input Character Traffic
CICCAU	CAUAVMEM	Average Transaction Memory Used
CICCAU	CAUAVOPS	Average number of CICS Calls
CICCAU	CAUAVOTC	Average Output Character Traffic
CICCAU	CAUAVTRT	Average Transaction CPU Time

CICCAU	CAUCINP	Input Characters - Primary Facility
CICCAU	CAUCOUTP	Output Characters - Primary Facility
CICCAU	CAUCINS	Input Characters - Secondary Facility
CICCAU	CAUCOUTS	Output Characters -Secondary Facility
CICCAU	CAUCPRTM	Task CPU Real Time
CICCAU	CAUCPUTM	Task CPU Time (elapsed)
CICCAU	CAUDLIWT	DL/I Call Elapsed Time (ASG-TMON)
CICCAU	CAUDLICC	DL/I Call Count (ASG-TMON)
CICCAU	CAUFCTOT	Total File Requests
CICCAU	CAUJCOPS	Number of Journal Control Calls
CICCAU	CAUMINP	Input Messages - Primary Facility
CICCAU	CAUMINS	Input Messages - Secondary Facility
CICCAU	CAUMOUTP	Output Messages - Primary Facility
CICCAU	CAUMOUTS	Output Messages - Secondary Facility
CICCAU	CAULTRN	Long Transaction Count
CICCAU	CAUMTRN	Medium Transaction Count
CICCAU	CAUPCOPS	Number of Program Control Calls
CICCAU	CAUSCOPS	Number of CICS Storage Control Calls
CICCAU	CAUSIOCT	Access Method Calls
CICCAU	CAUSPOPS	Number of CICS Synch Point Calls
CICCAU	CAUSTOTM	Storage Occupancy Time
CICCAU	CAUSTRN	Short Transaction Count
CICCAU	CAUTDGET	Transient Data Gets
CICCAU	CAUTDOPS	Number of CICS Transient Data Calls
CICCAU	CAUTDPUT	Transient Data Puts
CICCAU	CAUTDPRG	Transient Data Purges
CICCAU	CAUTRANS	Transactions processed
CICCAU	CAUTSGET	Temporary Storage Gets
CICCAU	CAUTSOPS	Number of CICS Temporary Storage Calls
CICCAU	CAUTSPTA	Temporary Storage Puts to AUX
CICCAU	CAUTSPTM	Temporary Storage Puts to Main

2.3.4 Analyzing Uptime

The CA MICS Analyzer Option for CICS assumes that the CICS system is available when system records from pre-CMF 3.1.1 and ASG-TMON are processed. It equates uptime to the duration of the system records. Duration is calculated by subtracting the start timestamp from the end timestamp. For pre-CMF 3.1.1, the start and end timestamps are both available in the data. For ASG-TMON, only the start timestamp is available. The CA MICS Analyzer Option for CICS uses the start timestamp of the next system record as the end timestamp of the current record. For the last ASG-TMON system record encountered for each region, the CA MICS Analyzer Option for CICS calculates the end timestamp by adding the start timestamp, the total dispatch time, and wait time for the interval.

CMF 3.1.1 and higher do not produce system records. The CA MICS Analyzer Option for CICS assumes that the CICS system is available when SMF type 110, subtype 57 records are processed. The subtype 57 records are written at user specified intervals. If a record represents either the first period after recording is initiated or the last period before recording is terminated the CA MICS Analyzer Option for CICS sets duration to the sum of quasi-reentrant dispatch and wait time, otherwise it sets duration to the user specified recording interval. The CA MICS Analyzer Option for CICS equates uptime to the duration of the subtype 57 records.

For CMF, the CA MICS Analyzer Option for CICS validates the calculated duration against an interval value that you define at product installation time. If the calculated duration is less than zero or 1.5 times greater than the expected interval, then DURATION is set to the specified interval value.

Validation is not performed for ASG-TMON, since its system records do not have to be written at an interval basis.

DETECTING A CICS OUTAGE

No event recorded by the supported CICS monitors indicates the beginning of CICS processing. Task numbers are not a reliable means of determining a CICS warm/cold start, because the CICS task counter rolls over at 100000 tasks executed. Therefore, the CA MICS Analyzer Option for CICS uses the absence of "up" time to infer the presence of "down" time. Actual detection of the down condition is difficult.

One CA MICS CICS management report shows CICS outage time. The outage shown on the report is downtime (absence of uptime) in any hour for which data was found. Gaps of no data in the histogram show periods in which no downtime was found. Unfortunately, the gaps can therefore represent periods of CICS uptime AND periods of no data.

DETECTING MONITOR DATA LOSS

The CICS monitor you are using may be turned off at any time during the day. This is a theoretical safety valve to enable a reduction of CICS overhead during periods of system stress or logging device unavailability. This condition looks exactly like a CICS system outage to the CA MICS Analyzer Option for CICS, because there is no data to indicate any type of CICS activity.

SYSTEM INTERVAL/HOUR SYNCHRONIZATION

Pre-CMF 3.1.1 produces periodic system records based on a user-defined interval. Usually, the interval is 15 minutes.

CICS Statistics 3.1.1 and above produce periodic statistics records on a user-defined interval basis. The default interval is 3 hours, but you should consider changing the interval to 1 hour. You may change the interval using CEMT SET STATISTICS and the EXEC CICS SET STATISTICS command.

Pre-CMF 3.1.1 and CICS Statistics 3.1.1 and above use timer facilities that key on time of day to trigger the production of a system/statistics record for an interval. The initial time of day (time the first record is cut) is calculated to be on an interval boundary. In the 15-minute case, system records are cut immediately after the hour and at 15, 30, and 45 minutes after the hour.

Because the CA MICS data elements HOUR and ZONE are both derived from end timestamp, system records with different starting and ending hours will always be summarized in the ending hour. This causes a misapplication of summarized system activity up to the interval of the system records. For instance, CMF system records produced on a 15-minute interval can be misapplied by up to 15 minutes. The following will clarify this:

Data for the period 08:30:00.20 to 08:45:00.20 is used by the CA MICS Analyzer Option for CICS to construct a record whose hour identification is hour '8'. Data for the period 08:45:00.20 to 09:00:00.20 is used by the CA MICS Analyzer Option for CICS to construct a record whose hour identification is hour '9', even though the data mostly describes events which occurred in the eighth hour of the day.

You should consider this misapplication when you perform any analysis with the CICS System Activity File (CICCSY).

REPORTS

The CA MICS Analyzer Option for CICS provides you with management reports to analyze CICS uptime. The reports are produced automatically by the CA MICS operational jobs DAILY, WEEKLY, and MONTHLY. In addition, MICF inquiries are also provided to report CICS uptime. The MICF inquiries are executed on an ad hoc basis. See Chapter 3 of this guide for detailed descriptions and samples of the following reports:

- o Management Objective Reports
 - Daily CICS Availability Report
 - Weekly CICS Availability Report
 - Monthly CICS Availability Report
- o Graphic Inquiries
 - CICCD3 - Daily CICS Availability Report
 - CICCW3 - Weekly CICS Availability Report
 - CICCMB - Monthly CICS Availability Report
- o Printer Graphic Inquiries
 - CICPD3 - Daily CICS Availability Report
 - CICPW3 - Weekly CICS Availability Report
 - CICPM3 - Monthly CICS Availability Report

In addition to the above reports, you can code SAS programs to analyze CICS uptime using the following data elements:

FILE	ELEMENT	DESCRIPTION
-----	-----	-----
CICCSY	CSYUPTM	CICS Availability Time
CICCSY	DURATION	Recording Interval Time
CICCSY	STARTTS	Start Timestamp
CICCSY	ENDTS	End Timestamp

2.4 Accounting and Chargeback Interface

The CICS Analyzer provides data elements that CA MICS Accounting and Chargeback can use to bill for resource or transaction usage. The elements that can be billed are:

CICS TRANSACTION ACCOUNTING

CICS Transaction Executions

CICS RESOURCE ACCOUNTING

CICS CPU Charges

CICS Instructions Executed
 CICS Service Units
 CICS System Resource Units
 CICS Task CPU Time (elapsed)
 CICS Task CPU Real Time
 CICS Pseudo Elapsed Time
 CICS SRB CPU Time
 CICS TCB CPU Time

CICS Terminal Charges

CICS Access Method Calls (SIO Count)
 CICS Input Messages Issued
 CICS Output Messages Issued
 CICS I/O (EXCPs) Generated
 CICS Input Message Character Traffic
 CICS Output Message Character Traffic
 CICS Number Of Input Messages

CICS Transaction Charges

CICS Conversational Transactions Processed
 CICS Long Transactions Processed
 CICS Medium Transactions Processed
 CICS Short Transactions Processed
 CICS Excessive Transactions Processed
 CICS Transactions Processed

CICS Resource Units

- CICS BMS Calls
- CICS File Control Calls
- CICS Interval Control Calls
- CICS Journal Control Calls
- CICS Program Control Calls
- CICS Storage Control Calls
- CICS Synch Point Calls
- CICS Transient Data Calls
- CICS Temp Storage Calls
- CICS Calls (Total)

CICS Data Base DL/I Logical I/O Charges

- CICS DL/I Calls - Get Unique
- CICS DL/I Calls - Get Next
- CICS DL/I Calls - Get Next (Par)
- CICS DL/I Calls - Get Hold Unique
- CICS DL/I Calls - Get Hold Next
- CICS DL/I Calls - Get Hold Next (Par)
- CICS DL/I Calls - Insert
- CICS DL/I Calls - Delete
- CICS DL/I Calls - Replace

CICS Memory Charges

- Memory Usage

2.5 Capacity Planner Interface

As an aid to rapid and effective workload forecasting of CICS data, the CA MICS Capacity Planner provides an interface to the CICS Analyzer that contains the necessary elements for CICS tracking and reporting.

Known as the CICS Planning Standard Application, this interface provides a capacity planning data base file with the data elements you need for basic tracking, reporting, and forecasting of the CICS workload. It uses the application unit features of the CICS Analyzer to group the individual CICS systems into logically similar workload groups for more meaningful tracking and reporting. The three methods for grouping CICS system are:

- o defining each CICS system as its own workload group,
- o combining all CICS systems (no workload group), or
- o defining customized workload groupings.

Overall, the interface of the CICS Analyzer with the Capacity Planner provides a set of tools for addressing what would otherwise be a more time-consuming and complex task.

Chapter 3: REPORTS

The CA MICS Analyzer Option for CICS provides a comprehensive reporting facility to help you manage and monitor CICS on a daily, weekly, and monthly basis. Long-term trending based on historical data is also provided to show the growth in CICS workload and resource consumption. The reporting facility delivers both graphics and tabular reports, all of which are accessible through the CA MICS Information Center Facility (MICF). Both can be executed in either batch or foreground mode. The CICS reports and graphics are available in the following areas:

- o Service Level - The level of service provided by CICS in terms of response time. The reports display response time information at the user or application level to show how well CICS is servicing its customers and the degree to which it is meeting your site's service level objectives.
- o Workload - The volume of transactions processed by CICS. The reports show the total and peak volume of work performed by CICS, as well as the large users or applications in your data center.
- o Performance - A summary of CICS performance in terms of response time, transaction volume, and availability at the system level for all regions processed by the CA MICS Analyzer Option for CICS. The reports track occurrences of abnormal conditions that impact CICS performance, such as short-on-storage and maximum task conditions.
- o Cost - The amount of charges incurred by CICS users or applications. The reports shows the total charges and the top users or applications (in terms of charges) in your data center.
- o Availability - The percentage of time CICS is available. This information is derived from the duration of the monitoring records processed by the CA MICS Analyzer Option for CICS.

To meet the reporting needs of those responsible for CICS in various positions, such as senior managers, technical managers, and performance analysts, the CA MICS Analyzer Option for CICS provides the following types of reports, most of which are available in the reporting areas discussed above.

- o Ranking - These tabular reports provide an integrated list of the top consumers of CICS. They are of most interest to technical managers and performance analysts who want to identify users with the highest cost, the worst response time, or the highest transaction volume.
- o Summary - These tabular reports provide technical managers and performance analysts with a quick summary of CICS activities at the individual system level, as well as the total activity within the enterprise.
- o Detail - These tabular reports provide the most detailed information by displaying CICS activities at the user or application level. They are typically used by the performance analyst on an as-needed basis for ad-hoc analysis.
- o Management - These reports provide a concise graphic or tabular representation of your data center's processing objectives and how well they have been met. They are targeted towards technical managers.
- o Exception - These tabular reports provide a concise integrated and itemized list of the problems affecting CICS in terms of service, workload, performance, availability, and security. Their primary audience is CICS performance analysts and systems programmers. Exception reports are discussed in detail in Chapter 4 of this guide.
- o Graphics - For presentation to senior managers, the CA MICS Analyzer Option for CICS provides a comprehensive set of color graphics (or printer graphics if you do not have color devices) to show the activities and trend in cost, workload, and service.

You can produce the above reports or generate your own by using the following facilities:

- o CA MICS Information Center Facility
- o Batch Operations
- o Interactive Reporting (MSAS)

For CICS Transaction Gateway, the CA MICS Analyzer Option for CICS provides a daily MICF report with both summary and detail information about Gateway transaction throughput and response times.

MICF information Center Facility

As mentioned previously, the CICS reporting facility is accessible through MICF, a menu-based system that operates under IBM's Interactive System Productivity Facility (ISPF). If you are familiar with ISPF commands, you will find that MICF menus are similar to standard ISPF menus. Each of the CICS tabular or graphic reports is shipped as a MICF inquiry, which comprise a series of SAS DATA steps or PROCs to manipulate the data in the CA MICS database and to produce the final report. You can augment the CICS reporting facility either by tailoring an existing MICF inquiry or by writing your own inquiries. See the MICF Reference Guide and MICF User Guide for more information on how to use MICF to run the distributed inquiries and create your own.

Batch Operations

Two types of reports, management and exception, are produced automatically by the CA MICS operational jobs. The daily format of these two reports is generated by the DAY400 step of the DAILY job or, if the AUTOSUBMIT parameter is specified in prefix.MICS.PARMS(JCLDEF), by the DAILYRPT job submitted during DAY400. The weekly reports are generated by either the WEEK400 step of the WEEKLY job or the WEEKRPT job. The monthly reports are produced by either the MONTH400 step of the monthly job or the MONTHRPT job.

The rest of the CA MICS Analyzer Option for CICS reports are not automatically produced by the CA MICS operational jobs. However, you can add them to the operational jobs using the MICF production interface facility. This is discussed in more detail in Section 3.3.1 of this guide.

Some of the CA MICS Analyzer Option for CICS reports, previously known as the standard analysis reports, were originally shipped as batch reports and were not accessible through MICE. Although they are now available through MICE, you can continue to produce them via batch jobs after the CA MICS operational jobs have completed. The sample JCL and the list of reports that can be produced this way are discussed in Section 3.3.2 of this guide.

Interactive Reporting (MSAS)

Interactive access to SAS with CA MICS is provided through the MSAS dialog from the CA MICS Workstation Facility (MWF). The MSAS dialog, running under ISPF, allows experienced SAS users to exercise a flexible set of options for invoking interactive full-screen SAS in the CA MICS environment. The MSAS dialog supports several types of interactive reporting, including:

- read-only access to a single unit database
- read-only access to multiple unit databases
- SAS without any unit databases

To support the interactive use of SAS with CA MICS, MSAS allocates the required SAS files, work files, user files, sort files, CA MICS libraries, and, optionally, the CA MICS database. In addition to allocating CA MICS database files, MSAS uses the standard CA MICS DDNAMEs (for example, SOURCE, INCLLIB, and USOURCE) to allocate the CA MICS library data sets. This lets you select SAS statements from the CA MICS libraries, modify and execute these statements under SAS/DMS, and then save the program for future use.

MSAS is a MICF application running under your private MICF options. Through MICF Options, you can control the set up of SAS execution parameters and temporary data set allocations. You can also control the allocation of additional data sets for one-time or repeated use.

To use MSAS, see Chapter 3 of this guide.

The remainder of this chapter discusses the format and execution of CA MICS Analyzer Option for CICS reports.

This section contains the following topics:

[3.1 MICF Inquiries](#) (see page 74)

[3.2 Changing Report Parameters](#) (see page 379)

[3.3 Running Reports in Batch](#) (see page 381)

3.1 MICF Inquiries

The CA MICS Analyzer Option for CICS provides a comprehensive set of report and color graphic inquiries that are implemented using the CA MICS Information Center Facility (MICF). The reports are categorized as general workload, service, performance, availability, and finance.

These inquiries are an immediate source of useful reporting that you can modify to meet your needs. You can use Execution Time Data Selection panels to tailor the output from these inquiries. In addition, you can develop customized reports by modifying copies of the distributed inquiries.

MICF inquiry names follow the form `cccptn`, where:

- `ccc` = the three-character produce identifier. `CIC` is for the CA MICS Analyzer Option for CICS.
- `p` = the type of graphic. The value of this field is either `C` (color graphic), `L` (list), or `P` (printer graphic).
- `t` = the frequency with which the report may be executed. The value of this field is `X` (detail), `D` (days), `W` (weeks), `M` (months), and `Y` (years).
- `n` = an integer that differentiates this inquiry from others.

For example, an inquiry named `CICLD4` corresponds to the following key:

```
CICLD4
---|||
| ||+= the fourth list inquiry at the
| ||  DAYS timespan delivered with the
| ||  CA MICS Analyzer Option for CICS
| |= may be run daily
| += a list inquiry
+= a CICS inquiry
```

The following figure lists the MICF inquiries distributed by the CA MICS Analyzer Option for CICS for analysis of CICS and CICS Transaction Gateway:

MICF Distributed Inquiries for CICS Analysis						
Report Class	Report Name	MICF Catalog Group	MICF Color Graphics	Inquiry Identifier		
				Printer Graphics	Tabular List	
Service	8am - 5pm CICS Service By Type	SERVICE	CICCDT			
	Daily CICS Executive Summary	SERVICE	CICCD0			
	Daily CICS Service	SERVICE	CICCD0	CICPDC		
	Daily CICS Service Ranking Report	SERVICE	CICCD8	CICPDB		CICLDB
	Daily CICS Service Report	SERVICE	CICCD2	CICPD2		
	Daily CICS Service by Type	SERVICE	CICCD0			
	Daily CICS Service Detail Report	SERVICE				CICLDF
	Daily CICS Service Summary Report	SERVICE				CICLDE
	Weekly CICS Service Report	SERVICE	CICCW2	CICPW2		
	Monthly CICS Executive Summary	SERVICE	CICCM0			
	Monthly CICS Service Report	SERVICE	CICCM2	CICPM2		
	Monthly CICS Service Ranking Report	SERVICE	CICCM8	CICPMB		CICLMB
	Monthly CICS Service Report by Application	SERVICE		CICPMC		
	Monthly CICS Service Detail Report	SERVICE				CICLMF
	Monthly CICS Service Summary Report	SERVICE				CICLME
Workload	Daily CICS Workload Ranking Report	WORKLOAD	CICCDH	CICPDH		CICLDH
	Daily CICS Workload Detail Report	WORKLOAD				CICLDM
	Daily CICS Workload Summary Report	WORKLOAD				CICLDL
	Daily CICS Workload Report	WORKLOAD	CICCDI	CICPDI		
	Daily CICS Workload Report	WORKLOAD	CICCD1	CICPD1		
	Daily CICS Workload Report All Regions	WORKLOAD	CICCDJ			
	Weekly CICS Workload Report	WORKLOAD		CICPWI		
	Weekly CICS Workload Report	WORKLOAD	CICCW1	CICPW1		
	Weekly CICS Workload Report All Regions	WORKLOAD	CICCWJ			
	Monthly CICS Workload Ranking Report	WORKLOAD	CICCMH	CICPMH		CICLMH
	Monthly CICS Workload Detail Report	WORKLOAD				CICLMM
	Monthly CICS Workload Summary Report	WORKLOAD				CICLML
	Monthly CICS Workload Report	WORKLOAD		CICPMI		
	Monthly CICS Workload Report	WORKLOAD	CICCM1			CICPM1
	Monthly CICS Workload Report All Regions	WORKLOAD	CICCMJ			
	CICS System Summary Report	WORKLOAD				CICLD1
	MRO Activity Summary	WORKLOAD	CICLD2			
System Performance	Daily CICS System Performance Summary	PERFORM				CICLDP
	Weekly CICS System Performance Summary	PERFORM				CICLWP
	Monthly CICS System Performance Summary	PERFORM				CICLMP

Figure 3-1. MICF Distributed Inquiries (part 1 of 2)

Availability	Daily CICS Availability	AVAIL	CICCDQ	CICPDQ	
	Daily CICS Availability Report	AVAIL	CICCD3	CICPD3	
	Weekly CICS Availability	AVAIL		CICPWQ	
	Weekly CICS Availability Report	AVAIL	CICCW3	CICPW3	
	Monthly CICS Availability	AVAIL		CICPMQ	
	Monthly CICS Availability Report	AVAIL	CICCM3	CICPM3	
Finance	Daily CICS Cost Ranking Report	FINANCE	CICCDR	CICPDR	CICLDR
	Daily CICS Cost Detail Report	FINANCE			CICLDX
	Daily CICS Cost Summary Report	FINANCE			CICLDW
	Daily CICS Cost All Systems	FINANCE	CICCMS		
	Daily CICS Cost	FINANCE		CICPDS	
	Monthly CICS Cost Ranking Report	FINANCE	CICCMR	CICPMR	CICLMR
	Monthly CICS Cost Detail Report	FINANCE			CICLMX
	Monthly CICS Cost Summary Report	FINANCE			CICLMW
	Monthly CICS Cost All Systems	FINANCE	CICCMR		
	Monthly CICS Cost	FINANCE		CICPMS	
General	Daily CICS System Overhead Analysis	GENERAL			CICLDA
MBO	Daily CICS Mgmt. by Obj. (MBO) Rpt.	MBO			CICMDO
	Monthly CICS Mgmt. by Obj. (MBO) Rpt.	MBO			CICMMO
	Weekly CICS Mgmt. by Obj. (MBO) Rpt.	MBO			CICMWO
MICF Distributed Inquiries for CICS Transaction Gateway Analysis					
Report Class	Report Name	MICF Catalog Group	MICF Color Graphics	Inquiry Identifier Printer Graphics	Tabular List
System Performance	CICS Transaction Gateway Daily Throughput	PERFORM			CICLDG

Figure 3-1. MICF Distributed Inquiries (part 2 of 2)

Figure 3-1 lists the CA MICS Analyzer Option for CICS MICF inquiries. The inquiries are described individually in the following sections:

- 1 - Service Analysis Inquiries
- 2 - Workload Analysis Inquiries
- 3 - Performance Analysis Inquiries
- 4 - Availability Analysis Inquiries
- 5 - Financial Analysis Inquiries
- 6 - General Analysis Inquiries
- 7 - CICS Transaction Gateway Inquiries

3.1.1 Service Analysis Inquiries

Service analysis inquiries pertain to the management of CICS service levels in terms of response time, and include items necessary for providing daily data processing services in the most efficient and effective manner, as well as meeting service objectives and agreements. The CICS Analyzer distributes the following service analysis MICF inquiries:

Catalog Group	Inquiry ID	Inquiry Name	Report format	Run-time execution parameters	User exits
SERVICE	CICCDB	Daily CICS Service Ranking	color graphic		
	CICCDC	Daily CICS Service	color graphic		
	CICCD0	Daily CICS Service by Type	color graphic		
	CICCD0	Daily CICS Executive Summary	color graphic		
	CICCDT	8am - 5pm CICS Service By Type	color graphic		
	CICCD2	Daily CICS Service Reports	color graphic		
	CICCMB	Monthly CICS Service Ranking	color graphic		
	CICCM0	Monthly CICS Executive Summary	color graphic		
	CICCM2	Monthly CICS Service Reports	color graphic		
	CICCW2	Weekly CICS Service Reports	color graphic		
	CICPDB	Daily CICS Service Ranking	printer graphic		
	CICPDC	Daily CICS Service by Application	printer graphic		
	CICPD2	Daily CICS Service Reports	printer graphic		
	CICPMB	Monthly CICS Service Ranking	printer graphic		
	CICPMC	Monthly CICS Service	printer graphic		
	CICPM2	Monthly CICS Service Reports	printer graphic		
	CICPM2	Weekly CICS Service Reports	printer graphic		
	CICLDB	Daily CICS Service Ranking Report	tabular report		
	CICLDE	Daily CICS Service Summary Report	tabular report		
	CICLDF	Daily CICS Service Detail Report	tabular report		
	CICMDB	Monthly CICS Service Ranking Report	tabular report		
	CICMDE	Monthly CICS Service Summary Report	tabular report		
	CICMDF	Monthly CICS Service Detail Report	tabular report		

Figure 3-2. MICF Distributed Inquiries

To provide report flexibility, the CICS Analyzer supplies MICF execution run time parameters as well as user exits for most of the service analysis inquiries. You can use either method to tailor the inquiries to suit your reporting needs.

The service analysis inquiries are described individually in the following sections:

- 1 - CICCDB: Daily CICS Service Ranking
- 2 - CICCDC: Daily CICS Service
- 3 - CICCDD: Daily CICS Service by Type
- 4 - CICCOT: 8am - 5pm CICS Service By Type
- 5 - CICCDD2: Daily CICS Service Reports
- 6 - CICCMB: Monthly CICS Service Ranking
- 7 - CICCOC: Monthly CICS Service
- 8 - CICCOC2: Monthly CICS Service by Type
- 9 - CICCOCG: Monthly CICS Service Trend
- 10- CICCOC2: Monthly CICS Service Reports
- 11- CICCWC: Weekly CICS Service
- 12- CICCWD: Weekly CICS Service by Type
- 13- CICCWG: Weekly CICS Service Trend
- 14- CICCW2: Weekly CICS Service Reports
- 15- CICPDB: Daily CICS Service Ranking
- 16- CICPDC: Daily CICS Service
- 17- CICPD2: Daily CICS Service Reports
- 18- CICPMB: Monthly CICS Service Ranking
- 19- CICPMC: Monthly CICS Service
- 20- CICPM2: Monthly CICS Service Reports
- 21- CICPWC: Weekly CICS Service
- 22- CICPM2: Weekly CICS Service Reports
- 23- CICLDB: Daily CICS Service Ranking Report
- 24- CICLDE: Daily CICS Service Summary Report
- 25- CICLDF: Daily CICS Service Detail Report
- 26- CICLMB: Monthly CICS Service Ranking Report
- 27- CICLME: Monthly CICS Service Summary Report
- 28- CICLMF: Monthly CICS Service Detail Report

3.1.1.1 CICCDB: Daily CICS Service Ranking

The Daily CICS Service Ranking inquiry produces a color graph which identifies the top 10 CICS applications that have received the worst average response time in a given day.

REPORT FORMAT:

This inquiry produces a side-by-side horizontal bar chart which shows the top 10 applications with the worst average response time. The chart also shows the transaction volume for each of the 10 applications in the same order as the response ranking. By default, only one chart is produced for all input data which is summarized across SYSIDs and CICSIDs. To produce a ranking graph for a specific system or CICS region, you must specify the value for SYSID, CICSID, or CENTER at inquiry execution time. See the description below for execution-time parameters.

INQUIRY ID:

CICCDB (color graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUS (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAUPU.

DATA ELEMENTS USED:

CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
CICAUPU - CICS Application Unit ID
CICSID - CICS System Identifier
DAY - Day of Month
MONTH - Month of Year
SYSID - System Identifier
YEAR - Year of Century
ZONE - Time Zone

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS

EXECUTION-TIME PARAMETERS:

- UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
- CYCLE - Optionally, define CA MICS input file cycle range.
- SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.
- DATE - Optionally, enter the date to be reported in ddmmmyy format.
- For this inquiry, the default is yesterday.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is `TERMINAL`. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is `NO`.

FILE ID- Optionally, select the `CICCSU` file as input when the `CICCAU` file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (`CICACT1-CICACT9`) identifies your CICS application.

CICAPU - Optionally, replace `CICAPU` with any of the `CICACTx` data elements, where `x` is 1 to 9. This parameter must be specified in conjunction with `FILE ID`.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module `CICRPTGL` in `sharedprefix.MICS.SOURCE`. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is `CICAPU`.

%GCIC - Select `CICSIDs` for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.1.2 CICCDC: Daily CICS Service

The Daily CICS Service inquiry produces a color graph which correlates, on an hourly basis, the overall average response time with short, medium, long, and conversational transaction workload. It shows the effect the volume of work has on response time and identifies the time of day when CICS failed to meet your service level objectives.

REPORT FORMAT:

This inquiry produces an overlay plot, using two vertical axes, which shows the hourly average response time and transaction volume. The average response time is plotted against the left vertical axis, while the short, medium, long and conversational transactions against the right.

INQUIRY ID:

CICCDC (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan.

DATA ELEMENTS USED:

CICSID	- CICS System Identifier
CSYCTRN	- Conversational Transactions Processed
CSYSTRN	- Short Transactions Processed
CSYMTRN	- Medium Transactions Processed
CSYLTRN	- Long Transactions Processed
CSYTRANS	- Total Transactions Ended
CSYTRSTM	- Total Response Time All Functions
DAY	- Day of Month
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

Average response time = CSYTRSTM/CSYTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.1.3 CICCDD: Daily CICS Service by Type

The Daily CICS Service by Type inquiry produces a color graph which correlates, on an hourly basis, the overall average response time with short, medium, and long response times. It shows the effect that a particular type of response events has on the overall CICS response time. It also identifies the time of day when CICS service fails to meet your service level objectives.

REPORT FORMAT:

This inquiry produces an overlay plot, using two vertical axes, which shows the hourly short, medium, long, and overall average response times. The average response time is plotted against the left vertical axis, while the short, medium, and long response times against the right.

INQUIRY ID:

CICCDD (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYSRSTM - Short Response Time Total
CSYSTRN - Short Transactions Processed
CSYMRSTM - Medium Response Time Total
CSYMTRN - Medium Transactions Processed
CSYLRSTM - Long Response Time Total
CSYLTRN - Long Transactions Processed
CSYTRANS - Total Transactions Ended
CSYTRSTM - Total Response Time All Functions
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = $CSYTRSTM / CSYTRANS$
Average short response time = $CSYSRSTM / CSYSTRN$
Average Medium response time = $CSYMRSTM / CSYMTRN$
Average long response time = $CSYLRSTM / CSYLTRN$

EXECUTION-TIME PARAMETERS:

- UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
- CYCLE - Optionally, define CA MICS input file cycle range.
- SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- DATE - Optionally, enter the date to be reported in ddmmmyy format.
- For this inquiry, the default is yesterday.
- BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.1.4 CICCDT: Daily 8AM - 5PM CICS Service by Type

The 8AM - 5PM CICS Service by Type Reports monitor prime time hourly CICS response time by transaction type and identify any hours during which CICS service falls below the management objective.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the hourly average response time for each transaction type from 8 a.m. to 5 p.m. The charts are available in color graphic format only.

INQUIRY ID:

CICCDT (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan

Data is selected for HOURS 8-17.

DATA ELEMENTS USED:

CSYAVLTM - Avg Long Response Time
CSYAVMTM - Avg Medium Response Time
CSYAVSTM - Avg Short Response Time

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to

be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is
used which causes all CICSIDs found in input to
be summarized together.

USER EXITS:

None

3.1.1.5 CICCD2: Daily CICS Service Reports

The Daily CICS Service Reports monitor CICS response time and identify hours during the day when service falls below the management objective.

REPORT FORMAT:

This inquiry produces bar charts by system ID, CICS ID, and transaction type showing the average response time for each hour and the percentage of transactions with less than one-or-five-second response time. The bar charts are available in color graphic format or printer format (see CICPD2).

INQUIRY ID:

CICCD2 (color graphics format)

DATA SOURCE (file/timespan):

CICCSY01 at the DAYS timespan

DATA ELEMENTS USED:

CICRVAL1 - CICRVAL7 Response Distribution Limits
CSYPCSR1 - CICPCSR7 Cumm Pct Short Resp within Limit x
CSYPCMR1 - CICPCSR7 Cumm Pct Medium Resp within Limit x
CSYPCTR1 - CICPCSR7 Cumm Pct All Resp within Limit x

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is

used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

USER EXITS:

None

3.1.1.6 CICCMB: Monthly CICS Service Ranking

The Monthly CICS Service Ranking inquiry produces a color graph which identifies the 10 CICS applications with the worst service (average response time) in a given month.

REPORT FORMAT:

This inquiry produces a side-by-side horizontal bar chart which shows the top 10 applications with the worst average response time. The chart also shows the transaction volume for each the 10 applications, in the same order as the response ranking. This chart is available in color graphic format only.

INQUIRY ID:

CICCMB (color graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUS (CICS User Activity) file will be used for reporting. The variable CICAUT2 will be used in place of the variable CICAUP.

One graph produced for each date and each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICAUP - CICS Application Unit ID
CICSID - CICS System Identifier
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by

entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is last month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by

MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.1.7 CICCMC: Monthly CICS Service

The Monthly CICS Service inquiry produces a color graph which correlates CICS workload and average response time for the past 6 months. It enables you to identify a trend in CICS service and workload.

REPORT FORMAT:

This inquiry produces an overlay plot, using two vertical axes, which shows the hourly CICS transaction volume and average response time. The transaction volume is plotted against the right vertical axis, while the average response time against the left. The transaction volume is further divided by transaction type (short, medium, long) to show the workload characteristics. This plot is available in color graphic format only.

INQUIRY ID:

CICCMC (color graphic format)

DATA SOURCE (file/timespan):

CICCSY file from monthly history file.

One graph produced for each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICSID	- CICS System Identifier
CSYTRANS	- Transactions Processed
CSYTRSTM	- Transaction Response Time Total
CSYSTRN	- Short Transactions Processed
CSYMTRN	- Medium Transactions Processed
CSYLTRN	- Long Transactions Processed
CSYCTRN	- Conv. Transactions Processed
DAY	- Day of Month
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

Average response time = CSYTRSTM/CSYTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is the last 6 months.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF

inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.1.8 CICCMTD: Monthly CICS Service by Type

The Monthly CICS Service by Type inquiry produces a color graph which correlates the overall average response time with short, medium, and long response times for the last six months. It shows the effect that a particular type of response events has on the overall CICS response time. It also identifies the trend in CICS service.

REPORT FORMAT:

This inquiry produces an overlay plot, using two vertical axes, which shows the short, medium, long, and overall average response times. The average response time is plotted against the left vertical axis, while the short, medium, and long response times against the right.

INQUIRY ID:

CICCMTD (color graphic format)

DATA SOURCE (file/timespan):

CICCSY in the monthly history file.

One graph produced for each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICSID	- CICS System Identifier
CSYCRSTM	- Conversational Response Time Total
CSYCTR	- Conversational Transactions Processed
CSYSRSTM	- Short Response Time Total
CSYSTR	- Short Transactions Processed
CSYMRSTM	- Medium Response Time Total
CSYMTR	- Medium Transactions Processed
CSYLRSTM	- Long Response Time Total
CSYLTR	- Long Transactions Processed
CSYTRANS	- Total Transactions Ended
CSYTRSTM	- Total Response Time All Functions
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

Average response time = CSYTRSTM/CSYTRANS

Average short response time = CSYSRSTM/CSYSTRN
Average Medium response time = CSYMRSTM/CSYMTRN
Average long response time = CSYLRSTM/CSYLTRN
Average conv response time = CSYCRSTM/CSYCTRN

EXECUTION-TIME PARAMETERS:

- UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
- CYCLE - Optionally, define CA MICS input file cycle range.
- SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.
- MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is the last 6 months.

- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.
- FOOTNOTE - Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input

file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

- %GCIC - Select CICSIDs for report.
- %GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.
- %GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.1.9 CICCMTG: Monthly CICS Service Trend

This inquiry is unavailable at this time.

3.1.1.10 CICC2: Monthly CICS Service Reports

The Monthly CICS Service Reports identify CA MICS ZONEs during which CICS service falls below the user-defined management objective and monitor the service trend by ZONE in a six-month period.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID, CICS ID, and transaction type showing the average response time per ZONE and the percentage of transactions with less than one- or five-second response time for a six-month period. This inquiry produces charts in color graphic format. The charts can also be produced in printer graphic format by executing MICF inquiry CIMPM2.

INQUIRY ID:

CICC2 (color graphic format)

DATA SOURCE (file/timespan):

CYCCSY01-CYCCSY06 at the MONTHS timespan

DATA ELEMENTS USED:

CICRVAL1 - CICRVAL7	Response Distribution Limits
CSYPCMR1 - CSYPCMR7	Cumm Pct Medium Resp within Limit x
CSYPCSR1 - CSYPCSR7	Cumm Pct Short Resp within Limit x
CSYPCTR1 - CSYPCTR7	Cumm Pct All Resp within Limit x

CALCULATIONS:

None

RUN-TIME EXECUTION PARAMETER OVERRIDES:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is

used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

None

3.1.1.11 CICCWC: Weekly CICS Service

The Weekly CICS Service inquiry produces a color graph which correlates CICS workload and average response time for the past 9 weeks. It enables you to identify a trend in CICS service and workload.

REPORT FORMAT:

This inquiry produces an overlay plot, using two vertical axes, which shows the hourly CICS transaction volume and average response time. The transaction volume is plotted against the right vertical axis, while the average response time against the left. The transaction volume is further divided by transaction type (short, medium, long) to show the workload characteristics. This plot is available in color graphic format only.

INQUIRY ID:

CICCWC (color graphic format)

DATA SOURCE (file/timespan):

CICCSY file from weekly history file.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
CSYCTRN - Conv. Transactions Processed
WEEK - Week of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CSYTRSTM/CSYTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs

with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is 01-09 cycles.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

DATE - Optionally, enter start of the report period in ddmmmyy format.

For this inquiry, the default is the last nine weeks.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

DEVICE - Optionally, override the default colors and

patterns used by specifying a different device type. The default is `TERMINAL`. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is `NO`.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module `CICRPTGL` in `sharedprefix.MICS.SOURCE`. The following exits are available to this inquiry:

`%GCIC` - Select `CICSIDs` for report.

`%GCNTR` - Assign value to the variable `CENTER` which defines the logical grouping of `SYSIDs`.
`CENTER` is set to `SYSID` by default.

`%GSYS` - Select `SYSIDs` for report.

In addition to the above exits, the module `CICRPTGL` also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.1.12 CICCWD: Weekly CICS Service by Type

The Weekly CICS Service by Type inquiry produces a color graph which correlates, for the past 9 weeks, the overall average response time with short, medium, and long response times. It shows the effect that a particular type of response events has on the overall CICS response time. It also identifies the trend in CICS service.

REPORT FORMAT:

This inquiry produces an overlay plot, using two vertical axes, which shows the short, medium, long, and overall average response times. The average response time is plotted against the left vertical axis, while the short, medium, and long response times against the right.

INQUIRY ID:

CICCWD (color graphic format)

DATA SOURCE (file/timespan):

CICCSY in the weekly history file.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYSRSTM - Short Response Time Total
CSYSTRN - Short Transactions Processed
CSYMRSTM - Medium Response Time Total
CSYMTRN - Medium Transactions Processed
CSYLRSTM - Long Response Time Total
CSYLTRN - Long Transactions Processed
CSYTRANS - Total Transactions Ended
CSYTRSTM - Total Response Time All Functions
WEEK - Week of Year
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = $CSYTRSTM / CSYTRANS$
Average short response time = $CSYSRSTM / CSYSTRN$
Average Medium response time = $CSYMRSTM / CSYMTRN$
Average long response time = $CSYLRSTM / CSYLTRN$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is 01-09 cycles.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

DATE - Optionally, enter start of the report period in ddmmmyy format.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.1.13 CICCWG: Weekly CICS Service Trend

The Weekly CICS Service Trend inquiry produces a color graph which shows the trend of CICS service in the past 9 weeks, as well as the projected service for the next 8 weeks. The predicted values are generated by PROC GPLOT using linear regression to fit a line through the actual data points.

REPORT FORMAT:

This inquiry produces an overlay plot, using two vertical axes, which shows the trend in CICS average response time and average transaction rate. The average response time is plotted against the left vertical axis, while the average transaction rate against the right. This plot is available in color graphic format only.

INQUIRY ID:

CICCWG (color graphic format)

DATA SOURCE (file/timespan):

CICCSY in the weekly history file.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYSRSTM - Short Response Time Total
CSYSTRN - Short Transactions Processed
CSYMRSTM - Medium Response Time Total
CSYMTRN - Medium Transactions Processed
CSYLRSTM - Long Response Time Total
CSYLTRN - Long Transactions Processed
CSYTRANS - Total Transactions Ended
CSYTRSTM - Total Response Time All Functions
DURATION - Recording Interval Time
WEEK - Week of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = $CSYTRSTM / CSYTRANS$
Average short response time = $CSYSRSTM / CSYSTRN$
Average Medium response time = $CSYMRSTM / CSYMTRN$
Average long response time = $CSYLRSTM / CSYLTRN$

EXECUTION-TIME PARAMETERS:

- UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
- CYCLE - Optionally, define CA MICS input file cycle range.
- SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.
- DATE - Optionally, enter the date to be reported in ddmmmyy format.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.
- DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.1.14 CICCW2: Weekly CICS Service Reports

The Weekly CICS Service Reports Monitor CICS response time trends in a seven-day period (excluding weekends) and identify any hour between 8 a.m. and 5 p.m. when service falls below the user-defined management objective.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID, CICS ID, and transaction type showing the average response time for each hour (from 8 a.m. to 5 p.m.) and the percentage of transactions with less than one- or five-second response time. The charts are based on data summarized at the DATE and HOUR levels. Data from Saturday and Sunday is excluded.

This inquiry produces charts in color graphic format. You can also produce the bar charts in printer graphic format by executing the MICF inquiry CICPW2.

INQUIRY ID:

CICCW2 (color graphic format)

DATA SOURCE (file/timespan):

CYCCSY01-CYCCSY07 at the DAYS timespan

Data is selected for HOURS 8-16, excluding DAYNAME = 'SAT' and DAYNAME = 'SUN'.

DATA ELEMENTS USED:

CICRVAL1 - CICRVAL7 Response Distribution Limits
CSYPCMR1 - CSYPCMR7 Cumm Pct Medium Resp within Limit x
CSYPCSR1 - CSYPCSR7 Cumm Pct Short Resp within Limit x
CSYPCTR1 - CSYPCTR7 Cumm Pct All Resp within Limit x

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle

range.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

None

3.1.1.15 CICPDB: Daily CICS Service Ranking

The Daily CICS Service Ranking inquiry produces a printer graph which identifies the 10 CICS applications with the worst average response time in a given day.

REPORT FORMAT:

This inquiry produces a horizontal bar chart which shows the top 10 applications with the worst average response time.

A sample output of this inquiry is shown in Figure 3-3.

INQUIRY ID:

CICPDB (printer graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUS (CICS User Activity) file will be used for reporting. The variable CICT2 will be used in place of the variable CICAU.

One graph produced for each date and each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICAU - CICS Application Unit ID
CICSID - CICS System Identifier
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs

with a blank following each ID value.

- CYCLE - Optionally, define CA MICS input file cycle range.
- SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- DATE - Optionally, enter the date to be reported in ddmmmyy format.
- For this inquiry, the default is yesterday.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.
- RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.
- FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.
- CICAPU - Optionally, replace CICAPU with any of the CICAx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

Figure 3-3. Daily CICS Service Ranking Sample Report

3.1.1.16 CICPDC: Daily CICS Service by Application

The Daily CICS Service inquiry produces a graph which correlates hourly CICS workload and average response time. It enables you to identify the hours during the day when CICS service failed to meet your service level objective.

REPORT FORMAT:

This inquiry produces an overlay plot which shows the hourly CICS transaction volume and average response time. A sample output of this inquiry is shown in Figure 3-4.

INQUIRY ID:

CICPDC (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
CSYCTRN - Conv. Transactions Processed
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CSYTRSTM/CSYTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

- SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- DATE - Optionally, enter the date to be reported in ddmmyy format.

For this inquiry, the default is yesterday.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.
- BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.
- FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.
- CICAPU - Optionally, replace CICAPU with any of the CICAx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

- %GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.
- %GCIC - Select CICSIDs for report.
- %GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.
- %GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.



3.1.1.17 CICPD2: Daily CICS Service Reports

The Daily CICS Service Reports monitor CICS response time and identify hours during the day when service falls below the management objective.

REPORT FORMAT:

This inquiry produces bar charts by system ID, CICS ID, and transaction type showing the average response time for each hour and the percentage of transactions with less than one or five-second response time.

The bar charts are produced in printer graphic format. You can also generate them by executing the MICF inquiry CICCD2. A sample output of this inquiry is shown in Figure 3-5.

INQUIRY ID:

CICPD2 (printer graphics format)

DATA SOURCE (file/timespan):

CICCSY01 at the DAYS timespan

DATA ELEMENTS USED:

CICRVAL1 - CICRVAL7 Response Distribution Limits
CSYPCSR1 - CICPCSR7 Cumm Pct Short Resp within Limit x
CSYPCMR1 - CICPCSR7 Cumm Pct Medium Resp within Limit x
CSYPCTR1 - CICPCSR7 Cumm Pct All Resp within Limit x

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is

used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICF time zones. Valid values are numbers 1 through 9.

USER EXITS:

None

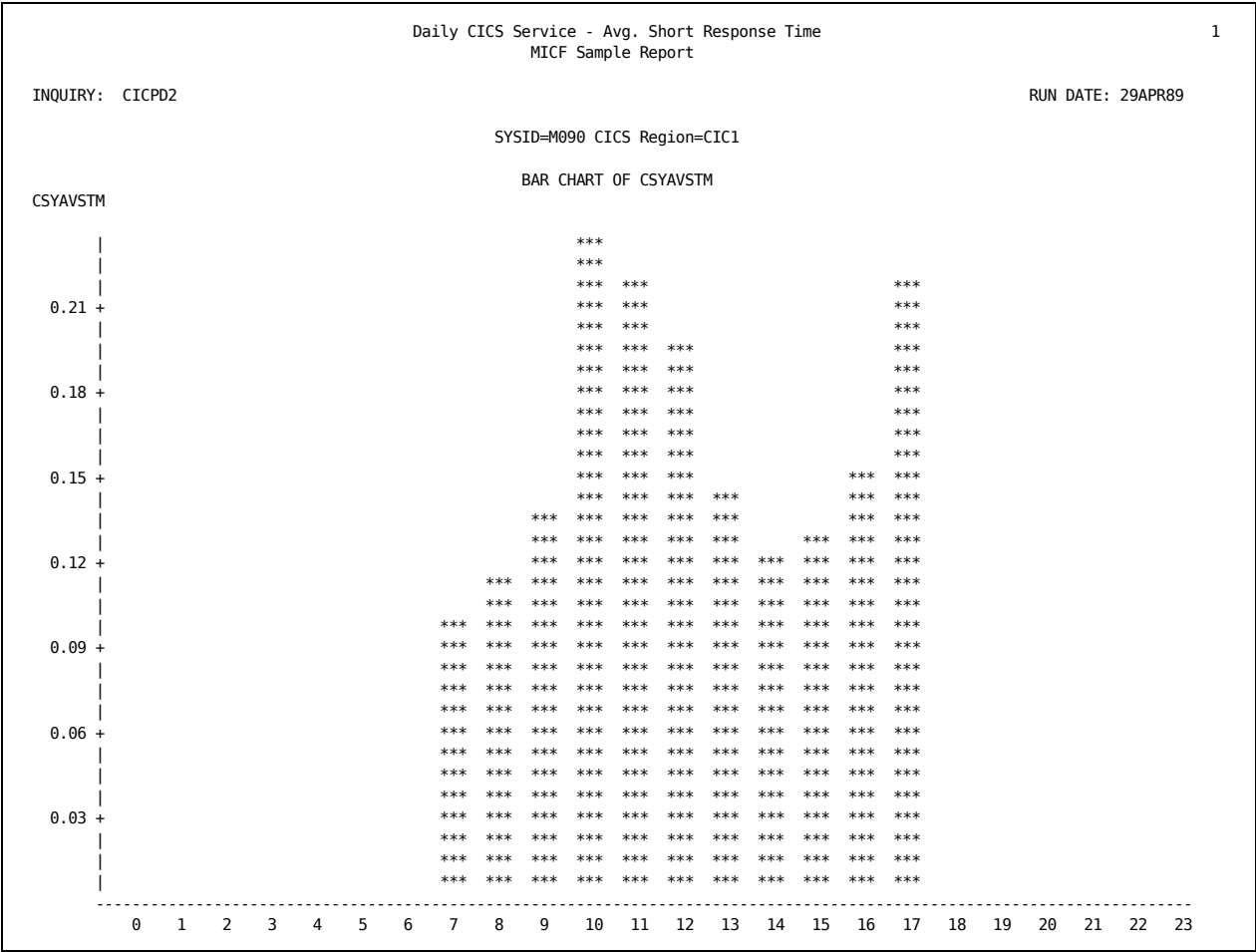


Figure 3-5. Daily CICS Service Report Sample Output

3.1.1.18 CICPMB: Monthly CICS Service Ranking

The Monthly CICS Service Ranking inquiry produces a graph that identifies the 10 CICS applications with the worst average response time in a given month.

REPORT FORMAT:

This inquiry produces a horizontal bar chart which shows the 10 applications with the worst average response time. A sample output of this inquiry is shown in Figure 3-6.

INQUIRY ID:

CICPMB (printer graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAPU.

DATA ELEMENTS USED:

CICAPU - CICS Application Unit ID
CICSID - CICS System Identifier
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is last month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

RANK - Optionally, specify the number of observations to be ranked.

For this inquiry, the default is 10.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICACTx data elements, where x is 1 to 9. This

parameter must be specified in conjunction with
FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

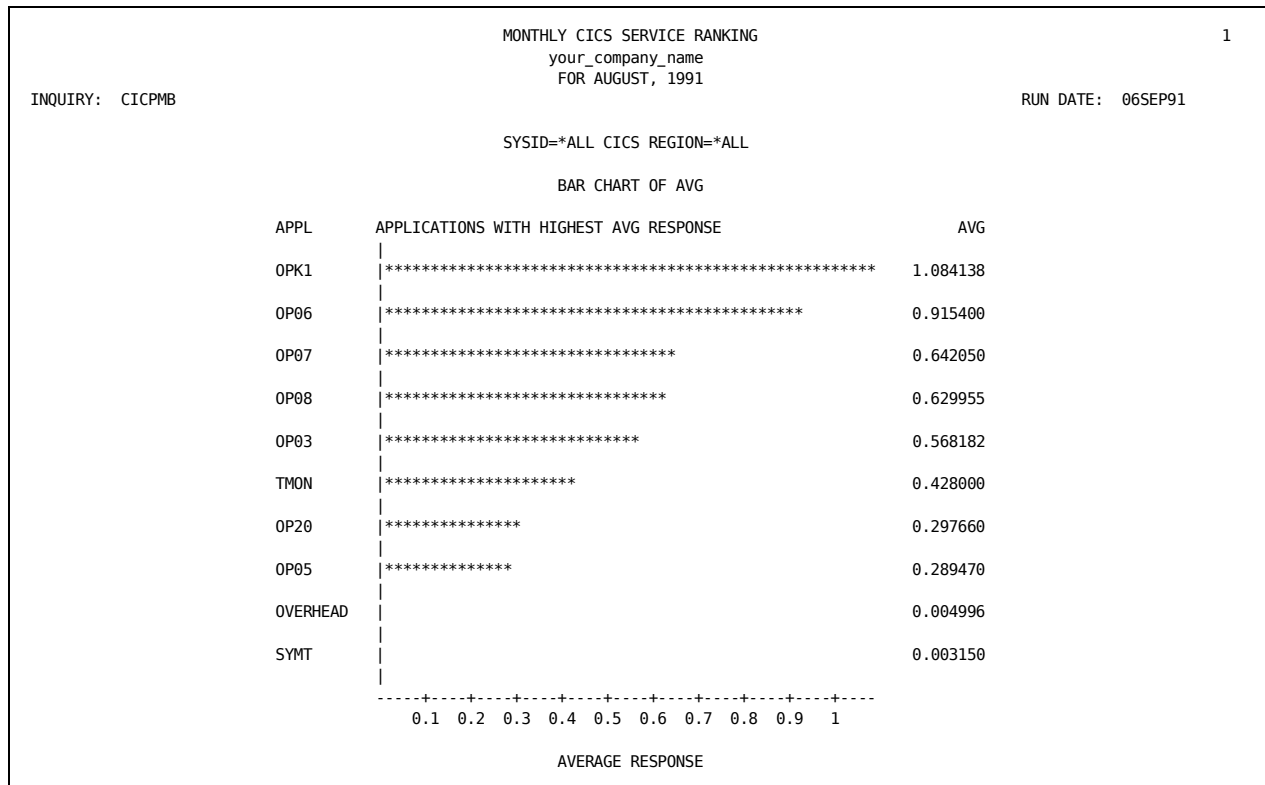


Figure 3-6. Monthly CICS Service Ranking Sample Output

3.1.1.19 CICPMC: Monthly CICS Service by Application

The Monthly CICS Service inquiry produces a graph which shows the hourly CICS workload and average response time in the past 24 months. It enables you to identify the trend in CICS service and the period during which CICS fails to meet your service level objectives.

REPORT FORMAT:

This inquiry produces an overlay plot which shows the hourly CICS transaction volume and average response time. A sample output of this inquiry is shown in Figure 3-7.

INQUIRY ID:

CICPMC (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY in the monthly history file.

One graph produced for each date and each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
CSYCTRN - Conv. Transactions Processed
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CSYTRSTM/CSYTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

CYCLE - Optionally, define CA MICS input file cycle range.

TYPE - Optionally, substitute average response with short, medium, long response time.

USER EXITS:

If the run-time execution parameters are insufficient to select the data you need, you can invoke one of the following global or local user exits to further manipulate the input file. For detailed descriptions on coding the exits, refer to Section 3.2.

Global exits - invoked before run-time parameter macros.

%RPTSEL - Select input data based on any variables in the input file.

Local exits - invoked after run-time parameter macros.

%CICPMC - Select input data based on any variables in the input file.

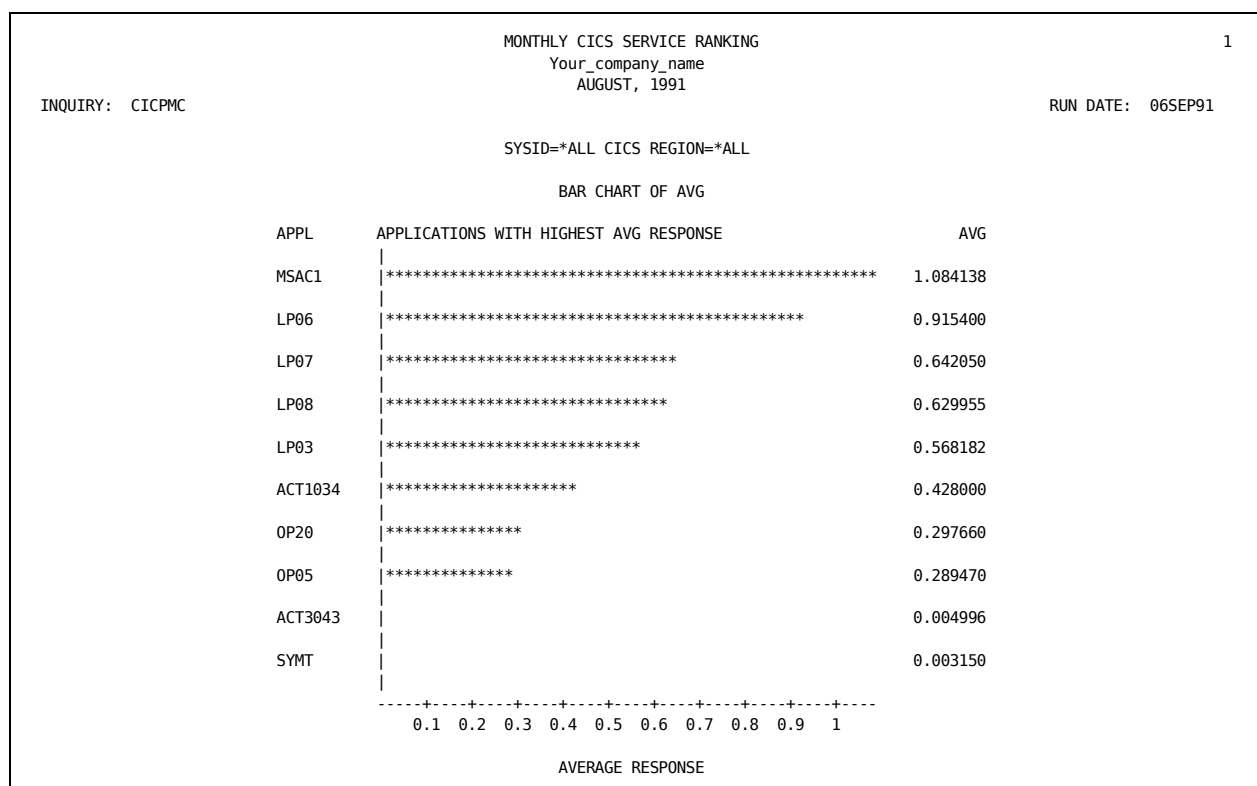


Figure 3-7. Monthly CICS Service Sample Output

3.1.1.20 CICPM2: Monthly CICS Service Reports

The Monthly CICS Service Reports identify CA MICS ZONEs during which CICS service falls below the user-defined management objective and monitor the service trend by ZONE in a six-month period.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID, CICS ID, and transaction type showing the average response time per ZONE and the percentage of transactions with less than one- or five-second response time for a six-month period.

The bar charts are available in printer graphic or color graphic format. A sample output from the inquiry is shown in Figure 3-8. Due to space constraints in the guide, the sample report shows a five month period.

INQUIRY ID:

CICPM2 (printer graphic format)
CICCM2 (color graphic format)

DATA SOURCE (file/timespan):

CYCCSY01-CYCCSY06 at the MONTHS timespan

DATA ELEMENTS USED:

CICRVAL1 - CICRVAL7 Response Distribution Limits
CSYPCMR1 - CSYPCMR7 Cumm Pct Medium Resp within Limit x
CSYPCSR1 - CSYPCSR7 Cumm Pct Short Resp within Limit x
CSYPCTR1 - CSYPCTR7 Cumm Pct All Resp within Limit x

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

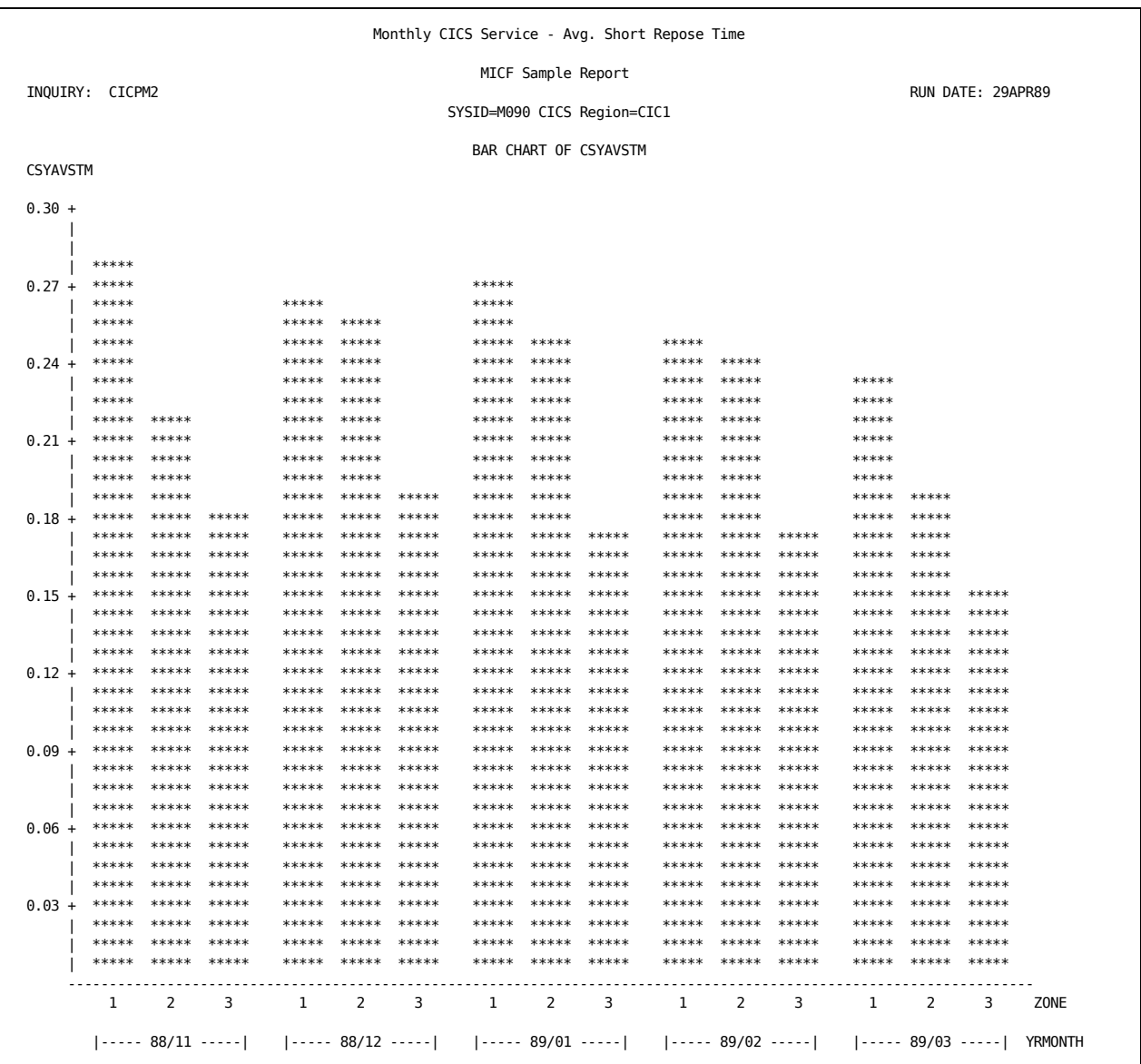
SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

None



3.1.1.22 CICPW2: Weekly CICS Service Reports

The Weekly CICS Service Reports Monitor CICS response time trends in a seven-day period (excluding weekends) and identify any hour between 8 a.m. and 5 p.m. when service falls below the user-defined management objective.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID, CICS ID, and transaction type showing the average response time for each hour (from 8 a.m. to 5 p.m.) and the percentage of transactions with less than one or five seconds response time. The charts are based on data summarized at the DATE and HOUR levels. Data from Saturday and Sunday is excluded.

The bar charts are available in printer graphic format. A sample output from this inquiry is shown in Figure 3-9.

INQUIRY ID:

CICPW2 (printer graphic format)

DATA SOURCE (file/timespan):

CYCCSY01-CYCCSY07 at the DAYS timespan

Data is selected for HOURS 8-16, excluding DAYNAME = 'SAT' and DAYNAME = 'SUN'.

DATA ELEMENTS USED:

CICRVAL1 - CICRVAL7	Response Distribution Limits
CSYPCMR1 - CSYPCMR7	Cumm Pct Medium Resp within Limit x
CSYPCSR1 - CSYPCSR7	Cumm Pct Short Resp within Limit x
CSYPCTR1 - CSYPCTR7	Cumm Pct All Resp within Limit x

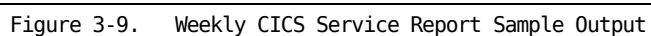
CALCULATIONS: None

EXECUTION-TIME PARAMETERS:

- UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
- CYCLE - Optionally, define CA MICS input file cycle range.
- SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

None



3.1.1.23 CICLDB: Daily CICS Service Ranking Report

The Daily CICS Service Ranking inquiry produces tabular reports that identify the top 10 CICS applications with the worst average response time in a given day.

REPORT FORMAT:

This inquiry produces a tabular report which contains a summary section and a ranking section. The summary section shows the overall response distribution by transaction type, and indicates whether or not the service level objective has been met for this day. The ranking section identifies the 10 applications with the worst average response time and provides other pertinent information such as average wait time and percentage of wait in file control, journal control, terminal control, and temporary storage.

The service level objectives used in this report are defined by module #CICMOBJ in prefix.MICS.SOURCE and CICRPTGL in sharedprefix.MICS.SOURCE. See section 3.2 of this guide if you want to override the service level objectives. A sample output of this inquiry is shown in Figure 3-10.

INQUIRY ID:

CICLDB (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAPU.

DATA ELEMENTS USED:

CAUABEND - Abnormal Termination Occurrences
CAUCDSTx - Count Conv. Responses Within Limit x
CAUCPUTM - Task CPU Time
CAUCRSTM - Conv. Response Time Total
CAUCTRN - Conv. Transactions Processed
CAUDAMSC - Damaged Storage Chain
CAUETRNL - Excessive Transactions Processed
CAUFCWAT - Wait Time for CICS File Control
CAUJCWAT - Wait Time for CICS Journal Control
CAUMDSTx - Count Medium Responses Within Limit x
CAUMRSTM - Medium Response Time Total
CAUMTRNL - Medium Transactions Processed
CAUMXTSK - Max Tasking Condition
CAUPETTM - Task Dispatch Time
CAURESTM - Transaction Residency Time
CAUSDSTx - Count Short Responses Within Limit x
CAUSOS - Short on Storage Condition
CAUSRSTM - Short Response Time Total
CAUSTRNL - Short Transactions Processed
CAUSUSTM - Time Transaction was on CICS Susp Chain
CAUTCWAT - Wait Time for CICS Terminal Control
CAUTDSTx - Count Responses Within Limit x
CAUTLWAT - Wait Time for CICS (Total)
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
CAUTSWAT - Wait Time for CICS Temp Storage
CICAPU - CICS Application Unit ID
CICSID - CICS System Identifier
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average conv. response time = $CAUCRSTM/CAUCTRN$
 Average dispatch = $CAUPETTM/SUM(CAUTRANS, CAUETRN)$
 Average long response time = $CAULRSTM/CAULTRN$
 Average medium response time = $CAUMRSTM/CAUMTRN$
 Average response time = $CAUTRSTM/CAUTRANS$
 Average short response time = $CAUSRSTM/CAUSTRN$
 Average suspend = $CAUSOS/SUM(CAUTRANS, CAUETRN)$
 Average wait = $(CAURESTM - CAUPETTM)/SUM(CAUTRANS, CAUETRN)$
 Average CPU = $CAUCPUTM/CAUTRANS$;
 Other wait = $total\ wait - SUM(CAUFCWAT, CAUJCWAT, CAUTCWAT, CAUTSWAT)$
 Percent FC wait = $CAUFCWAT/total\ wait * 100$
 Percent JC wait = $CAUJCWAT/total\ wait * 100$
 Percent OT wait = $Other\ wait/total\ wait * 100$
 Percent TC wait = $CAUTCWAT/total\ wait * 100$
 Percent TS wait = $CAUTSWAT/total\ wait * 100$
 Total wait = $CAURESTM - CAUPETTM$;

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.
- FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.
- CICAPU - Optionally, replace CICAPU with any of the CICTAx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

- %APPLOBJ- Define service level objective for individual applications. By default, the same objectives are set for each application. The objectives are defined in prefix.MICS.SOURCE(#CICMOBJ).
- %GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GSLO - Determine if SLO (service level objectives) has been met. No defaults.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

DAILY CICS SERVICE RANKING REPORT																		
your_company_name																		
FOR: SEPTEMBER 5, 1991																		
INQUIRY: CICLDB										RUN DATE: 06SEP91								
SYSTEM SUMMARY: 7 REGIONS																		
SLO					AVG	RESP	%OBJ	RESPONSE DISTRIBUTION										
MET	TYPE	# TRANSACTIONS	%	RESP	OBJ	MET		<1	<2	<3	<4	<5	<6	<7				
YES	TOTAL	644,548		0.32	15.00	100.0		95.6	2.9	1.1	0.3	0.0	0.0	0.0				
YES	SHORT	362,767	56.3	0.13	5.00	100.0		95.8	2.8	1.0	0.3	0.0	0.0	0.0				
YES	LONG	232,047	36.0	0.68	15.00	100.0		94.5	3.5	1.4	0.4	0.1	0.0	0.0				
YES	CONV	49,734	7.7	0.03	15.00	100.0		99.8	0.1	0.0	0.0	0.0	0.0	0.0				
** TOP 10 APPLICATIONS WITH WORST AVG RESPONSE TIME																		
RANK	APPL	AVG RESP	RESP OBJ	%OBJ MET	# TRANS	AVG DISPCH	AVG CPU	AVG WAIT	AVG SUSP	----- FC	% WAIT JC	TC	TS	OTH	----- SOS	%TRANS MXTSK	----- VIOL	ABND
1	TCSV	908.32	15.00	0.0	1	0.04	0.01	908.28	0.00	0	0	0	0	100	0	0	0	0
2	DSNC	700.14	15.00	66.7	3	1.03	0.55	699.11	1.67	0	0	0	0	100	0	0	0	0
3	OPID	691.24	15.00	83.3	6	0.05	0.00	691.19	690.88	0	0	50	0	50	0	0	0	0
4	LAON	71.69	15.00	63.6	11	0.98	0.01	70.71	2.39	0	0	0	0	100	0	0	0	0
5	GLST	42.70	15.00	89.9	247	0.05	0.03	42.65	34.15	0	0	44	0	55	0	0	0	1
6	ICST	13.80	15.00	100.0	1	0.15	0.01	13.66	12.62	0	0	0	0	100	0	0	0	0
7	RSFU	12.76	15.00	100.0	11	0.27	0.02	12.49	3.46	0	0	0	0	100	0	0	0	0
8	ISHI	12.62	15.00	100.0	1	0.15	0.02	12.47	10.00	0	0	0	0	100	0	0	0	0
9	DPRM	4.65	15.00	100.0	9	0.28	0.03	4.37	3.43	21	0	0	0	79	0	0	0	0
10	CSIR	2.56	15.00	97.3	1,479	0.00	0.00	2.55	0.04	0	1	0	0	99	0	0	0	0

Figure 3-10. Daily CICS Service Ranking Sample Output

3.1.1.24 CICLDE: Daily CICS Service Summary Report

The Daily CICS Service Summary Report shows the service level, primarily response times, provided by each CICS region on a given day. The report displays the actual service and the service objective, as well as flags indicating whether or not the service level objectives have been met. This report provides a quick summary of the activity and service provided by CICS, and enables you to easily identify the regions which have failed to meet your service level objectives.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for overall CICS service provided in the enterprise, and service provided by each system (SYSID). A sample output of this inquiry is shown in Figure 3-11. Each field in the report is described below:

SLO MET:	A flag indicating whether or not the service level objective has been met by CICS. For the overall summary line (SYSID=ALL CICSID=ALL), the flag is set to NO if any of the regions failed to meet its service level objectives. By default, this flag is set to YES if 95% of the transactions met the response objective. Otherwise, it is set to NO. You can override the default percentage or set the flag using actual value vs. objective in the %LDESLO user exit.
SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
AVG RESP ALL:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.
AVG RESP SHORT:	The average response time for all short transactions (TRANTYPE=S as assigned by the CICRLRT exit in prefix.MICS.PARMS).
AVG RESP	The average response time for all medium

MED:	transactions (TRANSTYPE=M as assigned by the CICLRT exit in prefix.MICS.PARMS).
AVG RESP LONG:	The average response time for all long transactions (TRANSTYPE=L as assigned by the CICLRT exit in prefix.MICS.PARMS).
AVG RESP CONV:	The average response time for all conversational transactions (TRANSTYPE=C as assigned by the CICLRT exit in prefix.MICS.PARMS).
TRANS:	The total number of transactions processed, excluding those assigned as TRANSTYPE X in your CICLRT exit in prefix.MICS.PARMS.
RESP OBJ:	The average response time objective for your installation. This number defaults to 2 seconds for all transactions; 1 for short transactions, 3 for medium transactions, and 5 for long transactions. You can override these values using either MICF execution-time parameters or report user exits.
%OBJ MET:	The percentage of transactions that met the response objective.
%AVAIL:	The CICS availability expressed in a percentage. This number is derived by dividing the total CICS uptime by the total report duration.
PEAK HOUR:	The hour during which the peak average response time occurred.
PEAK RESP:	The peak average hourly response time provided by CICS.
PEAK TRAN/SEC:	The transaction rate for the peak hour.
PEAK TRAN:	The total number of transactions processed in the peak hour.
PEAK %OBJ MET:	The percentage of transactions met the response objective during the peak hour.
INQUIRY ID:	
	CICLDE (tabular report)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYCDSTx - Count Conv. Responses Within Limit x
CSYCTRN - Conversational Transactions Processed
CSYETRN - Excessive Transactions Processed
CSYLDSTx - Count Long Responses Within Limit x
CSYLRSTM - Conversational Response Time Total
CSYLRSTM - Long Response Time Total
CSYLTRN - Long Transactions Processed
CSYMDSTx - Count Medium Responses Within Limit x
CSYMRSTM - Medium Response Time Total
CSYMTRN - Medium Transactions Processed
CSYSDSTx - Count Short Responses Within Limit x
CSYSRSTM - Short Response Time Total
CSYSTRN - Short Transactions Processed
CSYTDSTx - Count Responses Within Limit x
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYUPTM - CICS Availability Time
DAY - Day of Month
DURATION - Recording Interval Time
HOUR - Hour of Day
MONTH - Month of Year
SYSID - System Identifier
YEAR - Year of Century
ZONE - Time Zone

CALCULATIONS:

Average response time = $CSYTRSTM / CSYTRANS$
Average short response time = $CSYSRSTM / CSYSTRN$
Average medium response time = $CSYMRSTM / CSYMTRN$
Average long response time = $CSYLRSTM / CSYLTRN$
Average conv response time = $CSYCRSTM / CSYCTRN$
Transaction per second = $CSYTRANS / DURATION$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for

service level objectives. Please see Section 3.2 to override the default values.

DAILY CICS SERVICE SUMMARY REPORT														
your_company_name														
FOR: SEPTEMBER 5, 1991														
INQUIRY: CICLDE										RUN DATE: 06SEP91				
SLO	SYSID	CICSID	-----		AVG RESP	-----		# TRANS	RESP	%OBJ	----- PEAK ACTIVITY -----			
MET			TOTAL	SHORT	MED	LONG	CONV		OBJ	MET	HOUR	RESP	TRAN/SEC	TRANS %MET
YES	*ALL	*ALL	0.32	0.13	.	0.68	0.03	644,528	15.00	100.0	9	0.15	3.0	75,037 100.0
YES	PRD1	*ALL	0.32	0.13	.	0.68	0.03	644,528	15.00	100.0	9	0.15	3.0	75,037 100.0
YES		LPD1	0.12	0.13	.	0.13	0.01	469,815	15.00	100.0	8	0.10	15.8	56,911 100.0
YES		LPD2	0.13	0.22	.	0.13	0.08	77,933	15.00	100.0	10	0.14	2.8	10,161 100.0
YES		LPD3	0.34	0.23	.	0.34	0.19	68,652	15.00	100.0	10	0.14	2.6	9,283 100.0
YES		LPD4	0.26	0.20	.	0.39	0.02	2,606	15.00	100.0	15	0.16	0.1	334 100.0
YES		RPD1	0.61	0.35	.	0.62	0.17	24,056	15.00	100.0	11	0.70	0.8	3,032 100.0
YES		RPD2	0.27	0.21	.	0.42	0.05	1,288	15.00	100.0	13	0.09	0.1	234 100.0
YES		RPD3	0.27	0.21	.	0.42	0.05	1,288	15.00	100.0	13	0.09	0.1	234 100.0
TOTAL # REGIONS: 7														

Figure 3-11. Daily CICS Service Summary Report Sample Output

3.1.1.25 CICLDF: Daily CICS Service Detail Report

The Daily CICS Service Detail Report shows the CICS service provided to individual applications on a given day. It identifies those applications for which CICS delivered unsatisfactory service. The report provides information that is pertinent to response analysis and can help you identify the areas in which the response delay may have occurred. If the CICCAU file is inactive, the report is automatically generated from the CICCSU file, using the value CICTACT2 as the application or project identifier.

REPORT FORMAT:

This inquiry produces a tabular report which contains a summary section and a detail section. The summary section shows the response distribution by transaction type for all CICS regions found in the input. The detail section shows the response time and other pertinent information for individual applications. A detail section is generated for each combination of SYSID and CICSID.

A sample output of this inquiry is shown by Figure 3-12. Each field in this report is described below:

SYSTEM SUMMARY:

- SLO MET:** A flag indicating whether or not the service level objective has been met by CICS. For the overall summary line (SYSID=ALL CICSID=ALL), the flag is set to NO if any of the regions failed to meet its service level objectives. By default, this flag is set to YES if 95% of the transactions met the response objective. Otherwise, it is set to NO. You can override the default percentage or set the flag using actual value vs. objective in the %LDESLO user exit.
- TYPE:** Transaction type: all, short, medium, long, or conversational. Transaction type (TRANSTYPE) is assigned by the CICRLRT exit in prefix.MICS.PARMS.
- TRANS:** The total number of transactions processed, excluding those assigned as TRANSTYPE X in your CICRLRT exit in prefix.MICS.PARMS.
- AVG RESP:** The average response time for the transactions processed. For TYPE=ALL, this number excludes

transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).

RESP OBJ: The average response time objective for your installation. This number defaults to 2 seconds for all transactions; 1 for short transactions, 3 for medium transactions, and 5 for long transactions. You can override these values using either MICF execution-time parameters or report user exits.

%OBJ MET: The percentage of transactions that met the response objective.

RESPONSE DISTRIBU-
TION The response distributions for the 7 response thresholds which you have specified in prefix.MICS.PARMS(CICOPS).

DETAIL SECTION:

SYSID: System identification. This field is set to ALL for data that is summarized for all systems.

CICSID: CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.

APPL: Application identification.

AVG RESP: The average response time for all transactions processed for this application.

MAX RESP: The maximum response time received by the application.

RESP OBJ: The average response time objective for your installation. This number defaults to 2 seconds for all transactions; 1 for short transactions, 3 for medium transactions, and 5 for long transactions. You can override these values using either MICF execution-time parameters or report user exits.

%OBJ MET: The percentage of transactions that met the response objective.

TRANS: The total number of transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).

AVG DISPATCH: The average dispatch time for the application.

AVG CPU: The average TCB CPU time consumed by the application.

AVG WAIT: The average wait time for the application. It is calculated as the difference between total response time and total dispatch time divided by total transactions.

AVG SUSP: The average suspend time for the application.

%WAIT FC: The percentage of time the application spent waiting for file control.

%WAIT JC: The percentage of time the application spent waiting for journal control.

%WAIT TC: The percentage of time the application spent waiting for terminal control.

%WAIT TS: The percentage of time the application spent waiting for temporary storage.

%WAIT OTH: The percentage of time the application spent waiting for other reasons, such as waiting for data base management systems (DB2, IMS) or buffer and strings.

%TRANS SOS: The percentage of transactions processed for this application during short-on-storage condition.

%TRANS MXTSK: The percentage of transactions processed for this application when CICS was at maximum task.

%TRANS VIOL: The percentage of transactions processed for this application that had a storage violation.

%TRANS ABND: The percentage of transactions abend for this application.

INQUIRY ID:

CICLDF (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

DATA ELEMENTS USED:

CAUABEND - Abnormal Termination Occurrences
CAUCDSTx - Count Conv. Responses Within Limit x
CAUCPUTM - Task CPU Time
CAUCRSTM - Conv. Response Time Total
CAUCTRN - Conv. Transactions Processed
CAUDAMSC - Damaged Storage Chain
CAUETRNL - Excessive Transactions Processed
CAUFCWAT - Wait Time for CICS File Control
CAUJCWAT - Wait Time for CICS Journal Control
CAUMDSTx - Count Medium Responses Within Limit x
CAUMRSTM - Medium Response Time Total
CAUMTRNL - Medium Transactions Processed
CAUMXTSK - Max Tasking Condition
CAUPETTM - Task Dispatch Time
CAURESTM - Transaction Residency Time
CAUSDSTx - Count Short Responses Within Limit x
CAUSOS - Short on Storage Condition
CAUSRSTM - Short Response Time Total
CAUSTRNL - Short Transactions Processed
CAUSUSTM - Time Transaction was on CICS Susp Chain
CAUTCWAT - Wait Time for CICS Terminal Control
CAUTDSTx - Count Responses Within Limit x
CAUTLWAT - Wait Time for CICS (Total)
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
CAUTSWAT - Wait Time for CICS Temp Storage
CICAPU - CICS Application Unit ID
CICSID - CICS System Identifier
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average conv. response time = $CAUCRSTM/CAUCTRN$
Average dispatch = $CAUPETTM/SUM(CAUTRANS,CAUETRNL)$;
Average long response time = $CAULRSTM/CAULTRNL$
Average medium response time = $CAUMRSTM/CAUMTRNL$
Average response time = $CAUTRSTM/CAUTRANS$
Average short response time = $CAUSRSTM/CAUSTRNL$
Average suspend = $CAUSOS/SUM(CAUTRANS,CAUETRNL)$;
Average wait= $(CAURESTM-CAUPETTM)/SUM(CAUTRANS,CAUETRNL)$;
Average CPU = $CAUCPUTM/CAUTRANS$;


```
Other wait=total wait-SUM(CAUFCWAT,CAUJCWAT,CAUTCWAT,
                           CAUTSWAT);
Percent FC wait = CAUFCWAT/total wait * 100;
Percent JC wait = CAUJCWAT/total wait * 100;
Percent OT wait = Other wait/total wait * 100;
Percent TC wait = CAUTCWAT/total wait * 100;
Percent TS wait = CAUTSWAT/total wait * 100;
Total wait= CAURESTM-CAUPETTM;
```

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the

next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICA_x data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%APPLOBJ- Define service level objective for individual applications. By default, the same objectives are set for each application. The objectives are defined in prefix.MICS.SOURCE(#CICMOBJ).

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

DAILY CICS SERVICE DETAIL REPORT																	
your_company_name																	
FOR: SEPTEMBER 5, 1991																	
INQUIRY: CICLDF																	
RUN DATE: 06SEP91																	
----- SYSID=PRD2 CICSID=PCP1 -----																	
APPL	AVG RESP	MAX RESP	RESP OBJ	%OBJ MET	TRANS	AVG DISPCH	AVG CPU	AVG WAIT	AVG SUSP	FC	JC	TC	TS	OTH	SOS	MXTSK	VIOL
AAON	34.10	34.10	15.00	100.0	1	0.36	0.08	33.74	0.00	0	0	0	0	100	0	0	0
ACFM	0.02	0.13	15.00	100.0	163	0.02	0.01	53.56	53.55	0	0	100	0	0	0	0	0
CATD	0.07	3.80	15.00	100.0	15,759	0.02	0.01	0.06	0.00	0	0	0	0	100	0	0	0
CCMF	0.00	0.04	15.00	100.0	233	0.00	0.00	0.00	0.00	0	0	0	0	100	0	0	0
CEDA	0.20	2.55	15.00	100.0	21	0.05	0.01	5.14	4.99	3	0	97	0	0	0	0	0
CEMT	0.07	0.57	15.00	100.0	10	0.05	0.01	6.56	6.54	0	0	100	0	0	0	0	0
CRSQ	0.27	0.80	15.00	100.0	3	0.00	0.00	0.27	0.00	0	0	0	0	100	0	0	0
CRSR	0.02	2.91	15.00	100.0	29,214	0.00	0.00	0.02	0.00	0	0	0	0	100	0	0	0
CRTE	0.02	1.80	15.00	100.0	691	0.00	0.00	33.38	33.36	0	0	100	0	0	0	0	0
CSAC	0.00	0.06	15.00	100.0	350	0.00	0.00	0.00	0.00	0	0	71	0	29	0	0	0
CSFU	0.83	0.83	15.00	100.0	1	0.01	0.01	0.81	0.00	0	0	0	0	100	0	0	0
CSGM	0.03	3.93	15.00	100.0	16,516	0.01	0.00	0.03	0.01	0	0	26	0	74	0	0	0
CSIR	47.96	64.34	15.00	42.6	61	0.00	0.00	47.95	0.00	0	0	0	0	100	0	0	0
CSMI	0.01	0.11	15.00	100.0	11	0.01	0.00	0.01	0.00	0	0	0	0	100	0	0	0
CSNE	0.00	0.16	15.00	100.0	23,797	0.00	0.00	0.00	0.00	0	0	0	0	100	0	0	0
CSSF	0.01	0.18	15.00	100.0	38	0.01	0.00	0.34	0.33	0	0	98	0	2	0	0	0
CSSN	0.03	0.07	15.00	100.0	6	0.01	0.00	0.02	0.01	0	0	26	0	74	0	0	0
DATA	0.14	20.19	15.00	100.0	14,588	0.01	0.00	0.15	0.01	0	0	10	0	90	0	0	0
DISP	0.03	0.03	15.00	100.0	1	0.03	0.02	0.00	0.00	0	0	83	0	17	0	0	100
D2DA	1.42	7.37	15.00	100.0	15	0.01	0.00	1.70	0.30	0	0	18	0	82	0	0	0
ELON	0.24	1.01	15.00	100.0	10	0.04	0.01	0.44	0.23	0	0	53	0	47	0	0	10
EL00	0.06	0.31	15.00	100.0	40	0.00	0.00	0.06	0.00	0	0	0	0	100	0	0	0
EL10	0.04	0.12	15.00	100.0	9	0.00	0.00	0.03	0.00	0	0	0	0	100	0	0	0
EL20	0.15	1.07	15.00	100.0	41	0.00	0.00	0.15	0.00	0	0	0	0	100	0	0	0
EL22	0.14	0.80	15.00	100.0	50	0.00	0.00	0.13	0.00	0	0	0	0	100	0	0	0
EL29	0.11	0.44	15.00	100.0	21	0.00	0.00	0.30	0.19	0	0	64	0	36	0	0	0
EL30	0.07	0.38	15.00	100.0	14	0.00	0.00	0.07	0.00	0	0	0	0	100	0	0	0
EZ	0.46	3.99	15.00	100.0	324	0.00	0.00	0.46	0.00	0	0	0	0	100	0	0	0
IEOL	0.25	88.32	15.00	99.9	1,346	0.01	0.00	0.25	0.00	0	0	1	0	99	0	0	0
LLTP	0.01	0.01	15.00	100.0	1	0.00	0.00	0.01	0.00	0	0	25	0	75	0	0	100
LLXW	0.04	0.11	15.00	100.0	4	0.02	0.01	0.02	0.00	0	0	15	0	85	0	0	100
LPKD	0.00	0.01	15.00	100.0	3	0.00	0.00	0.00	0.00	0	0	0	0	100	0	0	0
LPKE	0.00	0.00	15.00	100.0	12	0.00	0.00	0.00	0.00	0	0	0	0	100	0	0	0
MSAS	0.17	1.19	15.00	100.0	69	0.00	0.00	0.17	0.00	0	0	0	0	100	0	0	0
MSA2	0.13	5.68	15.00	100.0	3,240	0.00	0.00	0.18	0.06	0	0	32	0	68	0	0	0
SA	0.48	8.66	15.00	100.0	39	0.04	0.00	388.03	387.59	1	0	99	0	0	0	0	3
STAR	0.00	0.01	15.00	100.0	21	0.00	0.00	0.00	0.00	0	0	32	0	68	0	0	5
TOSC	0.13	41.33	15.00	100.0	344,062	0.00	0.00	0.24	0.12	0	0	49	0	51	0	0	0
TOSS	0.40	32.06	15.00	100.0	2,558	0.04	0.01	0.40	0.04	0	0	8	0	92	0	0	0
TOS1	0.38	30.17	15.00	100.0	8,580	0.04	0.01	0.39	0.05	0	0	12	0	88	0	0	0
TOTAL:	0.12	3.72	15.00	100.0	469,835	0.00	0.00	0.03	0.22	0	0	65	0	35	0	0	0

Figure 3-12. Daily CICS Service Detail Report Sample Output

3.1.1.26 CICLMB: Monthly CICS Service Ranking Report

The Monthly CICS Service Ranking inquiry produces tabular reports which identifies the top 10 CICS applications with the worst average response time in a given month.

REPORT FORMAT:

This inquiry produces a tabular report which contains a summary section and a ranking section. The summary section shows the overall response distribution by transaction type, and indicates whether or not the service level objective has been met for the month. The ranking section identifies the 10 applications with the worst average response time and provides other pertinent information such as average wait time and percentage of wait in file control, journal control, terminal control, and temporary storage.

The service level objectives used in this report are defined by module #CICMOBJ in prefix.MICS.SOURCE and CICRPTGL in sharedprefix.MICS.SOURCE. See section 3.2 of this guide if you want to override the service level objectives. A sample output of this inquiry is shown in Figure 3-13.

INQUIRY ID:

CICLMB (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAPU.

DATA ELEMENTS USED:

- CAUABEND - Abnormal Termination Occurrences
- CAUCDSTx - Count Conv. Responses Within Limit x
- CAUCPUTM - Task CPU Time
- CAUCRSTM - Conv. Response Time Total
- CAUCTRN - Conv. Transactions Processed
- CAUDAMSC - Damaged Storage Chain
- CAUETRNL - Excessive Transactions Processed
- CAUFCWAT - Wait Time for CICS File Control
- CAUJCWAT - Wait Time for CICS Journal Control
- CAUMDSTx - Count Medium Responses Within Limit x
- CAUMRSTM - Medium Response Time Total

CAUMTRN - Medium Transactions Processed
 CAUMXTSK - Max Tasking Condition
 CAUPETTM - Task Dispatch Time
 CAURESTM - Transaction Residency Time
 CAUSDSTx - Count Short Responses Within Limit x
 CAUSOS - Short on Storage Condition
 CAUSRSTM - Short Response Time Total
 CAUSTRN - Short Transactions Processed
 CAUSUSTM - Time Transaction was on CICS Susp Chain
 CAUCWAT - Wait Time for CICS Terminal Control
 CAUDSTx - Count Responses Within Limit x
 CAUTLWAT - Wait Time for CICS (Total)
 CAUTRANS - Transactions Processed
 CAUTRSTM - Transaction Response Time Total
 CAUTSWAT - Wait Time for CICS Temp Storage
 CICAPU - CICS Application Unit ID
 CICSID - CICS System Identifier
 DAY - Day of Month
 MONTH - Month of Year
 YEAR - Year of Century
 ZONE - Time Zone
 SYSID - System Identifier

CALCULATIONS:

Average conv. response time = $CAUCRSTM/CAUSTRN$
 Average dispatch = $CAUPETTM/SUM(CAUTRANS, CAUETR N)$
 Average long response time = $CAULRSTM/CAULTRN$
 Average medium response time = $CAUMRSTM/CAUMTRN$
 Average response time = $CAUTRSTM/CAUTRANS$
 Average short response time = $CAUSRSTM/CAUSTRN$
 Average suspend = $CAUSOS/SUM(CAUTRANS, CAUETR N)$
 Average wait= $(CAURESTM-CAUPETTM)/SUM(CAUTRANS, CAUETR N)$
 Average CPU = $CAUCPUTM/CAUTRANS$;
 Other wait=total wait-SUM(CAUFCWAT, CAUJCWAT, CAUCWAT,
 CAUTSWAT)
 Percent FC wait = $CAUFCWAT/total\ wait * 100$
 Percent JC wait = $CAUJCWAT/total\ wait * 100$
 Percent OT wait = $Other\ wait/total\ wait * 100$
 Percent TC wait = $CAUCWAT/total\ wait * 100$
 Percent TS wait = $CAUTSWAT/total\ wait * 100$
 Total wait= $CAURESTM-CAUPETTM$;

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by
 entering the data base IDs. Enter multiple IDs
 with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICTACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit

definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%APPLOBJ- Define service level objective for individual applications. By default, the same objectives are set for each application. The objectives are defined in prefix.MICS.SOURCE(#CICMOBJ).

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GSL0 - Determine if SLO (service level objectives) has been met. No defaults.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

MONTHLY CICS SERVICE RANKING REPORT																		
your_company_name																		
FOR AUGUST, 1991																		
INQUIRY: CICLMB											RUN DATE: 06SEP91							
SYSTEM SUMMARY: 7 REGIONS																		
SLO				AVG	RESP	%OBJ	----- RESPONSE DISTRIBUTION -----											
MET	TYPE	# TRANSACTIONS	%	RESP	OBJ	MET	<1	<2	<3	<4	<5	<6	<7					
YES	TOTAL	644,548		0.32	15.00	100.0	95.6	2.9	1.1	0.3	0.0	0.0	0.0					
YES	SHORT	362,767	56.3	0.13	5.00	100.0	95.8	2.8	1.0	0.3	0.0	0.0	0.0					
YES	LONG	232,047	36.0	0.68	15.00	100.0	94.5	3.5	1.4	0.4	0.1	0.0	0.0					
YES	CONV	49,734	7.7	0.03	15.00	100.0	99.8	0.1	0.0	0.0	0.0	0.0	0.0					
** TOP 10 APPLICATIONS WITH WORST AVG RESPONSE TIME																		
RANK	APPL	AVG RESP	RESP OBJ	%OBJ MET	# TRANS	AVG DISPCH	AVG CPU	AVG WAIT	AVG SUSP	----- % WAIT -----				----- %TRANS -----				
										FC	JC	TC	TS	OTH	SOS	MXTSK	VIOL	ABND
1	TCSV	908.32	15.00	0.0	1	0.04	0.01	908.28	0.00	0	0	0	0	100	0	0	0	0
2	DSNC	700.14	15.00	66.7	3	1.03	0.55	699.11	1.67	0	0	0	0	100	0	0	0	0
3	OPID	691.24	15.00	83.3	6	0.05	0.00	691.19	690.88	0	0	50	0	50	0	0	0	0
4	LAON	71.69	15.00	63.6	11	0.98	0.01	70.71	2.39	0	0	0	0	100	0	0	0	0
5	GLST	42.70	15.00	89.9	247	0.05	0.03	42.65	34.15	0	0	44	0	55	0	0	0	1
6	ICST	13.80	15.00	100.0	1	0.15	0.01	13.66	12.62	0	0	0	0	100	0	0	0	0
7	RSFU	12.76	15.00	100.0	11	0.27	0.02	12.49	3.46	0	0	0	0	100	0	0	0	0
8	ISHI	12.62	15.00	100.0	1	0.15	0.02	12.47	10.00	0	0	0	0	100	0	0	0	0
9	DPRM	4.65	15.00	100.0	9	0.28	0.03	4.37	3.43	21	0	0	0	79	0	0	0	0
10	CSIR	2.56	15.00	97.3	1,479	0.00	0.00	2.55	0.04	0	1	0	0	99	0	0	0	0

Figure 3-13. Monthly CICS Service Ranking Report Sample Output

3.1.1.27 CICLME: Monthly CICS Service Summary Report

The Monthly CICS Service Summary Report shows the service level, primarily response times, provided by each CICS region on a given month. The report displays the actual service and the service objective, as well as flags indicating whether or not the service level objectives have been met. This report provides a quick summary of the service and activity performance by CICS, and enables you to quickly identify CICS regions which failed to meet your service level objectives.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for overall CICS service provided in the enterprise (summed across SYSIDs and CICSs), and CICS service provided by each system (summed across SYSIDs).

A sample output of this inquiry is shown in Figure 3-14. Each field in the report is described below:

SLO MET:	A flag indicating whether or not the service level objective has been met by CICS. For the overall summary line (SYSID=ALL CICSID=ALL), the flag is set to NO if any of the regions failed to meet its service level objectives. By default, this flag is set to YES if 95% of the transactions met the response objective. Otherwise, it is set to NO. You can override the default percentage or set the flag using actual value vs. objective in the %LDESLO user exit.
SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
AVG RESP ALL:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICLRT exit in prefix.MICS.PARMS.
AVG RESP SHORT:	The average response time for all short transactions (TRANTYPE=S as assigned by the CICLRT exit in prefix.MICS.PARMS).

AVG RESP MED:	The average response time for all medium transactions (TRANSTYPE=M as assigned by the CICLRT exit in prefix.MICS.PARMS).
AVG RESP LONG:	The average response time for all long transactions (TRANSTYPE=L as assigned by the CICLRT exit in prefix.MICS.PARMS).
AVG RESP CONV:	The average response time for all conversational transactions (TRANSTYPE=C as assigned by the CICLRT exit in prefix.MICS.PARMS).
TRANS:	The total number of transactions processed, excluding those assigned as TRANSTYPE X in your CICLRT exit in prefix.MICS.PARMS.
RESP OBJ:	The average response time objective for your installation. This number defaults to 2 seconds for all transactions; 1 for short transactions, 3 for medium transactions, and 5 for long transactions. You can override these values using either MICF execution-time parameters or report user exits.
%OBJ MET:	The percentage of transactions that met the response objective.
%AVAIL:	The CICS availability expressed in a percentage. This number is derived by dividing the total CICS uptime by the total report duration.
PEAK HOUR:	The hour during which the peak average response time occurred.
PEAK RESP:	The peak average hourly response time provided by CICS.
PEAK TRAN/SEC:	The transaction rate for the peak hour.
PEAK TRAN:	The total number of transactions processed in the peak hour.
PEAK %OBJ MET:	The percentage of transactions met the response objective during the peak hour.
INQUIRY ID:	
	CICLME (tabular report)

DATA SOURCE (file/timespan):

CICCSY01 in the MONTHS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
 CSYCDSTx - Count Conv. Responses Within Limit x
 CSYCTRN - Conversational Transactions Processed
 CSYETRN - Excessive Transactions Processed
 CSYLDSTx - Count Long Responses Within Limit x
 CSYLRSTM - Conversational Response Time Total
 CSYLRSTM - Long Response Time Total
 CSYLTRN - Long Transactions Processed
 CSYMDSTx - Count Medium Responses Within Limit x
 CSYMRSTM - Medium Response Time Total
 CSYMTRN - Medium Transactions Processed
 CSYSDSTx - Count Short Responses Within Limit x
 CSYSRSTM - Short Response Time Total
 CSYSTRN - Short Transactions Processed
 CSYTDSTx - Count Responses Within Limit x
 CSYTRANS - Transactions Processed
 CSYTRSTM - Transaction Response Time Total
 CSYUPTM - CICS Availability Time
 DAY - Day of Month
 DURATION - Recording Interval Time
 HOUR - Hour of Day
 MONTH - Month of Year
 SYSID - System Identifier
 YEAR - Year of Century
 ZONE - Time Zone

CALCULATIONS:

Average response time = $CSYTRSTM / CSYTRANS$
 Average short response time = $CSYSRSTM / CSYSTRN$
 Average medium response time = $CSYMRSTM / CSYMTRN$
 Average long response time = $CSYLRSTM / CSYLTRN$
 Average conv response time = $CSYCRSTM / CSYCTRN$
 Transaction per second = $CSYTRANS / DURATION$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

 CYCLE - Optionally, define CA MICS input file cycle

range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

MONTHLY CICS SERVICE SUMMARY REPORT															
your_company_name															
FOR AUGUST, 1991															
INQUIRY: CICLME															
RUN DATE: 06SEP91															
SLO	SYSID	CICSID	-----		AVG	RESP	-----		# TRANS	RESP	%OBJ	----- PEAK ACTIVITY -----			
MET			TOTAL	SHORT	MED	LONG	CONV		OBJ	MET	HOUR	RESP	TRAN/SEC	TRANS	%MET
YES	*ALL	*ALL	0.32	0.13	.	0.68	0.03	644,528	15.00	100.0	9	0.15	3.0	75,037	100.0
YES	PRD1	*ALL	0.32	0.13	.	0.68	0.03	644,528	15.00	100.0	9	0.15	3.0	75,037	100.0
YES		LPD1	0.12	0.13	.	0.13	0.01	469,815	15.00	100.0	8	0.10	15.8	56,911	100.0
YES		LPD2	0.13	0.22	.	0.13	0.08	77,933	15.00	100.0	10	0.14	2.8	10,161	100.0
YES		LPD3	0.34	0.23	.	0.34	0.19	68,652	15.00	100.0	10	0.14	2.6	9,283	100.0
YES		LPD4	0.26	0.20	.	0.39	0.02	2,606	15.00	100.0	15	0.16	0.1	334	100.0
YES		RPD1	0.61	0.35	.	0.62	0.17	24,056	15.00	100.0	11	0.70	0.8	3,032	100.0
YES		RPD2	0.27	0.21	.	0.42	0.05	1,288	15.00	100.0	13	0.09	0.1	234	100.0
YES		RPD3	0.27	0.21	.	0.42	0.05	1,288	15.00	100.0	13	0.09	0.1	234	100.0
TOTAL # REGIONS: 7															

Figure 3-14. Monthly CICS Service Summary Report Sample Output

3.1.1.28 CICLMF: Monthly CICS Service Detail Report

The Monthly CICS Service Detail Report shows the CICS service provided to individual applications on a given day. It identifies those applications for which CICS delivered unsatisfactory service. The report provides information that is pertinent to response analysis and can help you identify the areas in which the response delay may have occurred.

REPORT FORMAT:

This inquiry produces a tabular report which contains a summary section and a detail section. The summary section shows the response distribution by transaction type for all CICS regions found in the input. The detail section shows the response time and other pertinent information for individual applications. A detail section is generated for each combination of SYSID and CICSID.

A sample output of this inquiry is shown by Figure 3-15. Each field in this report is described below:

SYSTEM SUMMARY:

- SLO MET:** A flag indicating whether or not the service level objective has been met by CICS. For the overall summary line (SYSID=ALL CICSID=ALL), the flag is set to NO if any of the regions failed to meet its service level objectives. By default, this flag is set to YES if 95% of the transactions met the response objective. Otherwise, it is set to NO. You can override the default percentage or set the flag using actual value vs. objective in the %LDESLO user exit.
- TYPE:** Transaction type: all, short, medium, long, or conversational. Transaction type (TRANSTYPE) is assigned by the CICRLRT exit in prefix.MICS.PARMS.
- TRANS:** The total number of transactions processed, excluding those assigned as TRANSTYPE X in your CICRLRT exit in prefix.MICS.PARMS.
- AVG RESP:** The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).

RESP OBJ: The average response time objective for your installation. This number defaults to 2 seconds for all transactions; 1 for short transactions, 3 for medium transactions, and 5 for long transactions. You can override these values using either MICF execution-time parameters or report user exits.

%OBJ MET: The percentage of transactions that met the response objective.

RESPONSE
DISTRIBU- The response distributions for the 7 response
TION thresholds which you have specified in
prefix.MICS.PARMS(CICOPS).

DETAIL SECTION:

SYSID: System identification. This field is set to ALL for data that is summarized for all systems.

CICSID: CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.

APPL: Application identification.

AVG RESP: The average response time for all transactions processed for this application.

MAX RESP: The maximum response time received by the application.

RESP OBJ: The average response time objective for your installation. This number defaults to 2 seconds for all transactions; 1 for short transactions, 3 for medium transactions, and 5 for long transactions. You can override these values using either MICF execution-time parameters or report user exits.

%OBJ MET: The percentage of transactions that met the response objective.

TRANS: The total number of transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).

AVG
DISPATCH: The average dispatch time for the application.

AVG CPU:	The average TCB CPU time consumed by the application.
AVG WAIT:	The average wait time for the application. It is calculated as the difference between total response time and total dispatch time divided by total transactions.
AVG SUSP:	The average suspend time for the application.
%WAIT FC:	The percentage of time the application spent waiting for file control.
%WAIT JC:	The percentage of time the application spent waiting for journal control.
%WAIT TC:	The percentage of time the application spent waiting for terminal control.
%WAIT TS:	The percentage of time the application spent waiting for temporary storage.
%WAIT OTH:	The percentage of time the application spent waiting for other reasons, such as waiting for data base management systems (DB2, IMS) or buffer and strings.
%TRANS SOS:	The percentage of transactions processed for this application during short-on-storage condition.
%TRANS MXTSK:	The percentage of transactions processed for this application when CICS was at maximum task.
%TRANS VIOL	The percentage of transactions processed for this application that had a storage violation.
%TRANS ABND:	The percentage of transactions abend for this application.

INQUIRY ID:

CICLMF (tabular report)

DATA SOURCE (file/time-span):

CICCAU01 in the MONTHS time-span.

If the CICCAU (CICS Application Unit Activity) file is

not active, the CICCUS (CICS User Activity) file will be used for reporting. The variable CICT2 will be used in place of the variable CICAU.

DATA ELEMENTS USED:

CAUABEND - Abnormal Termination Occurrences
CAUCDSTx - Count Conv. Responses Within Limit x
CAUCPUTM - Task CPU Time
CAUCRSTM - Conv. Response Time Total
CAUCTRN - Conv. Transactions Processed
CAUDAMSC - Damaged Storage Chain
CAUETRNL - Excessive Transactions Processed
CAUFCWAT - Wait Time for CICS File Control
CAUJCWAT - Wait Time for CICS Journal Control
CAUMDSTx - Count Medium Responses Within Limit x
CAUMRSTM - Medium Response Time Total
CAUMTRNL - Medium Transactions Processed
CAUMXTSK - Max Tasking Condition
CAUPETTM - Task Dispatch Time
CAURESTM - Transaction Residency Time
CAUSDSTx - Count Short Responses Within Limit x
CAUSOS - Short on Storage Condition
CAUSRSTM - Short Response Time Total
CAUSTRNL - Short Transactions Processed
CAUSUSTM - Time Transaction was on CICS Susp Chain
CAUTCWAT - Wait Time for CICS Terminal Control
CAUTDSTx - Count Responses Within Limit x
CAUTLWAT - Wait Time for CICS (Total)
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
CAUTSWAT - Wait Time for CICS Temp Storage
CICAU - CICS Application Unit ID
CICSID - CICS System Identifier
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average conv. response time = CAUCRSTM/CAUCTRN
Average dispatch = CAUPETTM/SUM(CAUTRANS,CAUETRNL);
Average long response time = CAULRSTM/CAULTRNL
Average medium response time = CAUMRSTM/CAUMTRNL
Average response time = CAUTRSTM/CAUTRANS
Average short response time = CAUSRSTM/CAUSTRNL
Average suspend = CAUSOS/SUM(CAUTRANS,CAUETRNL);

```
Average wait= (CAURESTM-CAUPETTM)/SUM(CAUTRANS,CAUETRNM);
Average CPU = CAUCPUTM/CAUTRANS;
Other wait=total wait-SUM(CAUFCWAT,CAUJCWAT,CAUTCWAT,
                          CAUTSWAT);
Percent FC wait = CAUFCWAT/total wait * 100;
Percent JC wait = CAUJCWAT/total wait * 100;
Percent OT wait = Other wait/total wait * 100;
Percent TC wait = CAUTCWAT/total wait * 100;
Percent TS wait = CAUTSWAT/total wait * 100;
Total wait= CAURESTM-CAUPETTM;
```

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

FILE ID- Optionally, select the CICCUS file as input

when the CICC AU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICTACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICTRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%APPLOBJ- Define service level objective for individual applications. By default, the same objectives are set for each application. The objectives are defined in prefix.MICS.SOURCE(#CICMOBJ).

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICTRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1 MICF Inquiries

MONTHLY CICS SERVICE DETAIL REPORT																	
your_company_name																	
FOR AUGUST, 1991																	
INQUIRY: CICLMF										RUN DATE: 06SEP91							
----- SYSID=PRD2 CICSID=LCP1 -----																	
APPL	AVG RESP	MAX RESP	RESP OBJ	%OBJ MET	TRANS	AVG DISPCH	AVG CPU	AVG WAIT	AVG SUSP	FC	JC	TC	TS	OTH	SOS	MXTSK	%TRANS VIOL ABND
AAON	34.10	34.10	15.00	100.0	1	0.36	0.08	33.74	0.00	0	0	0	0	100	0	0	0
ACFM	0.02	0.13	15.00	100.0	163	0.02	0.01	53.56	53.55	0	0	100	0	0	0	0	0
CATD	0.07	3.80	15.00	100.0	15,759	0.02	0.01	0.06	0.00	0	0	0	0	100	0	0	0
CCMF	0.00	0.04	15.00	100.0	233	0.00	0.00	0.00	0.00	0	0	0	0	100	0	0	0
CEDA	0.20	2.55	15.00	100.0	21	0.05	0.01	5.14	4.99	3	0	97	0	0	0	0	0
CEMT	0.07	0.57	15.00	100.0	10	0.05	0.01	6.56	6.54	0	0	100	0	0	0	0	0
CRSQ	0.27	0.80	15.00	100.0	3	0.00	0.00	0.27	0.00	0	0	0	0	100	0	0	0
CRSR	0.02	2.91	15.00	100.0	29,214	0.00	0.00	0.02	0.00	0	0	0	0	100	0	0	0
CRTE	0.02	1.80	15.00	100.0	691	0.00	0.00	33.38	33.36	0	0	100	0	0	0	0	0
CSAC	0.00	0.06	15.00	100.0	350	0.00	0.00	0.00	0.00	0	0	71	0	29	0	0	0
CSFU	0.83	0.83	15.00	100.0	1	0.01	0.01	0.81	0.00	0	0	0	0	100	0	0	0
CSGM	0.03	3.93	15.00	100.0	16,516	0.01	0.00	0.03	0.01	0	0	26	0	74	0	0	0
CSIR	47.96	64.34	15.00	42.6	61	0.00	0.00	47.95	0.00	0	0	0	0	100	0	0	0
CSMI	0.01	0.11	15.00	100.0	11	0.01	0.00	0.01	0.00	0	0	0	0	100	0	0	0
CSNE	0.00	0.16	15.00	100.0	23,797	0.00	0.00	0.00	0.00	0	0	0	0	100	0	0	0
CSSF	0.01	0.18	15.00	100.0	38	0.01	0.00	0.34	0.33	0	0	98	0	2	0	0	0
CSSN	0.03	0.07	15.00	100.0	6	0.01	0.00	0.02	0.01	0	0	26	0	74	0	0	0
DATA	0.14	20.19	15.00	100.0	14,588	0.01	0.00	0.15	0.01	0	0	10	0	90	0	0	0
DISP	0.03	0.03	15.00	100.0	1	0.03	0.02	0.00	0.00	0	0	83	0	17	0	0	100
D2DA	1.42	7.37	15.00	100.0	15	0.01	0.00	1.70	0.30	0	0	18	0	82	0	0	0
ELON	0.24	1.01	15.00	100.0	10	0.04	0.01	0.44	0.23	0	0	53	0	47	0	0	10
EL00	0.06	0.31	15.00	100.0	40	0.00	0.00	0.06	0.00	0	0	0	0	100	0	0	0
EL10	0.04	0.12	15.00	100.0	9	0.00	0.00	0.03	0.00	0	0	0	0	100	0	0	0
EL20	0.15	1.07	15.00	100.0	41	0.00	0.00	0.15	0.00	0	0	0	0	100	0	0	0
EL22	0.14	0.80	15.00	100.0	50	0.00	0.00	0.13	0.00	0	0	0	0	100	0	0	0
EL29	0.11	0.44	15.00	100.0	21	0.00	0.00	0.30	0.19	0	0	64	0	36	0	0	0
EL30	0.07	0.38	15.00	100.0	14	0.00	0.00	0.07	0.00	0	0	0	0	100	0	0	0
EZ	0.46	3.99	15.00	100.0	324	0.00	0.00	0.46	0.00	0	0	0	0	100	0	0	0
IEOL	0.25	88.32	15.00	99.9	1,346	0.01	0.00	0.25	0.00	0	0	1	0	99	0	0	0
LLTP	0.01	0.01	15.00	100.0	1	0.00	0.00	0.01	0.00	0	0	25	0	75	0	0	100
LLXW	0.04	0.11	15.00	100.0	4	0.02	0.01	0.02	0.00	0	0	15	0	85	0	0	100
LPKD	0.00	0.01	15.00	100.0	3	0.00	0.00	0.00	0.00	0	0	0	0	100	0	0	0
LPKE	0.00	0.00	15.00	100.0	12	0.00	0.00	0.00	0.00	0	0	0	0	100	0	0	0
MSAS	0.17	1.19	15.00	100.0	69	0.00	0.00	0.17	0.00	0	0	0	0	100	0	0	0
MSA2	0.13	5.68	15.00	100.0	3,240	0.00	0.00	0.18	0.06	0	0	32	0	68	0	0	0
SA	0.48	8.66	15.00	100.0	39	0.04	0.00	388.03	387.59	1	0	99	0	0	0	0	3
STAR	0.00	0.01	15.00	100.0	21	0.00	0.00	0.00	0.00	0	0	32	0	68	0	0	5
TOSC	0.13	41.33	15.00	100.0	344,062	0.00	0.00	0.24	0.12	0	0	49	0	51	0	0	0
TOSS	0.40	32.06	15.00	100.0	2,558	0.04	0.01	0.40	0.04	0	0	8	0	92	0	0	0
TOS1	0.38	30.17	15.00	100.0	8,580	0.04	0.01	0.39	0.05	0	0	12	0	88	0	0	0
TOTAL:	0.12	3.72	15.00	100.0	469,835	0.00	0.00	0.03	0.22	0	0	65	0	35	0	0	0

Figure 3-15. Monthly CICS Service Detail Report Sample Output

3.1.2 Workload Analysis Inquiries

Workload analysis inquiries provide activity information at the system and user level to help you monitor and manage CICS workload. They are available for daily, weekly, and monthly reporting periods. The CICS Analyzer distributes the following workload analysis inquiries:

Catalog Group	Inquiry ID	Inquiry Name	Report format	Run-time execution parameters	User exits
WORKLOAD	CICCDH	Daily CICS Workload Ranking	color graphic		
	CICCDI	Daily CICS Workload	color graphic		
	CICCDJ	Daily CICS Workload All Regions	color graphic		
	CICCDK	Daily CICS Workload All Systems	color graphic		
	CICCDL	Daily CICS Workload Reports	color graphic		
	CICCMH	Monthly CICS Workload Ranking	color graphic		
	CICCMJ	Monthly CICS Workload	color graphic		
	CICCMK	Monthly CICS Workload All Regions	color graphic		
	CICCMN	Monthly CICS Workload All Systems	color graphic		
	CICCMO	Monthly CICS Workload Trend	color graphic		
	CICCPH	Monthly CICS Workload Reports	color graphic		
	CICCPJ	Weekly CICS Workload	color graphic		
	CICCPK	Weekly CICS Workload All Regions	color graphic		
	CICCPN	Weekly CICS Workload All Systems	color graphic		
	CICCPW	Weekly CICS Workload Trend	color graphic		
	CICCPX	Weekly CICS Workload Reports	color graphic		
	CICPDH	Daily CICS Workload Ranking	printer graphic		
	CICPDI	Daily CICS Workload	printer graphic		
	CICPDL	Daily CICS Workload Reports	printer graphic		
	CICPMH	Monthly CICS Workload Ranking	printer graphic		
	CICPMJ	Monthly CICS Workload	printer graphic		
	CICPMK	Monthly CICS Workload Reports	printer graphic		
	CICPMN	Weekly CICS Workload	printer graphic		
	CICPMO	Weekly CICS Workload Reports	printer graphic		
	CICLDH	Daily CICS Workload Ranking Report	tabular report		
	CICLDL	Daily CICS Workload Summary Report	tabular report		
	CICLDM	Daily CICS Workload Detail Report	tabular report		
	CICLMH	Monthly CICS Workload Ranking Report	tabular report		
	CICLML	Monthly CICS Workload Summary Report	tabular report		
	CICLMM	Monthly CICS Workload Detail Report	tabular report		

Figure 3-16. Distributed Workload Analysis Inquiries

To provide report flexibility, the CICS Analyzer supplies MICF execution run time parameters as well as user exits for most of the workload analysis inquiries. You can use either method to tailor the inquiries to suit your reporting needs.

The workload analysis inquiries are described individually in the following sections:

- 1 - CICCDB: Daily CICS Workload Ranking
- 2 - CICCDB: Daily CICS Workload
- 3 - CICCDB: Daily CICS Workload All Regions
- 4 - CICCDB: Daily CICS Workload All Systems
- 5 - CICCDB: Daily CICS Workload Reports
- 6 - CICCDB: Monthly CICS Workload Ranking
- 7 - CICCDB: Monthly CICS Workload
- 8 - CICCDB: Monthly CICS Workload All Regions
- 9 - CICCDB: Monthly CICS Workload All Systems
- 10- CICCDB: Monthly CICS Workload Trend
- 11- CICCDB: Monthly CICS Workload Reports
- 12- CICCDB: Weekly CICS Workload
- 13- CICCDB: Weekly CICS Workload All Regions
- 14- CICCDB: Weekly CICS Workload All Systems
- 15- CICCDB: Weekly CICS Workload Trend
- 16- CICCDB: Weekly CICS Workload Reports
- 17- CICCDB: Daily CICS Workload Ranking
- 18- CICCDB: Daily CICS Workload
- 19- CICCDB: Daily CICS Workload Reports
- 20- CICCDB: Monthly CICS Workload Ranking
- 21- CICCDB: Monthly CICS Workload
- 22- CICCDB: Monthly CICS Workload Reports
- 23- CICCDB: Weekly CICS Workload
- 24- CICCDB: Weekly CICS Workload Reports
- 25- CICCDB: Daily CICS Workload Ranking Report
- 26- CICCDB: Daily CICS Workload Summary Report
- 27- CICCDB: Daily CICS Workload Detail Report
- 28- CICCDB: Monthly CICS Workload Ranking Report
- 29- CICCDB: Monthly CICS Workload Summary Report
- 30- CICCDB: Monthly CICS Workload Detail Report

3.1.2.1 CICCDH: Daily CICS Workload Ranking

The Daily CICS Workload Ranking inquiry produces a color graph which identifies the top 10 CICS applications with the highest transaction volume in a given day.

REPORT FORMAT:

This inquiry produces a side-by-side horizontal bar chart which shows the top 10 applications with the highest transaction volume. The chart also shows the average response time for each the 10 applications in the same order as the workload ranking. This chart is available in color graphic format only.

INQUIRY ID:

CICCDH (color graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAPU.

DATA ELEMENTS USED:

CICAPU - CICS Application Unit ID
CICSID - CICS System Identifier
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on

the graph. The default value is NO.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.2 CICCDI: Daily CICS Workload

The Daily CICS Workload inquiry produces a color graph which shows the hourly CICS activity in a given day by transaction type. It enables you to identify the peak hour in CICS activity and the workload characteristics during that time period.

REPORT FORMAT:

This inquiry produces a vertical bar chart with subgroups that shows the hourly transaction volume. Each bar is subdivided by transaction type to show the workload characteristics at various hour during the day. This inquiry produces output in color graphic format only. You can produce the same chart in printer graphic format by executing the MICF inquiry CICPDI.

INQUIRY ID:

CICCDI (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
CSYCTRN - Conv. Transactions Processed
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs

with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit

definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.3 CICCDJ: Daily CICS Workload All Regions

The Daily CICS Workload All Regions graph shows the total transaction volume processed by each region in a given system. It identifies regions with high workload and shows the workload distributions in the system. In addition, the graph highlights regions that have missed the service level objective to alert you of potential performance problems.

REPORT FORMAT:

This inquiry produces a vertical bar chart that shows the total activity for each region in a given system (SYSID or a group of SYSIDs). The bars are displayed in descending order by transaction volume. Regions that have missed the user-defined service level objective are shown in a different color than those meeting the service level objectives. This chart is available in color graphic format only.

INQUIRY ID:

CICCDJ (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan.

DATA ELEMENTS USED:

CICSID	- CICS System Identifier
CSYCDSTx	- Count Conv. Responses Within Limit x
CSYCRSTM	- Conversational Response Time Total
CSYCTRN	- Conversational Transactions Processed
CSYLDSTx	- Count Long Responses Within Limit x
CSYLRSTM	- Long Response Time Total
CSYLTRN	- Long Transactions Processed
CSYMDSTx	- Count Medium Responses Within Limit x
CSYMRSTM	- Medium Response Time Total
CSYMTRN	- Medium Transactions Processed
CSYSDSTx	- Count Short Responses Within Limit x
CSYSRSTM	- Short Response Time Total
CSYSTRN	- Short Transactions Processed
CSYTDSTx	- Count Responses Within Limit x
CSYTRANS	- Transactions Processed
CSYTRSTM	- Total Response Time All Functions
DAY	- Day of Month
MONTH	- Month of Year
SYSID	- System Identifier
YEAR	- Year of Century

ZONE - Time Zone

CALCULATIONS:

Percent distribution = CSYTDSTx / CSYTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response

seconds.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is `TERMINAL`. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is `NO`.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module `CICRPTGL` in `sharedprefix.MICS.SOURCE`. The following exits are available to this inquiry:

%APPLOBJ- Define service level objective for individual applications. By default, the same objectives are set for each application. The objectives are defined in `prefix.MICS.SOURCE(#CICMOBJ)`.

%GAPPL - Define your CICS application based on any variable in the input file. The default is `CICAPU`.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable `CENTER` which defines the logical grouping of `SYSIDs`. `CENTER` is set to `SYSID` by default.

%GSLO - Determine if `SLO` (service level objectives) has been met. No defaults.

%GSYS - Select `SYSIDs` for report.

In addition to the above exits, the module `CICRPTGL` also contains macro variables which set default values for

service level objectives. Please see Section 3.2 to override the default values.

3.1.2.4 CICCDK: Daily CICS Workload All Systems

The Daily CICS Workload All Systems graph shows the total transaction volume processed by each system (SYSID) or data center (group of SYSIDs) within an enterprise. It identifies systems with high workload and shows the workload distributions in the enterprise. The graph also highlights systems that have missed the service level objective to alert you of potential performance problems.

REPORT FORMAT:

This inquiry produces a block bar chart that shows the total activity for each system (SYSID) or data center (group of SYSIDs). The bars are displayed in descending order by transaction volume. Systems that have missed the user-defined service level objective are shown in red, while the rest are shown in green. Each chart can display up to 7 systems or data centers. A bar chart is produced when there are more than 7 values. This chart is available in color graphic format only.

INQUIRY ID:

CICCDK (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan.

DATA ELEMENTS USED:

CSYTRANS - Transactions Processed
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
CSYCTRN - Conversational Transactions Processed
CSYTRSTM - Total Response Time All Functions
CSYSRSTM - Short Response Time Total
CSYMRSTM - Medium Response Time Total
CSYLRSTM - Long Response Time Total
CSYCRSTM - Conversational Response Time Total
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = $CSYTRSTM / CSYTRANS$
Average short response time = $CSYSRSTM / CSYSTRN$
Average medium response time = $CSYMRSTM / CSYMTRN$
Average long response time = $CSYLRSTM / CSYLTRN$
Average conv. response time = $CSYCRSTM / CSYCTRN$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

- SYSID** - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID** - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- CENTER** - Optionally, specify the CENTER to be reported.
If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.
- DATE** - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.
- ZONE** - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- BY** - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.
- RESP** - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is `TERMINAL`. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is `NO`.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module `CICRPTGL` in `sharedprefix.MICS.SOURCE`. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable `CENTER` which defines the logical grouping of `SYSIDs`. `CENTER` is set to `SYSID` by default.

%GSLO - Determine if `SLO` (service level objectives) has been met. No defaults.

%GSYS - Select `SYSIDs` for report.

In addition to the above exits, the module `CICRPTGL` also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.5 CICCD1: Daily CICS Workload Reports

The Daily CICS Workload Reports monitor CICS transaction volume on a daily basis and identify peak hour workload for each CICS system.

REPORT FORMAT:

This inquiry produces bar charts by system ID and CICS ID showing the number of transactions ended within each hour. The output of this inquiry is in color graphic format. You can also produce the bar charts in printer graphic format by executing the inquiry CICPD1.

INQUIRY ID:

CICCD1 (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan

DATA ELEMENTS USED:

CSYTRANS - Total Transactions Ended

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

- UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
- CYCLE - Optionally, define CA MICS input file cycle range.
- CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

USER EXITS:

None

3.1.2.6 CICCMH: Monthly CICS Workload Ranking

The Monthly CICS Workload Ranking inquiry produces a color graph which identifies the top 10 CICS applications with the highest transaction volume in a given month.

REPORT FORMAT:

This inquiry produces a side-by-side horizontal bar chart which shows the top 10 applications with the highest transaction volume. The chart also shows the average response time for each the 10 applications, in the same order as the workload ranking. This chart is available in color graphic format only.

INQUIRY ID:

CICCMH (color graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAPU.

DATA ELEMENTS USED:

CICAPU - CICS Application Unit ID
CICSID - CICS System Identifier
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on

the graph. The default value is NO.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.7 CICCMI: Monthly CICS Workload

The Monthly CICS Workload inquiry produces a color graph which shows CICS activity by transaction type (short, medium, and long) for the past 6 months. The graph shows the trend in total work volume as well as workload characteristics.

REPORT FORMAT:

This inquiry produces a vertical bar chart displays the monthly workload for the past 24 months. Each bar is subdivided by transaction type to show the workload characteristics. The output of this inquiry is in color graphic format. However, you can produce the same chart in printer graphic format by executing the MICF inquiry CICIPMI.

INQUIRY ID:

CICCMI (color graphic format)

DATA SOURCE (file/timespan):

CICCSY from the monthly history tape.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.8 CICCMI: Monthly CICS Workload All Regions

The Monthly CICS Workload All Regions graph shows the total transaction volume processed by each region in a given system (SYSID or a group of SYSIDs). It identifies regions with high workload and shows the workload distributions in the system. In addition, the graph highlights regions that have missed the service level objective to alert you of potential performance problems.

REPORT FORMAT:

This inquiry produces a vertical bar chart that shows the total activity for each region in a given system (SYSID or a group of SYSIDs). The bars are displayed in descending order by transaction volume. Regions that have missed the user-defined service level objective are shown in a different color than those meeting the service level objectives. This chart is available in color graphic format only.

INQUIRY ID:

CICCMI (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the MONTHS timespan.

DATA ELEMENTS USED:

CICSID	- CICS System Identifier
CSYCDSTx	- Count Conv. Responses Within Limit x
CSYCRSTM	- Conversational Response Time Total
CSYCTRN	- Conversational Transactions Processed
CSYLDSTx	- Count Long Responses Within Limit x
CSYLRSTM	- Long Response Time Total
CSYLTRN	- Long Transactions Processed
CSYMDSTx	- Count Medium Responses Within Limit x
CSYMRSTM	- Medium Response Time Total
CSYMTRN	- Medium Transactions Processed
CSYSDSTx	- Count Short Responses Within Limit x
CSYSRSTM	- Short Response Time Total
CSYSTRN	- Short Transactions Processed
CSYTDSTx	- Count Responses Within Limit x
CSYTRANS	- Transactions Processed
CSYTRSTM	- Total Response Time All Functions
DAY	- Day of Month
MONTH	- Month of Year
SYSID	- System Identifier
YEAR	- Year of Century

ZONE - Time Zone

CALCULATIONS:

Percent distribution = $CSYTDSTx / CSYTRANS$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG

which produces ranking by average response seconds.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%APPLOBJ- Define service level objective for individual applications. By default, the same objectives are set for each application. The objectives are defined in prefix.MICS.SOURCE(#CICMOBJ).

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSLO - Determine if SLO (service level objectives) has been met. No defaults.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.9 CICCCK: Monthly CICS Workload All Systems

The Monthly CICS Workload All Systems graph shows the total transaction volume processed by each system (SYSID) or data center (group of SYSIDs) within an enterprise. It identifies systems with high workload and shows the workload distribution within the enterprise. The graph also highlights systems that have missed the service level objective to alert you of potential performance problems.

REPORT FORMAT:

This inquiry produces a block bar chart that shows the total activity for each system (SYSID) or data center (group of SYSIDs). The bars are displayed in descending order by transaction volume. Systems that have missed the user-defined service level objective are shown in red, while the rest are shown in green. Each chart can display up to 7 systems or data centers. A bar chart is produced when there are more than 7 values. This chart is available in color graphic format only.

INQUIRY ID:

CICCCK (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the MONTHS timespan.

DATA ELEMENTS USED:

CSYTRANS - Transactions Processed
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
CSYCTRN - Conversational Transactions Processed
CSYTRSTM - Total Response Time All Functions
CSYSRSTM - Short Response Time Total
CSYMRSTM - Medium Response Time Total
CSYLRSTM - Long Response Time Total
CSYCRSTM - Conversational Response Time Total
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Percent distribution = CSYTDSTx / CSYTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

RESP - Optionally, select the type of responses to be

ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSL0 - Determine if SLO (service level objectives) has been met. No defaults.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.10 CICC MN: Monthly CICS Workload Trend

This inquiry is unavailable at this time.

3.1.2.11 CICCMI: Monthly CICS Workload Reports

The Monthly CICS Workload Reports monitor CICS transaction volume trend by ZONE in a six-month period.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the number of transactions ended per ZONE in a six-month period. Any month for which data is unavailable is omitted from the charts.

This inquiry produces bar charts in color graphic format.

INQUIRY ID:

CICCM1 (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01-CICCSY06 in the MONTHS timespan

DATA ELEMENTS USED:

CSYTRANS - Total Transactions Ended

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to

be summarized together.

USER EXITS:

None

3.1.2.12 CICCWI: Weekly CICS Workload

The Weekly CICS Workload Reports monitor CICS transaction volume trend in a five-day period and identify peak hour workload between 8 a.m. and 5 p.m. for each CICS system.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the number of transactions ended within each hour from 8 a.m. to 5 p.m. in a five-day period. The charts are based on data summarized at the DATE and HOUR levels. Data from Saturday and Sunday is excluded. The output of this inquiry is in color graphic format. However, you can produce the same charts in printer graphic format by executing the MICF inquiry CICPW1.

INQUIRY ID:

CICCWI (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 - 07 from the DAYS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle

range.

For this inquiry, the default is the 01 cycle.

- SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- DATE - Optionally, enter start of the report period in ddmmyy format.
For this inquiry, the default is the last seven days. Use the ZONE parameter to exclude weekends and holidays.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.
- DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.
- FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.13 CICCWJ: Weekly CICS Workload All Regions

The Weekly CICS Workload All Regions graph shows the total transaction volume processed by each region in a given system. It identifies regions with high workload and shows the workload distributions in the system. In addition, the graph highlights regions that have missed the service level objective to alert you of potential performance problems.

REPORT FORMAT:

This inquiry produces a vertical bar chart that shows the total activity for each region in a given system (SYSID or a group of SYSIDs). The bars are displayed in descending order by transaction volume. Regions that have missed the user-defined service level objective are shown in a different color than those meeting the service level objectives. This chart is available in color graphic format only.

INQUIRY ID:

CICCWJ (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the WEEKS timespan.

DATA ELEMENTS USED:

CICSID	- CICS System Identifier
CSYCDSTx	- Count Conv. Responses Within Limit x
CSYCRSTM	- Conversational Response Time Total
CSYCTRN	- Conversational Transactions Processed
CSYLDSTx	- Count Long Responses Within Limit x
CSYLRSTM	- Long Response Time Total
CSYLTRN	- Long Transactions Processed
CSYMDSTx	- Count Medium Responses Within Limit x
CSYMRSTM	- Medium Response Time Total
CSYMTRN	- Medium Transactions Processed
CSYSDSTx	- Count Short Responses Within Limit x
CSYSRSTM	- Short Response Time Total
CSYSTRN	- Short Transactions Processed
CSYTDSTx	- Count Responses Within Limit x
CSYTRANS	- Transactions Processed
CSYTRSTM	- Total Response Time All Functions
ENDTS	- End Time Stamp
SYSID	- System Identifier
YEAR	- Year of Century

WEEK - Week of Year
ZONE - Time Zone

CALCULATIONS:

Percent distribution = CSYTDSTx / CSYTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.
For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter start of the report period in ddmmmyy format.

For this inquiry, the default is the previous week.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is `TERMINAL`. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is `NO`.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module `CICRPTGL` in `sharedprefix.MICS.SOURCE`. The following exits are available to this inquiry:

%APPLOBJ- Define service level objective for individual applications. By default, the same objectives are set for each application. The objectives are defined in `prefix.MICS.SOURCE(#CICMOBJ)`.

%GAPPL - Define your CICS application based on any variable in the input file. The default is `CICAPU`.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable `CENTER` which defines the logical grouping of `SYSIDs`. `CENTER` is set to `SYSID` by default.

%GSLO - Determine if `SLO` (service level objectives) has been met. No defaults.

%GSYS - Select `SYSIDs` for report.

In addition to the above exits, the module `CICRPTGL` also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.14 CICCWK: Weekly CICS Workload All Systems

The Weekly CICS Workload All Systems graph shows the total transaction volume processed by each system (SYSID) or data center (group of SYSIDs) within an enterprise. It identifies systems with high workload and shows the workload distribution within the enterprise. The graph also highlights systems that have missed the service level objective to alert you of potential performance problems.

REPORT FORMAT:

This inquiry produces a block bar chart that shows the total activity for each system (SYSID) or data center (group of SYSIDs). The bars are displayed in descending order by transaction volume. Systems that have missed the user-defined service level objective are shown in red, while the rest are shown in green. Each chart can display up to 7 systems or data centers. A bar chart is produced when there are more than 7 values. This chart is available in color graphic format only.

INQUIRY ID:

CICCWK (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the WEEKS timespan.

DATA ELEMENTS USED:

CSYTRANS - Transactions Processed
 CSYSTRN - Short Transactions Processed
 CSYMTRN - Medium Transactions Processed
 CSYLTRN - Long Transactions Processed
 CSYCTRN - Conversational Transactions Processed
 CSYTRSTM - Total Response Time All Functions
 CSYSRSTM - Short Response Time Total
 CSYMRSTM - Medium Response Time Total
 CSYLRSTM - Long Response Time Total
 CSYCRSTM - Conversational Response Time Total
 WEEK - Week of Year
 YEAR - Year of Century
 ZONE - Time Zone
 SYSID - System Identifier

CALCULATIONS:

Percent distribution = CSYTDSTx / CSYTRANS

EXECUTION-TIME PARAMETERS:

- UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
- CYCLE - Optionally, define CA MICS input file cycle range.
For this inquiry, the default is the 01 cycle.
- SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- CENTER - Optionally, specify the CENTER to be reported.
If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.
- DATE - Optionally, enter start of the report period in ddmmmyy format.
For this inquiry, the default is the previous week.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.
- RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.
- DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by

MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

- %GCIC - Select CICSIDs for report.
- %GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.
- %GSLO - Determine if SLO (service level objectives) has been met. No defaults.
- %GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.2.15 CICCWN: Weekly CICS Workload Trend

This inquiry is unavailable at this time.

3.1.2.16 CICCW1: Weekly CICS Workload Reports

The Weekly CICS Workload Reports monitor CICS transaction volume trend in a five-day period and identify peak hour workload between 8 a.m. and 5 p.m. for each CICS system.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the number of transactions ended within each hour from 8 a.m. to 5 p.m. in a five-day period. The charts are based on data summarized at the DATE and HOUR levels. Data from Saturday and Sunday is excluded. The output of this inquiry is in color graphic format. However, you can produce the same charts in printer graphic format by executing the MICF inquiry CICPW1.

INQUIRY ID:

CICCW1 (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01-CICCSY07 in the DAYS timespan

Data is selected for HOURS 8-16, excluding DAYNAME='SAT' and DAYNAME='SUN'.

DATA ELEMENTS USED:

CSYTRANS - Total Transactions Ended

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported.

If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

HOURL - Optionally, specify range of hours.

DAY - Optionally, specify range of weekdays (SUN MON TUE WED THU FRI SAT).

USER EXITS:

None

3.1.2.17 CICPDH: Daily CICS Workload Ranking

The Daily CICS Workload Ranking inquiry produces a printer graph which identifies the top 10 CICS application in transaction volume.

REPORT FORMAT:

This inquiry produces a horizontal bar chart which shows the 10 applications with the highest workload.

A sample output of this inquiry is shown in Figure 3-17.

INQUIRY ID:

CICPDH (printer graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAPU.

One graph produced for each date and each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICAPU	- CICS Application Unit ID
CICSID	- CICS System Identifier
CAUTRANS	- Transactions Processed
CAUSTRN	- Short Transactions Processed
CAUMTRN	- Medium Transactions Processed
CAULTRN	- Long Transactions Processed
CAUCTRN	- Conv. Transactions Processed
DAY	- Day of Month
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

TRAN - Optionally, select the type of transaction to be ranked. The response types are TOTAL, SHORT, MED, LONG, or CONV. The default is TOTAL.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

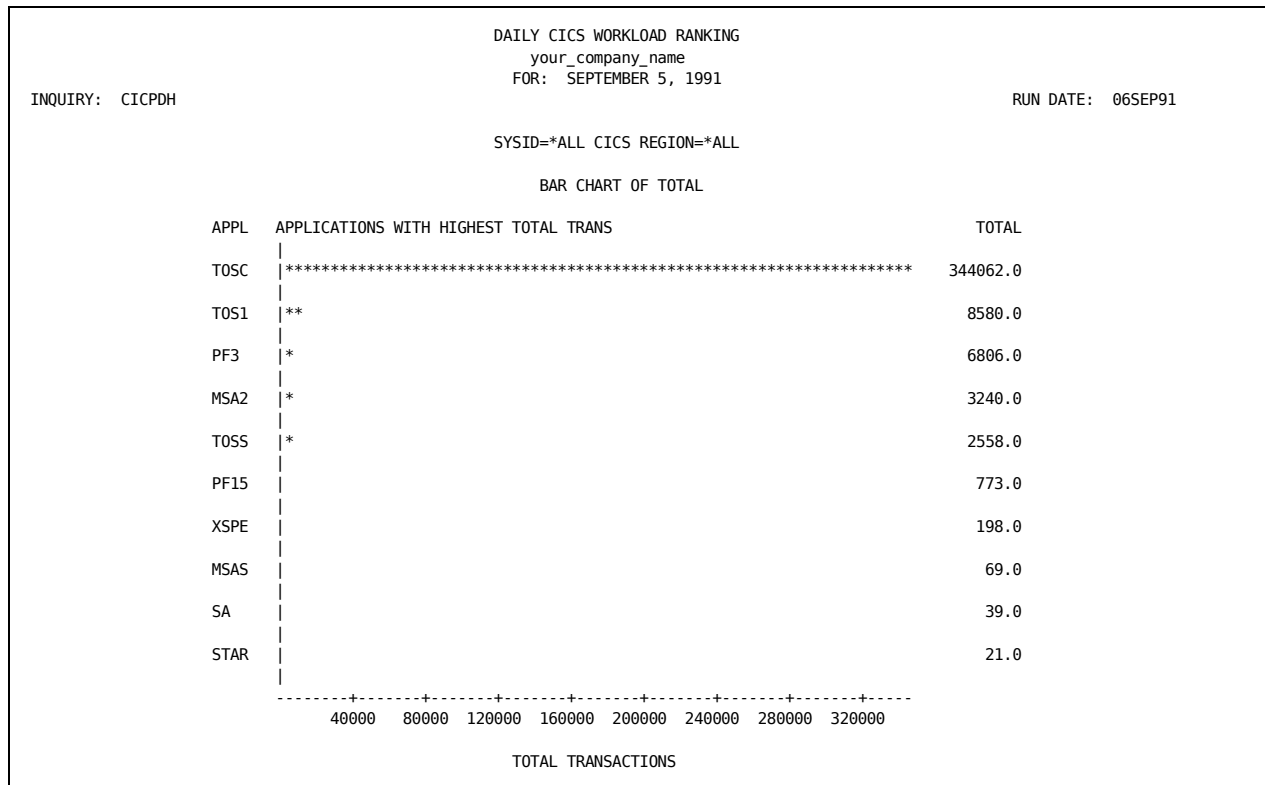


Figure 3-17. Daily CICS Workload Ranking Sample Output

3.1.2.18 CICPDI: Daily CICS Workload

The Daily CICS Workload inquiry produces multiple charts which show CICS workload by system (SYSID) and regions (CICSID). The charts show the workload distribution among systems.

REPORT FORMAT:

This inquiry produce horizontal bar charts which shows the total transaction volume per system. A sample output of this inquiry is shown in Figure 3-18.

INQUIRY ID:

CICPDI (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
CSYCTRN - Conv. Transactions Processed
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

USER EXITS:

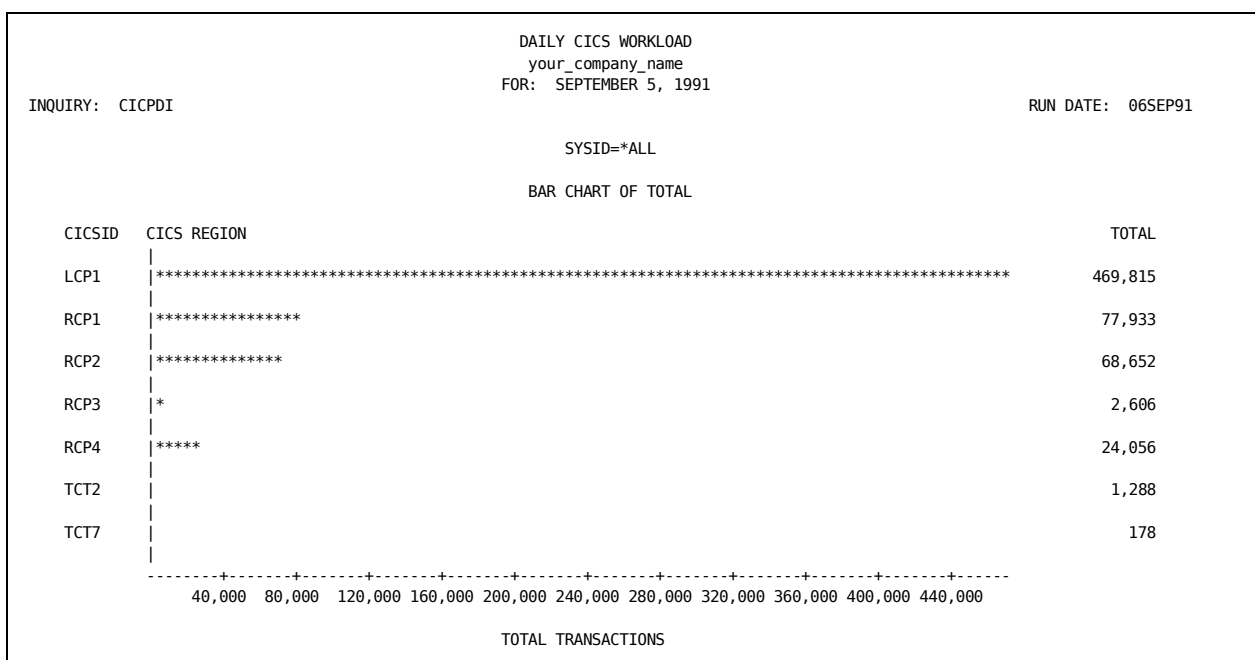
If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.



3.1.2.19 CICPD1: Daily CICS Workload Reports

The Daily CICS Workload Reports monitor CICS transaction volume on a daily basis and identify peak hour workload for each CICS system.

REPORT FORMAT:

This inquiry produces bar charts by system ID and CICS ID showing the number of transactions ended within each hour.

A sample output of this inquiry is shown in Figure 3-19.

INQUIRY ID:

CICPD1 (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan

DATA ELEMENTS USED:

CSYTRANS - Total Transactions Ended

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

USER EXITS:

None

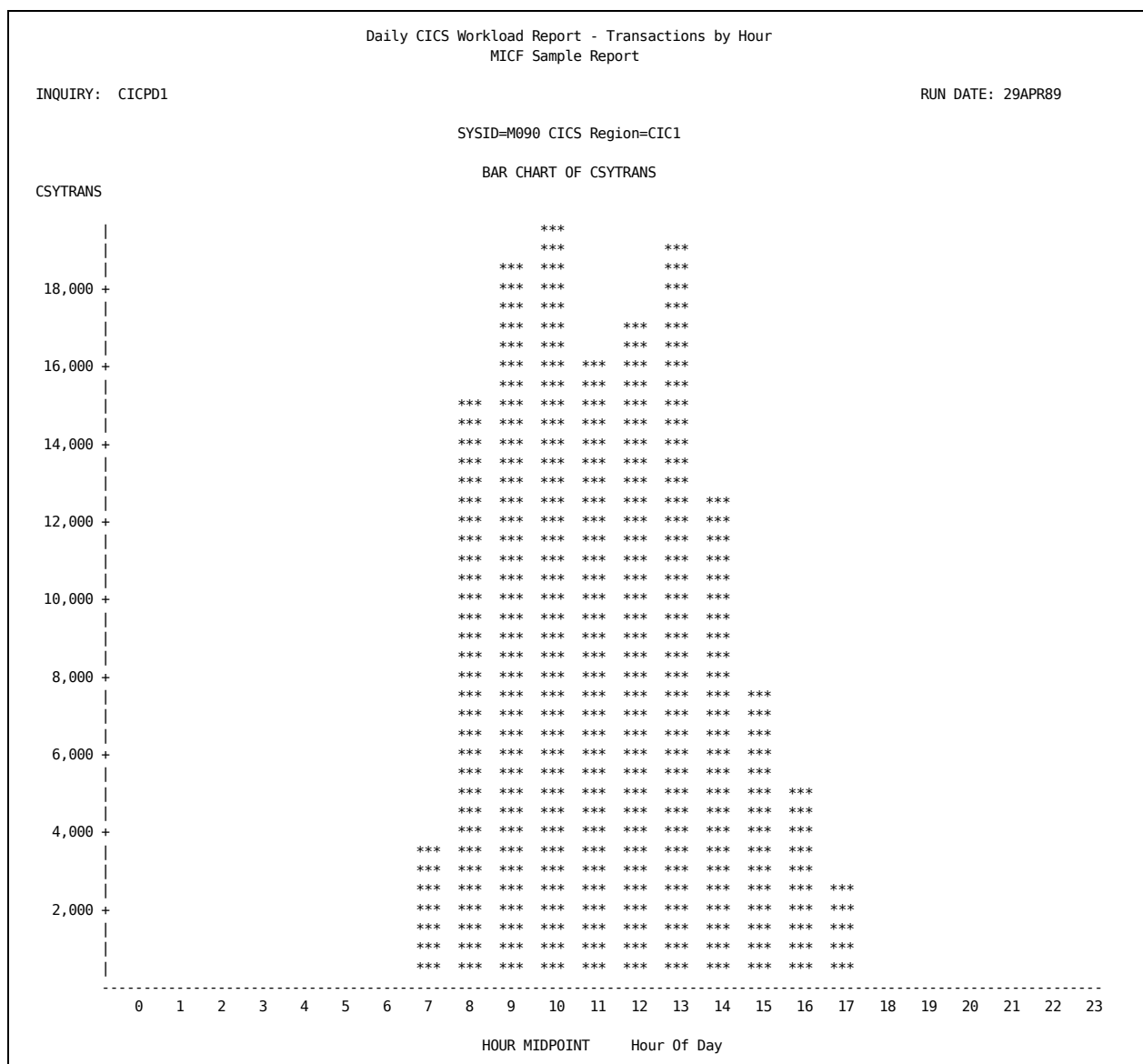


Figure 3-19. Daily CICS Workload Report Sample Output

3.1.2.20 CICPMH: Monthly CICS Workload Ranking

The Monthly CICS Workload Ranking graph shows the top 10 CICS application in transaction volume. By default, the applications are obtained from the CA MICS data element CICAPU in the CICCAU (CICS Application Activity) file. However, you can override the input file to use the CSU (CICS User Activity) file, and substitute CICAPU with any one of the CICTx data elements, which may identify an operator, an application, or a project.

REPORT FORMAT:

This inquiry produces a horizontal bar chart showing the 10 applications with the highest transaction volume. A sample output of this inquiry is shown in Figure 3-20.

INQUIRY ID:

CICPMH (printer graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUSU (CICS User Activity) file will be used for reporting. The variable CICT2 will be used in place of the variable CICAPU.

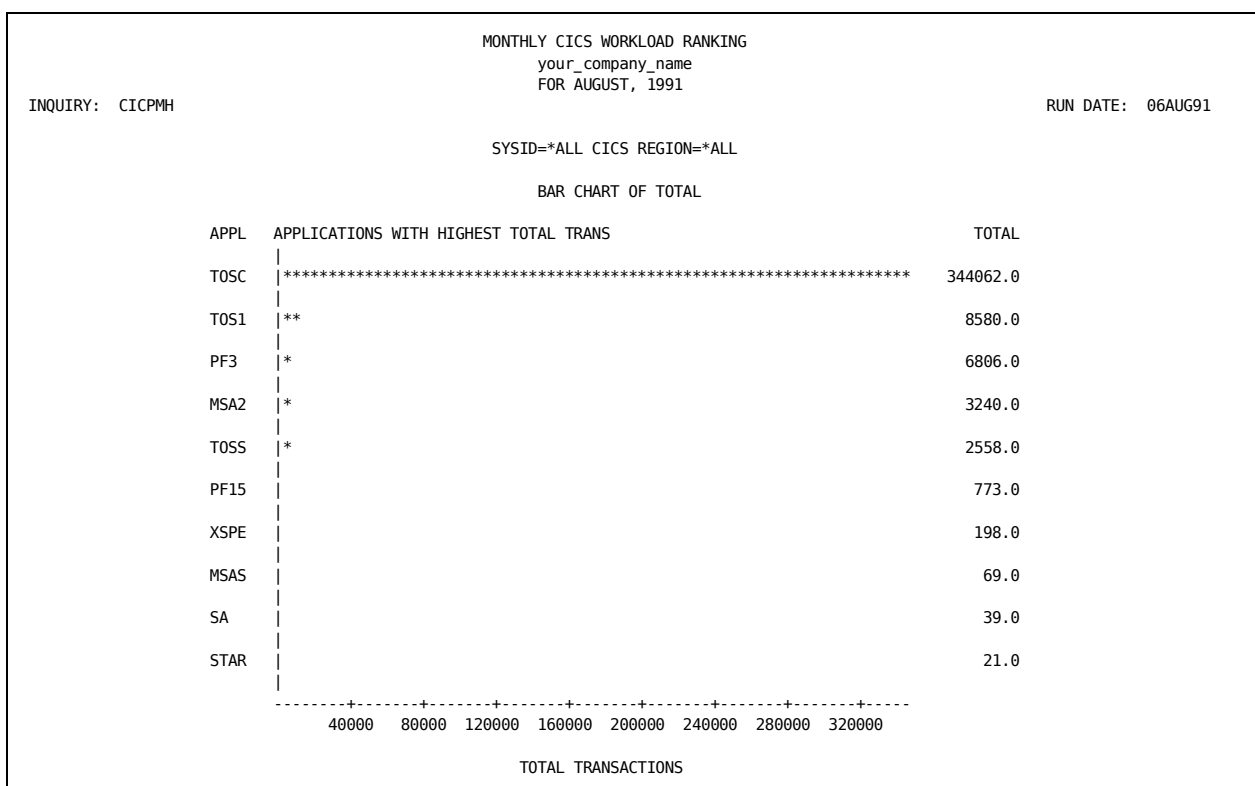
One graph produced for each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICAPU	- CICS Application Unit ID
CICSID	- CICS System Identifier
CAUTRANS	- Transactions Processed
CAUSTRN	- Short Transactions Processed
CAUMTRN	- Medium Transactions Processed
CAULTRN	- Long Transactions Processed
CAUCTRN	- Conv. Transactions Processed
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

None



3.1.2.21 CICPMI: Monthly CICS Workload

The Monthly CICS Workload inquiry produces multiple charts which show CICS workload by system (SYSID) and regions (CICSID). The charts show the workload distribution among systems.

REPORT FORMAT:

This inquiry produce horizontal bar charts which shows the total transaction volume per system.

A sample output of this inquiry is shown in Figure 3-21.

INQUIRY ID:

CICPMI (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the MONTHS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
CSYCTRN - Conv. Transactions Processed
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

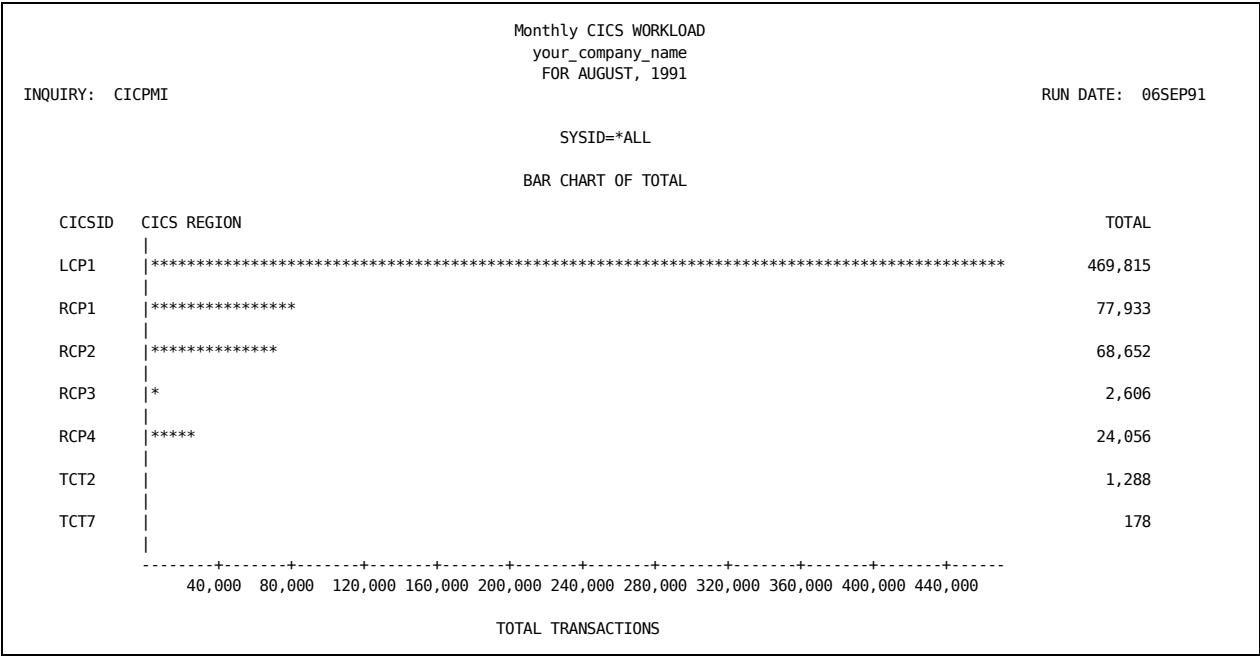


Figure 3-21. Monthly CICS Workload Sample Output

3.1.2.22 CICPM1: Monthly CICS Workload Reports

The Monthly CICS Workload Reports monitor CICS transaction volume trend by ZONE in a six-month period.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the number of transactions ended per ZONE in a six-month period. Any month for which data is unavailable is omitted from the charts.

A sample output from this inquiry is shown in Figure 3-22. (Due to space constraints in the guide, the sample report shows a five month period.)

INQUIRY ID:

CICPM1 (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01-CICCSY06 in the MONTHS timespan

DATA ELEMENTS USED:

CSYTRANS - Total Transactions Ended

CALCULATIONS:

None

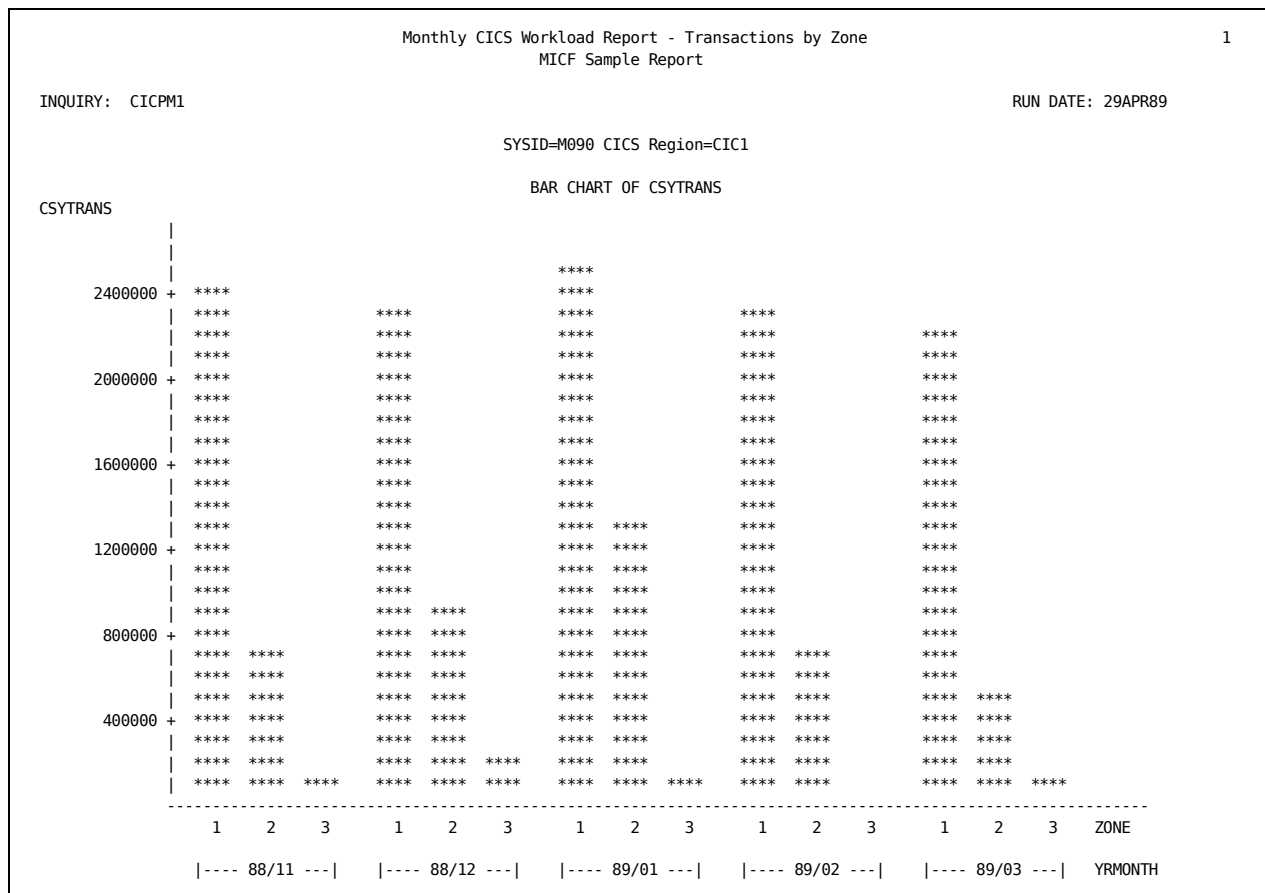


Figure 3-22. Monthly CICS Workload Report Sample Output

3.1.2.23 CICPWI: Weekly CICS Workload

The Weekly CICS Workload inquiry produces multiple charts which show CICS workload by system (SYSID) and regions (CICSID). The charts show the workload distribution among systems.

REPORT FORMAT:

This inquiry produce horizontal bar charts which shows the total transaction volume per system.

A sample output of this inquiry is shown in Figure 3-23.

INQUIRY ID:

CICPWI (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01 in the WEEKS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYSTRN - Short Transactions Processed
CSYMTRN - Medium Transactions Processed
CSYLTRN - Long Transactions Processed
CSYCTRN - Conv. Transactions Processed
ENDTS - End Time Stamp
WEEK - Week of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

- SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- DATE - Optionally, enter start of the report period in ddmmyy format.

For this inquiry, the default is the previous week.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- RESP - Optionally, select the type of responses to be ranked. The response types are AVG, SHORT, MED, LONG, or CONV. The default value is AVG which produces ranking by average response seconds.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

- %GCIC - Select CICSIDs for report.
- %GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

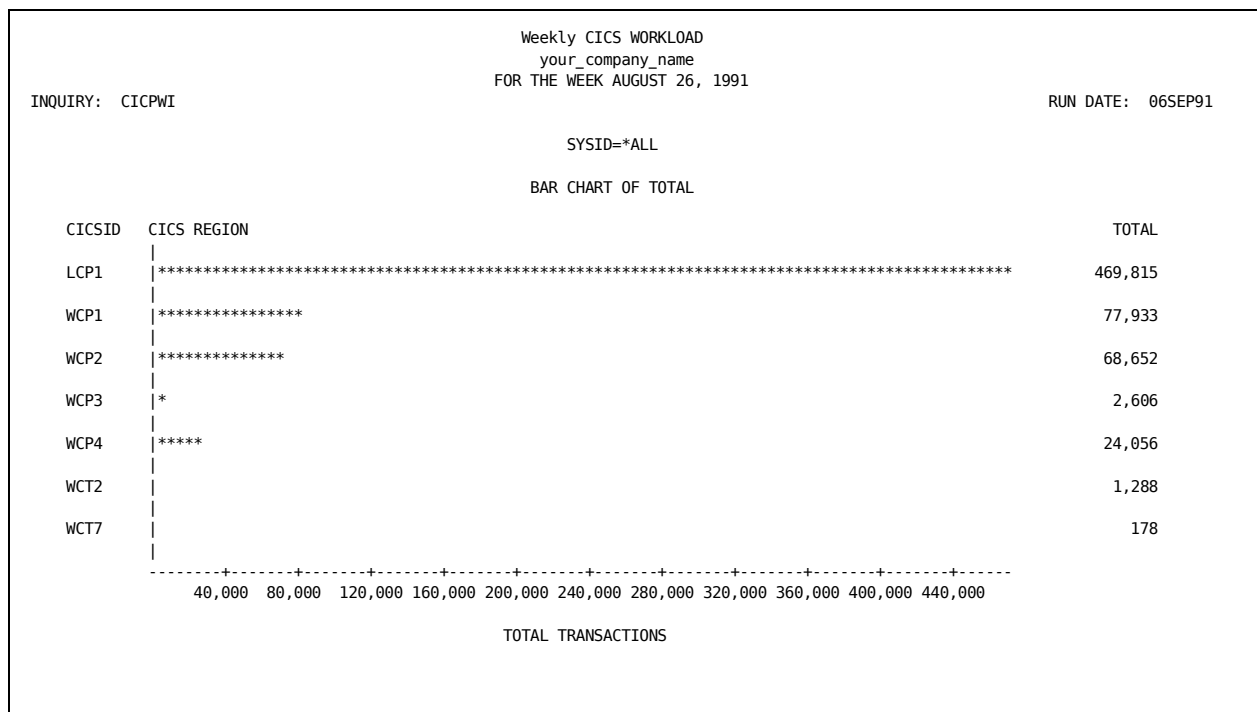


Figure 3-23. Weekly CICS Workload Sample Output

3.1.2.24 CICPW1: Weekly CICS Workload Reports

The Weekly CICS Workload Reports monitor CICS transaction volume trend in a five-day period and identify peak hour workload between 8 a.m. and 5 p.m. for each CICS system.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the number of transactions ended within each hour from 8 a.m. to 5 p.m. in a five-day period. The charts are based on data summarized at the DATE and HOUR levels. Data from Saturday and Sunday is excluded.

A sample output from this inquiry is shown in Figure 3-24.

INQUIRY ID:

CICPW1 (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01-CICCSY07 in the DAYS timespan

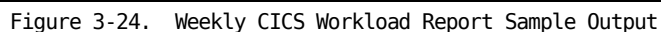
Data is selected for HOURS 8-16, excluding DAYNAME='SAT' and DAYNAME='SUN'.

DATA ELEMENTS USED:

CSYTRANS - Total Transactions Ended

CALCULATIONS:

None



3.1.2.25 CICLDH: Daily CICS Workload Ranking Report

The Daily CICS Workload Ranking inquiry produces tabular reports which identifies the top 10 CICS applications in transaction volume.

REPORT FORMAT:

This inquiry produces a tabular report which contains a summary section and a ranking section. The summary section shows the total workload and the peak hour workload. The ranking section identifies the 10 applications with the highest transaction volume and provides other pertinent information such as response time, CPU time, data base calls, character traffic, and storage usage.

A sample output of this inquiry is shown in Figure 3-25. Each field in this report is described below:

SYSTEM SUMMARY:

- # REGIONS: Total number of SYSID/CICSID combinations found in input.

- TRANS: The total number of transactions processed, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.

- %SHORT: The percentage of short transactions processed.

- %MED: The percentage of medium transactions processed.

- %LONG: The percentage of long transactions processed.

- AVG RESP: The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANTYPE=X as assigned by the CICRLRT exit).

- TOTAL CPU: The total TCB CPU time consumed by all transactions, excluding transaction type X.

- PEAK HOUR: The hour during which the peak average response time occurred.

- PEAK TRAN: The total number of transactions processed in the peak hour.

For this inquiry, transaction type X is excluded.

PEAK CPU: The total TCB CPU time consumed during peak hour.

PEAK RESP: The peak average hourly response time provided by CICS.

RANKING SECTION:

RANK: The ranking, number 1 to 10, in transaction volume.

APPL: Application identification.

TRANS: The total number of transactions processed for the application.

AVG RESP: The average response time for all transactions processed for this application.

TOT CPU: The total TCB CPU time consumed by this application.

FC CALLS: The total number of file control calls issued by this application.

DBMS CALLS: The total number of data base calls (DB2 and IMS) issued by this application.

TOT CHARS: The total number of input and output characters generated by this application.

MAX STORAGE: The maximum amount (in 1024 bytes) of terminal and user storage used by this application.

AVERAGE CPU: The average TCB CPU per transaction consumed by this application.

AVERAGE FC CALLS: The average number of file control calls per transaction issued by this application.

AVERAGE DBMS CALLS: The average number of data base (DB2 and IMS) calls per transaction issued by this application.

AVERAGE CHAR: The average number of input and out characters per transaction generated by this application.

AVERAGE The average amount of dynamic storage (terminal

STORAGE: and user) per transaction consumed by this application.

% ABND: The percentage of transactions that abended for this application.

INQUIRY ID:

CICLDH (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICALPU.

One graph produced for each date and each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CAUABEND - Abnormal Termination Occurrences
CAUCPUTM - Task CPU Time
CAUCTRN - Conv. Transactions Processed
CAUDLICC - Number of CICS DL/I Calls
CAUETRN - Excluded Transactions
CAUFCOPS - Number of CICS File Control Calls
CAUINCH - Input Message Character Traffic
CAULTRN - Long Transactions Processed
CAUMEMRC - Memory Usage Running Count
CAUMXMEM - Maximum Transaction Storage Used
CAUOUTCH - Output Message Character Traffic
CAUSQLTO - Total number of SQL Calls
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
CICALPU - CICS Application Unit ID
CICSID - CICS System Identifier
CAUMTRN - Medium Transactions Processed
CAUSTRN - Short Transactions Processed
DAY - Day of Month
MONTH - Month of Year
SYSID - System Identifier
YEAR - Year of Century
ZONE - Time Zone

CALCULATIONS:

Average character = $\frac{\text{SUM(CAUINCH,CAUOUTCH)}}{\text{SUM(CAUTRANS,CAUETR N)}}$
Average file control calls = $\frac{\text{CAUFCOPS}}{\text{SUM(CAUTRANS,CAUETR N)}}$
Average response time = $\frac{\text{CAUTRSTM}}{\text{CAUTRANS}}$
Average storage = $\frac{\text{CAUMEMRC}}{\text{SUM(CAUTRANS,CAUETR N)}}$
Average CPU time = $\frac{\text{CAUCPUTM}}{\text{CAUTRANS}}$
Average DBMS calls = $\frac{\text{SUM(CAUDLIC C,CAUSQLT O,CICPCT01)}}{\text{SUM(CAUTRANS,CAUETR N)}}$
Percent abend = $\frac{\text{CAUABEND}}{\text{SUM(CAUTRANS,CAUETR N)}}$
Percent conv. = $\frac{\text{CAUCTRN}}{\text{SUM(CAUTRANS,CAUETR N)}}$
Percent long = $\frac{\text{CAULTRN}}{\text{SUM(CAUTRANS,CAUETR N)}}$
Percent medium = $\frac{\text{CAUMTRN}}{\text{SUM(CAUTRANS,CAUETR N)}}$
Percent short = $\frac{\text{CAUSTRN}}{\text{SUM(CAUTRANS,CAUETR N)}}$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

TRAN - Optionally, select the type of transaction to be ranked. The response types are TOTAL,

SHORT, MED, LONG, or CONV. The default is TOTAL.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICACTION data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

DAILY CICS WORKLOAD RANKING REPORT													
your_company_name													
FOR: SEPTEMBER 5, 1991													
INQUIRY: CICLDH										RUN DATE: 06SEP91			
SYSTEM SUMMARY: 7 REGIONS													
# TRANSACTIONS	% SHORT	% MED	% LONG	% CONV	AVG RESP	TRANS/ SEC	----- HOUR	PEAK ACTIVITY -----					
								TRANS	TRAN/SEC	RESP			
644,528	56.3	0.0	36.0	7.7	0.32	1.6	9	75,037	3.0	0.15			
** TOP 10 APPLICATIONS IN TOTAL TRANSACTION VOLUME													
RANK	APPL	TOTAL TRANS	AVG RESP	TOT CPU	FILE CNTL CALLS	DBMS CALLS	TOT CHARS (000)	MAX STORAGE	----- CPU	FC CL	DBMS CL	CHAR STORAGE	% ABND
1	TOSC	344,062	0.13	0:11	42	0	416132	8K	0.00	0	0	1209	8K 0
2	CRSR	29,214	0.02	0:01	0	0	0	11K	0.00	0	0	0	11K 0
3	CSNE	26,610	0.02	0:01	0	0	45	33K	0.00	0	0	2	30K 0
4	PUBC	24,221	0.15	0:04	48134	0	26592	23K	0.01	2	0	1098	28K 0
5	ISHD	23,006	0.62	0:12	29458	0	29997	184K	0.03	1	0	1304	279K 0
6	CSGM	17,561	0.03	0:01	84	0	16785	8K	0.00	0	0	956	9K 0
7	CATD	17,344	0.09	0:02	0	0	0	23K	0.01	0	0	0	300K 0
8	DATA	16,126	0.15	0:01	6900	0	10762	1448K	0.00	0	0	667	20K 0
9	PUSC	15,199	0.04	0:02	14299	0	15168	92K	0.01	1	0	998	36K 0
10	PIBC	10,675	0.13	0:02	17170	0	11801	13K	0.01	2	0	1105	25K 0

Figure 3-25. Daily CICS Workload Ranking Report Sample Output

3.1.2.26 CICLDL: Daily CICS Workload Summary Report

The Daily CICS Workload Summary Report shows total activity performed by each CICS region on a given day. The report also displays peak hour activity and identify the top application in transaction volume for each region.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for all CICS regions the enterprise (summed across SYSIDs and CICSs), and a summary line by CICSIDs (summed across SYSIDs).

A sample output of this inquiry is shown in Figure 3-26. Each field in the report is described below:

SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
%SHORT:	The percentage of short transactions processed.
%MED:	The percentage of medium transactions processed.
%LONG:	The percentage of long transactions processed.
% CONV:	Percentage of conversational transactions processed.
AVG RESP:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.
CPU:	The total TCB CPU time consumed.
PEAK HOUR:	The hour during which CICS processed the highest number of transactions.
PEAK TRAN:	The total number of transactions processed in the peak hour.
PEAK RESP:	The average response time during peak hour.
TOP APPL:	The application with the highest workload volume for the day.

TOP TRANS: The total number of transactions (in thousands) processed for the top application.

TOP CPU: The total TCB CPU consumed by the top application in workload volume.

TOP RESP: The average response time received by the top application in workload volume.

INQUIRY ID:

CICLDL (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.
CICCSY01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICALPU.

DATA ELEMENTS USED:

CAUCPUTM - Task CPU Time
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
CICALPU - CICS Application Unit ID
CICCSID - CICS System Identifier
CSYCTRN - Conversational Transactions Processed
CSYLRSTM - Conversational Response Time Total
CSYLRSTM - Long Response Time Total
CSYLTRN - Long Transactions Processed
CSYMRSTM - Medium Response Time Total
CSYMTRN - Medium Transactions Processed
CSYSRSTM - Short Response Time Total
CSYSTRN - Short Transactions Processed
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYUPTM - CICS Availability Time
DAY - Day of Month
HOUR - Hour of Day
MONTH - Month of Year
SYSID - System Identifier
YEAR - Year of Century
ZONE - Time Zone

CALCULATIONS:

Average response time = CSYTRSTM/CSYTRANS
Average short response time = CSYSRSTM/CSYSTRN
Average medium response time = CSYMRSTM/CSYMTRN
Average long response time = CSYLRSTM/CSYLTRN
Average conv response time = CSYCRSTM/CSYCTRN

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICAx data elements, where x is 1 to 9. This parameter must be specified in conjunction with

FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

DAILY CICS WORKLOAD SUMMARY REPORT																
your_company_name																
FOR: SEPTEMBER 5, 1991																
INQUIRY: CICLDL																
RUN DATE: 06SEP91																
SYSID	CICSID	TRANS	% SHORT	% MED	% LONG	% CONV	AVG RESP	TRANS/ SEC	CPU HOUR	----- PEAK ACTIVITY -----	-----	-----	TOP APPLICATION	-----	-----	-----
										TRANS	TRAN/SEC	RESP	APPL	TRANS	RESP	
*ALL	*ALL	644,528	56	0	36	8	0.32	1.6	1:26 9	75037	3.0	0.15	TOSC	344062	0.13	
PRD2	*ALL	644,528	56	0	36	8	0.32	1.6	1:26 9	75037	3.0	0.15	TOSC	344062	0.13	
	LCP1	469,815	77	0	14	9	0.12	6.7	0:36 8	56911	15.8	0.10	TOSC	344062	0.13	
	WCP1	77,933	1	0	91	8	0.13	1.4	0:14 10	10161	2.8	0.14	PUBC	12085	0.16	
	WCP2	68,652	0	0	99	1	0.34	1.3	0:17 10	9283	2.6	0.14	PUBC	12085	0.15	
	WCP3	2,606	29	0	51	19	0.26	0.0	0:01 15	334	0.1	0.16	DATA	768	0.20	
	WCP4	24,056	0	0	96	4	0.61	0.4	0:17 11	3032	0.8	0.70	ISHD	23006	0.62	
	WCT2	1,288	0	0	60	40	0.27	0.0	0:01 13	234	0.1	0.09	CCMF	178	0.01	
	WCT7	178	0	0	4	96	563.10	0.0	0:00 6	19	0.0	0.65	CCMF	168	0.01	
TOTAL # REGIONS: 7																

Figure 3-26. Daily CICS Workload Summary Report Sample Output

3.1.2.27 CICLDM: Daily CICS Workload Detail Report

The Daily CICS Workload Detail Report displays CICS activity at the application level and provides workload characteristics and application profile information. It also provides a summary section which shows CICS activity by hour and the top application in transaction volume for each hour.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary section and a detail section for each combination of SYSID and CICSID. The summary section shows CICS activity for each hour. The detail section shows the transaction processed by each application in each CICS region on each system, and provides profile information such as average CPU, average file control calls, average DBMS calls, and average storage used.

A sample output of this inquiry is shown in Figure 3-27. Each field in the report is described below:

SYSTEM SUMMARY:

- # REGIONS: Total number of SYSID/CICSID combinations found in input.
- HOURL: Hour of the day.
- TRANS: The total number of transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
- %SHORT: The percentage of short transactions processed.
- %MED: The percentage of medium transactions processed.
- %LONG: The percentage of long transactions processed.
- % CONV: Percentage of conversational transactions processed.
- AVG RESP: The average response time for all transactions, excluding those assigned as TRANSTYPE X in your CICRLRT exit in prefix.MICS.PARMS.

CPU: The total TCB CPU time consumed.

AVG
FC CALLS: The average number of file control calls issued per transaction.

AVG
CHAR: The average number of input and output characters generated per transaction.

TOP APPL: The application for which CICS processed the highest number of transaction in an hour.

TOP TRANS: The total number transactions processed in an hour for the top application.

TOP RESP: The average response time received by the top application for the hour.

TOP CPU: The average TCB CPU time consumed by the top application for the hour.

TOP AVG
FC CALLS: The average number of file control calls issued per transaction by the top application.

TOP AVG
CHAR: The average number of input and output characters generated per transaction by the top application.

TOP AVG
MSG: The average number of input and output messages generated per transaction by the top application.

DETAIL SECTION:

SYSID: System identification. This field is set to ALL for data that is summarized for all systems.

CICSID: CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.

APPL: Application identification.

TRANS: The total number of transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).

AVG RESP: The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).

TOTAL CPU: The total TCB CPU time consumed by all transactions, excluding transaction type X.

FC CALLS: The total number of file control calls issued by this application.

DBMS CALLS: The total number of data base calls (DB2 and IMS) issued by this application.

TOT CHARS: The total number of input and output characters generated by this application.

MAX STORAGE: The maximum amount (in 1024 bytes) of terminal and user storage used by this application.

AVERAGE CPU: The average TCB CPU per transaction consumed by this application.

AVERAGE FC CALLS: The average number of file control calls per transaction issued by this application.

AVERAGE DBMS CALLS: The average number of data base (DB2 and IMS) calls per transaction issued by this application.

AVERAGE CHAR: The average number of input and out characters per transaction generated by this application.

AVERAGE STORAGE: The average amount of dynamic storage (terminal and user) per transaction consumed by this application.

% ABND: The percentage of transactions that abended for this application.

INQUIRY ID:

CICLDM (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICALPU.

DATA ELEMENTS USED:

- CAUABEND - Abnormal Termination Occurrences
- CAUCPUTM - Task CPU Time
- CAUCTRN - Conv. Transactions Processed
- CAUDLICC - Number of CICS DL/I Calls
- CAUETRN - Excluded Transactions
- CAUFCOPS - Number of CICS File Control Calls
- CAUINCH - Input Message Character Traffic
- CAULTRN - Long Transactions Processed
- CAUMEMRC - Memory Usage Running Count
- CAUMXMEM - Maximum Transaction Storage Used
- CAUOUTCH - Output Message Character Traffic
- CAUSQLTO - Total number of SQL Calls
- CAUTRANS - Transactions Processed
- CAUTRSTM - Transaction Response Time Total
- CICALPU - CICS Application Unit ID
- CICSID - CICS System Identifier
- CSUMTRN - Medium Transactions Processed
- CSUSTRN - Short Transactions Processed
- DAY - Day of Month
- HOURL - Hour of Day
- MONTH - Month of Year
- SYSID - System Identifier
- YEAR - Year of Century
- ZONE - Time Zone

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS
 Average character = $\text{SUM}(\text{CAUINCH}, \text{CAUOUTCH}) / \text{SUM}(\text{CAUTRANS}, \text{CAUETRN})$
 Average CPU time = CAUCPUTM/CAUTRANS
 Average DBMS calls = $\text{SUM}(\text{CAUDLICC}, \text{CAUSQLT0}, \text{CICPCT01}) / \text{SUM}(\text{CAUTRANS}, \text{CAUETRN})$
 Average file control calls = CAUFCOPS/
 $\text{SUM}(\text{CAUTRANS}, \text{CAUETRN})$
 Average storage = CAUMEMRC/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETRN})$
 Percent abend = CAUABEND/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETRN})$
 Percent conv. = CAUCTRN/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETRN})$
 Percent long = CAULTRN/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETRN})$
 Percent medium = CAUMTRN/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETRN})$
 Percent short = CAUSTRN/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETRN})$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

DAILY CICS WORKLOAD SUMMARY REPORT													
your_company_name													
FOR: SEPTEMBER 5, 1991													
INQUIRY: CICLDM													
RUN DATE: 06SEP91													
-----SYSID=PRD2 CICSID=LCP1-----													
APPL	TOTAL TRANS	AVG RESP	TOT CPU	FILE CNTL CALLS	DBMS CALLS	TOT CHARS (000)	MAX STORAGE	CPU	FC	CL	DBMS CL	CHAR STORAGE	% ABND
AAON	1	34.10	0:00	1	0	0	157K	0.08	1	0	0	157K	0
ACFM	163	0.02	0:00	0	0	162	72K	0.01	0	0	994	70K	0
CATD	15,759	0.07	0:02	0	0	0	297K	0.01	0	0	0	301K	0
CCMF	233	0.00	0:00	0	0	0	4K	0.00	0	0	0	4K	0
CEDA	21	0.20	0:00	82	0	13	282K	0.01	4	0	639	205K	0
CEMT	10	0.07	0:00	0	0	3	91K	0.01	0	0	347	90K	0
CRSQ	3	0.27	0:00	0	0	0	4K	0.00	0	0	0	4K	0
CRSR	29,214	0.02	0:01	0	0	0	11K	0.00	0	0	0	11K	0
CRTE	691	0.02	0:00	0	0	258	11K	0.00	0	0	373	11K	0
CSAC	350	0.00	0:00	0	0	33	11K	0.00	0	0	94	10K	0
CSFU	1	0.83	0:00	0	0	0	35K	0.01	0	0	0	35K	0
CSGM	16,516	0.03	0:01	84	0	14931	10K	0.00	0	0	904	9K	0
CSIR	61	47.96	0:00	0	0	0	5K	0.00	0	0	6	5K	0
CSMI	11	0.01	0:00	0	0	4	10K	0.00	0	0	396	10K	0
CSNE	23,797	0.00	0:01	0	0	0	27K	0.00	0	0	0	27K	0
CSSF	38	0.01	0:00	0	0	10	19K	0.00	0	0	268	19K	0
CSSN	6	0.03	0:00	0	0	4	19K	0.00	0	0	597	19K	0
DATA	14,588	0.14	0:01	21	0	9544	197K	0.00	0	0	654	16K	0
DISP	1	0.03	0:00	0	0	0	13K	0.02	0	0	314	13K	1
D2DA	15	1.42	0:00	0	0	21	16K	0.00	0	0	1425	39K	0
EL0N	10	0.24	0:00	0	0	8	198K	0.01	0	0	784	126K	0
EL00	40	0.06	0:00	0	0	35	8K	0.00	0	0	875	7K	0
EL10	9	0.04	0:00	0	0	6	8K	0.00	0	0	631	6K	0
EL20	41	0.15	0:00	0	0	54	8K	0.00	0	0	1322	7K	0
EL22	50	0.14	0:00	0	0	57	8K	0.00	0	0	1150	8K	0
EL29	21	0.11	0:00	0	0	21	8K	0.00	0	0	1011	7K	0
EL30	14	0.07	0:00	0	0	7	8K	0.00	0	0	491	6K	0
EZ	324	0.46	0:00	0	0	519	197K	0.00	0	0	1602	14K	0
IE0L	1,346	0.25	0:00	0	0	1147	13K	0.00	0	0	852	13K	0
LLTP	1	0.01	0:00	0	0	0	12K	0.00	0	0	136	12K	1
LLXW	4	0.04	0:00	0	0	1	11K	0.01	0	0	273	12K	1
LPKD	3	0.00	0:00	0	0	1	10K	0.00	0	0	376	8K	0
LPKE	12	0.00	0:00	0	0	18	8K	0.00	0	0	1460	8K	0
MSAS	69	0.17	0:00	0	0	69	8K	0.00	0	0	1007	11K	0
MSA2	3,240	0.13	0:00	57	0	5251	8K	0.00	0	0	1621	8K	0
SA	39	0.48	0:00	21	0	36	198K	0.00	1	0	927	90K	0
STAR	21	0.00	0:00	0	0	29	8K	0.00	0	0	1399	9K	0
TOSC	344,062	0.13	0:11	42	0	416132	8K	0.00	0	0	1209	8K	0
TOSS	2,558	0.40	0:00	0	0	3937	198K	0.01	0	0	1539	132K	0
TOS1	8,580	0.38	0:01	0	0	13190	197K	0.01	0	0	1537	135K	0
TOTAL:	469,835	0.12	0:18	308	0	465952	297K	0.00	0	0	992	22K	0

Figure 3-27. Daily CICS Workload Detail Report Sample Output

3.1.2.28 CICLMH: Monthly CICS Workload Ranking Report

The Monthly CICS Workload Ranking inquiry produces tabular reports which identifies the top 10 CICS applications in transaction volume.

REPORT FORMAT:

This inquiry produces a tabular report which contains a summary section and a ranking section. The summary section shows the total workload and the peak hour workload. The ranking section identifies the 10 applications with the highest transaction volume and provides other pertinent information such as response time, CPU time, data base calls, character traffic, and storage usage.

A sample output of this inquiry is shown in Figure 3-28. Each field in this report is described below:

SYSTEM SUMMARY:

- # REGIONS: Total number of SYSID/CICSID combinations found in input.
- TRANS: The total number of transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
- %SHORT: The percentage of short transactions processed.
- %MED: The percentage of medium transactions processed.
- %LONG: The percentage of long transactions processed.
- AVG RESP: The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
- TOTAL CPU: The total TCB CPU time consumed by all transactions, excluding transaction type X.
- PEAK HOUR: The hour during which CICS processed the highest number of transactions.
- PEAK TRAN: The total number of transactions processed during peak hour, excluding transaction type X.
- PEAK CPU: The total TCB CPU time consumed during peak hour.

PEAK RESP: The peak average hourly response time provided by CICS.

RANKING SECTION:

RANK: The ranking, number 1 to 10, in transaction volume.

APPL: Application identification.

TRANS: The total number of transactions processed for the application.

AVG RESP: The average response time for all transactions processed for this application.

TOT CPU: The total TCB CPU time consumed by this application.

FC CALLS: The total number of file control calls issued by this application.

DBMS CALLS: The total number of data base calls (DB2 and IMS) issued by this application.

TOT CHARS: The total number of input and output characters generated by this application.

MAX STORAGE: The maximum amount (in 1024 bytes) of terminal and user storage used by this application.

AVERAGE CPU: The average TCB CPU per transaction consumed by this application.

AVERAGE FC CALLS: The average number of file control calls per transaction issued by this application.

AVERAGE DBMS CALLS: The average number of data base (DB2 and IMS) calls per transaction issued by this application.

AVERAGE CHAR: The average number of input and out characters per transaction generated by this application.

AVERAGE STORAGE: The average amount of dynamic storage (terminal and user) per transaction consumed by this application.

% ABND: The percentage of transactions that abended for

this application.

INQUIRY ID:

CICLMH (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAPU.

One graph produced for each date and each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICAPU - CICS Application Unit ID
CICSID - CICS System Identifier
CAUABEND - Abnormal Termination Occurrences
CAUCTRN - Conv. Transactions Processed
CAUETRN - Excluded Transactions
CAUDLICC - Number of CICS DL/I Calls
CAUFCOPS - Number of CICS File Control Calls
CAUINCH - Input Message Character Traffic
CAULTRN - Long Transactions Processed
CAUMEMRC - Memory Usage Running Count
CSUMTRN - Medium Transactions Processed
CAUMXMEM - Maximum Transaction Storage Used
CAUOUTCH - Output Message Character Traffic
CAUCPUTM - Task CPU Time
CAUSQLTO - Total number of SQL Calls
CSUSTRN - Short Transactions Processed
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS
Average character = SUM(CAUINCH, CAUOUTCH)/
SUM(CAUTRANS, CAUETRN)

EXECUTION-TIME PARAMETERS:

TRAN - Optionally, select the type of transaction to be ranked. The response types are TOTAL, SHORT, MED, LONG, or CONV. The default is TOTAL.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICA_x data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

MONTHLY CICS WORKLOAD RANKING REPORT														
your_company_name														
FOR AUGUST, 1991														
INQUIRY: CICLMH								RUN DATE: 06SEP91						
SYSTEM SUMMARY: 7 REGIONS														
# TRANSACTIONS	% SHORT	% MED	% LONG	% CONV	AVG RESP	TRANS/ SEC	----- HOUR	PEAK ACTIVITY -----						
								TRANS	TRAN/SEC	RESP				
644,528	56.3	0.0	36.0	7.7	0.32	1.6	9	75,037	3.0	0.15				
** TOP 10 APPLICATIONS IN TOTAL TRANSACTION VOLUME														
RANK	APPL	TOTAL TRANS	AVG RESP	TOT CPU	FILE CNTL CALLS	DBMS CALLS	TOT CHARS (000)	MAX STORAGE	----- CPU	FC CL	AVERAGES -----	CHAR STORAGE	% ABND	
1	TOSC	344,062	0.13	0:11	42	0	416132	8K	0.00	0	0	1209	8K	0
2	CRSR	29,214	0.02	0:01	0	0	0	11K	0.00	0	0	0	11K	0
3	CSNE	26,610	0.02	0:01	0	0	45	33K	0.00	0	0	2	30K	0
4	PUBC	24,221	0.15	0:04	48134	0	26592	23K	0.01	2	0	1098	28K	0
5	ISHD	23,006	0.62	0:12	29458	0	29997	184K	0.03	1	0	1304	279K	0
6	CSGM	17,561	0.03	0:01	84	0	16785	8K	0.00	0	0	956	9K	0
7	CATD	17,344	0.09	0:02	0	0	0	23K	0.01	0	0	0	300K	0
8	DATA	16,126	0.15	0:01	6900	0	10762	1448K	0.00	0	0	667	20K	0
9	PUSC	15,199	0.04	0:02	14299	0	15168	92K	0.01	1	0	998	36K	0
10	PIBC	10,675	0.13	0:02	17170	0	11801	13K	0.01	2	0	1105	25K	0

Figure 3-28. Monthly CICS Workload Ranking Report Sample Output

3.1.2.29 CICLML: Monthly CICS Workload Summary Report

The Monthly CICS Workload Summary Report shows total activity performed by each CICS region in a given month. The report also displays peak hour activity and identify the top application in transaction volume for each region.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for all CICS regions the enterprise (summed across SYSIDs and CICSs), and a summary line by CICSIDs (summed across SYSIDs).

A sample output of this inquiry is shown in Figure 3-29. Each field in the report is described below:

SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
TRANS:	The total number of transactions (in thousands) processed.
%SHORT:	The percentage of short transactions processed.
%MED:	The percentage of medium transactions processed.
%LONG:	The percentage of long transactions processed.
% CONV:	Percentage of conversational transactions processed.
AVG RESP:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.
CPU:	The total TCB CPU time consumed.
PEAK HOUR:	The hour during which CICS processed the highest number of transactions.
PEAK TRAN:	The total number of transactions processed in the peak hour.

PEAK RESP: The peak average hourly response time provided by CICS.

TOP APPL: The application with the highest workload volume for the month.

TOP TRANS: The total number of transactions (in thousands) processed for the top application.

TOP CPU: The total TCB CPU consumed by the top application in workload volume.

TOP RESP: The average response time received by the top application in workload volume.

INQUIRY ID:

CICLML (tabular report)

DATA SOURCE (file/time-span):

CICCSY01 in the MONTHS time-span.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYSTRN - Short Transactions Processed
CSYSRSTM - Short Response Time Total
CSYMTRN - Medium Transactions Processed
CSYMRSTM - Medium Response Time Total
CSYLTRN - Long Transactions Processed
CSYLRSTM - Long Response Time Total
CSYCTRN - Conversational Transactions Processed
CSYLRSTM - Conversational Response Time Total
CSYUPTM - CICS Availability Time
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CSYTRSTM/CSYTRANS
Average short response time = CSYSRSTM/CSYSTRN
Average medium response time = CSYMRSTM/CSYMTRN
Average long response time = CSYLRSTM/CSYLTRN
Average conv response time = CSYCRSTM/CSYCTRN

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICA_x data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the

following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

MONTHLY CICS WORKLOAD SUMMARY REPORT																
your_company_name																
FOR AUGUST, 1991																
INQUIRY: CICLML										RUN DATE: 06SEP91						
SYSID	CICSID	TRANS	% SHORT	% MED	% LONG	% CONV	AVG RESP	TRANS/ SEC	CPU HOUR	----- PEAK ACTIVITY -----	----- TOP APPLICATION -----	TRANS	RESP			
										TRANS	TRAN/SEC	RESP	APPL			
*ALL	*ALL	644,528	56	0	36	8	0.32	1.6	1:26 9	75037	3.0	0.15	TOSC	344062	0.13	
PRD2	*ALL	644,528	56	0	36	8	0.32	1.6	1:26 9	75037	3.0	0.15	TOSC	344062	0.13	
	LCP1	469,815	77	0	14	9	0.12	6.7	0:36 8	56911	15.8	0.10	TOSC	344062	0.13	
	RCP1	77,933	1	0	91	8	0.13	1.4	0:14 10	10161	2.8	0.14	PUBC	12085	0.16	
	RCP2	68,652	0	0	99	1	0.34	1.3	0:17 10	9283	2.6	0.14	PUBC	12085	0.15	
	RCP3	2,606	29	0	51	19	0.26	0.0	0:01 15	334	0.1	0.16	DATA	768	0.20	
	RCP4	24,056	0	0	96	4	0.61	0.4	0:17 11	3032	0.8	0.70	ISHD	23006	0.62	
	TCT2	1,288	0	0	60	40	0.27	0.0	0:01 13	234	0.1	0.09	CCMF	178	0.01	
	TCT7	178	0	0	4	96	563.10	0.0	0:00 6	19	0.0	0.65	CCMF	168	0.01	
TOTAL # REGIONS: 7																

Figure 3-29. Monthly CICS Workload Summary Report Sample Output

3.1.2.29 CICLML: Monthly CICS Workload Summary Report

The Monthly CICS Workload Summary Report shows total activity performed by each CICS region in a given month. The report also displays peak hour activity and identify the top application in transaction volume for each region.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for all CICS regions the enterprise (summed across SYSIDs and CICSs), and a summary line by CICSIDs (summed across SYSIDs).

A sample output of this inquiry is shown in Figure 3-29. Each field in the report is described below:

SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
TRANS:	The total number of transactions (in thousands) processed.
%SHORT:	The percentage of short transactions processed.
%MED:	The percentage of medium transactions processed.
%LONG:	The percentage of long transactions processed.
% CONV:	Percentage of conversational transactions processed.
AVG RESP:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICLRT exit in prefix.MICS.PARMS.
CPU:	The total TCB CPU time consumed.
PEAK HOUR:	The hour during which CICS processed the highest number of transactions.
PEAK TRAN:	The total number of transactions processed in the peak hour.
PEAK RESP:	The peak average hourly response time provided by

CICS.

TOP APPL: The application with the highest workload volume for the month.

TOP TRANS: The total number of transactions (in thousands) processed for the top application.

TOP CPU: The total TCB CPU consumed by the top application in workload volume.

TOP RESP: The average response time received by the top application in workload volume.

INQUIRY ID:

CICLML (tabular report)

DATA SOURCE (file/time-span):

CICCSY01 in the MONTHS time-span.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYSTRN - Short Transactions Processed
CSYSRSTM - Short Response Time Total
CSYMTRN - Medium Transactions Processed
CSYMRSTM - Medium Response Time Total
CSYLTRN - Long Transactions Processed
CSYLRSTM - Long Response Time Total
CSYCTRN - Conversational Transactions Processed
CSYLRSTM - Conversational Response Time Total
CSYUPTM - CICS Availability Time
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CSYTRSTM/CSYTRANS
Average short response time = CSYSRSTM/CSYSTRN
Average medium response time = CSYMRSTM/CSYMTRN
Average long response time = CSYLRSTM/CSYLTRN
Average conv response time = CSYCRSTM/CSYCTRN

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICTACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input

file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

MONTHLY CICS WORKLOAD SUMMARY REPORT																
your_company_name																
FOR AUGUST, 1991																
INQUIRY: CICLML										RUN DATE: 06SEP91						
SYSID	CICSID	TRANS	% SHORT	% MED	% LONG	% CONV	AVG RESP	TRANS/ SEC	CPU HOUR	----- PEAK ACTIVITY -----	----- TOP APPLICATION -----	TRANS	RESP			
										TRANS	TRAN/SEC	RESP	APPL			
*ALL	*ALL	644,528	56	0	36	8	0.32	1.6	1:26 9	75037	3.0	0.15	TOSC	344062	0.13	
PRD2	*ALL	644,528	56	0	36	8	0.32	1.6	1:26 9	75037	3.0	0.15	TOSC	344062	0.13	
	LCP1	469,815	77	0	14	9	0.12	6.7	0:36 8	56911	15.8	0.10	TOSC	344062	0.13	
	RCP1	77,933	1	0	91	8	0.13	1.4	0:14 10	10161	2.8	0.14	PUBC	12085	0.16	
	RCP2	68,652	0	0	99	1	0.34	1.3	0:17 10	9283	2.6	0.14	PUBC	12085	0.15	
	RCP3	2,606	29	0	51	19	0.26	0.0	0:01 15	334	0.1	0.16	DATA	768	0.20	
	RCP4	24,056	0	0	96	4	0.61	0.4	0:17 11	3032	0.8	0.70	ISHD	23006	0.62	
	TCT2	1,288	0	0	60	40	0.27	0.0	0:01 13	234	0.1	0.09	CCMF	178	0.01	
	TCT7	178	0	0	4	96	563.10	0.0	0:00 6	19	0.0	0.65	CCMF	168	0.01	
TOTAL # REGIONS: 7																

Figure 3-29. Monthly CICS Workload Summary Report Sample Output

3.1.2.30 CICLMM: Monthly CICS Workload Detail Report

The Monthly CICS Workload Detail Report display CICS activity at the application level and provides workload characteristics and application profile information. It also provides a summary section which shows CICS activity by hour and the top application in transaction volume for each hour.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary section and a detail section for each combination of SYSID and CICSID. The summary section shows CICS activity for each hour. The detail section shows the transaction processed by each application in each CICS region on each system, and provides profile information such as average CPU, average file control calls, average DBMS calls, and average storage used.

A sample output of this inquiry is shown in Figure 3-30. Each field in the report is described below:

SYSTEM SUMMARY:

REGIONS: Total number of SYSID/CICSID combinations found in input.

TRANS: The total number of transactions processed, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.

%SHORT: The percentage of short transactions processed.

%MED: The percentage of medium transactions processed.

%LONG: The percentage of long transactions processed.

% CONV: Percentage of conversational transactions processed.

AVG RESP: The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.

CPU:	The total TCB CPU time consumed.
AVG FC CALLS:	The average number of file control calls issued per transaction.
AVERAGE CHAR:	The average number of input and out characters per transaction generated by this application.
TOP APPL:	The application for which CICS processed the highest number of transaction in an hour.
TOP TRANS:	The total number transactions processed in an hour for the top application.
TOP RESP:	The average response time received by the top application for the hour.
TOP CPU:	The average TCB CPU time consumed by the top application for the hour.
TOP AVG FC CALLS:	The average number of file control calls issued per transaction by the top application.
TOP AVG CHAR:	The average number of input and output characters generated per transaction by the top application.
TOP AVG MSG:	The average number of input and output messages generated per transaction by the top application.
DETAIL SECTION:	
SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.

APPL:	Application identification.
TRANS:	The total number of transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
AVG RESP:	The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
TOTAL CPU:	The total TCB CPU time consumed by all transactions, excluding transaction type X.
FC CALLS:	The total number of file control calls issued by this application.
DBMS CALLS:	The total number of data base calls (DB2 and IMS) issued by this application.
TOT CHARS:	The total number of input and output characters generated by this application.
MAX STORAGE:	The maximum amount (in 1024 bytes) of terminal and user storage used by this application.
AVERAGE CPU:	The average TCB CPU per transaction consumed by this application.
AVERAGE FC CALLS:	The average number of file control calls per transaction issued by this application.
AVERAGE DBMS CALLS:	The average number of data base (DB2 and IMS) calls per transaction issued by this application.
AVERAGE	The average number of input and out characters

CHAR: per transaction generated by this application.

AVERAGE STORAGE: The average amount of dynamic storage (terminal and user) per transaction consumed by this application.

% ABND: The percentage of transactions that abended for this application.

INQUIRY ID:

CICLMM (tabular report)

DATA SOURCE (file/time-span):

CICCAU01 in the MONTH time-span.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAUPU.

DATA ELEMENTS USED:

CICAUPU - CICS Application Unit ID
CICCSID - CICS System Identifier
CAUABEND - Abnormal Termination Occurrences
CAUCTRN - Conv. Transactions Processed
CAUETRN - Excluded Transactions
CAUDLICC - Number of CICS DL/I Calls
CAUFCOPS - Number of CICS File Control Calls
CAUINCH - Input Message Character Traffic
CAULTRN - Long Transactions Processed
CAUMEMRC - Memory Usage Running Count
CSUMTRN - Medium Transactions Processed
CAUMXMEM - Maximum Transaction Storage Used
CAUOUTCH - Output Message Character Traffic
CAUCPUTM - Task CPU Time
CAUSQLTO - Total number of SQL Calls
CSUSTRN - Short Transactions Processed
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS
Average character = $\text{SUM}(\text{CAUINCH}, \text{CAUOUTCH}) / \text{SUM}(\text{CAUTRANS}, \text{CAUETR N})$
Average CPU time = CAUCPUTM/CAUTRANS
Average DBMS calls = $\text{SUM}(\text{CAUDLIC C}, \text{CAUSQLT O}, \text{CICPCT01}) / \text{SUM}(\text{CAUTRANS}, \text{CAUETR N})$
Average file control calls = CAUFCOPS/
 $\text{SUM}(\text{CAUTRANS}, \text{CAUETR N})$
Average storage = CAUMEMRC/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETR N})$
Percent abend = CAUABEND/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETR N})$
Percent conv. = CAUCTRN/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETR N})$
Percent long = CAULTRN/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETR N})$
Percent medium= CAUMTRN/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETR N})$
Percent short = CAUSTRN/ $\text{SUM}(\text{CAUTRANS}, \text{CAUETR N})$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time

zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICA T_x data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1 MICF Inquiries

MONTHLY CICS WORKLOAD SUMMARY REPORT													
your_company_name													
FOR AUGUST, 1991													
INQUIRY: CICLMM										RUN DATE: 06SEP91			
-----SYSID=PRD2 CICSID=LCP1-----													
APPL	TOTAL TRANS	AVG RESP	TOT CPU	FILE CNTL CALLS	DBMS CALLS	TOT CHARS (000)	MAX STORAGE	CPU	FC	CL	AVERAGES DBMS CL	CHAR STORAGE	% ABND
AAON	1	34.10	0:00	1	0	0	157K	0.08	1	0	0	157K	0
ACFM	163	0.02	0:00	0	0	162	72K	0.01	0	0	994	70K	0
CATD	15,759	0.07	0:02	0	0	0	297K	0.01	0	0	0	301K	0
CCMF	233	0.00	0:00	0	0	0	4K	0.00	0	0	0	4K	0
CEDA	21	0.20	0:00	82	0	13	282K	0.01	4	0	639	205K	0
CEMT	10	0.07	0:00	0	0	3	91K	0.01	0	0	347	90K	0
CRSQ	3	0.27	0:00	0	0	0	4K	0.00	0	0	0	4K	0
CRSR	29,214	0.02	0:01	0	0	0	11K	0.00	0	0	0	11K	0
CRTE	691	0.02	0:00	0	0	258	11K	0.00	0	0	373	11K	0
CSAC	350	0.00	0:00	0	0	33	11K	0.00	0	0	94	10K	0
CSFU	1	0.83	0:00	0	0	0	35K	0.01	0	0	0	35K	0
CSGM	16,516	0.03	0:01	84	0	14931	10K	0.00	0	0	904	9K	0
CSIR	61	47.96	0:00	0	0	0	5K	0.00	0	0	6	5K	0
CSMI	11	0.01	0:00	0	0	4	10K	0.00	0	0	396	10K	0
CSNE	23,797	0.00	0:01	0	0	0	27K	0.00	0	0	0	27K	0
CSSF	38	0.01	0:00	0	0	10	19K	0.00	0	0	268	19K	0
CSSN	6	0.03	0:00	0	0	4	19K	0.00	0	0	597	19K	0
DATA	14,588	0.14	0:01	21	0	9544	197K	0.00	0	0	654	16K	0
DISP	1	0.03	0:00	0	0	0	13K	0.02	0	0	314	13K	1
D2DA	15	1.42	0:00	0	0	21	16K	0.00	0	0	1425	39K	0
EL0N	10	0.24	0:00	0	0	8	198K	0.01	0	0	784	126K	0
EL00	40	0.06	0:00	0	0	35	8K	0.00	0	0	875	7K	0
EL10	9	0.04	0:00	0	0	6	8K	0.00	0	0	631	6K	0
EL20	41	0.15	0:00	0	0	54	8K	0.00	0	0	1322	7K	0
EL22	50	0.14	0:00	0	0	57	8K	0.00	0	0	1150	8K	0
EL29	21	0.11	0:00	0	0	21	8K	0.00	0	0	1011	7K	0
EL30	14	0.07	0:00	0	0	7	8K	0.00	0	0	491	6K	0
EZ	324	0.46	0:00	0	0	519	197K	0.00	0	0	1602	14K	0
IE0L	1,346	0.25	0:00	0	0	1147	13K	0.00	0	0	852	13K	0
LLTP	1	0.01	0:00	0	0	0	12K	0.00	0	0	136	12K	1
LLXW	4	0.04	0:00	0	0	1	11K	0.01	0	0	273	12K	1
LPKD	3	0.00	0:00	0	0	1	10K	0.00	0	0	376	8K	0
LPKE	12	0.00	0:00	0	0	18	8K	0.00	0	0	1460	8K	0
MSAS	69	0.17	0:00	0	0	69	8K	0.00	0	0	1007	11K	0
MSA2	3,240	0.13	0:00	57	0	5251	8K	0.00	0	0	1621	8K	0
SA	39	0.48	0:00	21	0	36	198K	0.00	1	0	927	90K	0
STAR	21	0.00	0:00	0	0	29	8K	0.00	0	0	1399	9K	0
TOSC	344,062	0.13	0:11	42	0	416132	8K	0.00	0	0	1209	8K	0
TOSS	2,558	0.40	0:00	0	0	3937	198K	0.01	0	0	1539	132K	0
TOS1	8,580	0.38	0:01	0	0	13190	197K	0.01	0	0	1537	135K	0
TOTAL :	469,835	0.12	0:18	308	0	465952	297K	0.00	0	0	992	22K	0

3.1.3 Performance Analysis Inquiries

Performance Analysis Inquiries provide CICS system performance information to help you manage and tune CICS. The inquiries are available for daily, weekly, and monthly reporting periods. The CICS Analyzer distributes the following performance analysis inquiries:

Catalog Group	Inquiry ID	Inquiry Name	Report format	Run-time execution parameters	User exits
PERFORM	CICLDP	Daily CICS System Performance Summary	tabular		
	CICLMP	Monthly CICS System Performance Summary	tabular		
	CICLWP	Weekly CICS System Performance Summary	tabular		

Figure 3-31. Distributed Performance Analysis Inquiries

The performance analysis inquiries are described individually in the following sections:

- 1 - CICLDP: Daily CICS Performance Summary
- 2 - CICLMP: Monthly CICS Performance Summary
- 3 - CICLWP: Weekly CICS Performance Summary

3.1.3.1 CICLDP: Daily CICS Performance Summary Report

The Daily CICS Performance Summary Report shows the workload, service, and percent availability of each CICS region on a given day. The report also shows the occurrences of abnormal conditions that impact CICS performance, such as high number of program compressions, short-on-storage, and maximum tasks conditions. Once a problem has been detected using this report, you can then select one of the detail reports in the service or workload area to further investigate the cause of the problem.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for overall CICS performance for the enterprise (summed across SYSIDs and CICSs), as well as CICS performance at the system level (summarized by SYSID). A sample output of this inquiry is shown in Figure 3-32. Each field in the report is described below:

SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
AVG RESP:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICLRT exit in prefix.MICS.PARMS.
TRANS:	The total number of transactions processed, excluding those assigned as TRANTYPE X in your CICLRT exit in prefix.MICS.PARMS.
%AVAIL:	The CICS availability expressed in a percentage. This number is derived by dividing the total CICS uptime by the total report duration.
CPU TOTAL:	The total amount of TCB and SRB time consumed. This also includes subtask time for CMF.
CPU %USER:	Percentage of total CPU time consumed by user transactions.

CPU Percentage of total CPU time consumed by task
 %KCP: control (system overhead).

CPU Percentage of total CPU time consumed by journal
 %JCP: control (system overhead).

CPU Percentage of total CPU time consumed by terminal
 %TCP: control (system overhead).

CPU Percentage of SRB time out of total CPU time.
 %SRB:

CPU Percentage of subtask time out of total CPU time.
 %SUB:

DSA SIZE: The size of CICS dynamic storage area as defined
 in the SIT (system initialization table).

DSA HWM: The maximum dynamic storage area used.

COMPRS: The number of program compressions performed.

SOS: The number of time CICS experienced short-on-
 storage condition due to release of the storage
 cushion.

MAXTASK: The number of time CICS experienced MAXTASK
 conditions and stopped initiating new tasks.

VIOL: The number of transactions with storage
 violation.

% ABND: The percentage of transactions that abended for
 this application.

INQUIRY ID:

CICLDP (tabular report)

DATA SOURCE (file/timespan):

CICCSY01 in the DAYS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
 CSYABEND - Abnormal Terminations
 CSYADSTM - Total Address Space Time
 CSYCPUTM - Total CPU Time

CSYDAMSC - Storage Violations
CSYDSASZ - DSA Size
CSYETRN - Excluded transactions
CSYMEMSZ - DSA High Water Mark
CSYMXTSK - Max Task Conditions
CSYPCMDC - Number of Program Compressions
CSYSOS - Short-on-storage Conditions
CSYSRBTM - SRB Time
CSYTCPTM - Terminal Control CPU Time
CSYTCUTM - User CPU Time
CSYCPJTM - JCP CPU Time
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYTSRTM - Task Control CPU Time
CSYUPTM - CICS Availability Time
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = $CSYTRSTM / CSYTRANS$
Percent abend = $CSYABEND / SUM(CSYTRANS, CSYETRN)$
Percent availability = $CSYUPTM / \text{calculated up time}$
Percent user CPU = $CSYTCUTM / CSYADSTM * 100$
Percent KCP CPU = $CSYTSRTM / CSYADSTM * 100$
Percent JCP CPU = $CSYCPJTM / CSYADSTM * 100$
Percent TCP CPU = $CSYTCPTM / CSYADSTM * 100$
Percent SRB CPU = $CSYSRBTM / CSYADSTM * 100$
Percent SUB CPU = $(CSYADSTM - CSYTCBTM) / CSYADSTM * 100$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

DAILY CICS PERFORMANCE SUMMARY REPORT																	
your_company_name																	
FOR: MARCH 15, 2007																	
INQUIRY: CICLDP															RUN DATE: 15MAR07		
SYSID	CICSID	AVG RESP	TRANS	% SHORT	% MED	% LONG	% CONV	----- CPU -----					COMPRS	SOS	MAXTSK	VIOL	% ABND
*ALL	*ALL	0.32	644,528	56	0	36	8	1:42	57	13	14	16	0	0	62	0	0
PRD2	*ALL	0.32	644,528	56	0	36	8	1:42	57	13	14	16	0	0	62	0	0
	LCP1	0.12	469,815	77	0	14	9	0:43	51	13	19	17	0	0	61	0	0
	RCP1	0.13	77,933	1	0	91	8	0:17	51	13	19	17	0	0	0	0	0
	RCP2	0.34	68,652	0	0	99	1	0:19	72	11	4	13	0	0	0	0	0
	RCP3	0.26	2,606	29	0	51	19	0:01	59	13	8	20	0	0	0	0	0
	TCP4	0.61	24,056	0	0	96	4	0:20	65	11	8	17	0	0	1	0	0
	TCT2	0.27	1,288	0	0	60	40	0:01	26	41	18	15	0	0	0	0	0
	TCT7	563.10	178	0	0	4	96	0:00	25	43	14	18	0	0	0	0	0
TOTAL # REGIONS: 7																	

Figure 3-32. Daily CICS Performance Summary Report Sample Output

3.1.3.2 CICLMP: Monthly CICS Performance Summary Report

The Monthly CICS Performance Summary Report shows the workload, service, and percent availability of each CICS region in a given month. The report also shows the occurrences of abnormal conditions that impact CICS performance, such as high number of program compressions, short-on-storage, and maximum tasks conditions. Once a problem has been detected using this report, you can then select one of the detail reports in the service or workload area to further investigate the cause of the problem.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for overall CICS performance for the enterprise (summed across SYSIDs and CICSs), as well as CICS performance at the system level (summarized by SYSID). A sample output of this inquiry is shown in Figure 3-33. Each field in the report is described below:

SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
AVG RESP:	The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICLRT exit).
TRANS:	The total number of transactions processed, excluding those assigned as TRANSTYPE X in your CICLRT exit in prefix.MICS.PARMS.
%AVAIL:	The CICS availability expressed in a percentage. This number is derived by dividing the total CICS uptime by the total report duration.
CPU TOTAL:	The total amount of TCB and SRB time consumed. This also includes subtask time for CMF.
CPU %USER:	Percentage of total CPU time consumed by user transactions.

CPU %KCP:	Percentage of total CPU time consumed by task control (system overhead).
CPU %JCP:	Percentage of total CPU time consumed by journal control (system overhead).
CPU %TCP:	Percentage of total CPU time consumed by terminal control (system overhead).
CPU %SRB:	Percentage of SRB time out of total CPU time.
CPU %SUB:	Percentage of subtask time out of total CPU time.
DSA SIZE:	The size of CICS dynamic storage area as defined in the SIT (system initialization table).
DSA HWM:	The maximum dynamic storage area used.
COMPRS:	The number of program compressions performed.
SOS:	The number of time CICS experienced short-on-storage condition due to release of the storage cushion.
MAXTASK:	The number of time CICS experienced MAXTASK conditions and stopped initiating new tasks.
VIOL:	The number of transactions with storage violation.
% ABND:	The percentage of transactions that abended for this application.

INQUIRY ID:

CICLMP (tabular report)

DATA SOURCE (file/timespan):

CICCSY01 in the MONTHS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
CSYABEND - Abnormal Terminations

CSYADSTM - Total Address Space Time
 CSYCPUTM - Total CPU Time
 CSYDAMSC - Storage Violations
 CSYDSASZ - DSA Size
 CSYETRN - Excluded transactions
 CSYMEMSZ - DSA High Water Mark
 CSYMXTSK - Max Task Conditions
 CSYPCMDC - Number of Program Compressions
 CSYSOS - Short-on-storage Conditions
 CSYSRBTM - SRB Time
 CSYTCPTM - Terminal Control CPU Time
 CSYTCUTM - User CPU Time
 CSYCPJTM - JCP CPU Time
 CSYTRANS - Transactions Processed
 CSYTRSTM - Transaction Response Time Total
 CSYTSRTM - Task Control CPU Time
 CSYUPTM - CICS Availability Time
 MONTH - Month of Year
 YEAR - Year of Century
 ZONE - Time Zone
 SYSID - System Identifier

CALCULATIONS:

Average response time = $CSYTRSTM / CSYTRANS$
 Percent abend = $CSYABEND / SUM(CSYTRANS, CSYETRN)$
 Percent availability = $CSYUPTM / \text{calculated up time}$
 Percent user CPU = $CSYTCUTM / CSYADSTM * 100$
 Percent KCP CPU = $CSYTSRTM / CSYADSTM * 100$
 Percent JCP CPU = $CSYCPJTM / CSYADSTM * 100$
 Percent TCP CPU = $CSYTCPTM / CSYADSTM * 100$
 Percent SRB CPU = $CSYSRBTM / CSYADSTM * 100$
 Percent SUB CPU = $(CSYADSTM - CSYTCBTM) / CSYADSTM * 100$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to

be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

MONTHLY CICS PERFORMANCE SUMMARY REPORT																	
your_company_name																	
FOR FEBRUARY, 2007																	
INQUIRY: CICLMP														RUN DATE: 02MAR07			
SYSID	CICSID	AVG RESP	TRANS	% SHORT	% MED	% LONG	% CONV	----- CPU -----					COMPRS	SOS	MAXTSK	VIOL	% ABND
*ALL	*ALL	0.32	644,528	56	0	36	8	1:42	57	13	14	16	0	0	62	0	0
PRD2	*ALL	0.32	644,528	56	0	36	8	1:42	57	13	14	16	0	0	62	0	0
	LCP1	0.12	469,815	77	0	14	9	0:43	51	13	19	17	0	0	61	0	0
	RCP1	0.13	77,933	1	0	91	8	0:17	51	13	19	17	0	0	0	0	0
	RCP2	0.34	68,652	0	0	99	1	0:19	72	11	4	13	0	0	0	0	0
	RCP3	0.26	2,606	29	0	51	19	0:01	59	13	8	20	0	0	0	0	0
	TCP4	0.61	24,056	0	0	96	4	0:20	65	11	8	17	0	0	1	0	0
	TCT2	0.27	1,288	0	0	60	40	0:01	26	41	18	15	0	0	0	0	0
	TCT7	563.10	178	0	0	4	96	0:00	25	43	14	18	0	0	0	0	0
TOTAL # REGIONS: 7																	

Figure 3-33. Monthly CICS Performance Summary Report Sample Output

3.1.3.3 CICLWP: Weekly CICS Performance Summary Report

The Weekly CICS Performance Summary Report shows the workload, service, and percent availability of each CICS region in a given week. The report also shows the occurrences of abnormal conditions that impact CICS performance, such as high number of program compressions, short-on-storage, and maximum tasks conditions. Once a problem has been detected using this report, you can then select one of the detail reports in the service or workload area to further investigate the cause of the problem.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for overall CICS performance for the enterprise (summed across SYSIDs and CICSs), as well as CICS performance at the system level (summarized by SYSID). A sample output of this inquiry is shown in Figure 3-34. Each field in the report is described below:

SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
AVG RESP:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICLRT exit in prefix.MICS.PARMS.
TRANS:	The total number of transactions processed, excluding those assigned as TRANTYPE X in your CICLRT exit in prefix.MICS.PARMS.
%AVAIL:	The CICS availability expressed in a percentage. This number is derived by dividing the total CICS uptime by the total report duration.
CPU TOTAL:	The total amount of TCB and SRB time consumed. This also includes subtask time for CMF.
CPU %USER:	Percentage of total CPU time consumed by user transactions.

CPU Percentage of total CPU time consumed by task
 %KCP: control (system overhead).

CPU Percentage of total CPU time consumed by journal
 %JCP: control (system overhead).

CPU Percentage of total CPU time consumed by terminal
 %TCP: control (system overhead).

CPU Percentage of SRB time out of total CPU time.
 %SRB:

CPU Percentage of subtask time out of total CPU time.
 %SUB:

DSA SIZE: The size of CICS dynamic storage area as defined
 in the SIT (system initialization table).

DSA HWM: The maximum dynamic storage area used.

COMPRS: The number of program compressions performed.

SOS: The number of time CICS experienced short-on-
 storage condition due to release of the storage
 cushion.

MAXTASK: The number of time CICS experienced MAXTASK
 conditions and stopped initiating new tasks.

VIOL: The number of transactions with storage
 violation.

% ABND: The percentage of transactions that abended for
 this application.

INQUIRY ID:

CICLWP (tabular report)

DATA SOURCE (file/timespan):

CICCSY01 in the WEEKS timespan.

DATA ELEMENTS USED:

CICSID - CICS System Identifier
 CSYABEND - Abnormal Terminations
 CSYADSTM - Total Address Space Time
 CSYCPUTM - Total CPU Time

CSYDAMSC - Storage Violations
CSYDSASZ - DSA Size
CSYETRN - Excluded transactions
CSYMEMSZ - DSA High Water Mark
CSYMXTSK - Max Task Conditions
CSYPCMDC - Number of Program Compressions
CSYSOS - Short-on-storage Conditions
CSYSRBTM - SRB Time
CSYTCPTM - Terminal Control CPU Time
CSYTCUTM - User CPU Time
CSYCPJTM - JCP CPU Time
CSYTRANS - Transactions Processed
CSYTRSTM - Transaction Response Time Total
CSYTSRTM - Task Control CPU Time
CSYUPTM - CICS Availability Time
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = $CSYTRSTM / CSYTRANS$
Percent abend = $CSYABEND / SUM(CSYTRANS, CSYETRN)$
Percent availability = $CSYUPTM / \text{calculated up time}$
Percent user CPU = $CSYTCUTM / CSYADSTM * 100$
Percent KCP CPU = $CSYTSRTM / CSYADSTM * 100$
Percent JCP CPU = $CSYCPJTM / CSYADSTM * 100$
Percent TCP CPU = $CSYTCPTM / CSYADSTM * 100$
Percent SRB CPU = $CSYSRBTM / CSYADSTM * 100$
Percent SUB CPU = $(CSYADSTM - CSYTCBTM) / CSYADSTM * 100$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter start of the report period in ddmmyy format.

For this inquiry, the default is the previous week.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

3.1 MICF Inquiries

WEEKLY CICS PERFORMANCE SUMMARY REPORT																	
your_company_name																	
FOR WEEK STARTING MARCH 05, 2007																	
INQUIRY: CICLWP														RUN DATE: 16MAR07			
SYSID	CICSID	AVG RESP	TRANS	% SHORT	% MED	% LONG	% CONV	----- CPU -----					COMPRS	SOS	MAXTSK	VIOL	% ABND
								TOTAL	%USER	%KCP	%SRB	%SUB					
*ALL	*ALL	0.32	644,528	56	0	36	8	1:42	57	13	14	16	0	0	62	0	0
PRD2	*ALL	0.32	644,528	56	0	36	8	1:42	57	13	14	16	0	0	62	0	0
	LCP1	0.12	469,815	77	0	14	9	0:43	51	13	19	17	0	0	61	0	0
	RCP1	0.13	77,933	1	0	91	8	0:17	51	13	19	17	0	0	0	0	0
	RCP2	0.34	68,652	0	0	99	1	0:19	72	11	4	13	0	0	0	0	0
	RCP3	0.26	2,606	29	0	51	19	0:01	59	13	8	20	0	0	0	0	0
	TCP4	0.61	24,056	0	0	96	4	0:20	65	11	8	17	0	0	1	0	0
	TCT2	0.27	1,288	0	0	60	40	0:01	26	41	18	15	0	0	0	0	0
TCT7	563.10	178	0	0	4	96	0:00	25	43	14	18	0	0	0	0	0	
TOTAL # REGIONS: 7																	

Figure 3-34. Weekly CICS Performance Summary Report Sample Output

3.1.4 Availability Analysis Inquiries

Availability analysis inquiries quantify the system available time. The inquiries in this category are available for daily, weekly, and monthly reporting periods. The CICS Analyzer distributes the following availability analysis inquiries:

Catalog Group	Inquiry ID	Inquiry Name	Report format	Run-time execution parameters	User exits
AVAIL	CICCDQ	Daily CICS Availability	color graphic		
	CICCD3	Daily CICS Availability	color graphic		
	CICCMQ	Monthly CICS Availability	color graphic		
	CICCM3	Monthly CICS Availability	color graphic		
	CICCWQ	Weekly CICS Availability	color graphic		
	CICCW3	Weekly CICS Availability	color graphic		
	CICPDQ	Daily CICS Availability	printer graphic		
	CICPD3	Daily CICS Availability	printer graphic		
	CICPMQ	Monthly CICS Availability	printer graphic		
	CICPM3	Monthly CICS Availability	printer graphic		
	CICPWQ	Weekly CICS Availability	printer graphic		
	CICPW3	Weekly CICS Availability	printer graphic		

Figure 3-35. Distributed Availability Analysis inquiries

The availability analysis inquiries are described individually in the following sections:

- 1 - CICCDO: Daily CICS Availability
- 2 - CICCD3: Daily CICS Availability
- 3 - CICC MQ: Monthly CICS Availability
- 4 - CICC M3: Monthly CICS Availability
- 5 - CICC WQ: Weekly CICS Availability
- 6 - CICC W3: Weekly CICS Availability
- 7 - CICPDQ: Daily CICS Availability
- 8 - CICPD3: Daily CICS Availability
- 9 - CICPMQ: Monthly CICS Availability
- 10- CICPM3: Monthly CICS Availability
- 11- CICPWQ: Weekly CICS Availability
- 12- CICPW3: Weekly CICS Availability

3.1.4.1 CICCDQ: Daily CICS Availability

The Daily CICS Availability graph shows CICS availability as well as workload on an hourly basis for a given day. It identifies the hour during which CICS availability objective was missed.

REPORT FORMAT:

This inquiry produces an overlay plot for each CICS region found in input. It plots CICS hour availability (percent available) on the left vertical axis, while the corresponding workload on the right. A horizontal reference line is drawn at the user-specified point, the default of which is 100, to indicate the availability objective.

INQUIRY ID:

CICCDQ (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 at the DAYS timespan

DATA ELEMENTS USED:

CICSID - CICS Identification
CSYUPTM - CICS Availability Time
CSYTRANS - Total Transactions Processed
HOUR - Hour of Day
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
SYSID - System Identification

CALCULATIONS:

Percent availability = $CSYUPTM/3600*100$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.4.2 CICCD3: Daily CICS Availability Reports

The Daily CICS Availability Reports monitor CICS availability on a daily basis.

REPORT FORMAT:

This inquiry produces bar charts by system ID and CICS ID showing the amount of downtime that incurred within each hour. The output of this inquiry is in color graphic format. However, you can produce the same charts in printer graphic format by executing the MICF inquiry CICPD3.

INQUIRY ID:

CICCD3 (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01 at the DAYS timespan

DATA ELEMENTS USED:

CSYUPTM - CICS Availability Time

CALCULATIONS:

downtime = 60 - CSYUPTM

or

downtime = 60 if CSY observation is not found for an hour

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is

used. It causes all SYSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

CYCLE - Optionally, define CA MICS input file cycle range.

USER EXITS:

None.

3.1.4.3 CICCMTQ: Monthly CICS Availability

The Monthly CICS Availability graph shows monthly CICS availability as well as workload for the past 6 months. It shows how well CICS has been meeting its service level (availability) objective and the trend in CICS workload.

REPORT FORMAT:

This inquiry produces an overlay plot for each CICS region found in input. It plots CICS hour availability (percent available) on the left vertical axis, while the corresponding workload on the right. A horizontal reference line is drawn at the user-specified point, the default of which is 98, to indicate the availability objective.

INQUIRY ID:

CICCMTQ (color graphic format)

DATA SOURCE (file/timespan):

CICCSY from the MONTHS timespan

DATA ELEMENTS USED:

CICSID - CICS Identification
CSYUPTM - CICS Availability Time
CSYTRANS - Total Transactions Processed
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identification

CALCULATIONS:

Percent availability = $\text{CSYUPTM} / \text{calculated up time based on ZONE} \times 100$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

BY - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source

module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.4.4 CICC3: Monthly CICS Availability Reports

The Monthly CICS Availability Reports monitor CICS availability by CA MICS ZONES in a six-month period.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the amount of uptime per ZONE and MONTH in a six-month period. Any month for which data is unavailable is omitted from the charts. The output of this inquiry is in color graphic format. However, you can produce the same charts by executing the MICF inquiry CICPM3.

INQUIRY ID:

CICC3 (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01-CICCSY06 at the MONTHS timespan

DATA ELEMENTS USED:

CSYUPTM - CICS Availability Time

CALCULATIONS:

downtime = 60 - CSYUPTM

or

downtime = 60 if CSY observation is not found for an hour

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

SYSID - Optionally, specify the SYSID to be reported.

If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

CYCLE - Optionally, define CA MICS input file cycle range.

USER EXITS:

None.

3.1.4.5 CICCWQ: Weekly CICS Availability

The Weekly CICS Availability graph shows weekly CICS availability as well as workload for the last 7 days. It shows how well CICS has been meeting its service level (availability) objective and the trend in CICS workload.

REPORT FORMAT:

This inquiry produces an overlay plot for each CICS region found in input. It plots CICS hour availability (percent available) on the vertical axis, while the corresponding workload on the right. A horizontal reference line is drawn at the user-specified point, the default of which is 98, to indicate the availability objective.

INQUIRY ID:

CICCWQ (color graphic format)

DATA SOURCE (file/timespan):

CICCSY from the DAYS timespan

DATA ELEMENTS USED:

CICSID - CICS Identification
CSYUPTM - CICS Availability Time
CSYTRANS - Total Transactions Processed
WEEK - Week of Year
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identification

CALCULATIONS:

Percent availability = $\text{CSYUPTM} / \text{calculated up time based on ZONE} \times 100$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

- SYSID** - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- CICSID** - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- DATE** - Optionally, enter start of the report period in ddmmmyy format.
- For this inquiry, the default is the last seven days.
- ZONE** - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- BY** - Optionally, specify the BY variable (either SYSID or CICSID) to produce separate graphs for each system. Otherwise, all SYSIDs and CICSIDs are summarized together and displayed on one graph.
- DEVICE** - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.
- FOOTNOTE**-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. Please see Section 3.2 to override the default values.

3.1.4.6 CICCW3: Weekly CICS Availability Reports

The Weekly CICS Availability Reports monitor prime time (8 a.m. to 5 p.m.) CICS availability in a five-day period.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the amount of downtime that incurred in each hour (from 8 a.m. to 5 p.m.) in a five-day period. The output of this inquiry is in color graphic format. However, you can produce the same charts in printer graphic format by executing the MICF inquiry CICPW3.

INQUIRY ID:

CICCW3 (color graphic format)

DATA SOURCE (file/timespan):

CICCSY01-CICCSY07 at the DAYS timespan

Data is selected for HOURS 8-16, excluding DAYNAME='SAT' and DAYNAME='SUN'.

DATA ELEMENTS USED:

CSYUPTM - CICS Availability Time

CALCULATIONS:

downtime = 60 - CSYUPTM

or

downtime = 60 if CSY observation is not found for an hour

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to

be summarized together.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

CYCLE - Optionally, define CA MICS input file cycle range.

USER EXITS:

None.

3.1.4.7 CICPDQ: Daily CICS Availability

The Daily CICS Availability graph shows hour CICS availability for a given day. It identifies the period during the day when CICS availability objective was missed.

REPORT FORMAT:

This inquiry produces a bar chart showing the percent availability for each hour. A horizontal reference line is drawn at the user-specified point, the default of which is 100, to indicate the availability objective.

INQUIRY ID:

CICPDQ (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01 at the DAYS timespan

DATA ELEMENTS USED:

CICSID - CICS Identification
CSYUPTM - CICS Availability Time
HOUR - Hour of Day
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
SYSID - System Identification

CALCULATIONS:

Percent availability = $CSYUPTM/3600*100$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

CYCLE - Optionally, define CA MICS input file cycle range.

USER EXITS:

If the run-time execution parameters are insufficient to select the data you need, you can invoke one of the following global or local user exits to further manipulate the input file. For detailed descriptions on coding the exits, refer to Section 3.2.

Global exits - invoked before run-time parameter macros.

%RPTSEL - Select input data based on any variables in the input file.

%RPTAVAL- Define availability objective.

Local exits - invoked after run-time parameter macros.

%CICPDQ - Select input data based on any variables in the input file.

%PDQAVAIL- Define availability objective.

3.1.4.8 CICPD3: Daily CICS Availability Reports

The Daily CICS Availability Reports monitor CICS availability on a daily basis.

REPORT FORMAT:

This inquiry produces bar charts by system ID and CICS ID showing the amount of downtime that incurred within each hour.

A sample output from this inquiry is shown in Figure 3-36.

INQUIRY ID:

CICPD3 (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01 at the DAYS timespan

DATA ELEMENTS USED:

CSYUPTM - CICS Availability Time

CALCULATIONS:

downtime = 60 - CSYUPTM

or

downtime = 60 if CSY observation is not found for an hour

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is

used. It causes all SYSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

CYCLE - Optionally, define CA MICS input file cycle range.

USER EXITS:

None

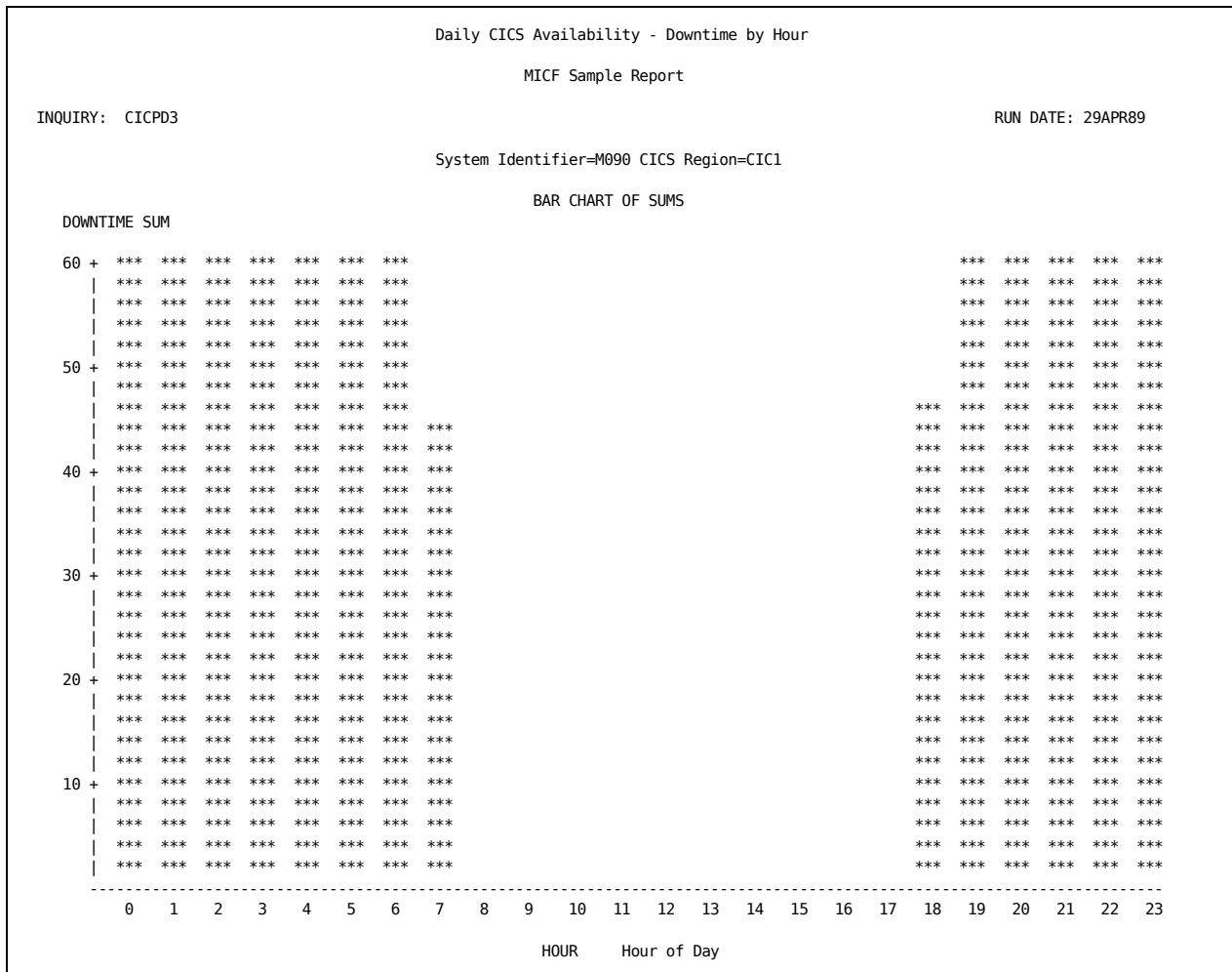


Figure 3-36. Daily CICS Availability Report Sample Output

3.1.4.9 CICPMQ: Monthly CICS Availability

The Monthly CICS Availability graph shows CICS availability for the past 6 months. It identifies the period during which CICS availability objective was missed.

REPORT FORMAT:

This inquiry produces a bar chart showing the percent availability for each month in the last 24 months. A horizontal reference line is drawn at the user-specified point, the default of which is 98, to indicate the availability objective.

INQUIRY ID:

CICPMQ (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY from MONTHS timespan

DATA ELEMENTS USED:

CICSID - CICS Identification
CSYUPTM - CICS Availability Time
MONTH - Month of Year
YEAR - Year of Century
SYSID - System Identification

CALCULATIONS:

Percent availability = $CSYUPTM/3600 \times 100$

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is

used. It causes all SYSIDs found in input to be summarized together.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

CYCLE - Optionally, define CA MICS input file cycle range.

USER EXITS:

If the run-time execution parameters are insufficient to select the data you need, you can invoke one of the following global or local user exits to further manipulate the input file. For detailed descriptions on coding the exits, refer to Section 3.2.

Global exits - invoked before run-time parameter macros.

%RPTSEL - Select input data based on any variables in the input file.

%RPTAVAL- Define availability objective.

Local exits - invoked after run-time parameter macros.

%CICPDQ - Select input data based on any variables in the input file.

%PDQAVAL- Define availability objective.

3.1.4.10 CICPM3: Monthly CICS Availability Reports

The Monthly CICS Availability Reports monitor CICS availability by CA MICS ZONEs in a six-month period.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the amount of uptime per ZONE and MONTH in a six-month period. Any month for which data is unavailable is omitted from the charts.

A sample output from this inquiry is shown in Figure 3-38. (Due to space constraints in the guide, the sample report shows a five month period.)

INQUIRY ID:

CICPM3 (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01-CICCSY06 at the MONTHS timespan

DATA ELEMENTS USED:

CSYUPTM - CICS Availability Time

CALCULATIONS:

downtime = 60 - CSYUPTM

or

downtime = 60 if CSY observation is not found for an
hour

EXECUTION-TIME PARAMETERS:

- UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
- CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.
- SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.
- ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.
- CYCLE - Optionally, define CA MICS input file cycle range.

USER EXITS:

None

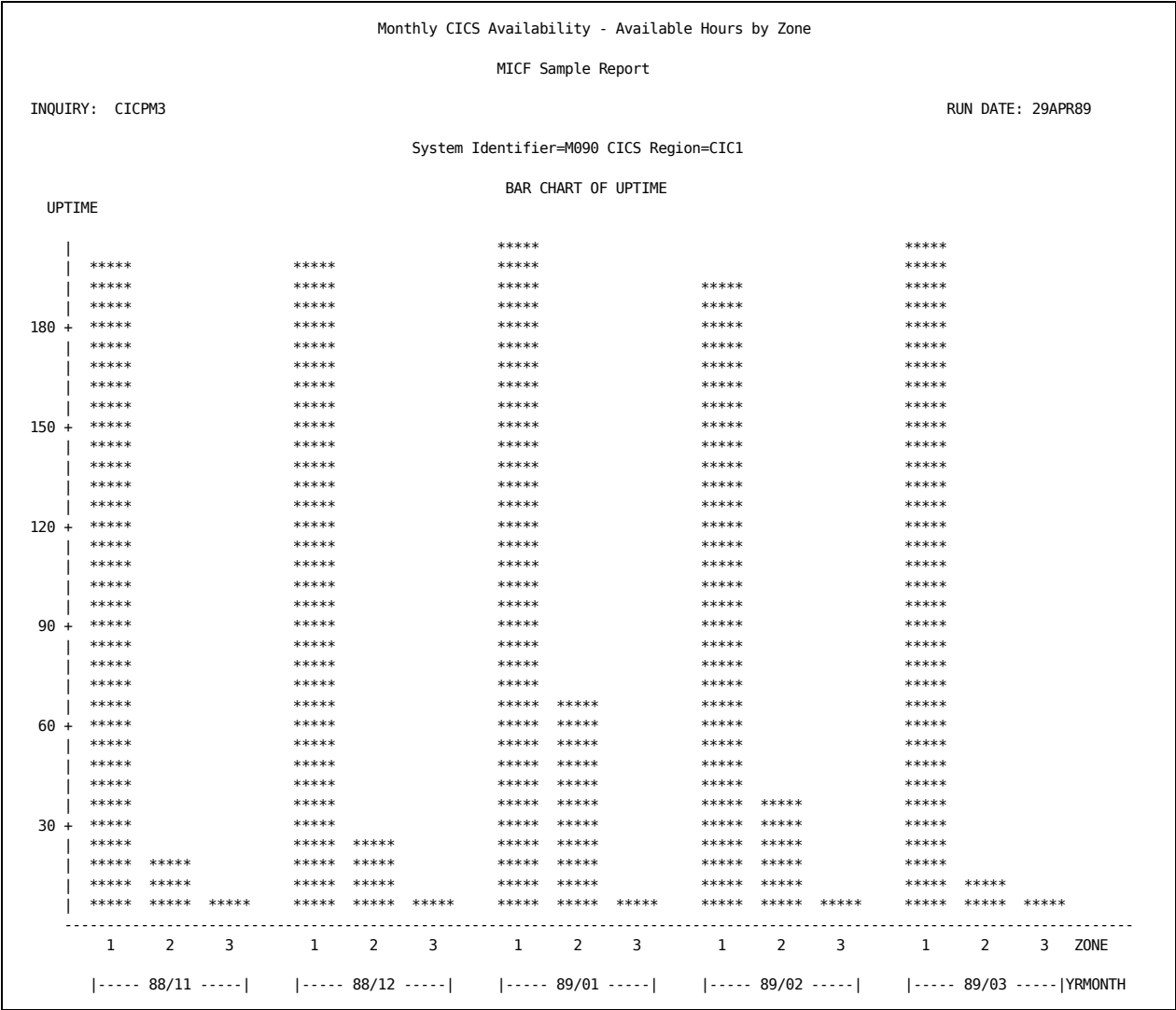


Figure 3-38. Monthly CICS Availability Report Sample Output

3.1.4.11 CICPWQ: Weekly CICS Availability

The Weekly CICS Availability graph shows CICS availability for the last 7 days. It identifies the period during which CICS availability objective was missed.

REPORT FORMAT:

This inquiry produces a horizontal bar chart showing the percent availability for each CICS region in the last 7 days. A vertical reference line is drawn at the user-specified point, the default of which is 98, to indicate the availability objective.

INQUIRY ID:

CICPWQ (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY from weekly history file

DATA ELEMENTS USED:

CICSID - CICS Identification
CSYUPTM - CICS Availability Time
WEEK - Week of Year
MONTH - Month of Year
YEAR - Year of Century
SYSID - System Identification

CALCULATIONS:

Total avail = 86400 - any excluded zones
Percent availability = CSYUPTM/total avail * 100

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter start of the report period in ddmmyy format.

For this inquiry, the default is the previous week.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs.
CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

3.1.4.12 CICPW3: CICS Availability Reports

The Weekly CICS Availability Reports monitor prime time (8 a.m. to 5 p.m.) CICS availability in a five-day period.

REPORT FORMAT:

This inquiry produces side-by-side bar charts by system ID and CICS ID showing the amount of downtime that incurred in each hour (from 8 a.m. to 5 p.m.) in a five-day period.

A sample output of this inquiry is shown in Figure 3-41.

INQUIRY ID:

CICPW3 (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01-CICCSY07 at the DAYS timespan

Data is selected for HOURS 8-16, excluding DAYNAME='SAT' and DAYNAME='SUN'.

DATA ELEMENTS USED:

CSYUPTM - CICS Availability Time

CALCULATIONS:

downtime = 60 - CSYUPTM

or

downtime = 60 if CSY observation is not found for an hour

3.1.5 Financial Analysis Inquiries

Financial analysis inquiries quantify the chargeable resources consumed by CICS users time. The inquiries in this category are available for daily and monthly reporting periods. The CICS Analyzer distributes the following availability analysis inquiries:

Catalog Group	Inquiry ID	Inquiry Name	Report format	Run-time execution parameters	User exits
FINANCE	CICCDR	Daily CICS Cost Ranking	color graphic		
	CICCMR	Monthly CICS Cost Ranking	color graphic		
	CICCMU	Monthly CICS Cost	color graphic		
	CICCMY	Monthly CICS Cost Trend	color graphic		
	CICPDR	Daily CICS Cost Ranking	printer graphic		
	CICPMR	Monthly CICS Cost Ranking	printer graphic		
	CICPMU	Monthly CICS Cost	printer graphic		
	CICLDR	Daily CICS Cost Ranking Report	tabular		
	CICLDW	Daily CICS Cost Summary Report	tabular		
	CICLDX	Daily CICS Cost Detail Report	tabular		
	CICLMR	Monthly CICS Cost Ranking	tabular		
	CICLMW	Monthly CICS Cost Summary Report	tabular		
	CICLMX	Monthly CICS Cost Detail Report	tabular		

Figure 3-42. Distributed Financial Analysis inquiries

To provide report flexibility, the CICS Analyzer supplies MICF execution run time parameters as well as user exits for the financial analysis inquiries. You can use either method to tailor the inquiries to suit your reporting needs.

The financial analysis inquiries are described individually in the following sections:

- 1 - CICCDR: Daily CICS Cost Ranking
- 2 - CICCMR: Monthly CICS Cost Ranking
- 3 - CICCMU: Monthly CICS Cost
- 4 - CICCMY: Monthly CICS Cost Trend
- 5 - CICPDR: Daily CICS Cost Ranking
- 6 - CICPMR: Monthly CICS Cost Ranking
- 7 - CICPMU: Monthly CICS Cost
- 8 - CICLDR: Daily CICS Cost Ranking Report
- 9 - CICLDW: Daily CICS Cost Summary Report
- 10 - CICLDX: Daily CICS Cost Detail Report
- 11 - CICLMR: Monthly CICS Cost Ranking
- 12 - CICLMW: Monthly CICS Cost Summary Report
- 13 - CICLMX: Monthly CICS Cost Detail Report

3.1.5.1 CICCDR: Daily CICS Cost Ranking

The Daily CICS Cost Ranking graph identifies the top 10 consumers of CICS chargeable resources in a given day.

REPORT FORMAT:

This inquiry produces a side-by-side horizontal bar chart showing the top 10 applications in processing charges. The chart also shows the total transactions for each the 10 applications in the same order as the cost ranking. This chart is available in color graphic format only.

INQUIRY ID:

CICCDR (color graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUS (CICS User Activity) file will be used for reporting. The variable CICT2 will be used in place of the variable CICAUP.

DATA ELEMENTS USED:

CICAUP	- CICS Application Unit ID
CICSID	- CICS System Identifier
CAUCOST	- Processing Charges
CAUTRANS	- Transactions Processed
CAUSTRN	- Short Transactions Processed
CAUMTRN	- Medium Transactions Processed
CAULTRN	- Long Transactions Processed
CAUCTRN	- Conversational Transactions Processed
DAY	- Day of Month
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by

entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

DATE - Optionally, enter the date to be reported in ddmmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the

next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICTACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICTRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

3.1.5.2 CICC MR: Monthly CICS Cost Ranking

INQUIRY ID:

CICCMR (color graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICC SU (CICS User Activity) file will be used for reporting. The variable CICT2 will be used in place of the variable CICA PU.

DATA ELEMENTS USED:

CICA PU - CICS Application Unit ID
CICSID - CICS System Identifier
CAUCOST - Processing Charges
CAUTRANS - Transactions Processed
CAUSTRN - Short Transactions Processed
CAUMTRN - Medium Transactions Processed
CAULTRN - Long Transactions Processed
CAUCTRN - Conversational Transactions Processed
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is

used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICAx data elements, where x is 1 to 9. This parameter must be specified in conjunction with

FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

- %GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.
- %GCIC - Select CICSIDs for report.
- %GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.
- %GSYS - Select SYSIDs for report.

3.1.5.3 CICCMS: Monthly CICS Cost

The Monthly CICS Cost graph shows the percentage of monthly CICS cost associated with each system (SYSID) or data center (group of SYSIDs) in the enterprise.

REPORT FORMAT:

This inquiry produces a pie chart showing the percentage of processing cost associated with each system or data center. When there are more than 10 SYSIDs or grouping of SYSIDs, all systems which consumed less than 5% of the total cost will be grouped together. When there is only one SYSID present, the pie slices will show the CICSIDs values. The output of this inquiry is in color graphic format. However, you can produce the same chart in printer graphic format by executing the MICF inquiry CICPMU.

INQUIRY ID:

CICCMU (color graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICALPU.

DATA ELEMENTS USED:

CAUCOST	- Processing Charges
CICALPU	- CICS Application Unit ID
CICSID	- CICS System Identifier
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

CENTER - Optionally, specify the CENTER to be reported. If no value is specified, the value *ALL is used which causes all CENTERS found in input to be summarized together. CENTER is a logical grouping of SYSIDs which you assign in the global exit %GCNTR. If this exit is not defined, CENTER is set to the value of SYSID.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

DEVICE - Optionally, override the default colors and patterns used by specifying a different device type. The default is TERMINAL. Because the code for color graphics is not generated by MICF, the device parameter you specified in MICF is not applicable here.

FOOTNOTE-Optionally, specify whether or not the MICF inquiry ID should be displayed as a footnote on the graph. The default value is NO.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the

next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

3.1.5.4 CICCMy: Monthly CICS Cost Trend

This inquiry is unavailable at this time.

3.1.5.5 CICPDR: Daily CICS Cost Ranking

The Daily CICS Cost Ranking graph identifies the top 10 consumers of CICS chargeable resources in a given day.

REPORT FORMAT:

This inquiry produces a horizontal bar chart showing the top 10 applications in processing charges.

The output of this inquiry is shown in Figure 3-43.

INQUIRY ID:

CICPDR (printer graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAPU.

DATA ELEMENTS USED:

CICAPU	- CICS Application Unit ID
CICSID	- CICS System Identifier
CAUCOST	- Processing Charges
DAY	- Day of Month
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT	- Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
CYCLE	- Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

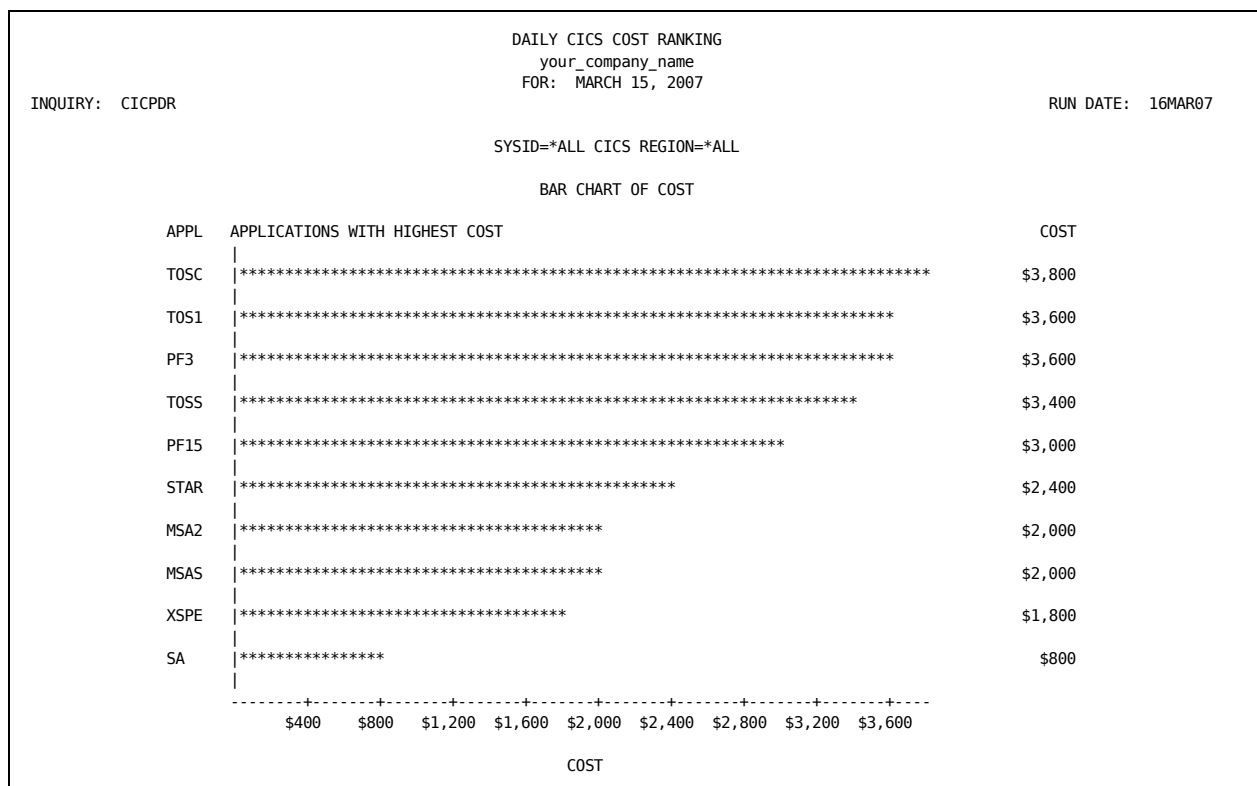


Figure 3-43. Daily CICS Cost Ranking Sample Output

3.1.5.6 CICPMR: Monthly CICS Cost Ranking

The Monthly CICS Cost Ranking graph identifies the top 10 consumers of CICS chargeable resources in a given month.

REPORT FORMAT:

This inquiry produces a horizontal bar chart showing the top 10 applications in processing charges.

The output of this inquiry is shown in Figure 3-44.

INQUIRY ID:

CICPMR (printer graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAPU.

DATA ELEMENTS USED:

CICAPU	- CICS Application Unit ID
CICSID	- CICS System Identifier
CAUCOST	- Processing Charges
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT	- Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.
CYCLE	- Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported.
If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

For this inquiry, the default is the previous month.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

Chapter 3: REPORTS 337

3.1.5.7 CICIPMS: Monthly CICS Cost

The Monthly CICS Cost graph shows the percentage of monthly CICS cost associated with each system (SYSID) or data center (group of SYSIDs) in the enterprise.

REPORT FORMAT:

This inquiry produces a pie chart showing the percentage of processing cost associated with each system or data center. When there are more than 10 SYSIDs or grouping of SYSIDs, all systems which consumed less than 5% of the total cost will be grouped together. When there is only one SYSID present, the pie slices will show the CICSIDs values.

INQUIRY ID:

CICPMU (printer graphic format)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUS (CICS User Activity) file will be used for reporting. The variable CICT2 will be used in place of the variable CICAU.

One graph produced for each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICAPU	- CICS Application Unit ID
CICSID	- CICS System Identifier
CAUCOST	- Processing Charges
MONTH	- Month of Year
YEAR	- Year of Century
ZONE	- Time Zone
SYSID	- System Identifier

CALCULATIONS:

None

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

3.1.5.8 CICLDR: Daily CICS Cost Ranking Report

The Daily CICS Cost Ranking Report identifies the top 10 consumers in CICS chargeable resource.

REPORT FORMAT:

This inquiry produces a tabular report which contains a summary section and a ranking section. The summary section shows the total cost for the enterprise and for each system. The ranking section identifies the 10 applications with the highest processing cost and provides total consumption of resources that are most often used for charging.

A sample output of this inquiry is shown in Figure 3-46. Each field in this report is described below:

SYSTEM SUMMARY:

- # REGIONS: Total number of SYSID/CICSID combinations found in input.
- SYSID: System identification. This field is set to ALL for data that is summarized for all systems.
- TRANS: The total number of transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
- COST (000): The total processing charges in thousands of dollars.
- COST/
TRAN: The average cost per transaction.
- AVG RESP: The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
- TOTAL CPU: The total TCB CPU time consumed by all transactions, excluding transaction type X.

RANKING SECTION:

- RANK: The ranking, number 1 to 10, in processing cost.
- APPL: Application identification.

COST: The total processing charges for the application.

TRANS: The total number of transactions processed for the application.

COST/
TRAN: The average cost per transaction.

AVG RESP: The average response time for all transactions processed for this application.

CPU: The total TCB CPU time consumed.

FC CALLS: The total number of file control calls issued by this application.

DBMS CALLS: The total number of data base calls (DB2 and IMS) issued by this application.

MSGs The total number of input and output messages generated by this application (in thousands).

CHAR (000): The total number of input and output characters generated by this application (in thousands).

AVG
STORAGE: The average amount (in 1024 bytes) of terminal and user storage used by this application.

INQUIRY ID:

CICLDR (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICALPU.

One graph produced for each date and each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICALPU - CICS Application Unit ID
CICSID - CICS System Identifier

CAUCOST - Processing Charges
CAUDLICC - Number of CICS DL/I Calls
CAUFCOPS - Number of CICS File Control Calls
CAUINCH - Input Message Character Traffic
CAUMEMRC - Memory Usage Running Count
CAUOUTCH - Output Message Character Traffic
CAUCPUTM - Task CPU Time
CAUSQLTO - Total number of SQL Calls
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS
Average cost = CAUCOST/CAUTRANS
Average storage = CAUMEMRC/SUM(CAUTRANS, CAUETRN)

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

DAILY CICS COST RANKING REPORT											
your_company_name											
FOR: MARCH 15, 2007											
INQUIRY: CICLDR										RUN DATE: 16MAR07	
SYSTEM SUMMARY: 7 REGIONS											
SYSID	# TRANSACTIONS	COST	COST/ TRANS	AVG RESP	TOTAL CPU						
*ALL	644,548	\$435	\$0.00	0.32	0:49						
** TOP 10 CONSUMERS IN TOTAL COST											
RANK	CONSUMER	COST	TRANS	COST/ TRAN	AVG RESP	CPU	FILE CNTL CALLS	DBMS CALLS	MSGs (000)	CHAR (000)	AVG STORAGE
1	GADF	\$45,400	454	\$100.00	0.42	0:00	15	0	0	5	91K
2	GAD2	\$42,200	715	\$13.33	0.16	0:00	69	0	0	12	124K
3	AAON	\$9,600	471	\$200.00	71.69	0:00	19	0	0	0	216K
4	ACFM	\$8,780	863	\$14.72	0.02	0:00	0	0	0	162	70K
5	TCRI	\$8,600	2,102	\$2.66	0.07	0:00	5814	0	3	1682	20K
6	MSR1	\$5,600	2,359	\$2.37	0.04	0:00	3991	0	4	2345	22K
7	MTR2	\$4,800	424	\$116.67	0.04	0:00	12	0	0	14	21K
8	ACTT3	\$3,600	780	\$4.62	0.04	0:00	2829	0	1	1680	25K
9	PRT4	\$3,100	31	\$100.00	0.02	0:00	17	0	0	5	19K
10	AISI	\$4,400	952	\$4.62	0.16	0:00	641	0	1	830	140K

Figure 3-46. Daily CICS Cost Ranking Report Sample Output

3.1.5.9 CICLDW: Daily CICS Cost Summary Report

The Daily CICS Cost Summary Report shows the total processing cost and the top application in cost for the enterprise, for each system, and for each CICS region.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for the enterprise, and for each system (SYSID).

A sample output of this inquiry is shown in Figure 3-47. Each field in the report is described below:

SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
TRANS (000)	The total number of transactions (in thousands) processed, excluding type X as defined by your CICRLRT exit in prefix.MICS.PARMS.
COST (000):	The total processing charges in thousands of dollars.
COST/ TRAN:	The average cost per transaction.
AVG RESP:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.
CPU:	The total TCB CPU time consumed.
FC/DB CALLS (000):	The total number of file control calls and data base (DB2, IMS) calls issued. The number is in thousands.
MSGS (000):	The total number of input and output messages (in thousands) issued.
CHAR (000):	The total number of input and output characters (in thousands) issued.

TOP APPL: The application with the highest processing cost.

TOP TRANS: The total number of transactions processed for the top application.

TOP COST: The total processing cost of the top application.

TOP COST/TRAN: The average cost per transaction for the top application.

INQUIRY ID:

CICLDW (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICALPU.

DATA ELEMENTS USED:

CICALPU - CICS Application Unit ID
CICCSID - CICS System Identifier
CAUCPUTM - Task CPU Time
CAUCOST - Processing Charges
CAUDLICC - Number of CICS DL/I Calls
CAUFCOPS - Number of CICS File Control Calls
CAUINCH - Input Message Character Traffic
CAUMEMRC - Memory Usage Running Count
CAUOUTCH - Output Message Character Traffic
CAUCPUTM - Task CPU Time
CAUSQLTO - Total number of SQL Calls
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
DAY - Day of Month
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS

Average cost = CAUCOST/CAUTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICA_x data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input

file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

DAILY CICS COST SUMMARY REPORT													
your_company_name													
FOR: MARCH 15, 2007													
INQUIRY: CICLDW										RUN DATE: 16MAR07			
SYSID	CICSID	TRANS (000)	COST (000)	COST/ TRAN	AVG RESP	TOT CPU	FC/DB CL (000)	MSG5 (000)	CHAR (000)	----- ID	TOP CONSUMER	-----	-----
											TRANS	COST	COST/TRAN
*ALL	*ALL	645	\$435	\$0.68	0.32	0:49	295	1111	639546	CCMF	1315	\$11,600	\$8.82
PRD2	*ALL	645	\$435	\$0.68	0.32	0:49	295	1111	639546	CCMF	1315	\$11,600	\$8.82
	LCP1	470	\$64	\$0.14	0.12	0:18	0	788	465952	TOSC	344062	\$3,800	\$0.01
	RCP1	78	\$161	\$2.07	0.13	0:06	0	204	93486	EUBC	3753	\$3,000	\$0.80
	RCP2	69	\$146	\$2.13	0.34	0:13	251	68	47501	EUBC	3753	\$3,000	\$0.80
	RCP3	3	\$14	\$5.30	0.26	0:00	11	2	1291	DATA	768	\$2,000	\$2.60
	RCP4	24	\$15	\$0.61	0.61	0:12	29	47	30468	ISHD	23006	\$3,200	\$0.14
	TCT2	1	\$32	\$25.08	0.27	0:00	3	2	847	SIGN	78	\$2,000	\$25.64
	TCT7	0	\$3	\$15.17	563.10	0:00	0	0	0	CCMF	168	\$1,400	\$8.33
TOTAL # REGIONS: 7													

Figure 3-47. Daily CICS Cost Summary Report Sample Output

3.1.5.10 CICLDX: Daily CICS Cost Detail Report

The Daily CICS Cost Detail Report displays CICS processing charges at both the system and application level.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary section and a detail section for each combination of SYSID and CICSID. The summary section shows CICS processing charges for each system, as well as for the enterprise. The detail section shows the processing charges of each application, and provides total consumption on resources that are most often used for charging.

A sample output of this inquiry is shown in Figure 3-48. Each field in the report is described below:

SYSTEM SUMMARY:

# REGIONS:	Total number of SYSID/CICSID combinations found in input.
SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
TRANS (000)	The total number of transactions (in thousands) processed, excluding type X as defined by your CICRLRT exit in prefix.MICS.PARMS.
COST/ TRAN:	The average cost per transaction.
AVG RESP:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.
CPU:	The total TCB CPU time consumed.
FC/DB CALLS (000):	The total number of file control calls and data base (DB2, IMS) calls issued. The number is in thousands.
MSGS (000):	The total number of input and output messages (in thousands) issued.

CHAR (000):	The total number of input and output characters (in thousands) issued.
TOP APPL:	The application with the highest processing cost.
TOP TRANS:	The total number of transactions processed for the top application.
TOP COST:	The total processing cost of the top application.
TOP COST/TRAN:	The average cost per transaction for the top application.
DETAIL SECTION:	
APPL:	Application identification.
COST:	The total processing charges for the application.
COST/ TRAN:	Average cost per transaction for the application.
TRANS:	The total number of transactions processed for the application.
AVG RESP:	The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
AVERAGE CPU:	The average TCB CPU per transaction consumed by this application.
AVG FC CALLS:	The average number of file control calls issued per transaction.
AVG DBMS:	The average number of data base (DB2 and IMS) calls issued per transaction.
AVG MSGS:	The average number of input and output messages per transaction generated by the application.
AVG CHAR:	The average number of input and output characters per transaction generated by the application.
TOT RESP:	Total response elapsed time.
CPU:	The total TCB CPU time consumed.

TOT FILE: Total file control calls issued.

TOT DBMS: Total number of data base (DB2 and IMS) calls issued.

TOT MSG: The total number of input and output messages generated.

TOT CHAR: The total number of input and output characters generated.

INQUIRY ID:

CICLDX (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the DAYS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICALPU.

DATA ELEMENTS USED:

CAUABEND - Abnormal Termination Occurrences
CAUCPUTM - Task CPU Time
CAUCTRN - Conv. Transactions Processed
CAUDLICC - Number of CICS DL/I Calls
CAUETRAN - Excluded Transactions
CAUFCOPS - Number of CICS File Control Calls
CAUIMSGS - Total Input Messages Issued
CAUINCH - Input Message Character Traffic
CAULTRN - Long Transactions Processed
CAUMEMRC - Memory Usage Running Count
CAUMTRN - Medium Transactions Processed
CAUMXMEM - Maximum Transaction Storage Used
CAUOMSGS - Total Output Messages Issued
CAUOUTCH - Output Message Character Traffic
CAUSQLTO - Total number of SQL Calls
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
CICALPU - CICS Application Unit ID
CICSID - CICS System Identifier
CSUSTRN - Short Transactions Processed
DAY - Day of Month
HOUR - Hour of Day

MONTH	- Month of Year
SYSID	- System Identifier
YEAR	- Year of Century
ZONE	- Time Zone

CALCULATIONS:

```

Average response time = CAUTRSTM/CAUTRANS
Average cost = CAUCOST/CAUTRANS
Average character = SUM(CAUINCH,CAUOUTCH)/
                    SUM(CAUTRANS,CAUETRNL)
Average CPU time = CAUCPUTM/CAUTRANS
Average DBMS calls = SUM(CAUDLICC,CAUSQLT0,CICPCT01)/
                    SUM(CAUTRANS,CAUETRNL)
Average file control calls = CAUFCOPS/
                    SUM(CAUTRANS,CAUETRNL)
Average message = SUM(CAUIMSGS,CAUOMSGS)/
                    SUM(CAUTRANS,CAUETRNL)

```

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported.
If no value is specified, the value *ALL is
used which causes all CICSIDs found in input to
be summarized together.

DATE - Optionally, enter the date to be reported in
ddmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICA T_x data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

In addition to the above exits, the module CICRPTGL also contains macro variables which set default values for service level objectives. See Section 3.2 to override the default values.

DAILY CICS COST DETAIL REPORT														
your_company_name														
FOR: MARCH 15, 2007														
INQUIRY: CICLDX										RUN DATE: 16MAR07				
-----SYSID=PRD2 CICSID=LCP1-----														
----- TOTAL (000) -----														
CONSUMER	COST	COST/ TRAN	TRANS	RESP	CPU	FILE	DBMS	MSGS	CHAR	CPU	FILE	DBMS	MSGS	CHAR
AAON	\$200	\$200.00	1	34.10	0.08	1	0	0	0	0:00	0	0	0	0
ACFM	\$2,400	\$14.72	163	0.02	0.01	0	0	2	994	0:00	0	0	3	1621
CATD	\$1,900	\$0.12	15,759	0.07	0.01	0	0	0	0	0:02	0	0	0	0
CCMF	\$2,100	\$9.01	233	0.00	0.00	0	0	0	0	0:00	0	0	0	0
CEDA	\$100	\$4.76	21	0.20	0.01	4	0	2	639	0:00	1	0	0	134
CEMT	\$100	\$10.00	10	0.07	0.01	0	0	2	347	0:00	0	0	0	35
CRSQ	\$300	\$100.00	3	0.27	0.00	0	0	0	0	0:00	0	0	0	0
CRSR	\$1,800	\$0.06	29,214	0.02	0.00	0	0	0	0	0:01	0	0	0	0
CRTE	\$1,100	\$1.59	691	0.02	0.00	0	0	2	373	0:00	0	0	14	2577
CSAC	\$1,400	\$4.00	350	0.00	0.00	0	0	2	94	0:00	0	0	7	329
CSFU	\$100	\$100.00	1	0.83	0.01	0	0	0	0	0:00	0	0	0	0
CSGM	\$1,800	\$0.11	16,516	0.03	0.00	0	0	2	904	0:01	1	0	248	149314
CSIR	\$100	\$1.64	61	47.96	0.00	0	0	0	6	0:00	0	0	0	4
CSMI	\$1,000	\$90.91	11	0.01	0.00	0	0	1	396	0:00	0	0	0	44
CSNE	\$1,900	\$0.08	23,797	0.00	0.00	0	0	0	0	0:01	0	0	0	0
CSSF	\$1,400	\$36.84	38	0.01	0.00	0	0	2	268	0:00	0	0	1	102
CSSN	\$200	\$33.33	6	0.03	0.00	0	0	2	597	0:00	0	0	0	36
DATA	\$2,600	\$0.18	14,588	0.14	0.00	0	0	2	654	0:01	0	0	292	95441
DISP	\$200	\$200.00	1	0.03	0.02	0	0	5	314	0:00	0	0	0	3
D2DA	\$1,000	\$66.67	15	1.42	0.00	0	0	2	1425	0:00	0	0	0	214
ELON	\$1,000	\$100.00	10	0.24	0.01	0	0	2	784	0:00	0	0	0	78
EL00	\$1,400	\$35.00	40	0.06	0.00	0	0	2	875	0:00	0	0	1	350
EL10	\$400	\$44.44	9	0.04	0.00	0	0	2	631	0:00	0	0	0	57
EL20	\$1,200	\$29.27	41	0.15	0.00	0	0	2	1322	0:00	0	0	1	542
EL22	\$800	\$16.00	50	0.14	0.00	0	0	2	1150	0:00	0	0	1	575
EL29	\$800	\$38.10	21	0.11	0.00	0	0	2	1011	0:00	0	0	0	212
EL30	\$400	\$28.57	14	0.07	0.00	0	0	2	491	0:00	0	0	0	69
EZ	\$1,400	\$4.32	324	0.46	0.00	0	0	2	1602	0:00	0	0	6	5190
IEOL	\$1,400	\$1.04	1,346	0.25	0.00	0	0	2	852	0:00	0	0	27	11469
LLTP	\$200	\$200.00	1	0.01	0.00	0	0	3	136	0:00	0	0	0	1
LLXW	\$600	\$150.00	4	0.04	0.01	0	0	5	273	0:00	0	0	0	11
LPKD	\$600	\$200.00	3	0.00	0.00	0	0	2	376	0:00	0	0	0	11
LPKE	\$1,400	\$116.67	12	0.00	0.00	0	0	2	1460	0:00	0	0	0	175
MSAS	\$2,000	\$28.99	69	0.17	0.00	0	0	2	1007	0:00	0	0	1	695
MSA2	\$2,000	\$0.62	3,240	0.13	0.00	0	0	2	1621	0:00	1	0	65	52507
SA	\$800	\$20.51	39	0.48	0.00	1	0	2	927	0:00	0	0	1	362
STAR	\$2,400	\$114.29	21	0.00	0.00	0	0	2	1399	0:00	0	0	0	294
TOSC	\$3,800	\$0.01	344,062	0.13	0.00	0	0	2	1209	0:11	0	0	6898	4161320
TOSS	\$3,400	\$1.33	2,558	0.40	0.01	0	0	2	1539	0:00	0	0	51	39371
TOS1	\$3,600	\$0.42	8,580	0.38	0.01	0	0	2	1537	0:01	0	0	172	131904
TOTAL:	\$64,100	\$0.14	469,835	0.12	0.00	0	0	2	992	0:18	3	0	7877	4659520

Figure 3-48. Daily CICS Cost Detail Report Sample Output

3.1.5.11 CICLMR: Monthly CICS Cost Ranking Report

The Monthly CICS Cost Ranking Report identifies the top 10 consumers in CICS chargeable resource.

REPORT FORMAT:

This inquiry produces a tabular report which contains a summary section and a ranking section. The summary section shows the total cost for the enterprise and for each system. The ranking section identifies the 10 applications with the highest processing cost and provides total consumption of resources that are most often used for charging.

A sample output of this inquiry is shown in Figure 3-49. Each field in this report is described below:

SYSTEM SUMMARY:

- # REGIONS: Total number of SYSID/CICSID combinations found in input.
- SYSID: System identification. This field is set to ALL for data that is summarized for all systems.
- TRANS: The total number of transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
- COST (000): The total processing charges in thousands of dollars.
- COST/
TRAN: The average cost per transaction.
- AVG RESP: The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).
- TOTAL CPU: The total TCB CPU time consumed by all transactions, excluding transaction type X.

RANKING SECTION:

- RANK: The ranking, number 1 to 10, in processing cost.
- APPL: Application identification.

COST: The total processing charges for the application.

TRANS: The total number of transactions processed for the application.

COST/
TRAN: Average cost per transaction for the application.

AVG RESP: The average response time for all transactions processed for this application.

TOT CPU: The total TCB CPU time consumed by this application.

FC CALLS: The total number of file control calls issued by this application.

DBMS CALLS: The total number of data base calls (DB2 and IMS) issued by this application.

MSGs The total number of input and output messages generated by this application (in thousands).

CHAR (000): The total number of input and output characters generated by this application (in thousands).

AVG
STORAGE: The average amount (in 1024 bytes) of terminal and user storage used by this application.

INQUIRY ID:

CICLMR (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCUSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICALPU.

One graph produced for each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CICALPU - CICS Application Unit ID
CICSID - CICS System Identifier

CAUCOST - Processing Charges
CAUDLICC - Number of CICS DL/I Calls
CAUFCOPS - Number of CICS File Control Calls
CAUINCH - Input Message Character Traffic
CAUMEMRC - Memory Usage Running Count
CAUOUTCH - Output Message Character Traffic
CAUCPUTM - Task CPU Time
CAUSQLTO - Total number of SQL Calls
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
MONTH - Month of Year
YEAR - Year of Century
ZONE - Time Zone
SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS
Average cost = CAUCOST/CAUTRANS
Average storage = CAUMEMRC/SUM(CAUTRANS, CAUETRN)

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

DATE - Optionally, enter the date to be reported in ddmmyy format.

For this inquiry, the default is yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

MONTHLY CICS COST RANKING REPORT											
your_company_name											
FOR FEBRUARY, 2007											
INQUIRY: CICLMR									RUN DATE: 02MAR07		
SYSTEM SUMMARY: 7 REGIONS											
SYSID	# TRANSACTIONS	COST	COST/ TRANS	AVG RESP	TOTAL CPU						
*ALL	644,548	\$435	\$0.00	0.32	0:49						
** TOP 10 CONSUMERS IN TOTAL COST											
RANK	CONSUMER	COST	TRANS	COST/ TRAN	AVG RESP	CPU	FILE CNTL CALLS	DBMS CALLS	MSGS (000)	CHAR (000)	AVG STORAGE
1	GADF	\$45,400	454	\$100.00	0.42	0:00	15	0	0	5	91K
2	GAD2	\$42,200	715	\$13.33	0.16	0:00	69	0	0	12	124K
3	AAON	\$9,600	471	\$200.00	71.69	0:00	19	0	0	0	216K
4	ACFM	\$8,780	863	\$14.72	0.02	0:00	0	0	0	162	70K
5	TCRI	\$8,600	2,102	\$2.66	0.07	0:00	5814	0	3	1682	20K
6	MSR1	\$5,600	2,359	\$2.37	0.04	0:00	3991	0	4	2345	22K
7	MTR2	\$4,800	424	\$116.67	0.04	0:00	12	0	0	14	21K
8	ACTT3	\$3,600	780	\$4.62	0.04	0:00	2829	0	1	1680	25K
9	PRT4	\$3,100	31	\$100.00	0.02	0:00	17	0	0	5	19K
10	AISI	\$4,400	952	\$4.62	0.16	0:00	641	0	1	830	140K

Figure 3-49. Monthly CICS Cost Ranking Report Sample Output

3.1.5.12 CICLMW: Monthly CICS Cost Summary Report

The Monthly CICS Cost Summary Report shows the total processing cost and the top application in cost for the enterprise, for each system, and for each CICS region.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary line for each CICS region found in input. In addition, the report provides a summary line for the enterprise, and for each system (SYSID).

A sample output of this inquiry is shown in Figure 3-50. Each field in the report is described below:

SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
TRANS (000)	The total number of transactions (in thousands) processed, excluding type X as defined by your CICRLRT exit in prefix.MICS.PARMS.
COST (000):	The total processing charges in thousands of dollars.
COST/ TRAN:	The average cost per transaction.
AVG RESP:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.
CPU:	The total TCB CPU time consumed.
FC/DB CALLS (000):	The total number of file control calls and data base (DB2, IMS) calls issued. The number is in thousands.
MSGS (000):	The total number of input and output messages (in thousands) issued.
CHAR (000):	The total number of input and output characters (in thousands) issued.

TOP APPL: The application with the highest processing cost.

TOP TRANS: The total number of transactions processed for the top application.

TOP COST: The total processing cost of the top application.

TOP COST/TRAN: The average cost per transaction for the top application.

INQUIRY ID:

CICLMW (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICAUPU.

DATA ELEMENTS USED:

CICAUPU - CICS Application Unit ID
 CICCSID - CICS System Identifier
 CAUCPUTM - Task CPU Time
 CAUCOST - Processing Charges
 CAUDLICC - Number of CICS DL/I Calls
 CAUFCOPS - Number of CICS File Control Calls
 CAUINCH - Input Message Character Traffic
 CAUMEMRC - Memory Usage Running Count
 CAUOUTCH - Output Message Character Traffic
 CAUCPUTM - Task CPU Time
 CAUSQLTO - Total number of SQL Calls
 CAUTRANS - Transactions Processed
 CAUTRSTM - Transaction Response Time Total
 MONTH - Month of Year
 YEAR - Year of Century
 ZONE - Time Zone
 SYSID - System Identifier

CALCULATIONS:

Average response time = CAUTRSTM/CAUTRANS
 Average cost = CAUCOST/CAUTRANS

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value *ALL is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCSU file as input when the CICCAU file is active. If you specify this parameter, then you must also specify the next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICTACTx data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

MONTHLY CICS COST SUMMARY REPORT													
your_company_name													
FOR FEBRUARY, 2007													
INQUIRY: CICLMW										RUN DATE: 02MAR07			
SYSID	CICSID	TRANS (000)	COST (000)	COST/ TRAN	AVG RESP	TOT CPU	FC/DB CL (000)	MSG5 (000)	CHAR (000)	----- ID	TOP CONSUMER	-----	-----
*ALL	*ALL	645	\$435	\$0.68	0.32	0:49	295	1111	639546	CCMF	1315	\$11,600	\$8.82
PRD2	*ALL	645	\$435	\$0.68	0.32	0:49	295	1111	639546	CCMF	1315	\$11,600	\$8.82
	LCP1	470	\$64	\$0.14	0.12	0:18	0	788	465952	TOSC	344062	\$3,800	\$0.01
	WCP1	78	\$161	\$2.07	0.13	0:06	0	204	93486	EUBC	3753	\$3,000	\$0.80
	WCP2	69	\$146	\$2.13	0.34	0:13	251	68	47501	EUBC	3753	\$3,000	\$0.80
	WCP3	3	\$14	\$5.30	0.26	0:00	11	2	1291	DATA	768	\$2,000	\$2.60
	WCP4	24	\$15	\$0.61	0.61	0:12	29	47	30468	ISHD	23006	\$3,200	\$0.14
	WCT2	1	\$32	\$25.08	0.27	0:00	3	2	847	SIGN	78	\$2,000	\$25.64
	WCT7	0	\$3	\$15.17	563.10	0:00	0	0	0	CCMF	168	\$1,400	\$8.33
TOTAL # REGIONS: 7													

Figure 3-50. Monthly CICS Cost Summary Report Sample Output

3.1.5.13 CICLMX: Monthly CICS Cost Detail Report

The Monthly CICS Cost Detail Report displays CICS processing charges at both the system and application level.

REPORT FORMAT:

This inquiry produces a tabular report which provides a summary section and a detail section for each combination of SYSID and CICSID. The summary section shows CICS processing charges for each system, as well as for the enterprise. The detail section shows the processing charges of each application, and provides total consumption on resources that are most often used for charging.

A sample output of this inquiry is shown in Figure 3-51. Each field in the report is described below:

SYSTEM SUMMARY:

# REGIONS:	Total number of SYSID/CICSID combinations found in input.
SYSID:	System identification. This field is set to ALL for data that is summarized for all systems.
CICSID:	CICS system identification as assigned by CA MICS. This field is set to ALL if the data is summarized for all CICS regions.
TRANS (000)	The total number of transactions (in thousands) processed, excluding type X as defined by your CICRLRT exit in prefix.MICS.PARMS.
COST/ TRAN:	The average cost per transaction.
AVG RESP:	The average response time for all transactions, excluding those assigned as TRANTYPE X in your CICRLRT exit in prefix.MICS.PARMS.
CPU:	The total TCB CPU time consumed.
MSG (000):	The total number of input and output messages (in thousands) issued.
CHAR (000):	The total number of input and output characters (in thousands) issued.

TOP APPL: The application with the highest processing cost.

TOP TRANS: The total number of transactions processed for the top application.

TOP COST: The total processing cost of the top application.

TOP COST/TRAN: The average cost per transaction for the top application.

DETAIL SECTION:

APPL: Application identification.

COST: The total processing charges for the application.

COST/TRAN: Average cost per transaction for the application.

TRANS: The total number of transactions processed for the application.

AVG RESP: The average response time for the transactions processed. For TYPE=ALL, this number excludes transaction type X (TRANSTYPE=X as assigned by the CICRLRT exit).

AVERAGE CPU: The average TCB CPU per transaction consumed by this application.

AVG FC CALLS: The average number of file control calls issued per transaction.

AVG DBMS: The average number of data base (DB2 and IMS) calls issued per transaction.

AVG MSGS: The average number of input and output messages per transaction generated by the application.

AVG CHAR: The average number of input and output characters per transaction generated by the application.

TOT RESP: Total response elapsed time.

CPU: The total TCB CPU time consumed.

TOT FILE: Total file control calls issued.

TOT DBMS: Total number of data base (DB2 and IMS) calls

issued.

TOT MSG: The total number of input and output messages generated.

TOT CHAR: The total number of input and output characters generated.

INQUIRY ID:

CICLMX (tabular report)

DATA SOURCE (file/timespan):

CICCAU01 in the MONTHS timespan.

If the CICCAU (CICS Application Unit Activity) file is not active, the CICCSU (CICS User Activity) file will be used for reporting. The variable CICTACT2 will be used in place of the variable CICALPU.

One graph produced for each combination of SYSID/CICSID.

DATA ELEMENTS USED:

CAUABEND - Abnormal Termination Occurrences
CAUCPUTM - Task CPU Time
CAUCTRN - Conv. Transactions Processed
CAUDLICC - Number of CICS DL/I Calls
CAUETRNL - Excluded Transactions
CAUFCOPS - Number of CICS File Control Calls
CAUIMSGS - Total Input Messages Issued
CAUINCH - Input Message Character Traffic
CAULTRNL - Long Transactions Processed
CAUMEMRC - Memory Usage Running Count
CAUMTRNL - Medium Transactions Processed
CAUMXMEM - Maximum Transaction Storage Used
CAUOMSGS - Total Output Messages Issued
CAUOUTCH - Output Message Character Traffic
CAUSQLTO - Total number of SQL Calls
CAUTRANS - Transactions Processed
CAUTRSTM - Transaction Response Time Total
CICALPU - CICS Application Unit ID
CICSID - CICS System Identifier
CSUSTRNL - Short Transactions Processed
DAY - Day of Month
HOUR - Hour of Day
MONTH - Month of Year
SYSID - System Identifier

YEAR - Year of Century
ZONE - Time Zone

CALCULATIONS:

```

Average response time = CAUTRSTM/CAUTRANS
Average cost = CAUCOST/CAUTRANS
Average character = SUM(CAUINCH,CAUOUTCH)/
                  SUM(CAUTRANS,CAUETRN)
Average CPU time = CAUCPUTM/CAUTRANS
Average DBMS calls = SUM(CAUDLIC,CAUSQLT,CICPCT01)/
                  SUM(CAUTRANS,CAUETRN)
Average file control calls = CAUFCOPS/
                  SUM(CAUTRANS,CAUETRN)
Average message = SUM(CAUIMSGS,CAUOMSGS)/
                  SUM(CAUTRANS,CAUETRN)

```

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the value ***ALL** is used. It causes all SYSIDs found in input to be summarized together.

CICSID - Optionally, specify the CICSID to be reported. If no value is specified, the value *ALL is used which causes all CICSIDs found in input to be summarized together.

MONTH - Optionally, enter the month (mm) and year (yy) to be reported.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

FILE ID- Optionally, select the CICCUS file as input when the CICCAS file is active. If you specify this parameter, then you must also specify the

next parameter to indicate which account code (CICACT1-CICACT9) identifies your CICS application.

CICAPU - Optionally, replace CICAPU with any of the CICA_x data elements, where x is 1 to 9. This parameter must be specified in conjunction with FILE ID.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available to this inquiry:

%GAPPL - Define your CICS application based on any variable in the input file. The default is CICAPU.

%GCIC - Select CICSIDs for report.

%GCNTR - Assign value to the variable CENTER which defines the logical grouping of SYSIDs. CENTER is set to SYSID by default.

%GSYS - Select SYSIDs for report.

MONTHLY CICS COST DETAIL REPORT														
your_company_name														
FOR FEBRUARY, 2007														
INQUIRY: CICLMX										RUN DATE: 02MAR07				
-----SYSID=PRD2 CICSID=LCP1-----														
CONSUMER	COST	COST/ TRAN	TRANS	RESP	CPU	FILE	DBMS	MSGS	CHAR	CPU	FILE	DBMS	MSGS	CHAR
AAON	\$200	\$200.00	1	34.10	0.08	1	0	0	0	0:00	0	0	0	0
ACFM	\$2,400	\$14.72	163	0.02	0.01	0	0	2	994	0:00	0	0	3	1621
CATD	\$1,900	\$0.12	15,759	0.07	0.01	0	0	0	0	0:02	0	0	0	0
CCMF	\$2,100	\$9.01	233	0.00	0.00	0	0	0	0	0:00	0	0	0	0
CEDA	\$100	\$4.76	21	0.20	0.01	4	0	2	639	0:00	1	0	0	134
CEMT	\$100	\$10.00	10	0.07	0.01	0	0	2	347	0:00	0	0	0	35
CRSQ	\$300	\$100.00	3	0.27	0.00	0	0	0	0	0:00	0	0	0	0
CRSR	\$1,800	\$0.06	29,214	0.02	0.00	0	0	0	0	0:01	0	0	0	0
CRTE	\$1,100	\$1.59	691	0.02	0.00	0	0	2	373	0:00	0	0	14	2577
CSAC	\$1,400	\$4.00	350	0.00	0.00	0	0	2	94	0:00	0	0	7	329
CSFU	\$100	\$100.00	1	0.83	0.01	0	0	0	0	0:00	0	0	0	0
CSGM	\$1,800	\$0.11	16,516	0.03	0.00	0	0	2	904	0:01	1	0	248	149314
CSIR	\$100	\$1.64	61	47.96	0.00	0	0	0	6	0:00	0	0	0	4
CSMI	\$1,000	\$90.91	11	0.01	0.00	0	0	1	396	0:00	0	0	0	44
CSNE	\$1,900	\$0.08	23,797	0.00	0.00	0	0	0	0	0:01	0	0	0	0
CSSF	\$1,400	\$36.84	38	0.01	0.00	0	0	2	268	0:00	0	0	1	102
CSSN	\$200	\$33.33	6	0.03	0.00	0	0	2	597	0:00	0	0	0	36
DATA	\$2,600	\$0.18	14,588	0.14	0.00	0	0	2	654	0:01	0	0	292	95441
DISP	\$200	\$200.00	1	0.03	0.02	0	0	5	314	0:00	0	0	0	3
D2DA	\$1,000	\$66.67	15	1.42	0.00	0	0	2	1425	0:00	0	0	0	214
EL0N	\$1,000	\$100.00	10	0.24	0.01	0	0	2	784	0:00	0	0	0	78
EL00	\$1,400	\$35.00	40	0.06	0.00	0	0	2	875	0:00	0	0	1	350
EL10	\$400	\$44.44	9	0.04	0.00	0	0	2	631	0:00	0	0	0	57
EL20	\$1,200	\$29.27	41	0.15	0.00	0	0	2	1322	0:00	0	0	1	542
EL22	\$800	\$16.00	50	0.14	0.00	0	0	2	1150	0:00	0	0	1	575
EL29	\$800	\$38.10	21	0.11	0.00	0	0	2	1011	0:00	0	0	0	212
EL30	\$400	\$28.57	14	0.07	0.00	0	0	2	491	0:00	0	0	0	69
EZ	\$1,400	\$4.32	324	0.46	0.00	0	0	2	1602	0:00	0	0	6	5190
IEOL	\$1,400	\$1.04	1,346	0.25	0.00	0	0	2	852	0:00	0	0	27	11469
LLTP	\$200	\$200.00	1	0.01	0.00	0	0	3	136	0:00	0	0	0	1
LLXW	\$600	\$150.00	4	0.04	0.01	0	0	5	273	0:00	0	0	0	11
LPKD	\$600	\$200.00	3	0.00	0.00	0	0	2	376	0:00	0	0	0	11
LPKE	\$1,400	\$116.67	12	0.00	0.00	0	0	2	1460	0:00	0	0	0	175
MSAS	\$2,000	\$28.99	69	0.17	0.00	0	0	2	1007	0:00	0	0	1	695
MSA2	\$2,000	\$0.62	3,240	0.13	0.00	0	0	2	1621	0:00	1	0	65	52507
SA	\$800	\$20.51	39	0.48	0.00	1	0	2	927	0:00	0	0	1	362
STAR	\$2,400	\$114.29	21	0.00	0.00	0	0	2	1399	0:00	0	0	0	294
TOSC	\$3,800	\$0.01	344,062	0.13	0.00	0	0	2	1209	0:11	0	0	6898	4161320
TOSS	\$3,400	\$1.33	2,558	0.40	0.01	0	0	2	1539	0:00	0	0	51	39371
TOS1	\$3,600	\$0.42	8,580	0.38	0.01	0	0	2	1537	0:01	0	0	172	131904
TOTAL :	\$64,100	\$0.14	469,835	0.12	0.00	0	0	2	992	0:18	3	0	7877	4659520

Figure 3-51. Monthly CICS Cost Detail Report Sample Output

3.1.6 General Analysis Inquiries

The Daily CICS System Overhead Analysis identifies the percentage of CPU time spent in system overhead.

REPORT FORMAT:

This inquiry produces a pie chart showing the percentage of total CPU time consumed by system overhead, including task control, terminal control, journal control, and SRB time.

A sample output of this inquiry is shown in Figure 3-52.

INQUIRY ID:

CICLDA (printer graphic format)

DATA SOURCE (file/timespan):

CICCSY01 at the DETAIL timespan

DATA ELEMENTS USED:

CSYCPJTM - JCA CPU TCB Time
CSYCPUTM - CPU Time Consumed
CSYSRBTM - SRB CPU Time
CSYTCPTM - Terminal Control TCB CPU Time
CSYTSRTM - Task Control CPUR Time

CALCULATIONS:

Application CPU = CSYCPUTM - SUM(CSYSRBTM,CSYCPJTM,
CSYTCPTM,CSYTSRTM)

Chapter 3: REPORTS 371

3.1.7 CICS Transaction Gateway Inquiries

CICS Transaction Gateway (CTG) inquiries provide information to help you understand the performance and throughput of the CTGs running on your z/OS systems. The following inquiries are provided for CTG analysis:

MICF Distributed Inquiries for CICS Transaction Gateway Analysis						
Report Class	Report Name	MICF Catalog Group	MICF Color Graphics	Inquiry Identifier Printer Graphics Tabular List		
System Performance	CICS Transaction Gateway Daily Throughput	PERFORM				CICLDG

Figure 3-53. Distributed CICS Transaction Gateway Inquiries

To provide report flexibility, the CA MICS Analyzer Option for CICS supplies MICF execution run-time parameters for the CTG inquiries. You can use these parameters to tailor the inquiries to suit your reporting needs.

The CTG inquiries are described individually in the following sections:

- 1 - CICLDG: CICS Transaction Gateway Daily Throughput

3.1.7.1 CICLDG: CICS Transaction Gateway Daily Throughput

The CA MICS CICS Transaction Gateway Daily Throughput inquiry, CICLDG, provides both summary and detailed reports of CICS Transaction Gateway (CTG) throughput.

The summary report is generated first. It shows the overall daily request throughput, request data volume, processing throughput rate, and average transaction response time for each CTG. The summary data is organized by z/OS SYSID so that overall throughput can be easily seen for each individual CTG by z/OS system.

The detail report follows and generates a report page for each CTG. The same data seen in the summary report is displayed for the CTG at the hourly level.

Both the summary and detail reports show throughput metrics in two groups:

- o The first group of metrics provides information about Gateway daemon throughput and response time.
- o The second group of metrics provides information about CICS region throughput and response time for CTG requests.

REPORT FORMAT:

The format of the summary and detail report sections are described below.

SUMMARY REPORT

The summary report header displays the report inquiry name, run date, report title, summarization level, and the date being reported. If a particular ZONE was requested for the report, the ZONE value and ZONE descriptive name are displayed.

The summary report body displays the following measurements from left to right:

o System Identification	- SYSID
o CICS Transaction Gateway ID	- CTGID

Gateway daemon statistics are presented next, beginning with the number of client transactions processed, megabytes of data for both requests and responses, the number of transactions processed per second, and the average transaction response time:

o Client Requests Processed	- GSAGREQP
o Client Request MB Data	- GSACLRQD
o Client Response MB Data	- GSACLRSD
o Client Requests per Second	- GSAPSGTW
o Average Client Request Response Time	- GSAAVGTM

CICS region statistics are presented next, beginning with the number of CICS transactions processed, megabytes of data for both requests and responses, the number of transactions processed per second, and the average request transaction response time:

o CICS Requests Processed	- GSACREQP
o CICS Request MB Data	- GSACIRQD
o CICS Response MB Data	- GSACIRSD
o CICS Requests per Second	- GSAPSCIC
o Average CICS Request Response Time	- GSAAVCTM

If multiple CTGIDs are found for a single SYSID, a total line will appear that provides z/OS system level totals for all CTGIDs on the system.

Note: If the CTGIDs for a system continue onto a second page, the SYSID value will have an asterisk (*) next to it to indicate the continuation.

DETAIL REPORT

The summary report displayed one line for each CTG, showing a days worth of throughput information. The detail report presents a page of data for each CTG, with the same statistics reported at the hourly level.

The detail report header displays the report inquiry name, run date, report title, summarization level, and the date being reported. If a particular ZONE was requested for the report, the ZONE value and ZONE descriptive name are displayed.

Next, the detail report header provides additional information about the CTG being reported:

o System Identification	- SYSID
o CICS Transaction Gateway ID	- CTGID
o CICS Transaction Gateway APPLID	- CTGAPPL
o CTG Release Level	- CTGRLVL

The detail report body displays the following measurements from left to right:

- o Day of week - from ENDTS
- o Hour of day - HOUR

Gateway daemon statistics are presented next, beginning with the number of client transactions processed, megabytes of data for both requests and responses, the number of transactions processed per second, and the average transaction response time:

- o Client Requests Processed - GSAGREQP
- o Client Request MB Data - GSACLRQD
- o Client Response MB Data - GSACLRSD
- o Client Requests per Second - GSAPSGTW
- o Average Client Request Response Time - GSAAVGTM

Next, CICS region statistics are presented beginning with the number of CICS transactions processed, megabytes of data for both requests and responses, the number of transactions processed per second, and the average request transaction response time:

- o CICS Requests Processed - GSACREQP
- o CICS Request MB Data - GSACIRQD
- o CICS Response MB Data - GSACIRSD
- o CICS Requests per Second - GSAPSCIC
- o Average CICS Request Response Time - GSAAVCTM

At the bottom of the detail report, a line is generated that shows summary totals, minimums, maximums, averages, or percentages, as appropriate for the hourly data in each column.

INQUIRY ID:

CICLDG (printer graphic format)

DATA SOURCE (file/timespan):

CTGGSA at the DAYS timespan

EXECUTION-TIME PARAMETERS:

UNIT - Select one or more CA MICS data base units by entering the data base IDs. Enter multiple IDs with a blank following each ID value.

CYCLE - Optionally, define CA MICS input file cycle range.

For this inquiry, the default is the 01 cycle.

SYSID - Optionally, specify the SYSID to be reported. If no value is specified, the report will include data from all SYSIDs in the input file.

CTGID - Optionally, specify the CTGID to be reported. If no value is specified, all Transaction Gateways in the input file will be included in the report.

DATE - Optionally, enter the date to be reported in dmmmyy format.

For this inquiry, the default is the data contained in the DAYS timespan 01 cycle, which is generally data from yesterday.

ZONE - Optionally, select one or more CA MICS time zones. Valid values are numbers 1 through 9.

USER EXITS:

If the MICF execution-time parameters are insufficient to select the data you need, you can invoke one of the following global exits to further manipulate the input file. Additionally, you can override the global exit definitions in MICF using independent source statements. For detailed information on coding the exits, refer to Section 3.2 of this guide.

The global exits are defined by macros in the source module CICRPTGL in sharedprefix.MICS.SOURCE. The following exits are available for this inquiry:

%GCTG - Select CTGIDs for report.

%GSYS - Select SYSIDs for report.

A sample output of this inquiry, where data for ZONE=1 was selected, is shown in Figure 3-54.

Inquiry : CICLDG		CA MICS - CICS Transaction Gateway Daily Throughput Summary										Page 001
Run Date: 27JANyy		Data Summarized by CTGID and DAY										Report Start: 26JANyy:07:59
Reporting on ZONE 1 - PRIME TIME		REPORT PAGE: Data for Date=26JANyy										Report End : 26JANyy:16:59

		+----- Gateway Daemon Activity -----+					+----- CICS Server Activity -----+					
		Client Requests Processed	Client Request MB Data	Client Resp MB Data	Client Request per/sec	Client Avg Resp Time	CICS Requests Processed	CICS Request MB Data	CICS Resp MB Data	CICS Request per/sec	CICS Avg Resp Time	
SYSID	CTGID											

SYS1	GTP0	119794	2875.80	2861.28	3.697	0.1268	119794	2862.53	2862.24	3.697	0.1227	
	GTP1	146823	3484.11	3466.19	4.532	0.1295	146823	3467.86	3467.25	4.532	0.1257	
	GTP2	91381	940.79	876.79	2.820	0.1120	91381	930.65	930.57	2.820	0.1101	
	GTP3	107977	1207.31	1134.22	3.333	0.0980	107977	1195.35	1195.31	3.333	0.0959	
	GTP5	1296	9.67	9.56	0.040	0.2282	1296	9.48	9.48	0.040	0.2184	
	GTP6	2625	18.56	18.35	0.081	0.2279	2625	18.19	18.19	0.081	0.2204	

Total	*ALL*	469896	8536.24	8366.39	14.503	0.1190	469896	8484.06	8483.04	14.503	0.1158	

SYS2	GTS0	204	2.49	2.23	0.007	1.6213	204	2.46	1.00	0.007	1.5368	
	GTU1	64673	988.84	972.44	1.996	0.1765	64673	981.42	954.33	1.996	0.1750	

Total	*ALL*	64877	991.33	974.67	2.003	0.1811	64877	983.88	955.33	2.003	0.1793	

SYS3	GTP0	167836	3808.77	3786.12	5.180	0.1283	167836	3790.20	3789.73	5.180	0.1246	
	GTP1	166221	3951.78	3932.20	5.130	0.1336	166221	3933.38	3932.92	5.130	0.1300	
	GTP2	105042	1252.70	1178.66	3.242	0.1339	105042	1241.05	1241.02	3.242	0.1318	
	GTP3	103851	1229.55	1146.53	3.205	0.1153	103851	1218.03	1217.98	3.205	0.1134	
	GTP5	1217	4.35	4.24	0.038	0.3950	1217	4.16	4.16	0.038	0.3939	
	GTP6	2235	20.91	20.73	0.069	0.1226	2235	20.60	20.60	0.069	0.1202	

Total	*ALL*	546402	10268.1	10068.5	16.864	0.1291	546402	10207.4	10206.4	16.864	0.1261	

SYS4	TGA0	35159	481.53	430.64	1.116	0.0741	35159	476.99	476.84	1.116	0.0731	
	TGE0	1	0.00	0.00	0.000	6.1920	1	0.00	0.00	0.000	5.6440	
	TGT0	10	0.07	0.07	0.000	1.1803	10	0.07	0.07	0.000	1.0568	

Total	*ALL*	35170	481.60	430.71	1.117	0.0746	35170	477.06	476.91	1.117	0.0735	

3.1 MICF Inquiries

Inquiry : CICLDG		CA MICS - CICS Transaction Gateway Daily Throughput Detail								Page 001	
Run Date: 27JANyy		Data Presented by HOUR for each CTGID								Report Start: 26JANyy:07:59	
Reporting on ZONE 1 - PRIME TIME		REPORT PAGE: Data for Date=26JANyy								Report End : 26JANyy:16:59	
SYSID : SYS1											
Gateway ID: GTP0		APPLID: CICSTGP0		Release Level: 7.1.0							

+----- Gateway Daemon Activity -----+ +----- CICS Server Activity -----+											
Day of Week	HOUR	Client	Client	Client	Client	Client	CICS	CICS	CICS	CICS	CICS
		Requests	Request	Resp	Request	Avg Resp	Requests	Request	Resp	Request	Avg Resp
		Processed	MB Data	MB Data	per/sec	Time	Processed	MB Data	MB Data	per/sec	Time

Tue	8	5499	131.08	130.28	1.527	0.1496	5499	130.47	130.47	1.527	0.1458
	9	9817	234.38	233.07	2.727	0.1590	9817	233.29	233.00	2.727	0.1551
	10	9739	238.35	237.33	2.705	0.1303	9739	237.27	237.27	2.705	0.1266
	11	19523	480.52	478.18	5.423	0.1231	19523	478.33	478.33	5.423	0.1191
	12	22538	550.99	548.51	6.261	0.1139	22538	548.50	548.50	6.261	0.1099
	13	12074	275.78	274.12	3.354	0.1326	12074	274.47	274.47	3.354	0.1290
	14	10721	244.90	243.47	2.978	0.1215	10721	243.71	243.71	2.978	0.1176
	15	16111	378.98	376.99	4.475	0.1384	16111	377.20	377.20	4.475	0.1339
	16	13772	340.81	339.32	3.826	0.1037	13772	339.28	339.28	3.826	0.0989
TOTALS		119794	2875.80	2861.28	3.697	0.1268	119794	2862.53	2862.24	3.697	0.1227

Figure 3-54. CICS Transaction Gateway Daily Throughput Report

3.2 Changing Report Parameters

The CICS Analyzer provides three methods with which you can tailor the input to meet your reporting needs:

- o MICF execution-time parameters

MICF execution-time parameters provide the means to specify input selection criteria which typically include parameters such as date, file cycle, ZONE, CICSID, and SYSID.

- o Report global exits

Global exits, which are invoked by specific inquiries as documented in this guide, enable you to specify global input selection criteria and to manipulate report elements such as application IDs and system IDs.

- o Global exit overrides

Global exit overrides, which are invoked after the global exits, provide additional flexibility in input data selection and are specified through MICF independent source statements. Global exits and their overrides are more flexible than the execution parameters since you can supply SAS statements for them to manipulate any data element in the input file. However, execution-time parameters take precedence over the global exits and global exit overrides since they allow you to change input selection criteria for each execution of the inquiry.

Detailed information on global exits are provided below.

GLOBAL EXITS

Global exits for CICS Analyzer reports are defined in `sharedprefix.MICS.SOURCE(CICRPTGL)`. Most of the exits are defined as dummy SAS macros; however, some are shipped with default values. The exits apply to all MICF inquiries in which they are invoked. You can override most of the exits using either independent SAS statements or execution time parameters. The following exits, in alphabetical order, are provided in the `CICRPTGL` module.

`%GAPPL` - This exit identifies your CICS application. The default value is `CICAPU` which is saved in the report variable `APPL`. You can override the default if you

wish to use a data element other than CICAPU to identify your CICS application, or to manipulate CICAPU or other data elements in the input file to assign APPL. When you code the exit, you must assign a value to APPL, the application field used in CICS Analyzer reports, and APPLLBL, the report heading for the APPL field. The CA MICS data elements that are available during this exit include all fields in the input file, which by default is the CICCAU (CICS Application Unit Activity) file.

Default exit code:

```
%MACRO GAPPL;
  %* SET APPLICATION VALUE;
  %GLOBAL APPLLBL;
  %LET APPLLBL=APPLICATION;
  APPL=CICAPU;
%MEND GAPPL;
```

Sample exit code to use CICTAx for application reporting:

```
%MACRO GAPPL;
  %* SET APPLICATION VALUE USING CICTACT1 FROM CICCSTU
    FILE;
  %GLOBAL APPLLBL;
  %LET APPLLBL=APPLICATION;
  APPL=CICTACT1;
%MEND GAPPL;
```

GLOBAL EXIT OVERRIDE

To override the global exits defined in sharedprefix.MICS.SOURCE(CICRPTGL) for individual inquiries, you must redefine the exit macros using independent SAS statements within the inquiry. This requires that you modify the inquiry after it has been copied from the MICF shared catalog to your private catalog. Once you have modified and tested the changes in your private catalog, the inquiry can then be moved to the shared catalog either as a new inquiry or replacement for the existing inquiry.

The global exits are currently invoked via %INCLUDE statement in the Independent Source Statement step of the MICF inquiry. To override any of the exits, you must redefine the exit macros after the %INCLUDE statement. For example, to override the GAPPL exit to use data element CICTACT1 for

application reporting, you can code the following statements after the %INCLUDE statement for module CICRPTGL:

```
%INCLUDE SOURCE(CICRPTGL,$CICRMAC);
***** DEFINE GLOBAL EXIT OVERRIDE *****;
%MACRO GAPPL;
%* SET APPLICATION VALUE USING CICTACT1 FROM CICCUSU
  FILE;
%GLOBAL APPLLBL;
%LET APPLLBL=APPLICATION;
  APPL=CICTACT1;
%MEND GAPPL;
```

3.3 Running Reports in Batch

You can execute any of the CICS Analyzer reports in batch by either submitting the inquiry as a batch job through MICF or using the MICF production report interface, which enables automatic report generation upon completion of CA MICS production jobs, such as DAILY, WEEKLY, and MONTHLY. In addition, some of the CICS Analyzer inquiries, which were previously shipped as batch reports, can be produced using your own JCL external to MICF. The following sections discuss the MICF production interface and the JCL requirement for these selected inquiries.

- 1 - Production Report Interface Using MICF
- 2 - JCL Requirements

3.3.1 Production Report Interface Using MICF

MICF provides a facility which automatically generates CICS Analyzer reports upon completion of CA MICS production jobs, such as DAILY, WEEKLY, and MONTHLY. It enables you to select shared inquiries for batch execution and store reports and color graphics in a "production catalog" for later online review and printing.

3.3.2 JCL Requirements

For selected CICS Analyzer inquiries, which were first shipped as batch summary reports, you can execute them using your own JCL external to MICF. The inquiries that can be executed in this manner include:

- o CICLD4 - Daily Activity Overview Report
- o CICLD5 - Application Limit Summary Report
- o CICLD6 - Daily User Activity Summary Report
- o CICLD7 - FILE/DBD Activity Summary Report
- o CICLD8 - DB2 Call Summary Report

To code your own JCL, simply create a member in a PDS that contains the following:

```
//jobname JOB ...  
//S1      EXEC MICSSHri  
//MICS.SYSIN DD DSN=sharedprefix.MICS.SOURCE(CICRPTnn),  
//          DISP=SHR
```

The MICSSHri procedure is used to execute the CICS summary reports, where i is the single-character identifier for the CA MICS unit data base that contains the CIC Information Area files. A SYSIN DD statement is required to select the requested CICS report program source member, which are named CICRPTnn, where nn is 01 through 05.

The reports and their corresponding program source members are:

- o CICRPT01 - Daily Activity Overview Report
- o CICRPT02 - Application Limit Summary Report
- o CICRPT03 - Daily User Activity Summary Report
- o CICRPT04 - FILE/DBD Activity Summary Report
- o CICRPT05 - DB2 Call Summary Report - CICRPT05

Once you have coded the JCL, you can submit it upon completion of the CA MICS DAILY job to produce the reports. However, since the reports are available through MICF, the alternative and the recommended method to generate them in batch is through MICF production interface, which automatically creates the reports after the CA MICS DAILY, WEEKLY, or MONTHLY job has completed. See Section 3.3.1 for more information on MICF production interface.

Chapter 4: EXCEPTIONS

The CA MICS CICS Analyzer supports the standard CA MICS exception process. CA MICS exception processing enables the I/S organization to focus on problems impacting its effectiveness in terms of availability, service, workload, standards, security, and performance for its different areas of responsibility (like CICS, TSO, IMS, and VSE/POWER).

The standard exceptions for the CA MICS CICS Analyzer are shown in Figure 4-1. Each standard exception test is shipped with default values. However, to make effective use of the exception process, you must evaluate your data center's needs and modify these sample values accordingly.

Number	Severity	Management Area	Exception Description
04001	Warning	Service	CICS Application Exceeds Avg Elapsed Time Standard
04002	Warning	Service	CICS Application Exceeds Avg TCB/SRB CPU Limit
04004	Warning	Performance	CICS Application Exceeds Avg Wait Time Limit
04005	Warning	Service	CICS Application Exceeds Residency Time Limit
04006	Warning	Workload	CICS Application Exceeds Max Memory Limit
04007	Critical	Performance	CICS Short-On-Storage Transaction Delay
04008	Critical	Performance	CICS Maxtask Transaction Delay
04009	Critical	Security	CICS SIGNON Failure-Probable Security Violation
04010	Critical	Performance	CICS Storage Violation Detected
04011	Critical	Performance	CICS Abnormal Condition Transaction
04012	Impacting	Performance	CICS Transaction Abnormal Termination
04013	Critical	Security	CICS Dynamic Transaction Backout Failure
04014	Warning	Performance	CICS Active Max Task
04015	Critical	Performance	CICS Short on Storage
04016	Warning	Performance	CICS VSAM String Wait
04017	Warning	Performance	CICS VSAM Buffer Wait
04018	Warning	Performance	CICS DL/I Thread Wait
04019	Impacting	Performance	CICS Wait for Storage in DSA
04020	Impacting	Performance	CICS Wait for Storage in EDSA
04021	Impacting	Performance	CICS Wait for Temporary Storage (Main)
04022	Impacting	Performance	CICS Wait for File String
04023	Impacting	Performance	CICS Wait for File Buffer
04024	Impacting	Performance	CICS LSR Pool Buffer Wait
04025	Impacting	Performance	CICS Program Compression
04026	Impacting	Performance	CICS Storage Control Suspend
04027	Impacting	Performance	CICS Temporary Storage Put Wait
04028	Impacting	Performance	CICS Main Storage Wait
04030	Impacting	Performance	CICS Wait for Storage in UDSA
04031	Impacting	Performance	CICS Wait for Storage in CDSA
04032	Impacting	Performance	CICS Wait for Storage in EUDSA
04033	Impacting	Performance	CICS Wait for Storage in ECDSA
04034	Impacting	Performance	CICS Wait for Storage in ERDSA
04035	Impacting	Performance	CICS Wait for Storage in SD SA
04036	Impacting	Performance	CICS Wait for Storage in ESDSA
04037	Impacting	Performance	CICS Wait for Storage in RDSA
04038	Impacting	Performance	CICS Wait for Temp Storage String
04039	Impacting	Performance	CICS Wait for Temp Storage Buffer
04040	Impacting	Performance	CICS LSR Pool String Wait
04041	Impacting	Performance	CICS Wait for CF Data Tables Locking
04042	Impacting	Performance	CICS Wait for CF Data Tables Non-Locking
04101	Critical	Service	Prime Time CICS Throughput Degradation
04102	Impacting	Performance	Hourly CICS System Resource Overload
04103	Critical	Performance	Hourly CICS Capacity Limit Objective Exceeded
04104	Critical	Performance	Hourly CICS Paging Rate Limit Exceeded
04105	Critical	Service	Hourly CICS Short Service Objective Missed
04106	Critical	Service	Hourly CICS Medium Service Objective Missed
04107	Warning	Service	Hourly CICS Long Service Objective Missed
04108	Critical	Service	Hourly CICS Total Service Objective Missed
04109	Critical	Availability	Hourly CICS Availability Objective Missed
04110	Critical	Performance	Hourly CICS Throughput Limit Objective Exceeded
04201	Warning	Workload	Hourly CICS User Resource Overload
04202	Critical	Service	Prime Time CICS User Total Service Objective Missed

Figure 4-1. CICS Exception List

This section contains the following topics:

[4.1 Exception Process Overview](#) (see page 385)

[4.2 Setting Exception Values](#) (see page 388)

[4.3 Detailed Exception Descriptions](#) (see page 389)

4.1 Exception Process Overview

The CA MICS exception process consists of the exceptions themselves, a set of standard reports, CA MICS Information Center Facility II (MICF) inquiries, an exception test routine, and an exception value analysis routine. Each of these is described below.

EXCEPTIONS

An exception is the occurrence of an event that merits visibility and attention. It can be any of the following:

- o An occurrence that is a distinct problem (for example, CICS abended at 2:00 p.m.)
- o A problem that requires further research (for example, a TSO user overloaded the system from 1:00 to 1:30 p.m.)
- o A standard, security, or audit violation (for example, user XYZ is not authorized to use PDZAP and was detected using it seven times yesterday)

Because the volume of exception occurrences can be quite large, CA MICS provides means to categorize, aggregate, consolidate, and prioritize these occurrences to meet your needs. Each exception has:

- o An exception number for unique definition
- o A severity level to signify degree of importance
- o A management area to identify area of responsibility

STANDARD REPORTS

The standard exception reports provide a concise, integrated method for problem reporting. The following reports can be produced as part of the CA MICS DAILY job:

- o Exception Management Overview Report
- o Severity Level Exception Summary Report
- o Management Area Exception Summary Report

You control which reports are produced via the REPORT EXCEPTIONS statement in prefix.MICS.PARMS(EXECDEF). Refer to Section 2.3.5 of the PIOM for more information.

Two additional standard reports can be produced as required to provide the necessary background detail to effectively

analyze reported exceptions. The two reports are:

- o Full Exception Detail Report
- o Short Exception Detail Report

MICF INQUIRIES

The catalog group EXCEPT that is shipped with CA MICS contains a number of standard MICF inquiries that can be used to report exception conditions. In the following inquiry list, graphic reports whose value of x is C produce color graphic reports using SAS/GRAPH. If the value of x is P, printer graphic reports are produced without using SAS/GRAPH.

- o BASxM1 - Monthly Exception Summary Report
- o BASxM2 - Monthly Mgmt. Area Exception Summary Report
- o BASxM3 - Monthly Info. Area Exception Summary Report
- o BASxM4 - Monthly Exception Management Overview Report
- o BASxM5 - Monthly Info. Area Exception Overview Report
- o BASxW1 - Weekly Exception Summary Report
- o BASxW2 - Weekly Mgmt. Area Exception Summary Report
- o BASxW3 - Weekly Info. Area Exception Summary Report

The following inquiries that produce printed reports are also available:

- o BASLD2 - Daily Severity Level Exception Summary Report
- o BASLD3 - Daily Mgmt. Area Exception Summary Report
- o BASLD4 - Daily Short Exception Detail Report
- o BASLD5 - Daily Full Exception Detail Report
- o BASLD6 - Daily Exception Ranking Report
- o BASLM6 - Monthly Exception Ranking Report

These standard inquiries have execution-time parameter selection that permits you to report on a subset of the exceptions. For example, inquiry BASLD5 allows selection on SYSID, Information Area, Management Area, Severity Level, and other criteria.

EXCEPTION TEST ROUTINE

Each CA MICS component has an exception test routine that is invoked in the DAY200 step of the CA MICS DAILY job. You control which routines are invoked using the CREATE EXCEPTIONFILES statement in prefix.MICS.PARMS(EXECDEF). Refer to the PIOM, Section 2.3.5, for more information on

EXECDEF.

An exception test routine, written in the SAS language, defines the exception and tests to determine whether or not the exception condition is present in the data being processed. The distributed exception test routine for each CA MICS component is contained in `sharedprefix.MICS.SOURCE(DYcccEXC)`, where `ccc` is the component identifier. When the `DAY200` step invokes the test routines, it does so by `%INCLUDE`ing the `DYcccEXC` member from `prefix.MICS.USER.SOURCE`. As distributed, that member then `%INCLUDEs` the member from `sharedprefix.MICS.SOURCE`. See Section 4.2 of this guide, *Setting Exception Values*, for more information on `DYcccEXC`.

EXCEPTION VALUE ANALYSIS ROUTINE

Each CA MICS component has an exception value analysis (EVA) routine that you can use to help determine values for the exception conditions. The EVA routine extracts information from the CA MICS database. Descriptive statistics for the values of variables used in exception tests are printed in the Exception Value Analysis Report.

The EVA routine for each CA MICS component is stored in `sharedprefix.MICS.SOURCE(cccEVA)`. The JCL to execute the EVA process is contained in `prefix.MICS.CNTL(cccEVA)`.

4.2 Setting Exception Values

Exception test routines contain exception tests that completely define the tests made to determine the exception condition. The exception tests also contain the definitions that identify and classify the exception for reporting and analysis. A sample exception test is:

```
*
** 04001
** CICS APPLICATION EXCEEDS AVG ELAPSED TIME STANDARD
*;
  SECONDS=seconds;
  IF CSUAVTTM > SECONDS THEN DO;
    EXCCODE ='04001';
    SEVERITY='W';
    MGMTAREA='SERVICE';
    EXCDISC1=
    'CICS APPLICATION EXCEEDS AVG ELAPSED TIME STANDARD';
    EXCDISC2='APPL=' || CICAPU ||
              ', AVG ELAPSED TIME=' ||
              PUT(CSUAVTTM,TIME.);
    LINK HIT;
  END;
```

This exception test is processed for each observation that is in the latest cycle of the CICS User Activity File (CICCSU). If your site has activated the CICS Application Unit Activity File (CICCAU), that file's AVTTM element (CAUAVTTM) is used to value this exception condition.

The test is positive when the variable CSUAVTTM has a value greater than 'seconds', which indicates that the average response time was greater than the time specified. When the test is positive, the exception is categorized by providing the appropriate values for EXCCODE, SEVERITY, and MGMTAREA. EXCDISC1 provides a constant title for the exception. EXCDISC2 provides variable information for the conditions that caused the test to be positive. The LINK HIT statement invokes a routine that causes the exception condition to eventually be written to the Exception Activity File (ADMEXC) for later processing by the standard reports or MICF inquiries.

The values for most exception conditions should be determined uniquely for different environments within a single organization. For example, a system paging rate that would be excessive during the nightly batch processing may be normal during the daytime hours, which have heavy interactive

usage. Also, different processors can support different paging rates. Numbers used as exception values should not be defined without some analysis of installation history, performance, and user requirements.

The exception value analysis routine uses the CA MICS Data Base to produce a report that provides a statistical analysis of the values of the variables used in the standard exception tests distributed with the products. Using the results of this analysis, along with your installation's internal political, security, or standards policies, industry publications, and your own analysis of data element behavior, you can determine the modifications that you should make to each exception test to provide meaningful exceptions in your environment.

You can also modify the exception test routine to add your own tests for additional exception conditions not detected by the standard tests delivered with the product.

4.3 Detailed Exception Descriptions

This section provides an in-depth description of each CICS Analyzer standard exception available in the exception report process. The exceptions are organized by number and appear sequentially, starting with exception 04001.

The description format provides the title, the number, statements on the purpose and rationale, and a definition for each exception to give you some insight as to the meaning and use of the information. Finally, SAS code is listed, with a short explanation on modifying the exception threshold values.

Note: if the CICCAU (CICS Application Unit Activity) file is not active, the CICCUSU (CICS User Activity) file will be used for reporting.

04001: CICS Application Exceeds Avg Elapsed Time Standard

FILE: CICS Application Unit Activity File
SAS FILE NAME: DAYS.CICCAU01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those CICS applications whose average elapsed time totalled more than the installation-defined elapsed time service objective.

RATIONALE: An installation can set a service objective for application elapsed time. Setting the service objective this way recognizes the importance of servicing a large number of individual requests while identifying CICS applications that reserve valuable resources for extended periods.

DEFINITION: This exception is detected when an individual application's elapsed time exceeds the installation's elapsed time objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04001  
** CICS APPLICATION EXCEEDS AVG ELAPSED TIME STANDARD  
*;  
SECONDS=seconds;  
IF CAUAVTTM > SECONDS  
THEN DO;  
EXCCODE='04001'; SEVERITY='W'; MGMTAREA='SERVICE';  
EXCDESC1=  
'CICS APPLICATION EXCEEDS AVG ELAPSED TIME STANDARD';  
EXCDESC2= 'APPL=' || CICAPU ||  
          ', AVG ELAPSED TIME=' ||  
          PUT(CAUAVTTM,TIME.);  
  LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the value of seconds to define the installation service objective according to the following convention:

seconds - The number of seconds in the service objective.
An objective of five seconds appears as:

SECONDS = 5;

04002: CICS Application Exceeds Avg TCB/SRB CPU Limit

FILE: CICS Application Unit Activity File
SAS FILE NAME: DAYS.CICCAU01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those CICS applications whose average TCB/SRB time totalled more than the installation-defined CPU time service objective.

RATIONALE: An installation can set a service objective for application TCB/SRB time. Setting the service objective this way recognizes the importance of servicing a large number of individual requests, while identifying CICS applications which use large amounts of CPU time. Such CPU resource users could be tuned by restructuring the programs involved or compensated for by careful priority assignment.

DEFINITION: This exception is detected when an individual application's elapsed time exceeds the installation's TCB/SRB time objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04002  
** CICS APPLICATION EXCEEDS AVG TCB/SRB CPU LIMIT  
*;  
SECONDS=seconds;  
IF CAUAVTRT > SECONDS  
THEN DO;  
EXCCODE='04002'; SEVERITY='W'; MGMTAREA='SERVICE';  
EXCDESC1=  
'CICS TRANSACTION EXCEEDED TCB/SRB CPU USAGE LIMIT';  
EXCDESC2= 'APPL=' || CICAPU ||  
          ', AVG CPU TIME=' ||  
          PUT(CAUAVTRT,TIME12.2);  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the value of seconds to define your installation's service objective according to the

following convention:

seconds - The number of CPU seconds for the service
objective. An objective of five seconds appears as:

```
SECONDS = 5;
```

04004: CICS Application Exceeds Avg Wait Time Limit

FILE: CICS Application Unit Activity File
SAS FILE NAME: DAYS.CICCAU01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those CICS applications whose average wait time totalled more than the installation-defined wait time objective.

RATIONALE: An installation can set a service objective for application wait time. Setting the service objective this way recognizes the impact of individual applications that monopolize CICS resources while they are undispatchable. Such applications may be controlled by transaction class max task limits or restructuring to resolve such problems as long-term enqueue lockouts.

DEFINITION: This exception is detected when an application's average hourly wait time exceeds the installation's wait time objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04004  
** CICS APPLICATION EXCEEDS AVG WAIT TIME LIMIT  
*;  
SECONDS=seconds;  
FORMAT AVG TIME.;  
IF CAUTRANS > 0 THEN AVG=CAUWAITM/CAUTRANS; ELSE AVG=0;  
IF AVG > SECONDS  
THEN DO;  
EXCCODE='04004'; SEVERITY='W'; MGMTAREA='SERVICE';  
EXCDESC1=  
'CICS APPLICATION EXCEEDS AVG WAIT TIME LIMIT';  
EXCDESC2= 'APPL=' || CICAPU ||  
          ', AVG WAIT TIME=' ||  
          PUT(AVG,TIME.);  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the value of seconds to define your installation's service objective according to the following convention:

seconds - The number of wait time seconds in the objective. An objective of five seconds appears as:

SECONDS = 5;

04005: CICS Application Exceeds Avg Residency Time Limit

FILE: CICS Application Unit Activity File
SAS FILE NAME: DAYS.CICCAU01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those CICS applications whose average residency time totalled more than the installation-defined residency time objective.

RATIONALE: An installation can set a service objective for application residency time. Setting the service objective this way recognizes the importance of servicing a large number of individual requests, while identifying CICS applications that reserve valuable resources for extended periods.

DEFINITION: This exception is detected when an individual application's average residency time exceeds the installation's residency time objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04005  
** CICS APPLICATION EXCEEDS AVG RESIDENCY TIME LIMIT  
*;  
SECONDS=seconds;  
IF CAUTRANS > 0 THEN AVG=CAURESTM/CAUTRANS; ELSE AVG=0;  
IF AVG > SECONDS  
THEN DO;  
EXCCODE='04005'; SEVERITY='W'; MGMTAREA='SERVICE';  
EXCDESC1=  
'CICS APPLICATION EXCEEDS AVG RESIDENCY TIME LIMIT';  
EXCDESC2= 'APPL=' || CICAPU ||  
          ', AVG RESIDENCY TIME=' ||  
          PUT(AVG,TIME.);  
  LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the value of seconds to define your installation's residency time objective according

to the following convention:

seconds = The number of seconds for the service
objective. An objective of 60 seconds appears as:

```
SECONDS = 60;
```

04006: CICS Application Exceeds Max Memory Limit

FILE: CICS Application Unit Activity File
SAS FILE NAME: DAYS.CICCAU01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Workload (MGMTAREA='WORKLOAD')

PURPOSE: Identifies those CICS applications whose memory was greater than the installation-defined memory limit.

RATIONALE: An installation can set a limit for maximum memory usage. Setting the limit this way recognizes the importance of conserving CICS dynamic storage, since monopolizing large amounts of storage for long periods of time can lead to overall system degradation from fragmentation and short-on-storage conditions.

DEFINITION: This exception is detected when an individual transaction's memory high-water mark exceeds the installation's memory objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04006  
** CICS APPLICATION EXCEEDS MAX MEMORY LIMIT  
*;  
COREMAX=number-of-bytes;  
IF CAUMEMHI > COREMAX  
THEN DO;  
EXCCODE='04006'; SEVERITY='W'; MGMTAREA='WORKLOAD';  
EXCDESC1=  
'CICS APPLICATION EXCEEDS MAX MEMORY LIMIT';  
EXCDESC2= 'APPL=' || CICAPU ||  
          ', MAX MEMORY USAGE=' ||  
          PUT(CAUMEMHI,7.);  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the value number-of bytes to define your installation's memory limit according to the following convention:

number-of-bytes = The number of bytes of main memory which a typical transaction should not exceed. A 40,000 byte limit appears as:

COREMAX = 40000;

04007: CICS DL/I DMB Pool Wait

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICIN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identify an overutilized DL/I DMB pool.

RATIONALE: CICS DL/I DMB pool wait may indicate that the DL/I DMB pool is too small. CICS will suspend one or more tasks until storage in the DL/I DMB pool becomes available.

DEFINITION: This exception is noted when the DL/I DMB pool wait indicator in any transaction record is set.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****
** 04007
** CICS DL/I DMB POOL WAIT
*;
IF CINCODE = 7
THEN DO;
EXCCODE='04007'; SEVERITY='W'; MGMTAREA='PERFORMANCE';
EXCDESC1 = CINTXT;
EXCDESC2= 'TRAN=' || TRANCODE ||
          ' TERM=' || TERMINAL ||
          ' OPER=' || OPERID;
LINK HIT;
END;
```

THRESHOLD MODIFICATION: None required

04008: CICS DB2 Pool Overflow

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identify an overutilized DB2 thread.

RATIONALE: CICS transactions have a set number of DB2 threads assigned to them. If all those threads are being utilized at the time that the transaction requires an thread, the transaction may get the thread from a pool of threads that has been set aside for such instances.

DEFINITION: This exception is noted when the DB2 pool overflow indicator in any transaction record is set.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****  
** 04008  
** CICS DB2 POOL OVERFLOW  
*;  
IF CINCODE = 8  
THEN DO;  
EXCCODE='04008'; SEVERITY='W'; MGMTAREA='PERFORMANCE';  
EXCDESC1 = CINTXT;  
EXCDESC2= 'TRAN=' || TRANCODE ||  
          ' TERM=' || TERMINAL ||  
          ' OPER=' || OPERID;  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: None required

04009: CICS DB2 Wait Without Thread

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICIN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identify an overutilized DB2 thread.

RATIONALE: CICS transactions have a set number of DB2 threads assigned to them. If a transaction has gone into a wait state after requesting I/O from DB2 and if the transaction has not been assigned a thread, the transaction may be waiting for a thread.

DEFINITION: This exception is noted when the DB2 wait without thread indicator is set for any transaction.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****
** 04009
** CICS DB2 WAIT WITHOUT THREAD
*;
IF CINCODE = 9
THEN DO;
EXCCODE='04009'; SEVERITY='W'; MGMTAREA='PERFORMANCE';
EXCDESC1 = CINTXT;
EXCDESC2= 'TRAN=' || TRANCODE ||
          ' TERM=' || TERMINAL ||
          ' OPER=' || OPERID;
LINK HIT;
END;
```

THRESHOLD MODIFICATION: None required

04010: CICS Storage Violation Detected

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a storage violation.

RATIONALE: CICS storage violations can be due to invalid storage accounting areas or bad area pointer addresses. In some instances, CICS may not be able to recover from a storage violation and willabend.

DEFINITION: This exception is noted when a bad storage area is examined in the course of processing a Storage Control event.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****  
** 04010  
** CICS STORAGE VIOLATION DETECTED  
*;  
IF CINCODE = 10  
THEN DO;  
EXCCODE='04010'; SEVERITY='C'; MGMTAREA='PERFORMANCE';  
EXCDISC1 = CINTXT;  
EXCDISC2= 'TRAN=' || TRANCODE ||  
          ' TERM=' || TERMINAL ||  
          ' OPER=' || OPERID;  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: None required

04011: CICS Abnormal Condition Transaction

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICCN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a possible intentional attempt to gain unauthorized access to the CICS system.

RATIONALE: CICS security violations can take many forms. All unsuccessful attempts to access transactions which the user is not authorized to access will result in a CSAC Abnormal Condition Transaction monitor record.

DEFINITION: This exception is noted when the monitor record for any CSAC transaction is processed.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
** 04011
** CICS ABNORMAL CONDITION TRANSACTION
*;
IF CINCODE = 11
THEN DO;
EXCCODE='04011'; SEVERITY='C'; MGMTAREA='SECURITY';
EXCDESC1 = CINTXT;
EXCDESC2= 'TRAN=' || TRANCODE ||
          ' TERM=' || TERMINAL ||
          ' OPER=' || OPERID;
LINK HIT;
END;
```

THRESHOLD MODIFICATION: None required

04012: CICS LSR String Wait

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies an overutilized LSR string.

RATIONALE: Some files use local-shared resources (LSR) for their VSAM buffers and strings. A CICS transaction requires a string in order to access the file. If a string is not available when the transaction needs to access the file, the transaction will have to wait until a string becomes available.

DEFINITION: This exception is noted when the LSR string wait indicator is set for any transaction record.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****  
** 04012  
** CICS LSR STRING WAIT  
*;  
IF CINCODE = 12  
THEN DO;  
EXCCODE='04012'; SEVERITY='W'; MGMTAREA='PERFORMANCE';  
EXCDESC1 = CINTXT;  
EXCDESC2= 'TRAN=' || TRANCODE ||  
          ' TERM=' || TERMINAL ||  
          ' OPER=' || OPERID;  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: None required

04013: CICS Max Task

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICIN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a CICS task control overload.

RATIONALE: CICS max task conditions occur when CICS has received more requests for task initiations than it can fill. In this case, CICS does not initiate one or more tasks until an executing task terminates. The max task limits (system total and by task class) exist to prevent over-committing system resources by initiating too many tasks or too many tasks of the same class at once. Max task requires constant monitoring to be an effective CICS tuning aid. Inattention to max task limits for long periods can cause great delays for transactions and users of lesser priority.

DEFINITION: This exception is noted when the max task indicator in any transaction record is set.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
** 04013
** CICS MAX TASK
*;
IF CINCODE = 1
THEN DO;
EXCCODE='04013'; SEVERITY='C'; MGMTAREA='PERFORMANCE';
EXCDESC1 = CINTXT;
EXCDESC2= 'TRAN=' || TRANCODE ||
          ' TERM=' || TERMINAL ||
          ' OPER=' || OPERID;
LINK HIT;
END;
```

THRESHOLD MODIFICATION: None required

04014: CICS Active Max Task

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a CICS task control overload.

RATIONALE: CICS active max task condition occurs when CICS has reached the specified limit on the number of tasks which may be active at one time. CICS will not activate any more tasks until an active task either terminates or becomes inactive.

DEFINITION: This exception is noted when the active max task indicator in any transaction record is set.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
** 04014
** CICS ACTIVE MAX TASK
*;
IF CINCODE = 2
THEN DO;
EXCCODE='04014'; SEVERITY='W'; MGMTAREA='PERFORMANCE';
EXCDESC1 = CINTXT;
EXCDESC2= 'TRAN=' || TRANCODE ||
          ' TERM=' || TERMINAL ||
          ' OPER=' || OPERID;
LINK HIT;
END;
```

THRESHOLD MODIFICATION: None required

04015: CICS Short on Storage

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICCN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a CICS storage overutilization.

RATIONALE: CICS short on storage situations occur when CICS has received more requests for dynamic storage than it can fill. In this case, CICS suspends one or more tasks until enough storage to satisfy their outstanding requests becomes available. Under certain conditions with long-running tasks, storage may never become available, and system performance will degrade over long periods of time.

DEFINITION: This exception is noted when the short-on-storage indicator in any transaction record is set.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****
** 04015
** CICS SHORT-ON-STORAGE TRANSACTION DELAY
*;
IF CINCODE = 3
THEN DO;
EXCCODE='04015'; SEVERITY='C'; MGMTAREA='PERFORMANCE';
EXCDESC1 = CINTXT;
EXCDESC2= 'TRAN=' || TRANCODE ||
          ' TERM=' || TERMINAL ||
          ' OPER=' || OPERID;
LINK HIT;
END;
```

THRESHOLD MODIFICATION: None required

04016: CICS VSAM String Wait

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a VSAM string overutilization.

RATIONALE: To access some files, CICS transactions require a VSAM string. If a string is not available when the transaction needs to access the file, the transaction will have to wait until a string becomes available.

DEFINITION: This exception is noted when the VSAM string wait indicator in any transaction record is set.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****  
** 04016  
** CICS VSAM STRING WAIT  
*;  
IF CINCODE = 4  
THEN DO;  
EXCCODE='04016'; SEVERITY='W'; MGMTAREA='PERFORMANCE';  
EXCDESC1 = CINTXT;  
EXCDESC2= 'TRAN=' || TRANCODE ||  
          ' TERM=' || TERMINAL ||  
          ' OPER=' || OPERID;  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: None required

04017: CICS VSAM Buffer Wait

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICIN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a VSAM buffer overutilization.

RATIONALE: To access some files, CICS transactions require a VSAM i/o buffer. If a buffer is not available when the transaction needs to access the file, the transaction will have to wait until a buffer becomes available.

DEFINITION: This exception is noted when the VSAM buffer wait indicator in any transaction record is set.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
** 04017
** CICS VSAM BUFFER WAIT
*;
IF CINCODE = 5
THEN DO;
EXCCODE='04017'; SEVERITY='W'; MGMTAREA='PERFORMANCE';
EXCDDESC1 = CINTXT;
EXCDDESC2= 'TRAN=' || TRANCODE ||
           ' TERM=' || TERMINAL ||
           ' OPER=' || OPERID;
LINK HIT;
END;
```

THRESHOLD MODIFICATION: None required

04018: CICS DL/I Thread Wait

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a DL/I thread overutilization.

RATIONALE: The CICS / DL/I interface is defined as having a set number of threads. The CICS transactions require threads in order to access the DL/I database. If a thread is not available when a transaction needs to access the DL/I database, the transaction will have to wait until one becomes available.

DEFINITION: This exception is noted when the DL/I thread wait indicator in any transaction record is set.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****  
** 04018  
** CICS DL/I THREAD WAIT  
*;  
IF CINCODE = 6  
THEN DO;  
EXCCODE='04018'; SEVERITY='W'; MGMTAREA='PERFORMANCE';  
EXCDISC1 = CINTXT;  
EXCDISC2= 'TRAN=' || TRANCODE ||  
          ' TERM=' || TERMINAL ||  
          ' OPER=' || OPERID;  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: None required

04019: CICS Wait for Storage in DSA

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICIN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies CICS storage constraint or unreasonably large storage requests.

RATIONALE: A wait on CICS DSA (dynamic storage area) occurs when an unconditional request (SUSPEND=YES) for resource SMDSA cannot be satisfied. This can be attributed to the task having issued an unconditional GETMAIN request for an unreasonably large amount of storage, or to the system getting too close to SOS, or to the storage becoming too fragmented to satisfy the request.

DEFINITION: This exception is noted when an CMF exception class record is produced for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */
/* EX NR: 04019 */
/* TITLE: CICS WAIT FOR STORAGE IN DSA */
/* FILE: DETAIL.CICIN01 */
/* ***** */
IF CINCODE = 19
  THEN DO;
    EXCCODE='04019'; SEVERITY='I'; MGMTAREA='PERFORMANCE';
    EXCDISC1 = CINTXT;
    EXCDISC2= 'TRAN=' || TRANCODE ||
              'TERM=' || TERMINAL ||
              'USER=' || USERID ;
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION: None required

04020: CICS Wait for Storage in EDSA

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies CICS storage constraint or unreasonably large storage requests above the 16 megabyte line.

RATIONALE: A wait on CICS EDSA (extended dynamic storage area) occurs when an unconditional request (SUSPEND=YES) for resource SMEDSA cannot be satisfied. This can be attributed to the task having issued an unconditional GETMAIN request for an unreasonably large amount of storage, or to the systems getting too close to SOS, or to the storage becoming too fragmented for the request to be satisfied.

DEFINITION: This exception is noted when an CMF exception class record is produced for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */
/* EX NR: 04020 */
/* TITLE: CICS WAIT FOR STORAGE IN EDSA */
/* FILE: DETAIL.CICIN01 */
/* ***** */
IF CINCODE = 20
  THEN DO;
    EXCCODE='04020'; SEVERITY='I'; MGMTAREA='PERFORMANCE';
    EXCDISC1 = CINTXT;
    EXCDISC2= 'TRAN=' || TRANCODE ||
              'TERM=' || TERMINAL ||
              'USER=' || USERID ;
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION: None required

04021: CICS Wait for Temporary Storage (Main)

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICIN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a constraint in temporary storage or unreasonably large requests for temporary storage.

RATIONALE: A wait on temporary storage occurs when an unconditional request (SUSPEND specified) cannot be satisfied. This can be attributed either to the task having issued a request requiring an unreasonably large amount of storage, or to too little available storage or fragmented storage.

DEFINITION: This exception is noted when an CMF exception class record is produced for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */
/* EX NR: 04021 */
/* TITLE: CICS WAIT FOR TEMPORARY STORAGE (MAIN) */
/* FILE: DETAIL.CICIN01 */
/* ***** */
IF CINCODE = 21
  THEN DO;
    EXCCODE='04021'; SEVERITY='I'; MGMTAREA='PERFORMANCE';
    EXCDISC1 = CINTXT;
    EXCDISC2= 'TRAN=' || TRANCODE ||
              'TERM=' || TERMINAL ||
              'USER=' || USERID ;
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION: None required

04022: CICS Wait for File String

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies insufficient string definition for concurrent access.

RATIONALE: The number of strings defined for a VSAM data set determines the number of tasks that can access the data set concurrently. A wait occurs when the task cannot get a VSAM string because all strings are in use. A frequent occurrence of this condition for the same file may warrant an increase in the value of STRINGS.

DEFINITION: This exception is noted when an CMF exception class record is written for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */
/* EX NR: 04022 */
/* TITLE: CICS WAIT FOR FILE STRING */
/* FILE: DETAIL.CICIN01 */
/* ***** */
IF CINCODE = 22
  THEN DO;
    EXCCODE='04022'; SEVERITY='I'; MGMTAREA='PERFORMANCE';
    EXCDESC1 = CINTXT;
    EXCDESC2= 'TRAN=' || TRANCODE ||
              'TERM=' || TERMINAL ||
              'USER=' || USERID ;
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION: None required

04023: CICS Wait for File Buffer

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICIN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies insufficient file buffer specification.

RATIONALE: A file buffer wait occurs when a VSAM buffer is unavailable. A frequent occurrence of this condition for the same file may warrant an increase in buffer definitions.

DEFINITION: This exception is noted when a CMF exception class record is written for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */
/* EX NR: 04023 */
/* TITLE: CICS WAIT FOR FILE BUFFER */
/* FILE: DETAIL.CICIN01 */
/* ***** */
IF CINCODE = 23
  THEN DO;
    EXCCODE='04023'; SEVERITY='I'; MGMTAREA='PERFORMANCE';
    EXCDESC1 = CINTXT;
    EXCDESC2= 'TRAN=' || TRANCODE ||
              'TERM=' || TERMINAL ||
              'USER=' || USERID ;
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION: None required

04024: CICS LSRPOOL Buffer Wait

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies insufficient LSRPOOL buffers.

RATIONALE: A LSRPOOL buffer wait occurs when a VSAM buffer is unavailable. A frequent occurrence of this condition for the same file may warrant an increase in buffer definitions.

DEFINITION: This exception is noted when an CMF exception class record is written for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */
/* EX NR: 04024 */
/* TITLE: CICS LSRPOOL BUFFER WAIT */
/* FILE: DETAIL.CICIN01 */
/* ***** */
IF CINCODE = 24
  THEN DO;
    EXCCODE='04024'; SEVERITY='I'; MGMTAREA='PERFORMANCE';
    EXCDISC1 = CINTXT;
    EXCDISC2= 'TRAN=' || TRANCODE ||
              'TERM=' || TERMINAL ||
              'USER=' || USERID ;
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION: None required

04025: CICS Program Compressions

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICIN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies CICS storage constraint.

RATIONALE: Dynamic storage for non-resident programs are not freed at end of task. Therefore, CICS performs a function called "program compression" to remove programs from storage when either a GETMAIN cannot be satisfied from available free storage, or the number of free storage pages is lower than the threshold defined for storage cushion. A high number of program compressions indicates storage constraint and tuning is needed.

DEFINITION: This exception is noted when a CMF exception class record (pre-CICS 3.1.1) is written for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */
/* EX NR: 04025 */
/* TITLE: CICS PROGRAM COMPRESSION */
/* FILE: DETAIL.CICIN01 */
/* ***** */
IF CINCODE = 25
  THEN DO;
    EXCCODE='04025'; SEVERITY='I'; MGMTAREA='PERFORMANCE';
    EXCDISC1 = CINTXT;
    EXCDISC2= 'TRAN=' || TRANCODE ||
              'TERM=' || TERMINAL ||
              'OPER=' || OPERID ;
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION: None required

04026: CICS Storage Control Suspend

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a CICS storage constraint.

RATIONALE: A wait on dynamic storage occurs when an unconditional request (SUSPEND=YES) cannot be satisfied. This can be attributed either to the task having issued an unconditional GETMAIN request for an unreasonably large amount of storage, or to the system getting too close to SOS, or to the storage becoming too fragmented for the request to be satisfied.

DEFINITION: This exception is noted when a CMF exception record (pre-CICS 3.1.1) is written for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */  
/* EX NR: 04026 */  
/* TITLE: CICS STORAGE CONTROL SUSPEND */  
/* FILE: DETAIL.CICIN01 */  
/* ***** */  
IF CINCODE = 26  
  THEN DO;  
    EXCCODE='04026'; SEVERITY='I'; MGMTAREA='PERFORMANCE';  
    EXCDISC1 = CINTXT;  
    EXCDISC2= 'TRAN=' || TRANCODE ||  
              'TERM=' || TERMINAL ||  
              'OPER=' || OPERID ;  
    LINK HIT;  
  END;
```

THRESHOLD MODIFICATION: None required

04027: CICS Temporary Storage PUT Wait

FILE: CICS Incident File
 SAS FILE NAME: DETAIL.CICIN01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies a shortage in temporary storage.

RATIONALE: A wait on temporary storage occurs when an unconditional request (SUSPEND specified) cannot be satisfied. This can be attributed either to the task having issued a request requiring an unreasonably large amount of storage, or to too little available storage or fragmented storage.

DEFINITION: This exception is noted when an CMF exception class record is produced for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */
/* EX NR: 04027 */
/* TITLE: CICS TEMP STORAGE PUT WAIT */
/* FILE: DETAIL.CICIN01 */
/* ***** */
IF CINCODE = 27
  THEN DO;
    EXCCODE='04027'; SEVERITY='I'; MGMTAREA='PERFORMANCE';
    EXCDISC1 = CINTXT;
    EXCDISC2= 'TRAN=' || TRANCODE ||
              'TERM=' || TERMINAL ||
              'OPER=' || OPERID ;
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION: None required

04028: CICS Main Storage Wait

FILE: CICS Incident File
SAS FILE NAME: DETAIL.CICIN01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies an unreasonably large storage request.

RATIONALE: A wait on CICS dynamic storage occurs when an unconditional request (SUSPEND=YES) cannot be satisfied. This can be attributed either to the task having issued a request for an unreasonably large amount of storage, or to the system getting too close to SOS, or to storage becoming too fragmented for the request to be satisfied.

DEFINITION: This exception is noted when an CMF exception class record (pre-CICS 3.1.1) is produced for this condition.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
/* ***** */
/* EX NR: 04028 */
/* TITLE: CICS MAIN STORAGE WAIT */
/* FILE: DETAIL.CICIN01 */
/* ***** */
IF CINCODE = 28
  THEN DO;
    EXCCODE='04028'; SEVERITY='I'; MGMTAREA='PERFORMANCE';
    EXCDISC1 = CINTXT;
    EXCDISC2= 'TRAN=' || TRANCODE ||
              'TERM=' || TERMINAL ||
              'OPER=' || OPERID ;
    LINK HIT;
  END;

THRESHOLD MODIFICATION: None required
```

04101: Prime Time CICS Throughput Degradation

FILE: CICS System Activity File
 SAS FILE NAME: DAYS.CICCSY01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
 MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those hours during prime time when the actual number of CICS transactions processed is below the objective.

RATIONALE: An installation can set a service objective for transaction throughput. Setting the objective this way recognizes the importance of servicing a consistent number of requests during prime time.

DEFINITION: This exception is detected when the number of transactions processed during a prime-time hour falls below a given objective number.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;
*
** 04101
** PRIME TIME CICS THROUGHPUT DEGRADATION
*;
EVENTS=number-of-responses;
IF CSYTRANS < EVENTS
AND (DAYNAME='MON' OR DAYNAME='TUE' OR DAYNAME='WED'
    OR DAYNAME='THU' OR DAYNAME='FRI')
    AND (HOUR >= hh1 AND HOUR <= hh2)
THEN DO;
EXCCODE='04101'; SEVERITY='C'; MGMTAREA='SERVICE';
EXCDESC1=
'PRIME TIME CICS THROUGHPUT DEGRADATION';
EXCDESC2='OBJCTV ' || PUT(EVENTS,6.) ||
'/HOUR, ACTUAL ' ||
    PUT (CSYTRANS,6.) || '/HOUR. ' ;
LINK HIT;
END;
```

THRESHOLD MODIFICATION: Modify the values of number of responses, times, and days to define your installation's

service objective according to the following conventions:

number-of-responses - The minimum number of total response events for which this exception is to be considered and would appear as:

EVENTS = 1000;

hh1 and hh2 - The hour range and days of the week to be included in the exception. The hour of the day must be within the hh1 through hh2 range. Prime time specified as 8 a.m. through 6:59 p.m. (18:59) appears as:

AND (HOUR >= 08 AND HOUR <= 18)

The days of the week are represented by the first three characters of their names. They must be enclosed in quotes and separated by ORs. Prime time specified as MONDAY through FRIDAY appears as:

AND (DAYNAME='MON' OR DAYNAME='TUE' OR
DAYNAME='WED' OR DAYNAME='THU' OR
DAYNAME='FRI')

04102: Hourly CICS System Resource Overload

FILE: CICS System Activity File
 SAS FILE NAME: DAYS.CICCSY01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Impacting (SEVERITY='I')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies those hours when the CICS workload is so large that service may be impacted.

RATIONALE: Poor performance will result when the CICS workload exceeds the configuration limit. It is possible for one or more CICS users to cause degraded response for all by consuming large amounts of CPU time or by doing large numbers of I/Os. At times this may be the result of a program which is in an infinite loop.

It may be necessary to:

- o Review the heaviest user's work for efficiency improvements.
- o Have some users run at a lower priority.
- o Defer some of the CICS work until off-hours.

If the installation wishes to provide some overall distribution of service between major subsystems, e.g., batch, TSO, CICS, IMS, etc., then, if MVS/SE is installed, time slice groups may be used to achieve this.

DEFINITION: This exception is detected when the hourly CICS CPU time or I/O exceeds the installation-defined limits.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;
*
** 04102
** HOURLY CICS SYSTEM RESOURCE OVERLOAD
*;
IF CSYCPUTM > cpu-time
OR CSYSERVU > service-units
OR CSYEXCPS > excps
```

```
THEN DO;  
EXCCODE='04102'; SEVERITY='I'; MGMTAREA='PERFORMANCE';  
EXCDESC1='HOURLY CICS SYSTEM RESOURCE OVERLOAD';  
EXCDESC2='CPU TIME(HH:MM:SS)= '  
    || PUT(CSYCPUTM,TIME.) ||  
    ', EXCPS=' || PUT(CSYEXCPS,7.) ||  
    ', SUS=' || PUT(CSYSERVU,8.);  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the values used for CPU time, service units, and EXCPs according to the following conventions:

cpu-time - the amount of CPU time is described as hours, minutes, and seconds. A threshold of 3 minutes and 20 seconds appears as:

```
CSYCPUTM > HMS(0,3,20)
```

service-units - the number of service units. A threshold of 9000000 service units appears as:

```
CSYSERVU > 9000000
```

excps - the number of EXCPs (I/O). A threshold of 150000 EXCPs appears as:

```
CSYEXCPS > 150000
```


04103: Hourly CICS Capacity Limit Objective Exceeded

FILE: CICS System Activity File
 SAS FILE NAME: DAYS.CICCSY01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies those hours when the installation-defined capacity limit is exceeded.

RATIONALE: Poor performance will result when CICS capacity, measured in service units, exceeds the configuration limit. This can be caused by growth in the number of CICS users, in which case system reconfiguration or CICS restriction will be required to meet service objectives. It is also possible that this condition can be caused by a program that is in a loop or by a few users who are putting an exceptional load on the CICS system.

If installation management's objective is to restrict CICS resource consumption to the defined capacity limit, the detection of this exception may indicate the need to do one or more of the following:

- o Lower the service objectives for CICS work.
- o Limit the number of concurrent CICS users by maxtask or transaction class controls.
- o Adjust priorities for certain CICS transactions.

DEFINITION: This exception is detected when the hourly service unit consumption exceeds the installation-defined limits.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04103  
** HOURLY CICS CAPACITY LIMIT OBJECTIVE EXCEEDED  
*;  
IF CSYSERVU > service-units
```

```
THEN DO;  
EXCCODE='04103'; SEVERITY='C'; MGMTAREA='PERFORMANCE';  
EXCDISC1='HOURLY CICS CAPACITY LIMIT OBJECTIVE EXCEEDED';  
EXCDISC2='UPTIME(HH:MM:SS)= '  
    || PUT(CSYUPTM,TIME.) ||  
    ', AVG USERS=' || PUT(CSYAVUSR,3.) ||  
    ', SUS=' || PUT(CSYSERVU,8.);  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the value used for service units according to the following convention:

service-units - The number of service units. A threshold of 12000000 service units appears as:

```
CSYSERVU > 12000000
```

04104: Hourly CICS Paging Rate Limit Exceeded

FILE: CICS System Activity File
 SAS FILE NAME: DAYS.CICCSY01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies those hours during which the CICS storage paging exceeded the installation-defined limit.

RATIONALE: An installation may set a limit based on the CPU power and configuration, which represents the maximum hourly paging activity that can be maintained and still meet system service objectives.

DEFINITION: This exception is detected when the hourly CICS paging exceeds the installation-defined value.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;
*
** 04104
** HOURLY CICS PAGING RATE LIMIT EXCEEDED
*;
IF (CSYPAGIN + CSYPAGOT) > page-count
THEN DO;
EXCCODE='04104'; SEVERITY='C'; MGMTAREA='PERFORMANCE';
EXCDISC1='CICS PAGING RATE LIMIT EXCEEDED';
EXCDISC2='PAGE INS = ' || PUT(CSYPAGIN,8.) ||
        'PAGE OUTS=' || PUT(CSYPAGOT,8.);
LINK HIT;
END;
```

THRESHOLD MODIFICATION: Modify the value used for page-count according to the following convention:

page-count - The total number of paging operations to be executed within one hour without system degradation.

(CSYPAGIN + CSYPAGOT) > 12000

04105: Hourly CICS Short Service Objective Missed

FILE: CICS System Activity File
SAS FILE NAME: DAYS.CICCSY01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those hours when CICS users did not receive the installation-defined short response service objective.

RATIONALE: An installation can set a service objective for all users as a percent of all short responses which will be completed within a certain time. Setting the service objective this way recognizes the importance of consistency of response, which is not reflected in an average.

DEFINITION: This exception is detected when average user short response for an hour falls below the installation's objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04105  
** HOURLY CICS SHORT SERVICE OBJECTIVE MISSED  
*;  
SECONDS=seconds;  
OBJECT=percent;  
EVENTS=number-of-responses;  
_PCTCICR  
PERCENT=SPCT;  
IF (PERCENT + .5) < OBJECT  
AND CSYSTRN > EVENTS  
THEN DO;  
EXCCODE='04105'; SEVERITY='C'; MGMTAREA='SERVICE';  
EXCDESC1='HOURLY CICS SHORT SERVICE OBJECTIVE MISSED';  
EXCDESC2='OBJCTV ' || PUT(OBJECT,3.) || '% WAS ' ||  
        PUT (PERCENT,3.) || '% IN ' || PUT (SECONDS,3.)  
        || ' SECONDS';  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the values of seconds, percent and number of responses to define the installation's service objective (the percent of responses within a certain number of seconds), according to the following conventions:

seconds - The number of seconds in the service objective.
An objective of five seconds appears as:

SECONDS = 5;

percent - The target percentage that is associated with the seconds given above. A target of 90% appears as:

OBJECT = 90;

Together, these two statements define an objective of 90% within 5 seconds.

number-of-responses - The minimum number of short response events for which this exception is to be considered. A minimum of 20 appears as:

EVENTS = 20;

04106: Hourly CICS Medium Service Objective Missed

FILE: CICS System Activity File
SAS FILE NAME: DAYS.CICCSY01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those hours when CICS users did not receive the installation-defined medium response service objective.

RATIONALE: An installation can set a service objective for all users as a percent of all medium responses that will be completed within a certain time. Setting the service objective this way recognizes the importance of consistency of response, which is not reflected in an average.

DEFINITION: This exception is detected when average user medium response for an hour falls below the installation's objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04106  
** HOURLY CICS MEDIUM SERVICE OBJECTIVE MISSED  
*;  
SECONDS=seconds;  
OBJECT=percent;  
EVENTS=number-of-responses;  
_PCTCICR  
PERCENT=MPCT;  
IF (PERCENT + .5) < OBJECT  
AND CSYMTRN > EVENTS  
THEN DO;  
EXCCODE='04106'; SEVERITY='C'; MGMTAREA='SERVICE';  
EXCDESC1='HOURLY CICS MEDIUM SERVICE OBJECTIVE MISSED';  
EXCDESC2='OBJCTV ' || PUT(OBJECT,3.) || '% WAS ' ||  
        PUT (PERCENT,3.) || '% IN ' || PUT (SECONDS,3.)  
        || ' SECONDS';  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the values of seconds, percent, and number of responses to define the installation's service objective (the percent of responses within a certain number of seconds) according to the following conventions:

seconds - The number of seconds in the service objective.
An objective of five seconds appears as:

SECONDS = 5;

percent - The target percentage that is associated with the seconds given above. A target of 90% appears as:

OBJECT = 90;

Together, these two statements define an objective of 90% within 5 seconds.

number-of-responses - The minimum number of medium response events for which this exception is to be considered. A minimum of 20 appears as:

EVENTS = 20;

04107: Hourly CICS Long Service Objective Missed

FILE: CICS System Activity File
SAS FILE NAME: DAYS.CICCSY01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those hours when CICS users did not receive the installation-defined long response service objective.

RATIONALE: An installation can set a service objective for all users as a percent of all long responses that will be completed within a certain time. Setting the service objective this way recognizes the importance of consistency of response, which is not reflected in an average.

DEFINITION: This exception is detected when average user long response for an hour falls below the installation's objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04107  
** HOURLY CICS LONG SERVICE OBJECTIVE MISSED  
*;  
SECONDS=seconds;  
OBJECT=percent;  
EVENTS=number-of-responses;  
_PCTCICR  
PERCENT=LPCT;  
IF (PERCENT + .5) < OBJECT  
AND CSYLTRN > EVENTS  
THEN DO;  
EXCCODE='04107'; SEVERITY='W'; MGMTAREA='SERVICE';  
EXCDESC1='HOURLY CICS LONG SERVICE OBJECTIVE MISSED';  
EXCDESC2='OBJCTV ' || PUT(OBJECT,3.) || '% WAS ' ||  
        PUT (PERCENT,3.) || '% IN ' || PUT (SECONDS,3.)  
        || ' SECONDS';  
LINK HIT;  
END;
```


THRESHOLD MODIFICATION: Modify the values of seconds, percent, and number of responses to define the installation's service objective (the percent of responses within a certain number of seconds) according to the following conventions:

seconds - The number of seconds in the service objective.
An objective of five seconds appears as:

SECONDS = 5;

percent - The target percentage that is associated with the seconds given above. A target of 90% appears as:

OBJECT = 90;

Together, these two statements define an objective of 90% within 5 seconds.

number-of-responses - The minimum number of long response events for which this exception is to be considered.
A minimum of 20 appears as:

EVENTS = 20

04108: Hourly CICS Total Service Objective Missed

FILE: CICS System Activity File
SAS FILE NAME: DAYS.CICCSY01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those hours when CICS users did not receive the installation-defined total response service objective.

RATIONALE: An installation can set a service objective for users as a percent of all responses that will be completed within a certain time. Setting the service objective this way recognizes the importance of consistency of response, which is not reflected in an average.

DEFINITION: This exception is detected when average user response for an hour falls below the installation's objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04108  
** HOURLY CICS TOTAL SERVICE OBJECTIVE MISSED  
*;  
SECONDS=seconds;  
OBJECT=percent;  
EVENTS=number-of-responses;  
_PCTCICR  
PERCENT=TPCT;  
IF (PERCENT + .5) < OBJECT  
AND CSYTRANS > EVENTS  
THEN DO;  
EXCCODE='04108'; SEVERITY='C'; MGMTAREA='SERVICE';  
EXCDDESC1='HOURLY CICS USER TOTAL SERVICE OBJECTIVE MISSED';  
EXCDDESC2='OBJCTV ' || PUT(OBJECT,3.) || '% WAS ' ||  
          PUT (PERCENT,3.) || '% IN ' || PUT (SECONDS,3.)  
          || ' SECONDS';  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the values of seconds, percent, and number of responses to define the installation's service objective (the percent of responses within a certain number of seconds) according to the following conventions:

seconds - The number of seconds in the service objective.
An objective of five seconds appears as:

SECONDS = 5;

percent - The target percentage that is associated with the seconds given above. A target of 90% appears as:

OBJECT = 90;

Together, these two statements define an objective of 90% within 5 seconds.

number-of-responses - The minimum number of total response events for which this exception is to be considered. A minimum of 20 appears as:

EVENTS = 20;

04109: Hourly CICS Availability Objective Missed

FILE: CICS System Activity File
SAS FILE NAME: DAYS.CICCSY01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
MANAGEMENT AREA: Availability (MGMTAREA='AVAILABILITY')

PURPOSE: Identifies those hours during which CICS was not available.

RATIONALE: An installation can monitor those times when CICS was not available to ensure that users are receiving adequate availability.

DEFINITION: This exception is detected when total CICS available time is less than the uptime specified.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04109  
** HOURLY CICS AVAILABILITY OBJECTIVE MISSED  
*;  
IF CSYUPTM < uptime  
THEN DO;  
EXCCODE='04109'; SEVERITY='C'; MGMTAREA='AVAILABILITY';  
EXCDISC1='HOURLY CICS AVAILABILITY OBJECTIVE MISSED';  
EXCDISC2='AVAILABILITY (HH:MM:SS)= ' || PUT(CSYUPTM,TIME.);  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the value of uptime according to the following convention:

uptime - The minimum amount of time during the hour that CICS was available. A threshold of 57 minutes of uptime appears as:

CSYUPTM < (0,57,00)

04110: Hourly CICS Throughput Limit Objective Exceeded

FILE: CICS System Activity File
 SAS FILE NAME: DAYS.CICCSY01
 SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
 MANAGEMENT AREA: Performance (MGMTAREA='PERFORMANCE')

PURPOSE: Identifies those hours during which the CICS command throughput exceeded the installation-defined limit.

RATIONALE: An installation may set a limit based on the CPU power and configuration that represents the maximum hourly throughput that can be maintained and still meet its service objectives.

DEFINITION: This exception is detected when the hourly CICS command count exceeds the installation-defined value.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04110  
** HOURLY CICS THROUGHPUT LIMIT EXCEEDED  
*;  
IF CSYTRANS > command-count  
THEN DO;  
EXCCODE='04110'; SEVERITY='C'; MGMTAREA='PERFORMANCE';  
EXCDISC1='CICS THROUGHPUT LIMIT OBJECTIVE EXCEEDED';  
EXCDISC2='COMMAND COUNT=' || PUT(CSYTRANS,8.);  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the value used for command count according to the following convention:

command-count - The total number of commands executed within one hour. A threshold of 12000 commands appears as:

```
CSYTRANS > 12000
```

04201: Hourly CICS User Resource Overload

FILE: CICS User Activity File
SAS FILE NAME: DAYS.CICCSU01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Warning (SEVERITY='W')
MANAGEMENT AREA: Workload (MGMTAREA='WORKLOAD')

PURPOSE: Identifies those users whose hourly workload is large enough to impact the service to other users.

RATIONALE: It is possible for a CICS user to cause degraded response for others by consuming large amounts of CPU time or by doing a large number of I/Os. At times this may be the result of a program that is in an infinite loop. If this user's workload is impacting others, it may be necessary to:

- o Review the heaviest user's work for efficiency improvements.
- o Have some users run at a lower priority.
- o Defer some of the CICS work until off-hours.
- o Alter the CICS system tuning parameters for a better handling of the user's resource drain.

DEFINITION: This exception is detected when an individual user's CPU time, I/O, or service units exceed the installation-defined limits.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation and describing the condition are stored in the source member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04201  
** HOURLY CICS USER RESOURCE OVERLOAD  
*;  
IF CSUCPUTM > cpu-time  
OR CSUEXCPS > i/o  
OR CSUSERVU > service-units  
THEN DO;  
EXCCODE='04201'; SEVERITY='W'; MGMTAREA='WORKLOAD';  
EXCDESC1='HOURLY CICS USER RESOURCE OVERLOAD';  
EXCDESC2='CPU TIME(HH:MM:SS)= '
```

```
|| PUT(CSUCPUTM,TIME.) ||  
' , EXCPS=' || PUT(CSUEXCPS,6.) ||  
' , SU=' || PUT(CSUSERVU,7.);  
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the values used for CPU time, I/O, and service units according to the following conventions:

cpu-time - The amount of CPU time is described as hours, minutes, and seconds. A threshold of 3 minutes and 20 seconds appears as:

CSUCPUTM > (0,3,20)

i/o - The number of non-terminal I/O operations. A threshold of 1000 EXCPs appears as:

CSUEXCPS > 1000

service-units - The number of service units. A threshold of 35000 service units appears as:

CSUSERVU > 35000

04202: Prime Time CICS User Total Service Objective Missed

FILE: CICS User Activity File
SAS FILE NAME: DAYS.CICCSU01
SOURCE LOCATION: prefix.MICS.USER.SOURCE(DYCICEXC)

SEVERITY: Critical (SEVERITY='C')
MANAGEMENT AREA: Service (MGMTAREA='SERVICE')

PURPOSE: Identifies those hours during prime time when a user did not receive the installation-defined total response service objective.

RATIONALE: An installation can set a service objective for users as a percent of all responses that will be completed within a certain time. Setting the service objective this way recognizes the importance of consistency of response, which is not reflected in an average.

DEFINITION: This exception is detected when an individual user's response for an hour falls below the installation's objective.

EXCEPTION STATEMENTS: The SAS statements identifying the exception situation are stored in the member named in SOURCE LOCATION and are described below.

```
*****;  
*  
** 04202  
** PRIME TIME CICS USER TOTAL SERVICE OBJECTIVE MISSED  
*;  
SECONDS=seconds;  
OBJECT=percent;  
EVENTS=number-of-responses;  
_PCTCICR  
PERCENT=TPCT;  
IF (PERCENT + .5) < OBJECT  
AND CSUTRANS > EVENTS  
AND (DAYNAME='MON' OR DAYNAME='TUE' OR DAYNAME='WED'  
OR DAYNAME='THU' OR DAYNAME='FRI')  
AND (HOUR >= hh1 AND HOUR <= hh2)  
THEN DO;  
EXCCODE='04202'; SEVERITY='C'; MGMTAREA='SERVICE';  
EXCDESC1=  
'PRIME TIME CICS USER TOTAL SERVICE OBJECTIVE MISSED';  
EXCDESC2='OBJECTV ' || PUT(OBJECT,3.) || '% WAS ' ||  
PUT (PERCENT,3.) || '% IN ' || PUT (SECONDS,3.)  
|| ' SECONDS';
```



```
LINK HIT;  
END;
```

THRESHOLD MODIFICATION: Modify the values shown below to define the installation's service objective (the percent of responses within a certain number of seconds) according to the following conventions:

seconds - The number of seconds in the service objective.
An objective of five seconds appears as:

```
SECONDS = 5;
```

percent - The target percentage that is associated with the seconds given above. A target of 90% appears as:

```
OBJECT = 90;
```

Together, these two statements define an objective of 90% within 5 seconds.

number-of-responses - The minimum number of total response events for which this exception is to be considered appears as:

```
EVENTS = 1000;
```

hh1 and hh2 - The hour range and days of the week to be included in the exception. The hour of the day must be within the hh1 through hh2 range. Prime time specified as 8 a.m. through 6:59 p.m. (18:59) appears as:

```
AND ( HOUR >= 8 AND HOUR <= 18 )
```

The days of the week are represented by the first three characters of their names. They must be enclosed in quotes, separated with ORs. Prime time specified as MONDAY through FRIDAY appears as:

```
AND ( DAYNAME='MON' OR DAYNAME='TUE' OR  
      DAYNAME='WED' OR DAYNAME='THU' OR  
      DAYNAME='FRI' )
```


Chapter 5: FILES

This chapter identifies each file in the CICS and CICS Transaction Gateway (CTG) Information Areas and describes its level of summarization and data sequencing as the files appear in the applicable timespan.

The files maintained in the CICS Information Area are:

- o CICCAC - CICS User Application Count File
- o CICCAU - CICS Application Unit Activity File
- o CICCDC - CICS Dictionary File
- o CICCIN - CICS Incident File
- o CICCFS - CICS File/DBD Activity File
- o CICCST - CICS User Activity File
- o CICCST - CICS System Activity File
- o CICCST - CICS MRO Activity File

The files maintained in the CTG Information Area are:

- o CTGGSA - Gateway Server Activity File

The table shown in Figure 5-1 identifies the individual database files and categorizes them by the information area to which they belong. The following information is provided for each file in the information area:

XDWMYT - This entry defines the timespans in which the file is supported. A nonblank indicates that the file is supported. A period (.) indicates that the file is not supported. The timespans indicated by XDWMYT are as follows:

X - DETAIL
D - DAYS
W - WEEKS
M - MONTHS
Y - YEARS
T - TABLES AREA

File - The SAS data set name used to access this file.

File Name - The descriptive label for the file.

CICS Activity			Date Generated:
Information Area			Tue, May 12, 2009
XDWMYT	File	File Name	
.....	CICCAC	CICS USER APPLICATION COUNT FILE	
X.....	CICCDC	CICS DICTIONARY FILE	
X.....	CICCCIN	CICS INCIDENT FILE	
.....	CICCMR	CICS MRO ACTIVITY FILE	
.....	CICCSF	CICS FILE/DBD ACTIVITY FILE	
X.....	CICCSX	CICS SYSTEM MONITOR WORK FILE	
X.....	CICCTD	TMON/CICS DSA WORK FILE	
.D.MY.	CICCSU	CICS USER ACTIVITY FILE	
.....	CICCAU	CICS APPLICATION UNIT ACTIVITY FILE	
XDWMY.	CICCSY	CICS SYSTEM ACTIVITY FILE	

Figure 5-1. CICS Files' Active Timespans

This section contains the following topics:

- [5.1 Data Element Naming Conventions](#) (see page 445)
- [5.2 CICS Information Area Files](#) (see page 446)
- [5.3 CTG Information Area Files](#) (see page 503)

5.1 Data Element Naming Conventions

The data elements in the CA MICS Data Base files follow naming conventions that depend on whether they are standard or common data elements. Standard data elements use the first three characters of their name to identify the file in which they are defined. The following chart lists the three-character prefixes with which the CICS standard data element names begin and the files in which they are contained.

File	File Name	Prefix
CICCAC	CICS User Application Count File	CAC
CICCAU	CICS Application Unit Activity File	CAU
CICCDC	CICS Dictionary File	CDC
CICCIN	CICS Incident File	CIN
CICCSF	CICS File/DBD Activity File	CSF
CICCSU	CICS User Activity File	CSU
CICCSY	CICS System Activity File	CSY
CICCMR	CICS MRO Activity File	CMR

Common data elements do not use a data element prefix. They have a common definition across data base information areas or across files within an information area. You will find common data elements listed in the Sequence/Summary Data Elements and Common Data Elements sections of the Data Elements List that accompanies each of the file descriptions in this chapter.

5.2 CICS Information Area Files

This section identifies each file in the CICS Information Area and defines its level of summarization and data sequencing as the files appear in the applicable timespan levels. It also lists the data elements contained in each file.

The files are described in the following sections:

- 1 - CICS User Application Count File (CICCAC)
- 2 - CICS Application Unit Activity File (CICCAU)
- 3 - CICS Dictionary File (CICCDC)
- 4 - CICS Incident File (CICGIN)
- 5 - CICS File/DBD Activity File (CICCSF)
- 6 - CICS User Activity File (CICCSU)
- 7 - CICS System Activity File (CICCSY)
- 8 - CICS MRO Activity File (CICCMR)

5.2.1 CICS User Application Count File (CICCAC)

The CICS User Application Count File contains data quantifying the resources consumed by each user executing each CICS transaction type. Access measures are provided. This file is derived from the detailed transaction records from CMF or ASG-TMON.

If you choose to activate this file, which is shipped inactive, follow the instructions in section 10.1 for altering a file's active timespans.

If you use CA MICS Accounting and Chargeback to perform transaction accounting, you must activate the CICCAC file.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

- 1 - CICCAC File Organization
- 2 - CICCAC Data Elements List
- 3 - CICCAC Usage Considerations
- 4 - CICCAC Retrieval Examples

5.2.1.1 CICCAC File Organization

The table below identifies data elements by which the file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in a timespan. At the DETAIL level, data is sequenced but not summarized.

NOTE: The timespans in which a file is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

Timespan	Level of Data Granularity
DETAIL	N/A
DAYS	N/A
WEEKS	N/A
MONTHS	N/A
YEARS	N/A
TABLES	N/A

Generation Date: Tue, May 12, 2009

NOTE: This file was generated with ESSENTIAL=ALL option in effect. All data elements defined in the file are generated.

NOTE: This file was generated with DERIVED=DEFAult option in effect. Whether data elements are kept on the file on auxiliary storage or not is controlled by the complex definition of the DERIVED option.

Figure 5-2. CICCAC Timespan Granularity Chart

5.2.1.2 CICCAC Data Elements List

The table below identifies data elements contained in this file. The entries for each data element are:

TIMESPAN: Defines the timespans in which the data element is supported. The timespans are indicated by the letters "XDWMYT" as follows:

- X - DETAIL
- D - DAYS
- W - WEEKS
- M - MONTHS
- Y - YEARS
- T - TABLES AREA
- . - File is not supported

The timespan field also indicates Essential Elements with the letter E, if applicable.

DATA ELEMENT: The data element name.

DATA ELEMENT DESCRIPTION: The data element's long name.

The timespans in which a data element is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

GENERATION DATE: Tue, May 12, 2009

Note: Essential data elements are identified by an "E" under the Timespan asterisk (*) column.

Time- Span *	Data Element	Data Element Description (LABEL)
-----	-----	-----

Sequence/Summary Data Elements

.....E	CICACT1	- COST CENTER
.....E	CICACT2	- TRANSACTION ID
.....E	CICACT3	- USER ID
.....E	CICAPU	- CICS Application Unit ID
.....E	CICSID	- CICS System Identification
.....E	DAY	- Day of Month
.....E	HOURL	- Hour of Day


```
.....E MONTH   - Month of Year
.....E SYSID    - System Identifier
.....E WEEK     - Week of Year
.....E YEAR     - Year of Century
.....E ZONE     - Time Zone
```

Common Data Elements

```
.....E CICSsver - CICS Monitor Version Number
.....E CICSREL  - CICS Release Number
.....E DAYNAME  - Name of Day of Week
.....E ENDTS    - End Time Stamp
.....E MICSVER  - CA MICS Version Number
.....E STARTTS  - Start Time Stamp
```

Accumulated Data Elements

```
..... CACABEND - Abnormal Termination Occurrences
.....E CACCOST  - User Cost Field
.....E CACIMSGS - Total Input Messages Issued
.....E CACOMSGS - Total Output Messages Issued
.....E CACTRANS - Number Of Transaction Occurrences
```

5.2.1.3 CICCAC Usage Considerations

This section identifies some special considerations or techniques related to using the CICCAC File.

1. To activate the CICCAC file at the DETAIL timespan, refer to the instructions in section 10.1.4 of this guide.
2. The ENDTS and STARTTS, when appearing in the DAYS, WEEKS, MONTHS, or YEARS timespans, bound the span of time over which the data has been summarized, with STARTTS being the lowest date and time, and ENDTS the highest date and time for the data summarized.
3. The CICCAC data elements are renamed from the transaction DETAIL work file (CICCSW) created during daily input processing. A rename macro, generated at component generation time based on GENLIB definitions, is used to create the CAC data elements from the work file. This type of file creation does not affect the way you access the data -- the standard CA MICS summarization macros are available for reporting purposes.

5.2.1.4 CICCAC Retrieval Examples

In the examples, a SAS macro variable is used to specify the DDname part of the CA MICS file name. These macro variables are a standard part of CA MICS and are available for all files. The macro variable name has the form &diiiit, where d is the database identifier, iii is the information area name, and t is the timespan. For the examples, a database identifier of P is used. The identifier is installation dependent, so you should find out what the identifiers are at your installation.

1. Generate a horizontal bar graph of application executions by the first level of user identification.

```
PROC CHART DATA=&PCICD..CICCAC01;  
HBAR CICTACT1 / SUMVAR=CACTTRANS DESCENDING;  
RUN;
```

2. Generate a frequency table of the occurrence of ABENDING applications by transaction group, where transaction group is identified by the second-level user identifier. Use month-to-date data.

```
PROC FREQ DATA=&PCICM..CICCAC00;  
TABLES CICTACT2 / NOROW NOCOL; WEIGHT CACABEND;  
RUN;
```

5.2.2 CICS Application Unit Activity File (CICCAU)

The CICS Application Unit Activity File contains data that quantifies the resources consumed by each application unit within CICS. Service, load, access, and performance measures are provided. This file is derived from the detailed transaction records from CMF or ASG-TMON.

If you choose to activate this file, which is shipped inactive, follow the instructions in section 10.1 for altering a file's active timespans.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

- 1 - CICCAU File Organization
- 2 - CICCAU Data Elements List
- 3 - CICCAU Usage Considerations
- 4 - CICCAU Retrieval Examples

5.2.2.1 CICCAU File Organization

The table below identifies data elements by which the file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in a timespan. At the DETAIL level, data is sequenced but not summarized.

NOTE: The timespans in which a file is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

Timespan	Level of Data Granularity
DETAIL	N/A
DAYS	N/A
WEEKS	N/A
MONTHS	N/A
YEARS	N/A
TABLES	N/A

Generation Date: Tue, May 12, 2009

NOTE: This file was generated with ESSENTIAL=ALL option in effect. All data elements defined in the file are generated.

NOTE: This file was generated with DERIVED=DEFAult option in effect. Whether data elements are kept on the file on auxiliary storage or not is controlled by the complex definition of the DERIVED option.

Figure 5-3. CICCAU Timespan Granularity Chart

5.2.2.2 CICCAU Data Elements List

The table below identifies data elements contained in this file. The entries for each data element are:

TIMESPAN: Defines the timespans in which the data element is supported. The timespans are indicated by the letters "XDWMYT" as follows:

- X - DETAIL
- D - DAYS
- W - WEEKS
- M - MONTHS
- Y - YEARS
- T - TABLES AREA
- . - File is not supported

The timespan field also indicates Essential Elements with the letter E, if applicable.

DATA ELEMENT: The data element name.

DATA ELEMENT DESCRIPTION: The data element's long name.

The timespans in which a data element is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

GENERATION DATE: Tue, Jul 30, 2013

Note: Essential data elements are identified by an "E" under the Timespan asterisk (*) column.

Time- Span *	Data Element	Data Element Description (LABEL)
-----	-----	-----

Sequence/Summary Data Elements

.....E	CICAPU	- CICS Application Unit ID
.....E	CICSID	- CICS System Identification
.....E	DAY	- Day of Month
.....E	HOUR	- Hour of Day
.....E	MONTH	- Month of Year
.....E	SYSID	- System Identifier

.....E WEEK - Week of Year
.....E YEAR - Year of Century
.....E ZONE - Time Zone

Common Data Elements

.....E CICNODST - Number of Distribution Values
.....E CICRVAL1 - Response Distribution Limits 1
.....E CICRVAL2 - Response Distribution Limits 2
.....E CICRVAL3 - Response Distribution Limits 3
.....E CICRVAL4 - Response Distribution Limits 4
.....E CICRVAL5 - Response Distribution Limits 5
.....E CICRVAL6 - Response Distribution Limits 6
.....E CICRVAL7 - Response Distribution Limits 7
.....E CICRVAL8 - Response Distribution Limits 8
.....E CICSMMVER - CICS Monitor Version Number
.....E CICSREL - CICS Release Number
.....E DAYNAME - Name of Day of Week
.....E ENDTS - End Time Stamp
.....E PROGRAM - Program Name
.....E STARTTS - Start Time Stamp
.....E TRANETS - Transaction End Time Stamp
.....E TRANSTS - Transaction Start Time Stamp

Retained Data Elements

.....E CAUFACTY - Transaction Facility Type
.....E CAUJOB - CICS JOB Name
.....E CAURTYPE - Transaction Record Type
.....E CAUTLUN - Terminal VTAM LUNAME
..... CAUTSKID - Task Identification Number

Accumulated Data Elements

.....E CAUABEND - Abnormal Termination Occurrences
.....E CAUCDST1 - Count Conv. Responses Within Limit 1
.....E CAUCDST2 - Count Conv. Responses Within Limit 2
.....E CAUCDST3 - Count Conv. Responses Within Limit 3
.....E CAUCDST4 - Count Conv. Responses Within Limit 4
.....E CAUCDST5 - Count Conv. Responses Within Limit 5
.....E CAUCDST6 - Count Conv. Responses Within Limit 6
.....E CAUCDST7 - Count Conv. Responses Within Limit 7
.....E CAUCDST8 - Count Conv. Responses Within Limit 8
..... CAUCOST - Processing Charges
..... CAUCPUNI - Instructions Executed
.....E CAUCPUTM - Task CPU Time
.....E CAUCRSTM - Conv. Response Time Total
.....E CAUCTRN - Conv. Transactions Processed
.....E CAUDAMSC - Damaged Storage Chain

.....E CAUDLICC - Number of CICS DL/I Calls
.....E CAUDLIWT - Elapsed Time For CICS DL/I Calls
.....E CAUDSWAT - Wait for Dispatch (Queue Time)
..... CAUDS1WT - Wait for First Dispatch Time
..... CAUEQSUS - Enqueue Suspend Wait Time
.....E CAUERSTM - Excessive Response Time Total
.....E CAUETRNL - Excessive Transactions Processed
.....E CAUFCOPS - Number of CICS File Control Calls
.....E CAUFCTOT - Total File Requests
.....E CAUFCWAT - Wait Time For CICS File Control
.....E CAUIMSGS - Total Input Messages Issued
.....E CAUINCH - Input Message Character Traffic
.....E CAUIRWAT - Wait Time For MRO/ISC Communications
.....E CAUJCOPS - Number of CICS Journal Control Calls
.....E CAUJCWAT - Wait Time For CICS Journal Control
.....E CAULDST1 - Count Long Responses Within Limit 1
.....E CAULDST2 - Count Long Responses Within Limit 2
.....E CAULDST3 - Count Long Responses Within Limit 3
.....E CAULDST4 - Count Long Responses Within Limit 4
.....E CAULDST5 - Count Long Responses Within Limit 5
.....E CAULDST6 - Count Long Responses Within Limit 6
.....E CAULDST7 - Count Long Responses Within Limit 7
.....E CAULDST8 - Count Long Responses Within Limit 8
.....E CAULRSTM - Long Response Time Total
.....E CAULTRNL - Long Transactions Processed
.....E CAUMDST1 - Count Medium Responses Within Limit 1
.....E CAUMDST2 - Count Medium Responses Within Limit 2
.....E CAUMDST3 - Count Medium Responses Within Limit 3
.....E CAUMDST4 - Count Medium Responses Within Limit 4
.....E CAUMDST5 - Count Medium Responses Within Limit 5
.....E CAUMDST6 - Count Medium Responses Within Limit 6
.....E CAUMDST7 - Count Medium Responses Within Limit 7
.....E CAUMDST8 - Count Medium Responses Within Limit 8
.....E CAUMEMRC - Memory Usage Running Count
.....E CAUMRSTM - Medium Response Time Total
.....E CAUMTRNL - Medium Transactions Processed
.....E CAUMXTSK - Max Tasking Condition
.....E CAUNZZTM - Normalized zIIP/zAAP CPU Time
.....E CAUOCPTC - User Task Spec Proc Eligible CPU Count
.....E CAUOMSGS - Total Output Messages Issued
.....E CAUOUTCH - Output Message Character Traffic
..... CAUPAGIN - Page Ins
..... CAUPAGOT - Page Outs
.....E CAUPCOPS - Number of CICS Program Control Calls
..... CAUPCWCT - Program Fetch Wait Count
.....E CAUPETTM - Task Dispatch Time
.....E CAURESTM - Transaction Residency Time
.....E CAURSPCT - Total Number of Response Counts
.....E CAUSCOPS - Number of CICS Storage Control Calls

```

.....E CAUSDST1 - Count Short Responses Within Limit 1
.....E CAUSDST2 - Count Short Responses Within Limit 2
.....E CAUSDST3 - Count Short Responses Within Limit 3
.....E CAUSDST4 - Count Short Responses Within Limit 4
.....E CAUSDST5 - Count Short Responses Within Limit 5
.....E CAUSDST6 - Count Short Responses Within Limit 6
.....E CAUSDST7 - Count Short Responses Within Limit 7
.....E CAUSDST8 - Count Short Responses Within Limit 8
..... CAUSIOCT - Access Method Calls (sio Count)
.....E CAUSOS - Short On Storage Condition
.....E CAUSQLTM - Total Elapsed Time in SQL Calls
.....E CAUSQLT0 - Total Number of SQL Calls
.....E CAUSRSTM - Short Response Time Total
.....E CAUSTRN - Short Transactions Processed
.....E CAUSUSDT - Unidentified Suspend Wait Time
.....E CAUSUSTM - Task Suspend (Wait) Time
..... CAUTCBTM - TCB CPU Time
.....E CAUTCPTC - User Task Standard Processor CPU Count
.....E CAUTCWAT - Wait Time For CICS Terminal Control
.....E CAUTDOPS - Number of CICS Transient Data Calls
.....E CAUDST1 - Count Responses Within Limit 1
.....E CAUDST2 - Count Responses Within Limit 2
.....E CAUDST3 - Count Responses Within Limit 3
.....E CAUDST4 - Count Responses Within Limit 4
.....E CAUDST5 - Count Responses Within Limit 5
.....E CAUDST6 - Count Responses Within Limit 6
.....E CAUDST7 - Count Responses Within Limit 7
.....E CAUDST8 - Count Responses Within Limit 8
.....E CAUTLOPS - Number of CICS Calls (Total)
.....E CAUTLWAT - Wait Time For CICS (Total)
.....E CAUTRANS - Transactions Processed
.....E CAUTRSTM - Transaction Response Time Total
..... CAUTSIWT - Temp Storage (Aux+Main) Input Wait Time
.....E CAUTSOPS - Number of CICS Temporary Storage Calls
.....E CAUTSWAT - Wait Time For CICS Temp Storage
.....E CAUWAITM - Transaction Wait Time
.....E CAUXCSTM - Task CPU Time Without zIIP/zAAP Eligible
.....E CAUXTSTM - TCB CPU Time Without zIIP/zAAP Eligible
.....E CAUZZCTM - zIIP/zAAP Eligible CPU Time on CP

```

Maximum Data Elements

```

..... CAUCLKOP - Number Of Clocks Generated
..... CAUHISTG - Hi-water Terminal And User Memory
..... CAUMXCTM - Max Conv. Response Time
..... CAUMXETM - Max Excessive Response Time
..... CAUMXLTM - Max Long Response Time
.....E CAUMXMEM - Max Transaction Memory Used
..... CAUMXMTM - Max Medium Response Time

```


..... CAUMXOPS - Maximum Number of CICS Calls
..... CAUMXSTM - Max Short Response Time
..... CAUMXTRT - Max Transaction CPU Time
..... CAUMXTTM - Max Response Time
..... CAUMXWAT - Maximum Wait Time For CICS
..... CAUNTR0P - Number Of Counters Generated
.....E CAU64HWM - USER & CICS 64 Pool High-water Mark

Derived Data Elements

..... CAUAVCTM - Avg Conv. Response Time
.....E CAUAVINC - Average Input Character Traffic
..... CAUAVLTM - Avg Long Response Time
.....E CAUAVMEM - Average Transaction Memory Used
..... CAUAVMTM - Avg Medium Response Time
.....E CAUAVOPS - Average Number of CICS Calls
.....E CAUAVOTC - Average Output Character Traffic
..... CAUAVSTM - Avg Short Response Time
.....E CAUAVTRT - Average Transaction CPU Time
.....E CAUAVTTM - Avg Response Time
..... CAUAVWAT - Average Wait Time For CICS Functions
..... CAUPCTR1 - Cumm Pct Resp Within Limit 1
..... CAUPCTR2 - Cumm Pct Resp Within Limit 2
..... CAUPCTR3 - Cumm Pct Resp Within Limit 3
..... CAUPCTR4 - Cumm Pct Resp Within Limit 4
..... CAUPCTR5 - Cumm Pct Resp Within Limit 5
..... CAUPCTR6 - Cumm Pct Resp Within Limit 6
..... CAUPCTR7 - Cumm Pct Resp Within Limit 7

5.2.2.3 CICCAU Usage Considerations

This section identifies some special considerations or techniques related to using the CICCAU File.

1. To activate the CICCAU file at the DETAIL timespan, refer to the instructions in section 10.1.4 of this guide.
2. The following data elements are defined within the CICGENIN member but are shipped as inactive. They are reserved for your definitions.

CAUCONTM - Terminal Connect Time
CAUEXCPS - I/O (EXCPS) Generated
CAUSERVU - Service Units
CAUSRMTR - SRM Ended Transactions
CAUSRU - System Resource Units

3. The following data elements are dependent upon the setting of specific options within the various members in sharedprefix.MICS.PARMS and prefix.MICS.PARMS. This dependency is outlined below.

prefix.MICS.PARMS(SYSID)
CAUCPUNI - Instructions Executed

prefix.MICS.PARMS(CICOPS) -- parameter RESP
CICRVAL1 - CICRVAL7 - Response Distribution Limits

sharedprefix.MICS.PARMS(CICAURT)
CICAPU - Application Unit Identifier

4. For transactions that are classified as excessive (TRANSTYPE="X"), the transaction counts and response times are excluded from the following data elements:

CAUTRANS - Transaction Processed
CAUTRSTM - Transaction Response Time Total
CAURESTM - Transaction Residency Time
CAUAVTTM - Avg Response Time
CAUMXTTM - Max Response Time

The excessive counts are available in the following data elements:

CAUETRN - Excessive Transactions Processed
CAUERSTM - Excessive Response Time Total
CAUMXETM - Max Excessive Response Time

5. The ENDTS and STARTTS, when appearing in the DAYS or MONTHS timespans, bound the span of time over which the data has been summarized, with STARTTS being the lowest date and time, and ENDTS the highest date and time for the data summarized.

5.2.2.4 CICCAU Retrieval Examples

In the examples, a SAS macro variable is used to specify the DDname part of the CA MICS file name. These macro variables are a standard part of CA MICS and are available for all files. The macro variable name has the form &diiiit, where d is the database identifier, iii is the information area name, and t is the timespan. For the examples, a database identifier of P is used. The identifier is installation dependent, so you should find out what the identifiers are at your installation.

1. Generate a horizontal bar graph of transaction CPU time used yesterday by application unit.

```
PROC CHART DATA=&PCICD..CICCAU01;
HBAR CICAPU / SUMVAR=CAUCPUTM DESCENDING;
RUN;
```

2. Generate a summary report showing workload and performance information by CICSID and application unit for the last three days.

```
%LET BY = CICSID CICAPU MONTH YEAR DAY;
%LET BREAK = DAY;
DATA CICCAU(KEEP=%CAUKEEP(OP=FILEOPTS,TS=DAYS));
SET &PCICD..CICCAU03
    &PCICD..CICCAU02
    &PCICD..CICCAU01;
RUN;
PROC SORT DATA=CICCAU;
BY &BY;
RUN;

DATA SUMCAU(KEEP=CICSID CICAPUB MONTH YEAR DAY
    CAUCPUTM CAUSIOCT CAUTLOPS CAUTRANS CAUAVTTM
    CAUSOS CAUMXTSK CAUPAGIN CAUFCWAT CAUTLWAT);
SET CICCAU;
BY &BY;
IF FIRST.CICAPU THEN CICAPUB=CICAPU;
ELSE CICAPUB=' ';
%CAUSUM(SUMBY=&BY,SUMBREAK=&BREAK,OUT=SUMCAU,
    DERV=YES);
RUN;

PROC PRINT DATA=SUMCAU N; BY CICSID;
PAGEBY CICSID;
ID CICAPUB;
```

```
VAR MONTH YEAR DAY  
CAUCPUTM CAUTRANS CAUTLOPS CAUSIOCT CAUAVTTM  
CAUSOS CAUMXTSK CAUPAGIN CAUFCWAT CAUTLWAT;  
RUN;
```

5.2.3 CICS Dictionary File (CICCDC)

The CICS Dictionary File contains information on each field contained in the CMF records for performance, accounting, and exception class. The dictionary records are required in order to process the CMF records.

The CICCDC File exists only in cycle 01 at the DETAIL timespan.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

- 1 - CICCDC File Organization
- 2 - CICCDC Data Elements List
- 3 - CICCDC Usage Considerations
- 4 - CICCDC Retrieval Examples

5.2.3.1 CICCDC File Organization

The table below identifies data elements by which the file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in a timespan. At the DETAIL level, data is sequenced but not summarized.

NOTE: The timespans in which a file is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

Timespan	Level of Data Granularity					
DETAIL	ORGSYSID	CDCAPPL	CICSID	CICSREL	CDCCLASS	
	CDCCONN	ENDTS				
DAYS	N/A					
WEEKS	N/A					
MONTHS	N/A					
YEARS	N/A					
TABLES	N/A					

Generation Date: Tue, May 12, 2009

NOTE: This file was generated with ESSENTIAL=ALL option in effect. All data elements defined in the file are generated.

NOTE: This file was generated with DERIVED=Default option in effect. Whether data elements are kept on the file on auxiliary storage or not is controlled by the complex definition of the DERIVED option.

Figure 5-4. CICCDC Timespan Granularity Chart

5.2.3.2 CICCDC Data Elements List

The table below identifies data elements contained in this file. The entries for each data element are:

TIMESPAN: Defines the timespans in which the data element is supported. The timespans are indicated by the letters "XDWMYT" as follows:

X - DETAIL
D - DAYS
W - WEEKS
M - MONTHS
Y - YEARS
T - TABLES AREA
. - File is not supported

The timespan field also indicates Essential Elements with the letter E, if applicable.

DATA ELEMENT: The data element name.

DATA ELEMENT DESCRIPTION: The data element's long name.

The timespans in which a data element is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

GENERATION DATE: Tue, May 12, 2009

Note: Essential data elements are identified by an "E" under the Timespan asterisk (*) column.

Time- Span *	Data Element	Data Element Description (LABEL)
-----	-----	-----

Sequence/Summary Data Elements

X....E CDCAPPL - Dictionary Record Applid
X....E CDCCLASS - Dictionary Class of Data
X....E CDCCONN - Dictionary Element Connector
X....E CICSID - CICS System Identification
X....E CICSREL - CICS Release Number
X....E ENDTS - End Time Stamp
X....E ORGSYSID - Originating System Identification

Retained Data Elements

X.....E CDCENTRY - Dictionary Element Description
X.....E CDCLREF - Dictionary Date of Last Reference

5.2.3.3 CICCDC Usage Considerations

This section identifies some special considerations or techniques related to using the CICCDC file.

1. This file exists only in cycle 01 at the DETAIL timespan. One observation is created for each field defined by the dictionary record for each CMF class and CICS region. The CICS Analyzer processes the CMF data using the most current version of the data dictionary record read from input. However, if the input does not contain data dictionary records, CMF data is processed using the dictionary records stored in the CICCDC01 file from previous DAILY runs. If CICCDC01 is empty, the CMF data is rejected until the data dictionary record is read from input.
2. This file contains data dictionary records from multiple DAILY runs. The data is deleted when the number of unreferenced days reaches the threshold defined by the CICCDCEX parameter in prefix.MICS.PARMS(CICTHRSH). For more information on this threshold definition, please refer to Section 7.3.8 in this guide.

5.2.3.4 CICCDC Retrieval Examples

There are no retrieval examples for the CICCDC File, since it is not used for reporting purposes. However, a utility program that lists the CMF data dictionary entries from the CICCDC File is shipped with the CICS Analyzer. The generated JCL is in prefix.MICS.CNTL(CICDCPT). For a description of this utility program, see Section 9.3.2 of this guide.

5.2.4 CICS Incident File (CICIN)

The CICS Incident File contains data identifying problems in the operation, performance, or integrity of the CICS system. This file is derived from detailed transaction and interval/checkpoint monitor data.

All CICS incidents recorded on this file are reported as standard-format CA MICS exceptions. Therefore, this file is maintained only in the DETAIL timespan.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

- 1 - CICIN File Organization
- 2 - CICIN Data Elements List
- 3 - CICIN Usage Considerations
- 4 - CICIN Retrieval Examples

5.2.4.1 CICCIN File Organization

The table below identifies data elements by which the file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in a timespan. At the DETAIL level, data is sequenced but not summarized.

NOTE: The timespans in which a file is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

Timespan	Level of Data Granularity					
DETAIL	SYSID DAY	CICSID HOUR	CINCODE ENDTS	YEAR	MONTH	
DAYS	N/A					
WEEKS	N/A					
MONTHS	N/A					
YEARS	N/A					
TABLES	N/A					

Generation Date: Tue, May 12, 2009

NOTE: This file was generated with ESSENTIAL=ALL option in effect. All data elements defined in the file are generated.

NOTE: This file was generated with DERIVED=Default option in effect. Whether data elements are kept on the file on auxiliary storage or not is controlled by the complex definition of the DERIVED option.

Figure 5-5. CICCIN Timespan Granularity Chart

5.2.4.2 CICCIN Data Elements List

The table below identifies data elements contained in this file. The entries for each data element are:

TIMESPAN: Defines the timespans in which the data element is supported. The timespans are indicated by the letters "XDWMYT" as follows:

X - DETAIL
D - DAYS
W - WEEKS
M - MONTHS
Y - YEARS
T - TABLES AREA
. - File is not supported

The timespan field also indicates Essential Elements with the letter E, if applicable.

DATA ELEMENT: The data element name.

DATA ELEMENT DESCRIPTION: The data element's long name.

The timespans in which a data element is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

GENERATION DATE: Tue, May 12, 2009

Note: Essential data elements are identified by an "E" under the Timespan asterisk (*) column.

Time- Span *	Data Element	Data Element Description (LABEL)
-----	-----	-----

Sequence/Summary Data Elements

X....E	CICSID	- CICS System Identification
X....E	CINCDE	- CICS Incident Identifier
X....E	DAY	- Day of Month
X....E	MONTH	- Month of Year
X....E	SYSID	- System Identifier
X....E	WEEK	- Week of Year
X....E	YEAR	- Year of Century

X.....E ZONE - Time Zone

Common Data Elements

X.....E DAYNAME - Name of Day of Week
X.....E ENDTS - End Time Stamp
X.....E HOUR - Hour of Day
X.....E MICSVER - CA MICS Version Number
X.....E OPERID - Terminal Operator Initials
X.....E STARTTS - Start Time Stamp
X.....E TERMINAL - CICS Terminal ID
X.....E TRANCODE - CICS Transaction Code
X.....E TRANETS - Transaction End Time Stamp
X.....E TRANSTS - Transaction Start Time Stamp
X.....E USERID - RACF Userid

Retained Data Elements

X.....E CINABEND - Abnormal Termination Code
X.....E CINTEXT - Description Of Incident

Accumulated Data Elements

X.....E CINCOUNT - Incident Count

5.2.4.3 CICCIN Usage Considerations

This section identifies some special considerations or techniques related to using the CICCIN File.

1. The data element STARTTS represents the monitored CICS incident start time. The data element ENDTS represents the monitored CICS incident end time.

5.2.4.4 CICCIN Retrieval Examples

In the examples, a SAS macro variable is used to specify the DDname part of the CA MICS file name. These macro variables are a standard part of CA MICS and are available for all files. The macro variable name has the form &diit, where d is the database identifier, ii is the information area name, and t is the timespan. For the examples, a database identifier of P is used. The identifier is installation dependent, so you should find out what the identifiers are at your installation.

1. Generate a horizontal bar graph of yesterday's incident counts by hour.

```
PROC CHART DATA=&PCICX..CICCIN01;  
HBAR HOUR / SUMVAR=CINCOUNT DISCRETE;  
RUN;
```

2. Generate a frequency table of the occurrence of incidents by incident code. Use month-to-date data.

```
PROC FREQ DATA=&PCICM..CICCIN00;  
TABLES CINCODE / NOROW NOCOL; WEIGHT CINCOUNT;  
RUN;
```

5.2.5 CICS File/DBD Activity File (CICCSF)

The CICS File/DBD Activity File is an optional CICS file that contains data quantifying the CICS system's file or data base usage for the DAYS, WEEKS, MONTHS, and YEARS timespans. This file is derived from the transaction file segments from Monitor data.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

- 1 - CICCSF File Organization
- 2 - CICCSF Data Elements List
- 3 - CICCSF Usage Considerations
- 4 - CICCSF Retrieval Examples

5.2.5.1 CICCSE File Organization

The table below identifies data elements by which the file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in a timespan. At the DETAIL level, data is sequenced but not summarized.

NOTE: The timespans in which a file is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

Timespan	Level of Data Granularity
DETAIL	N/A
DAYS	N/A
WEEKS	N/A
MONTHS	N/A
YEARS	N/A
TABLES	N/A

Generation Date: Tue, May 12, 2009

NOTE: This file was generated with ESSENTIAL=ALL option in effect. All data elements defined in the file are generated.

NOTE: This file was generated with DERIVED=Default option in effect. Whether data elements are kept on the file on auxiliary storage or not is controlled by the complex definition of the DERIVED option.

Figure 5-6. CICCSE Timespan Granularity Chart

5.2.5.2 CICCFS Data Elements List

The table below identifies data elements contained in this file. The entries for each data element are:

TIMESPAN: Defines the timespans in which the data element is supported. The timespans are indicated by the letters "XDWMYT" as follows:

- X - DETAIL
- D - DAYS
- W - WEEKS
- M - MONTHS
- Y - YEARS
- T - TABLES AREA
- . - File is not supported

The timespan field also indicates Essential Elements with the letter E, if applicable.

DATA ELEMENT: The data element name.

DATA ELEMENT DESCRIPTION: The data element's long name.

The timespans in which a data element is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

GENERATION DATE: Tue, May 12, 2009

Note: Essential data elements are identified by an "E" under the Timespan asterisk (*) column.

Time- Span *	Data Element	Data Element Description (LABEL)
-----	-----	-----

Sequence/Summary Data Elements

.....E	CICSID	- CICS System Identification
.....E	CSFFID	- File Designation
.....E	HOURL	- Hour of Day
.....E	MONTH	- Month of Year
.....E	SYSID	- System Identifier
.....E	WEEK	- Week of Year
.....E	YEAR	- Year of Century

.....E ZONE - Time Zone

Common Data Elements

.....E CICSVER - CICS Monitor Version Number
.....E CICSREL - CICS Release Number
.....E DAY - Day of Month
.....E DAYNAME - Name of Day of Week
.....E DEVADDR - Device Address
.....E DEVNUM - Device Number
.....E ENDTS - End Time Stamp
.....E MICSVER - CA MICS Version Number
.....E STARTTS - Start Time Stamp
.....E VOLSER - Volume Serial Number

5.2.5.3 CICCFS Usage Considerations

This section identifies some special considerations or techniques related to using the CICCFS File.

1. The CICCFS File is available only when The Monitor for CICS is the input data source. The input to this file consists of the file segments contained within The Monitor's Task Accounting Detail record.

Refer to Section 10.1.9 for information on activating this file.

2. The USRXCSF exit can be used to limit the amount of data to be summarized into the CICCFS File. This can be done using two techniques. First, code the exit to reject a file segment from further processing. Second, the element CSFFID, which is the file identifier from the file segment, can be modified to a generic name instead of retaining the actual file identifier.
3. The ENDTS and STARTTS, when appearing in the DAYS, WEEKS, MONTHS, or YEARS timespans, bound the span of time over which the data has been summarized, with STARTTS being the lowest date and time, and ENDTS the highest date and time for the data summarized.

5.2.5.4 CICCSE Retrieval Examples

In the examples, a SAS macro variable is used to specify the DDname part of the CA MICS file name. These macro variables are a standard part of CA MICS and are available for all files. The macro variable name has the form &diit, where d is the database identifier, ii is the information area name, and t is the timespan. For the examples, a database identifier of P is used. The identifier is installation dependent, so you should find out what the identifiers are at your installation.

1. Print yesterday's file usage by hour.

```
PROC PRINT DATA=&PCICD..CICCSF01;
  ID HOUR; VAR CICSID CSFFID CSFADD CSFGET
  CSFUPD CSFBROW;
  SUM CSFADD CSFGET CSFUPD CSFBROW;
RUN;
```

2. Provide a summary of file access by VOLSER from the day before yesterday.

```
%LET BY= VOLSER CSFFID;
%LET BREAK = VOLSER;

PROC SORT DATA=&PCICD..CICCSF02 OUT=CICCSF;
  BY &BY;
RUN;

DATA SUMVOL(KEEP=VOLSER FILE_TOT CSFACCCT CSFACCTM
  CSFVSMBW CSFVSMSP CSFVMSW);
  SET CICCSF; BY &BY;
  IF FIRST.CSFFID THEN FILE_NUM+1;
  IF LAST.&BREAK THEN DO;
    FILE_TOT=FILE_NUM;
    FILE_NUM=0;
  END;
  %CSFSUM(SUMBY=&BY;SMBREAK=&BREAK,OUT=SUMVOL);
RUN;

PROC PRINT DATA=SUMVOL;
  VAR VOLSER FILE_TOT CSFACCCT CSFACCTM CSFVSMBW
  CSFVMSW CSFVSMSP;
  SUM FILE_TOT CSFACCCT CSFACCTM CSFVSMBW CSFVMSW
  CSFVSMSP;
RUN;
```

5.2.6 CICS User Activity File (CICCSU)

The CICS User Activity File contains data quantifying the resources consumed by each user executing each CICS transaction type. Service, load, access, and performance measures are provided. This file is derived from the detailed transaction records from CMF or ASG-TMON.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

- 1 - CICCSU File Organization
- 2 - CICCSU Data Elements List
- 3 - CICCSU Usage Considerations
- 4 - CICCSU Retrieval Examples

5.2.6.1 CICCSU File Organization

The table below identifies data elements by which the file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in a timespan. At the DETAIL level, data is sequenced but not summarized.

NOTE: The timespans in which a file is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

+-----+-----+-----+-----+-----+-----+-----+						
Timespan		Level of Data Granularity				
+-----+-----+-----+-----+-----+-----+-----+						
	DETAIL	N/A				
	DAYS	SYSID	CICSID	CICACT1	CICACT2	CICACT3
		YEAR	MONTH	DAY	HOURL	
	WEEKS	N/A				
	MONTHS	SYSID	CICSID	CICACT1	CICACT2	CICACT3
		YEAR	MONTH	ZONE		
	YEARS	SYSID	CICSID	CICACT1	CICACT2	CICACT3
		YEAR	ZONE			
+-----+-----+-----+-----+-----+-----+-----+						
	TABLES	N/A				
+-----+-----+-----+-----+-----+-----+-----+						

Generation Date: Tue, May 12, 2009

NOTE: This file was generated with ESSENTIAL=ALL option in effect. All data elements defined in the file are generated.

NOTE: This file was generated with DERIVED=DEFAULT option in effect. Whether data elements are kept on the file on auxiliary storage or not is controlled by the complex definition of the DERIVED option.

Figure 5-7. CICCSU Timespan Granularity Chart

5.2.6.2 CICCISU Data Elements List

The table below identifies data elements contained in this file. The entries for each data element are:

TIMESPAN: Defines the timespans in which the data element is supported. The timespans are indicated by the letters "XDWMYT" as follows:

- X - DETAIL
- D - DAYS
- W - WEEKS
- M - MONTHS
- Y - YEARS
- T - TABLES AREA
- . - File is not supported

The timespan field also indicates Essential Elements with the letter E, if applicable.

DATA ELEMENT: The data element name.

DATA ELEMENT DESCRIPTION: The data element's long name.

The timespans in which a data element is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

GENERATION DATE: Tue, Jul 30, 2013

Note: Essential data elements are identified by an "E" under the Timespan asterisk (*) column.

Time- Span *	Data Element	Data Element Description (LABEL)
-----	-----	-----

Sequence/Summary Data Elements

XD.MY.E	CICACT1	- COST CENTER
XD.MY.E	CICACT2	- TRANSACTION ID
XD.MY.E	CICACT3	- USER ID
XD.MY.E	CICSID	- CICS System Identification
XD....E	DAY	- Day of Month
XD....E	HOUR	- Hour of Day

XD.M..E MONTH - Month of Year
 XD.MY.E SYSID - System Identifier
 XD....E WEEK - Week of Year
 XD.MY.E YEAR - Year of Century
 XD.MY.E ZONE - Time Zone

Common Data Elements

X..... ABNDCODE - Abend Codes
 X.....E CICAPU - CICS Application Unit ID
 X.....E CICNETNM - MRO/ISC Network Name
 .D.MY.E CICNODST - Number of Distribution Values
 .D.MY.E CICRVAL1 - Response Distribution Limits 1
 .D.MY.E CICRVAL2 - Response Distribution Limits 2
 .D.MY.E CICRVAL3 - Response Distribution Limits 3
 .D.MY.E CICRVAL4 - Response Distribution Limits 4
 .D.MY.E CICRVAL5 - Response Distribution Limits 5
 .D.MY.E CICRVAL6 - Response Distribution Limits 6
 .D.MY.E CICRVAL7 - Response Distribution Limits 7
 .D.MY.E CICRVAL8 - Response Distribution Limits 8
 XD.MY.E CICSVER - CICS Monitor Version Number
 XD.MY.E CICSREL - CICS Release Number
 X.....E CICUOWID - MRO/ISC UOWID
 XD....E DAYNAME - Name of Day of Week
 XD.MY.E ENDTS - End Time Stamp
 XD.MY.E JOB - Job Identification
 X.....E OPERID - Terminal Operator Initials
 X.....E PROGRAM - Program Name
 XD.MY.E RDRTS - Reader Time Stamp
 XD.MY.E STARTTS - Start Time Stamp
 X.....E TERMINAL - CICS Terminal ID
 X.....E TRANCODE - CICS Transaction Code
 X.....E TRANETS - Transaction End Time Stamp
 X...Y.E TRANSTS - Transaction Start Time Stamp
 X.....E TRANTYPE - Transaction Relative Longevity Code
 X.....E UOWID - Unit of Work Id
 X.....E UOWIDSEQ - Unit of Work Id Sequence Number
 X.....E USERID - RACF Userid

Retained Data Elements

XD.MY.E CSUAPPL - CICS VTAM APPLID
 X.....E CSUFACTY - Transaction Facility Type
 XD.MY.E CSUJOB - CICS JOB Name
 X.....E CSURTYPE - Transaction Record Type
 XD.MY.E CSUTLUN - Terminal VTAM LUNAME
 X.....E CSUTSKID - Task Identification Number

Accumulated Data Elements

XD.MY.E CSUABEND - Abnormal Termination Occurrences
.D.MY.E CSUCDST1 - Count Conv. Responses Within Limit 1
.D.MY.E CSUCDST2 - Count Conv. Responses Within Limit 2
.D.MY.E CSUCDST3 - Count Conv. Responses Within Limit 3
.D.MY.E CSUCDST4 - Count Conv. Responses Within Limit 4
.D.MY.E CSUCDST5 - Count Conv. Responses Within Limit 5
.D.MY.E CSUCDST6 - Count Conv. Responses Within Limit 6
.D.MY.E CSUCDST7 - Count Conv. Responses Within Limit 7
.D.MY.E CSUCDST8 - Count Conv. Responses Within Limit 8
XD.MY.E CSUCOST - Processing Charges
XD.MY.E CSUCPRTM - Task CPU Real Time
XD.MY.E CSUCPUNI - Instructions Executed
XD.MY.E CSUCPUTM - Task CPU Time
.D.MY.E CSUCRSTM - Conv. Response Time Total
.D.MY.E CSUCTRN - Conv. Transactions Processed
XD.MY.E CSUDAMSC - Damaged Storage Chain
XD.MY.E CSUDLICC - Number of CICS DL/I Calls
XD.MY.E CSUDLIPW - DL/I Pool Wait Count
XD.MY.E CSUDLITW - DL/I Thread Wait Count
XD.MY.E CSUDLIWT - Elapsed Time For CICS DL/I Calls
XD.MY.E CSUDPCPU - TCB CPU TIME USED
XD.MY. CSUDSWAT - Wait for Dispatch (Queue Time)
XD.MY.E CSUDS1WT - Wait for First Dispatch Time
XD.MY. CSUEQSUS - Enqueue Suspend Wait Time
.D.MY.E CSUERSTM - Excessive Response Time Total
.D.MY.E CSUETRN - Excessive Transactions Processed
.D....E CSUEXCPS - I/O (excps) Generated
XD.MY.E CSUFCOPS - Number of CICS File Control Calls
XD.MY.E CSUFCTOT - Total File Requests
XD.MY.E CSUFCWAT - Wait Time For CICS File Control
XD.MY.E CSUIMSGS - Total Input Messages Issued
XD.MY.E CSUINCH - Input Message Character Traffic
XD.MY.E CSUIRWAT - Wait Time For MRO/ISC Communications
XD.MY.E CSUJCOPS - Number of CICS Journal Control Calls
XD.MY.E CSUJCWAT - Wait Time For CICS Journal Control
.D.MY.E CSULDST1 - Count Long Responses Within Limit 1
.D.MY.E CSULDST2 - Count Long Responses Within Limit 2
.D.MY.E CSULDST3 - Count Long Responses Within Limit 3
.D.MY.E CSULDST4 - Count Long Responses Within Limit 4
.D.MY.E CSULDST5 - Count Long Responses Within Limit 5
.D.MY.E CSULDST6 - Count Long Responses Within Limit 6
.D.MY.E CSULDST7 - Count Long Responses Within Limit 7
.D.MY.E CSULDST8 - Count Long Responses Within Limit 8
.D.MY.E CSULRSTM - Long Response Time Total
.D.MY.E CSULTRN - Long Transactions Processed
.D.MY.E CSUMDST1 - Count Medium Responses Within Limit 1
.D.MY.E CSUMDST2 - Count Medium Responses Within Limit 2
.D.MY.E CSUMDST3 - Count Medium Responses Within Limit 3

.D.MY.E CSUMDST4 - Count Medium Responses Within Limit 4
.D.MY.E CSUMDST5 - Count Medium Responses Within Limit 5
.D.MY.E CSUMDST6 - Count Medium Responses Within Limit 6
.D.MY.E CSUMDST7 - Count Medium Responses Within Limit 7
.D.MY.E CSUMDST8 - Count Medium Responses Within Limit 8
XD.MY.E CSUMEMRC - Memory Usage Running Count
.D.... CSUMROCT - MRO Request Count
.D.MY.E CSUMRSTM - Medium Response Time Total
.D.MY.E CSUMTRN - Medium Transactions Processed
XD.MY.E CSUMXTSK - Max Tasking Condition
XD.MY.E CSUNZZTM - Normalized zIIP/zAAP CPU Time
X....E CSUOCPCT - User Task Spec Proc Eligible CPU Count
XD.MY.E CSUOMSGS - Total Output Messages Issued
XD.MY.E CSUOUTCH - Output Message Character Traffic
XD.MY.E CSUPAGIN - Page Ins
XD.MY.E CSUPAGOT - Page Outs
XD.MY.E CSUPCOPS - Number of CICS Program Control Calls
.D.... CSUPCWAT - Program Fetch Wait Time
XD.MY. CSUPCWCT - Program Fetch Wait Count
XD.MY.E CSUPETTM - Task Dispatch Time
XD.MY.E CSURESTM - Transaction Residency Time
XD.MY.E CSURSPCT - Total Number of Response Counts
XD.MY.E CSUSCOPS - Number of CICS Storage Control Calls
.D.MY.E CSUSDST1 - Count Short Responses Within Limit 1
.D.MY.E CSUSDST2 - Count Short Responses Within Limit 2
.D.MY.E CSUSDST3 - Count Short Responses Within Limit 3
.D.MY.E CSUSDST4 - Count Short Responses Within Limit 4
.D.MY.E CSUSDST5 - Count Short Responses Within Limit 5
.D.MY.E CSUSDST6 - Count Short Responses Within Limit 6
.D.MY.E CSUSDST7 - Count Short Responses Within Limit 7
.D.MY.E CSUSDST8 - Count Short Responses Within Limit 8
XD.MY.E CSUSIOCT - Access Method Calls (sio Count)
XD.MY.E CSUSOS - Short On Storage Condition
.D.... CSUSPCOM - Program Compressions
XD.MY.E CSUSPDST - TCB Elapsed Time
XD.MY. CSUSPICT - TADSP Segment Count
XD.MY. CSUSPSWC - TCB Switch Count
XD.MY.E CSUSPSWT - TCB Switch Time
XD.MY. CSUSPWRC - Wait for Redispatch Count
XD.MY.E CSUSQLTM - Total Elapsed Time in SQL Calls
XD.MY.E CSUSQLTO - Total Number of SQL Calls
.D....E CSUSRBTM - SRB CPU Time
.D.MY.E CSUSRSTM - Short Response Time Total
.D.MY.E CSUSTRN - Short Transactions Processed
XD.MY. CSUSUSDT - Unidentified Suspend Wait Time
XD.MY.E CSUSUSTM - Task Suspend (Wait) Time
XD.MY.E CSUTCBTM - TCB CPU Time
X....E CSUTCPTC - User Task Standard Processor CPU Count
XD.MY.E CSUTCWAT - Wait Time For CICS Terminal Control

```
.D.... CSUTDECT - Transient Data (Extra) Request Count
.D.... CSUTDICT - Transient Data (Intra) Request Count
.D.... CSUTDIWC - Transient Data (Intra) Wait Count
XD.MY.E CSUTDOPS - Number of CICS Transient Data Calls
.D.MY.E CSUTDST1 - Count Responses Within Limit 1
.D.MY.E CSUTDST2 - Count Responses Within Limit 2
.D.MY.E CSUTDST3 - Count Responses Within Limit 3
.D.MY.E CSUTDST4 - Count Responses Within Limit 4
.D.MY.E CSUTDST5 - Count Responses Within Limit 5
.D.MY.E CSUTDST6 - Count Responses Within Limit 6
.D.MY.E CSUTDST7 - Count Responses Within Limit 7
.D.MY.E CSUTDST8 - Count Responses Within Limit 8
XD.MY.E CSUTLOPS - Number of CICS Calls (Total)
XD.MY.E CSUTLWAT - Wait Time For CICS (Total)
XD.MY.E CSUTRANS - Transactions Processed
XD.MY.E CSUTRSTM - Transaction Response Time Total
XD.MY. CSUTSIWT - Temp Storage (Aux+Main) Input Wait Time
XD.MY.E CSUTSOPS - Number of CICS Temporary Storage Calls
XD.MY.E CSUTSWAT - Wait Time For CICS Temp Storage
XD.MY.E CSUVSMBW - VSAM Buffer Wait Count
XD.MY.E CSUVMSW - VSAM String Wait Count
XD.MY.E CSUWAITM - Transaction Wait Time
XD.MY.E CSUXCSTM - Task CPU Time Without zIIP/zAAP Eligible
XD.MY.E CSUXTSTM - TCB CPU Time Without zIIP/zAAP Eligible
XD.MY.E CSUZZCTM - zIIP/zAAP Eligible CPU Time on CP
```

Minimum Data Elements

```
X.....E CSUMIRTS - MSAC Mirror Start Time Stamp
```

Maximum Data Elements

```
XD.MY.E CSUCLKOP - Number Of Clocks Generated
XD.MY.E CSUHISTG - Hi-water Terminal And User Memory
.D.MY. CSUMXCTM - Max Conv. Response Time
.D.MY. CSUMXETM - Max Excessive Response Time
.D.MY. CSUMXLTM - Max Long Response Time
XD.MY.E CSUMXMEM - Max Transaction Memory Used
.D.MY. CSUMXMTM - Max Medium Response Time
.D.MY. CSUMXOPS - Maximum Number of CICS Calls
.D.MY. CSUMXSTM - Max Short Response Time
.D.MY. CSUMXTRT - Max Transaction CPU Time
.D.MY. CSUMXTTM - Max Response Time
.D.MY. CSUMXWAT - Maximum Wait Time For CICS
XD.MY.E CSUNTR0P - Number Of Counters Generated
XD.MY.E CSUSTHWA - Task Storage High-water Above 16MB
X.....E CSU64HMM - USER & CICS 64 Pool High-water Mark
```

Derived Data Elements

.D.MY. CSUAVCTM - Avg Conv. Response Time
.D.MY.E CSUAVINC - Average Input Character Traffic
.D.MY. CSUAVLTM - Avg Long Response Time
.D.MY.E CSUAVMEM - Average Transaction Memory Used
.D.MY. CSUAVMTM - Avg Medium Response Time
.D.MY.E CSUAVOPS - Average Number of CICS Calls
.D.MY.E CSUAVOTC - Average Output Character Traffic
.D.MY. CSUAVSTM - Avg Short Response Time
.D.MY.E CSUAVTRT - Average Transaction CPU Time
.D.MY.E CSUAVTTM - Avg Response Time
.D.MY.E CSUAVWAT - Average Wait Time For CICS Functions
.D.MY.E CSUPCTR1 - Cumm Pct Resp Within Limit 1
.D.MY.E CSUPCTR2 - Cumm Pct Resp Within Limit 2
.D.MY.E CSUPCTR3 - Cumm Pct Resp Within Limit 3
.D.MY.E CSUPCTR4 - Cumm Pct Resp Within Limit 4
.D.MY.E CSUPCTR5 - Cumm Pct Resp Within Limit 5
.D.MY.E CSUPCTR6 - Cumm Pct Resp Within Limit 6
.D.MY.E CSUPCTR7 - Cumm Pct Resp Within Limit 7

5.2.6.3 CICCSU Usage Considerations

This section identifies some special considerations or techniques for using the CICCSU File.

1. To activate the CICCSU file at the DETAIL timespan, refer to the instructions in section 10.1.4 of this guide.
2. The following data elements are defined within the CICGENIN member, but are inactive as shipped. They are reserved for user definitions.

CSUCONTM - Terminal Connect Time
CSUEXCPS - I/O (EXCPS) Generated
CSUSERVU - Service Units
CSUSRMTR - SRM Ended Transactions
CSUSRU - System Resource Units

3. The following data elements are dependent upon the setting of specific options within the various members in sharedprefix.MICS.PARMS and prefix.MICS.PARMS. This dependency is outlined below.

prefix.MICS.PARMS(SYSID)
CSUCPUNI - Instructions Executed

prefix.MICS.PARMS(CICOPS) -- parameter RESP
CICRVAL1 - CICRVAL7 - Response Distribution Limits

sharedprefix.MICS.PARMS members CICACCT and CICACRT
CICACT(s)- User Account Identifiers

4. For transactions that are classified as excessive (TRANSTYPE="X"), the transaction counts and response times are excluded from the following data elements:

CSUTRANS - Transaction Processed
CSUTRSTM - Transaction Response Time Total
CSURESTM - Transaction Residency Time
CSUAVTTM - Avg Response Time
CSUMXTTM - Max Response Time

The excessive counts are available in the following data elements:

CSUETRN - Excessive Transactions Processed
CSUERSTM - Excessive Response Time Total
CSUMXETM - Max Excessive Response Time

5. CA MICS Accounting and Chargeback uses data from the CICCISU file for resource accounting.
6. The ENDTS and STARTTS, when appearing in the DAYS or MONTHS timespans, bound the span of time over which the data has been summarized, with STARTTS being the lowest date and time, and ENDTS the highest date and time for the data summarized.

NOTE: For time stamp identification purposes, a CICS-monitored interaction is defined as follows:

- o For pseudo-conversational transactions, for non-conversational transactions, and for conversational transactions not monitored in segmented conversational mode, the entire transaction is one interaction.
- o For conversational transactions, monitored in segmented conversational mode, one interaction is:
 - from the beginning of the transaction to the first suspension for a terminal control read,
 - from one terminal control read suspension to the next, or
 - from a terminal control read suspension to the termination of the transaction.

5.2.6.4 CICCISU Retrieval Examples

In the examples, a SAS macro variable is used to specify the DDname part of the CA MICS file name. These macro variables are a standard part of CA MICS and are available for all files. The macro variable name has the form &ddiit, where d is the database identifier, iiii is the information area name, and t is the timespan. For the examples, a database identifier of P is used. The identifier is installation dependent, so you should find out what the identifiers are at your installation.

1. Generate a horizontal bar graph of transaction CPU time used by the first level of user identification.

```
PROC CHART DATA=&PCICD..CICCSU01;  
HBAR CICTACT1 / SUMVAR=CSUCPUTM DESCENDING;  
RUN;
```

2. Generate a summary report showing workload and performance-related information by CICSID and the first level of user identification for the day before yesterday.

```
%LET BY = CICSID CICTACT1;  
%LET BREAK = CICTACT1;  
  
PROC SORT DATA=&PCICD..CICCSU02;  
BY &BY;  
RUN;  
  
DATA SUMCSU(KEEP=CICSID CICTACT1 MONTH YEAR DAY  
  CSUCPUTM CSUSIOCT CSUTLOPS CSUTRANS CSUAVTTM  
  CSUSOS CSUMXTSK CSUPAGIN CSUFCWAT CSUTLWAT);  
SET CICTCSU;  
BY &BY;  
%CSUSUM(SUMBY=&BY,SUMBREAK=&BREAK,OUT=SUMCSU,  
  DERV=YES);  
RUN;  
  
PROC PRINT DATA=SUMCSU N; BY CICSID;  
PAGEBY CICSID;  
ID CICTACT1;  
VAR MONTH YEAR DAY  
  CSUCPUTM CSUTRANS CSUTLOPS CSUSIOCT CSUAVTTM  
  CSUSOS CSUMXTSK CSUPAGIN CSUFCWAT CSUTLWAT;  
RUN;
```

5.2.7 CICS System Activity File (CICCSY)

The CICS System Activity File contains data quantifying the total activity for the entire CICS system, including all users. Resource consumption, service, availability, and performance measures are included. The CICS system's service (response), load, and usage are stored in the DETAIL timespan. Total usage resides in the DAYS, WEEKS, MONTHS, and YEARS timespans. This file is derived from the global system activity records generated by CMF, or depending on your CICS Version, CICS TS Statistics records, or the system checkpoint records produced by the Monitor. For a detailed description of the data sources required to populate this file, refer to section 6.1 of this guide.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

- 1 - CICCSY File Organization
- 2 - CICCSY Data Elements List
- 3 - CICCSY Usage Considerations
- 4 - CICCSY Retrieval Examples

5.2.7.1 CICCXY File Organization

The table below identifies data elements by which the file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in a timespan. At the DETAIL level, data is sequenced but not summarized.

NOTE: The timespans in which a file is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

+-----+-----+-----+-----+-----+-----+-----+						
Timespan		Level of Data Granularity				
+-----+-----+-----+-----+-----+-----+-----+						
	DETAIL	SYSID	CICSID	YEAR	MONTH	DAY
		HOUR	ENDTS			
	DAYS	SYSID	CICSID	YEAR	MONTH	DAY
		HOUR				
	WEEKS	SYSID	CICSID	YEAR	WEEK	ZONE
		HOUR				
	MONTHS	SYSID	CICSID	YEAR	MONTH	ZONE
	YEARS	SYSID	CICSID	YEAR	ZONE	
+-----+-----+-----+-----+-----+-----+-----+						
	TABLES	N/A				
+-----+-----+-----+-----+-----+-----+-----+						

Generation Date: Tue, May 12, 2009

NOTE: This file was generated with ESSENTIAL=ALL option in effect. All data elements defined in the file are generated.

NOTE: This file was generated with DERIVED=DEfault option in effect. Whether data elements are kept on the file on auxiliary storage or not is controlled by the complex definition of the DERIVED option.

Figure 5-8. CICCXY Timespan Granularity Chart

5.2.7.2 CICCXY Data Elements List

The table below identifies data elements contained in this file. The entries for each data element are:

TIMESPAN: Defines the timespans in which the data element is supported. The timespans are indicated by the letters "XDWMYT" as follows:

- X - DETAIL
- D - DAYS
- W - WEEKS
- M - MONTHS
- Y - YEARS
- T - TABLES AREA
- . - File is not supported

The timespan field also indicates Essential Elements with the letter E, if applicable.

DATA ELEMENT: The data element name.

DATA ELEMENT DESCRIPTION: The data element's long name.

The timespans in which a data element is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

GENERATION DATE: Tue, Jul 30, 2013

Note: Essential data elements are identified by an "E" under the Timespan asterisk (*) column.

Time- Span *	Data Element	Data Element Description (LABEL)
-----	-----	-----

Sequence/Summary Data Elements

XDWMY.E	CICSID	- CICS System Identification
XD....E	DAY	- Day of Month
XDW...E	HOURL	- Hour of Day
XD.M..E	MONTH	- Month of Year
XDWMY.E	SYSID	- System Identifier
XDW...E	WEEK	- Week of Year

XDWMY.E YEAR - Year of Century
XDWMY.E ZONE - Time Zone

Common Data Elements

.DWMY.E CICNODST - Number of Distribution Values
.DWMY.E CICRVAL1 - Response Distribution Limits 1
.DWMY.E CICRVAL2 - Response Distribution Limits 2
.DWMY.E CICRVAL3 - Response Distribution Limits 3
.DWMY.E CICRVAL4 - Response Distribution Limits 4
.DWMY.E CICRVAL5 - Response Distribution Limits 5
.DWMY.E CICRVAL6 - Response Distribution Limits 6
.DWMY.E CICRVAL7 - Response Distribution Limits 7
.DWMY.E CICRVAL8 - Response Distribution Limits 8
XDWMY.E CICSVER - CICS Monitor Version Number
XDWMY.E CICSREL - CICS Release Number
XDWMY.E CPUMODEL - CPU Model Identification
XD....E DAYNAME - Name of Day of Week
XDWMY.E DURATION - Recording Interval Time
XDWMY.E ENDTS - End Time Stamp
XDWMY.E INTERVL - Number of Recording Intervals
XDWMY.E ORGSYSID - Originating System Identification
XDWMY.E STARTTS - Start Time Stamp

Retained Data Elements

XDWMY.E CSYAPPL - CICS VTAM APPLID
XDWMY.E CSYJOB - CICS Job Name
XDWMY. CSYKCTOD - Monitor Relative Collection Time
XDWMY.E CSYMIPS - MIPS rating of this SYSID
XDWMY. CSYOSCAV - OSCOR Bytes Available
XDWMY. CSYOSCOR - OSCOR Total Bytes
XDWMY. CSYRGNS - Region Size

Accumulated Data Elements

.DWMY.E CSYABEND - Abnormal Termination Occurrences
XDWMY.E CSYAMXTC - Total Task During Active Max Task
XDWMY. CSYAPPGI - Application Page-ins
XDWMY. CSYAPPGO - Application Page-outs
.DWMY.E CSYCDST1 - Count Conv. Responses Within Limit 1
.DWMY.E CSYCDST2 - Count Conv. Responses Within Limit 2
.DWMY.E CSYCDST3 - Count Conv. Responses Within Limit 3
.DWMY.E CSYCDST4 - Count Conv. Responses Within Limit 4
.DWMY.E CSYCDST5 - Count Conv. Responses Within Limit 5
.DWMY.E CSYCDST6 - Count Conv. Responses Within Limit 6
.DWMY.E CSYCDST7 - Count Conv. Responses Within Limit 7
.DWMY.E CSYCDST8 - Count Conv. Responses Within Limit 8
XDWMY. CSYCOST - Processing Charges

XDWMY.E CSYCPJTM - JCA CPU TCB Time
XDWMY. CSYCPUNI - Instructions Executed
XDWMY.E CSYCPUTM - CPU Time Consumed
.DWMY.E CSYCRSTM - Conversational Response Time Total
XDWMY.E CSYCSNDP - CSNC Dispatched Time
XDWMY.E CSYCSNPI - CSNC Page In Count
XDWMY.E CSYCSNPO - CSNC Page Out Count
XDWMY.E CSYCSYTM - Task Control Dispatched Elapsed Time
.DWMY.E CSYCTRN - Conversational Transactions Processed
XDWMY.E CSYDAMSC - Storage Violations
XDWMY. CSYDBOVF - DB2 Pool Overflow Count
XDWMY. CSYDBTHW - DB2 Wait Without Thread Hit Count
XDWMY.E CSYDENTM - Dependent Enclave CPU Time
XDWMY.E CSYDISTM - CICS/VS Dispatcher CPU Time
XDWMY. CSYDLDMW - DL/I DMB Pool Wait Count
XDWMY.E CSYDLICT - DL/I Calls
.DWMY. CSYDLIPW - DL/I Pool Wait Count
XDWMY.E CSYDLITI - DL/I Calls Elapsed Time
.DWMY. CSYDLITW - DL/I Thread Wait Count
XDWMY. CSYDLTWC - DL/I Thread Wait Count
.DWMY. CSYDS1WT - Total Wait First Dispatch Time
.DWMY.E CSYERSTM - Excessive Response Time Total
.DWMY.E CSYETRN - Excessive Transactions Processed
XDWMY. CSYEXCPS - I/O (excps) Generated
XDWMY.E CSYFCTOT - Total File Requests
XDWMY. CSYFECT - File Events
XDWMY. CSYFETI - File Elapsed Time
XDWMY. CSYIDLTM - Idle (non-dispatched) Time
.DWMY.E CSYIMSGS - Total Input Messages Issued
.DWMY.E CSYINCH - Input Message Character Traffic
XDWMY. CSYJECT - Journal Events
XDWMY. CSYJETI - Journal Elapsed Time
XDWMY. CSYJPGI - Journal Control Program Page-ins
XDWMY. CSYJPGO - Journal Control Program Page-outs
XDWMY. CSYKPGI - Task Control Program Page-ins
XDWMY. CSYKPGO - Task Control Program Page-outs
.DWMY.E CSYLDST1 - Count Long Responses Within Limit 1
.DWMY.E CSYLDST2 - Count Long Responses Within Limit 2
.DWMY.E CSYLDST3 - Count Long Responses Within Limit 3
.DWMY.E CSYLDST4 - Count Long Responses Within Limit 4
.DWMY.E CSYLDST5 - Count Long Responses Within Limit 5
.DWMY.E CSYLDST6 - Count Long Responses Within Limit 6
.DWMY.E CSYLDST7 - Count Long Responses Within Limit 7
.DWMY.E CSYLDST8 - Count Long Responses Within Limit 8
.DWMY.E CSYLRSTM - Long Response Time Total
.DWMY.E CSYLTRN - Long Transactions Processed
.DWMY.E CSYMDST1 - Count Medium Responses Within Limit 1
.DWMY.E CSYMDST2 - Count Medium Responses Within Limit 2
.DWMY.E CSYMDST3 - Count Medium Responses Within Limit 3

```

.DWMY.E CSYMDST4 - Count Medium Responses Within Limit 4
.DWMY.E CSYMDST5 - Count Medium Responses Within Limit 5
.DWMY.E CSYMDST6 - Count Medium Responses Within Limit 6
.DWMY.E CSYMDST7 - Count Medium Responses Within Limit 7
.DWMY.E CSYMDST8 - Count Medium Responses Within Limit 8
XDWMY. CSYMROCT - MRO Request Count
.DWMY.E CSYMRSTM - Medium Response Time Total
.DWMY.E CSYMTRN - Medium Transactions Processed
XDWMY.E CSYMXTSK - Max Tasking Condition Detected
XDWMY.E CSYNRAMX - Cumulative Active Max Tasks
XDWMY.E CSYNRINT - Number of Measurement Intervals
XDWMY.E CSYNRMXT - Cumulative Max Tasks
XDWMY.E CSYNZZTM - Normalized zIIP/zAAP CPU Time
.DWMY.E CSYOMSGS - Total Output Messages Issued
.DWMY.E CSYOUTCH - Output Messages Character Traffic
XDWMY.E CSYPAGIN - Page Ins
XDWMY.E CSYPAGOT - Page Outs
XDWMY.E CSYPCMDC - Number of Program Compressions
XDWMY.E CSYPCOM - Program Compressions
XDWMY.E CSYPCWAT - Program Fetch Wait Time
XDWMY. CSYPCWCT - Program Fetch Wait Count
XDWMY. CSYPETTM - Dispatched Time
XDWMY.E CSYPGMCT - Program Load Count
XDWMY.E CSYPGMTI - Program Load Time
.DWMY.E CSYSDST1 - Count Short Responses Within Limit 1
.DWMY.E CSYSDST2 - Count Short Responses Within Limit 2
.DWMY.E CSYSDST3 - Count Short Responses Within Limit 3
.DWMY.E CSYSDST4 - Count Short Responses Within Limit 4
.DWMY.E CSYSDST5 - Count Short Responses Within Limit 5
.DWMY.E CSYSDST6 - Count Short Responses Within Limit 6
.DWMY.E CSYSDST7 - Count Short Responses Within Limit 7
.DWMY.E CSYSDST8 - Count Short Responses Within Limit 8
XDWMY.E CSYSIOCT - Access Method Calls (sio Count)
XDWMY.E CSYSOS - Short On Storage Condition
XDWMY. CSYSPCOM - Doublewords Program Storage Freed
XDWMY. CSYSQLT0 - Total SQL Calls
XDWMY.E CSYSRBTM - CPU SRB Time
.DWMY.E CSYSRSTM - Short Response Time Total
.DWMY.E CSYSTRN - Short Transactions Processed
XDWMY.E CSYSUBTM - Total Subtask TCB Time
XDWMY. CSYSUSOT - Unidentified Suspend Wait Time
XDWMY.E CSYTCBTM - Total CICS TCB CPU Time
XDWMY.E CSYTCPJT - Accumulated TCP + JCA CPU Time
XDWMY.E CSYTCPTM - Terminal Control TCB CPU Time
XDWMY.E CSYTCUTM - User (Application) CPU Time
XDWMY. CSYTDECT - Transient Data Events
XDWMY. CSYTDETI - Transient Data Elapsed Time
XDWMY. CSYTDICT - Transient Data (Intra) Request Count
XDWMY. CSYTDIWC - Transient Data (Intra) Wait Count

```

.DWMY.E CSYTDST1 - Count Total Responses Within Limit 1
 .DWMY.E CSYTDST2 - Count Total Responses Within Limit 2
 .DWMY.E CSYTDST3 - Count Total Responses Within Limit 3
 .DWMY.E CSYTDST4 - Count Total Responses Within Limit 4
 .DWMY.E CSYTDST5 - Count Total Responses Within Limit 5
 .DWMY.E CSYTDST6 - Count Total Responses Within Limit 6
 .DWMY.E CSYTDST7 - Count Total Responses Within Limit 7
 .DWMY.E CSYTDST8 - Count Total Responses Within Limit 8
 XDWMY.E CSYTMJCD - JCP Dispatched Time
 XDWMY.E CSYTMKCD - KCP Dispatched Time
 XDWMY.E CSYTMTC D - TCP Dispatched Time
 XDWMY. CSYTOTE - Transactions Excluded
 XDWMY. CSYTOTI - Transactions Initiated
 XDWMY. CSYTOTT - Nonexcluded Transactions Ended
 XDWMY. CSYTPGI - Terminal Control Program Page-ins
 XDWMY. CSYTPGO - Terminal Control Program Page-outs
 .DWMY.E CSYTRANS - Total Transactions Ended
 .DWMY.E CSYTRSTM - Total Response Time All Functions
 XDWMY. CSYTSECT - Temporary Storage Events
 XDWMY. CSYTSETI - Temporary Storage Time
 XDWMY. CSYTSOTA - Temp Storage (Aux) Output Requests
 XDWMY. CSYTSOTM - Temp Storage (Main) Output Requests
 XDWMY.E CSYTSRTM - Task Control CPU Time
 XDWMY.E CSYTTIM - Nonexcluded Transaction Time
 XDWMY.E CSYUPTM - CICS Availability Time
 XDWMY.E CSYUSRTM - User (application) Dispatched Time
 XDWMY. CSYVCPU - CVST CPU Time
 XDWMY. CSYVDTIM - CVST Dispatched Time
 XDWMY. CSYVPGI - CVST Page-ins
 XDWMY. CSYVPGO - CVST Page-outs
 .DWMY. CSYVSMBW - VSAM Buffer Wait Count
 .DWMY. CSYVMSW - VSAM String Wait Count
 XDWMY. CSYWTCPU - CPU Time Across OPSYS Wait
 XDWMY.E CSYWTIM - CICS Dispatcher Wait Time
 XDWMY.E CSYXCSTM - Task CPU Time Without zIIP/zAAP Eligible
 XDWMY.E CSYXTSTM - TCB CPU Time Without zIIP/zAAP Eligible
 XDWMY.E CSYZZCTM - zIIP/zAAP Eligible CPU Time on CP

Maximum Data Elements

XDWMY.E CSYMEMSZ - Hi-water Dynamic Area Memory
 XDWMY. CSYMPCPU - Max CPU Usage
 XDWMY.E CSYMXAMX - Maximum Active Task Limit
 .DWMY. CSYMXCTM - Max Conversational Response Time
 .DWMY. CSYMXETM - Max Excessive Response Time
 .DWMY. CSYMXLTM - Max Long Response Time
 .DWMY. CSYMXMTM - Max Medium Response Time
 XDWMY.E CSYMXMXT - Maximum Maxtask Value
 .DWMY. CSYMXSTM - Max Short Response Time

.DWMY. CSYMXTTM - Max Response Time All Functions

Derived Data Elements

.DWMY.E CSYAVCTM - Avg Conversational Response Time
.DWMY.E CSYAVLTM - Avg Long Response Time
.DWMY.E CSYAVMTM - Avg Medium Response Time
.DWMY.E CSYAVSTM - Avg Short Response Time
.DWMY.E CSYAVTTM - Avg Response Time All Functions
.DWMY. CSYMSGs - Total Messages Issued (in/out)
.DWMY.E CSYPCCPU - Pct CPU Usage
.DWMY. CSYPCCR1 - Cumulative Pct Conv Resp Within Limit 1
.DWMY. CSYPCCR2 - Cumulative Pct Conv Resp Within Limit 2
.DWMY. CSYPCCR3 - Cumulative Pct Conv Resp Within Limit 3
.DWMY. CSYPCCR4 - Cumulative Pct Conv Resp Within Limit 4
.DWMY. CSYPCCR5 - Cumulative Pct Conv Resp Within Limit 5
.DWMY. CSYPCCR6 - Cumulative Pct Conv Resp Within Limit 6
.DWMY. CSYPCCR7 - Cumulative Pct Conv Resp Within Limit 7
.DWMY. CSYPCLR1 - Cumulative Pct Long Resp Within Limit 1
.DWMY. CSYPCLR2 - Cumulative Pct Long Resp Within Limit 2
.DWMY. CSYPCLR3 - Cumulative Pct Long Resp Within Limit 3
.DWMY. CSYPCLR4 - Cumulative Pct Long Resp Within Limit 4
.DWMY. CSYPCLR5 - Cumulative Pct Long Resp Within Limit 5
.DWMY. CSYPCLR6 - Cumulative Pct Long Resp Within Limit 6
.DWMY. CSYPCLR7 - Cumulative Pct Long Resp Within Limit 7
.DWMY. CSYPCMR1 - Cumulative Pct Medium Resp Within Limit 1
.DWMY. CSYPCMR2 - Cumulative Pct Medium Resp Within Limit 2
.DWMY. CSYPCMR3 - Cumulative Pct Medium Resp Within Limit 3
.DWMY. CSYPCMR4 - Cumulative Pct Medium Resp Within Limit 4
.DWMY. CSYPCMR5 - Cumulative Pct Medium Resp Within Limit 5
.DWMY. CSYPCMR6 - Cumulative Pct Medium Resp Within Limit 6
.DWMY. CSYPCMR7 - Cumulative Pct Medium Resp Within Limit 7
.DWMY. CSYPCSR1 - Cumulative Pct Short Resp Within Limit 1
.DWMY. CSYPCSR2 - Cumulative Pct Short Resp Within Limit 2
.DWMY. CSYPCSR3 - Cumulative Pct Short Resp Within Limit 3
.DWMY. CSYPCSR4 - Cumulative Pct Short Resp Within Limit 4
.DWMY. CSYPCSR5 - Cumulative Pct Short Resp Within Limit 5
.DWMY. CSYPCSR6 - Cumulative Pct Short Resp Within Limit 6
.DWMY. CSYPCSR7 - Cumulative Pct Short Resp Within Limit 7
.DWMY.E CSYPCTR1 - Cumulative Pct All Resp Within Limit 1
.DWMY.E CSYPCTR2 - Cumulative Pct All Resp Within Limit 2
.DWMY.E CSYPCTR3 - Cumulative Pct All Resp Within Limit 3
.DWMY.E CSYPCTR4 - Cumulative Pct All Resp Within Limit 4
.DWMY.E CSYPCTR5 - Cumulative Pct All Resp Within Limit 5
.DWMY.E CSYPCTR6 - Cumulative Pct All Resp Within Limit 6
.DWMY.E CSYPCTR7 - Cumulative Pct All Resp Within Limit 7

5.2.7.3 CICCXY Usage Considerations

This section identifies some special considerations or techniques for using the CICCXY File.

1. The following data elements are defined within the CICCENIN member of sharedprefix.MICS.GENLIB and are reserved for user derivation:

- CSYSRU - System Resource Units
- CSYEXCPS - I/O (EXCPs) Generated
- CSYSERVU - Service Units
- CSYSRMTR - SRM Ended Transactions
- CSYCONTM - Terminal Connect Time

2. The following data elements are dependent upon the options set within the prefix.MICS.PARMS(CICOPS) member or the prefix.MICS.PARMS(SYSID) member. This dependency is outlined below.

- prefix.MICS.PARMS(SYSID)
 - CSYCPUNI - Instructions Executed

- prefix.MICS.PARMS(CICOPS) - parameter RESP
 - CSYRVAL1-CSYRVAL8 - Response Distribution Limits

3. The ENDTS and STARTTS, when appearing in the DAYS, WEEKS, MONTHS, or YEARS timespans, bound the span of time over which the data has been summarized, with STARTTS being the lowest date and time, and ENDTS the highest date and time for the data summarized.
4. The data elements in the following list do not exist in the DETAIL timespan for the CICCXY File. These data elements are not derived from the raw data, but rather are derived by a summarization of the CICCXY DAYS file.

CSYCDST1 - CSYCDST8 - Count Conv. Responses In Limit
CSYCRSTM - Conversational Response Time Total
CSYCTRN - Conversational Transactions Processed
CSYDAMSC - Damaged Storage Chain
CSYERSTM - Excessive Response Time Total
CSYETRN - Excessive Transactions Processed
CSYINCH - Input Message Character Traffic
CSYINVPA - Invalid CMF Call Encountered
CSYLDST1 - CSYLDST8 - Count Long Responses In Limit
CSYLRSTM - Long Response Time Total
CSYLTRN - Long Transactions Processed
CSYMDST1 - CSYMDST8 - Count Medium Responses In Limit
CSYMRSTM - Medium Response Time Total
CSYMTRN - Medium Transactions Processed
CSYMXCTM - Max Conversational Response Time
CSYMXETM - Max Excessive Response Time
CSYMXLTM - Max Long Response Time
CSYMXMTM - Max Medium Response Time
CSYMXSTM - Max Short Response Time
CSYMXTSK - Max Tasking Condition
CSYMXTTM - Max Response Time All Functions
CSYOUTCH - Output Messages Character Traffic
CSYPCCR1 - CSYPCCR7 - Cumm Pct Conv Resp In Limit
CSYPCLR1 - CSYPCLR7 - Cumm Pct Long Resp In Limit
CSYPCMR1 - CSYPCMR7 - Cumm Pct Medium Resp In Limit
CSYPCSR1 - CSYPCSR7 - Cumm Pct Short Resp In Limit
CSYPCTR1 - CSYPCTR7 - Cumm Pct All Resp In Limit
CSYSDST1 - CSYSDST8 - Count Short Responses In Limit
CSYSOS - Short On Storage Condition
CSYSRSTM - Short Response Time Total
CSYSTRN - Short Transactions Processed
CSYTMERR - CMF Timing Error Encountered
CSYTRANS - Total Transactions Ended in Interval
CSYTRSTM - Total Response Time All Functions

5. CICS System File User Data Accumulation

While most data elements in the CICCXY File contain system data, some elements originate from the CICCXY File and provide transaction statistics. This combination of elements allows the CICCXY File to retain a complete picture of system activity. This is necessary because response time distribution or system stress data is not in the system records produced by any supported monitor.

The data elements derived from the DAYS.CICCXY transaction data are:

- CSYCDSTX - Count Conv. Responses Within Limit
- CSYCRSTM - Conversational Response Time Total
- CSYCTRN - Conversational Transactions Processed
- CSYDAMSC - Damaged Storage Chain
- CSYERSTM - Excessive Response Time Total
- CSYETRN - Excessive Transactions Processed
- CSYFCTOT - Total File Requests
- CSYIMSGS - Input Messages Issued
- CSYINCH - Input Message Character Traffic
- CSYINVPA - Invalid CMF Call Encountered
- CSYLDSTx - Count Long Responses Within Limit
- CSYLRTM - Long Response Time Total
- CSYLTRN - Long Transactions Processed
- CSYMDSTx - Count Medium Responses Within Limit
- CSYMRSTM - Medium Response Time Total
- CSYMTRN - Medium Transactions Processed
- CSYMXCTM - Max Conversational Response Time
- CSYMXETM - Max Excessive Response Time
- CSYMXLTM - Max Long Response Time
- CSYMXMTM - Max Medium Response Time

CSYMXSTM - Max Short Response Time
CSYMXTSK - Max Tasking Condition
CSYMXTTM - Max Response Time All Functions
CSYOMSGS - Output Messages Issued
CSYOUTCH - Output Messages Character Traffic
CSYSDSTx - Count Short Responses Within Limit
CSYSIOCT - Access Method Calls
CSYSOS - Short On Storage Condition
CSYSRSTM - Short Response Time Total
CSYSTRN - Short Transactions Processed
CSYTDSTx - Count Total Responses Within Limit
CSYTMERR - Monitor Timing Error Encountered
CSYTRANS - Total Transactions Ended
CSYTRSTM - Total Response Time All Functions

During the summarization process for the DAYS.CICCSY File, a subset of the DAYS.CICCSU DAYS data is created (the CSR work file). This file contains the data elements being carried over from the transaction data to the CICCSY system records. The CSY data being summarized and the CSR data previously created is then merged to create the complete CICCSY observation for this CICS system.

Note: Missing CICS system records will cause transaction data for that interval to be lost.

Example:

The data element CSYTRANS is the number of transaction records for the interval. This statistic is not contained in the CMF global records or the monitor task accounting system records, so it must be derived from the transaction data. During the special daily summarization process for the CSY file, the summarized DAYS.CICCSU00 file is passed and certain data elements are extracted, including CSUTRANS. The intermediate file, called CICCSR, is sorted into CICCSY DAYS timespan sequence. Then CICCSR and DAYS.CICCSY00 are input into the same DATA statement. The values of CSUTRANS are summarized into a retained data element called SSYTRANS. When the associated CICCSY record is encountered and rewritten, the value of CSYTRANS is assigned from SSYTRANS, and SSYTRANS is reset to zero.

5.2.7.4 CICCXY Retrieval Examples

In the examples, a SAS macro variable is used to specify the DDname part of the CA MICS file name. These macro variables are a standard part of CA MICS and are available for all files. The macro variable name has the form &diiiit, where d is the database identifier, iii is the information area name, and t is the timespan. For the examples, a database identifier of P is used. The identifier is installation dependent, so you should find out what the identifiers are at your installation.

1. Print yesterday's average short response by hour.

```
PROC PRINT DATA=&PCICD.CICCXY01;
  ID HOUR; VAR SYSID CICSID CSYAVSTM;
RUN;
```

2. Generate a vertical bar graph of ended transaction activity for each hour of the day for the day before yesterday.

```
PROC CHART DATA=&PCICD..CICCXY02;
  VBAR HOUR /
  MIDPOINTS=0 TO 23 BY 01 SUMVAR=CSYTRANS TYPE=SUM
  DISCRETE;
RUN;
```

3. Generate a block chart of the percentage of CPU time spent in each major CICS service facility.

```
DATA SUMF (KEEP= TIMETYPE PERCENT);
  SET &PCICD..CICCXY01 END=EOF;
  RETAIN TOTTOT TSRTOT SRBTOT CPJTOT TCPTOT
    USRTOT 0;
  TSRTOT + CSYTSRTM;
  SRBTOT + CSYSRBTM;
  CPJTOT + CSYCPJTM;
  TCPTOT + CSYTCPTM;
  USRTOT + CSYUSRTM;
  IF EOF THEN DO;
    TOTTOT+TSRTOT+SRBTOT+
      CPJTOT+TCPTOT+USRTOT;
    IF TOTTOT NE 0 THEN DO;
      TIMETYPE = 'TSR';
      PERCENT = 100 * TSRTOT / TOTTOT;
      OUTPUT SUMF;
      TIMETYPE = 'SRB';
    END;
  END;
```

```
PERCENT = 100 * SRBTOT / TOTTOT;  
OUTPUT SUMF;  
TIMETYPE = 'CPJ';  
PERCENT = 100 * CPJTOT / TOTTOT;  
OUTPUT SUMF;  
TIMETYPE = 'TCP';  
PERCENT = 100 * TCPTOT / TOTTOT;  
OUTPUT SUMF;  
TIMETYPE = 'USR';  
PERCENT = 100 * USRTOT / TOTTOT;  
OUTPUT SUMF;  
END;  
END; RUN;  
PROC CHART DATA=SUMF;  
BLOCK TIMETYPE / SUMVAR=PERCENT ; RUN;
```

5.2.8 CICS MRO Activity File (CICCMR)

The CICS MRO Activity File is an optional CICS file that contains data quantifying the CICS system's MRO usage for the DAYS, WEEKS, MONTHS, and YEARS timespans. This file is derived from the MRO segments from Monitor data (versions 8.0 and above).

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

- 1 - CICCMR File Organization
- 2 - CICCMR Data Elements List
- 3 - CICCMR Usage Considerations
- 4 - CICCMR Retrieval Examples

5.2.8.1 CICC MR File Organization

The table below identifies data elements by which the file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in a timespan. At the DETAIL level, data is sequenced but not summarized.

NOTE: The timespans in which a file is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

Timespan	Level of Data Granularity
DETAIL	N/A
DAYS	N/A
WEEKS	N/A
MONTHS	N/A
YEARS	N/A
TABLES	N/A

Generation Date: Tue, May 12, 2009

NOTE: This file was generated with ESSENTIAL=ALL option in effect. All data elements defined in the file are generated.

NOTE: This file was generated with DERIVED=Default option in effect. Whether data elements are kept on the file on auxiliary storage or not is controlled by the complex definition of the DERIVED option.

Figure 5-9. CICC MR Timespan Granularity Chart

5.2.8.2 CICC MR Data Elements List

The table below identifies data elements contained in this file. The entries for each data element are:

TIMESPAN: Defines the timespans in which the data element is supported. The timespans are indicated by the letters "XDWMYT" as follows:

- X - DETAIL
- D - DAYS
- W - WEEKS
- M - MONTHS
- Y - YEARS
- T - TABLES AREA
- . - File is not supported

The timespan field also indicates Essential Elements with the letter E, if applicable.

DATA ELEMENT: The data element name.

DATA ELEMENT DESCRIPTION: The data element's long name.

The timespans in which a data element is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

GENERATION DATE: Mon, Apr 5, 2010

Note: Essential data elements are identified by an "E" under the Timespan asterisk (*) column.

Time- Span *	Data Element	Data Element Description (LABEL)
-----	-----	-----

Sequence/Summary Data Elements

.....E	CICAPU	- CICS Application Unit ID
.....E	CICSID	- CICS System Identification
.....E	HOURL	- Hour of Day
.....E	MONTH	- Month of Year
.....E	SYSID	- System Identifier
.....E	WEEK	- Week of Year
.....E	YEAR	- Year of Century

.....E ZONE - Time Zone

Common Data Elements

.....E DAY - Day of Month
.....E DAYNAME - Name of Day of Week
.....E ENDTS - End Time Stamp
.....E MICSVER - CA MICS Version Number
.....E STARTTS - Start Time Stamp

5.2.8.3 CICC MR Usage Considerations

This section identifies some special considerations or techniques related to using the CICC MR File.

1. The CICC MR File is available only when The Monitor for CICS is the input data source and The Monitor is at release 8.0 or above. The input to this file consists of MRO segments contained within The Monitor's Detail Transaction record.

Refer to Section 10.1.13 for information on activating this file.

2. The USRSCMR exit can be used to limit the amount of data to be summarized into the CICC MR File. This can be done coding the exit to reject a MRO segment from further processing.
3. The ENDTS and STARTTS, when appearing in the DAYS, WEEKS, MONTHS, or YEARS timespans, bound the span of time over which the data has been summarized, with STARTTS being the lowest date and time, and ENDTS the highest date and time for the data summarized.

5.2.8.4 CICC MR Retrieval Examples

In the examples, a SAS macro variable is used to specify the DDname part of the CA MICS file name. These macro variables are a standard part of CA MICS and are available for all files. The macro variable name has the form &diiiit, where d is the database identifier, iii is the information area name, and t is the timespan. For the examples, a database identifier of P is used. The identifier is installation dependent, so you should find out what the identifiers are at your installation.

1. Print yesterday's function ship request count by hour.

```
PROC PRINT DATA=&PCICD..CICCMR01;
  ID HOUR; VAR CICSID CICAPU CMRCOUNT CMRELPTM;
  SUM CMRCOUNT CMRELPTM;
RUN;
```

2. Generate a summary report showing function shipping activity by CICSID and application unit (CICAPU) for the last three days.

```
%LET BY = CICSID CICAPU MONTH YEAR DAY;
%LET BREAK = DAY;
DATA CICC MR(KEEP=%CMRKEEP(OP=FILEOPTS,TS=DAYS));
  SET &PCICD..CICCMR03
      &PCICD..CICCMR02
      &PCICD..CICCMR01;
RUN;
PROC SORT DATA=CICC MR;
  BY &BY;
RUN;

DATA SUMC MR(KEEP=CICSID CICAPUB MONTH YEAR DAY
  CMRCOUNT CMRELPTM);
  SET CICC MR;
  BY &BY;
  IF FIRST.CICAPU THEN CICAPUB=CICAPU;
  ELSE CICAPUB=' ';
  %CMRSUM(SUMBY=&BY,SUMBREAK=&BREAK,OUT=SUMC MR,
  DERV=YES);
RUN;

PROC PRINT DATA=SUMC MR N; BY CICSID;
  PAGEBY CICSID;
  ID CICAPUB;
  VAR MONTH YEAR DAY CMRCOUNT CMREPLTM;
```

RUN;

5.3 CTG Information Area Files

This section identifies each file in the CICS Transaction Gateway (CTG) Information Area and defines its level of summarization and data sequencing as the files appear in the applicable timespan levels. It also lists the data elements contained in each file.

The files are described in the following sections:

- 1 - Gateway Server Activity File (CTGGSA)

5.3.1 Gateway Server Activity File (CTGGSA)

The Gateway Server Activity contains data quantifying the activity, performance, state, and configuration of each CICS Transaction Gateway (CTG) z/OS address space. This file is derived from the SMF type 111 CICS Transaction Gateway interval record.

This file is shipped as active, but is only populated when a GATEWAY ACTIVE statement is added to prefix.MICS.PARMS(CICOPS). See section 7.3.2.6 for more information about activating CTG record processing.

There is no CA MICS Accounting and Chargeback Option interface for the CTGGSA file. The data in the file is suitable for performance analysis and capacity planning related to z/OS CICS Transaction Gateway address spaces.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

- 1 - CTGGSA File Organization
- 2 - CTGGSA Data Elements List
- 3 - CTGGSA Usage Considerations
- 4 - CTGGSA Retrieval Examples

5.3.1.1 CTGSA File Organization

The table below identifies data elements by which the file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in a timespan. At the DETAIL level, data is sequenced but not summarized.

NOTE: The timespans in which a file is supported are defined by each installation when CA MICS is installed. Therefore, this table has been generated as part of the installation process to accurately reflect the CA MICS system at your installation.

+-----+-----+-----+-----+-----+-----+-----+						
Timespan	Level of Data Granularity					
+-----+-----+-----+-----+-----+-----+-----+						
DETAIL	SYSID	CTGID	YEAR	MONTH	DAY	
	HOUR	ENDTS				
DAYS	SYSID	CTGID	YEAR	MONTH	DAY	
	HOUR					
WEEKS	SYSID	CTGID	YEAR	WEEK	ZONE	
	HOUR					
MONTHS	SYSID	CTGID	YEAR	MONTH	ZONE	
YEARS	SYSID	CTGID	YEAR	ZONE		
+-----+-----+-----+-----+-----+-----+-----+						
TABLES	N/A					
+-----+-----+-----+-----+-----+-----+-----+						

Generation Date: Mon, Apr 5, 2010

NOTE: This file was generated with ESSENTIAL=ALL option in effect. All data elements defined in the file are generated.

NOTE: This file was generated with DERIVED=Default option in effect. Whether data elements are kept on the file on auxiliary storage or not is controlled by the complex definition of the DERIVED option.

Figure 5-10. CTGSA Timespan Granularity Chart

Chapter 6: DATA SOURCES

The CA MICS CICS Analyzer supports data from the following sources:

- o IBM's CICS Monitoring Facility (CMF), CTS 1.2 through CTS 3.2

CICS CMF monitoring data is written out using the SMF 110 record type with the following subtypes:

- 0 - CICS journaling
- 1 - CICS monitoring
- 2 - CICS statistics
- 3 - Shared temporary storage queue server
- 4 - Coupling facility data table server statistics
- 5 - Named counter sequence number server statistics

The CA MICS Analyzer Option for CICS uses only the subtype 1 and 2 records at this time. Subtypes 0, 3, 4 and 5 are not used currently.

Within the different subtypes, there are one or more classes of data produced, depending on the requirements. The classes of data within the CICS monitoring record (subtype 1) currently available are:

- 1 - Dictionary data
- 2 - unused
- 3 - Performance data
- 4 - Exception data
- 5 - Transaction Resource data

The primary input is the subtype 1 (monitoring) record, class 1, dictionary data and class 3, performance data. Processing for the exception data (class 4) is optional.

- o CICS Statistics Records

For all supported releases, the CA MICS Analyzer Option for CICS processes selected CICS statistics written to SMF type 110, subtype 1, class 1 record.

For CTS 1.3, statistics record types 2, 6, 10, 27, 45, 48, and 55 are processed. For CTS 2.1 and above, statistic record types 2, 6, 10, 27, 45, 48, and 60 are processed.

- o ASG-TMON for CICS TS (TCE) through Release 3.2

The CA MICS Analyzer Option for CICS provides support for the Region Interval (TR) record in addition to Transaction Performance activity (TA) and Transaction Performance History interval (TI) records.

- o CMF type 110 look-alike records

CMF type 110 look-alike records produced by products such as OMEGAMON are also supported, to the extent they are compatible with IBM's CMF data at a given CICS release level. Currently, the CA MICS Analyzer Option for CICS supports DBCTL for IMS usage reporting and OMEGAMON for general performance reporting only. If additional data elements are introduced, then these elements must be defined within the \$CICGENIN member and an appropriate exit must be used to process the input data. See section 10.1.15 for details.

- o CICS Transaction Gateway SMF type 111 records

SMF type 111 interval records produced by IBM's CICS Transaction Gateway Release 7.1 and above.

Consult your organization's CICS systems programmer to ensure that at least one of these products is available on your system.

This chapter discusses details about records used from each data source and the files and data elements that they populate in the CA MICS database.

This section contains the following topics:

- [6.1 Data Source Descriptions](#) (see page 509)
- [6.2 File/Data Source Cross-Reference](#) (see page 517)
- [6.3 Data Element/Data Source Cross-Reference](#) (see page 519)
- [6.4 CA MICS and CA SMF Director Interface](#) (see page 548)

6.1 Data Source Descriptions

The CA MICS Analyzer Option for CICS processes raw data generated by a variety of monitors. The following sections describe each of these raw data sources:

- 1 CICS Monitoring Facility (CMF)
- 2 ASG-TMON for CICS TS for z/OS
- 3 OMEGAMON II for CICS
- 4 CICS Transaction Gateway SMF Type 111

6.1.1 CICS Monitoring Facility (CMF)

The CICS Monitoring Facility (CMF) collects performance data at the system and transaction levels during online processing and produces records for later batch analysis.

CMF at CTS 2.2 and Higher Releases

CMF collects three classes of data: dictionary, performance, and exception, all of which are stored in type 110 subtype 1 records and then processed by the CA MICS Analyzer Option for CICS. Each of the three classes is discussed as follows:

Dictionary Class

The dictionary class record (class 1) describes the format and length of each field that is contained in the performance class records. It is written when the performance class data are activated, either at CICS startup or while CICS is running. The dictionary record contains the definitions for all of the CMF-supplied fields, with or without excluded fields. It also includes definitions for all user-defined fields, such as clocks, counters, and user areas, that you have added through EMPs.

The dictionary record is required by the CA MICS Analyzer Option for CICS to process the input data in the performance class records only. The dictionary records are not used in reading exception class records or in reading statistics (subtype 2) records. These records are read and stored in the CICS Dictionary (CICCDC) file at the DETAIL timespan only.

The CICCDC file retains the last copy of the dictionary record read for each region, so that CA MICS can continue processing even when the dictionary record is absent in the next day's input. The lack of a dictionary record in input is a typical condition for CICS systems that are not shut down each day.

If the dictionary record is not present in your input, you can generate one using the IBM utility program DFHMNDUP. You can concatenate this generated file in front of the SMF input for the first DAILY run. This will avoid data rejection errors in CA MICS caused by the lack of dictionary records. The DFHMNDUP utility is described in the IBM CICS Operations and Utilities Guide.

Performance Class

Performance class data (class 3) provides detailed resource usage information about the execution of each transaction. Each record is written when the transaction terminates. For conversational transactions, one transaction record is written for each segment of the conversation (a pair of terminal I/Os) when CONV=YES is specified in the DFHMCT TYPE=INITIAL macro, or when MNCONV=YES is specified in the SIT (CTS 2.2 and higher). Otherwise, CMF writes one transaction record for each conversational transaction.

CMF allows a user application program to add data to the transaction record through EMPs. These data fields are known as user clocks, user counters, and user areas. The CA MICS Analyzer Option for CICS uses the dictionary record to process the transaction records, including the user clocks, user counters, and one user area. The information is then saved in the CICS User Application Count (CICCAC) file, the CICS Application Unit Activity (CICCAU) file, and the CICS User Activity (CICCSU) file.

Beginning with CTS 3.2, CICS can perform data compression on the SMF 110 monitoring records output by the CICS monitoring facility (CMF). When data compression is active, CICS uses the standard z/OS Data Compression and Expansion Services (CSRCEsrv) to compress the CICS data section of each monitoring record before writing it to SMF. The CICS-supplied monitoring sample program DFH\$MOLS can be used to copy the compressed monitoring records to an output data set in their expanded format, with the records that were never compressed. The CA MICS DAILY or incremental update job determines whether input records have been compressed and expands (decompress) them automatically before processing their fields.

Exception Class

The exception class data (class 4) is written when an exception condition occurs during execution of a transaction, such as waiting for a VSAM string. A separate record is produced for each exception that is encountered by the transaction. The following exceptions are recorded:

- o Wait for storage in DSA
- o Wait for storage in EDSA
- o Wait for storage in CDSA
- o Wait for storage in UDSA
- o Wait for storage in ECDSA

- o Wait for storage in EUDSA
- o Wait for storage in ERDSA
- o Wait for Temporary storage
- o Wait for file string
- o Wait for LSRPOOL buffer
- o Wait for LSRPOOL string

No data dictionary records are produced for the exception class. The CA MICS Analyzer Option for CICS reads the exception records and stores the exception conditions, one per observation, in the CICS Incident (CICICIN) file.

CICS Statistics

Beginning with CICS/ESA 3.1.1, IBM realigned CMF and CICS statistics to make it easier for users to relate monitoring and statistics data. CMF, which no longer produces system global records on an interval basis, provides monitoring at the transaction level only. CICS statistics, enhanced to record to SMF on an interval basis, provide monitoring at the system and resource levels. Together, the two data sources provide a complete picture of CICS activities, and can be used to do more effective capacity planning and performance tuning. Given these changes in CMF and CICS statistics, the CA MICS Analyzer Option for CICS processes selected CICS global statistics to provide the system-wide data that were previously obtained from CMF. The CICS statistics data are reported in type 110 subtype 2 records, which in turn has several different types of statistics. These statistics types have numbers of their own. Although CA MICS does not process all of these statistics types, it does process the following, where the type numbers for a given statistic type vary from one release of CICS to another:

- o Dispatcher statistics (type 62, 60, 57, 56, or 55)
- o Loader statistics (type 30)
- o Temporary storage statistics (type 48)
- o Transaction manager (type 10)
- o Transient data statistics (type 45)
- o Storage manager task subpool (type 6)
- o Storage manager DSA statistics (type 29, 14, or 2)

All the previous records are processed and stored in the CICS System Activity (CICCSY) file.

The CICS statistics are written to SMF on an interval basis; the default is 3 hours. Because the CA MICS Analyzer Option for CICS performs a merge, based on hour, between user activity data and system data, the 3-hour interval causes

invalid data in the CICS SY file at the DAYS and higher timespans. Therefore, we recommend that you reset the statistics interval to one hour or less by using the EXEC CICS SET STATISTICS INTERVALHRS(01) command. For more information about this command, see section 2.151 of the IBM CICS Systems Programming Reference guide.

6.1.2 ASG-TMON for CICS TS for z/OS

ASG-TMON is an online CICS monitor. It records performance data and file activity data for later batch analysis.

With releases 2.0 and higher of ASG-TMON, the record formats changed. The Transaction record (TA) has been updated to reflect CICS/ESA and later measurements while dropping old XA measurements that are no longer valid.

The TI record that, in previous releases was a system interval summary, is now an accumulation of the data from the transaction performance detail records (TA) for tasks that ended within the time interval of the TI record. There are, however, some metrics on this record that do contain system (region) level information. These include region CPU times and DL/I resource usage.

It is very important that the intervals for the TI and TR records be the same and something less than or equal to 60 minutes, for example, 60, 30, or 15, in order to populate the CICS System Activity (CICCSY) file properly. The data from the TI and the TR records is merged together by an interval timestamp. If there is no matching TI record for a TR record, the data from the TR record will be dropped. However, CA MICS normalizes the end timestamp values from both the TI and TR records in such a way that the end timestamps will have an exact match in almost all cases.

Decompression Load Module Requirement for ASG-TMON

A decompression module is required to process ASG-TMON data in compressed format. You must either copy the appropriate decompression module supplied by the vendor to sharedprefix.MICS.LOAD, or concatenate the library containing the module in DAY040's STEPLIB in the CA MICS DAILY job.

When copying the load module, ensure format compatibility; the CA MICS load data set is delivered with a blocksize of 6160. Using IEBCOPY with the COPY statement will not reblock the module. If the DCB attributes are different, use either COPYMOD with IEBCOPY, or SAS PROC PDSCOPY to copy the decompression module.

Decompression module for ASG-TMON:

\$CRCPRS - ASG-TMON for CICS TS 2.0 and above

6.1.3 OMEGAMON II for CICS

The CA MICS Analyzer Option for CICS also processes the type 110 records written by OMEGAMON II for CICS. The data is recorded to SMF and is treated by CA MICS the same way as CMF input. A data dictionary record is required for each CICS region being processed by CA MICS. For CICS 2.1, the type 110 records produced by OMEGAMON do not have the same format as the CMF type 110 records. Therefore, you must input either the OMEGAMON data or CMF data in CA MICS, but never both. An abend in DAY040 during DAILY processing will occur if both OMEGAMON and CMF records are encountered in input, because an incorrect copy of the dictionary record may be used to read the data records. This problem does not occur in CICS 3.1.1 and higher releases.

The OMEGAMON type 110 record contains multiple user areas to provide DB2, DL/I, and other database usage information. The DL/I user area contains data for both local DL/I and DBCTL usage. The CA MICS Analyzer Option for CICS automatically decodes these user areas and populates the appropriate data elements that are activated by OPTION OMEGU in the CICGENIN member in the sharedprefix.MICS.GENLIB data set.

6.1.4 CICS Transaction Gateway SMF Type 111

The CA MICS Analyzer Option for CICS processes the SMF type 111 interval records generated by IBM's CICS Transaction Gateway. The Gateway Server Activity (CTGGSa) file is created from these SMF records and provides comprehensive information about the activity and performance of each Transaction Gateway address space.

Parameters within the IBM CICS Transaction Gateway control both the recording of SMF type 111 interval records, as well as the interval duration (default 3 hours), and the end-of-day time (default 00:00:00 - midnight).

We recommend a minimum of 1 hour for interval duration, and that the end-of-day time be set to 23:59:00 (1 minute before midnight). This will ensure that the correct interval records are summarized into the appropriate HOUR as CA MICS summarizes the CTGGSa file to the DAYS and higher timespans.

Refer to the IBM CICS z/OS Gateway Administration guide for specific instructions on how to set these SMF recording related parameters.

6.2 File/Data Source Cross-Reference

Figure 6-1 provides a cross-reference between the CIC Information Area files and the corresponding input data sources.

CIC Information Area Files	IBM Type 110	ASG-TMON
CICS System Activity File (CICCSY)	CICS Statistics Type 110 and Subtype 2 records for each of these release levels: CICS 3.1.1 through 3.3 (types 8, 9, 27, 45, 48, 57) CICS 4.1 (types 2, 6, 10, 27, 45, 48, 57) CICS/TS 1.1 and 1.2 (types 2, 6, 10, 27, 45, 48, 56) CICS/TS 1.3 (types 2, 6, 10, 27, 45, 48, 55) CICS/TS 2.1 thru 3.1 (types 2, 6, 10, 30, 45, 48, 60) CICS/TS 3.2 and 4.1 (types 6, 10, 14, 30, 45, 48, 60) CICS/TS 4.2 (types 6, 10, 29, 30, 45, 48, 60) CICS/TS 5.1 and above (types 6, 10, 29, 30, 45, 48, 62)	ASG-TMON V1.5 (type TD) ASG-TMON V2.0 and above (type TR and type TI)
CICS User Activity File (CICCSU)		ASG-TMON V1.5 and above (type TA)

CIC Information Area Files	IBM Type 110	ASG-TMON
CICS Application Unit Activity File (CICCAU)		ASG-TMON V1.5 and above (type TA)
CICS Incident File (CICCIN)	Exception subtype 1, class 4	ASG-TMON V1.5 and above (type TA)
CICS User Application Count File (CICCAC)		ASG-TMON V1.5 and above (type TA)
CICS File/DBD Activity File (CICCSF)		ASG-TMON V1.5 and above (type TA) File Segments
CICS MRO Activity File (CICCMR)	Not available in type 110 records	MVS: Detail transaction records (type D or type A) MRO Segments ASG-TMON V1.5 and above (type TA) MRO Segments
CICS Dictionary File (CICCDC)	Data dictionary records subtype 1, class 1	Not available from ASG-TMON

Figure 6-1. CICS Analyzer File/Data Source Cross-Reference Chart

6.3 Data Element/Data Source Cross-Reference

Figures 6-2 through 6-4 provide cross-references from the CA MICS data elements in the CIC Information Area files to their corresponding input data field names. This information is useful when you are familiar with an input data source and you want to identify the CA MICS data element that contains the data field. It also helps to identify the CA MICS data elements that are unique to a data source, and it lets you to locate comparable fields when converting from one CICS monitor to another.

Because the data elements from the CICS User Activity File (CICCSU), the CICS User Application Count File (CICAC), and the CICS Application Unit Activity File (CICCAU) are all created from the same fields in the data sources, this section only provides the cross-reference for the CICCSU file.

For the CICS Transaction Server, the CA MICS Analyzer Option for CICS processes several types of CICS statistics data to populate data elements in the CICS System Activity (CICCSY) file. Therefore, figures 6-2 and 6-3 contain an additional column for STATS to map the common and CICCSY data elements to the corresponding statistics input fields.

In the figures, the following values can appear in the data source columns:

- N = not available in source data
- O = available from OMEGAMON II User Area
- X = derived by input processor (implies N)
- U = user-maintained field
- V = available, but source field varies by release;
refer to the Data Dictionary entry

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
COMMON DATA ELEMENTS IN MICS CICS FILES		CMF	STATS	MONITOR	TCE 2.0+
ABND CODE	ABEND Codes	099	N	TABENDCD	TAABNCDE
CICACTX	User Identification Fields	U	N	U	U
CICAPU	Application Unit ID	U	N	U	U
CICNETNM	MRO/ISC Network Name	097(NETUOWPX)	N	TAUOW	TAUOWLMN
CICNODST	Number of Distribution Values	U	N	U	U
CICPCKX	User Clock x	066(ICTOTCT)	N	UTDATA	TAUTDATA
CICPCTX	User Counter x	065	N	UTDATA	TAUTCNT
CICPCXX	User Clock Counter x	X	N	N	N
CICPUA01	User Retained Field 1	U	N	U	U
CICPUA02	User Retained Field 2	U	N	U	U
CICRVALX	Response Distribution Limit	U	N	U	U
CICSID	CICS System Identification	U	N	U	U
CICSMVER	CICS Monitor Version Number	X	X	TMONVER	*TAMONVER
CICSREL	CICS Release Number	X	X	TACICSLV	TACICLVL
CICUOWID	MRO/ISC Unit of Work Identification	098(NETUOWSX)	N	TAUCLK	TAUOWCTM
				TAUNUM	TAUOWNUM
JOB	Job Identification	SMFSTJBN	N	N	N
OPERID	CICS Terminal Operator Identification	003	N	TAOPID	TAOPRID
ORGSYSID	Originating System ID	X	X	TASMFSID	TASMFSID
PROGRAM	Program Name	071(PGMNAME)	N	TAPGM	TAPGM1
RTYPE	Performance Record Type	112(RTYPE)	N	N	N
TERMINAL	CICS Terminal Identifier	002(TERM)	N	TATERM	TATERID
TRANCODE	CICS Transaction Code	001(TRAN)	N	TATRANS	TAPTRAN
TRAN TYPE	Transaction Type	004(T)	N	TAFLAG2	TAFLAG2
USER	User Identification	089(USERID)	N	TAUSERID	TAUSERID
USERID	User Identification	089(USERID)	N	TAUSERID	TAUSERID

FIGURE 6-2. Common Data Element/Data Source Cross-Reference

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICCXY (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYABEND Abnormal Termination Occurrences	X	N	X	X	
CSYADSTM Address Space Cpu Time	096	DSGTCT (QR) (RO) (CO)	N	N	
CSYAMXT Current Tasks Eligible for Dispatch	N	DSGAMXTC	N	N	
CSYAMXTC Total Task During Active Max Task	N	N	TIAMXTCT	N	
CSYAMXTP Max Concurrent Dispatchable Tasks	N	DSGAMXTP	N	N	
CSYAMXTQ Maximum Queued User Transactions	N	XMGPQT	N	N	
CSYAPPGI Application Page-ins	N	N	TIAPLPGI	TIPGEICT	
CSYAPPGO Application Page-outs	N	N	TIAPLPGO	TIPGEOCT	
CSYAPPL CICS VTAM APPLID	N	N	TIENAPLD	TICICAPL	
CSYAVAMX Average Active Task Limit	078	DSGAMXTL	N	N	
CSYAVNIU Avg NIU Queue Membership Time	N	LDGDPSCS	N	N	
CSYAVCTM Avg Conversational Response Time	U	N	U	U	
CSYAVLDT Avg Program Load Time	N	LDGLLT LDGLLR	N	N	
CSYAVLTM Avg Long Response Time	U	N	U	U	
CSYAVMTM Avg Medium Response Time	U	N	U	U	
CSYAVMXT Average Maxtask Value	077	DSGTL	N	N	
CSYAVNIU Avg NIU Queue Membership Time	N	LDGDPSCS	N	N	
CSYAVQTM Average MAXTASK Queuing Time	U	N	N	N	
CSYAVSTM Avg Short Response Time	U	N	U	U	
CSYAVTTM Avg Response Time All Functions	005(START) 006(STOP)	N	TADATE TASTIME TAPTME	TATSKDTM TATSKWTM	
CSYAVUSR Average Number of Concurrent Users	N	N	N	N	
CSYCDSA CDSA Size (in K)	N	SMSDSASZ	TDIDESZE	TRDSASZE	
CSYCDSTX Count Conv. Responses Within Limit	U	N	U	U	
CSYCCOCPU Concurrent TCB CPU Time	N	DSGTCT	N	N	
CSYCODSP Concurrent TCB Dispatch Time	N	DSGTDI	N	N	
CSYCOWAT Concurrent TCB MVS Waits	N	DSGSYSW	N	N	
CSYCOWTM Concurrent TCB MVS Wait Time	N	DSGTWT	N	N	
CSYCONTM Terminal Connect Time	U	N	U	U	
CSYCCOST Processing Charges	U	N	U	U	
CSYCPJTM JCA CPU TCB Time	019	N	TIJCPU	N	
CSYCPUCT Number of CPUs Active at Startup	N	N	TIENCPU#	TIACTCT	

FIGURE 6-3. CICCXY Data Element/Data Source Cross-Reference (Part 1 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICCXY (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYCPUNI Instructions Executed	U	N	U	U	
CSYCPUTM CPU Time Consumed	016	DSGTCT	TICPUR	TITCBCTM	
	019	(OR)	TISRBCPU	TISRBCTM	
	022	(RO)			
	028	(CO)			
	081				
CSYCRLCD Cushion Release Count (CDSA)	N	N	TDIDECRC	TRDSACRC	
CSYCRLEC Cushion Release Count (ECDSA)	N	N	TDIDECRC	TRDSACRC	
CSYCRLER Cushion Release Count (ERDSA)	N	N	TDIDECRC	TRDSACRC	
CSYCRLES Cushion Release Count (ESDSA)	N	N	TDIDECRC	TRDSACRC	
CSYCRLEU Cushion Release Count (EUDSA)	N	N	TDIDECRC	TRDSACRC	
CSYCRLRD Cushion Release Count (RDSA)	N	N	TDIDECRC	TRDSACRC	
CSYCR LSD Cushion Release Count (SDSA)	N	N	TDIDECRC	TRDSACRC	
CSYCR LUD Cushion Release Count (UDSA)	N	N	TDIDECRC	TRDSACRC	
CSYCSNDP CSNC Dispatched Time	N	N	TIIDTIME	N	
CSYCSNPI CSNC Page In Count	N	N	TIIPGIN	N	
CSYCSNPO CSNC Page Out Count	N	N	TIIPGOT	N	
CSYCSNTM CSNC CPU Time	N	N	TIICPU	N	
CSYCRSTM Conversational Response Time Total	U	N	U	U	
CSYCSYTM Task Control Dispatched Elapsed Time	024	N	TIKDTIME	N	
CSYCTRM Conversational Transactions Processed	U	N	U	U	
CSYDAMSC Storage Violations	064	SMSSV	TAFLAG1	TAFLAG1	
	(TASKFLAG)		TISVDCNT	N	
CSYDBCTC DBCTL Control Task CPU Time	N	N	TIDBTC	TIDBTC	
CSYDBNSC Number of Non-SQL Calls to DB2	N	N	TINSQCSL	TINSQCCT	
CSYDBNST Total Elapsed Time of DB2 Non-SQL Calls	N	N	TINSQTIM	TINSQCTM	
CSYDBOVF DB2 Pool Overflow Count	N	N	TIPLNOVF	TIDB2OCT	
CSYDBTCB Thread TCB CPU Time (DBCTL)	N	N	TIDBTHC	TIDBCTHC	
CSYDBTHW DB2 Wait Without Thread Hit Count	N	N	TIDB2THW	TIDB2HCT	
CSYDBWAT DB2 Wait Time	N	N	TIDB2TM	TIDB2WTM	
CSYDBWCT DB2 Wait Count	N	N	TIDB2CT	TIDB2WCT	
CSYDENTM Dependent Enclave CPU Time	N	N	N	TIENCCTM	
CSYDISTM CICS/VS Dispatcher CPU Time	018	N	TIIDTIME	N	
	024		TIJDTIME		
	015		TIKDTIME		
			TIIDTIME		
			TIJDTIME		
CSYDLDMB DL/I DMB Pool Size (Byte Count)	N	N	TIENDMBP	N	
CSYDLDMH DL/I DMB Pool Use HMM for Interval	N	N	TIDMBHWI	TIDMBHWM	
CSYDLDMW DL/I DMB Pool Wait Count	N	N	TIDMBWCT	N	

FIGURE 6-3. CICCXY Data Element/Data Source Cross-Reference (Part 2 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICS (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYDLENQ DL/I Enqueue Pool Size (Byte Count)	N	N	TIENENQP	N	
CSYDLEQH DL/I DMB ENQ Pool Use HWM for Interval	N	N	TIENQHWI	TIENQHWM	
CSYDLEQW DL/I DMB ENQ Pool Wait Count	N	N	TIENQWCT	N	
CSYDLICT DL/I Calls	N	N	TIDLICCT	TIDLIRCT	
CSYDLIPW DL/I Pool Wait	N	N	X	X	
CSYDLITI DL/I Calls Elapsed Time	N	N	TIDLICTI	TIDLIRTM	
CSYDLITW DL/I Thread Wait	N	N	X	X	
CSYDLPSB DL/I PSB Pool Size (Byte Count)	N	N	TIENPSBP	N	
CSYDLPSH DL/I PSB Pool Use HWM for Interval	N	N	TIPSBHWI	TIPSBHWM	
CSYDLPSW DL/I PSB Pool Wait Count	N	N	TIPSBWCT	N	
CSYDLTHD DL/I Threads Count in SIT	N	N	TIENDLT	N	
CSYDLTHI DL/I Thread Use HWM for Interval	N	N	TIISBHWI	TIIRDHWM	
CSYDLTWC DL/I Thread Wait Count	N	N	TIITRWCT	N	
CSYDLWAT DL/I Wait Time	N	N	TIDLIWTM	TIDLIWTM	
CSYDLWCT DL/I Wait Count	N	N	TIDLIWCT	TIDLIWCT	
CSYDS1WC Total Wait 1st Dispatch Count	N	N	TIWIDSPC	N	
CSYDS1WT Total Wait 1st Dispatch Time	N	N	TIWIDSPC	N	
CSYDSA Current DSA Total (in K)	N	SMSDSATOTAL	N	N	
CSYDSAL Current DSA Limit (in K)	N	SMSDSALIMIT	N	N	
CSYDSAPZ DSA Page Size	N	N	TIPGESZE	N	
CSYDSARS DSA Size after Loading Resident Program	N	N	TIDSARES	N	
CSYDSASZ DSA Size (in K)	N	N	TIDSASZE	N	
CSYDSWAT Total Dispatch Queue Time	N	N	TIWQUEUE	TIIDSQST	
CSYDSWCT Total Transactions w/Disp Queue Time	N	N	TIWQUECT	TIIDSWRT(2.1)	
				TIIDSWRC(2.1)	
CSYDTSCT DTP (MRO) Suspend Wait Count	N	N	TIDTPSCT	N	
CSYDTSUS DTP (MRO) Suspend Wait Time	N	N	TIDTPSTM	N	
CSYDTWAT MRO/DTP Wait Time	N	N	TIDTPWTM	TIITWTMS	
CSYDTWCT MRO/DTP Wait Count	N	N	TIDTPWCT	TIITWTCT	
CSYECBWC Unidentified ECB Wait Count	N	N	TIECBCNT	N	
CSYECDSA ECDSA Size (in K)	N	SMSDSASZ	TDIDESZE	TRDSASZE	
CSYECBWT Unidentified ECB Wait Time	N	N	TIECBTIM	N	
CSYEDSA Current EDSA Total (in K)	N	SMSDSATOTAL	N	N	
CSYEDSAL Current EDSA Limit (in K)	N	SMSDSALIMIT	N	N	
CSYEQSCCT Enqueue Suspend Wait Count	N	N	TIENQTMCT	N	
CSYEQSUS Enqueue Suspend Wait Time	N	N	TIENQTMW	N	
CSYERDSA ERDSA Size (in K)	N	SMSDSASZ	TDIDESZE	TRDSASZE	
CSYERSTM Excessive Transaction Response Time	U	N	U	U	
CSYESDSA ESDSA Size (in K)	N	SMSDSASZ	TDIDESZE	TRDSASZE	
CSYETRN Excessive Transactions Processed	U	N	U	U	

FIGURE 6-3. CICS Data Element/Data Source Cross-Reference (Part 3 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICCXY (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYEUDSA EUDSA Size (in K)	N	SMSDSASZ	TDIDESZE	TRDSASZE	
CSYEXCPS I/O (EXCPs) Generated	U	N	TIEXCPCT	N	
CSYEXCTM Elapse Time of User Excluded Trans	N	N	TIEXTIME	N	
CSYEXDTM Elapse Dispatch Time User Exclude	N	N	TIEXDSP	N	
CSYFCWAT FCP File Wait Time	N	N	TIFILWMT	TIFCPWTM	
CSYFCWCT FCP File Wait Count	N	N	TIFILWCT	TIFCPWCT	
CSYFECT File Events	N	N	TIFCECNT	TIFCPRCT	
CSYFETI File Elapsed Time	N	N	TIFCETI	TIFCPRTM	
CSYFRAME Real (MVS) Storage Frame Count	N	N	TIFRAMES	N	
CSYGMSCD Storage Suspend Count (CDSA)	N	N	TDIDESUS	TRDSASUS	
CSYGMSCCT Storage Suspend Wait Count	N	N	TIGMAWCT	TIGMAWCT	
CSYGMSEC Storage Suspend Count (ECDSA)	N	N	TDIDESUS	TRDSASUS	
CSYGMSEER Storage Suspend Count (ERDSA)	N	N	TDIDESUS	TRDSASUS	
CSYGMSES Storage Suspend Count (ESDSA)	N	N	TDIDESUS	TRDSASUS	
CSYGMSEU Storage Suspend Count (EUDSA)	N	N	TDIDESUS	TRDSASUS	
CSYGMSRD Storage Suspend Count (RDSA)	N	N	TDIDESUS	TRDSASUS	
CSYGMSSD Storage Suspend Count (SDSA)	N	N	TDIDESUS	TRDSASUS	
CSYGMSUD Storage Suspend Count (UDSA)	N	N	TDIDESUS	TRDSASUS	
CSYGMSUS Storage Suspend Wait Time	N	N	TIGMATIM	TISTGWTM	
CSYHIAED Peak Storage Allocated (CDSA)	N	N	TDIDEAHI	TRDSASHI	
CSYHIAEC Peak Storage Allocated (ECDSA)	N	N	TDIDEAHI	TRDSASHI	
CSYHIAER Peak Storage Allocated (ERDSA)	N	N	TDIDEAHI	TRDSASHI	
CSYHIAES Peak Storage Allocated (ESDSA)	N	N	TDIDEAHI	TRDSASHI	
CSYHIAEU Peak Storage Allocated (EUDSA)	N	N	TDIDEAHI	TRDSASHI	
CSYHIAMC High Task I/O Request Count	N	N	TIHTAMC	TIHAITCT	
CSYHIARD Peak Storage Allocated (RDSA)	N	N	TDIDEAHI	TRDSASHI	
CSYHIASD Peak Storage Allocated (SDSA)	N	N	TDIDEAHI	TRDSASHI	
CSYHIATK Active Task HWM this Interval	N	N	TIMATHMM	N	
CSYHIAUD Peak Storage Allocated (UDSA)	N	N	TDIDEAHI	TRDSASHI	
CSYHICPU High Task CPU Consumption	N	N	TIHTCPU	TIHTRCTM	
CSYHIDSS High Task Data Stream Size	N	N	TIHTDSS	*TIHDSSZ	
CSYHIFCW High Task File Request Wait Time	N	N	TIHTAMR	TIHWFWTM	
CSYHIPAG High Task Paging Count	N	N	TIHTPAGE	N	
CSYHIQCD Peak Storage Request Queued (CDSA)	N	N	TDIDEQHI	TRDSAQHI	
CSYHIQEC Peak Storage Request Queued (ECDSA)	N	N	TDIDEQHI	TRDSAQHI	
CSYHIQER Peak Storage Request Queued (ERDSA)	N	N	TDIDEQHI	TRDSAQHI	
CSYHIQES Peak Storage Request Queued (ESDSA)	N	N	TDIDEQHI	TRDSAQHI	
CSYHIQEU Peak Storage Request Queued (EUDSA)	N	N	TDIDEQHI	TRDSAQHI	
CSYHIQRD Peak Storage Request Queued (RDSA)	N	N	TDIDEQHI	TRDSAQHI	

FIGURE 6-3. CICCXY Data Element/Data Source Cross-Reference (Part 4 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICCXY (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYHIQSD Peak Storage Request Queued (SDSA)	N	N	TDIDEQHI	TRDSAQHI	
CSYHIQUD Peak Storage Request Queued (UDSA)	N	N	TDIDEQHI	TRDSAQHI	
CSYHIRES High Task Response Time	N	N	TIHTRT	TIHRSRTM	
CSYHISTG High Task Storage Utilization	N	N	TIHTSTG	TIHIOHMM	
CSYHITSK High Task Level This Interval	N	N	TIHITASK	TIHINTHM	
CSYHTAMC Trans ID of Task w/Highest I/O Request	N	N	TIHTAMCT	TIHAITID	
CSYHTCPU Trans ID of Task with Highest CPU	N	N	TIHTCPUT	TICPUTID	
CSYHTDSS Trans ID of Task w/Highest Data Stream	N	N	TIHTDSST	TIHDSTID	
CSYHTFCW Trans ID of Task w/Highest File Wait	N	N	TIHTAMRF	TIHWWFFID	
CSYHTPAG Trans ID of Task w/Highest Paging	N	N	TIHPAGET	N	
CSYHTRES Trans ID of Task w/Highest Resp Time	N	N	TIHTRTT	TIHRSRTM	
CSYHTSTG Trans ID of Task w/Highest Storage	N	N	TIHTSTG	TIHTIOID	
CSYICSCCT ICP Suspend Wait Count	N	N	TIICPSCCT	N	
CSYICCSUS ICP Suspend Wait Time	N	N	TIICPSTMT	N	
CSYICVSD ICVTSD (Term Scan Delay) Value	N	DSGICVSC	N	N	
CSYIDLTM Idle (Non-Dispatched) Time	016	N	TICPUR	TITCBCTM	
	019		TISRBCPU	TISRBCCTM	
	022				
	028				
	081				
CSYIMSGS Input Messages Issued	034(TCMGGIN1)N		TATERMIN	TATERICT	
	067(TCMGGIN2)				
CSYINCH Input Message Character Traffic	083(TCCHRIN1)N		TATERMIL	TATERILG	
	085(TCCHRIN2)				
CSYINVPA Invalid CMF Call Encountered	064(TASKFLAG)N		N	N	
CSYIRCNT MRO Function Ship Requests	N	N	TIIMROCNT	N	
CSYIRWAT MRO/IRC Wait Time	N	N	TIIMROIR	TIIWTTMS	
CSYIRWCT MRO/IRC Event Wait Count	N	N	TIIMROIR	TIIWTTCT	
CSYISSCT ISC (MRO) Suspend Wait Count	N	N	TIISPSCCT	N	
CSYISSUS ISC (MRO) Suspend Wait Time	N	N	TIISPSTMT	N	
CSYISWAT MRO/ISC Wait Time	N	N	TIIMROTIM	TIIWTTMS	
CSYJCWAT Journal Wait Time	N	N	TIJCWTM	TIJCPWTM	
CSYJCWCT Journal Wait Count	N	N	TIJCWCT	TIJCPWCT	
CSYJECT Journal Elapsed Events	N	N	TIJCECNT	TIJCPRTM	
CSYJETI Journal Elapsed Time	N	N	TIJCETI	TIJCPRCT	
CSYJOB CICS Job Name	N	N	MMDJOB	LMRKJOBN	
CSYJPGI Journal Control Program Page-ins	N	N	TIJPGIN	N	
CSYJPGO Journal Control Program Page-outs	N	N	TIJPGOT	N	
CSYJ8CPU J8 Owning Mode TCB CPU Time	N	DSGTCT	N	N	

FIGURE 6-3. CICCXY Data Element/Data Source Cross-Reference (Part 5 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICCXY (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYJ8DSP J8 Owning Mode TCB Dispatch Time	N	DSGTD	N	N	
CSYJ8WAT J8 Open Mode MVS Waits	N	DSGYSW	N	N	
CSYJ8WTM J8 Owning Mode TCB MVS Wait Time	N	DSGTWT	N	N	
CSYKCT00 Monitor Relative Collection Time	006(STOP)	N	TMMITIME	N	
CSYKPGI Task Control Program Page-ins	N	N	TIKPGIN	N	
CSYKPGO Task Control Program Page-outs	N	N	TIKPGOT	N	
CSYLDSTx Count Long Responses Within Limit	U	N	U	U	
CSYLDWAT Number of Requests Waiting for Loader	N	LDGWLR	N	N	
CSYLOWCD Free Storage Low-water Mark (CDSA)	N	N	TDIDEFLO	TRDSAFLO	
CSYLOWEC Free Storage Low-water Mark (ECDSA)	N	N	TDIDEFLO	TRDSAFLO	
CSYLOWER Free Storage Low-water Mark (ERDSA)	N	N	TDIDEFLO	TRDSAFLO	
CSYLOWES Free Storage Low-water Mark (ESDSA)	N	N	TDIDEFLO	TRDSAFLO	
CSYLOWEU Free Storage Low-water Mark (EUDSA)	N	N	TDIDEFLO	TRDSAFLO	
CSYLOWRD Free Storage Low-water Mark (RDSA)	N	N	TDIDEFLO	TRDSAFLO	
CSYLOWSD Free Storage Low-water Mark (SDSA)	N	N	TDIDEFLO	TRDSAFLO	
CSYLOWUD Free Storage Low-water Mark (UDSA)	N	N	TDIDEFLO	TRDSAFLO	
CSYLRSTM Long Response Time Total	U	N	U	U	
CSYLSRB0 LSR Pool 0 Buffer Wait Count	N	N	TILPL0BW	N	
CSYLSRB1 LSR Pool 1 Buffer Wait Count	N	N	TILPL1BW	N	
CSYLSRB2 LSR Pool 2 Buffer Wait Count	N	N	TILPL2BW	N	
CSYLSRB3 LSR Pool 3 Buffer Wait Count	N	N	TILPL3BW	N	
CSYLSRB4 LSR Pool 4 Buffer Wait Count	N	N	TILPL4BW	N	
CSYLSRB5 LSR Pool 5 Buffer Wait Count	N	N	TILPL5BW	N	
CSYLSRB6 LSR Pool 6 Buffer Wait Count	N	N	TILPL6BW	N	
CSYLSRB7 LSR Pool 7 Buffer Wait Count	N	N	TILPL7BW	N	
CSYLSRS0 LSR Pool 0 String Wait Count	N	N	TILPL0SW	N	
CSYLSRS1 LSR Pool 1 String Wait Count	N	N	TILPL1SW	N	
CSYLSRS2 LSR Pool 2 String Wait Count	N	N	TILPL2SW	N	
CSYLSRS3 LSR Pool 3 String Wait Count	N	N	TILPL3SW	N	
CSYLSRS4 LSR Pool 4 String Wait Count	N	N	TILPL4SW	N	
CSYLSRS5 LSR Pool 5 String Wait Count	N	N	TILPL5SW	N	
CSYLSRS6 LSR Pool 6 String Wait Count	N	N	TILPL6SW	N	
CSYLSRS7 LSR Pool 7 String Wait Count	N	N	TILPL7SW	N	
CSYLTRN Long Transactions Processed	U	N	U	U	
CSYL8CPU L8 Owning Mode TCB CPU Time	N	DSGTCT	N	N	
CSYL8DSP L8 Owning Mode TCB Dispatch Time	N	DSGTD	N	N	
CSYL8WAT L8 Open Mode MVS Waits	N	DSGYSW	N	N	
CSYL8WTM L8 Owning Mode TCB MVS Wait Time	N	DSGTWT	N	N	
CSYMCTSK Maximum Concurrent Tasks	N	DSGPNT	N	N	

FIGURE 6-3. CICCXY Data Element/Data Source Cross-Reference (Part 6 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICCXY (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYMDSTX Count Medium Responses Within Limit	U	N	U	U	
CSYMEMSE Hi-Water Dynamic Area Memory (EDSA)	026	V	N	N	
CSYMEMSZ Hi-Water Dynamic Area Memory	026	V	TIIDYNHWN	N	
CSYMIISCT Mirror Suspend Wait Count	N	N	TIICMISCT	N	
CSYMIISUS Mirror Suspend Wait Time	N	N	TIICMISTM	N	
CSYMPCPU Max CPU Usage	016	N	TIICPUR	TIICBCTM	
	019		TIISRBCTM	TIISRBCTM	
	022				
	028				
	081				
CSYMLKQ MRO Link Queuing Count	N	N	TIIMLKUCT	N	
CSYMR0CT MRO Request Count	N	N	TIISPRCT	TIISPRCT	
CSYMR0TM MRO Request Time	N	N	TIISPRTM	TIISPRTM	
CSYMRSTM Medium Response Time Total	U	N	U	U	
CSYMSG5 Total Messages Issued (in/out)	034(TCM5GIN1)	N	TIITRMSG	N	
	067(TCM5GIN2)				
	035(TCM5G0U1)				
	068(TCM5G0U2)				
CSYMTRN Medium Transactions Processed	U	N	U	U	
CSYMXAMX Maximum Active Task Limit	078	V	N	N	
CSYMXBWC Maximum Temp Storage Buffer Waits	N	A12BUWTH	N	N	
CSYMXCDS Peak Page Storage Used (CDSA)	U	V	U	TRDSASHI	
CSYMXCTM Max Conversational Response Time	U	N	U	U	
CSYMXECD Peak Page Storage Used (ECDSA)	U	V	U	TRDSASHI	
CSYMXERD Peak Page Storage Used (ERDSA)	U	V	U	TRDSASHI	
CSYMXESD Peak Page Storage Used (ESDSA)	U	V	U	TRDSASHI	
CSYMXEUD Peak Page Storage Used (EUDSA)	U	V	U	TRDSASHI	
CSYMXETM Max Excessive Response Time	U	N	U	U	
CSYMXLDS Max Loader Suspended Tasks	N	LDGWL RHW	N	N	
CSYMXLTM Max Long Response Time	U	N	U	U	
CSYMXMTM Max Medium Response Time	U	N	U	U	
CSYMXMXT Maximum Maxtask Value	077	V	N	N	
CSYMXRDS Peak Page Storage Used (RDSA)	U	V	U	TRDSASHI	
CSYMXSDS Peak Page Storage Used (SDSA)	U	V	U	TRDSASHI	
CSYMXSTM Max Short Response Time	U	N	U	U	
CSYMXSTU Maximum Temp Storage String in Use	N	A12NVCAH	N	N	
CSYMXSWC Maximum Temp Storage String Waits	N	A12VUWTH	N	N	
CSYMXTCT Total Task During Max Task	N	N	TIIMXTCT	N	
CSYMXUDS Peak Page Storage Used (UDSA)	U	V	U	TRDSASHI	

FIGURE 6-3. CICCXY Data Element/Data Source Cross-Reference (Part 7 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICS (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYMXTSK Max Tasking Condition	064	DSGTAMXT	N	N	
CSYMXTTM Max Response Time All Functions	005(START)	N	TADATE	TATSKDTM	
	006(STOP)		TASTIME	TATSKWTM	
			TAPTIME		
CSYMXUSR Maximum Number of Concurrent Users	N	N	N	N	
CSYNIURE Number of Reclaims form NIU Chain	N	V	N	N	
CSYNIUTM Not in Use (NIU) Queue Membership TIme	N	LDGDSPC	N	N	
CSYNMSIT Last SIT Suffix this Timespan	076	N	N	N	
CSYNRAMX Cumulative Active Maximum Tasks	078	DSGAMXT	TIENAMX	N	
CSYNRINT Number of Measurement Intervals	X	X	X	X	
CSYNRJCD JCP Dispatch Count	019	N	N	N	
CSYNRKCD KCP Dispatch Count	024	N	N	N	
CSYNRMXT Cumulative Maximum Tasks	077	V	TIENMXT	N	
CSYNRTCB Number of CICS Dispatcher TCBS	N	DSGNTCB	N	N	
CSYNRTCD TCP Dispatch Count	016	N	N	N	
CSYNRTSK Current Number of Task	N	DSGCNT	N	N	
CSYNRUSR Number of Concurrent Users	N	N	N	N	
CSYNUCSZ Size in Bytes of CICS Nucleus	N	N	TINUCSZ	N	
CSYNZZTM Normalized zIIP/zAAP CPU TIME	N	N	N	N	
CSYOCPTC User Task Spec Proc Eligible CPU Count	437(OFFLCPUT)	N	N	TISTCPOC	
CSYOMSGS Output Messages Issued	035(TCMMSGOU1)	N	TATERMOT	N	
	068(TCMMSGOU2)				
CSYOSCAV OSCOR Bytes Available	N	N	TIOPFAV	N	
CSYOSCFA Total Number of OSCOR Free Areas	N	N	TIOPFAF	N	
CSYOSCHI Highest OSCOR Used	N	N	TIOPHIUZ	N	
CSYOSCLF Largest OSCOR Free Area	N	N	TIOPLF	N	
CSYOSCNF Total Non Region OSCOR Free Area	N	N	TIOPNRFA	N	
CSYOSCOT OSCOR Total Bytes	N	N	TIOPSCOT	N	
CSYOSCTO Total OSCOR Available at Startup	N	N	TIOPSTOT	N	
CSYOUTCH Output Messages Character Traffic	084(TCCHROU1)	N	TATERMOL	N	
	086(TCCHROU2)				
CSYPAGIN Page Ins	029	N	TIIPGIN	N	
			TIKPGIN	N	
			TIJPGIN	N	
			TIAPLGIN	TIPGEICT	
			TIIPGIN	N	
			TIVPGIN	N	
CSYPAGOT Page Outs	030	N	TIIPGOT	N	
			TIKPGOT	N	
			TIJPGOT	N	
			TIAPLGOT	TIPGEOCT	
			TIIPGOT	N	
			TIVPGOT	N	

FIGURE 6-3. CICS SY Data Element/Data Source Cross-Reference (Part 8 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICCXY (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYPCCPU Pct CPU Usage	016 019 022 028 081	N	TICPUR TISRBCPU	TITCBCTM TISRBCTM	
CSYPCCRZ Cumulative Pct Conv Resp Within Limit	U	N	U	U	
CSYPCLRX Cumulative Pct Long Resp Within Limit	U	N	U	U	
CSYPCMDC Number of Program Compressions	094	V	TIPGMCM	N	
CSYPCMRX Cumulative Pct Medium Resp Within Limit	U	N	U	U	
CSYPCOM Program Compressions	N	N	TIALERT1	N	
CSYPCSRX Cumulative Pct Short Resp Within Limit	U	N	U	U	
CSYPCTRX Cumulative Pct All Resp Within Limit	U	N	U	U	
CSYPCWAT Program Fetch Wait Time	N	LDGTTW	TIPCWTM	TIPGMWTM	
CSYPCWCT Program Fetch Wait Count	N	LDRWTDLR	TIPCWCT	TIPGMWCT	
CSYPETTM Dispatched Time	015 018 021 024	X/V	TIAPLDSP TIIDTIME TIJDTIME TIKDTIME TIIDTIME TIIVDTIME	TITSKDTM	
CSYPGMCT Program Load Count	N	LDGLLR	TIPGMLCT	TIPGMRCT	
CSYPGMTI Program Load Time	N	LDGLLT	TIPGMLTI	TIPGMRTM	
CSYPMWAT Preempt Wait Time	N	N	TIPMTWTM	N	
CSYPMWCT Preempt Wait Count	N	N	TIPMTWCT	N	
CSYQRCPU Quasi-reentrant TCB CPU Time	N	DSGTCT	N	N	
CSYQRDSP Quasi-reentrant TCB Dispatch Time	N	DSGTDI	N	N	
CSYQRSTR Quasi-reentrant Dispatcher Start	N	DSGSTAR	N	N	
CSYQRWAT Quasi-reentrant TCB MVS Waits	N	DSGSYSW	N	N	
CSYQRWTM Quasi-reentrant TCB MVS Wait Time	N	DSGTWT	N	N	
CSYRDSA RDSA Size (in K)	N	SMSDSASZ	TDIDESZE	TRDSASZE	
CSYRESCT Resident CICS Program Count	N	N	TIRESCT	N	
CSYRESSZ Resident CICS Program Size in Bytes	N	N	TIRESSZ	N	
CSYREST1 Total Trans w/0-1 Second Response Time	N	N	TIRESP1	TIRPS01	
CSYREST2 Total Trans w/1-2 Second Response Time	N	N	TIRESP2	TIRPS12	
CSYREST3 Total Trans w/2-3 Second Response Time	N	N	TIRESP3	TIRPS23	
CSYREST4 Total Trans w/3-4 Second Response Time	N	N	TIRESP4	TIRPS34	
CSYREST5 Total Trans w/4-5 Second Response Time	N	N	TIRESP5	TIRPS45	
CSYREST6 Total Trans over 5 Second Response Time	N	N	TIRESP6	TIRPS05	
CSYRGNS Region size	N	N	TIARGNS	N	
CSYRJDTM Total Elapse Dispatch Time User Reject	N	N	TIURDSP	N	
CSYRJTCT Total User Rejected Transactions	N	N	TIURCNT	N	

FIGURE 6-3. CICCXY Data Element/Data Source Cross-Reference (Part 9 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICS (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYRJTTM Total Elapse Time of Rejected Trans	N	N	TIURTIME	N	
CSYROCPU Resource-owning TCB CPU Time	N	DSGTCT	N	N	
CSYRODSP Resource-owning TCB Dispatch Time	N	DSGTD	N	N	
CSYROWAT Resource-owning TCB MVS Waits	N	DSGSYSW	N	N	
CSYROWTM Resource-owning TCB MVS Wait Time	N	DSGTWT	N	N	
CSYSCPTM User Task Standard Processor CPU Time	N	N	N	TISTCPUT	
CSYSDSA SDSA Size (in K)	N	SMDSASZ	TDIDESZE	TRDSASZE	
CSYSDSTX Count Short Responses Within Limit	U	N	U	U	
CSYSERVU Service Units	U	N	U	U	
CSYSIOCT Access Method Calls (sio Count)	070(FCAMCT)	N	TAFILCT	TAFPCRCT	
CSYSLCPU Sockets Owning Mode (SL) TCB CPU Time	N	DSGTCT	N	N	
CSYSLDSP Sockets Owning Mode (SL) TCB Dispatch	N	DSGTD	N	N	
CSYSLWAT Sockets Owning Mode (SL) MVS Waits	N	DSGSYSW	N	N	
CSYSLWTM Sockets Owning Mode (SL) TCB MVS Wait	N	DSGTWT	N	N	
CSYS0CPU Sockets Owning Mode (S0) TCB CPU Time	N	DSGTCT	N	N	
CSYS0DSP Sockets Owning Mode (S0) TCB Dispatch	N	DSGTD	N	N	
CSYS0S Short On Storage Condition	064	SMSS0S	TAFLAG1	TAFLAG1	
			TIS0SCNT	N	
CSYS0SCD Short On Storage Condition (CDSA)	N	SMSS0S	TDIDESSC	TRDSASSC	
CSYS0SEC Short On Storage Condition (ECDSA)	N	SMSS0S	TDIDESSC	TRDSASSC	
CSYS0SER Short On Storage Condition (ERDSA)	N	SMSS0S	TDIDESSC	TRDSASSC	
CSYS0SES Short On Storage Condition (ESDSA)	N	SMSS0S	TDIDESSC	TRDSASSC	
CSYS0SEU Short On Storage Condition (EUDSA)	N	SMSS0S	TDIDESSC	TRDSASSC	
CSYS0SRD Short On Storage Condition (RDSA)	N	SMSS0S	TDIDESSC	TRDSASSC	
CSYS0SSD Short On Storage Condition (SDSA)	N	SMSS0S	TDIDESSC	TRDSASSC	
CSYS0SUD Short On Storage Condition (UDSA)	N	SMSS0S	TDIDESSC	TRDSASSC	
CSYS0WAT Sockets Owning Mode (S0) MVS Waits	N	DSGSYSW	N	N	
CSYS0WTM Sockets Owning Mode (S0) TCB MVS Wait	N	DSGTWT	N	N	
CSYSPCOM Program Storage Compressions	088	N	N	N	
CSYSQLTM Total Elapse Time in SQL Calls	N	N	TISQLTIM	TISQLCTM	
CSYSQLTO Total SQL Calls	N	N	TISQLCLS	TISQLCCT	

FIGURE 6-3. CICS SY Data Element/Data Source Cross-Reference (Part 10 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICCXY (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYSRBTM CPU SRB Time	028	N	TISRBCPU	TISRBCMT	
CSYSRMTR SRM Ended Transactions	U	N	U	U	
CSYSRSTM Short Response Time Total	U	N	U	U	
CSYSRU System Resource Units	U	N	U	U	
CSYSTDAT Date of Startup	N	N	TIESDATE	TIDCSDA	
CSYSTRN Short Transactions Processed	U	N	U	U	
CSYSUBTM Total Subtask TCB Time	N	N	TISUBTCB	TISUBTTM	
CSYSUSOC Unidentified Suspend Wait Count	N	N	TISPNDCT	N	
CSYSUSOT Unidentified Suspend Wait Time	N	N	TISPNDTM	N	
CSYSZCPU Secondary LU TCB CPU Time	N	DSGACT	N	N	
CSYSZDSP Secondary LU TCB Dispatch Time	N	DSGTDI	N	N	
CSYSZWAT Secondary LU TCB MVS Waits	N	DSGYSW	N	N	
CSYSZWMT Secondary LU TCB MVS Wait Time	N	DSGTWT	N	N	
CSYS8CPU S8 Sockets Mode TCB CPU Time	N	DSGTCT	N	N	
CSYS8DSP S8 Sockets Mode TCB Dispatch Time	N	DSGTDI	N	N	
CSYS8WAT S8 Sockets Mode MVS Waits	N	DSGYSW	N	N	
CSYS8WMT S8 Sockets Mode TCB MVS Wait Time	N	DSGTWT	N	N	
CSYTCBTM Total CICS TCB CPU Time	016	N	TICPUR	TITCBCTM	
	019				
	022				
	081				
CSYTCPTJ Accumulated TCP + JCA CPU Time	016	N	TITCPU	N	
	019		TIJCPU	N	
CSYTCPTC User Task Standard Processor CPU Count	N	N	N	TISTCPUC	
CSYTCPTM Terminal Control TCB CPU Time	016	N	TITCPU	N	
CSYTCUTM User TCB CPU Time	022	N	TIAPLCPU	TIAPLCTM	
CSYTCWAT Total Terminal Wait Time	N	N	TITCWTIM	TITERWTM	
CSYTCWCT Total Terminal Wait Count	N	N	TITCWCNT	TITERWCT	
CSYTDECN Transient Data (Extra) Request Count	N	N	TITDECNT	N	
CSYTDECT Transient Data Events	N	N	TITDICNT	TITDQRCT	
			TITDECNT		
CSYTDETI Transient Data Elapsed Time	N	N	TITDITI	TITDQRTM	
			TITDETI		
CSYTDETM Transient Data (Extra) Request Time	N	N	TITDETI	N	
CSYTDEWC Transient Data (Extra) Wait Count	N	N	TITDEWCT	N	
CSYTDEWT Transient Data (Extra) Wait Time	N	N	TITDEWTM	N	
CSYTDICT Transient Data (Intra) Request Count	N	N	TITDICNT	N	
CSYTDITM Transient Data (Intra) Request Time	N	N	TITDITI	N	

FIGURE 6-3. CICCXY Data Element/Data Source Cross-Reference (Part 11 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICS SY (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYTDIWC Transient Data (Intra) Wait Count	N	N	TIIDIWCT	N	
CSYTDIWT Transient Data (Intra) Wait Time	N	N	TIIDIWTM	N	
CSYTDSTX Count Total Responses Within Limit	U	N	U	U	
CSYTMERR Monitor Timing Error Encountered	064	N	N	N	
CSYTMICV Interval Control Value	074	DSGICVT	N	N	
CSYTMJCD JCP Dispatched Time	018	N	TIJDTIME	N	
CSYTMKCD KCP Dispatched Time	024	N	TIKDTIME	N	
CSYTMTCB TCP Dispatched Time	015	N	TIIDTIME	N	
CSYMTSD Terminal Scan Delay Time	079	DSGICVS	N	N	
CSYTOQTM Total Time Queued due to MXT	N	XMGQTQM	N	N	
CSYTODT Total Delayed Transactions due to MXT	N	XMGDT	N	N	
CSYTOTE Transactions Excluded	N	N	TIITOTE	TITRNXCT	
CSYTOTI Transactions Initiated	N	N	TIITOTI	N	
CSYTOTT Nonexcluded Transactions Ended	N	N	TIITOTT	N	
CSYTPGI Terminal Control Program Page-ins	N	N	TIITPGIN	N	
CSYTPGO Terminal Control Program Page-outs	N	N	TIITPGOT	N	
CSYTRANS Total Transactions Ended	X	N	X	X	
CSYTRSTM Total Response Time All Functions	005(START) 006(STOP)	N	TADATE TASTIME TAPTIME	TATSKDTM TATSKWTM	
CSYTSATM Temp Storage Aux (Output) Request Time	N	N	TIISATIM	N	
CSYSAUX Temporary Auxiliary Storage Used	N	N	TIISAXU	N	
CSYSAWC Temp Storage Aux (Output) Wait Count	N	N	TIISAWCT	N	
CSYSAWT Temp Storage Aux (Output) Wait Time	N	N	TIISAWTM	N	
CSYTSBWC Temp Storage Buffer Waits	N	A12BWTN	N	N	
CSYSECT Temporary Storage Events	N	N	TIISBCT	N	
			TIISACNT	N	
			TIISMCNT	N	
CSYSETI Temporary Storage Time	N	N	TIISATIM	N	
			TIISBTIM	N	
CSYTRSTM Total Response Time All Functions	005(START) 006(STOP)	N	TADATE TASTIME TAPTIME	N N N	
CSYTSATM Temp Storage Aux (Output) Request Time	N	N	TIISATIM	N	
CSYSAUX Temporary Auxiliary Storage Used	N	N	TIISAXU	N	
CSYSAWC Temp Storage Aux (Output) Wait Count	N	N	TIISAWCT	N	
CSYSAWT Temp Storage Aux (Output) Wait Time	N	N	TIISAWTM	N	
CSYTSBWC Temp Storage Buffer Waits	N	A12BWTN	N	N	

FIGURE 6-3. CICS SY Data Element/Data Source Cross-Reference (Part 12 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICS (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYTSECT Temporary Storage Events	N	N	TITSBICT	N	
			TITSACNT	N	
			TITSMCNT	N	
CSYTSETI Temporary Storage Time	N	N	TITSATIM	N	
			TITSBTIM	N	
CSYTSICT Temp Storage (Aux+Main) Input Requests	N	N	TITSBICT	N	
CSYTSIM Temp Storage (Main) Input Requests	N	A12NMG	N	N	
CSYTSIOE Number of Temp Storage I/O Errors	N	A12STA AF	N	N	
CSYTSITA Temp Storage (Aux) Input Requests	N	A12NAG	N	N	
CSYTSITM Temp Stor (Aux+Main) Input Request Time	N	N	TITSBTIM	N	
CSYTSIWC Temp Stor (Aux+Main) Input Wait Count	N	N	TITSBWCT	N	
CSYTSIWT Temp Storage (Aux+Main) Input Wait Time	N	N	TITSBWTM	N	
CSYTSMAN Temporary Main Storage Used	N	N	TITSMAN	N	
CSYTSMMX Maximum Temp Storage (Main) Used	N	A12STA6F	N	N	
CSYTSMMN Maximum Temp Storage Name in Use	N	A12QNUMH	N	N	
CSYTSMMQ Maximum Entries in Temp Storage Queue	N	A12QINH	N	N	
CSYTSNSP NOSPACE Condition in Temp Storage Aux	N	A12STA8F	N	N	
CSYTSOTA Temp Storage (Aux) Output Requests	N	A12STA7F	TITSACNT	N	
CSYTSOTM Temp Storage (Main) Output Requests	N	A12STA5F	TITSMCNT	N	
CSYTSQCR Number of Temp Storage Queues Created	N	A12STA3F	N	N	
CSYTSQEC No. Temp Stor Queue Extension Created	N	A12STA4F	N	N	
CSYTSRTM Task Control CPUR Time	081	N	TIKCPU	N	
CSYTSWC Temp Storage String Waits	N	A12VWTN	N	N	
CSYTSWCI Number of Write Length Greater Than CI	N	A12STABG	N	N	
CSYTTIM Nonexcluded Transaction Time	N	N	TITIME	N	
CSYUDBCT User Database Request Count	N	N	TIUDBCCT	TIUDBRCT	
CSYUDBTM User Database Request Time	N	N	TIUDBCTI	TIUDBRTM	
CSYUDSA UD SA Size (in K)	N	SMDSASZ	TDIDESZE	TRDSASZE	
CSYUDWAT User Database Wait Time	N	N	TIUDBWTM	N	
CSYUDWCT User Database Wait Count	N	N	TIUDBWCT	N	
CSYUPTM CICS Availability Time	005 (START)	N	TMMIDATE	TISTRDTS	
	006 (STOP)		TMMITIME	TIENDDTS	
			TIESTIME		
CSYUSRTM User (Application) CPUR Time	021	N	TIAPLDSP	TITSKDTM	
CSYVCPU CVST CPU Time	N	N	TIVCPU	N	
CSYVDTIM CVST Dispatched Time	N	N	TIVDTIME	N	
CSYVERSN Version and Release Number	N	N	TMMIVREL	N	
CSYVILCD Storage Violations (CDSA)	N	SMSSV	TDIDESVD	TRDSASVD	
CSYVILEC Storage Violations (ECDSA)	N	SMSSV	TDIDESVD	TRDSASVD	
CSYVILER Storage Violations (ERDSA)	N	SMSSV	TDIDESVD	TRDSASVD	

FIGURE 6-3. CICS SY Data Element/Data Source Cross-Reference (Part 13 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE					
THE CICS (CICS SYSTEM ACTIVITY) FILE	CMF	STATS	MONITOR	TCE 2.0+	
CSYVILES Storage Violations (ESDSA)	N	SMSSV	TDIDESVD	TRDSASVD	
CSYVILEU Storage Violations (EUDSA)	N	SMSSV	TDIDESVD	TRDSASVD	
CSYVILRD Storage Violations (RDSA)	N	SMSSV	TDIDESVD	TRDSASVD	
CSYVILSD Storage Violations (SDSA)	N	SMSSV	TDIDESVD	TRDSASVD	
CSYVILUD Storage Violations (UDSA)	N	SMSSV	TDIDESVD	TRDSASVD	
CSYVPGI CVST Page-ins	N	N	TIVPGIN	N	
CSYVPGO CVST Page-outs	N	N	TIVPGOT	N	
CSYVSMBW VSAM Buffer Wait	N	N	X	X	
CSYVMSW VSAM String Wait	N	N	X	X	
CSYWTCPU CPU across OPSYS WAIT	N	N	TIWATCPU	N	
CSYWTIM CICS Dispatcher Wait Time	027	DSGTWT	TISYSWAT	N	
		(QR)			
		(RO)			
		(CO)			
CSYXACOR Total CICS Global XA Storage Used	N	N	TIXACORE	N	
CSYXCSTM Task CPU TIME withiyt zIIP/zAAP Eligible	N	N	N	N	
CSYXTSTM TCB CPU TIME without zIIP/zAAP Eligible	N	N	N	N	
CSYZZCTM ZIIP/ZAAP Eligible CPU TIME on CP	437(OFFLCPUT)	N	N	TASTCPOT	

FIGURE 6-3. CICSY Data Element/Data Source Cross-Reference (Part 14 of 14)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICSU (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUABEND Abnormal Termination Occurrences	099	TABENDCD	TAABNCDE	
CSUACTID Activity ID	203(ACTVTYID)	N	N	
CSUACTNM Activity Name	204(ACTVTYNM)	N	N	
CSUADATO Total ADABAS Calls	MADABN(OMEG)	N	N	
CSUADAWT Total ADABAS Wait Time	MADABT(OMEG)	N	N	
CSUAPPL CICS VTAM APPLID	X	N	N	
CSUAPLNM Application Name in Appl Context Data	451(ACAPPLNM)	N	N	
CSUAVCTM Avg Conversational Response Time				
CSUCRSTM				
CSUCTRN				
CSUAVINC Average Input Character Traffic				
CSUINCH				
CSUTRANS				
CSUAVLTM Avg Long Response Time				
CSULRSTM				
CSULTRN				
CSUAVMEM Average Transaction Memory Used				
CSUMEMRC				
CSUTRANS				
CSUAVMTM Avg Medium Response Time				
CSUMRSTM				
CSUMTRN				
CSUAVOPS Average Number of CICS Calls				
CSUTLOPS				
CSUTRANS				
CSUAVOTC Average Output Character Traffic				
CSUOUTCH				
CSUTRANS				
CSUAVSTM Avg Short Response Time				
CSUSRSTM				
CSUSTRN				
CSUAVTRT Average Transaction CPU Time				
CSUCPUTM				
CSUTRANS				
CSUAVTTM Avg Response Time				
CSUTRSTM				
CSUTRANS				

FIGURE 6-4. CICSU Data Element/Data Source Cross-Reference (Part 1 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICCUS (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUAVWAT Average CICS Wait Time				
CSUTLWAT				
CSUTRANS				
CSUBAADC No. Activity Data Container Requests	217(BAACDCCT)	N	N	
CSUBAAPC No. Acquire Process Requests	214(BAACQPCT)	N	N	
CSUBADAC No. Define Activity Requests	209(BADACTCT)	N	N	
CSUBADIC No. Define Input Event Requests	220(BADFIECT)	N	N	
CSUBADPC No. Define Process Requests	208(BADPROCT)	N	N	
CSUBADTC No. Define Timer Event Requests	221(BATIAECT)	N	N	
CSUBALKC No. Link Process/Activity Requests	207(BALKPACT)	N	N	
CSUBAPDC No. Process Data Container Requests	216(BAPRDCCT)	N	N	
CSUBARAC No. Run Process/Activity Async	206(BARASYCT)	N	N	
CSUBAREC No. Retrieve Reattach Event Requests	219(BARATECT)	N	N	
CSUBARSC No. Run Process/Activity Sync	205(BARSYNCT)	N	N	
CSUBATCC Total No. Data Container Requests	218(BATOTCCT)	N	N	
CSUBATEC Total No. Event Requests	222(BATOTECT)	N	N	
CSUBATPC Total No. Process/Activity Requests	215(BATOTPCT)	N	N	
CSUBDCPC No. Delete and Cancel Activity Requests	213(BADCPACT)	N	N	
CSUBMOPS Number of CICS BMS Calls	050(BMSMAPCT)	N	N	
	051(BMSINCT)			
	052(BMSOUTCT)			
CSUBRPAC No. Resume Process/Activity Requests	212(BARMPACT)	N	N	
CSUBRSAC No. Reset Activity Requests	210(BARSPACT)	N	N	
CSUBRTRN 3270 Bridge Transaction ID	124(BRDGTRAN)	N	N	
CSUBSPAC No. Suspend Process/Activity Requests	211(BASUPACT)	N	N	
CSUCBSRV CorbaServer Handling Requests	311(CBSRVNRM)	N	N	
CSUCDSTX Count Conv. Responses Within Limit	U	U	U	
CSUCFACT No. 00 Class Library API Requests	025(CFCAPICT)	N	N	
CSUCFDTM CF Data Table Wait Time	176(CFDTWAIT)	N	N	
CSUCHMDC No. CICS Dispatcher Change Modes	248(CHMODECT)	N	N	
CSUCINP Input Characters - Primary Facility	083(TCCHRIN1)	N	N	
CSUCIN62 Input Characters - Alt. Facility. LU6.2	137(TCC62IN2)	N	N	
CSUCIPAD Client IP Address	244(CLIPADDR)	N	N	
CSUCOUTP Output Characters - Primary Facility	084(TCCHROU1)	N	N	
CSUCINS Input Characters - Secondary Facility	085(TCCHRIN2)	N	N	
CSUCONWT DB2 Connection Wait Time	188(DB2CONWT)	N	N	
CSUCOU62 Output Characters - Alt. Facility. LU6.2	138(TCC62OU2)	N	N	
CSUCOUTS Output Characters - Secondary Facility	086(TCCHROU2)	N	N	
CSUCLKOP Number Of Clocks Generated	U	U	U	
CSUCLOTM Elapsed Time in SQL Close Cursors	MBCLOT(OMEG)	U	U	

FIGURE 6-4. CICCUS Data Element/Data Source Cross-Reference (Part 2 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICCUS (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUCONTM Terminal Connect Time	U	U	U	
CSUCOST Processing Charges	U	U	U	
CSUCPRTM Task CPU Real Time	008(USRCPUT)	TACPU	TACPURTM	
CSUCPUNI Instructions Executed	U	U	U	
CSUCPUTM Task CPU Time (elapsed)	008(USRCPUT)	TACPU	TACPURTM	
	175(RLSCPUT)			
CSUCRSTM Conversational Response Time	U	U	U	
CSUCTRN Conv. Transaction Count	U	U	U	
CSUCURTA Active User Trans When User Task Attach	434(CURTASKS)	N	N	
CSUDAMSC Damaged Storage Chain	064(TASKFLAG)	TAFLAG1	TAFLAG1	
CSUDBCLO Number of SQL Close Cursors	MBCLON(OMEG)	N	N	
CSUDBDEL Number of SQL Deletes	MBDELN(OMEG)	N	N	
CSUDBEND Schedule End Time (DBCTL)	STATDATN	N	N	
CSUDBDES Number of SQL Describes	MBDESN(OMEG)	N	N	
CSUDBEXE Number of SQL Executes	MBEXEN(OMEG)	N	N	
CSUDBEXI Number of SQL Execute Immediate	MBEXIN(OMEG)	N	N	
CSUDBFET Number of SQL Fetches	MBFETN(OMEG)	N	N	
CSUDBINS Number of SQL Inserts	MBINSN(OMEG)	N	N	
CSUDBISR Database ISRT Calls (DBCTL)	STATISRT	N	N	
CSUDBMIS Number of SQL Miscellaneous Calls	MBMISN(OMEG)	N	N	
CSUDBNSC DB2 Non-SQL Count	N	TANSQCT	TANSQCCT	
CSUDBNST DB2 Non-SQL Call Time	N	TANSQTIM	TANSQCTM	
CSUDBOPN Number of SQL Open Cursors	MBOPNN(OMEG)	N	N	
CSUDBPRE Number of SQL Prepares	MBPREN(OMEG)	N	N	
CSUDBSEL Number of SQL Selects	MBSELN(OMEG)	N	N	
CSUDBSTR Schedule Start Time (DBCTL)	STATDATS	N	N	
CSUDBUPD Number of SQL Updates	MBUPDN(OMEG)	N	N	
CSUDBWAT DB2 Wait Time	N	TANDB2TM	TADB2WTM	
CSUDBWCT DB2 Wait Count	N	TANDB2CT	TADB2WCT	
CSUDB2RC Total No. DB2 Requests	180(DB2REQCT)	N	N	
CSUDB2WT DB2 Wait Time	189(DB2WAIT)	N	N	
CSUDELTM Elapsed Time in SQL Delete Calls	MBDELT(OMEG)	N	N	
CSUDELTW User Task Wait Time -Extrapartition TDL	404(TDELWTT)	N	N	
CSUDESTM Elapsed Time in SQL Describe Calls	MBDEST(OMEG)	N	N	
CSUDGHPT Elapsed Time for DL/I GHNP Calls	MIGPNT(OMEG)	N	N	
CSUDHCRC No. Document Create Requests	226(DHCRECT)	N	N	
CSUDHINC No. Document Insert Requests	227(DHINSCT)	N	N	
CSUDHRTC No. Document Retrieve Requests	229(DHRETCT)	N	N	
CSUDHSTC No. Document Set Requests	228(DHSETCT)	N	N	

FIGURE 6-4. CICCUS Data Element/Data Source Cross-Reference (Part 3 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICCUS (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUDHTC Total No. Document Requests	230(DHTOTCT)	N	N	
CSUDHTDL Total Document Created Length	240(DHTOTDCL)	N	N	
CSUDILWT User Task Wait Time -Intrapartition TDL	403(TDILWTT)	N	N	
CSUDLDEL DL/I DLET Calls	MIDELN(OMEG)	N	N	
CSUDLDLT Elapsed Time for DL/I DLET Calls	MIDELT(OMEG)	N	N	
CSUDLGHP DL/I GHNP Calls	MIGPNN(OMEG)	N	N	
CSUDLGHU DL/I GHU Calls	MIGHUN(OMEG)	N	N	
CSUDLGN DL/I GN Calls	MIGNN(OMEG)	N	N	
CSUDLGNP DL/I GNP Calls	MIGNPN(OMEG)	N	N	
CSUDLGNT Elapsed Time for DL/I GN Calls	MIGNT(OMEG)	N	N	
CSUDLGPT Elapsed Time for DL/I GNP Calls	MIGNPT(OMEG)	N	N	
CSUDLGU DL/I GU Calls	MIGUN(OMEG)	N	N	
CSUDLGUT Elapsed Time for DL/I GU Calls	MIGUT(OMEG)	N	N	
CSUDLHUT Elapsed Time for DL/I GHU Calls	MIGHUT(OMEG)	N	N	
CSUDLICC DL/I Call Count	N	TADLICT	TADLIRCT	
CSUDLIPW DL/I Pool Wait	N	TAFLAG6	TAFLAG6	
CSUDLISR DL/I ISRT Calls	MIINSN(OMEG)	N	N	
CSUDLIST Elapsed Time for DL/I ISRT Calls	MIINST(OMEG)	N	N	
CSUDLITW DL/I Thread Wait	N	TAFLAG6	TAFLAG6	
CSUDLIWT DL/I Call Elapsed Time	N	TADLITIM	TADLIRTM	
CSUDLREP DL/I REPL Calls	MIREPN(OMEG)	N	N	
CSUDLRPT Elapsed Time for DL/I REPL Calls	MIREPT(OMEG)	N	N	
CSUDLSCH DL/I Schedules	MISCHN(OMEG)	N	N	
CSUDLSCT Elapsed Time for DL/I Schedules	MISCHT(OMEG)	N	N	
CSUDLTRN DL/I Terminates	MITERN(OMEG)	N	N	
CSUDLTRT Elapsed Time for DL/I Terminates	MITERN(OMEG)	N	N	
CSUDLWAT DL/I Wait Time	N	TADLIWTM	TADLIWTM	
CSUDLWCT DL/I Wait Count	N	TADLIWCT	TADLIWCT	
CSUDSIMX First Dispatch Delay due to MXT	127(MXTDELAY)	N	N	
CSUDSPCT Transaction Dispatch Count	N	TADSPCNT	TADSPDSC	
			TADSPWRC(2.1)	
CSUDSTMX Maximum Number of Open TCBs (key 8/9)	252(DSTCMHWM)	N	N	
CSUDSWAT Wait for Dispatch (Queue Time)	102(DISPWTT)	TAWQUEUE	TADSPQST	
			TADSPWRT(2.1)	
CSUDS1TC First Dispatch Delay due to TCLASS	126(TCLDELAY)	N	N	
CSUDS1WT Wait for First Dispatch Time	125(DSPDELAY)	TAW1DSP	TAW1DSP	
CSUDTSCT DTP (MRO) Suspend Wait Count	N	TADTPSCT	N	
CSUDTSUS DTP (MRO) Suspend Wait Time	N	TADTPSTM	N	
CSUDTWAT MRO/DTP Wait Time	N	TADTPWTM	N	
CSUDTWCT MRO/DTP Wait Count	N	TADTPWCT	N	
CSUDWWTM Dispatchable-Wait Wait Time	184(GVUPWAIT)	N	N	
CSUECBWC Unidentified ECB Wait Count	N	TAECBCNT	N	

FIGURE 6-4. CICCUS Data Element/Data Source Cross-Reference (Part 4 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICCUSU (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUECBWT Unidentified ECB Wait Time	N	TAECBTIM	N	
CSUEJBAC Number of Bean Activations	312(EJBSACCT)	N	N	
CSUEJBACR Number of Bean Creation Calls	314(EJBCRECT)	N	N	
CSUEJBMT Number of Bean Method Calls	316(EJBMTHCT)	N	N	
CSUEJBPA Number of Bean Passivations	313(EJBSFACT)	N	N	
CSUEJBRE Number of Bean Removal Calls	315(EJBREMTCT)	N	N	
CSUEJBTO Number of Total Bean Calls	317(EJBTOTCT)	N	N	
CSUEQSCT Enqueue Suspend Wait Count	129(ENQDELAY)	TAENQWCT	N	
CSUEQSUS Enqueue Suspend Wait Time	129(ENQDELAY)	TAENQWTM	N	
CSUERSTM Excessive Transaction Response Time	U	U	U	
CSUETRNR Excessive Transactions Processed	U	U	U	
CSUETTRN Terminal Originated Excessive Trans.	U	N	N	
CSUEXCPS I/O (EXCPs) Generated	U	U	U	
CSUEXETM Elapsed Time in SQL Execute Calls	MBEXET(OMEG)	N	N	
CSUEXITM Elapsed Time in SQL Execute Immediate	MBEXIT(OMEG)	N	N	
CSUEXWAT Exception Condition Wait Time	103(EXWTTIME)	N	N	
CSUEXWTM EXEC CICS External Wait Time	181(WTEXWAIT)	N	N	
CSUFACTY User Defined Facility Type	004(T)	TAFACCTYP	TAFACCTYP	
		TAFLAG2	TAFLAG2	
CSUFCADD File Control Adds	039(FCADDCT)	N	N	
CSUFCBRW File Control Browses (Getnext+Getprev)	038(FCBRWCT)	N	N	
CSUFCDLT File Control Deletes	040(FCDELCT)	N	N	
CSUFCGET File Control Gets (Reference+Update)	036(FCGETCT)	N	N	
CSUFCOPS Number of CICS File Control Calls	036(FCGETCT)	TAFILCT	TAFICPRCT	
	037(FCPUTCT)			
	038(FCBRWCT)			
	039(FCADDCT)			
	040(FCDELCT)			
CSUFCPUT File Control Puts	037(FCPUTCT)	N	N	
CSUFCTIM FCP File Request Time	N	TAFILTIM	TAFICPRTM	
CSUFCTOT Total File Requests	093(FCTOTCT)	N	N	
CSUFCVWT User Task Wait Time -VSAM String	427(FCVSWTT)	N	N	
CSUFCWAT File Control Wait Time	063(FCIOWTT)	TAFILTIM	TAFICPWTM	
		TAFILWTM		
CSUFCWCT FCP Wait Count	N	TAFILWCT		
CSUFCXWT User Task Wait Time -Excl CI Control	426(FCXCWTT)	N	N	
CSUFETTM Elapsed Time in SQL Fetch Calls	MBFETT(OMEG)	N	N	
CSUFMBES Bytes freemained Above 16MB ESDSA	149(SC31FSHR)	N	N	
CSUFMBSD Bytes freemained Below 16MB ESDSA	146(SC24FSHR)	N	N	
CSUGMBES Bytes getmained Above 16MB ESDSA	148(SC31GSHR)	N	N	
CSUGMBSD Bytes getmained Below 16MB ESDSA	145(SC24GSHR)	N	N	
CSUGMES Getmains Above 16MB ESDSA	147(SC31SGCT)	N	N	
CSUGMSCT Storage Suspend Wait Count	N	TAGMAWCT	TAGMAWCT	

FIGURE 6-4. CICCUSU Data Element/Data Source Cross-Reference (Part 5 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICSU (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUGMSD Getmains Below 16MB SDSA	144(SC24SGCT)	N	N	
CSUGQDTM Global ENQ Delay Time	123(GNQDELAY)	N	N	
CSUHIHSTG Hi-water Terminal And User Memory	033(SCUSRHWM)	TAHSTG	TAU31HWM	TAU24HWM
CSUICDTM Interval Control Delay Time	183(ICDELAY)	N	N	
CSUICOPS Number of CICS Interval Control Calls	059(ICPUINCT)	N	N	
CSUICSCT ICP Suspend Wait Count	N	TAICPSCT	N	
CSUICSUS ICP Suspend Wait Time	N	TAICPSTM	N	
CSUICTCT Total Interval Control Requests	066(ICTOTCT)	N	N	
CSUIDMT0 Total IDMS Calls	MIDMSN(OMEG)	N	N	
CSUIDMWT Total IDMS Wait Time	MIDMSS(OMEG)	N	N	
CSUIMSGS Input Messages Issued	034(TCMGSGIN1)	TATERMIN	TATERICT	
	067(TCMGSGIN2)			
CSUIMSRC Total No. IMS Requests	179(IMSREQCT)	N	N	
CSUIMSWT IMS Wait Time	186(IMSWAIT)	N	N	
CSUINCH Input Message Character Traffic	083(TCCHRIN1)	TATERMIL	TATERILG	
	085(TCCHRIN2)			
CSUINSTM Elapsed Time in SQL Insert Calls	MBINST(OMEG)	N	N	
CSUINVPA Invalid CMF Call Encountered	064(TASKFLAG)	N	N	
CSUIPHER Cipher Suite Code Selected	320(SOCIPHER)	N	N	
CSUIRCWT MRO/IRC Wait Time	N	TAMMROIR	TAAWTTMS	
CSUIRSHP MRO Ship Count	N	TACMROIS	N	
CSUIRWAT MRO/ISC Wait Time	100(IRIOWTT)	TAMMROIR	N	
		TAMMROIS	N	
CSUIRWCT MRO/IRC Event Wait Count	N	TACMROIR	N	
CSUIRCWT MRO/ISC Wait Time	N	TAMMROIS	N	
CSUISSCT ISC (MRO) Suspend Wait Count	N	TAISPSTCT	N	
CSUISSUS ISC (MRO) Suspend Wait Time	N	TAISPSTM	N	
CSUJCOPS Number of CICS Journal Control Calls	058(JNLWRTCT)	TAJCTCT	TAJCPRCT	
CSUJCTIM Journal Request Time	N	TAJCTIM	TAJCPRTM	
CSUJCWAT Journal Control Wait Time	010(JCIOWTT)	TAJCTIM	TAJCPWTM	
		TAJCWTM		
CSUJCWCT Journal Wait Count	N	TAJCWCT	TAJCPWCT	
CSUJOB CICS JOB Name	N	MMDJOB	LMRKJOBN	
CSUJVMST Total JVM Suspend Time	254(JVMSUSP)	N	N	
CSUJVMTM Total JVM Elapse Time	253(JVMTIME)	N	N	
CSUJ8CTM User Task J8 Mode CPU Time	260(J8CPUT)	N	N	
CSUJ9CTM User Task J9 Mode Cpu Time	267(J9CPUT)	N	N	
CSUK8CTM User Task Key 8 Mode Cpu Time	263(KY8CPUT)	N	N	
CSUK9CTM User Task Key 9 Mode Cpu Time	265(KY9CPUT)	N	N	
CSUK9DTM User Task K9 TCB Dispatch Time	264(KY9DISPT)	N	N	
CSULATNM Platform Name in Appl Context Data	452(ACPLATNM)	N	N	
CSULDSTX Count Long Responses Within Limit	U	U	U	
CSULMDTM Lock Manager Delay Time	128(LMDELAY)	N	N	
CSULOGCT CICS Logger Write Count	172(LOGWRTCT)	N	N	

FIGURE 6-4. CICSU Data Element/Data Source Cross-Reference (Part 6 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICSU (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSULRSTM Long Response Time	U	U	U	
CSULTRN Long Transaction Count	U	U	U	
CSULWTT User Task Wait Time -AllC Req IPIC Sess	319(ISALWTT)	N	N	
CSUL8CTM User Task L8 Mode CPU Time	259(L8CPUT)	N	N	
CSUL9CTM User Task L9 Mode CPU Time	266(L9CPUT)	N	N	
CSUMAXTA MAXTASKS Value When User Task Attached	433(MAXTASKS)	N	N	
CSUMCHTP CEC Machine Type Where CICS is Running	430(CECMCHTP)	N	N	
CSUMDLID CEC Model Number Where CICS is Running	431(CECMDLID)	N	N	
CSUMDSTX Count Medium Responses Within Limit	U	U	U	
CSUMEMHI Hi Water Program Memory Size	087(PCSTGHWM)	N	N	
CSUMEMRC Memory Usage Running Count	033(SCUSRHWM)	TAHSTG	TAU31HWM	
	087(PCSTGHWM)		TAU24HWM	
CSUMINP Input Messages - Primary Facility	034(TCM5GIN1)	N	N	
CSUMINS Input Messages - Secondary Facility	067(TCM5GIN2)	N	N	
CSUMIN62 Input Messages - Alt. Facility LU 6.2	135(TCM62IN2)	N	N	
CSUMISCT Mirror Suspend Wait Count	N	TACMISCT	N	
CSUMISTM Elapsed Time in SQL Miscell. Calls	MBMIST(OMEG)	N	N	
CSUMISUS Mirror Suspend Wait Time	N	TACMISTM	N	
CSUMIVER Micro Ver of Appl in Appl Context Data	455(ACMICVER)	N	N	
CSUMJVER Major Ver of Appl in Appl Context Data	453(ACMAJVER)	N	N	
CSUMMSTM User Task Unavailable TCB wait Time	279(DSMMSCT)	N	N	
CSUMNVER Minor Ver of Appl in Appl Context Data	454(ACMINVER)	N	N	
CSUMOUTP Output Messages - Primary Facility	035(TCM5GOU1)	N	N	
CSUMOUTS Output Messages - Secondary Facility	068(TCM5GOU2)	N	N	
CSUMOU62 Output Messages - Alt. Facility LU6.2	136(TCM62OU2)	N	N	
CSUMROCT MRO Request Count	N	TAISPRCT	TAISPRCT	
CSUMROTM MRO Request Time	N	TAISPRTM	TAISPRTM	
CSUMRSTM Medium Response Time	U	U	U	
CSUMSCTM User Task Other Mode CPU Time	258(MSCPUT)	N	N	
CSUMSDTM User Task Other Mode Dispatch Time	257(MSDISPT)	N	N	
CSUMTRN Medium Transaction Count	U	U	U	
CSUMXCTM Max Conversational Response Time	U	U	U	
CSUMXETM Max Excessive Response Time	U	U	U	
CSUMXLTM Max Long Response Time	U	U	U	
CSUMXMEM Max Transaction Memory Used	033(SCUSRHWM)	TAHISTG	TAU31HWM	
	087(PCSTGHWM)		TAU24HWM	
CSUMXMTM Max Medium Response Time	U	U	U	
CSUMXOPS Maximum Number of CICS Calls				
	CSUTLOPS			
CSUMXSTM Max Short Response Time	U	U	U	
CSUMXTRT Max Transaction CPU Time	U	U	U	
CSUMXTSK Max Tasking Condition	U	U	U	
CSUMXTTM Max Response Time	U	U	U	
CSUMXWAT Maximum CICS Wait Time	009(TCIOWTT)	TAWAIT	TATSKWTM	
	010(JCIOWTT)			
	011(TSIOWTT)			
	014(SUSPTIME)			
	063(FCIOWTT)			

FIGURE 6-4. CICSU Data Element/Data Source Cross-Reference (Part 7 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICSU (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUNTR0P Number Of Counters Generated	U	U	U	
	068(TCM5G0U2)			
CSUNZTM Normalized zIIP/zAAP CPU TIME	N	N	N	
CSUOCPTC User Task Spec Proc Eligible CPU Count	437(OFFLCPUT)	N	TASTCPOC	
CSUOMSGS Output Messages Issued	035(TCM5G0U1)	TATERMOT	TATEROCT	
CSUOPNTM Elapsed Time in SQL Open Cursors	MBOPNT(OMEG)	N	N	
	068(TCM5G0U2)			
CSUOPRNM Operation Name in Appl Context Data	456(ACOPERNM)	N	N	
CSUOTDLT Max Open TCB Delay Time	250(MXTOTDLY)	N	N	
CSUOUTCH Output Message Character Traffic	084(TCCHROU1)	TATERMOL	TATEROLG	
	086(TCCHROU2)			
CSUPAGIN Page Ins	061	TAPAGIN	TAPGEICT	
CSUPAGOT Page Outs	N	TAPAGOT	TAPGEICT	
CSUPCDPL No. DPL Program Links	073(PCDPLCT)	N	N	
CSUPCHCD Program Storage High-water Mark CDSA	143(PC24CHWM)	N	N	
CSUPCHEC Program Storage High-water Mark ECDSA	142(PC31CHWM)	N	N	
CSUPCHER Program Storage High-water Mark ERDSA	122(PC31RHWM)	N	N	
CSUPCHES Program Storage High-water Mark ESDSA	161(PC31SHWM)	N	N	
CSUPCHRD Program Storage High-water Mark RDSA	162(PC24RHWM)	N	N	
CSUPCHSD Program Storage High-water Mark SDSA	160(PC24SHWM)	N	N	
CSUPCLCT Total Program Link URM Count	072(PCLURMCT)	N	N	
CSUPCLNK Program Control Links	055(PCLINKCT)	N	N	
CSUPCL0D Program Control Loads	057(PCL0ADCT)	N	N	
CSUPCMA Program Storage High-water Above 16MB	139(PC31AHWM)	N	N	
CSUPCOPS Program Control Calls	X	TAPCCT	TAPGMRCT	
CSUPCTIM Program Fetch Time	N	TAPCTIM	TAPGMRCT	
CSUPCTRX Cumulative Pct Resp Within Limit	U	U	U	
CSUPCWAT Program Fetch Wait Time	115(PCL0ADTM)	TAPCWTM	TAPGMWTM	
CSUPCWCT Program Fetch Wait Count	N	TAPCWCT	TAPGMWCT	
CSUPCXCL Program Control XCTLs	056(PCXCTLCT)	N	N	
CSUPETTM Pseudo Elapsed Time	007(USRDISPT)	TADTIME	TATSKDTM	
CSUPFWAT Primary Facility Wait Time	N	TAOPRWTM	TAOPRWTM	
CSUPMWAT Preempt Wait Time	N	TAPMTWTM	N	
CSUPMWCT Preempt Wait Count	N	TAPMTWCT	N	
CSUPNAME Process Name	200(PRC5NAME)	N	N	
CSUPOLST # of Policy Rule Threshold Exceeded	449(MPPRTXCD)	N	N	
CSUPRCID Process ID	202(PRC5ID)	N	N	
CSUPRETM Elapsed Time in SQL Prepare Calls	MBPRET(OMEG)	N	N	
CSUPTYPE Process Type	201(PRC5TYPE)	N	N	
CSUQRCTM User Task QR Mode CPU Time	256(QRCPUT)	N	N	
CSUQRDTM User Task QR Mode Dispatch Time	255(QRDISPT)	N	N	
CSUQRDLY QR Mode Delay Time	249(QRMODDLY)	N	N	
CSURDQWT DB2 ReadyQ Wait Time	187(DB2RDYQW)	N	N	
CSURESTM Transaction Residency Time	005(START)	TADATE	TASTRDTS	
	006(STOP)	TASTIME	TAENDDTS	
		TAPTIME		

FIGURE 6-4. CICSU Data Element/Data Source Cross-Reference (Part 8 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICSU (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSURLSWT Wait Time for RLS FC I/O	174(RLSWAIT)	N	N	
CSURLSTM SRB CPU Time for RLS	175(RLSCPUT)	N	N	
CSURMICT Resource Manager Interface Count	170(RMITIME)	N	N	
CSURMITM Time in Resource Manager Interface	170(RMITIME)	N	N	
CSURMSCT RMI Suspend Count	171(RMISUSP)	N	N	
CSURMSUS Wait for Resource Manager Interface	171(RMISUSP)	N	N	
CSUROCTM User-task R0 TCB CPU time	270(ROCPUT)	N	N	
CSUROMTM User Task Wait Time -Redispatch on R0 TCB	348(ROMODDLY)	N	N	
CSURRSWT RRMS/MVS Wait Time	191(RRMSWAIT)	N	N	
CSURSYS Remote System ID	130(RSYSID)	N	N	
CSURTRWT Run Transaction Wait Time	195(RUNTRWTT)	N	N	
CSURTRWTK Recovery Token	132(RMUOWID)	N	N	
CSUSCGMA GETMAINS Above 16MB ECDSA	120(SCCGETCT)	N	N	
CSUSCGMB GETMAINS Below 16MB CDSA	117(SCCGETCT)	N	N	
CSUSCHCD Task Storage High-water Mark CDSA	116(SC24CHWM)	N	N	
CSUSCHEC Task Storage High-water Mark ECDSA	119(SC31CHWM)	N	N	
CSUSCOCA Storage Occupancy Above 16MB ECDSA	121(SC31COCC)	N	N	
CSUSCOCB Storage Occupancy Below 16MB CDSA	118(SC24COCC)	N	N	
CSUSCOPS Number of CICS Storage Control Calls	054(SCUGETCT)	TAGETMS	TAGETMCT	
CSUSCPTM User Task Standard Processor CPU Time	436(CPUTONCP)	N	TASTCPUT	
CSUSCSUS Storage Suspend Wait Time	N	TAGMATIM	TASTGWMT	
CSUSCUCA Storage Occupancy Above 16 Meg	107(SCUCRSTG)	N	N	
CSUSDSTX Count Short Responses Within Limit	U	U	U	
CSUSELTM Elapsed Time in SQL Select Calls	MBSELT(OMEG)	N	N	
CSUSERVU Service Units	U	U	U	
CSUSIOCT Access Method Calls (SIO Count)	070(FCAMCT)	TAFILCT	TAFICPRCT	
CSUSOBDE No. Bytes Decrypted	243(SOBYDECT)	N	N	
CSUSOBEN No. Bytes Encrypted	242(SOBYENCT)	N	N	
CSUSOMTM User Task Wait Time -Redispatch on S0 TCB	349(SOMODDLY)	N	N	
CSUSOS Short On Storage Condition	064(TASKFLAG)	TAFLAG1	TAFLAG1	
CSUSOWTM Socket I/O Wait Time	241(SOIOWTT)	N	N	
CSUSPCOM Program Compression	N	TAFLAG3	TAFLAG3	
CSUSPETM Synpoint Elapsed Time	173(SYNCTIME)	N	N	
CSUSPNDU Unidentified Suspend Wait Count	N	TASPNDC	N	
CSUSPOPS Number of CICS Synch Point Calls	060(SPSYNCT)	N	N	
CSUSQLTM Total Elapsed Time in SQL Calls	N	TASQLTIM	TASQLCTM	

FIGURE 6-4. CICSU Data Element/Data Source Cross-Reference (Part 9 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICCSTU (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUSQLTO Total SQL Calls	N	TASQLCT	TASQLCCT	
CSUSRBTM SRB CPU Time	U	U	U	
CSUSRMTR SRM Ended Transactions	U	U	U	
CSUSRSTM Short Response Time	U	U	U	
CSUSRU System Resource Units	U	U	U	
CSUTAWTT User Task Wait Time -MRO Alloc Request	343(TCALWTT)	N	N	
CSUSTGMA Getmains Above XA Line	105(SCUGETCT)	N	N	
CSUSTGMB Getmains Below XA Line	N	N	N	
CSUSTHWA Task Storage High-water Mark Above XA	106(SCUSRHWM)	N	N	
CSUSTHMB Task Storage High-water Mark Below XA	108(PC24BHWM)	N	N	
CSUSTOTM Storage Occupancy Time	095(SCUSRSTG)	N	N	
CSUSTRN Short Transaction Count	U	U	U	
CSUSUSDT Unidentified Suspend Wait Time	N	TASPNDTM	N	
CSUSUSTM Time Transaction Was on CICS Susp Chn	014(SUSPTIME)	TADTPSTM	TASTGWTM	
		TAICPSTM	TATERWTM	
		TAISPSTM		
		TAENQWTM		
		TACMISTM		
		TAGMATIM		
		TASPNDTM		
		TATCWITM		

FIGURE 6-4. CICCSTU Data Element/Data Source Cross-Reference (Part 10 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICCUS (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUSYDTM Syncpoint Delay Time	196(SYNCPLY)	N	N	
CSUSYWTT Server Syncpoint Wait Time	177(SRVSYWTT)	N	N	
CSUSZACT FEPI Time Outs Waiting to Alloc Conv.	157(SZALLCTO)	N	N	
CSUSZALC FEPI Conversations Allocated Count	150(SZALLOCT)	N	N	
CSUSZCIN Characters Received Through FEPI Count	155(SZCHRIN)	N	N	
CSUSZCOT Characters Sent Through FEPI Count	154(SZCHROUT)	N	N	
CSUSZRCC FEPI RECEIVE Requests Count	151(SZRVCVT)	N	N	
CSUSZRCT FEPI Time Outs Waiting to Receive Data	158(SZRVCVO)	N	N	
CSUSZSEC FEPI SEND Requests Count	152(SZSENDCT)	N	N	
CSUSZSTC FEPI START Requests Count	153(SZSTRCT)	N	N	
CSUSZTCT Total FEPI API and SPI Requests Count	159(SZTOTCT)	N	N	
CSUSZWIM FEPI I/O Wait Time	156(SZWAIT)	N	N	
CSUS8CTM User Task S8 Mode CPU Time	261(S8CPUT)	N	N	
CSUTCALC Terminal Control Allocates Requested	069(TCALLOCT)	N	N	
CSUTCABC No. CICS Dispatcher TCB Attachs	251(TCBATTCT)	N	N	
CSUTCBTM TCB CPU Time	008(USRCPUT)	TACPU	TACPURT	
CSUTCLAS Transaction Class at Task Creation	110	N	N	
CSUTCOMTM User Task TCB Mismatch Wait Time	268(DSTCBMWT)	N	N	
CSUTCPTC User Task Standard Processor CPU Count	436(CPUTONCP)	N	TASTCPUC	
CSUTCSES Terminal Session Connection Name	169(TERMCMNM)	N	N	
CSUTCSTG TIOA Storage Allocated to Terminal	104	N	N	
CSUTCWAT Terminal Control Wait Time	009(TCIOWTT)	TATCWITM	TATERWTM	
CSUTCWCT Terminal Suspend Wait Count	N	TATCWCT	TATERWCT	
CSUTCW61 TC I/O Wait Time LU 6.1	133(LU61WTT)	N	N	
CSUTCW62 TC I/O Wait Time LU 6.2	134(LU62WTT)	N	N	
CSUTDECT Transient Data (Extra) Request Count	N	TATDCNT	N	
CSUTDETM Transient Data (Extra) Request Time	N	TATDIME	N	
CSUTDEWC Transient Data (Extra) Wait Count	N	TATDEWCT	N	
CSUTDEWT Transient Data (Extra) Wait Time	N	TATDEWTM	N	
CSUTDGET Transient Data Gets	041(TDGETCT)	N	N	
CSUTDICT Transient Data (Intra) Request Count	N	TATDICNT	N	
CSUTDITM Transient Data (Intra) Request Time	N	TATDITIM	N	
CSUTDIWC Transient Data (Intra) Wait Count	N	TATDIWCT	N	
CSUTDIWT Transient Data (Intra) Wait Time	N	TATDIWTM	N	
CSUTDOPS Number of CICS Transient Data Calls	041(TDGETCT)	TATDCNT	TATDQRCT	
	042(TDPUTCT)	TATDICNT		
	043(TDPURCT)			
CSUTDPRG Transient Data Purges	043(TDPURCT)	N	N	
CSUTDPUT Transient Data Puts	042(TDPUTCT)	N	N	
CSUTDSTX Count Responses Within Limit	U	U	U	
CSUTDTIM Transient Data Time	N	TATDIME	TATDQRTM	
		TATDITIM		
CSUTFACL Transaction Facility Name	163(FACTYNAME)	N	N	

FIGURE 6-4. CICCUS Data Element/Data Source Cross-Reference (Part 11 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICSU (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUTLOPS Number of CICS Calls (Total)				
CSUBMOPS				
CSUFCOPS				
CSUICOPS				
CSUJCOPS				
CSUPCOPS				
CSUSCOPS				
CSUSPOPS				
CSUTDOPS				
CSUTSOPS				
CSUTLUN Terminal VTAM LUNAME	111(LUNAME)	TANETNAM	TANETLNM	
CSUTLWAT Total Wait Time				
CSUFCWAT				
CSUJCWAT				
CSUSUSTM				
CSUTCWAT				
CSUTDWAT				
CSUTSWAT				
CSUWAITM				
CSUTMERR Monitor Timing Error Encountered	064(TASKFLAG)	N	N	
CSUTRANS Transactions Processed	X	X	X	
CSUTRCLS Transaction Class Name	166(TCLSNAME)	N	N	
CSUTRNPR Transaction Priority at Task Creation	109(TRANPRI)	N	N	
CSUTRSTM Transaction Response Time	005(START)	TADATE	TATSKDTM	
	006(STOP)	TASTIME	TATSKWTM	
		TAPTIME		
CSUTSATM Temp Stor Aux (Output) Request Time	N	TATSATIM	N	
CSUTSAWC Temp Stor Aux (Output) Wait Count	N	TATSAWCT	N	
CSUTSAWT Temp Stor Aux (Output) Wait Time	N	TATSAWTM	N	
CSUTSICT Temp Stor (Aux+Main) Input Request Cnt	N	TATSBICT	N	
CSUTSITM Temp Stor (Aux+Main) Input Request Time	N	TATSBITM	N	
CSUTSIWC Temp Stor (Aux+Main) Input Wait Count	N	TATSBWCT	N	
CSUTSIWT Temp Stor (Aux+Main) Input Wait Time	N	TATSBWTM	N	
CSUTSKID Task Identification Number	031(TRANNUM)	TAKCTTA	TATASKID	
CSUTSGET Temporary Storage Gets	044(TSGETCT)	N	N	
CSUTSMCT Temp Stor Main (Output) Byte-Count	N	TATSMBCT	N	
CSUTSOPS Number of CICS Temporary Storage Calls	044(TSGETCT)	TATSCT	TATSORCT	
	046(TSPUTACT)	TATSBICT		
	047(TSPUTMCT)	TATSACNT		
		TATSMCNT		
CSUTSPTA Temporary Storage Puts to AUX	046(TSPUTACT)	TATSACNT	N	
CSUTSPTM Temporary Storage Puts to Main	047(TSPUTMCT)	TATSMCNT	N	
CSUTSVMT VSAM Transient I/O Wait Time	101(TDIOWTT)	N	N	

FIGURE 6-4. CICSU Data Element/Data Source Cross-Reference (Part 12 of 13)

CICS ANALYZER DATA ELEMENT/DATA SOURCE CROSS-REFERENCE				
THE CICSU (CICS USER ACTIVITY) FILE	CMF	MONITOR	TCE 2.0+	
CSUTSWAT Temporary Storage Wait Time	011(TSIOWTT)	TATSTIM TATSAWTM TATSBWTM	TATSQWTM	
CSUTSWTM Shared TS I/O Wait Time Time	178(TSSHWAIT)	N	N	
CSUTTRAN Terminal Originated Transactions Prcd	X	N	N	
CSUUDBC User Database Request Count	N	TAUDBCNT	TAUDBRCT	
CSUUDBTM User Database Request Time	N	TAUDBTIM	TAUDBRTM	
CSUUDWAT User Database Wait Time	N	TAUDBWTM	N	
CSUUDWCT User Database Wait Count	N	TAUDBWCT	N	
CSUUMPTC Umbrella Transaction Pseudo-Transid	MUMBPTC(OMEG)	N	N	
CSUUMUSR Umbrella Transaction User Field	MUMBUSR(OMEG)	N	N	
CSUUMWRK Umbrella Transaction User Work Area	MUSRWRK(OMEG)	N	N	
CSUUPDTM Elapsed Time in SQL Update Calls	MBUPDT(OMEG)	N	N	
CSUURID RRMS/MVS Unit of Recovery ID	190(RRMSURID)	N	N	
CSUVSMBW VSAM Buffer Wait	N	TAFLAG2	TAFLAG2	
CSUVSMSW VSAM String Wait	N	TAFLAG2	TAFLAG2	
CSUWAITM Transaction Wait Time	009(TCIOWTT) 010(JCIOWTT) 011(TSIOWTT) 014(SUSPTIME) 063(FCIOWTT)	TAWAIT	TATSKWTM	
CSUWBCIN No. Characters Received via WEB Req	232(WBCHRIN)	N	N	
CSUWBCOT No. Characters Sent via WEB Req	234(WBCHROUT)	N	N	
CSUWBRCT No. WEB Receive Requests	231(WBRVCVT)	N	N	
CSUWBRPR No. Repository Reads	236(WBREPRCT)	N	N	
CSUWBRPW No. Repository Writes	237(WBREPWCT)	N	N	
CSUWBSCT No. WEB Send Requests	233(WBSENDCT)	N	N	
CSUWBTC Total No. WEB Requests	235(WBTOTWCT)	N	N	
CSUWEWTM EXEC CICS Wait Event Wait Time	182(WTCEWAIT)	N	N	
CSUWFCTM User Task TCP/IP Wait Time	300(TSIOWTT)	N	N	
CSUWLMRC MVS Workload Manager Report Class	168(RPTCLASS)	N	N	
CSUWLMSC MVS Workload Manager Service Class	167(SRVCLASS)	N	N	
CSUWTFIL Waiting Resource Name	N	MDEXFIL	N	
CSUWTTUP Waiting Resource Name	N	MDEXTYP	N	
CSUXCSTM Task CPU TIME withiyt zIIP/zAAP Eligible	N	N	N	
CSUXTSTM TCB CPU TIME without zIIP/zAAP Eligible	N	N	N	
CSUX8CTM User Task X8 Mode CPU Time	271(X9CPUT)	N	N	
CSUX9CTM User Task X9 Mode CPU Time	272(X9CPUT)	N	N	
CSUZZCTM zIIP/zAAP Eligible CPU TIME on CP	437(OFFLCPUT)	N	TASTCPOT	
CSU64CGC User storage +2G GETMAIN Reqs in GCDSA	441(SC64CGCT)	N	N	
CSU64CMX Max Allocated +2G User Storage -GCDSA	442(SC64CHWM)	N	N	
CSU64FSH FREEMAINed 4K Pages of +2G Shared Stor	447(SC64FSHR)	N	N	
CSU64GSH GETMAINed 4K Pages of +2G Shared Stor	446(SC64GSHR)	N	N	
CSU64HWM USER & CICS 64 Pool High-water Mark	N	N	TAC64HWM	
CSU64SGC Shared Storage +2G GETMAIN Requests	445(SC64SGCT)	N	N	
CSU64UGC User storage +2G GETMAIN Reqs in GUDSA	443(SC64UGCT)	N	N	
CSU64UMX Max Allocated +2G User Storage -GUDSA	444(SC64UHWM)	N	N	

FIGURE 6-4. CICSU Data Element/Data Source Cross-Reference (Part 13 of 13)

6.4 CA MICS and CA SMF Director Interface

CA SMF Director is an SMF management product with special features that can be used to optimize CA MICS processing. CA SMF Director can significantly reduce operational overhead by creating content specific files while simultaneously creating an SMF historical archive during the SMF dump process.

CA MICS DAILY and incremental update operational jobs can be modified to take advantage of this product, provided your site is licensed for and has CA SMF Director installed in your environment.

CA SMF Director provides functions that extract SMF data at dump time or from previously archived SMF files. These functions eliminate the need for any external utilities used for preprocessing of the SMF dump tape for data extraction prior to execution of the DAILY job. In addition, the DAYSMF step of the CA MICS DAILY job may no longer be required. For details on how to eliminate the DAYSMF step, see section 5.10, Removing the DAYSMF Step from the DAILY Job of the PIOM guide.

CA SMF Director SPLIT and EXTRACT statement operands provide a way for you to create content specific files to meet your requirements. The main functions of the operands include the ability to:

- o Select system identifiers
- o Select or exclude SMF record types and subtypes
- o Use conditional statements for granularity
- o Define time boundaries for SMF data

To use the SPLIT function of CA SMF Director for CA MICS components, one or more SPLIT statements must be coded. The split is performed at SMF dump time and the files can then be used as input to products in one or more units. The same results can be accomplished by using the EXTRACT function of CA SMF Director, which retrieves data from the previously created history files. For more information on the SPLIT and EXTRACT functions, see the CA SMF Director User Guide at <http://ca.com/support>.

A complete set of SMF record types and subtypes, for each CA MICS component that inputs SMF data, can be found in `sharedprefix.MICS.PARMS(cccSMFD)`, where `ccc` is the three-character product identifier. Each PARMS member lists the available SMF record types and subtypes for that component in a format suitable for use in CA SMF Director SPLIT and EXTRACT statements. These members are provided as examples only; the CA SMF Director JCL should not reference `sharedprefix.MICS.PARMS(cccSMFD)`.

Chapter 7: PARAMETERS

Defining the installation parameters for the CA MICS Analyzer Option for CICS requires you to understand your data center and its needs and to translate that understanding into CA MICS parameters. In particular, this chapter instructs you to:

- o Make several policy decisions
- o Complete various worksheets
- o Translate the worksheet entries into the corresponding CA MICS parameter library member entries

These activities represent the major portion of the product installation process.

Information in this chapter focuses on parameter specifications that are unique to the CA MICS Analyzer Option for CICS. Chapters 2 and 3 of the the PIOM document the mechanics of the CA MICS installation process and include checklists that describe each installation step. You should use this chapter as a detailed reference in conjunction with the PIOM.

If you have a question at any time, contact Technical Support at <http://ca.com/support>.

This section contains the following topics:

[7.1 Environmental Considerations](#) (see page 552)

[7.2 Complex Level Parameters](#) (see page 553)

[7.3 Unit Level Parameters](#) (see page 566)

7.1 Environmental Considerations

Before coding the installation parameters, you need to know about the CICS environment at your data center. To do so, we suggest that you perform the following:

- o Identify the CICS regions whose data is to be processed by the CA MICS Analyzer Option for CICS. Determine the monitor that performs the data collection for each region. The CA MICS Analyzer Option for CICS supports data from the CICS Monitoring Facility (CMF) and ASG-TMON. Detailed information on the types of records used by the CA MICS Analyzer Option for CICS is provided in Chapter 6 of this guide.
- o Examine existing user groups, account codes, and job control information used by CICS. This information helps you code appropriate values for the account code parameters of the CA MICS Analyzer Option for CICS.
- o Review the default setting of the installation parameters provided by the CA MICS Analyzer Option for CICS to determine their applicability to your environment.
- o Review current CA MICS parameter specifications:
 - Review database unit specifications to determine which unit or units should include CICS data.
 - Review the SYSID parameter to determine whether it correlates to the CICS data. Your site's SYSID definition is located in the prefix.MICS.PARMS data set associated with each database unit.
 - Review the CA MICS ZONE parameter to ensure that it reflects the variations in CICS data. This parameter is located in the prefix.MICS.PARMS data set associated with each database unit.
 - Examine existing accounting standards for other CA MICS components, if applicable, to ensure that the CA MICS Accounting and Chargeback Option aggregates information consistently.

7.2 Complex Level Parameters

This section shows you how to specify the complex level parameters that define the processing of the CA MICS CICS Analyzer. The definitions apply to all CA MICS data base units in which the CICS Analyzer is installed.

The following topics are presented:

- 1 - Analyzer Definition Statements (CICGENIN)
- 2 - Define CICS Account Codes (CICACCT)
- 3 - Code CICS Account Code Derivation Exit (CICACRT)

7.2.1 Definition Statements (CICGENIN)

Each CA MICS component has a member that defines component generation statements in sharedprefix.MICS.GENLIB. The member's name is cccGENIN, where ccc is the three-character component identifier. For the CA MICS Analyzer Option for CICS, this member is CICGENIN.

Chapter 4 of the System Modification Guide describes the statements that comprise the GENIN members. Statements specific to this component are described below.

THE OPTION STATEMENT

CICGENIN provides the OPTION statement to indicate which groups of data elements are to be kept in the database. The CA MICS Analyzer Option for CICS provides a default OPTION statement that specifies that response distribution elements such as count and percent distributions are kept and that only data elements from the CMF data source are kept. You should change the default definition to match your organization's requirements.

Note: CICGENIN contains both uppercase and lowercase characters. The keywords for the OPTION statement should be entered in uppercase.

The statement format is:

```
OPTION keyword keyword keyword keyword keyword keyword
```

The valid keywords are:

CMF	- Keep CMF data elements
MON	- Keep ASG-TMON data elements
EPLDB2	- Keep OMEGAMON II DB2 data elements
RESP	- Keep response distribution elements (RESPONSE is also valid)
OMEGU	- Keep OMEGAMON II user area data elements (OMEGUSER is also valid)
DBCTL	- Keep DB2 Data Base Control (DBCTL) data elements
NOCMF	- Do not keep CMF data elements
NOMON	- Do not keep ASG-TMON data elements
NOEPLDB2	- Do not keep OMEGAMON II DB2 data elements
NORESP	- Do not keep response distribution elements (NORESPONSE is also valid)
NOOMEGU	- Do not keep OMEGAMON II user area data elements

(NOOMEGUSER is also valid)
NODBCTL - Do not keep DB2 Data Base Control (DBCTL) data elements (For new install, NODBCTL is the default)

You must use the keyword CMF if you are processing type 110 look-alike records such as those produced by OMEGAMON II for CICS.

You must specify one keyword for each data source. The keywords can be specified in any order and are separated by one or more blanks.

Each of the above keywords controls the activation of data elements that are associated with a cluster code. The following table describes the cluster codes used in the CA MICS Analyzer Option for CICS.

CA MICS Analyzer Option for CICS Cluster Codes

Code	Keyword	Description
----	-----	-----
09	MON	ASG-TMON for CICS
11	CMF	CMF
15	CMF MON	CMF and ASG-TMON for CICS
16	CMF	CMF
17	MON	ASG-TMON for CICS
18	EPLDB2	OMEGAMON
19	MON	ASG-TMON for CICS
20	RESPONSE	Response distribution
21	OMEGU	OMEGAMON
25	CMF MON	CMF and MON DBCTL
26	CMF	CMF DBCTL
29	MON	MON DBCTL

Note: For CICS, because a data element can be derived from multiple monitor data sources, each monitor option (CMF, MON) may be associated with multiple cluster codes. For example, if you specified `OPTION CMF RESP NOMON NOEPLDB2`, the data elements with cluster codes 11, 15, 16, and 20 are activated, while elements with cluster codes 9, 17, and 18 are deactivated.

Examples:

The `OPTION` statement for data centers using only CMF as input data source is:

```
OPTION CMF NOMON RESP NOOMEGU DBCTL
```

The OPTION statement for data centers using only ASG-TMON as input data source is:

```
OPTION NOCMF MON RESP NOOMEGU DBCTL
```

The OPTION statement for data centers using CMF and ASG-TMON as input data sources is:

```
OPTION CMF MON RESP NOOMEGU DBCTL
```

In addition to controlling the content of your database through the OPTION statement, you can also manually activate or deactivate data elements by modifying sharedprefix.MICS.GENLIB(CICGENIN). The CA MICS Analyzer Option for CICS provides data element definitions for most of the data fields in the input records. However, a small percentage of the data elements are shipped as inactive. You may want to review the CICGENIN member to determine if any of the inactive fields should be activated.

THE COMPRESS STATEMENT

The SAS system gives you the option to create variable-length or fixed-length observations in a SAS data set. Variable-length observations differ from fixed-length observations in that the former are usually smaller because the blank spaces used to pad fixed-length observations are removed.

You instruct SAS to create variable length observations by specifying the COMPRESS= option. SAS data set compression can be implemented for individual data sets or across the entire SAS system by specifying COMPRESS= on either a DATA statement (for the individual data set named on the DATA statement) or an OPTIONS statement (for the entire system).

For more information on the COMPRESS= option, see the SAS Institute documentation.

Files that should not be compressed are:

- CICCDC, the CICS Dictionary file
- the CICCXY parallel files (CICGDP, CICGSM, CICCXM, CICCXY, CICGLD, CICGTS, CICGTD, and CICGST)

7.2.2 Define CICS Account Codes (CICACCT)

The CICS Analyzer provides nine levels of account codes for you to associate users and user groups with the CICS workload. Account codes are used as file summarization keys by the CICS User Activity (CICCSU) and CICS User Application Count (CICCAC) files, which means that the account code structure you define impacts the amount of DASD space required for your CICS Analyzer files. The more levels of summarization you require, the more DASD space you will use.

To define the account codes to the CICS Analyzer, you need to modify the member CICACCT in `sharedprefix.MICS.PARMS`. CICACCT defines the number of account codes that are to be carried in the CICCSU and CICCAC files and the length and the label associated with each account code. Its definitions apply to the above files in all CA MICS data base units that contain the CICS Analyzer.

PREPARATION

Each installation has its own method for associating the work performed through CICS transactions with the responsible user, projects, or departments. Before defining the CICS account codes, it is important that you investigate your installation's accounting standards to:

- o Identify the current organizational coding system (e.g., cost center coding system identifying the division, department, project, and employee) and determine how the account codes are specified. For example, the division and department may be determined by the first character of the CICS terminal identifier, and the project may be determined from the first two characters of the CICS transaction ID.
- o Identify if, and how, the codes are verified to ensure that they correspond to a valid definition. We recommend that account code validation be performed in all cases and that unidentified or invalid account codes be assigned to a special installation overhead account code. This approach provides you with two benefits. First, it groups all invalid codes under the same category and therefore requires less DASD space for storing the CICS Information Area files. Second, it enables you to easily determine how much of this overhead activity is taking place.

Consider the following suggestions in determining the number of account codes required to meet your installation's reporting and analysis needs:

- o Account codes are part of the summarization keys for the files that support them. As such, at least one record is generated for each combination of values. Therefore, the higher the number of account codes, the more DASD space the data base will require. But at the same time, a higher number of account codes also supports data analysis at a greater detail.
- o If you anticipate needing to expand the account code structure in the future, establish an extra account code now to eliminate the need to retrofit the data base later to add the new account code.

DEFINING ACCOUNT CODES

You can define up to nine levels of account codes to the CICS Analyzer. The account codes are named CICTACT1 to CICTACT9 in the CA MICS Data Base.

Figure 7-1 provides a worksheet for defining the account codes. You can fill in the worksheet as you go through the following explanation on coding CICTACCT.

The CICTACCT member in sharedprefix.MICS.PARMS contains one statement per account code definition. Each statement contains several parameters, which are free-form (separated by one or more blanks) but positional. Blank statements are not permitted. Comments can be coded by beginning the statement with an asterisk (*).

The statement format is:

level length 'descriptive title'

or

level mask length 'descriptive title'

where:

level = The level of importance of each account code, with level 1 being the most important and the highest level number being the least important. The levels are defined sequentially starting

with 1. You can define up to nine levels. This parameter is required.

mask = An optional parameter that can be used to deactivate the account code in specified timespans. Specifying a timespan mask requires coding T(.....) as the second parameter in a statement, where each "." represents a file timespan, in the order of DETAIL, DAYS, WEEKS, MONTHS, YEARS, and TABLES. For each timespan, you can specify a Y to indicate that the account code is active or an N to indicate that the account code is inactive.

If this parameter is skipped, it has a default value of T(YYYYYY).

The following rules apply if you code the mask:

- o DETAIL must always be Y. Account codes may not be deactivated in this timespan.
- o If DAYS is N, then WEEKS, MONTHS, and YEARS must also be N. If the account code is inactive in the DAYS timespan, it cannot be active in higher timespans.
- o If MONTHS is N, then YEARS must be N. If the account code is inactive in the MONTHS timespan, it cannot be active in the YEARS timespan.

length = The length of the account code. The length may range from 1 to 30. This parameter is required.

descriptive title = The title that describes the account code. The length of the title is 1 to 40 characters. The title is embedded in quotes (''). This is a required parameter.

EXAMPLES

A sample CICACCT with three account codes, the last two of which are turned off in the MONTHS and YEARS timespans:

```
1          2 'DIVISION'
2 T(YYNNN) 8 'DEPARTMENT'
3 T(YYNNN) 4 'USER ID'
```


A sample CICACCT that enables you to report on which terminal a given user was using when that user performed work:

```
1          4 'COST CENTER'  
2 T(YYNNNN) 4 'USER ID'  
3 T(YNNNNNN) 4 'TERMINAL'
```

The default CICACCT delivered with the CICS Analyzer has the three account codes shown below:

```
1 4 'COST CENTER'  
2 4 'TRANSACTION ID'  
3 8 'USER ID'
```

The CICS Analyzer's reports use CICTACT2 for application reporting if the CICC AU file is not active. If the CICC AU file is active, the application reports use CICA PU (the application unit identifier), whose default value is the transaction identifier.

If you change the structure of either CICTACT2 or CICA PU, ensure that your application reports are accurate.

INSTALLATION PREPARATION WORKSHEET: CICS Account Code Specifications				
PARMS Library Member is CICACCT				
Reference: Section 7.2.2, CA MICS CICS Analyzer Guide				
ACCOUNT CODE LEVEL (1-9)	TIMESPAN MASK	LENGTH	DESCRIPTIVE TITLE	
-	T(_____)	--	'_____'	
-	T(_____)	--	'_____'	
-	T(_____)	--	'_____'	
-	T(_____)	--	'_____'	
-	T(_____)	--	'_____'	
-	T(_____)	--	'_____'	
-	T(_____)	--	'_____'	
-	T(_____)	--	'_____'	
-	T(_____)	--	'_____'	
-	T(_____)	--	'_____'	

.....5...10...15...20...25...30...35...40...45...50...55...60...65...70..

Figure 7-1. CICS Account Code Specifications Worksheet

7.2.3 Code CICS Account Code Derivation Exit (CICACRT)

After you have defined CICACCT for the number of account codes to be carried in the CICS files, your next step is to code the CICS Account Code Derivation Exit (CICACRT) in `sharedprefix.MICS.PARMS` to create the SAS routine that will derive the account code data elements during the DAY040 step. For example, if you have defined four account codes in CICACCT, then the CICACRT exit must contain SAS code to derive the data elements CICA1, CICA2, CICA3, and CICA4 for each transaction record processed.

You are responsible for testing the accuracy of the exit routine; however, the CA MICS Analyzer Option for CICS provides a sample exit routine to help validate the modifications that you make to the routine. The worksheet for coding the CICACRT exit is shown in Figure 7-2.

The CICACRT exit is invoked after each transaction record from CMF or ASG-TMON for CICS TS (TCE) has been read by the CA MICS Analyzer Option for CICS. As a result, you can access any CA MICS data element that is created during the input processing step. This includes any data fields that are read from the records, but are not retained as data elements in the CA MICS database.

The member `CICDEPEL` in `sharedprefix.MICS.SOURCE` contains SAS variable names for each data field input from CMF and ASG-TMON for CICS TS (TCE) records. You can browse this member to determine the variables you need for coding the CICACRT Exit. As an option, you can also browse member `CICGENIN` in `sharedprefix.MICS.GENLIB`. All data elements that are active for the transaction work file `CICCSW` are available for use with the CICACRT Exit.

The following CA MICS data elements or temporary SAS variables are most often used to derive the account code values:

- TRANCODE - The CICS transaction ID
- TERMINAL - CICS terminal identifier
- USERID - The CICS RACF user identification
- TRANSTYPE - C, S, M, or L for conversational, short, medium, or long transaction types
- OPERID - CICS operator identification from the CICS Signon

Table (only applicable to releases prior to CICS)

SYNTAX

The CICACRT exit may contain any valid SAS statements for a DATA step except RETURN and DELETE. Coding either of these two statements may cause unpredictable results in DAY040 processing, since the input processing is performed within a loop, and control is not returned to the beginning of the SAS DATA step for each record input.

Examples:

The default CICACCT describes a data center with three account codes that are derived as described below:

- o CICTACT1 is the first two characters of the transaction ID.
- o CICTACT2 is the full four-character transaction ID.

Because the default CICALPU contains the transaction ID, the account code structure described here is the same as setting CICTACT2 to CICALPU.

- o CICTACT3 is the eight-character user ID.

Furthermore, if the transaction ID is either blank or begins with the letter C, the transaction is accounted for in the data center's overhead account.

The default CICACRT exit contains the following SAS code:

```
/* CICS ACCOUNT CODE DERIVATION EXIT                                */
/*                                                                    */
/* ACCOUNT CODE 1 REPRESENTS COST CENTER WHICH IS DERIVED          */
/* FROM THE FIRST TWO POSITIONS OF CICS TRANSACTION ID.            */
/*                                                                    */
/* ACCOUNT CODE 2 REPRESENTS CICS TRANSACTION ID.                  */
/*                                                                    */
/* ACCOUNT CODE 3 REPRESENTS CICS USER ID.                          */
/*                                                                    */

IF TRANCODE NE : 'C' AND TRANCODE NE : ' ' THEN
  CICTACT1=SUBSTR(TRANCODE,1,2);
ELSE CICTACT1='OVHD';
CICTACT2=TRANCODE;
CICTACT3=USERID;
```

The following CICACRT might be used by a data center choosing to report on activity by the terminal used to access the system:

```
CICACT1=SUBSTR(TRANCODE,1,2);
CICACT2=TRANCODE;
IF TERMINAL NE : ' ' THEN
  CICACT3=TERMINAL
ELSE CICACT3='OVHD';
```

<pre> +-----+ INSTALLATION PREPARATION WORKSHEET: CICS Account Code Routine Definition PARMs Library Member is CICACRT Reference Section: 7.2.3, CA MICS Analyzer Option for CICS Guide +-----+ * VALIDATE FOR VALID ACCOUNT CODES, WHERE POSSIBLE ; IF account data is not valid GOTO ACCTOVHD ; _____ _____ _____ _____ * BUILD ACCOUNT CODE FIELDS CICACT1=field source 1 ; CICACTn=field source n ; _____ _____ _____ GOTO ACCTRTEX ; * LINKED ROUTINE TO BUILD INSTALLATION OVERHEAD ACCOUNT CODES ; ACCTOVHD: CICACT1='overhead category' ; CICACTn='overhead category' ; _____ _____ _____ _____ ACCTRTEX: _____ 5...10...15...20...25...30...35...40...45...50...55...60...65...70.. +-----+ </pre>

Figure 7-2. CICS Account Code Derivation Exit Worksheet

7.3 Unit Level Parameters

This section shows you how to define the unit level parameters for the CICS Analyzer. These parameter definitions exist for each CA MICS database unit in which the CICS Analyzer is installed. Some of the parameters are used by the CICS Analyzer parameter generation CICPGEN to generate code that will be executed during the DAY040 step. Other parameters are read directly during DAY040 to determine the derivation of CA MICS data elements and handling of error diagnostics.

The following topics are presented:

- 1 - CICS Parameter Generation (CICPGEN)
- 2 - CICS Processing Options (CICOPS)
- 3 - INPUTRDR and INPUTCIC PARMS Members
- 4 - Reserved
- 5 - CICS Application Unit Definition (CICAPU)
- 6 - CICS Application Unit Derivation Routine (CICAURT)
- 7 - CICS Relative Longevity Routine (CICRLRT)
- 8 - CICS Processing Thresholds (CICTHRSH)
- 9 - CICS Multisystem Account Derivation Exit (CICMSAC)
- 10 - Database Space Modeling (DBMODEL)

For information on report options (CICRPTnn), see section 3.3.2 in this guide.

For information on CICALONE, see section 9.3.3 in this guide.

7.3.1 CICS Parameter Generation (CICPGEN)

The CICS Parameter Generation (CICPGEN) inputs the unit level parameters and generates SAS code that is executed by the DAY040 step. The generated code contains SAS macros and formats that dictate many aspects of input processing logic. For example, it identifies the CICS regions to be processed, the ddnames from which the data is read, and the number of work file pairs to be used.

The JCL for CICPGEN resides in prefix.MICS.CNTL.

The following chart shows the input parameters processed and output source modules written by CICPGEN. The input parameters are documented in subsequent sections of this chapter. The parameters are coded in the following members in prefix.MICS.PARMS:

- o CICOPS - CICS Processing Options Definition
- o INPUTCIC - CICS Input DD Statement

INPUT prefix. MICS. PARMS	OUTPUT prefix. MICS. PARMS	OUTPUT prefix.MICS. USER.SOURCE module: macros:	OUTPUT prefix. MICS. MUOLIB
CICOPS INPUTCIC	WORKCIC	#CICMSTR: %CICCDAL %CICCDCU %CICCMR %CICCSF %CICCCST %CICCSW %CICIPCV %CICPROC %CICRVAL %CICTRAN %CMFINP %CMRDEL %CSFDEL %CSFOUT %CSFMERG %CSTMERG %CSUSORT %CSUMERG %CSUDEL %CSWOUT %CSWDEL %CSYMERG %EPLINP %EPLJFCB %MONFLGS %MONINP %MONJFCB %TAPECSU %TAPEDSN #CICMSAC	\$CICSEL \$CICTRAN \$CICTYP \$CTGAPID

7.3.2 CICS Processing Options (CICOPS)

The CICOPS member of prefix.MICS.PARMS contains CA MICS Analyzer Option for CICS processing options for each CICS region that should be processed in a unit database. In addition, CICOPS contains environmental options that are unique to each unit, such as work file specification and optional tape file creation.

Regions that are not defined in CICOPS are excluded from input processing. The CA MICS DAILY job will abend when undefined regions are encountered during input. However, you can bypass the abend by specifying a tolerance threshold for undefined regions. If you want to exclude regions from CA MICS processing, see section 7.3.8 for details on specifying a tolerance threshold.

Operational statements are stored in the prefix.MICS.PARMS cccOPS member, where ccc is the component identifier, and are incorporated into the CA MICS system by running the prefix.MICS.CNTL(cccPGEN) job.

```
*****
*
* NOTE: CHANGES to prefix.MICS.PARMS(cccOPS) members
*        REQUIRE EXECUTION of prefix.MICS.CNTL(cccPGEN)
*        to take effect.
*
* In addition, any change to parameters that
* impact the DAILY operational job JCL such as,
*
*   o changing RESTART NO to RESTART YES,
*
*   o WORK parameter changes when RESTART NO is in
*     effect,
*
*   o Specifying TAPEfff (if this product supports
*     a DETAIL level TAPE option),
*
*   o or changes to prefix.MICS.PARMS(INPUTccc),
*
* will require regeneration of the DAILY job by
* executing prefix.MICS.CNTL(JCLGEND) or by
* specifying DAILY in prefix.MICS.PARMS(JCLGENU)
* and executing prefix.MICS.CNTL(JCLGENU).
*
* Refer to the checklist (if provided) for updating
* cccOPS parameters and running required generation
* jobs.
```

The CICOPS member contains the following statements,
described in detail below. A worksheet for recording the
statement values for CICSOPS is shown in Figure 7-3.

CICOPTS statement
MSACCOUNT statement
RESP statement
STATCOPY statement
TRANSLATE statement
GATEWAY ACTIVE and APPLID statements
WORK, SASWORK, and MULTWORK statements
Internal Step RESTART statements
Incremental Update statements
DETAIL Tape Processing statements
VERIFY RELEASE statement

7.3.2.1 CICOPTS Statement

The CICOPTS statement identifies a CICS region whose data should be processed by the CA MICS Analyzer Option for CICS. One CICOPTS statement is required for each CICS region. Any region found in input that is not defined by a CICOPTS statement is excluded during input processing.

If the same CICS region can be executed on multiple systems, use the generic SYSID feature to define the regions using a single CICOPTS statement. Otherwise, you must specify one CICOPTS statement for each unique combination of CICS region ID and SMF system ID. For example, if CICSPROD can be run on either SYSA or SYSB and you do not use generic SYSID support, you must define two CICOPTS statements: one for CICSPROD on SYSA and another for CICSPROD on SYSB. Details on generic SYSID support are given below in the discussion of the ORGSYSID parameter.

The CICOPTS statement format is:

```
CICOPTS orgsysid cicsid ddname mon gmt-ofs applid_or_sysidnt
        int selcode
```

Note: The parameters are positional and are separated by one or more blanks.

where:

orgsysid = Original system ID. This is the one- to four-character SMF system identification. This identifies the system on which this CICS system is executed. The value of this identifier must match an ORGSYSID value defined in prefix.MICS.PARMS(SYSID).

A generic SYSID is defined as four question marks (????). By specifying a value of ????, the actual SMF system ID is obtained from the input data. This option cannot be used with CMF journal data, which does not contain the SMF system ID. The SMF system ID obtained from the input data must match an ORGSYSID defined by prefix.MICS.PARMS(SYSID). By using the generic SYSID feature, you can use one CICOPTS statement to associate a CICS region with multiple SMF system ID values.

cicsid = One- to four-byte CICS identifier that you assign

to identify a CICS region to CA MICS. The CICSID must start with a letter. It may contain letters, digits, and underscores. Special characters are not allowed.

CA MICS combines CICSID and ORGSYSID (SMF system ID) to uniquely identify the CICS region. That is, two CICS regions executing on two different systems may have the same CICS ID, because the combination of ORGSYSID and CICSID would be unique. (The SMF systems are different.) Two CICS regions running on the same system must have different CICSIDs to ensure the uniqueness of the ORGSYSID/CICSID combination.

Note that the value chosen for CICSID cannot be the same as a value chosen for CTGID in a GATEWAY APPLID statement.

The maximum number of system, subsystem, and combined entries cannot exceed the limit of the prefix.MICS.CHECKPT.DATA data set. The default maximum number of entries is 100. This number is adequate for most data centers; however you can reconfigure a unit's checkpoint size to allow a maximum of 1000 entries.

Note: For more information about the checkpoint file size and changing the default, see Section 2.3.2.4 of the PIOM, Site Characteristics (SITE).

ddname = Input ddname. This parameter specifies the ddname from which the CA MICS Analyzer Option for CICS is to read the input data. Values for this parameter must be either SMF, MON, TCE or a one-to eight-byte ddname. The names INPUTSMF and INPUTMON are both reserved and cannot be specified on the CICOPTS statement.

SMF - The measurement data for this CICS region is found on the SMF log. If you specify SMF here, then you should modify prefix.MICS.PARMS(INPUTSMF) and specify the JCL for the SMF data set under the ddname INPUTSMF. Multiple data sets can be concatenated under the ddname INPUTSMF.

Notes:

1. If you specify SMF for this parameter, you should specify SMFRECORDING CIC in prefix.MICS.PARMS(JCLDEF).
2. Normally, the INPUTCIC member of the prefix.MICS.PARMS library should not contain any JCL with the ddname INPUTSMF. However, if you have modified CA MICS to remove the DAYSMF step, you can code the ddname INPUTSMF in prefix.MICS.PARMS(INPUTCIC) and remove the SMFRECORDING CIC statement from prefix.MICS.PARMS(JCLDEF).
3. Do not specify SMF for this parameter if you are using ASG-TMON since it does not write to SMF.
4. Specify SMF for this parameter if the input data contains CMF type 110 look-alike records that are written to SMF, such as those created by OMEGAMON II for CICS.

MON - The measurement data is from ASG-TMON release 7.1 or higher. If you specify MON here, then you must modify prefix.MICS.PARMS(INPUTCIC) and specify JCL for the input data under the ddname INPUTMON. Multiple data sets can be concatenated under the ddname INPUTMON.

TCE - The measurement data is from ASG-TMON release 2.0 or higher. If you specify TCE here, then you must modify prefix.MICS.PARMS(INPUTCIC) and specify JCL for the input data under the ddname INPUTTCE. Multiple data sets can be concatenated under the ddname INPUTTCE.

ddname - The measurement data is either written on a data set other than the SMF log, such as CMF journal. You must modify prefix.MICS.PARMS(INPUTCIC) and specify JCL for the input data set under this ddname. The names INPUTSMF and INPUTMON and INPUTTCE are reserved and cannot be used here.

Notes:

1. You may code the same ddname on multiple CICOPTS statements to specify that data for multiple regions is read from the same input. You must code the JCL for this ddname in prefix.MICS.PARMS(INPUTCIC). Multiple data sets can be concatenated under the same ddname.
2. You may define each CICS region under a separate ddname. This method requires a unique ddname for each region. It is not normally used unless you want to specify NULL for the CICS APPLID or system identification. More details on NULL APPLID are given below.

mon CICS monitor identification. This parameter specifies the CICS monitor that collected data for this CICS system. This parameter must be one of the following:

CMF - The data is collected by the CICS Monitoring Facility (CMF) or other CICS monitors that produce CMF type 110 look-alike records.

MON - The data is collected by ASG-TMON.

TCE - The data is collected by ASG-TMON 2.0+

gmt-ofs Time offset from Greenwich Mean Time. For ASG-TMON, specify 0. For CMF input at CICS/ESA release 4.1 and higher, specify 0. Otherwise, specify the offset (time difference) from Greenwich Mean Time (GMT) as recorded at Greenwich, Great Britain. Note that the GMT offset, even if specified as non-zero, is ignored for all CICS systems that are V4.1 or later.

If your systems are set to local time instead of GMT, specify 0. The MVS specification for this offset is found in SYS1.PARMLIB(CLOCKxx). If this member is empty or does not exist, your system is running at local time.

If your system is running at GMT time and your

CMF input is pre-CICS/ESA 4.1, then you must specify the GMT offset to adjust the CMF transaction timestamps from GMT to local time.

The GMT offset must be specified in whole hours ranging from 13 to -13. Since CA MICS subtracts the GMT offset from CMF timestamps, you should specify a positive value for North American countries and a negative value for European countries east of Great Britain. For example, you should specify 5 for gmt-ofs if you are in the United States at Eastern Standard time, since you are 5 hours behind GMT time. (GMT time is 11 a.m.; local time is 6 a.m.)

Note: This offset is affected by daylight savings time. In order for the database to remain accurate, you must update this parameter and run the CICPGEN job if you change to or from daylight savings time.

applid
or
sysidnt

VTAM application identifier or the CICS system identifier (SYSIDNT).

For CMF and ASG-TMON/ESA (TCE) 2.0 and higher, you should specify the VTAM application identifier (APPLID) as defined in the CICS system initialization table DFHSIT or the CICS terminal control table DFHTCT.

An exception to the rule is when your input data contains data for only one CICS region. In this case, you can use the word NULL in place of the APPLID on each CICOPTS statement that references a unique ddname.

Note: If you do not specify NULL and the APPLID is incorrect, the data will be excluded.

For ASG-TMON, you should specify the four-byte CICS system identification (SYSIDNT) as defined in the CICS system initialization table DFHSIT or the CICS terminal control table DFHTCT.

An exception to the rule is when your input data contains data for only one CICS region. In this case, you can use the word NULL in place of the SYSIDNT on each CICOPTS statement that references a unique ddname.

Note: If you do not specify NULL and the SYSIDNT is incorrect, the data will be excluded.

You may need to ask your CICS system programmer for assistance in determining the correct APPLID or SYSIDNT for your CICS regions.

int = Default CICS system sample interval. This parameter is not used for CMF (3.1 and higher) or ASG-TMON. Specify two question marks (??) for any of these input sources. For CMF 2.1 and prior releases, specify either the number of minutes for the system recording interval (see FREQ option in the CICS monitoring control table DFHMCT), or two question marks (??) if you do not know what the recording interval is.

For CMF 2.1, the two question marks cause the CA MICS Analyzer Option for CICS to derive duration from the input data without validating it against the expected interval value specified here.

selcode = Optional parameter to selectively input CMF accounting and exception class data. Its values are 0, 2, 4, or 6, which mean the following:

- 0 - Do not input accounting and exception data.
- 2 - Input accounting data (CMF 2.1).
- 4 - Input exception data.
- 6 - Input both accounting and exception data.

The default value is 0. Do not code this parameter for ASG-TMON input.

CMF performance class data is always processed by the CA MICS Analyzer Option for CICS. This parameter does not affect CMF performance class data.

If you want to input accounting or exception class data, you are responsible for coding the USRSCMF user exit to output the data to the appropriate CICS files.

Examples:

Sample CICOPTS statements are shown below.

1. CICOPTS ???? CIC1 SMF CMF 0 CICSPROD ??
 CICOPTS ???? CIC2 SMF CMF 0 CICSTEST ??
 CICOPTS ???? CIM1 MON MON 0 CIMP ??
 CICOPTS S090 CIM2 MON MON 0 CIMT ??
 CICOPTS ???? D40X TCE TCE 0 D40XCICS ??

The above statements define five CICS regions. The first two regions input CMF data (including CMF look-alike data produced by OMEGAMON for CICS/ESA) written to SMF, and the next two input data from ASG-TMON. Regions CIC1 and CIC2 are defined using the generic SYSID feature. The APPLIDs are required for each region, since they are concatenated to the same ddname (in JCL the ddname is INPUTSMF).

Regions CIM1 and CIM2 are from ASG-TMON. CIM1 is defined using the generic SYSID feature while CIM2 is not.

Note: For each of the ASG-TMON regions, a four-byte system identification is specified in place of the eight-byte APPLID.

Region D40X is from ASG-TMON for CICS/ESA 2.0. D40X is defined using the generic SYSID feature.

Note: For each TCE region, the eight-byte VTAM application ID is used.

2. CICOPTS R033 CICP INCMFA CMF 5 NULL ??
 CICOPTS R033 CICT INCMFB CMF 5 NULL ??

The above statements define two CICS regions with an SMF system ID of R033. Both regions are recording to CICS journal and are input to CA MICS through separate ddnames (INCMFA and INCMFB). The APPLIDs are both NULL, since the ddnames are unique. The interval value is ??, which means the CA MICS Analyzer Option for CICS will not validate the calculated duration value.

A complete CICOPTS member follows.

Sample CICOPTS member

```
RESP  1 2 5 10 15 30 60
TRANSLATE PF1 -----PF1
WORK 3
CICOPTS ???? CIC1 SMF      CMF 0 CICSPROD ??
```

```
CICOPTS ???? CIC2 SMF    CMF 0 CICSTEST ??
CICOPTS R033 CIGP INCMFA CMF 5 NULL    ??
CICOPTS R032 CIGT INCMFB CMF 5 NULL    ??
CICOPTS ???? CIMN MON    MON 0 CIMN    ??
CICOPTS S090 CITX INMON1 MON 0 CITX    ??
CICOPTS S084 CITX INMON1 MON 0 CITX    ??
CICOPTS ???? ROCK TCE    TCE 0 ROCKVILL ??
GATEWAY ACTIVE
GATEWAY APPLID CTGPROD1 TGP1
GATEWAY APPLID CTGPROD2 TGP2
```

7.3.2.2 MSACCOUNT Statement

The MSACCOUNT statement is used to activate the Multisystem Account Derivation Routine. This routine allows you to reassign previously derived account codes based on Unit of Work ID and Network Name for data from CICS regions connected via Multiregion Option (MRO) or Intersystem Communication (ISC). For example, a record produced by a File Owning Region (FOR) does not contain the true transaction ID. Instead, it may contain the mirror transaction ID CSMI. To account for the file access under the true transaction ID, the detail data from the Application Owning Regions (AOR) must be sorted ahead of the FOR record so that the real transaction ID (usually stored in one of the CA MICS account codes) can be propagated to the FOR record.

The MSACCOUNT statement has no parameters. It is specified as follows:

MSACCOUNT

If the MSACCOUNT statement is included in the CICOPS member, then the Multisystem Account Derivation Exit (CICMSAC) is invoked during DAY040 processing. Otherwise, the exit is not invoked. For instructions on coding the exit routine, see Section 7.3.9 in this guide.

Note that the MSACCOUNT option does not collapse related ISC/MRO records to single observations in the CA MICS database. Its sole purpose is to propagate accounting information for participating ISC/MRO regions as needed. Activating this option will prolong DAILY processing time because extra sort and data steps are required to manipulate the detail data.

Eliminating the MSACCOUNT option for CTS 4.2 and above

For all methods of transport, CTS 4.2 and above provides, the originating transaction's identification fields for records shipped via MRO or ISC to other regions. The two commonly used identification fields for CICS account code assignment, TRANCODE and USERID, have corresponding originating fields in the CSU file for CTS 4.2 and above:

CSUOTRAN	Originating Task Transaction ID
CSUOUSID	Originating Userid

By using these originating fields for account code derivation, you might be able to eliminate the need to invoke MSACCOUNT processing to propagate account code information from the originating transaction, thereby saving substantial processing time in the CICS Daily step. Also, removing MSACCOUNT processing eliminates the requirement that data from all related TOR, AOR and FOR regions be input to the same MICS unit.

You can eliminate MSACCOUNT processing if:

- o All your CTS data is 4.2 and above. If you process 4.2+ data with data for any releases earlier than 4.2, you need to continue using the MSACCOUNT feature until you have migrated all input regions to release 4.2 or above.
- o Your CICACRT routine does not depend on a data element whose value is propagated from the originating transaction in your CICMSAC routine, and there is not a corresponding originating field. If so, you need to continue using the MSACCOUNT feature.
- o There is no adverse impact on your accounting. Under certain conditions, removal of MSACCOUNT processing and using originating fields may produce different account code values than with MSACCOUNT processing. Be sure to test to determine if there are any differences and if so, consider their impact on your account processing.

Steps to eliminate MSACCOUNT processing:

- o Modify your CICACRT routine to use the originating fields in place of the existing identification fields if a surrogate transaction is encountered. For example:

Current sharedprefix.PARMS(CICACRT):

```
/* CICS ACCOUNT CODE DERIVATION EXIT      */
/*                                          */
CICACT1=TRANCODE;
CICACT2=USERID;
```

New sharedprefix.PARMS(CICACRT):

```
/* CICS ACCOUNT CODE DERIVATION EXIT      */
/*                                          */
/* IF MRO TRANSACTION, USE ORIGINATING ELEMENTS */
IF CSUTRTFL EQ 0FX THEN DO;
  CICTACT1=CSU0TRAN;
  CICTACT2=CSU0USID;
END;
/* IF NOT MRO TRANSACTION, USE STANDARD ELEMENTS */
ELSE DO;
  CICTACT1=TRANCODE;
  CICTACT2=USERID;
END;
```

- o Remove the MSACCOUNT statement from prefix.PARMS(CICOPS) and run CICPGEN.

7.3.2.3 RESP Statement

The RESP statement defines seven response time threshold limits that are used by the CA MICS Analyzer Option for CICS for CICS response time distribution calculations. A RESP statement is required if you are processing CICS data and you have coded one or more CICOPTS statements in CICOPS.

If your DAY040 step only processes CICS Transaction Gateway SMF type 111 records, you do not need a RESP statement.

The RESP statement format is:

```
RESP limit1 limit2 limit3 limit4 limit5 limit6 limit7
```

where:

limit1-limit7 = Each response time threshold limit is defined in seconds. It can be one to four digits, with a maximum value of 3600. Subsecond response time limits are coded in the form .nnn. The limits can be separated by one or more blanks.

Sample RESP statement:

```
RESP .25 .50 .75 1 5 10 15
```

With the above response time limits, the CA MICS Analyzer Option for CICS accumulates response distribution counters for each transaction as illustrated below:

(ffffDSTx)		(ffffTRSTM)
Response counter 1:	0 <	response <= .25
Response counter 2:	.25 <	response <= .50
Response counter 3:	.50 <	response <= .75
Response counter 4:	.75 <	response <= 1
Response counter 5:	1 <	response <= 5
Response counter 6:	5 <	response <= 10
Response counter 7:	10 <	response <= 15
Response counter 8:	15 <	response

Note that the CA MICS Analyzer Option for CICS creates an eighth counter to capture all response events that are greater than limit 7 as defined by the RESP statement.

7.3.2.4 STATCOPY Statement

The STATCOPY statement allows you to copy CICS statistics data to a sequential tape file during DAILY processing. STATCOPY is only available to CMF users who are running CICS release 3.1 and higher. By creating a sequential file with only statistics data, you are able to analyze a smaller amount of data using programs such as IBM's DFHSTUP without inputting the entire SMF data set.

The STATCOPY statement has no parameters. It is specified as:

STATCOPY

If you do not want to copy any statistics data on tape, simply omit this statement from prefix.MICS.PARMS(CICOPS).

By default, STATCOPY will copy all CICS 3.1 and higher statistics data to tape during DAILY processing. However, the exit USRSTAT is available for you to selectively copy a subset of the statistics data. See section 10.2.3 for detailed information on coding the USRSTAT exit.

Prior to running the DAILY with STATCOPY option activated, you must first define the tape GDG for the sequential output file. This is accomplished by executing the job CICGDGDN. See section 9.3.4 for information related to the CICGDGDN job. See section 10.1.4 for the specific instructions on activating the STATCOPY feature.

7.3.2.5 TRANSLATE Statement

The TRANSLATE statement converts any hexadecimal transaction identifiers, specified by the CICS PCT XTRAN= and TASKREQ= parameters, or attention identifiers (such as program function keys) into any eight-character printable string. The TRANSLATE statement is optional and is only applicable to the processing of CICS data. You can specify multiple TRANSLATE statements, one for each transaction ID to be translated.

Note: If your DAY040 step only processes CICS Transaction Gateway SMF type 111 records, TRANSLATE statements should not be coded.

The TRANSLATE statement format is:

```
TRANSLATE old_id new_id
```

where:

old_id = The original transaction ID to be translated. It must be a recognized attention identifier, such as PF1 or PF2; otherwise, it is assumed to be specified in hexadecimal. Valid attention identifiers are PF1 through PF24, PA1, PA2, PA3, OPID, LPA, CLEAR, or ENTER. A maximum of 300 TRANSLATE statements may be given.

new-id = The eight-byte character transaction ID created from the translation.

Two sample TRANSLATE statements are shown below:

```
TRANSLATE PF1      -----PF1
TRANSLATE F8FFFFFF -PAGEFWD
```

The first TRANSLATE statement converts transaction ID for the Program Function Key 1 into a printable TRANCODE equal to "-----PF1". This transaction record represents data for a transaction defined to the CICS PCT as TASKREQ=PF1.

The second TRANSLATE statement converts monitor data for the transaction identifier F8FFFFFF to a printable TRANCODE equal to "-PAGEFWD". This transaction record represents data for a transaction defined to the CICS PCT as XTRAN=F8FFFFFF.

7.3.2.6 GATEWAY ACTIVE and APPLID Statements

The GATEWAY ACTIVE statement is used to activate the processing of CICS Transaction Gateway SMF type 111 records in a unit where the CA MICS Analyzer Option for CICS is installed.

The GATEWAY APPLID statement is used to associate four-character user-selected Gateway ID values with the eight-byte Transaction Gateway address space APPLID names found in the SMF type 111 records. The user-selected values are stored in data element CICS Transaction Gateway ID (CTGID).

The GATEWAY ACTIVE and GATEWAY APPLID statements are discussed in detail below.

GATEWAY ACTIVE

To initiate processing of the SMF type 111 interval records generated by z/OS-based CICS Transaction Gateway address spaces, a GATEWAY ACTIVE statement must be added to CICOPS.

The GATEWAY ACTIVE statement has the following format:

GATEWAY ACTIVE

There are no additional arguments.

The addition of a GATEWAY ACTIVE statement and execution of prefix.MICS.PARMS(CICPGEN) results in the following:

- o Step DAY040 in the DAILY update job will accept and process SMF type 111 records.
- o SMF type 111 records generated by CICS Transaction Gateway address spaces whose APPLID name is recognized (via GATEWAY APPLID statements) will yield CA MICS Gateway Server Activity (CTGSA) file observations.
- o Where the APPLID name is not recognized, records will be rejected, but the DAY040 step MICSLOG will generate CIC07081W warning messages as shown below:

```

CIC07081W  Rejected CICS Transaction Gateway data
CIC07081W  APPLIDs not defined in CICOPS
CIC07081W
CIC07081W  ORGSYSID  APPLID  RECORD COUNT
CIC07081W  ----  -
CIC07082W  SYS1    CTGAPPL1          96

```

CIC07082W	SYS1	CTGAPPL2	96
CIC07082W	SYS2	CTGAPPLX	96

Note: Specifying a GATEWAY ACTIVE statement without any accompanying GATEWAY APPLID statements is a convenient way to determine the CICS Transaction Gateway APPLID names at your site. The CIC07081W message data can be used to construct GATEWAY APPLID statements for those APPLIDs you want to process.

Because these records are rejected, there is no check to see if your prefix.MICS.PARMS(SYSID) or sharedprefix.MICS.PARMS(CPLXSID) members have ORGSYSID entries associated with the undefined APPLIDs. If you add GATEWAY APPLID statements for any of these APPLIDs, make sure that your SYSID member has entries for the associated ORGSYSIDs.

GATEWAY APPLID

GATEWAY APPLID statements associate a four-byte user-assigned transaction Gateway ID value, with the eight-byte Transaction Gateway APPLID.

The user-assigned four-byte value is used for CA MICS checkpoint processing. It is also stored in data element CICS Transaction Gateway ID (CTGID), which serves as the second-level sort summarization key (after SYSID) in the Gateway Server Activity (CTGGSA) file.

The GATEWAY APPLID statement has the following format:

GATEWAY APPLID applid ctgid

Note: The parameters are positional and are separated by one or more blanks.

where:

applid = Application identifier for the CICS Transaction Gateway address space. The best way to identify the APPLID values at your site is to add a GATEWAY ACTIVE statement to CICOPS, execute prefix.MICS.CNTL(CICPGEN), then run a DAY040 daily update step with SMF type 111 records in the SMF data input.

The DAY040 step MICSLOG will contain CIC07081W messages that specify all Gateway APPLID values

encountered and the z/OS system where they were found.

ctgid = One- to four-byte CICS Transaction Gateway identifier that you assign to identify a Gateway address space to CA MICS. The CTGID must start with a letter. It may contain letters, digits, and underscores. Other special characters are not allowed.

CA MICS combines CTGID and ORGSYSID (SMF system ID) to uniquely identify Gateway APPLIDs. Two Gateways executing on two different systems may have the same APPLID, because the combination of ORGSYSID and CTGID would be unique. (The SMF systems are different.) Two Gateways running on the same system must have different APPLIDs to ensure the uniqueness of the ORGSYSID/CTGID combination.

Note: The value chosen for CTGID cannot be the same as a value chosen for CICSID in a CICOPTS statement.

The maximum number of system, subsystem, and combined entries cannot exceed the limit of the prefix.MICS.CHECKPT.DATA data set. The default maximum number of entries is 100. This number is adequate for most data centers; however you can reconfigure a unit's checkpoint size to allow a maximum of 1000 entries.

For further discussion of the checkpoint file size and changing the default, see Section 2.3.2.4 of the PIOM, Site Characteristics (SITE).

Example:

Sample GATEWAY ACTIVE and GATEWAY APPLID statements are shown below:

```
GATEWAY ACTIVE
GATEWAY APPLID TGWYPRD1 GWP1
GATEWAY APPLID TGWYPRD1 GWP2
GATEWAY APPLID TGWYTST1 GWT1
```

Refer to the checklist in section 10.1.18 for the steps required to process CTG SMF type 111 records.

7.3.2.7 WORK, SASWORK, and MULTWORK Statements

WORK

This statement is optional. It enables sites experiencing either SAS WORK space allocation problems or out of work space conditions during DAYnnn or INCRnnn (where nnn is the job step number), daily or incremental update processing, to allocate multiple WORK files.

You can allocate multiple WORK files for use during the daily and/or incremental update job step. The maximum number of WORK files you can allocate varies by product. These additional work files are used in conjunction with the single work data set allocated by default using the JCLDEF parameters WORKUNIT and WORKSPACE.

Because the individual space allocation requirement for each WORK file is typically much smaller, it is more likely to be satisfied.

To take advantage of multiple WORK files support, edit prefix.MICS.PARMS(cccOPS) and insert a WORK statement as shown below:

```
WORK n data_set_allocation_parameters
```

where n is the number of WORK data sets

Note: The default is one (1).
The maximum is thirty (30).

data_set_allocation_parameters is one or more data set allocation parameters (for example, STORCLAS or SPACE) separated by spaces.

You can also specify the WORK parameter as the following:

```
WORK n XXX pppp ssss
```

where:

n is the number of WORK data sets
XXX is TRK or CYL
pppp is the primary allocation
ssss is the secondary allocation

Note: When allocating any number of SAS WORK data sets, be

aware that one additional SAS WORK data set is automatically allocated to facilitate sorting. For example, if you allocate six SAS WORK data sets, you will actually get seven.

If you omit the `data_set_allocation_parameters` or the `WORK` parameter, the work data sets are allocated according to the values you specified for the `WORKUNIT` and `WORKSPACE` parameters in `prefix.MICS.PARMS(JCLDEF)`. Use the `data_set_allocation_parameters` to override this default, either to alter the space allocation or to use System Managed Storage (SMS) parameters to control data set placement and characteristics.

Note: If you allocate insufficient space for the WORK data sets, `DAYnnn` and/or `INCRnnn` processing will fail and can only be restarted from the beginning.

Note: If internal step restart is active, you can override the WORK data set allocation parameters at execution-time using the `//PARMOVRD` facility. For more information about execution-time override of dynamic data set allocation parameters, see the PIOM, section 2.3.6.

Specify data set allocation parameters, separated by blanks, according to SAS LIBNAME statement syntax. If you need multiple lines, repeat the `WORK` keyword on the continuation line.

`WORK` accepts the engine/host options documented in the SAS Companion for the z/OS environment, including `STORCLAS`, `UNIT`, `SPACE`, `BLKSIZE`, `DATACLAS`, `MGMTCLAS`, and `VOLSER`.

Important! Do not specify the `DISP` parameter.

Example 1:

```
WORK n STORCLAS=MICSTEMP SPACE=(XXX,(pppp,ssss),RLSE)
```

where:

- `n` - is the number of WORK data sets.
- `STORCLAS` - specifies a storage class for a new data set.
The name can have up to 8 characters.
- `SPACE` - specifies how much disk space to provide for a new data set being allocated.
- `XXX` - is TRK or CYL.
- `pppp` - is the primary allocation.
- `ssss` - is the secondary allocation.
- `RLSE` - specifies that free-space should be released

when the data set is closed.

Example 2:

```
WORK n XXX pppp ssss
```

where:

n	- is the number of WORK data sets.
XXX	- is TRK or CYL.
pppp	- is the primary allocation.
ssss	- is the secondary allocation.

Example 3 (multiple lines):

```
WORK n STORCLAS=MICSTEMP UNIT=SYSDA  
WORK SPACE=(xxxx,(pppp,ssss),,,ROUND))
```

where:

n	- is the number of WORK data sets.
STORCLAS	- specifies a storage class for a new data set. The name can have up to eight characters.
UNIT	- specifies the generic unit for a new data set. The name can have up to eight characters.
SPACE	- specifies how much disk space to provide for a new data set being allocated.
XXX	- is TRK or CYL.
pppp	- is the primary allocation.
ssss	- is the secondary allocation.

Note: Since there is some performance impact when using multiple WORK files, you should specify the minimum number of WORK data sets to meet your work space requirements. As a start, try incrementing the number gradually beginning from the default.

WORK Considerations

How Much Space Should You Allocate?

o First Time Implementation of Multiple Work Files

If this is the first time you are implementing multiple work files for this product in this unit, review prefix.MICS.PARMS(JCLDEF) and find the WORKSPACE parameter. It will resemble this sample statement:

WORKSPACE TRK 500 250

The value shows the current SAS WORK space allocation for the unit as a single data set. It also serves as the default value used in the unit's DAYnnn daily update (and/or INCRnnn incremental update) step unless you provide a WORK parameter.

To achieve the equivalent work space allocation of WORKSPACE TRK 500 250 using multiple WORK data sets that will collectively share the work space requirements of the daily and/or incremental update step, you could code either one of these:

WORK 2 SPACE=(TRK,(250,125))

WORK 5 SPACE=(TRK,(100,50))

To determine the total work space, multiply the number of WORK files (n) by the primary (pppp) and secondary (ssss) values specified.

Note: To simplify the example, only the SPACE parameter is shown above. You can follow either with data set allocation parameters like UNIT or STORCLAS as required for your site.

o Adjusting Allocation for Existing Multiple WORK Files

If you have previously implemented multiple WORK file support for this product in this unit, and you want to change either the number of WORK files or the space allocations, examine prefix.MICS.PARMS(cccOPS) and find the existing WORK statement.

- If the existing WORK statement only specifies the number of WORK files but does not contain space allocation information as shown below:

WORK 5

Then each of the multiple WORK files is allocated using the values from the WORKSPACE parameter of prefix.MICS.PARMS(JCLDEF), as described earlier under First Time Implementation of Multiple Work Files.

To increase workspace, you can increase the number of WORK files (for example, change WORK 5 to WORK 6,7,8,

or 9), increase the space allocation in the WORKSPACE parameter, or do both.

To decrease workspace, you can decrease the number of WORK files (for example, change WORK 5 to WORK 4,3,2, or 1), decrease the space allocation in the WORKSPACE parameter, or do both.

You can also elect to explicitly specify the multiple WORK file space allocation by adding the space allocation values directly to the WORK statement. This will remove the link to the prefix.MICS.PARMS(JCLDEF) WORKSPACE parameter for multiple WORK file space allocation. This is recommended as it serves to clearly document, in one place, how multiple WORK files are allocated.

- If the existing WORK statement does include space allocation as shown in the examples below:

```
WORK 5 TRK 200 100
```

or

```
WORK 5 SPACE=(TRK,(200,100)) STORCLAS=MICSTEMP
```

Simply change the values to meet your needs.

If you need more work space, you can increase the number of WORK files (for example, change WORK 5 to WORK 6,7,8, or 9), increase the space allocation (for example, change TRK 200 100 to TRK 250 120), or do both.

To decrease work space, you can decrease the number of WORK files (for example, change WORK 5 to WORK 4,3,2, or 1), decrease the space allocation (for example, change TRK 200 100 to TRK 150 80), or do both.

Note: If internal step restart is NOT active (RESTART NO) and you change the WORK parameter, you must:

- o Run cccPGEN
- o Run JCLGENU for DAILY (to regenerate DAILY) and, if incremental update is enabled, INCRccc

When internal step restart is active, (RESTART YES), then, when you change WORK and run cccPGEN, changes take effect immediately. There is no need to run JCLGENU.

SASWORK

This statement is optional.

The WORK DD statement in the CA MICS procedures allocates a temporary data set where SAS keeps its temporary data files and other items that SAS uses during processing of the current job.

By default, the space allocated is defined in the member prefix.MICS.PARMS(JCLDEF) with the WORKSPACE and WORKUNIT parameters, then generated into all the JCL procedures for a given unit.

With the SASWORK statement you have the option to override this unit-wide definition to specify the space allocation individually for the current step.

The format of the SASWORK statement is:

SASWORK data_set_allocation_parameters

where data_set_allocation_parameters is one or more data set allocation parameters (for example, STORCLAS or SPACE) separated by spaces.

You can also specify the SASWORK parameter as the following:

SASWORK XXX pppp ssss

where:

XXX is TRK or CYL
pppp is the primary allocation
ssss is the secondary allocation

If you omit the data_set_allocation_parameters or the SASWORK statement, the WORK data set is allocated according to the values you specified for the WORKUNIT and WORKSPACE parameters in prefix.MICS.PARMS(JCLDEF). Use the data_set_allocation_parameters to override this default, either to alter the space allocation or to use System Managed Storage (SMS) parameters to control data set placement and characteristics.

Specify data set allocation parameters, separated by blanks,

according to SAS LIBNAME statement syntax. If you need multiple lines, repeat the SASWORK keyword on the continuation line.

Example:

```
SASWORK STORCLAS=MICSTEMP SPACE=(XXX,(pppp,ssss))
```

where:

STORCLAS	- specifies a storage class for a new data set. The name can have up to 8 characters.
SPACE	- specifies how much disk space to provide for a new data set being allocated.
XXX	- is TRK or CYL.
pppp	- is the primary allocation.
ssss	- is the secondary allocation.

Note: If you change the SASWORK parameter, you must:

- o Run cccPGEN
- o Run JCLGENU for DAILY (to regenerate DAILY) and, if incremental update is enabled, INCRccc

```
MULTWORK|NOMULT fff fff ... fff  
-----
```

Since multiple work files usage impacts performance, this product provides these optional parameters so you can restrict multiple work files usage to only those files having excessive space requirements.

Note: You can only use one of these optional parameters with the WORK statement, NOT both.

The MULTWORK parameter restricts the use of multiple WORK files to ONLY those listed after the MULTWORK keyword.

```
MULTWORK fff fff ... fff
```

where fff is the unique three character identifier

If you need multiple lines, repeat the MULTWORK on the continuation line.

The NOMULT parameter forces the use of multiple WORK files for all files EXCEPT those specified after the NOMULT

keyword.

NOMULT fff fff ... fff

where fff is the unique three character identifier

If you need multiple lines, repeat the NOMULT on the continuation line.

The default is MULTWORK CSW CMR CSF if neither MULTWORK nor NOMULT parameters are specified.

The following files are eligible for multiple WORK support:

CSW	CICS Transaction Detail Work File
CMR	CICS MRO Activity File
CSF	CICS File/DBD Activity File

7.3.2.8 Internal Step RESTART Statements

RESTART YES/NO

This statement is optional. Specify this to activate internal step restart for this product's DAILY and/or INCRccc database update job steps:

RESTART YES

If you do not specify or enable the RESTART parameter, then this option defaults to the following and internal step restart is disabled:

RESTART NO

```
*****
*                                                                 *
* Note: Changing the RESTART parameter (either from NO        *
*       to YES or from YES to NO) requires regeneration      *
*       of the DAILY operational job by executing            *
*       prefix.MICS.CNTL(JCLGEND) or by specifying           *
*       DAILY in prefix.MICS.PARMS(JCLGENU) and              *
*       executing prefix.MICS.CNTL(JCLGENU).                  *
*                                                                 *
*       If incremental update is active for this product,    *
*       you must also regenerate the INCRccc job.            *
*                                                                 *
*****
```

Internal step restart can significantly reduce time and resource usage to recover from daily and/or incremental update processing failures. CA MICS uses a checkpoint/restart technique.

- o When internal step restart is activated, the database update job step "checkpoints" (or saves) intermediate results (work file contents) and the operational environment at the end of each processing phase.
- o Then, if required, the database update step can resume execution at the beginning of the processing phase in which the failure occurred.
- o Restart is accomplished by restoring the operational environment from the last checkpoint, bypassing completed

processing phases, and resuming execution using intermediate results (work files) from the last checkpoint.

Note: When you activate internal step restart (RESTART YES), the following optional restart parameters are enabled. These parameters have no effect if restart is disabled (RESTART NO). For more details, see the individual parameter descriptions later in this section.

- o RESTARTCKPT data_set_allocation_parameters
- o RESTARTWORK data_set_allocation_parameters
- o DYNAMWAIT minutes

Processing Phases:

This product employs three database update processing phases followed by the two common roll-up phases.

Phase	Description
-----	-----
FORMAT	Read raw input data, convert to SAS format, and output to intermediate work files.
CSYMERG	Sort intermediate work file contents, eliminate duplicate input data, and prepare for DETAIL cycle creation.
MSAC	Invoke CICMSAC routine if MSACCOUNT was specified in CICOPS.
DYSUM	Summarize DETAIL data to create new DAYS cycles and to update current week-to-date and month-to-date cycles.
DYAGE	Cutover new database cycles to production and "age" existing cycles.

RESTART Considerations

- o Overhead

Enabling internal step restart adds some overhead to the database update job step -- the cost of taking checkpoints and managing saved materials. Since this overhead is relatively constant and independent of input data volume, you may find that costs outweigh potential savings when input data volume is low, for example in a test unit. For high volume, production units, internal step restart support overhead should be a minor portion of total resource usage.

o Cataloged Work Files

When internal step restart is enabled, the SAS work data set, internal step restart control data set, and multiple work file data sets are allocated and cataloged with permanent dataset names so they will be retained for use in restart if the step abends. These data sets are deleted when the step completes successfully.

Prior to enabling internal step restart support, these data sets were probably allocated on system "scratch" space with a temporary, system assigned data set names. If your installation standards do not allow "permanent" data sets on DASD volumes used for temporary work space, you may need to use the WORK, RESTARTCKPT, and RESTARTWORK parameters to direct the internal step restart data sets to a generic unit or storage class that allows cataloged data sets.

o Dynamic Allocation

When internal step restart is active, dynamic allocation is employed for the work data sets. If your installation restricts dynamic allocation of large, cataloged data sets, you may need to use the WORK, RESTARTCKPT, and RESTARTWORK parameters to direct work data set allocation to a generic unit or storage class where dynamic allocation is allowed.

o Data Set Names

The SAS work data set, internal step restart control data set, and multiple work file data sets are allocated and cataloged according to the standard CA MICS unit database data set name conventions. The default DDNAME and data set names are:

- o SAS work data set,

```
//cccXWORK DD DSN=prefix.MICS.cccXWORK,.....
```

- o Internal step restart control data set,
//cccXCKPT DD DSN=prefix.MICS.cccXCKPT,.....
- o Multiple work file data sets,
//WORKnn DD DSN=prefix.MICS.cccWRKnn,.....

Since these data sets conform to the same data set name conventions as your existing CA MICS data sets, there should be few, if any, data set name related allocation issues. However, it is possible to override the data set names if required. Please contact CA MICS Product Support for assistance if you must alter data set names.

RESTARTCKPT

This statement is optional. Specify the following to override default data set allocation parameters for the internal step restart checkpoint data set:

```
RESTARTCKPT data_set_allocation_parameters
```

Note: RESTARTCKPT is ignored when you specify RESTART NO.

The internal step restart checkpoint data set (or cccXCKPT data set) contains processing status, control, and SAS environmental information for internal step restart processing checkpoints. This includes a copy of the SAS WORK format and macro catalogs, current macro variable values, and a description of work files that may be needed to restart DAYnnn processing.

By default, the cccXCKPT data set is allocated according to the values you specified for the WORKUNIT and WORKSPACE parameters in prefix.MICS.PARMS(JCLDEF). Specify RESTARTCKPT to override this default, either to alter the space allocation or to use System Managed Storage (SMS) parameters to control data set placement and characteristics.

Note: If you allocate insufficient space for the cccXCKPT data set, DAYnnn processing will fail and can only be restarted from the beginning.

Note: You can override the RESTARTCKPT data set allocation parameters at execution-time using the //PARMOVRD facility.

For more information about execution-time override of dynamic data set allocation parameters, see the PIOM, section 2.3.6.

Specify data set allocation parameters, separated by blanks, according to SAS LIBNAME statement syntax. If you need multiple lines, repeat the RESTARTCKPT keyword on the continuation line.

RESTARTCKPT accepts the engine/host options documented in the SAS Companion for the z/OS Environment, including STORCLAS, UNIT, SPACE, BLKSIZE, DATACLAS, MGMTCLAS, and VOLSER.

Important! DO NOT SPECIFY THE DISP PARAMETER.

Example 1:

```
RESTARTCKPT STORCLAS=MICSTEMP SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.
The name can have up to eight characters.

SPACE - specifies how much disk space to provide for a new data set being allocated, where:

xxxx is TRK, CYL, or blklen
pp is the primary allocation
ss is the secondary allocation

and ROUND specifies that the allocated space be "rounded" to a cylinder boundary when the unit specified was a block length. ROUND is ignored with the TRK or CYL options.

Example 2 (multiple lines):

```
RESTARTCKPT STORCLAS=MICSTEMP UNIT=SYSDA  
RESTARTCKPT SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.
The name can have up to eight characters.

UNIT - specifies the generic unit for a new data set.
The name can have up to eight characters.

SPACE - specifies how much disk space to provide for

a new data set being allocated.

RESTARTWORK

This statement is optional. Specify the following to override default data set allocation parameters for the internal step restart WORK data set:

RESTARTWORK data_set_allocation_parameters

Note: RESTARTWORK is ignored when you specify RESTART NO.

The internal step restart WORK data set (or cccXWORK data set) contains the intermediate work files that are not enabled to multiple work file support, including those files you may have specified on the optional NOMULT statement.

By default, the cccXWORK data set is allocated according to the values you specified for the WORKUNIT and WORKSPACE parameters in prefix.MICS.PARMS(JCLDEF). Specify RESTARTWORK to override this default, either to alter the space allocation or to use System Managed Storage (SMS) parameters to control data set placement and characteristics.

Note: If you allocate insufficient space for the cccXWORK data set, DAYnnn processing will fail and can only be restarted from the beginning.

Note: You can override the RESTARTWORK data set allocation parameters at execution-time using the //PARMOVRD facility. For more information about execution-time override of dynamic data set allocation parameters, see the PIOM, section 2.3.6.

Specify data set allocation parameters, separated by blanks, according to SAS LIBNAME statement syntax. If you need multiple lines, repeat the RESTARTWORK keyword on the continuation line.

RESTARTWORK accepts the engine/host options documented in "SAS Companion for the z/OS Environment", including STORCLAS, UNIT, SPACE, BLKSIZE, DATACLAS, MGMTCLAS, and VOLSER.

Important! DO NOT SPECIFY THE DISP PARAMETER.

Example 1:

```
RESTARTWORK STORCLAS=MICSTEMP SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.
The name can have up to eight characters.

SPACE - specifies how much disk space to provide for
a new data set being allocated, where:

xxxx is TRK, CYL, or blklen

pp is the primary allocation

ss is the secondary allocation

and ROUND specifies that the allocated space be
"rounded" to a cylinder boundary when the unit
specified was a block length. ROUND is ignored
with the TRK or CYL options.

Example 2 (multiple lines):

```
RESTARTWORK STORCLAS=MICSTEMP UNIT=SYSDA
```

```
RESTARTWORK SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.
The name can have up to eight characters.

UNIT - specifies the generic unit for a new data set.
The name can have up to 8 characters.

SPACE - specifies how much disk space to provide for
a new data set being allocated.

7.3.2.9 Enable Internal Step Restart

To enable the internal step restart in the CA MICS Analyzer Option for CICS, follow the checklist provided below:

```
*****
*                                     *
*          ENABLE INTERNAL STEP RESTART          *
*                                     *
*****
```

- ___ 1. Edit prefix.MICS.PARMS(cccOPS), where (ccc) is the component identifier, and specify:

RESTART YES

For additional information on related topic, review the documentation for this product on WORK, RESTARTWORK, and RESTARTCKPT parameters to override default data set allocation parameters.

- ___ 2. Submit the job in prefix.MICS.CNTL(cccPGEN).
- ___ 3. Edit prefix.MICS.PARMS(JCLGENU) so that it contains a single line that reads:

DAILY

or, if incremental update is enabled for this product in this unit database, specify:

DAILY INCRccc

where ccc is the product ID.

- ___ 4. Submit the job in prefix.MICS.CNTL(JCLGENU). Ensure that there are no error messages in MICSLOG or SYSTSPRT, that the MICSLOG contains the normal termination message, BAS10999I, and that the job completes with a condition code of zero.

- ___ 5. The following operational job(s) have changed:

DAILY

INCRccc (if incremental update is enabled)

If your site has implemented the operational CA MICS processes in a scheduling product, the JCL may have

to be refreshed in that product. See the scheduling product's administrator for the exact processes involved in updating that product's representation of the CA MICS jobs.

7.3.2.10 Incremental Update Statements

INCRUPDATE

This statement is optional. Specify this to enable incremental update for this product:

INCRUPDATE YES

If you do not specify or enable the INCRUPDATE parameter, then this option defaults to this and incremental update is disabled:

INCRUPDATE NO

```
*****
*
* Note: Changing the INCRUPDATE parameter (either from NO
*       to YES or from YES to NO) requires regeneration
*       of the DAILY operational job by executing
*       prefix.MICS.CNTL(JCLGEN) or by specifying
*       DAILY in prefix.MICS.PARMS(JCLGEN) and
*       executing prefix.MICS.CNTL(JCLGEN).
*
*       If you specify INCRUPDATE YES, you must also
*       generate the INCRccc, cccIUALC, and cccIUGDG jobs
*       (where ccc is the 3 character product ID).
*       Depending on the options you select, you may also
*       need to execute the cccIUALC and/or cccIUGDG
*       jobs.
*
*****
```

Incremental update can significantly reduce time and resource usage in the DAILY job by letting you split out a major portion of daily database update processing into multiple, smaller, incremental updates executed throughout the day.

- o Standard CA MICS database update processing involves (1) reading and processing raw input data to generate DETAIL and DAYS level CA MICS database files, followed by (2) summarization of DETAIL/DAYS level data to update week-to-date and month-to-date database files.
- o When you activate incremental update:
 - You can execute the first-stage processing (raw data input to create DETAIL/DAYS files) multiple times

throughout the day, each time processing a subset of the total day's input data.

- Then, during the final update of the day (in the DAILY job), the incremental DETAIL/DAYS files are "rolled-up" to the database DETAIL and DAYS timespans, and then summarized to update the week-to-date and month-to-date files.
- o Incremental update is independent of your internal step restart or DBSPLIT specifications. You have the option to perform incremental updates with or without internal step restart support.
- o Incremental update is activated and operates independently by product. The incremental update job for this product, INCRccc (where ccc is the product ID), can execute concurrently with the incremental update job for another product in the same unit database.
- o The CA MICS database remains available for reporting and analysis during INCRccc job execution.

```
*****
*
* Note: CA MICS is a highly configurable system
* supporting up to 36 unit databases, each of which
* can be configured and updated independently.
* Incremental update is just one of the options you
* can use to configure your CA MICS complex.
*
* All efforts should be made to employ CA MICS
* configuration capabilities to minimize issues
* prior to activating incremental update. For
* example:
*
* o Splitting work to multiple units is an
* effective way to enable parallel database
* update processing
*
* o Adjusting account code definitions to ensure
* adequate data granularity while minimizing
* total database space and processing time
*
* o Tailoring the database to drop measurements
* and metrics of lesser value to your
* data center, thereby reducing database update
* processing and resource consumption
*
```

```

*           While incremental update is intended to reduce      *
*           DAILY job elapsed time, total resource usage of    *
*           the combined INCRccc and DAILY jobs steps can      *
*           increase due to the additional processing           *
*           required to maintain the incremental update         *
*           "to-date" files and for roll-up to the unit        *
*           database. The increased total resource usage        *
*           will be more noticeable with small data volumes,    *
*           where processing code compile time is a greater     *
*           percentage of total processing cost.                *
*
*****

```

Note: When you activate incremental update (INCRUPDATE YES), the following optional incremental update parameters are enabled. These parameters have no effect if incremental update is disabled (INCRUPDATE NO). For more details, see the individual parameter descriptions later in this section.

- o INCRDB PERM/TAPE/DYNAM
- o INCRDETAIL data_set_allocation_parameters
- o INCRDAYS data_set_allocation_parameters
- o INCRCKPT data_set_allocation_parameters
- o INCRSPLIT USE/IGNORE data_set_allocation_parameters

Incremental update processing reads and processes raw measurement data to create and maintain DETAIL and DAYS level "to-date" files for the current day.

- o These incremental update database files are maintained on unique z/OS data sets, independent of the standard CA MICS database files, and independent of any other product's incremental update database files. There is one data set each for DETAIL and DAYS level "to-date" data and a single incremental update checkpoint data set for this product in this unit.
- o The incremental update DETAIL and DAYS files can be permanent DASD data sets, or they can be allocated dynamically as needed and deleted after DAILY job processing completes. Optionally, you can keep the incremental update DETAIL and DAYS files on tape, with the data being loaded onto temporary DASD space as needed for incremental update or DAILY job processing. See the INCRDB PERM/TAPE/DYNAM option for more

information.

After activating incremental update, you will use three incremental update facility jobs found in prefix.MICS.CNTL (Note that ccc is the product ID):

- o cccIUALC

You execute this job to allocate and initialize the incremental update checkpoint file, and optionally the incremental update DETAIL and DAYS database files. cccIUALC is generally executed just ONE time.

- o cccIUGDG

You execute this job to add generation data group (GDG) index definitions to your system catalog in support of the INCRDB TAPE option. cccIUGDG is generally executed just ONE time.

- o INCRccc

This is the job you execute for each incremental update. You will integrate this job into your database update procedures for execution one or more times per day to process portions of the total day's measurement data.

Note: The DAILY job is run once at the end of the day. It will perform the final incremental update for the day's data, and then roll-up the incremental DETAIL/DAYS files to the database DETAIL and DAYS timespans and update the week-to-date and month-to-date files.

INCRUPDATE Considerations

- o Overhead

Incremental update is intended to reduce DAILY job resource consumption and elapsed time by offloading a major portion of database update processing to one or more executions of the INCRccc job. In meeting this objective, incremental update adds processing in the INCRccc and DAILY jobs to accumulate data from each incremental update execution into the composite "to-date" DETAIL and DAYS incremental update files, and also adds processing in the DAILY job to copy the incremental update files to the unit database DETAIL and DAYS

timespans. The amount of this overhead and the savings in the DAILY job are site-dependent, and will vary based on input data volume and on the number of times INCRccc is executed each day.

In addition, activating incremental update will cause additional compile-based CPU time to be consumed in the DAYnnn DAILY job step. The increase in compile time is due to additional code included for each file structure in support of the feature. This increase should be static based on the scope of the CA MICS data integration product in terms of files. This compile-time increase does not imply an increase in elapsed or execution time. Incremental update allows I/O bound, intensive processing (raw data inputting, initial CA MICS transformation, etc.) to be distributed outside of the DAILY job. I/O processing is the largest contributor to elapsed time in large volume applications. Thus, the expected overall impact is a decrease in the actual runtime of the DAYnnn job step.

- o Increased "Prime Time" Workload

By offloading work from the DAILY job to one or more INCRccc executions throughout the day, you are potentially moving system workload and DASD work space usage from the "off-hours," (when the DAILY job is normally executed) to periods of the day where your system resources are in highest demand. You should schedule INCRccc executions carefully to avoid adverse impact to batch or online workloads. For example, if your site's "prime shift" is 8:00 AM to 5:00 PM, you might choose to schedule incremental updates for 7:00 AM (just before "prime shift") and 6:00 PM (just after "prime shift"), with the DAILY job executing just after midnight.

- o Increased DASD Usage

The DASD space required for the incremental update DETAIL and DAYS database files is in addition to the DASD space already reserved for the CA MICS database. By default, the incremental update database files are permanently allocated, making this DASD space unavailable for other applications. In general, you can assume that the incremental update database files will require space equivalent to two cycles of this product's DETAIL and DAYS timespan files.

Alternatively, the incremental update database files can

be allocated in the first incremental update of the day and deleted by the DAILY job (see the INCRDB DYNAM option later in this section). This approach reduces the amount of time that the DASD space is dedicated to incremental update, and lets the amount of DASD space consumed increase through the day as you execute each incremental update.

A third option is to store the incremental update database files on tape (see the INCRDB TAPE option). With this approach, the DASD space is required just for the time that each incremental update or DAILY job step is executing. Note that while this alternative reduces the "permanent" DASD space requirement, the total amount of DASD space required while the incremental update or DAILY jobs are executing is unchanged. In addition, the TAPE option adds processing to copy the incremental update files to tape, and to reload the files from tape to disk.

Note: The incremental update checkpoint file is always a permanently allocated disk data set. This is a small data set and should not be an issue.

- o Operational Complexity

Incremental update expands your measurement data management and job scheduling issues. You must ensure that each incremental update and the DAILY job processes your measurement data chronologically; that is, each job must see data that is newer than the data processed by the prior job. By incrementally updating the database, you have more opportunities to miss a log file, or to process a log out of order.

- o Interval End Effects

Each incremental update processes a subset of the day's measurement data, taking advantage of early availability of some of the day's data, for example, when a measurement log fills and switches to a new volume. This can cause a problem if the measurement log split occurs while the data source is logging records for the end of a measurement interval, thus splitting the data for a single measurement interval across two log files. When an incremental update processes the first log file, the checkpoint high end timestamp is set to indicate that this split measurement interval has been processed. Then, when the rest of the measurement interval's data is

encountered in a later update, it can be dropped as duplicate data (because data for this measurement interval end timestamp has already been processed).

Appropriate scheduling of log dumps and incremental updates can avoid this problem. For example, if you plan to run incremental updates at 7:00 AM and 6:00 PM, you could force a log dump in the middle of the measurement interval just prior to the scheduled incremental update executions. This is an extension of the procedure you may already be using for end-of-day measurement log processing. The objective is to ensure that all records for each monitor interval are processed in the same incremental update.

- o Dynamic Allocation

When you activate incremental update and specify TAPE or DYNAM for the INCRDB parameter, dynamic allocation is employed for the incremental update database files. If your site restricts dynamic allocation of large, cataloged data sets, you must use the INCRDETAIL and INCRDAYS parameters to direct incremental update data set allocation to a generic unit or storage class where dynamic allocation is allowed.

- o Data Set Names

The incremental update database files are allocated and cataloged according to standard CA MICS unit database data set name conventions. The DDNAME and default data set names are (where ccc is the product ID):

- o Incremental update checkpoint file,
//IUCKPT DD DSN=prefix.MICS.ccc.IUCKPT,.....
- o Incremental update DETAIL
//IUDETAIL DD DSN=prefix.MICS.ccc.IUDETAIL,.....
- o Incremental update DAYS
//IUDAYS DD DSN=prefix.MICS.ccc.IUDAYS,....

Since these data sets conform to the same data set name conventions as your existing CA MICS data sets, there should be few, if any, data-set-name-related allocation issues. However, it is possible to override the data set names if required. Contact Technical Support at <http://ca.com/support> for assistance if you must change data set names.

Note: If your data center uses the TAPEfff option or USRXfff exits, be sure to review the important considerations in Section 10.1.4.1 of this guide.

INCRDB

This statement is optional. The default is this:

INCRDB PERM

Note: INCRDB is ignored when you specify INCRUPDATE NO.

Specify this statement or take the default, to keep the incremental update database DETAIL and DAYS files on permanently allocated DASD data sets:

INCRDB PERM

Execute the prefix.MICS.CNTL(cccIUALC) job to allocate the incremental update database files.

```
*****
*
* Note: The incremental update checkpoint file is always *
*       a permanently allocated DASD data set.           *
*
*****
```

Specify this to offload the incremental update DETAIL and DAYS files to tape between incremental update executions:

INCRDB TAPE #gdgs UNIT=name

With the TAPE option, the incremental update DETAIL and DAYS DASD data sets are dynamically allocated at the beginning of the incremental update job or DAILY job step, and then are deleted after the job step completes.

- o The first incremental update job of the day allocates and initializes the incremental update database files. At the end of the job, the DETAIL and DAYS files are copied to a new (+1) generation of the incremental update tape data sets. Then the DASD files are deleted.
- o Subsequent incremental update jobs restore the DASD

incremental update database files from the current, (0) generation, incremental update tape data sets before processing the input measurement data. At the end of the job, the DETAIL and DAYS files are copied to a new (+1) generation of the incremental update tape data sets. Then the DASD files are deleted.

- o The DAILY job step also restores the DASD incremental update database files from the (0) generation tape files before processing the input data, but does NOT copy the incremental update database files to tape. Thus, the DAILY job actually creates a new, null (+1) generation.
- o Use the #gdgs parameter to specify the maximum number of incremental update tape generations. The minimum is 2 and the maximum is 99, with a default of 5. Set the number of generations equal to or greater than the number of incremental updates, including the DAILY job you plan to execute each day. This facilitates restart and recovery if you encounter problems requiring you to reprocess portions of the daily measurement data.
- o Use the optional UNIT=name parameter to specify a tape unit name for the incremental update database output tapes. The default is to use the same tape unit as the input tapes.
- o A special index must be created in your system catalog for each of the incremental update tape data set generation data groups. The prefix.MICS.CNTL(cccIUGDG) job will generate the statements to create the incremental update GDG index definitions.
 - Before each index is built, it is deleted. These DLTX (or DELETE) statements causes an error message if no entry exists. This is done so that you can change the number of entries without having to delete each of the index entries.
 - DLTX and BLDG (or DELETE and DEFINE) fail if there is a cataloged data set with the same index. IDCAMS (or IEHPROGM) issues a message and gives a return code of 8. This issue is not a problem for non-GDG entries or if the GDG already has the desired number of entries.
 - If you want to change the number of entries kept in a GDG with cataloged data sets, do the following:

1. Uncatalog any existing entries in the GDG.
 2. Delete the index with a DLTX (or DELETE).
 3. Create the index with a BLDG (or DEFINE).
 4. Catalog any entries that are uncataloged in step 1.
- o The incremental update tape data set names are as follows, where ccc is the product ID:
- Incremental update tape DETAIL file
tapeprefix.MICS.ccc.IUXTAPE.GnnnnV00
 - Incremental update tape DAYS file
tapeprefix.MICS.ccc.IUDTAPE.GnnnnV00

```
*****
*
* Note: The INCRDETAIL and INCRDAYS parameters are
*       required when you specify INCRDB TAPE.
*
*****
```

Specify this parameter to allocate dynamically the incremental update DETAIL and DAYS DASD data sets in the first incremental update of the day, and then delete these data sets at the end of the DAILY job step:

INCRDB DYNAM

- o With this option, no space is used for the incremental update database files during the time between the end of the DAILY job step and the beginning of the next day's first incremental update.
- o With this approach, you can set the data set allocation parameters so that the incremental update DETAIL and DAYS data sets start out with a minimum allocation and then grow through secondary allocations as more space is required for subsequent incremental updates. For example, enough space for one incremental update.

```
*****
*
* Note: The INCRDETAIL and INCRDAYS parameters are
*       required when you specify INCRDB DYNAM.
*
*****
```

INCRDETAIL

This statement is required if you specify either of these:

INCRDB TAPE**INCRDB DYNAM**

Otherwise, this statement is optional. There is no default.

Specify this to define data set allocation parameters for the incremental update DETAIL data set (IUDETAIL):

INCRDETAIL data_set_allocation_parameters

Note: INCRDETAIL is ignored when you specify INCRUPDATE NO.

The incremental update DETAIL data set (IUDETAIL) contains the current incremental update detail-level database files, and the DETAIL "to-date" data for the current daily update cycle. You should allocate DASD space equivalent to two cycles of this product's DETAIL timespan data.

If you specified INCRDB PERM (the default), your INCRDETAIL parameter specifications are used in generating the cccIUALC job (where ccc is the product ID).

- o You will execute the cccIUALC job to allocate and initialize the incremental update database and checkpoint files.
- o Omit the INCRDETAIL parameter if you prefer to specify data set allocation parameters directly in the generated prefix.MICS.CNTL(cccIUALC) job.

If you specified INCRDB TAPE or INCRDB DYNAM, your INCRDETAIL parameter specifications are used in incremental update DETAIL data set dynamic allocation during incremental update or DAILY job step execution.

- o The INCRDETAIL parameter is required for the TAPE or DYNAM option.
- o Specify data set allocation parameters, separated by

blanks, according to SAS LIBNAME statement syntax. If you need multiple lines, repeat the INCRDETAIL keyword on the continuation line.

- o INCRDETAIL accepts the engine/host options documented in the SAS Companion for the z/OS Environment, including STORCLAS, UNIT, SPACE, BLKSIZE, DATACLAS, MGMTCLAS, and VOLSER.

Important! DO NOT SPECIFY THE DISP PARAMETER.

- o You can override the INCRDETAIL data set allocation parameters at execution-time using the //PARMOVRD facility. For more information about execution-time override of dynamic data set allocation parameters, see the PIOM, section 2.3.6.

Example 1:

```
INCRDETAIL STORCLAS=MICSTEMP SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.
The name can have up to eight characters.

SPACE - specifies how much disk space to provide for a new data set being allocated, where:

xxxx is TRK, CYL, or blklen
pp is the primary allocation
ss is the secondary allocation

and ROUND specifies that the allocated space be "rounded" to a cylinder boundary when the unit specified was a block length. ROUND is ignored with the TRK or CYL options.

Example 2 (multiple lines):

```
INCRDETAIL STORCLAS=MICSTEMP UNIT=SYSDA  
INCRDETAIL SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.
The name can have up to eight characters.

UNIT - specifies the generic unit for a new data set.

The name can have up to eight characters.

SPACE - specifies how much disk space to provide for a new data set being allocated.

INCRDAYS

This statement is required if you specify either of these:

INCRDB TAPE

INCRDB DYNAM

Otherwise, this statement is optional. There is no default.

Specify this to define data set allocation parameters for the incremental update DAYS data set (IUDAYS):

INCRDAYS data_set_allocation_parameters

Note: INCRDAYS is ignored when you specify INCRUPDATE NO.

The incremental update DAYS data set (IUDAYS) contains the current incremental update days-level database files, and the DAYS "to-date" data for the current daily update cycle. You should allocate DASD space equivalent to two cycles of this product's DAYS timespan data.

If you specified INCRDB PERM (the default), your INCRDAYS parameter specifications are used in generating the cccIUALC job (where ccc is the product ID).

- o You will execute the cccIUALC job to allocate and initialize the incremental update database and checkpoint files.
- o Omit the INCRDAYS parameter if you prefer to specify data set allocation parameters directly in the generated prefix.MICS.CNTL(cccIUALC) job.

If you specified INCRDB TAPE or INCRDB DYNAM, your INCRDAYS parameter specifications are used in incremental update DAYS data set dynamic allocation during incremental update or DAILY job step execution.

- o The INCRDAYS parameter is required for the TAPE or DYNAM

option.

- o Specify data set allocation parameters, separated by blanks, according to SAS LIBNAME statement syntax. If you need multiple lines, repeat the INCRDAYS keyword on the continuation line.
- o INCRDAYS accepts the engine/host options documented in the SAS Companion for the z/OS Environment, including STORCLAS, UNIT, SPACE, BLKSIZE, DATACLAS, MGMTCLAS, and VOLSER.

Important! DO NOT SPECIFY THE DISP PARAMETER.

- o You can override the INCRDAYS data set allocation parameters at execution-time using the //PARMOVRD facility. For more information about execution-time override of dynamic data set allocation parameters, see the PIOM, Section 2.3.6.

Example 1:

```
INCRDAYS      STORCLAS=MICSTEMP SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.
The name can have up to eight characters.

SPACE - specifies how much disk space to provide for a new data set being allocated, where:

xxxx is TRK, CYL, or blklen
pp is the primary allocation
ss is the secondary allocation

and ROUND specifies that the allocated space be "rounded" to a cylinder boundary when the unit specified was a block length. ROUND is ignored with the TRK or CYL options.

Example 2 (multiple lines):

```
INCRDAYS      STORCLAS=MICSTEMP UNIT=SYSDA  
INCRDAYS      SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.

The name can have up to eight characters.

UNIT - specifies the generic unit for a new data set.
The name can have up to eight characters.

SPACE - specifies how much disk space to provide for
a new data set being allocated.

INCRCKPT

This statement is optional. Specify this to override default data set allocation parameters for the incremental update checkpoint data set:

INCRCKPT data_set_allocation_parameters

Note: INCRCKPT is ignored when you specify INCRUPDATE NO.

The incremental update checkpoint data set tracks incremental update job status and the data that has been processed during the current daily update cycle. The incremental update checkpoint is used to detect and block the input of duplicate data during incremental update processing. This data set will be exactly the same size as prefix.MICS.CHECKPT.DATA (the unit checkpoint data set), usually 20K to 200K depending on the prefix.MICS.PARMS(SITE) CKPTCNT parameter (100-9999).

Your INCRCKPT parameter specifications are used in generating the cccIUALC job (where ccc is the product ID).

- o You will execute the cccIUALC job to allocate and initialize the incremental update checkpoint file. If you specified INCRDB PERM, then the cccIUALC job will also allocate the incremental update DETAIL and DAYS database files.
- o By default the incremental update checkpoint data set is allocated as SPACE=(TRK,(5,2)) using the value you specified for the prefix.MICS.PARMS(JCLDEF) DASDUNIT parameter.
- o Omit the INCRCKPT parameter if you prefer to override data set allocation parameters directly in the generated prefix.MICS.CNTL(cccIUALC) job.

Specify data set allocation parameters, separated by blanks, according to SAS LIBNAME statement syntax. If you need multiple lines, repeat the INCRCKPT keyword on the continuation line.

INCRCKPT accepts the engine/host options documented in the SAS Companion for the MVS Environment, including STORCLAS, UNIT, SPACE, BLKSIZE, DATACLAS, MGMTCLAS, and VOLSER.

Important! DO NOT SPECIFY THE DISP PARAMETER.

Example 1:

```
INCRCKPT    STORCLAS=MICSTEMP SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.
The name can have up to eight characters.

SPACE - specifies how much disk space to provide for
a new data set being allocated, where:

xxxx is TRK, CYL, or blklen

pp is the primary allocation

ss is the secondary allocation

and ROUND specifies that the allocated space be
"rounded" to a cylinder boundary when the unit
specified was a block length. ROUND is ignored
with the TRK or CYL options.

Example 2 (multiple lines):

```
INCRCKPT    STORCLAS=MICSTEMP UNIT=SYSDA  
INCRCKPT    SPACE=(xxxx,(pp,ss),,,ROUND)
```

where:

STORCLAS - specifies a storage class for a new data set.
The name can have up to eight characters.

UNIT - specifies the generic unit for a new data set.
The name can have up to eight characters.

SPACE - specifies how much disk space to provide for a new data set being allocated.

DYNAMWAIT

This statement is optional. Specify the following:

DYNAMWAIT minutes

to override the default amount of time, in minutes, the DAILY and/or INCRccc job will wait for an unavailable data set.

Note: This optional parameter is not normally specified.
The system default is adequate for most data centers.

Internal Step Restart and Incremental Update facilities use z/OS dynamic allocation services to create new data sets and to access existing data sets. Data set naming conventions and internal program structure are designed to minimize data set contention. However, if data set allocation does fail because another batch job or online user is already using a data set, DAILY and/or INCRccc processing will wait 15 seconds and then try the allocation again. By default, the allocation will be attempted every 15 seconds for up to 15 minutes. After 15 minutes, the DAILY or INCRccc job will abort.

If data set contention in your data center does cause frequent DAILY or INCRccc job failures, and you are unable to resolve the contention through scheduling changes, you may want to use the DYNAMWAIT parameter to increase the maximum number of minutes the DAILY and/or INCRccc jobs will wait for the data set to become available.

On the other hand, if your data center standards require that the DAILY and/or INCRccc jobs fail immediately if required data sets are unavailable, specify the following:

DYNAMWAIT 0

Note: You can override the DYNAMWAIT parameter at execution-time using the //PARMOVRD facility. For more information about execution-time override of dynamic data set allocation parameters, see the PIOM, section 2.3.6.

7.3.2.11 Implement Incremental Update

To implement incremental update in the CA MICS Analyzer Option for CICS, follow the checklist provided below.

```
*****
*
* Note: If you are using USRXfff exits, be sure to
*       review the important considerations in Section
*       10.1.4.1 of this guide before activating
*       incremental update.
*
*****
```

```
*****
*
*       IMPLEMENT INCREMENTAL UPDATE
*
*****
```

- ___ 1. Edit prefix.MICS.PARMS(cccOPS), where (ccc) is the component identifier:
 - o Specify the following:


```
INCRUPDATE YES
```
 - o If you want to store the incremental update database files on tape between incremental updates, specify this:


```
INCRDB TAPE #gdgs
```
 - o If you want to allocate the incremental update database files during the first incremental update of the day and delete these data sets at the end of the DAILY job step, specify this:


```
INCRDB DYNAM
```
 - o If you specified INCRDB TAPE or INCRDB DYNAM, then you must also specify this:


```
INCRDETAIL data_set_allocation_parameters
INCRDAYS   data_set_allocation_parameters
```
 - o If you want the incremental update job for this product to get input measurement data from the output of the SPLITSMF job, specify this:

INCRSPLIT USE data_set_allocation_parameters

- o For additional information on related topic, review the documentation for this product on INCRCKPT, INCRDETAIL, INCRDAYS, or INCRSPLIT parameters to override default data set allocation parameters.

___ 2. Submit the job in prefix.MICS.CNTL(cccPGEN).

___ 3. Edit prefix.MICS.PARMS(JCLGENU) so that it contains two or more lines reading:

```
DAILY
INCRccc cccIUALC cccIUGDG
```

___ 4. Submit the job in prefix.MICS.CNTL(JCLGENU). Ensure that there are no error messages in MICSLOG or SYSTSPRT, that the MICSLOG contains the normal termination message, BAS10999I, and that the job completes with a condition code of zero.

___ 5. Edit the job in prefix.MICS.CNTL(cccIUALC).

- o Inspect and/or specify data set allocation parameters for the incremental update database and checkpoint files. If you specified INCRDB TAPE or INCRDB DYNAM, the cccIUALC job will only allocate the incremental update checkpoint data set.
- o Submit the job. Ensure that there are no error messages in MICSLOG or SASLOG, and that the job completes with a condition code of zero.

___ 6. If you specified INCRDB TAPE, submit the job in prefix.MICS.CNTL(cccIUGDG) to define generation group indexes for the incremental update DETAIL and DAYS tape data sets. Examine SASLOG, MICSLOG, and SYSPRINT to verify that the generation group indexes were correctly defined.

Note: You may see error messages for the DLTX (or DELETE) statements. This is not a problem. cccIUGDG deletes each index prior to defining it, and an error message is issued if the index does not yet exist (e.g., if this is the first time you ran the cccIUGDG job).

- ___ 7. The following operational job(s) have changed:

DAILY INCRccc

If your site has implemented the operational CA MICS processes in a scheduling product, the JCL may have to be refreshed in that product. See the scheduling product's administrator for the exact processes involved in updating that product's representation of the CA MICS jobs.

- ___ 8. Implement operational procedures for gathering input measurement data and executing incremental updates (INCRccc) during the day.

You may also need to modify operational procedures for the DAILY job to ensure that processing is limited to input measurement data that has not been input to one of the day's incremental update executions.

7.3.2.12 DETAIL Tape Overview

For certain high volume files, the CA MICS architecture provides the ability to create optional DETAIL timespan data sets. These optional data sets are separate from the normal CA MICS database and have the following characteristics:

- o They are created during the component step execution: during either an incremental update or the DAILY update run. Each DETAIL tape file has a separate DD statement in the DAILY or incremental update job that specifies the data set name and other data set characteristics.
- o The data set name (DSN) for each DETAIL tape in a component step is:

tapeprefix.tmics.DETAIL.iiiiff(+1)

where:

- tapeprefix is the tape prefix value specified with the TAPEPREFIX statement in prefix.MICS.PARMS(JCLDEF)
- tmics is either null when NOMICSLEVEL is specified, or MICS if MICSLEVEL is specified with the TAPEPREFIX statement in prefix.MICS.PARMS(JCLDEF)
- iii is the three character CA MICS information area associated with the file
- fff is the three character CA MICS file identifier

Example: For the CA MICS DB2 Analyzer, if DETAIL tape processing were activated for the DB2DSU file, the following prefix.MICS.PARMS(JCLDEF) statement:

TAPEPREFIX CPLXA.U1T NOMICSLEVEL

yields the following DETAIL tape DSN:

CPLXA.U1T.DETAIL.DB2DSU(+1)

- o There is no cycle aging associated with these data sets. Instead, each DETAIL tape data set created is an 01 cycle (for example, DB2DSU01), and multiple instances are managed using Generation Data Groups (GDGs).

- o The creation of DETAIL tape data sets is independent of any sharedprefix.MICS.GENLIB customization choices. You can have one or more cycles of a file written to and managed in the unit DETAIL timespan database, and also use the DETAIL tape feature to create independent data sets for that same file.
- o The decision to create DETAIL tape data sets is made at the unit level. You can choose to create DETAIL tape data sets for a component file in unit A, but not in unit B.
- o While the expectation is that these files will be written to tape, you can make JCL choices to target other media such as DASD. Note that the SAS sequential tape engine will be used to write these files for both DASD and tape data sets.
- o Unique output exits are provided to permit manipulation and subsetting of the records written to the DETAIL tape data sets. This manipulation and subsetting, however, has no impact on files written to the standard CA MICS database.

The activation of DETAIL tape data sets is accomplished by specifying a TAPEfff statement in prefix.MICS.PARMS(cccOPS).

Customization of the JCL used to create each data set is accomplished using a cascading hierarchy of parameters beginning with choices made with the TAPEPARM statement in prefix.PARMS(JCLDEF).

The following sections provide the information required to activate DETAIL tape processing:

- 1 - Component Files Supporting DETAIL Tape Activation
- 2 - DETAIL Tape JCL Customization Overview
- 3 - The TAPEfff Statement
- 4 - The DETAILTAPEPARM Statement
- 5 - DETAILTAPESMSPARM and TAPEfffsMS Statements
- 6 - DETAIL Tape User Exits
- 7 - Activating DETAIL Tape Checklist
- 8 - Deactivating DETAIL Tape Checklist

7.3.2.12.1 CICS DETAIL Tape Supported Files

The CA MICS CICS Analyzer supports DETAIL tape processing for the following files:

- o CICCSU - CICS User Activity file

7.3.2.12.2 Overview of JCL Customization Parameters

The JCL for DETAIL tape data sets is constructed using tape-related parameters that are specified in `prefix.MICS.PARMS(JCLDEF)` and `prefix.MICS.PARMS(cccOPS)`.

Each DETAIL tape data set has a unique DD in the component step. The JCL for each DETAIL tape data set is customized according to user-specified JCL parameters such as the number of generation data group (GDG) entries that are maintained, unit names, volume count, retention, and expiration dates, and System Managed Storage (SMS) keywords. For example, `STORCLAS=storclas`.

CA MICS provides a hierarchical means to specify tape-related JCL parameters suitable for each data center.

The hierarchy, as it applies to DETAIL tape specification, is described:

- o Unit Level - `TAPEPARM` and `TAPESMSPARM`

Used for all tape data sets created by all components in the unit. Specified in `prefix.MICS.PARMS(JCLDEF)`

- o Unit DETAIL Tape Level - `DETAILTAPEPARM` and `DETAILTAPESMSPARM`

Used for all DETAIL tape data sets created by all components in the unit. Overrides choices that were made in `TAPEPARM` and `TAPESMSPARM`. Specified in `prefix.MICS.PARMS(JCLDEF)`.

- o Component Level - `DETAILTAPEPARM` and `DETAILTAPESMSPARM`

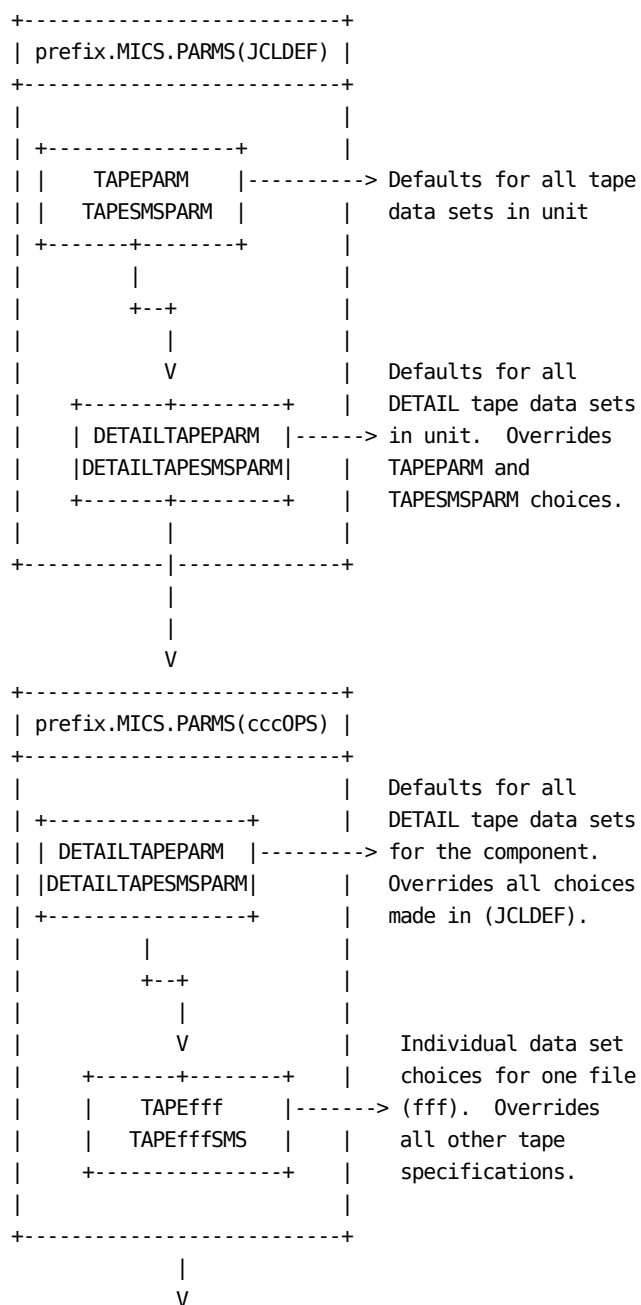
Used for all DETAIL tape data sets created by a particular component. Overrides all choices that were made in `JCLDEF`. Specified in `prefix.MICS.PARMS(cccOPS)`, where `ccc` represents the component.

- o File Level - `TAPEfff` and `TAPEfffsMS`

Used for a particular DETAIL tape data set. The `fff` identifies the specific DETAIL file. Overrides all choices made in `JCLDEF` and choices that were made with `DETAILTAPEPARM` and `DETAILTAPESMSPARM` in `prefix.MICS.PARMS(cccOPS)`. Specified in `prefix.MICS.PARMS(cccOPS)`, where `ccc` represents the component.

The JCL parameter specification hierarchy that is used for DETAIL tape customization is shown graphically in the following diagram:

DETAIL Tape JCL Parameter Hierarchy



```
// DD statements and GDG generation
```

7.3.2.12.3 The TAPEfff Statement

The TAPEfff statement is used to:

- o Activate DETAIL tape data set creation for the file indicated by fff. The component files eligible for DETAIL tape processing are listed in a previous section.
- o Override, for the indicated file, any JCL-related tape specifications made in prefix.MICS.PARMS(JCLDEF) and prefix.MICS.PARMS(cccOPS) with TAPEPARM or DETAILTAPEPARM statements.

The syntax is as follows:

TAPEfff <optional parameters>

where fff is the unique 3-character file identifier

The optional parameters include both keyword parameters and keyword=value specifications. You can include multiple TAPEfff statements if required in order to specify the desired optional parameters. If you specify a parameter multiple times, the last instance defines the value in effect.

You can specify as many parameters as you want on each statement, but each TAPEfff statement must end by column 72. You may also choose to specify only one parameter per TAPEfff statement. Either approach is acceptable.

The optional parameters available for the TAPEfff statement are listed below:

```
TAPEfff GDG=nnn UNIT=unitname VOLCOUNT=n
TAPEfff EXPDT=expdt / RETPD=nnn / NOEXPDT
TAPEfff COMP/NOCOMP
TAPEfff STACK/NOSTACK AFF/NOAFF
TAPEfff MODEL='modeldsn' / MODEL=NOMODEL
TAPEfff DISK / TAPE SPACE=spacedef
TAPEfff VOLSER=volser
```

The following is a sample TAPEfff statement:

```
TAPEfff GDG=3 VOLCOUNT=7
```

In this sample, only the number of generation data groups (GDGs) and VOLCOUNT parameters are explicitly specified for the file fff. All other JCL related parameters for the data

set will be set according to specifications made with TAPEPARM and DETAILTAPEPARM statements according to the hierarchy described in a previous section.

The optional parameters for the TAPEfff statement are identical to those supported by the DETAILTAPEPARM statement. A detailed description of the optional parameters is presented immediately following the DETAILTAPEPARM statement description.

7.3.2.12.4 The DETAILTAPEPARM Statement

In `prefix.MICS.PARMS(cccOPS)`, the optional `DETAILTAPEPARM` statement is used to specify parameters defining processing options for CA MICS DETAIL tape data sets at the component (`ccc`) level.

When specified in `prefix.MICS.PARMS(cccOPS)`, the `DETAILTAPEPARM` parameter specifications override unit level tape data set defaults established by `TAPEPARM` and `DETAILTAPEPARM` statements in `prefix.MICS.PARMS(JCLDEF)`.

Refer to chapter 2 of the CA MICS Planning, Installation, Operations, and Maintenance Guide for unit level `TAPEPARM` and `DETAILTAPEPARM` statements in `prefix.MICS.PARMS(JCLDEF)`.

The primary reason to include a `DETAILTAPEPARM` statement in `prefix.MICS.PARMS(cccOPS)` is to specify unique parameters for the DETAIL tapes created for a particular component (`ccc`).

For example, your site may want to direct the CA MICS DETAIL tapes for a particular component (`ccc`) to a unique tape volume pool, or to a tape device type such as a virtual tape unit, or to a DASD pool backed by an archive facility. If so, this could be accomplished by including the appropriate `DETAILTAPEPARM` statements in `prefix.MICS.PARMS(cccOPS)`.

`DETAILTAPEPARM <parameters>`

Note: In `prefix.MICS.PARMS(cccOPS)`, all `DETAILTAPEPARM` statements must precede any `TAPEfff` or `TAPEfffsms` statements.

The general form of the `DETAILTAPEPARM` statement is the keyword `DETAILTAPEPARM`, followed by one or more `DETAILTAPEPARM` parameters. `DETAILTAPEPARM` parameters include both keyword parameters and keyword=value specifications. You can specify the `DETAILTAPEPARM` statement as many times as you want in order to specify all of the tape data set definitions you require. If you specify a parameter multiple times, the last instance defines the value in effect.

The following summarizes the `DETAILTAPEPARM` parameters:

- o Each parameter is described in detail below.
- o While the following syntax sample shows only one or two parameters per statement (line), you have the

option to specify as many parameters as you want on each statement (limited only by the 72-character-maximum line length).

```
DETAILTAPARM GDG=nnn UNIT=unitname VOLCOUNT=n
DETAILTAPARM EXPDT=expdt / RETPD=nnn / NOEXPDT
DETAILTAPARM COMP/NOCOMP
DETAILTAPARM STACK/NOSTACK AFF/NOAFF
DETAILTAPARM MODEL='modeldsn' / MODEL=NOMODEL
DETAILTAPARM DISK / TAPE SPACE=spacedef
DETAILTAPARM VOLSER=volser
```

The following is a sample DETAILTAPARM statement:

```
DETAILTAPARM GDG=3 VOLCOUNT=7
```

In this example, the default parameter values are used for most of the DETAILTAPARM parameters.

The following describes each of the DETAILTAPARM parameters:

- o GDG=nnn

Defaults to 7.

The number of entries to be created in the GDG for each DETAIL tape data set.

Overrides the TAPARM statement GDG=nnn.

- o UNIT=unitname

Defaults to the value specified on the TAPARM statement or to UNIT=3490 if UNIT=unitname is not specified on the TAPARM statement.

The generic unit name for tape file allocation. The parameter value (unitname) must be a valid 1 to 8 character generic unit name for your installation. The value will be used in generated tape data set JCL DD statement UNIT parameters. CA MICS does not validate the unitname value except to verify that it is non-blank and no more than 8 characters in length.

Overrides the TAPARM statement UNIT=unitname Value, the old-style TAPEUNIT statement, or both.

- o VOLCOUNT=n

Defaults to the value specified on the TAPEPARM statement. If no VOLCOUNT is specified, the z/OS default volume count of 5 applies.

The volume count JCL parameter for CA MICS tape data sets. Specify a decimal number from 1 to 255.

Overrides the TAPEPARM statement VOLCOUNT=nnn parameter or the old-style VOLCOUNT statement.

- o EXPDT=expdt

Defaults to the value specified on the TAPEPARM statement. If no expiration date or retention period is specified, a default retention period is used which equals the GDG=nnn value.

Specify an expiration date definition which replaces the derived default retention period (RETPD) specifications. This facility is specifically intended for use by installations with tape management systems which require specific EXPDT definitions.

Note: For some tape management systems, certain EXPDT values makes the tape available for scratching immediately. Such values are NOT suitable for CA MICS tapes.

The EXPDT=expdt parameter is optional, and, if omitted, CA MICS derives a RETPD=nnn specification as appropriate for the specific CA MICS tape data set being created and the number of GDG entries defined.

Overrides the TAPEPARM statement EXPDT=expdt or NOEXPDT parameter, or the old-style EXPDT statement.

Note: The EXPDT=expdt, RETPD=nnn, and NOEXPDT parameters are mutually exclusive.

- o NOEXPDT

Defaults to the value specified on the TAPEPARM statement.

Specify NOEXPDT if you want CA MICS to omit both the EXPDT=date and RETPD=days JCL parameters from generated JCL statements for new tape data set creation. This option is provided for those installations where the tape management system or SMS specifications automatically control tape expiration dates, retention periods, or both.

Overrides the TAPEPARM statement EXPDT=expdt or NOEXPDT parameters, or the old-style EXPDT statement.

Note: The NOEXPDT, EXPDT=expdt, and RETPD=nnn parameters are mutually exclusive.

o COMP / NOCOMP

Defaults to the value specified on the TAPEPARM statement. If COMP or NOCOMP is not specified, the z/OS default for the selected tape unit type applies.

This parameter specifies data compaction (COMP) or no data compaction (NOCOMP) for a tape device enabled for hardware compaction. Data compaction is only supported for IBM standard labels. If specified, the value will be coded in the TRTCH subparameter of the DCB JCL parameter. If not specified, no TRTCH subparameter will be generated.

Overrides the TAPEPARM statement COMP/NOCOMP parameter or the old-style IDRC statement.

o STACK / NOSTACK

Defaults to NOSTACK for DETAIL tapes.

Specify STACK to place multiple CA MICS tape files on a single tape volume. This mode of operation uses a minimum number of tape volumes and tape mounts are minimized. CA MICS JCL statements are generated with the VOL=REF=ddname construct and unit affinity is

enforced (that is, STACK also means AFF).

Specify NOSTACK (the default) to place exactly one CA MICS tape file on a tape volume. With this option the generated CA MICS JCL statements will NOT contain the VOL=REF=ddname construct. This option is useful if your implementation of a virtual tape system prohibits "stacked" tape volumes, or if you are directing CA MICS tape files to a DASD pool backed with an archival facility.

Overrides the TAPEPARM statement STACK/NOSTACK parameter.

o AFF / NOAFF

Defaults to the value specified on the TAPEPARM statement or to AFF.

Specify AFF to use a single tape unit for all related tape files. This is the normal mode of operation where a minimum number of tape units are used and tape mounts are minimized. CA MICS JCL statements are generated with the UNIT=AFF=ddname construct. AFF is required for the STACK option.

Specify NOAFF to enable allocation of different tape units for each CA MICS tape data set. With this option the generated CA MICS JCL statements will NOT contain the UNIT=AFF=ddname construct. You may find this option useful with certain virtual tape system implementations, or where you are directing CA MICS tape files to a DASD pool backed with an archival facility.

Note, if you specify NOAFF, then NOSTACK is automatically set.

Overrides the TAPEPARM statement AFF/NOAFF parameter.

o MODEL='modeldsn' / MODEL=NOMODEL

Defaults to the value specified on the TAPEPARM statement.

The fully qualified name of a cataloged data set to be used when defining or extending GDG entries.

Ensure that the data set's DCB characteristics are LRECL=0 and BLKSIZE=0. Coding BLKSIZE=0 causes SAS to use the blocksize defined by the BLKSIZE(TAPE) option defined in the configuration member(s) referenced by the CONFIG DD statement. If this parameter is omitted, a CA MICS generation job will allocate a default model GDG data set.

If NOMODEL is specified as the model dataset name, the use of the model dataset in the DCB specification will be bypassed.

Overrides the TAPEPARM statement MODEL='modeldsn' parameter or the old-style SYSTEMMODEL statement.

- o DISK / TAPE

Defaults to TAPE.

Specify DISK if you want to write the dataset to a disk instead of a tape. Specify TAPE to write the dataset to a tape.

When you specify DISK, make sure that you have both a UNIT parameter with a generic unit name of a disk device and a SPACE parameter defining disk space. You can also define a VOLSER parameter to direct the data to a specific disk.

Overrides the TAPEPARM statement DISK/TAPE parameter.

- o SPACE=spacedef

No default.

Defines the space allocation parameter for a data set defined with the DISK parameter. Specify this parameter according to JCL language syntax.

Overrides the TAPEPARM statement SPACE= parameter.

- o VOLSER=volser

No default.

Defines the volume serial number of a disk for a data set defined with the DISK parameter. Specify this parameter according to JCL language syntax.

Overrides the TAPEPARM statement VOLSER= parameter.

7.3.2.12.5 DETAILTAPESMSPARM and TAPEfffsSMS Statements

In prefix.MICS.PARMS(cccOPS), the optional DETAILTAPESMSPARM and TAPEfffsSMS statements allow you to specify System Managed Storage (SMS) keywords (for example,STORCLAS=storclass) for DETAIL tape data sets at the component and individual file level, respectively.

DETAILTAPESMSPARM

When specified in prefix.MICS.PARMS(cccOPS), the DETAILTAPESMSPARM parameter specifications override unit level tape data set SMS defaults established by TAPESMSPARM and DETAILTAPESMSPARM statements in prefix.MICS.PARMS(JCLDEF).

Refer to chapter 2 of the CA MICS Planning, Installation, Operations, and Maintenance Guide for unit level TAPESMSPARM and DETAILTAPESMSPARM statements in prefix.MICS.PARMS(JCLDEF).

The primary reason to include a DETAILTAPESMSPARM statement in prefix.MICS.PARMS(cccOPS) is to specify unique SMS parameters for the DETAIL tapes created for a particular component (ccc).

For example, your site may want to direct the CA MICS DETAIL tapes for a particular component (ccc) to a unique storage class.

Sample DETAILTAPESMSPARM statement:

```
DETAILTAPESMSPARM STORCLAS=DTPCLASS
```

In this example, the JCL for all DETAIL tape data sets for the component will include the SMS STORCLAS=DTPCLASS assignment.

TAPEfffsSMS

The TAPEfffsSMS statement is only permitted in prefix.MICS.PARMS(cccOPS). It is not supported in prefix.MICS.PARMS(JCLDEF). TAPEfffsSMS keyword specifications override, for the indicated file, any SMS related tape specifications made in prefix.MICS.PARMS(JCLDEF) and prefix.MICS.PARMS(cccOPS) with TAPESMSPARM or DETAILTAPESMSPARM statements.

The primary reason to include a TAPEfffsSMS statement in

prefix.MICS.PARMS(cccOPS) is to specify unique SMS parameters for a specific DETAIL tape file (fff).

Sample TAPEfffSMS statement:

```
TAPEfffSMS STORCLAS=fffCLASS
```

In this example, the JCL for the DETAIL tape data set for file ffft will include the SMS STORCLAS=fffCLASS assignment.

7.3.2.12.6 DETAIL Tape User Exits

A user exit is invoked just prior to the output of each record to a DETAIL tape file. The exits are named _USRTfff, where fff matches the file identifier for the DETAIL tape file.

Refer to chapter 10 of this guide for a complete description of the DETAIL tape user exits.

7.3.2.12.7 Activating DETAIL Tape Checklist

- ___ 1. Add or modify the desired TAPEfff and TAPEfffSMS statements in prefix.MICS.PARMS(cccOPS), where fff is the three-character file identifier for the file to be written to tape.

- ___ 2. Submit prefix.MICS.CNTL(cccPGEN).

- ___ 3. Edit prefix.MICS.PARMS(JCLGENU) so that it reads:

cccGDGDN DAILY

If Incremental Update is active, add this:

INCRccc

- ___ 4. Submit prefix.MICS.CNTL(JCLGENU) to regenerate the JCL for DAILY and cccGDGDN.

Note: If you want to use specific JCL and SMS parameter values for your DETAIL tape data sets, make sure you have coded the appropriate DETAILTAPEPARM and/or DETAILTAPESMSPARM statements in prefix.MICS.PARMS(JCLDEF) for unit level, or prefix.MICS.PARMS(cccOPS) for component level DETAIL tape JCL parameters. Refer to the previous sections for a detailed discussion on JCL parameters available for DETAIL tape data sets.

- ___ 5. Submit prefix.MICS.CNTL(cccGDGDN) to define the tape GDG.

Note: If you want to override the number of entries kept in a GDG with cataloged data sets for a TAPEfff file, you must delete and redefine the index with the new limit for GDG entries. Follow the checklist in section 3.5.6.3 of the PIOM.

- ___ 6. You have just activated DETAIL tape for one or more database files. As mentioned in the DETAIL Tape Processing Overview section, the choice to create DETAIL tape files is independent of, and does not affect the creation of DETAIL files in the unit database.

If you are currently creating one or more cycles of the files you just activated for DETAIL tape in your unit prefix.MICS.DETAIL timespan, your DAILY update

job will continue to do so.

Now that you are creating DETAIL tape cycles for the files, you may want to deactivate, or limit the number of DETAIL cycles written to the unit DETAIL timespan database.

The CA MICS System Modification Guide contains instructions for changing the number of cycles of a file as well as instructions for deactivation of the DETAIL timespan of a file.

If you deactivate the files in the DETAIL timespan, make sure you delete the existing cycles to free up space in the prefix.MICS.DETAIL database. If you reduce the number of cycles, make sure you delete any existing cycles in prefix.MICS.DETAIL beyond the new cycle limit.

The following JCL can be used to delete specific cycles of a file from the DETAIL timespan. Make sure to replace 'x' with the unit identifier:

a) Delete all cycles for file fff (01-nn)

```
//DELETE EXEC MICSDbX
//SYSIN DD *
PROC DATASETS DDNAME=%DDNx(TS=DETAIL);
  DELETE cccfff01 cccfff02 ... cccfffn ;
RUN;
```

b) Delete cycles 02, 03, and 04 for file fff:

```
//DELETE EXEC MICSDbX
//SYSIN DD *
PROC DATASETS DDNAME=%DDNx(TS=DETAIL);
  DELETE cccfff02 cccfff03 cccfff04 ;
RUN;
```

7.3.2.12.8 Deactivating DETAIL Tape Checklist

- ___ 1. Delete the desired TAPEfff and TAPEfffSMS statements from prefix.MICS.PARMS(cccOPS).
- ___ 2. If there are no longer any TAPEfff statements in prefix.MICS.PARMS(cccOPS), delete any DETAILTAPEPARM and DETAILTAPESMSPARM statements.
- ___ 3. Submit prefix.MICS.CNTL(cccPGEN).
- ___ 4. Edit prefix.MICS.PARMS(JCLGENU) so that it reads:

DAILY

If Incremental Update is active, add this:

INCRccc
- ___ 5. Submit prefix.MICS.CNTL(JCLGENU) to regenerate the JCL for the DAILY job.
- ___ 6. (Optional) Delete the index and the cataloged datasets for each TAPEfff statements deleted in step 1.

+-----+ DATA CENTER PREPARATION WORKSHEET: CICS Component Options Definition PARMs Library Member is CICOPS Reference Section: 7.3.2, CA MICS CICS Analyzer Guide +-----+	
	CICS PROCESSING OPTIONS:
RESPONSE TIME THRESHOLDS-->	RESP limit 1 limit 2 limit 3 Limit 4 Limit 5 Limit 6 Limit 7
NUMBER OF WORK FILE PAIRS>	WORK data_set_allocation_parameters
INTERNAL STEP RESTART---->	RESTART YES/NO
INCREMENTAL UPDATE----->	INCRUPDATE YES/NO INCRDB PERM/TAPE/DYNAM INCRDETAIL data_set_allocation_parameters ...
MULTI-SYSTEM ACCOUNTING-->	MSACCOUNT
DETAIL.CICCSU01 on tape ->	TAPECSU <specification overrides> TAPECSUSMS <SMS parameter override>
DETAIL on tape all files->	DETAILTAPEPARM <specification overrides> DETAILTAPEMSPARM <SMS parameter override>
COPY CICS STATISTICS ->	STATCOPY
TRANS ID TRANSLATION----->	TRANSLATE Original Transid New Transid TRANSLATE Original Transid New Transid
CICS SYSTEM DEFINITIONS-->	Orig CICS SMF/ Input GMT Appl ID/ SYSID ID ddname Type Offset SYSIDNT Int Selcode
One CICOPTS statement per CICS system to be considered.	CICOPTS _____ CMF _____ CICOPTS _____ MON _____ CICOPTS _____ EPL _____ (Note: Generic SYSID spec.) CICOPTS ???? _____ TCE _____
+-----+	

7.3.2.13 VERIFY RELEASE Statement

The VERIFY RELEASE statement is used when some or all of the input records come from a CICS release that might not yet be supported by CA MICS. It is a parameter with possible values ABORT or NOABORT.

The ABORT option causes the DAY040 step of the DAILY or the INCR040 step of the Incremental Update (INCRIC) job to ABEND when the first unsupported record is read. The NOABORT option allows the DAY040 or INCR040 to complete with no ABEND.

Insert the VERIFY RELEASE statement into the prefix.MICS.PARMS(CICOPS) and code as:

VERIFY RELEASE ABORT

or

VERIFY RELEASE NOABORT

If this statement is omitted from the CICOPS member, the default option is NOABORT.

The prefix.MICS.CNTL(CICPGEN) job must be run to make the VERIFY RELEASE effective.

Usage Notes

- o The ABORT option and prefix.MICS.PARMS(CICTHRSH)

If ABORT has been coded and the CICPGEN has been run, then the DAY040 or INCR040 steps will ABEND by default if any unsupported records are detected. To override this default, one can code a sufficiently high value for CIC06020 in prefix.MICS.PARMS(CICTHRSH). Since there is only one count for each CICS region that has an unsupported release, the value that is assigned to CIC06020 does not have to be extremely high to eliminate the possibility of an ABEND.

For example, the following would be sufficient to eliminate the ABEND in any reasonable case:

CIC06020 999999

- o The NOABORT option and prefix.MICS.PARMS(CICTHRSH)

If NOABORT is coded and the CICPGEN was run, then it overrides whatever is coded for CIC06020 in the prefix.MICS.PARMS(CICTHRSH). Thus the CICTHRSH has no impact in this case.

- o When there is no ABEND

If NOABORT has been taken as the option or if ABORT is the option but CIC06020 has been coded with a value that is higher than the number of unsupported CICS regions, the DAY040 or INCR040 steps skips the records from the unsupported regions and processes the records from any supported regions that it finds.

- o What Happens to Unsupported Records that are Skipped?

If the DAY040 or INCR040 steps are allowed to process supported records and to skip unsupported records, maintenance to support any new release of CICS or of Allen Systems Group (ASG) TMON (TCE) should be obtained from CA MICS Product Support and should be applied as soon as possible. Once the maintenance is applied, run a later DAY040 or INCR040 update to process the records that have been rejected in earlier runs. This can require the use of the Force option as described in the CA MICS Planning, Install, Operation, Maintenance Guide (PIOM) sections 4.3.4.4.4 and 4.3.7.5.

7.3.3 INPUTRDR and INPUTCIC PARMS Members

The prefix.MICS.PARMS members INPUTRDR and INPUTccc are used to specify the input data for the Data Integration Application component step.

To determine whether INPUTccc or INPUTRDR or both are used, review the DAYSMF, SMFRECORDING, and SMFDRCRTR specifications in prefix.MICS.PARMS(JCLDEF). Consider the following table and comments for the row:

ROW	PARAMETER IN JCLDEF	INPUTRDR WHERE USED	INPUTccc WHERE USED
---	-----	-----	-----
1	DAYSMF OFF	Not Used	Each Comp Step
2	DAYSMF FILE(S) ...	DAYSMF Step	Not Used
3	Only 1 SMF Comp	Comp Step	Not Used
4	SMFDIRECTOR	Not Used	Comp Step
5	DAYSMF EXCLUDE ccc	Not Used	Comp ccc Step
6	SMFRECORDING ccc	It depends	Comp ccc Step

Row 1: When DAYSMF OFF is specified in JCLDEF, then the prefix.MICS.PARMS(INPUTccc) member is used for each component step.

Row 2: When DAYSMF FILES PERMANENT or TEMPORARY is specified in JCLDEF, the prefix.MICS.PARMS(INPUTRDR) member is used in DAYSMF.

Row 3: If there is only one SMF recording component in this unit, DAYSMF will not be generated and the component step with SMF input data will use the prefix.MICS.PARMS(INPUTRDR) member.

Row 5: If the use of DAYSMF EXCLUDE ccc results in only one component remaining as DAYSMF eligible, DAYSMF will automatically be deactivated, and row 1 applies.

Row 6: Some components accept data from both SMF and non-SMF sources, such as CIC and VCA. In this case, the component may optionally input data from the INPUTRDR member, but will always include the INPUTccc member. Refer to the component's guide.

For details on the DAYSMF parameters, see PIOM section 2.3.3.2.1, JCL Option Definitions (JCLDEF).

For details on deactivating DAYSMF, see PIOM section 5.10, Removing the DAYSMF Step from the DAILY Job.

The following sections describe how to specify the input data set JCL definitions.

- 1 - INPUTRDR PARMS Member JCL Definition
- 2 - INPUTCIC PARMS Member JCL Definition

7.3.3.1 INPUTRDR PARMS Member JCL Definition

The INPUTRDR member of prefix.MICS.PARMS defines the DD statements that specify the SMF input data for all products in the units that use SMF data. The INPUTSMF DD statements are used by the DAYSMF step, which reads and splits the SMF data into separate files, one for each CA MICS product.

The INPUTRDR member defines the input SMF data sets when DAYSMF FILES PERMANENT or TEMPORARY is specified in the prefix.MICS.PARMS(JCLDEF) member.

An INPUTSMF DD statement is required for each SMF data source. A worksheet for preparing the INPUTRDR member is provided below. If you change the contents of the INPUTRDR member, you must regenerate the CA MICS DAILY job using either JCLGENU or JCLGEND in prefix.MICS.CNTL.

For example, if you want to read your SMF data from a data set named SMF.DAILY.DATA, you would change the INPUTRDR member to contain the name of the SMF data set as follows:

```
//INPUTSMF DD DISP=SHR,DSN=SMF.DAILY.DATA
```

If there is only one SMF recording component in this unit, DAYSMF will not be generated and the component step with SMF input data will use the prefix.MICS.PARMS(INPUTRDR) member directly.

If DAYSMF has been deactivated by the DAYSMF OFF statement specified in the prefix.MICS.PARMS(JCLDEF) member, then each component step will read its input from their corresponding prefix.MICS.PARMS(INPUTccc) member.

```

+-----+
| INSTALLATION PREPARATION WORKSHEET:  SMF Input DD Statements |
|
| PARMs Library Member is INPUTRDR |
+-----+
|
| This definition is required to specify the DD statement for SMF data |
| which will be read by the DAILY CA MICS job. |
|
|   //@
|   //@ WARNING: ALWAYS MAKE CHANGES IN PARMs(INPUTRDR) AND NOT
|   //@           &CNTL(DAILY).
|   //@           CHANGES MADE TO &CNTL(DAILY)
|   //@           WILL BE GONE WHEN DAILY REGENERATED BY JCLGEN.
|   //@
|   //INPUTSMF DD DISP=SHR,DCB=BUFNO=3,DSN=_____
|   //          DD DISP=SHR,DCB=BUFNO=3,DSN=_____
|
+-----+

```

Figure 7-4. INPUTRDR JCL Definition Worksheet

7.3.3.2 INPUTCIC PARMS Member JCL Definition

The INPUTCIC member of prefix.MICS.PARMS contains the DD statements to specify input data for the CA MICS Analyzer for CICS. The //INPUTSMF DD and the optional //SMFDRCTR DD statements are used by the DAY040 step of the DAILY job.

The INPUTCIC member contains the input SMF data sets when ANY of the following JCLDEF options is used:

- o DAYSMF OFF is specified
- o SMFDRCTR/SMFDIRECTOR CIC is specified
- o DAYSMF EXCLUDE CIC is specified

A //INPUTSMF DD statement is required for each SMF data source.

A //SMFDRCTR DD statement is only required when CA SMF Director split indices are used.

A worksheet for preparing the INPUTCIC member is provided below in Figure 7-5.

If you change the contents of the INPUTCIC member, you must regenerate the CA MICS DAILY job using either JCLGENU or JCLGEND in prefix.MICS.CNTL.

The INPUTCIC member can contain JCL definitions for non-SMF input, such as data from ASG-TMON for CICS TS (TCE) and CMF journals, or it can contain JCL definitions for SMF input if you do not specify 'SMFRECORDING CIC' in prefix.MICS.PARMS(JCLDEF). If 'SMFRECORDING CIC' is specified in JCLDEF, then it must be defined in the input JCL in the INPUTRDR member in prefix.MICS.PARMS.

- o If you specified MON as the ddname for ASG-TMON for CICS TS (TCE) input in CICOPS, supply the ASG-TMON for CICS TS (TCE) data sets in INPUTCIC using the INPUTMON ddname.
- o If you specified TCE as the ddname for ASG-TMON for CICS TS (TCE) 2.0+ input in CICOPS, supply the ASG-TMON for CICS TS (TCE) 2.0+ data sets in INPUTCIC using the INPUTTCE ddname.

```
//@ TMON INPUT
//INPUTMON DD DISP=SHR,DSN=TMON.CICSA1.DATA
```

```
//          DD DISP=SHR,DSN=TMON.CICSA2.DATA
//          DD DISP=SHR,DSN=TMON.CICSB1.DATA
//@ TCE INPUT
//INPUTTCE DD DISP=SHR,DSN=TCE.CICSR1.DATA
//          DD DISP=SHR,DSN=TCE.CICSR2.DATA
//@ CICS JOURNALS
//INPUTJ1  DD DISP=SHR,DSN=PROD.CICSPD1.JRNLA
//INPUTJ2  DD DISP=SHR,DSN=PROD.CICSTD1.JRNLA
```

Note: If your site has CA SMF Director installed, you can use it to extract specific SMF record types and subtypes at SMF dump time. For more details on this product feature, see section 6.4, CA MICS and CA SMF Director Interface.

```
+-----+
| INSTALLATION PREPARATION WORKSHEET: INPUTCIC JCL Definitions |
| PARMs Library Member is INPUTCIC                             |
+-----+
| This definition is required to specify the DD statement for data |
| that will be read by the DAILY CA MICS job.                     |
|                                                                    |
| //@                                                              |
| //@ WARNING: ALWAYS MAKE CHANGES IN PARMs(INPUTCIC) AND NOT  |
| //@ &CNTL(DAILY) .                                              |
| //@ CHANGES MADE TO &CNTL(DAILY)                              |
| //@ WILL BE GONE WHEN DAILY REGENERATED BY JCLGEN.             |
| //@                                                              |
| //INPUTCIC DD DISP=SHR,DCB=BUFNO=5,DSN= _____            |
| //          DD DISP=SHR,DCB=BUFNO=5,DSN= _____            |
| //INPUTTCE DD DISP=SHR,DCB=BUFNO=5,DSN= _____            |
| //INPUTMON DD DISP=SHR,DCB=BUFNO=5,DSN= _____            |
| //@                                                              |
| //SMFDRCTR DD DISP=SHR,DCB=BUFNO=5,DSN= _____            |
| //          DD DISP=SHR,DCB=BUFNO=5,DSN= _____            |
+-----+
```

Figure 7-5. INPUTCIC JCL Definition Worksheet

7.3.4 Reserved for Future Use

This section is reserved for future use.

7.3.5 CICS Application Unit Definition (CICAPU)

The CICS Analyzer provides application-level usage and workload information by means of user-defined application units. The application unit is stored in the CA MICS data element CICAPU (CICS Application Unit ID) and is derived via a user exit that you supply to CA MICS. CICAPU is used as a file summarization key for the CICC AU (CICS Application Unit Activity) and CICC AC (CICS Application Count) files. It enables you to classify and summarize CICS workload data in a customized fashion to meet your reporting needs. Although the method of classification varies among CA MICS users, it is usually based on one of the following approaches:

- o Classification by transaction identification

This approach classifies the CICS workload by transaction ID. For example, transactions 'ABCD' and 'WXYZ' are always heavy resource drains, while other transactions are quick inquiry transactions. Normally this approach utilizes a table lookup of CICS transaction IDs to group the work.

- o Classification by service area requested

This approach classifies work based on the service area exercised. For example, a bank may divide transactions into the application categories of demand deposits, time deposits, administrative services, and system support activity. This method typically identifies the application unit by a prefix of the CICS Transaction ID (such as demand deposit transaction IDs beginning with the letter R).

- o Classification by user

This approach classifies transaction data according to the user who requested the service. This method can use various ways to extract the identity of the requester from the CICS terminal or user identifiers.

DEFINING THE APPLICATION UNIT

The data element CICAPU is derived by the user exit CICAURT (CICS Application Unit Derivation Routine) in prefix.MICS.PARMS(CICAURT). This exit is invoked for each transaction record processed during CA MICS daily processing. Any data elements read from the input transaction record, such as transaction ID, program, and user ID, can be stored

in CICAPU. Details on coding the CICAURT exit is provided in Section 7.3.6 of this guide.

CICAPU is 12 bytes in length. All unused bytes are padded with blanks. You can populate CICAPU by concatenating or manipulating multiple data elements during the CICAURT exit routine. For example, you might define your application structure as having two parts: project and transaction identifier. You can assign the first two bytes of the 12-byte CICAPU field to the project identification and the next four bytes to the CICS transaction ID. In this example, the actual values of the field might be:

111		
123456789012	Project	Transaction
-----	-----	-----
'DDRBAL'	Demand Deposits	Account Balance Inquiry
'TDINQN'	Time Deposits	Name and Address Inquiry

You should consider the following when defining CICAPU:

1. CICAPU is a file summarization key. Therefore, its content will have a direct impact on the size of the file where it is stored. The higher the number of combinations, the larger the file and the CA MICS data base become. Since certain transactions may be of more interest when considered as a group than as individual transaction ID, you may want to combine them in a single CICAPU to conserve space. Examples of these types of transactions are:
 - o CICS system transactions, the identifiers of which all begin with the letter prefixes such as CS or CE.
 - o Transaction IDs associated with purchased application packages, such as IBM Field Developed Programs.
 - o Trivial applications, especially those very frequently used, such as simple menu processors.
2. Any data element that may be used in later reporting from the CICCAC or CICCAC files must be coded into CICAPU if the data element is not kept in the file. An example is the data element TRANTYPE (transaction type), which may be useful in some sites to group CICS workloads by short, medium, long, or conversational transaction type. TRANTYPE is not part of the CICCAC or CICCAC file, so save it in CICAPU if you plan to report on it from these files.

7.3.6 CICS Application Unit Derivation Routine (CICAURT)

The CICS application unit derivation routine, CICAURT, is a user exit that you provide to derive the data element CICAPU (CICS Application Unit ID). This exit is written in SAS and is stored in member CICAURT of prefix.MICS.PARMS. It is called by the DAY040 step of the DAILY job for each transaction record processed by the CICS Analyzer. You can define a different version of the CICAURT exit for each unit data base to suit your CICS processing and reporting needs.

You can use any of the data elements that CA MICS reads from the CICS transaction record to derive the application unit. Some of the elements most commonly used to populate the CICAPU data element include:

- TRANCODE - The CICS transaction ID or the translation of the identifier as described in prefix.MICS.PARMS(CICOPS).
- PROGRAM - Program name.
- TRANSTYPE - C/S/M/L/X for conversational, short, medium, long, or excessive transaction types.
- TERMINAL - CICS terminal identifier.

You may modify the CICAURT exit routine at any time. You do not have to run any CA MICS generation jobs after making a change. The change will take effect in the next DAILY job execution.

CODING CONSIDERATIONS

Follow these guidelines for coding this CA MICS exit routine:

1. Validate input data where possible. When invalid codes are encountered, they should be assigned to a CICAPU value representing the installation's overhead accumulator.

NOTE: Allowing invalid or garbage application units into the CA MICS Data Base significantly increases the number of records and therefore the DASD space requirements of the CICS Information Area files.

2. Ensure that all of the fields that you require are available for application unit construction. For

example, certain transactions may be executed without being attached to a terminal facility. This situation may occur for miscellaneous overhead transactions, such as the BMS message routing control transaction. Such a data collection record would have no valid terminal identifier present.

3. Refer to the discussion of exit coding in Section 2.3.1.5, Notes on Coding CA MICS Parameters, in the CA MICS Planning, Installation, Operation, and Maintenance Guide.

The CICS Analyzer delivers a sample CICAURT exit routine in prefix.MICS.PARMS(CICAURT). You should review and modify the sample code to suit your workload reporting needs. The sample exit is shown below. A worksheet for coding the CICAURT exit routine is shown in Figure 7-6.

```

/* ***** */
/* APPLICATION UNIT DERIVATION EXIT */
/* ***** */

/* ***** */
/* ALL CICS SYSTEM TRANSACTION IDS */
/* BEGIN WITH THE LETTER 'C'. CICS */
/* USES THIS NAMING CONVENTION TO */
/* RECOGNIZE SUCH TRANSACTIONS, SO */
/* THEY CAN BE TREATED SPECIALLY. */
/* FOR EXAMPLE, TRANSIDS THAT BEGIN */
/* WITH 'C' CANNOT BE DISABLED. */
/* SINCE THERE IS LITTLE CONSIDERA- */
/* TION FOR DETAILED ANALYSIS OF */
/* CICS SYSTEM TRANSACTIONS, GROUP */
/* THESE INTO 'OVERHEAD'. */
/* IF YOUR SITE HAS APPLICATION */
/* TRANSIDS THAT BEGIN WITH 'C', */
/* PUT SPECIFIC TESTS FOR THEM */
/* BEFORE THIS CONDITIONAL. */
/* ***** */
      IF TRANCODE = : 'C' THEN GOTO AURTOVHD;

/* ***** */
/* DEFAULT CICAUR TO TRANS ID. */
/* ***** */

      CICAUR = TRANCODE;
      GOTO AURTRTEX;
AURTOVHD:
      CICAUR = 'OVERHEAD';

```

AURTRTEX:

INSTALLATION PREPARATION WORKSHEET: CICS Application Unit Derivation Routine Definition PARMS Library Member is CICAURT Reference Sections: 7.3.5 and 7.3.6, CA MICS CICS Analyzer Guide
* VALIDATE FOR VALID APPLICATION UNITS, WHERE POSSIBLE ; IF application data is not valid GOTO AURTOVHD ;
* BUILD APPL. UNIT FIELDS; CICAPU=field source 1 field source n ;
GOTO AURTRTEX ; * LINKED ROUTINE TO BUILD INSTALLATION OVERHEAD APPLICATION UNITS; AURTOVHD: CICAPU='overhead category' ;
AURTRTEX:
...5...10...15...20...25...30...35...40...45...50...55...60...65...70..

Figure 7-6. CICS Application Unit Derivation Routine Worksheet

7.3.7 CICS Relative Longevity Routine (CICRLRT)

In order to classify CICS transactions into response categories, the CICS Analyzer requires that you write an exit to assign a relative longevity code to each transaction as the transaction data is being processed during DAY040 step of the CA MICS DAILY job. This classification allows CA MICS to maintain response distributions of several transaction types.

The exit is called the the CICS relative longevity routine and is stored in member CICRLRT of prefix.MICS.PARMS. The CICRLRT exit assigns a value to the CA MICS data element TRANTYPE, a one-character field, which is set to S, M, L, or C for short, medium, long, or conversational, respectively. The CICS Analyzer uses the value of TRANTYPE to maintain separate response distributions for each type of transaction.

TRANTYPE may also be set to a value of X, denoting an excessive transaction. Such excessive transactions are NOT considered for purposes of total response distributions or for calculating average total response times.

Several approaches you might take to create the code include:

- o Classification based on transaction identification

This approach identifies the individual transactions according to their characteristic longevity. Normally, this approach utilizes a table lookup of CICS transaction IDs to group the work. For example, transactions 'ABCD' and 'WXYZ' are always long transactions, while other transactions may be medium or short.

- o Classification based on estimated resource usage

This approach classifies a transaction type based on the transaction's resource consumption such as CPU and I/Os.

- o Classification based on facility area

This approach classifies work based on the service area exercised. For example, all transactions which execute as destination-attached may be of "medium" duration.

The CICRLRT exit routine may reference any data elements CA MICS reads from the CICS transaction record. Some of the data elements most commonly used to determine the value of TRANTYPE include:

TRANCODE - The CICS transaction ID or the translation of the identifier as described in prefix.MICS.PARMS(CICOPS).

PROGRAM - Program name.

CSUTRSTM - Transaction response time.

CSUCPUTM - Task CPU time.

CSUFACTY - Facility type.

When invalid input makes the proper assignment of relative longevity impossible, you should assign a default code, usually L (long). Failure to do this significantly reduces the usability of the transaction group response time statistics. For example, certain transactions may be executed without being attached to a terminal facility. This situation may occur for miscellaneous overhead transactions, such as the BMS message routing control transactions. This kind of data collection record has no valid terminal identifier present.

You may modify the CICRLRT exit routine at any time. No CA MICS generation jobs are required after making a change. The change will take effect in the next DAILY job execution.

The CICS Analyzer delivers a sample CICRLRT exit routine in prefix.MICS.PARMS. This code classifies transactions based on the transaction ID. You should review and modify the sample code to suit your processing and reporting needs. The sample code is shown below.

```
* THE FOLLOWING TRANSACTIONS WILL BE CLASSIFIED AS
* CONVERSATIONAL:
*
* TRANSID  APPLICATION
* ----  -----
* CSMT    CICS MASTER TERMINAL FUNCTIONS
* CEMT    CICS EXTENDED MASTER TERMINAL FUNCTIONS
* AUTH    APPLICATION AUTHORIZATION FUNCTION
* RBAL    DEMAND DEPOSIT ACCOUNT BALANCE INQUIRY
* TBAL    TIME DEPOSIT ACCOUNT BALANCE INQUIRY
* UPAY    ADMINISTRATIVE PAYROLL FILE UPDATE
*
;
IF
  TRANCODE = :'CSMT' OR TRANCODE = :'CEMT' OR
  TRANCODE = :'AUTH' OR TRANCODE = :'RBAL' OR
  TRANCODE = :'TBAL' OR TRANCODE = :'UPAY'
THEN TRANTYPE = 'C';
*
* THE FOLLOWING TRANSACTIONS WILL BE CLASSIFIED AS
* SHORT:
*
* TRANSID  APPLICATION
* ----  -----
* RINQ    DEMAND DEPOSIT ACCOUNT VERIFICATION
* TINQ    TIME DEPOSIT ACCOUNT VERIFICATION
* INQU    CUSTOMER NAME AND ADDRESS INQUIRY
* HELP    NEW USER TUTORIAL
* CS--    (ANY TRANSACTION BEGINNING WITH 'CS')
*
```

```
      ;
      ELSE IF
        TRANCODE = :'RINQ' OR TRANCODE = :'TINQ' OR
        TRANCODE = :'INQU' OR TRANCODE = :'HELP' OR
        TRANCODE = :'CS'
      THEN TRANTYPE = 'S';
*
* THE FOLLOWING TRANSACTIONS WILL BE CLASSIFIED AS
* MEDIUM:
*
* TRANSID  APPLICATION
* ----  -----
* PRIN    TERMINAL PRINT SPOOLER
* EVAL    ARITHMETIC UTILITIES
* MEDM    ANOTHER MEDIUM TRANSACTION
* ACCT    ACCOUNT CROSS INDEX
* APAY    ADMINISTRATIVE PAYROLL TRIAL REPORT
* R---    (ANY TRANSACTION BEGINNING WITH 'R')
* T---    (ANY TRANSACTION BEGINNING WITH 'T')
*
;
      ELSE IF
        TRANCODE = :'PRIN' OR TRANCODE = :'EVAL' OR
        TRANCODE = :'MEDM' OR TRANCODE = :'ACCT' OR
        TRANCODE = :'APAY' OR TRANCODE = :'R' OR
        TRANCODE = :'T'
      THEN TRANTYPE = 'M';
*
* ALL OTHER TRANSACTIONS WILL BE CLASSIFIED LONG.
*
;
      ELSE TRANTYPE = 'L';
```

The worksheet for coding modifications to the Relative Longevity Code Derivation Routine is shown in Figure 7-7. If you modify the routine that is distributed with the CA MICS CICS Analyzer, you must test the accuracy of the code and data.

Note the construction of the above sample. Such a cascade of IF statements may not be efficient for large-volume applications, but it was included here for illustration.

```

+-----+
| INSTALLATION PREPARATION WORKSHEET:  CICS Relative Longevity Routine |
|                                     Determination                       |
| PARMs Library Member is CICRLRT                                         |
| Reference Section:  7.3.7, CA MICS CICS Analyzer Guide                 |
+-----+
|
| * VALIDATE FOR VALID SOURCE DATA, IF APPLICABLE:
|
| IF
| (condition c-1) _____
|
| (condition c-2) _____
|
| (condition . .) _____
|
| (condition . .) _____
|
| (condition c-n) _____
| THEN TRANTYPE = 'C'; /* CONVERSATIONAL TYPE */
| ELSE IF
| (condition s-1) _____
|
| (condition s-2) _____
|
| (condition . .) _____
|
| (condition . .) _____
|
| (condition s-n) _____
| THEN TRANTYPE = 'S'; /* SHORT TYPE */
| ELSE IF
| (condition m-1) _____
|
| (condition m-2) _____
|
| (condition . .) _____
|
| (condition . .) _____
|
| (condition m-n) _____
| THEN TRANTYPE = 'M'; /* MEDIUM TYPE */
| ELSE TRANTYPE = 'L'; /* LONG TYPE */
|
+-----+
| ....5...10...15...20...25...30...35...40...45...50...55...60...65...70.. |
+-----+

```

Figure 7-7. CICS Relative Longevity Routine Worksheet

7.3.8 CICS Processing Thresholds (CICTHRSH)

The CICTHRSH member of prefix.MICS.PARMS contains statements and parameters that define input processing thresholds to the CA MICS Analyzer Option for CICS. The primary purpose of the thresholds is to allow daily processing to continue, at your discretion, after invalid data or errors have been encountered during input.

* Please note that careful consideration should be given when setting input processing error tolerance thresholds because the result will be loss of data. Setting these threshold values should only be done when the data being rejected is not important and its loss is acceptable. If not, the underlying problem must be addressed instead.

* Also note that this member is only used for processing of CICS data. If your DAY040 step only processes CICS Transaction Gateway SMF type 111 records, you should not update this member. Simply leave the default values as delivered.

In addition, you can set thresholds to limit the size of the CICCDC (CICS Data Dictionary) file and the number of error messages displayed in the DAY040 MICSLOG.

You may modify CICTHRSH definitions at any time. No CA MICS generation jobs are required after making a change. The change will take effect in the next DAILY job execution.

The structure of the CICTHRSH member follows this format:

- o One CICCDEX statement
- o One CICMSGTH statement
- o One CIC060nn statement for each CIC060nn error that can be tolerated

Refer to Section 9.1.3.2 for more information.

CICCDEX STATEMENT: CMF Data Dictionary Record Retention
Period

CICCDEX number_of_days

Code one CICCDEX statement. This statement defines the period of non-reference that must occur before a CMF data dictionary record in the CICCDC01 file expires. The expired

records are physically deleted from the CICCDC01 file at the DETAIL timespan. Number_of_days defaults to 9999 if CICCDC01 is not defined.

Each CICS region in which CMF is active creates a set of data dictionary records during CMF initialization. The most recently created set of records is retained in the CICCDC01 file. This permits the processing of CMF data without data dictionary records present in the input. When a CICS CMF region becomes obsolete, its data dictionary records should be removed from the CICCDC01 file. This process is automated by detecting obsolete CICCDC01 records based on their period of inactivity or non-reference.

A period of non-reference is defined as the number of days between the date of last reference (data from a region was processed by the CA MICS Analyzer Option for CICS) and today's date. If the period of non-reference is greater than the CICCDC01 number_of_days, then the CICCDC01 record is deleted.

CICMSGTH STATEMENT: Input Processing Error Message Display

CICMSGTH number_of_iterations

Code one CICMSGTH statement. This statement defines the number of times an input processing message (CIC060nn messages, where nn is the message number 01-99) can be displayed for an input ddname. Number_of_iterations defaults to zero if CICMSGTH is not specified.

CIC060nn messages may be generated during the DAY040 processing step as errors are detected in the input. For example, CIC06011 indicates data is present for a CICS region not defined in prefix.MICS.PARMS(CICOPS). This message will be generated for each input record related to the undefined region. If there are 1,000 input records for the undefined region, there will be 1,000 occurrences of the CIC06011 condition for the input ddname containing data for this region. The number of times the CIC06011 message is displayed for this input ddname is limited by the CICMSGTH number_of_iterations value. If the value is 20, then the CIC06011 message will be displayed up to 20 times.

CICMSGTH specification prevents an unexpected condition from filling MICSLOG with redundant messages. The actual number of occurrences of any CIC060nn message for each input ddname is summarized and displayed in MICSLOG. Using the example

above, although a maximum of 20 CIC06011 messages are displayed for the ddname containing the undefined region, an informational message is displayed that indicates the condition occurred 1,000 times. The CICMSGTH threshold is message-specific: if the CIC06024 condition was encountered during processing of the input ddname that contained the CIC06011 condition, up to 20 occurrences of both messages are displayed.

CIC060nn STATEMENT: Input Processing Error Tolerance

CIC060nn number_of_occurrences

Code one CIC060nn statement for each CIC060nn error that can be tolerated. This statement defines the number of times an input processing error (CIC060nn messages, where nn is the message number 01-99) can be tolerated before end-of-input processing is followed by a user 998 abend.

Number_of_occurrences is set to null (no tolerance) if CIC060nn is not specified for a value of nn.

Using the example described in the preceding CICMSGTH description, a CIC06011 condition may be tolerated. An input ddname may contain data for regions that are not to be stored in CA MICS (for example, test regions). Specification of a tolerance value for the CIC06011 condition permits successful processing of input data that contains undefined (to CICOPS) regions. Suppression of the user 998 abend will not occur unless the number_of_occurrences tolerance value equals or exceeds the number of times the error condition occurred.

Alternatively, the input ddname may contain data for a new and undefined (to CICOPS) region that should be stored in CA MICS. If the CIC06011 statement is not specified (or number_of_occurrences is set to 0), then a user 998 abend will occur in the daily CICS input processing step, signaling a potential loss of data. The CA MICS Analyzer Option for CICS provides a default tolerance value of 999999 for message CIC06011.

We suggest that all CIC060nn statements be defined in response to a specific situation. When the error condition is corrected, the CIC060nn statement should be removed. If a condition is transient and the loss of data insignificant, the CIC060nn number_of_occurrences can be set to a small value. If the transient condition results in an unexpected significant loss of data, the user 998 abend will occur, indicating an unusual event.

A sample CICTHRSH member is illustrated below:

```
CICCDCEX 32  
CICMSGTH 20  
CIC06013 100
```

The effect of these statements is to:

- o Retain CICDC01 records for 32 days of non-reference.
- o Limit the display of any CIC060nn message for each input ddname to a maximum of 20 iterations.
- o Permit rejection of a maximum of 100 "short records" before a user 998 abend is issued after completion of input processing.

7.3.9 CICS Multisystem Account Derivation Exit (CICMSAC)

CICS provides a common token to identify transactions that are executed in an MRO (multiregion option) or ISC (intersystem communication) environment. This common token is known as the unit of work ID, which is stored in the CA MICS variable CICUOWID. The CA MICS Analyzer Option for CICS uses the token variable to sort detail transaction data and provides you with an exit point during which accounting-related information from the TOR is propagated to the AORs and FORs. The exit is known as the CICS multisystem account derivation exit. It is a user exit that you code in SAS. The exit is stored in member CICMSAC of prefix.MICS.PARMS.

You can activate the CICMSAC exit by specifying the MSACCOUNT statement in prefix.MICS.PARMS(CICOPS). If MSACCOUNT is present in CICOPS, then the CICMSAC exit code is included during the DAY040 step of the DAILY job. Otherwise, the code is omitted. By default, the CICMSAC exit is shipped turned off.

The purpose of the CICMSAC exit is to reset the CICS accounting data elements (CICACT1 through CICTACT9) in the AORs and FORs with that of the TOR. This ensures that MRO records written for the same transaction are summarized under identical account code values. This exit does not collapse multiple transaction records into one record to show end-to-end execution.

Prior to calling the CICMSAC exit, the CA MICS Analyzer Option for CICS first sorts the detail transaction data in the proper sequence. To ensure uniqueness, the CA MICS Analyzer Option for CICS sorts the data by CICNETNM (the originating system ID known to VTAM) and CICUOWID. To ensure that TOR data is sorted ahead of AOR and FOR data, mirror transaction start time is also used as part of the sort sequence keys.

To code the CICMSAC exit, you may consider using one of the commonly used approaches described below.

- o Changing the accounting fields to correspond directly to the accounting fields that were derived for the initial transaction. This approach retains the accounting field values on the initial transaction by using the statements:

```
IF FIRST.CICUOWID THEN DO;
```

```
        retain the accounting field values
    END;
    ELSE DO;
        set the accounting fields equal to the saved
        values
    END;
```

- o Changing the accounting fields to the value of the initial transaction, but using one of the accounting fields to designate that the transaction was the result of an MRO/ISC interaction.

```
    IF FIRST.CICUOWID THEN DO;
        save the accounting field values
    END;
    ELSE DO;
        set the accounting fields 1-3 equal to the saved
        values
        CICTACT4 = 'MRO';
    END;
```

You are responsible for testing the accuracy of modifications to the sample exit routine supplied in the CA MICS Analyzer Option for CICS distribution libraries. Be sure to follow the guidelines for coding exits in Section 4.3 of the System Modification Guide. The worksheet for programming the CICS Multisystem Derivation Exit is shown in Figure 7-8.

You can reference any of the data elements contained in the CICS monitor data collection records during the CICMSAC exit. The data elements that are most often used in deriving the account variables are:

TRANCODE - The CICS transaction ID or the translation of the identifier as described in the CICS Processing Options

TERMINAL - CICS terminal identifier

USERID - CICS RACF user identification

CICACT1-CICACT9 - The derived accounting codes

CICNETNM - The name of the originating CICS region

CICUOWID - The unique identifier for the originating transaction

OPERID - CICS operator identification from the CICS

Figure 7-8. CICS Multisystem Account Derivation Worksheet

7.3.10 Database Space Modeling (DBMODEL)

This section describes the information required to define the Analyzer to the Database Space Modeling Facility. Specifically, the user must provide values for the cycle (data retention) definitions. The special considerations for defining the FILE input statements to the Database Space Modeling facility are discussed.

Section 2.3.4 of the PIOM, Database Space Modeling Facility, explains how these values are used in estimating the DASD requirements of the database, and how the user can input these values to the modeling facility.

This section covers the following topics:

- 1 - Data Retention Specifications (FILE Statements)
- 2 - DBMODEL Input Statements

7.3.10.1 Data Retention Specifications (FILE Statements)

Data retention specifications tell the CA MICS database how many cycles of data to save both online and in archive mode for each file in each supported timespan.

Figure 7-9 provides a worksheet to define the different retention limits for the online and archive database files. The numbers shown on the worksheet reflect the default retention values in `prefix.MICS.PARMS(DBMODEL)`.

Use the worksheet to define your retention specifications, which, in turn, will allow you to determine the appropriate values for the modeling process.

Note that the DBMODEL worksheet shown here contains values for this product only. For additional information, see section 2.3.4.1, *Preparing the Modeling Input of the PIOM guide*.

The worksheet is organized by information area. Each file in the area is listed by name. For each file, a line is formatted to allow six definitions in the online database and two in the archive database:

- o The online database files quantify the number of cycles of data that is maintained in the DETAIL, DAYS, WEEKS, MONTHS, and YEARS timespans and the TABLES data area.
- o The two definitions for the archive database files quantify the number of cycles of data to be retained, up to the cutoff limit defined. The archive definitions have no impact on the size of the database and can be specified whethert the weekly and/or monthly archive history files have actually been activated (see Section 2.3.3, *CA MICS JCL Planning and Parameters*, of the PIOM).

The worksheet formats provide an underscored area for the user's definition, followed by the recommended value, shown within parentheses. If the underscored area contains a value of 00, the file is not supported for the indicated timespan. To add support, perform database tailoring as described in Section 6.2, *Tailoring the Database*, of the *System Modification Guide (SMG)*.

When specifying a retention limit, remember that the number may never be zero if the file has been defined to be active in the timespan.

A typical set of CICS file statements follows:

```
FILE CIC CICCXY 10 33 09 06 01 00 053 024
FILE CIC CICCXY 00 02 00 02 01 00 000 024
FILE CIC CICCXY 00 00 00 00 00 00 000 000
FILE CIC CICCXY 00 00 00 00 00 00 000 000
FILE CIC CICCIN 33 00 00 00 00 00 000 000
FILE CIC CICCDC 01 00 00 00 00 00 000 000
FILE CIC CICCMT 00 00 00 00 00 00 000 000
FILE CIC CICCST 00 00 00 00 00 00 000 000
```

+-----+ INSTALLATION PREPARATION WORKSHEET: Database Data Retention Definitions PARMS Library Member is DBMODEL Reference Section: 7.3.10, CA MICS CICS Analyzer Guide +-----+									
File	DETAIL	Online Database Retention					Archive Cut-Off		
Name		DAYS	WEEKS	MONTHS	YEARS	TABLES	WEEKS	MONTHS	
+-----+									
CICCSY	__ (10)	__ (33)	__ (09)	__ (06)	_ (1)	00 (00)	__ (053)	__ (024)	
CICCSU	__ (00)	__ (02)	__ (00)	__ (02)	_ (1)	00 (00)	__ (000)	__ (024)	
CICCAU	__ (00)	__ (00)	__ (00)	__ (00)	_ (0)	00 (00)	__ (000)	__ (000)	
CICCAC	__ (00)	__ (00)	__ (00)	__ (00)	_ (0)	00 (00)	__ (000)	__ (000)	
CICCIN	__ (33)	00 (00)	00 (00)	00 (00)	0 (0)	00 (00)	000 (000)	000 (000)	
CICCDC	__ (01)	00 (00)	00 (00)	00 (00)	0 (0)	00 (00)	000 (000)	000 (000)	
CICCMR	00 (00)	00 (00)	00 (00)	00 (00)	0 (0)	00 (00)	000 (000)	000 (000)	
CICCSF	00 (00)	00 (00)	00 (00)	00 (00)	0 (0)	00 (00)	000 (000)	000 (000)	
+-----+									
...5...10...15...20...25...30...35...40...45...50...55...60...65...70...									
+-----+									

Figure 7-9. Data Retention Specifications Worksheet

7.3.10.2 DBMODEL Input Statements

The DBMODEL member of prefix.MICS.PARMS provides the input to the Database Space Modeling Facility.

Update the DBMODEL member using the information collected on the worksheet in Figure 7-9. To actually perform the space modeling, submit the jobs as described in Section 2.3.4.2 of the PIOM.

Chapter 8: INSTALLATION

You install the CA MICS Analyzer Option for CICS by following the instructions in the PIOM. Chapter 7 of this guide provides instructions for the correct definition of parameters that are specific to the CA MICS Analyzer Option for CICS. The SYSID, SITE, ZONE, and other parameter files that are used by more than one CA MICS component are defined in Chapter 2 of the PIOM. Section 3.8 of the PIOM provides step-by-step checklists that you must follow to ensure a successful installation.

Chapter 9: PROCESSING

This chapter describes the daily processing flow of the CICS Analyzer, the exit points available during the daily update, and considerations for running the daily DAY040 update.

This section contains the following topics:

[9.1 Processing Considerations](#) (see page 677)

[9.2 Daily Update Processing Flow](#) (see page 691)

[9.3 Utility Programs](#) (see page 717)

9.1 Processing Considerations

The DAY040 step of the daily update job is the step that processes CICS data. This section discusses the types of data processed, CMF dictionary processing, and dynamic processing.

- 1 - Types of Data Processed
- 2 - CMF Dictionary Processing
- 3 - Dynamic Processing

9.1.1 Types of Data Processed

The DAY040 step of the daily update job can process various types of CICS data during a single execution. The following sections describe the types of CICS data that can be processed.

- 1 - CMF Data
- 2 - CICS Statistics Data
- 3 - ASG-TMON for CICS Data

9.1.1.1 CMF Data

The CICS Monitoring Facility (CMF) collects performance data at the transaction level for later offline analysis. The records are type 110 records, which are in an SMF format. The type 110 records that are processed into CA MICS are those that have subtype 1 (Monitoring) and subtype 2 (Statistics).

For subtype 1 records, CA MICS elements are populated from Performance records (Class 3) and optionally from Exception records (Class 4). Before the Performance records for a particular CICS region can be processed, a special type 110 Data Dictionary record (subtype 1 class 1) must be input. This record provides the information that is needed to decode the Performance records for that region.

Each CMF data field has associated with it a unique numeric value called a connector, which is used by CA MICS to locate the entry in the data dictionary that describes the data field. This dictionary entry contains a field ID, a type, and a length. The type value indicates the format of the data field, such as packed or character, and the length indicates the number of bytes for the field. CA MICS stores the type and length of each field as elements in SAS arrays and uses the unique field ID as the subscript for the arrays.

Data dictionary records are required only to decode the Performance records, which are those of subtype 1 and Class 3 within this subtype. No other type 110 records or records from another data source such as ASG-TMON require Data dictionary records.

Data sets containing CMF records are in a Variable Blocked Spanned (VBS) format.

CICS Transaction Server 1.2 and Above

The CICS/TS release numbers are of the form 1.x, 2.x, and 3.x. CICS/TS is often abbreviated as CTS. In CICS TS release 1.2 and higher, CMF produces three classes of data: data dictionary (class 1), performance (class 3), and exception (class 4). In all classes of data, CMF records contain a SMF header, a SMF product section, and one CICS data section. Depending on the class of data, the CICS data section can contain a single dictionary or exception record, or multiple transaction records.

- o SMF Header

The SMF header contains the date and time when the record was written to SMF. This timestamp is used by CA MICS for checkpoint processing. The header also contains the record type and system identification. For CMF, the record type is 110, subtype 1.

- o SMF Product Section

The SMF product section identifies the CICS version number, CICS application ID as known to VTAM, and the class of data contained in the CICS data section that follows.

- o CICS Data Section

There is only one CICS data section in a type 110, subtype 1 record. For the dictionary class, the section contains a single dictionary record that describes the format of the performance class transaction record.

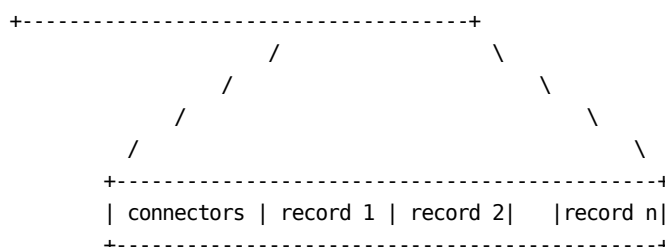
Note: There is no dictionary record for the exception class data.

The figure below shows the format of a dictionary class record.

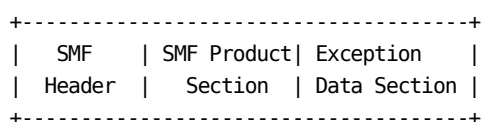
```
+-----+
|  SMF   | SMF Product| Dictionary  |
| Header |   Section  | Data Section |
+-----+
```

For performance class, the section contains a string of field connectors, followed by multiple logical transaction records. The field connectors identify the data fields that are present in each transaction record in the section, and are used by CA MICS to locate the entries in the dictionary record that describes the data fields. For each logical record in the section, a separate observation is created in the CA MICS database. As a result, the number of observations in the CA MICS detail transaction file will not match the number of physical records read. The figure below illustrates the format of a performance class record.

```
+-----+
|  SMF   | SMF Product| Performance |
| Header |   Section  | Data Section |
+-----+
```



For exception class, the section contains a single record that is written when a CICS exception has occurred. The figure below illustrates the format of an exception class record.



CICS 2.x

For CICS releases 2.x, CMF produces three classes of data: accounting (class 2), performance (class 3), and exception (class 4). In all classes of data, CMF records contain a SMF header, a product section, and one or more CICS data sections.

- o SMF Header

The SMF header contains the date and time when the record was written to SMF. This timestamp is used by CA MICS for checkpoint processing. The header also contains the record type and system identification. For CMF, the record type is 110.

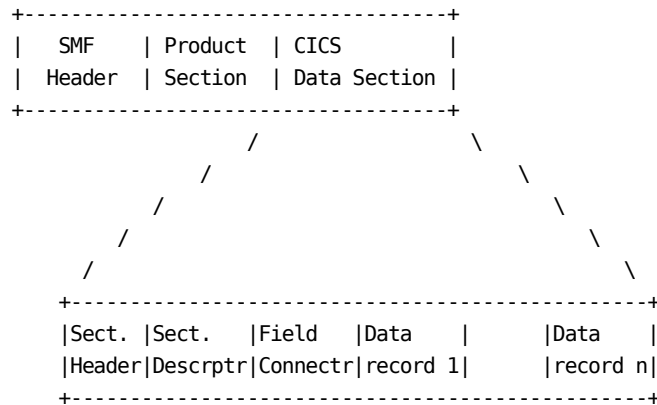
- o Product Section

The product section identifies the CICS application ID as known to VTAM, and the class of data contained in subsequent CICS data sections.

- o CICS Data Section

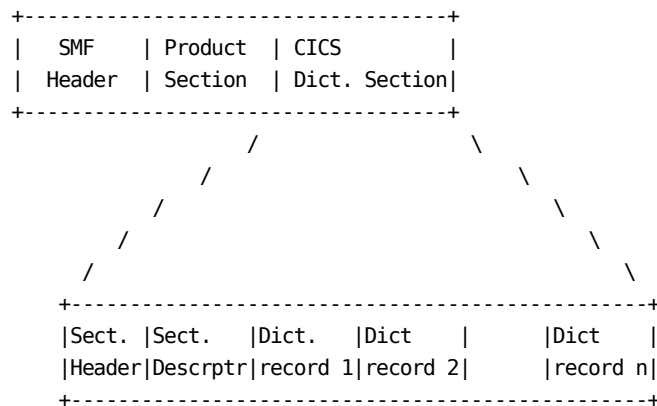
Except for the dictionary class data, each CMF record contains one or more CICS data sections, which are comprised of a section header, a section descriptor, a string of field connectors, and one or more data records. The figure below shows the format of a CMF performance,

accounting, or exception class record.



The section header contains the class of data in this CICS section: accounting, performance, or exception. The section descriptor contains the type of data within the performance class: dictionary, transaction, or global. The field connectors describe the data records that follow. Each data record is a group of CICS data elements stored adjacently. The connector values correspond one to one with the data elements in the data records.

For the dictionary class data, the CICS data section contains one or more data dictionary records, each of which describes the format of the other classes of data. The figure below shows the format of a dictionary class record.



DICTIONARY RECORD PROCESSING

As mentioned earlier, the CA MICS Analyzer Option for CICS requires data dictionary records to process CMF data

records. This is true except when processing Performance class 3 records in CICS 3.1.1 and higher releases. No data dictionary records are required to process Exception class 4 records.

The CA MICS Analyzer Option for CICS processes a data dictionary record by reading the dictionary data entry associated with each data field and storing that information in arrays.

The arrays contain field ID, type, and length for each data element. Each array is identified by APPLID, CICSID, and CICS release number. These arrays are discussed in more detail in Section 9.1.2, CMF Dictionary Processing.

DATA RECORD PROCESSING

The CA MICS Analyzer Option for CICS processes each data record by first reading the string of connectors that precedes the data. It passes each connector to a special informat called \$CICDEP, which uses the connector to locate the dictionary entry necessary for reading a subsequent data field. Once the dictionary entry has been found, \$CICDEP uses the type and length information from the dictionary to input the data field, and converts it to its proper format. The CA MICS Analyzer Option for CICS processes each connector and its corresponding data field until all the connectors have been exhausted. This process is repeated for each data record found in each CICS data section. If a data dictionary cannot be located for a data record, the data record will be skipped.

The use of the \$CICDEP routine to process an entire data record in a given call greatly reduces the number of SAS INPUT calls that would normally have to be performed, thus greatly reducing SAS processing overhead.

9.1.1.2 CICS Statistics Data

Beginning with CICS 3.1.1, CICS statistics are written to SMF on an interval basis. The statistics data have replaced CMF, which no longer produces system global records, as the source for system level information. To accommodate this change, the CICS Analyzer processes six global statistics records to populate the CICCXY (CICS System Activity) file with information previously obtained from CMF. This processing is done for CMF input at CICS 3.1.1 and higher releases.

The six global statistics records are:

- o Dispatcher (type 55, 56, 60, and 62)
- o Loader (type 30)
- o Storage manager task subpool (type 6)
- o Storage manager (type 2, 14, and 29)
- o Temporary storage (type 48)
- o Transient data (type 45)

The default SMF recording interval for CICS statistics is 3 hours. For CA MICS processing, a maximum of 1 hour is required; otherwise, the CICCXY file at the DAYS timespan contains missing values for observations during the hour when no statistics records are found.

During DAY040 input processing, the CICS Analyzer creates a work file for each of these statistics records. At end of input, the files are merged and written to the CICCXY work file, which is then sorted, summarized, and saved in the CA MICS data base at various timespans. All statistics work files, except for the dispatcher file (CICGDP), are deleted at the end of DAY040 processing. The CICGDP file is used to calculate delta CPU value for each interval. Only one cycle of the CICGDP file is kept at the DETAIL timespan.

9.1.1.3 ASG-TMON for CICS Data

The CA MICS Analyzer Option for CICS processes ASG-TMON system and transaction records, which can be in a compressed format. To decompress the data, a special function, CICONUT, is called by DAY040 to intercept the SAS INPUT statement and pass control to ASG-TMON's decompressor (\$CRCPRS for Version 8 and above). After decompression, control is returned to SAS to continue with normal input processing.

Although ASG-TMON does not produce data dictionary records, the \$CICDEP informat used to process CMF data is also used to process the decompressed ASG-TMON data. During DAY040 initialization, a pseudo dictionary is built for ASG-TMON, which is then used by \$CICDEP to read the data records.

9.1.2 CMF Dictionary Processing

To read CMF records, the CICS Analyzer must first obtain the description of each data field from the data dictionary records. The only exception to this is when reading the exception class data in CICS 3.1 and higher releases, which do not produce a data dictionary for this class of data. The following provides a general flow of CMF dictionary processing.

In the first DAILY execution after CICS Analyzer has been installed, DAY040 expects to find the dictionary records ahead of the data records for each region in the input file. When a dictionary record is found, it is saved in storage so that it can be used to read subsequent data records. If a data record is found before the dictionary record, and it is not a CICS 3.1 (or higher) exception record, then it is rejected. If the input file contains multiple dictionary record for the same region and CMF class, the CICS Analyzer keeps the latest copy in storage.

To keep the dictionary information in storage, the CICS Analyzer stores the field ID, type, and length of each data field in SAS arrays. Each array references multiple monitoring class, such as performance, accounting (pre 3.1) and exception (pre 3.1). The index for these arrays is derived based on the connector value that associates a dictionary entry with its corresponding data field.

Since the connector values are only unique within a particular class, and multiple classes of data exist in each array, a class index is added to the connector value to derive the subscript for the dictionary arrays. The class index for accounting data is 0; the class index for performance data is 20; and the class index for exception data is 296. To find the dictionary array index for a particular data element, the CICS Analyzer adds the element's connector value to 0, 20, or 296, respectively, depending on the element's class.

When storing the dictionary information in SAS arrays, DAY040 performs additional conversion to accommodate inconsistencies in the data from one CICS release to the next, such as overlapping field IDs for user clocks and counters in CICS release 2.1 and higher. For user clocks, DAY040 changes the left-most digit of the field ID to a 7; for user counters the left-most digit is changed to an 8, and for user areas a 9.

DAY040 also changes the field ID to type S data fields for

CICS release 2.1 and above. This is because type S data fields contain an additional flag byte which was absent in CICS 1.6. If the field ID is less than 100, DAY040 changes the left-most digit to 5; otherwise, it is changed to 6.

Because the size of the SAS Program Data Vector, which contains the names and locations of a program's SAS variables, is limited, the dictionary arrays (one set per region) are maintained in a storage table outside of the Program Data Vector. This means that only one set of field ID, type, and length arrays is actually defined to SAS. As a result, the SAS arrays are refreshed each time the region and class have changed, so that they always contain the dictionary information necessary to process the current data record.

To move the dictionary information between the SAS arrays and the storage table, the function CICVMAM is used. This function builds and accesses the storage table via GET and PUT requests. A GET request causes CICVMAM to locate the dictionary information in the storage table and move them to the SAS arrays. A PUT request moves the data in the SAS arrays to the storage table.

Once the SAS arrays have been loaded with the proper dictionary information by CICVMAM, DAY040 calls the informat \$CICDEP to read the data records and store the data in SAS variables.

At the end of input processing in the first DAILY, assuming no errors have occurred, the CICS Analyzer moves the dictionary information from the storage table to DETAIL.CICCDC01. Each observation in CICCDC01 contains the connector, type, length, and field ID of a single CMF data field. The observations are keyed by APPLID, CICSID, and CICSREL (CICS release number). They also contain a last reference date, which is used to delete observations that are not referenced within a specified period of time.

In subsequent DAILY executions, the CICS Analyzer loads the content of DETAIL.CICCDC01 in storage prior to processing the input file. This is done to ensure the successful processing of data records which have no corresponding dictionary records in the input file. It is normal for dictionary records to be absent from input, since many shops do not shut down CICS on a daily basis (dictionary records are written when CMF is first activated). Once the dictionary information is loaded in storage, the same logic described above is used to read the data records. At the end of input

processing, if all goes well, the content of the storage table is moved to DETAIL.CICCDC01. The same dictionary processing logic takes place for the next DAILY run.

9.1.3 Dynamic Processing

The DAY040 step of the daily update job inputs the following SOURCE and PARMs library members to configure special processing features:

- 1 - CICDEPEL - CICS Data Element Processor Control
- 2 - CICTHRSH - CICS Input Processing Thresholds

9.1.3.1 CICDEPEL - CICS Data Element Processor Control

Sharedprefix.MICS.SOURCE member CICDEPEL is used during daily processing of CMF data to assign the name of the SAS variable used to store each data element. It is also used for processing The Monitor data, for which CA MICS builds a temporary data dictionary for the duration of the input routine. CICDEPEL is read using the \$CICDEP informat. This causes \$CICDEP to input CICDEPEL and to configure itself for CMF data processing. CICDEPEL contains three types of information, described in conjunction with the statements that contain the information.

The first type of information is contained in the CONTROLVAR statements. \$CICDEP requires certain control data that describes the various CMF dictionary arrays that are used to process the CMF data elements. The required information includes:

- o the name of the dictionary field ID array
- o the name of the dictionary length array
- o the name of the dictionary type array
- o the variable containing the current record length
- o the variable containing the current data class
- o the size of the three dictionary arrays
- o the current lower bound array index for this class
- o the current upper bound array index for this class
- o the name of the element connector array
- o the variable containing the current connector length
- o the variable containing the current connector count

The second type of information is contained in the DEFINEID statements. These statements indicate to \$CICDEP the mapping of CICS field IDs to SAS data variable names. Each field ID is preceded by a one-digit number that indicates the class of the data element as follows: 2 is accounting class, 3 is performance class, and 4 is exception class. There are also DEFINEID statements for The Monitor system and transaction data. These elements have been assigned sequential field IDs and class values of 5 and 6 respectively for The Monitor version 7.0/7.1 and class values of 7 and 8 respectively for The Monitor version 8.0 and above.

Certain CICS field IDs are comprised of more than a single piece of data, such as type S elements, which are a clock value, a flag value, and a count. Such elements resolve to multiple SAS data variables and are described in the DEFINEID statements by using identical field IDs.

Because different releases of CMF data use the same field IDs and because \$CICDEP requires uniqueness of field IDs in order to correctly process the data, there is a deviance between some field IDs in the CMF dictionary and the IDs used in the DEFINEID statements. The following transformations were made to allow unique field IDs:

- o In CICS release 2.1 and above, the type S data field contains a flag byte that is absent from CICS 1.6. To differentiate between the two formats, the type S fields in CICS 1.7 and above with a field ID less than 100 have a 5 moved into the left-most ID position. Otherwise, a 6 is moved into the left-most ID position.
- o In CICS release 2.1 and above, the user clocks, user counters, and user area use field IDs 1 to n, where n is the number of clocks, counters, or areas. Since these field IDs are used by other variables, the left-most ID position is set to 7 for clocks, 8 for counters, and 9 for user areas.

The DEFINEIDCOUNT statement is used to indicate to \$CICDEP the number of entries for a repeating variable that has only one field ID. This number is essentially the number of elements defined in the SAS array into which the repeating variables are stored. This is used by CICS 1.6 user counters (field ID 65) and clocks (field ID 66) and by The Monitor file segments (field ID 115) and The Monitor user segments (field ID 130).

9.1.3.2 CICTHRSH - CICS Input Processing Thresholds

Prefix.MICS.PARMS member CICTHRSH defines three threshold values that are used during daily update processing: CICCDC01, CICSMSGTH, and CICO60nn.

CICCDC01

The CICCDC01 parameter indicates how long CMF dictionary records are to be maintained in the SAS CICCDC01 file. The parameter is entered as follows:

CICCDC01 nn

where nn is the number of days a CMF dictionary record will remain in the CICCDC01 without being referenced. After nn days, the record is deleted from the CICCDC01 during the next DAILY update process.

CICSMSGTH

The CICSMSGTH parameter is a value that indicates how many times a particular error message will be displayed on the MICSLOG data set before it is suppressed. The parameter is entered as follows:

CICSMSGTH nn

After a message has been display on the MICSLOG nn times, any further issuance of the message will be suppressed. The number nn cannot exceed 999999999.

CICO60nn

The CICO60nn parameter indicates the number of times message nn can be issued before daily update processing abnormally terminates. The parameter is entered as follows:

CICO60nn mm

After message CICO60nn has been issued mm times, daily update processing will abnormally terminate.

9.2 Daily Update Processing Flow

This section describes the daily processing flow for the CA MICS Analyzer Option for CICS and its relation to the locations of the CICS user exit routines. Figure 9-1 gives a breakdown of the modules that comprise the DAY040 step of the daily update process.

Module:	Calls:	Description:	Created by:
INCLLIB(DAY040)			
	USOURCE(\$BASMSTR)	BAS unit master processing code	BASPGEN
	SOURCE(\$BAS SFD,	BAS SFD flags	BASCGEN
	\$CICSFD)	CIC SFD flags	CICCGEN
	USOURCE(#BASEXIT,	BAS unit level exit overrides	installation tailored
	\$CICCYCS,	CIC unit cycle definition	CYCLEGEN
	#CICMSTR,	CIC complex master processing code	CICPGEN
	USOURCE #CICEXIT,	CIC unit level exit overrides	installation tailored
	USOURCE #USRMAC)	Installation defined macros	installation tailored
	USOURCE(\$CICDYST,	CIC Checkpoint processing steps	CICPGEN
	USOURCE \$CICXKPT)	CIC Checkpoint processing options	CICPGEN
	INCLLIB(BASACT)	Accounting and Chargeback code	delivered code
	SOURCE(CICACTCK)	CIC verify accounting files active	delivered code
	SOURCE(BASDAILY)	BAS checkpoint restart control	delivered code
		%includes the following:	
	SOURCE(CHECK,	checkpoint validation code	delivered code
	DYCICFM1,	CIC Input processing	delivered code
	DYCICFM2,	CIC sort and merge CSY data to DETAIL	delivered code
	DYCICFM3,	CIC MRO account processing	delivered code
	DYCICSUM,	CIC summarization routines	CICCGEN
	DYCICAGE,	CIC file aging routine	CYCLEGEN
	POST)	checkpoint update routine	delivered code

Figure 9-1. DAY040 Structure

The DAY040 step of the daily update job processes information in the phases described below, using the routines shown. The phases 1 and 2 are done in sharedprefix.MICS.SOURCE(DYCICFM1)

Phase 1 - Input CICS Data Dictionary

- CICCDALD - Load and control conversion of dictionary records.
- CICCDCCV - Converts CDC dictionary records into internal array formats.
- CICCDCTF - Performs dictionary element conversions.
- CICVMAMU - Manages memory dictionary array elements.

Phase 2 - Input Raw CICS Data

- CICSMFIP - Processes SMF header for CMF.
- CICCMFIP - Processes raw CMF data.
- CICCSTIP - Processes CICS statistics (CICS 3.1+).
- CICMONIP - Processes raw ASG-TMON data.
- CICTCEIP - Processes raw ASG-TMON for CICS TS (TCE) 2.0+.

Phase 3 - Information Area Processing

- DYCICFM2 - Eliminates duplicate data, summarizes system-level data, computes necessary interval values.
- DYCICFM3 - Executes CICMSAC routine to match MRO data allowing map of account codes from TOR to AOR and FOR transaction records.

Phase 4 - Database Timespan Processing

- DYCICSUM - Creates DAYS timespan files from DETAIL and updates week-to-date and month-to-date information, where applicable. The journal files used by CA MICS Accounting and Chargeback are populated during this phase.

Phase 5 - File Aging

- DYCICAGE - Ages the DETAIL and DAYS cycles, deleting the oldest cycles. This routine also replaces week-to-date and month-to-date cycles.

The following sections describe the general flow in each of the phases outlined above and the location in the flow of the standard user exits.

- 1 - Input CICS Data Dictionary
- 2 - Input Raw CICS Data
- 3 - Information Area Processing
- 4 - Database Timespan Processing
- 5 - File Aging
- 6 - MICSLOG Operational Report

9.2.1 Input CICS Data Dictionary

In this phase of processing, the CICS Dictionary File `DETAIL.CICCDC01` is loaded in a table in storage which is managed by a special routine, `CICVMAM`. The dictionary is used to drive the input processing for both CMF and The Monitor data. The dictionary describes each element within a CICS record: the element's field ID, the type of data, and its length. This information is used by the special INFORMAT `$CICDEP` to input, convert, and place the CICS data into SAS data variables.

The `$CICDEP` INFORMAT is also used to process data from The Monitor. During this phase, a dictionary is constructed in memory to describe The Monitor data.

No user exits are invoked during this phase.

The first part of Figure 9-2 outlines the flow of this phase.

9.2.2 Input Raw CICS Data

In the input raw CICS data phase, records are read from one of the data sources specified in prefix.MICS.PARMS(CICOPS). Data can come from CMF or ASG-TMON. Any combination of these data sources can be processed in a given DAY040 step. For CICS release 3.1.1 and above, the CA MICS Analyzer Option for CICS also reads CICS statistics data to obtain system level information previously provided by CMF global data.

The CA MICS Analyzer Option for CICS also processes any data that has the same format as any of the above data sources, such as the CMF type 110 look-alike records produced by OMEGAMON II for CICS.

The data from these sources is processed sequentially. That is, all CMF data is processed, followed by all ASG-TMON data. The data is formatted to create SAS observations.

Figure 9-2 outlines the flow of this phase.

The user exits invoked in this phase are listed below. An X in the data source column means that you can code the exit using data from that source.

	CMF	MON
o General exits	---	---

You have the option to define your own database file during the main input step using the following exits:

_USRSDKP - user file name & keep list	X	X
_USRSLFL - user file len, fmt, lbl	X	X

After reading the SMF record header, you can use the following exit to skip records before any data validation takes place:

%USRSKIP - skip unwanted data	X

After the product section has been input, but prior to input of the data sections, the following exit is invoked:

_USRSEL - input record selection	X	X

This exit can be used to change the original SYSID since it is invoked prior to the SYSID macro. It cannot reference any transaction-level information since it is invoked prior to the input of the data sections.

%USRSCID - set CICSID X

This exit can be used to set the CICSID when the record's ORGSYSID/APPLID is not defined in CICOPS.

	CMF	MON
o Output exits	---	---

Prior to the output of the specified file, these exits can be used for data element manipulation:

_USRSCIN - CICCIN element derivation	X	X
_USRSCSW - CICC SW element derivation	X	X
_USRSCMF - CMF transaction-level exit	X	
%USRSCST - CICS statistics (CICS 3.1 and higher)	X	
%USRSTAT - STATCOPY for CICS stats (CICS 3.1 and higher)	X	
_USRSCDC - CICCDC element derivation	X	X
_USRSCSY - CICC SY element derivation	X	X
_USRSCSU - CICC SU element derivation	X	X
_USRSCAU - CICC AU element derivation	X	X
_USRSCAC - CICC AC element derivation	X	X
_USRSCSF - CICC SF element derivation		X
_USRUTID - ASG-TMON user segment ID		X
_USRSCMR - CICC MR element derivation		X

	CMF	MON
o Parameter-related exits	---	---

CICRLRT - relative longevity	X	X
CICACRT - account code derivation	X	X
CICAURT - application unit deriv.	X	X
CICMSAC - multi-system account derivation exit	X	X

```

*****
* Initialization *
*****
|
+-----+
| Delete level 00 |
| and oldest cycles|
+-----+
|
+-----+
| Delete any      |
| work files      |
+-----+
|
+-----+
| Initialize      |
| process control |
| variables       |
+-----+
|
+-----+
| Input thresholds|
| from PARMs      |
| member CICTHRS  |
+-----+
|
+-----+
| Initialize      |
| $CICDEP INFORMAT|
| Input SOURCE     |
| (CICDEPEL)       |
+-----+
|
| /-----\ no +-----+
| < $CICDEP OK? >----->| ABEND 470 |
| \ +-----+
|
+-----+
| Load data       |
| dictionary       |
| arrays          |
+-----+
|
+-----+
| 2-1 |
+-----+

```

Figure 9-2. Input Raw Data Processing (Part 1 of 13)

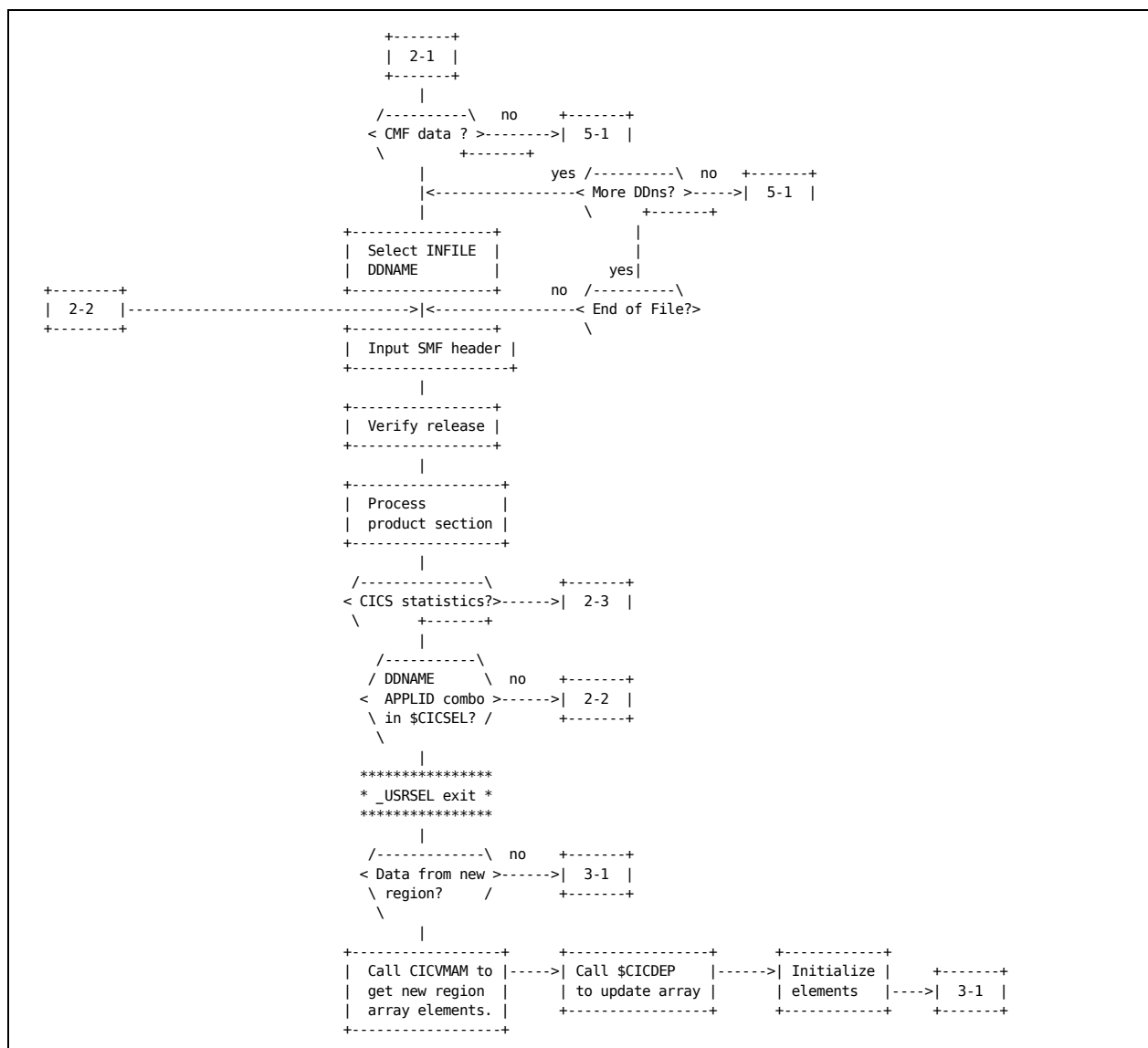


Figure 9-2. Input Raw Data Processing (Part 2 of 13)

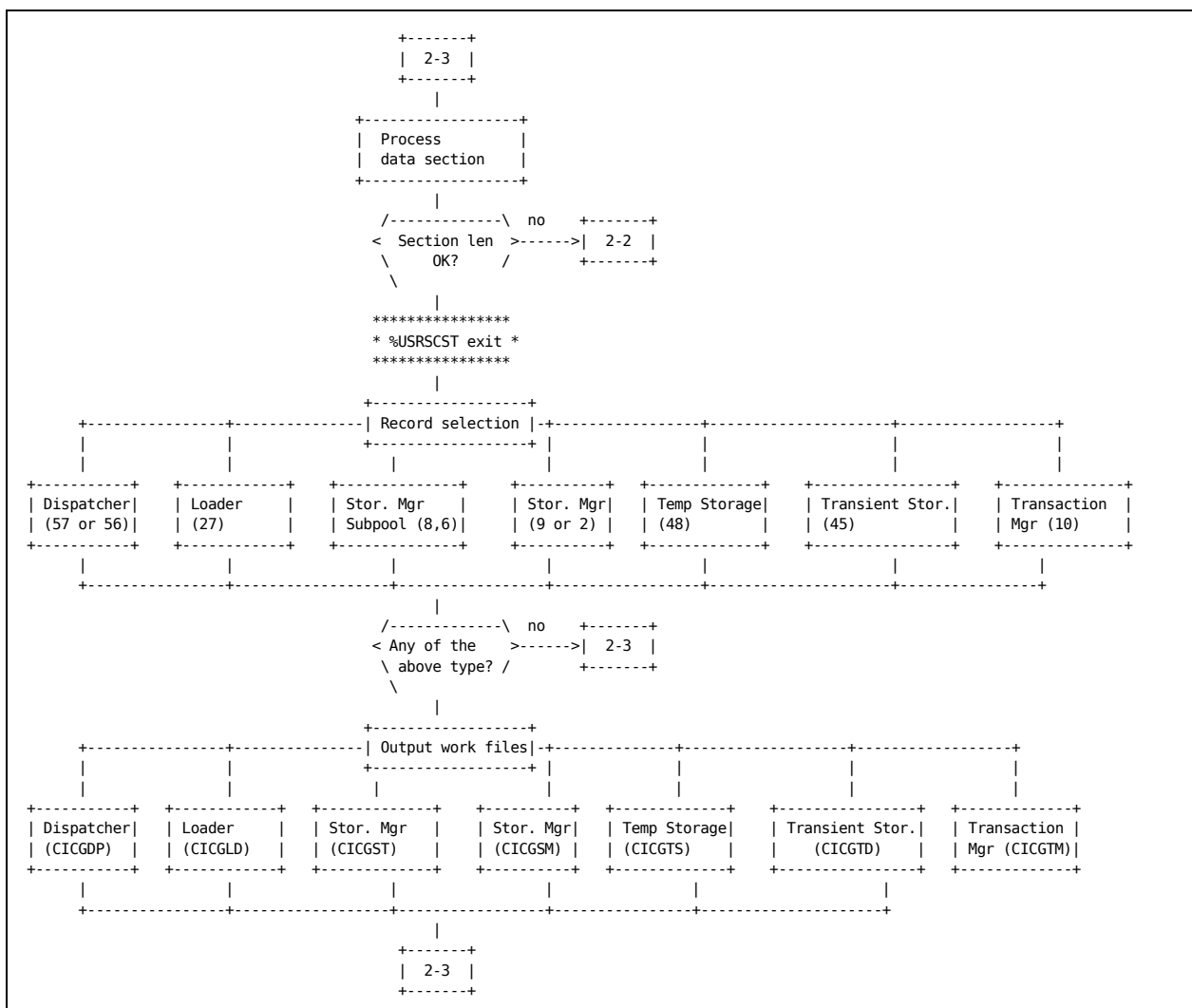


Figure 9-2. Input Raw Data Processing (Part 3 of 13)

```

+-----+
| 3-1 |
+-----+
|
+-----+
| Process CICS |
| section      |
+-----+
|
+-----+
| /-----\ no +-----+
| < More >----->| 2-2 |
| \ sections?/    +-----+
| \
|
+-----+
| /-----\ no +-----+
| < Section len >----->| 2-2 |
| \ OK? /              +-----+
| \
|
+-----+
| /-----\ no +-----+
| < Valid class?>----->| 3-1 |
| \                    +-----+
|
+-----+
| /-----\ no +-----+
| < Dictionary >----->| 3-2 |
| \ data? /            +-----+
| \
|
+-----+
| Output record |<----->* User output *
| to CICCDC    |          * exit USRSCDC *
+-----+
|
+-----+
| Call CICVMAM to | +-----+
| update region   |<----->| 3-1 |
| array elements. | +-----+
+-----+

```

Figure 9-2. Input Raw Data Processing (Part 4 of 13)

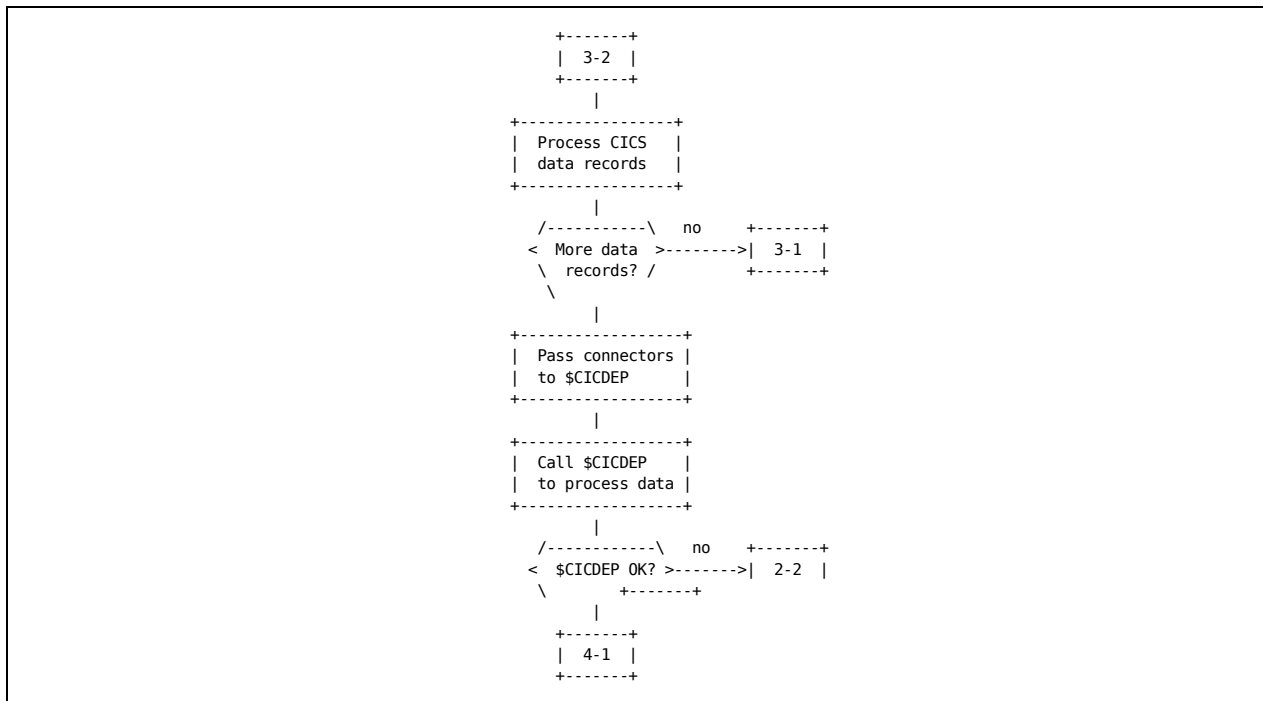


Figure 9-2. Input Raw Data Processing (Part 5 of 13)

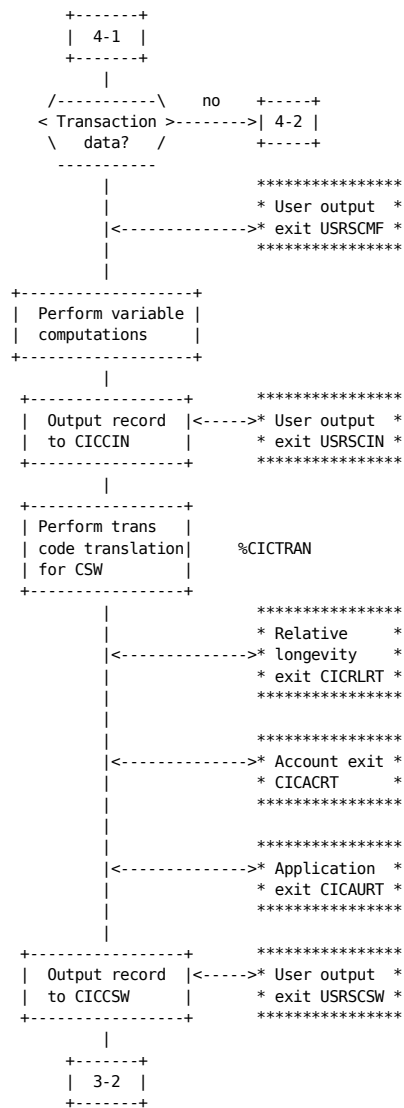


Figure 9-2. Input Raw Data Processing (Part 6 of 13)

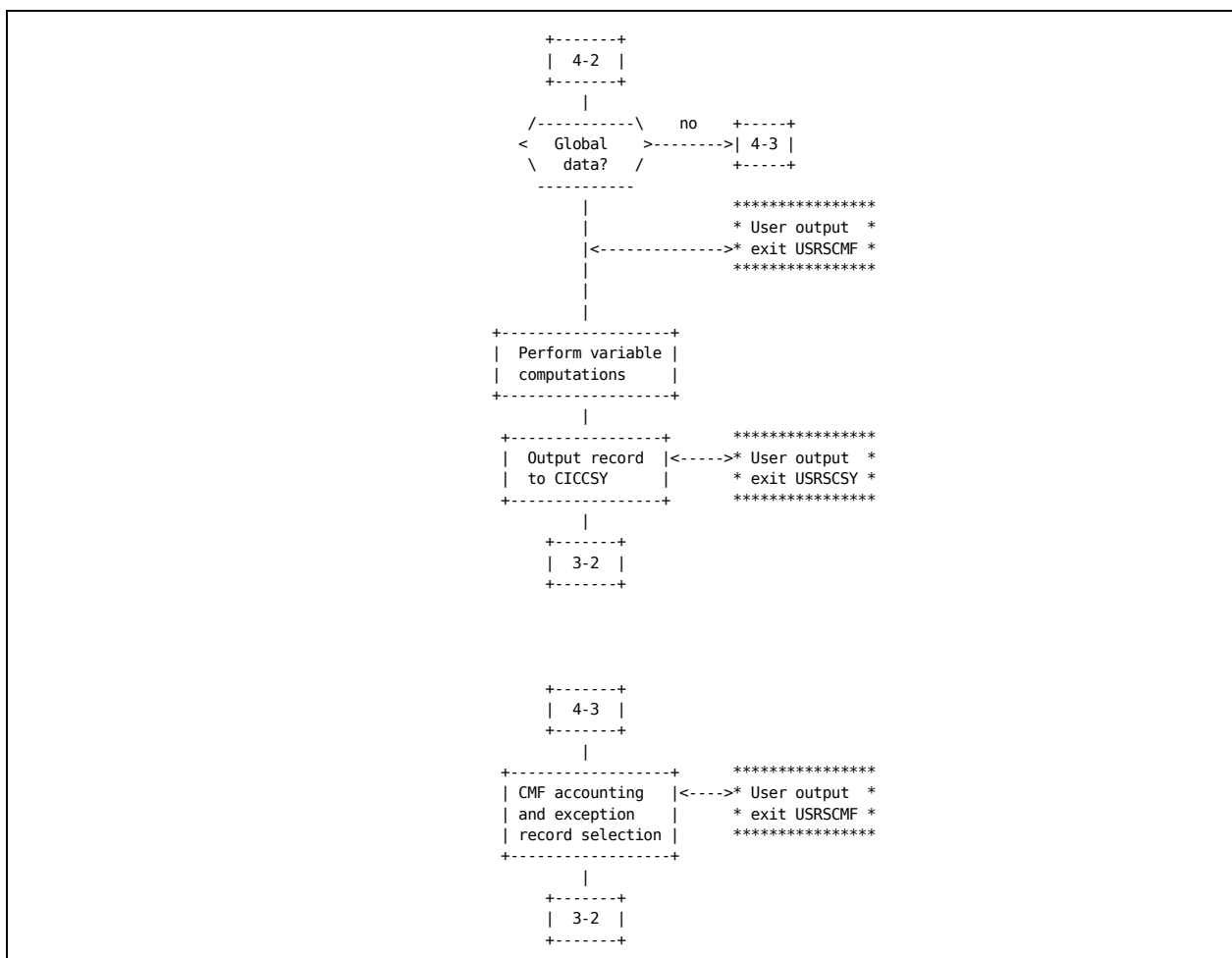


Figure 9-2. Input Raw Data Processing (Part 7 of 13)

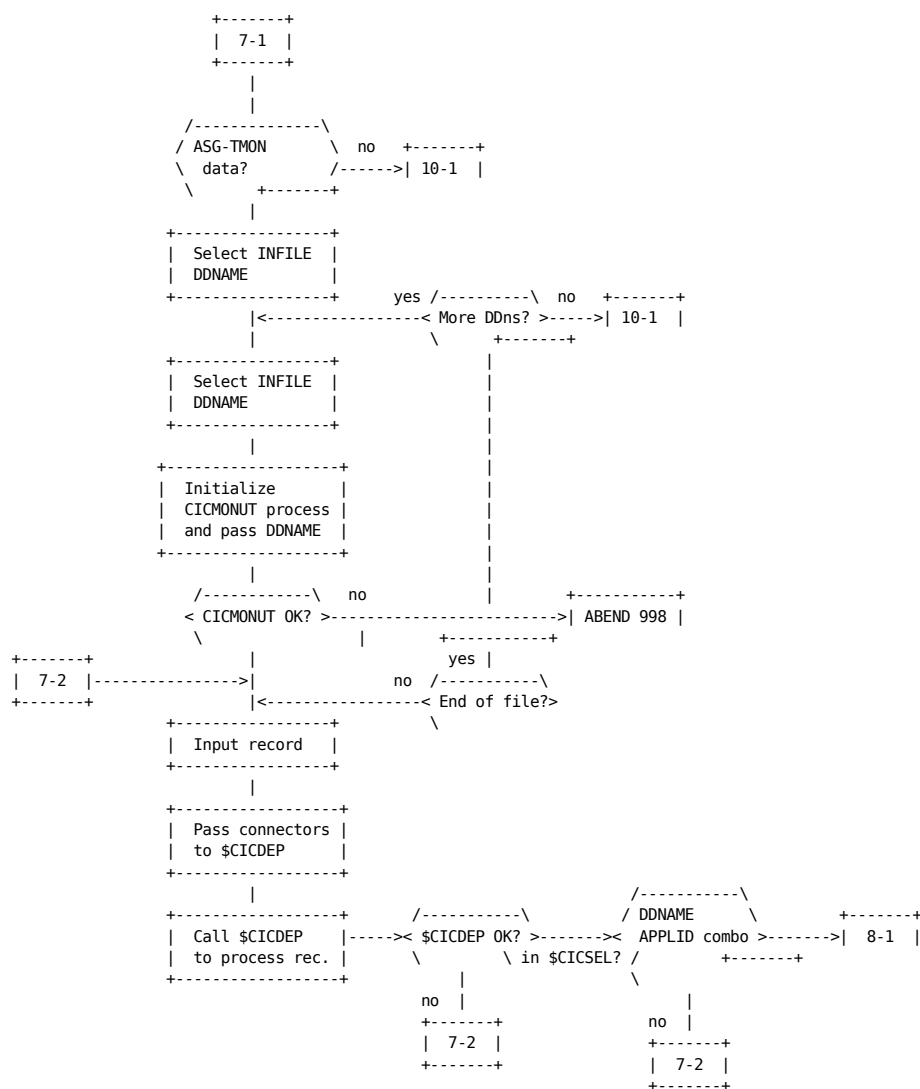
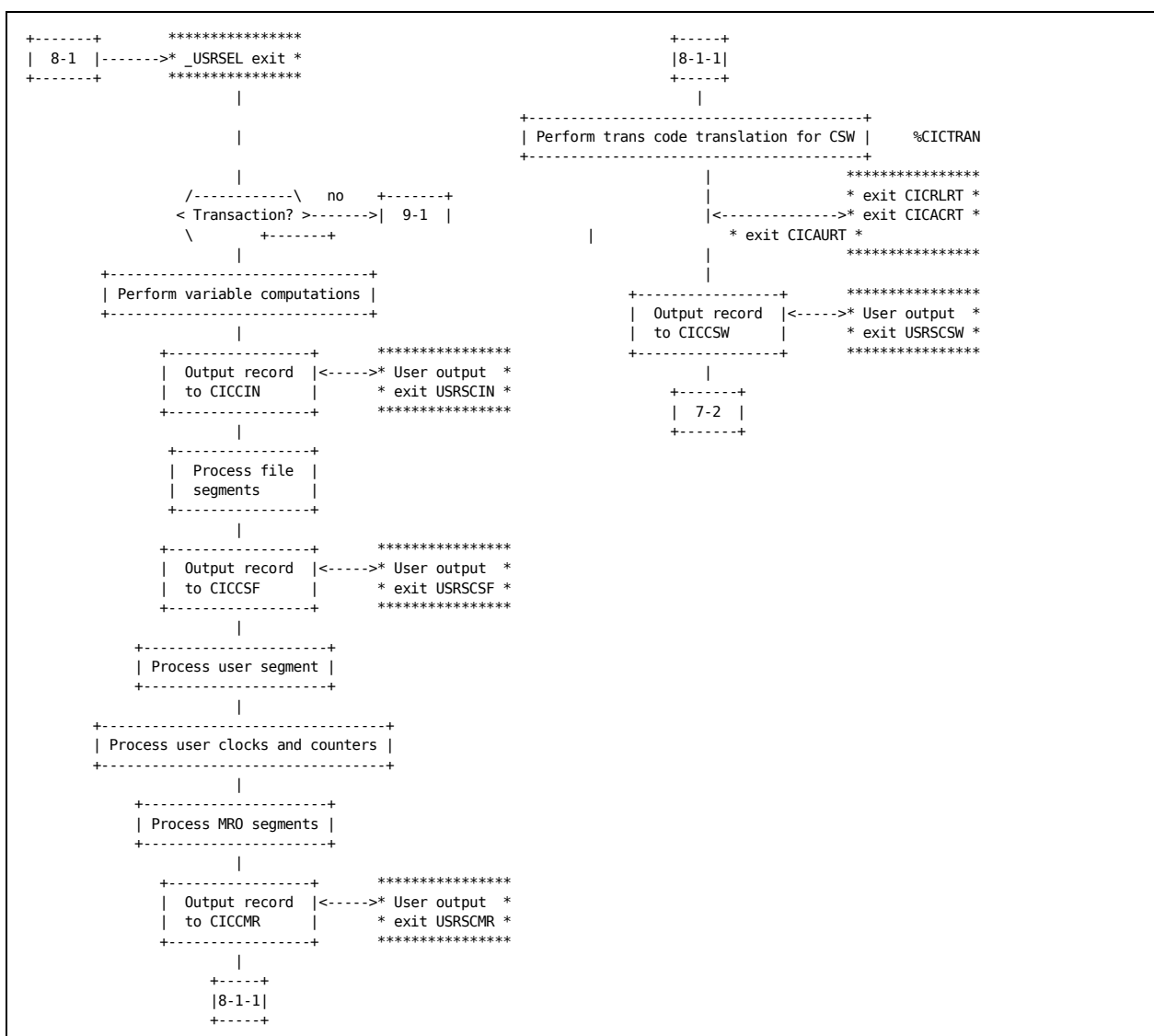


Figure 9-2. Input Raw Data Processing (Part 10 of 13)



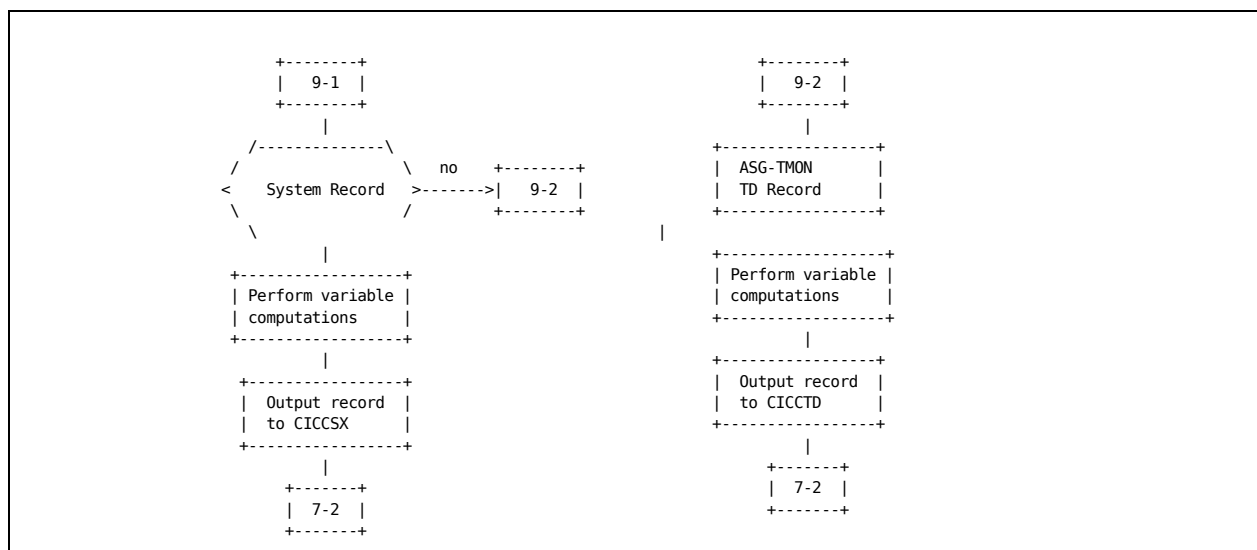


Figure 9-2. Input Raw Data Processing (Part 12 of 13)

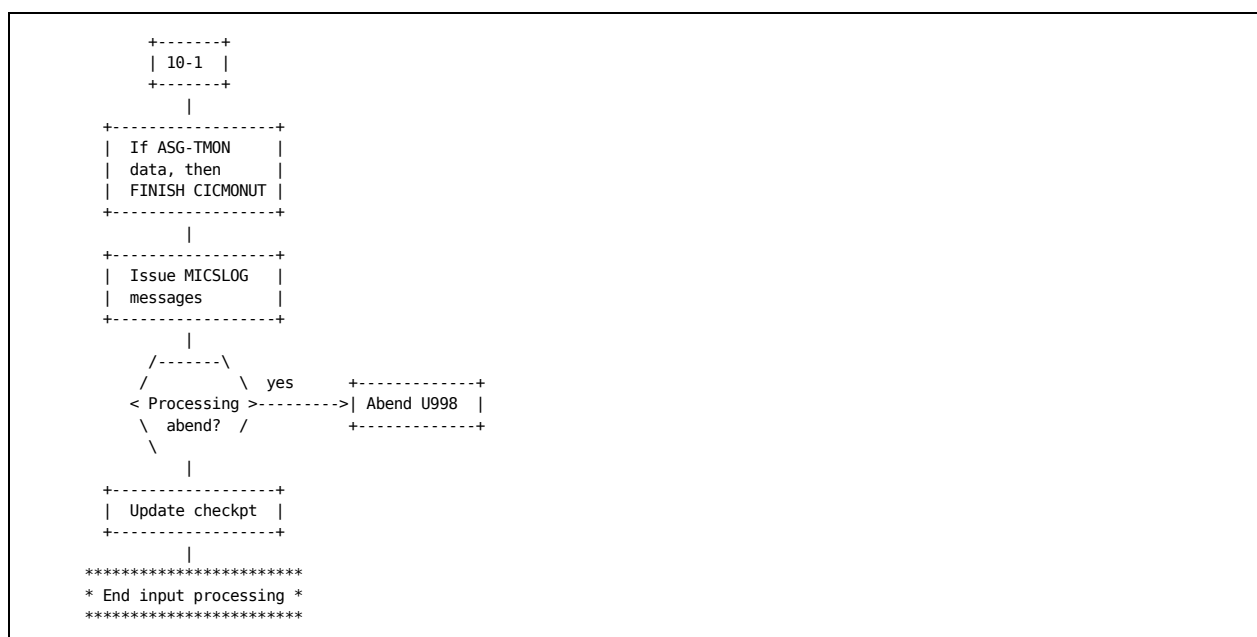


Figure 9-2. Input Raw Data Processing (Part 13 of 13)

9.2.3 Information Area Processing

In the information area processing phase, the work files from the input raw data phase are sorted and processed to produce the cycles for the DETAIL timespan. This phase includes the merge of CICS statistics work files (CICS 3.1), the calculation of CPU delta values for CICS statistics dispatcher data, the calculation of ENDTS and delta values for The Monitor CSM (checkpoint) file, and the execution of the CICMSAC exit if the multi-system account option in prefix.MICS.PARMS(CICOPS) is in effect.

Because ENDTS is not in The Monitor detail accounting system records, it is calculated from the STARTTS of the next record. For the last record in a given CICS region, it is computed from STARTS + TCP dispatched time + KCP dispatched time + JCP dispatched time + user (application) dispatched time + dispatcher wait time. Once the ENDTS values have been calculated, DURATION is derived as ENDTS-STARTTS, the delta values are calculated for each CSM record, and the CSM file is merged with the CMF work CSY file. All this processing occurs in sharedprefix.MICS.SOURCE member #CSYMRG8 for Monitor version 8 data.

The user exits invoked in this phase are listed below:

- o Parameter related exits

CICMSAC - Multi-system accounting exit

Figure 9-3 outlines the flow of this phase.

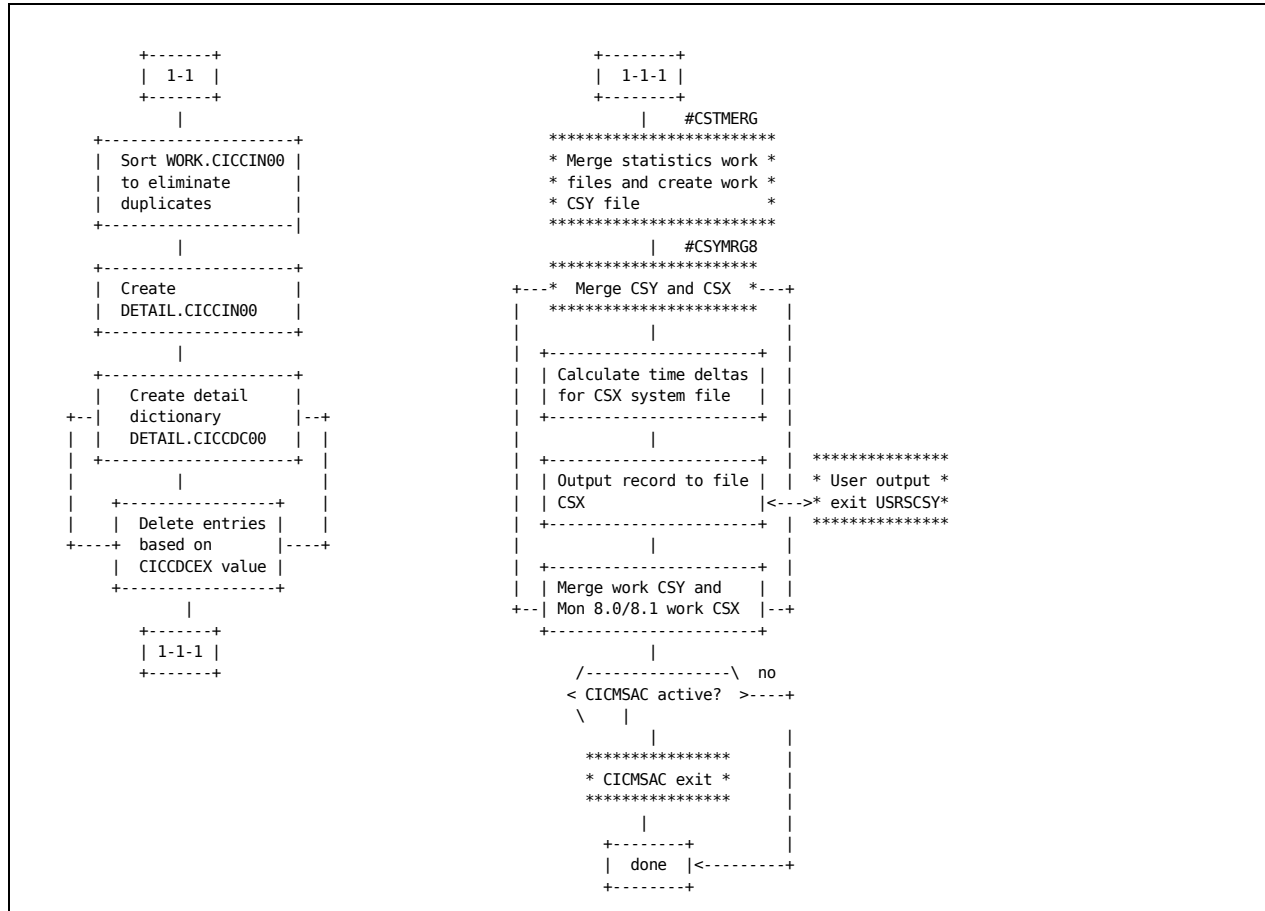


Figure 9-3. Information Area Processing

9.2.4 Data Base Timespan Processing

The data base timespan processing phase performs the required summarization to produce the cycles for the DAYS timespan. Also, where applicable, week-to-date and/or month-to-date cycles are updated.

The user exits invoked in this phase are listed below:

- o Detail timespan exits

Exits are provided for optional creation of the DETAIL timespan for those files that normally do not have a DETAIL timespan.

- USRXCAC - CICCAC DETAIL timespan creation
- USRXCAU - CICCAC DETAIL timespan creation
- USRXCSF - CICCAC DETAIL timespan creation
- USRXCSU - CICCAC DETAIL timespan creation

- o Output exits

During summarization to the DAYS level, these exits may be used for data element manipulation.

- USRSCAC - CICCAC element derivation
- USRSCSU - CICCAC element derivation
- USRSCAU - CICCAC element derivation

- o Parameter-related exits

- CICDCAC - cost derivation
- CICDCSU - cost derivation

Figure 9-4 outlines the flow of this phase.

```

+-----+
| 1-1 |----->* CICCAC *
+-----+
*****
|
/-----\ yes *****
< Detail time >----->* User detail TS *
\ span exit? / * exit USRXCAC *
\ *****
|
+-----+ *****
| Summarize CSW file |<--->* User output *
| at days level. |<--+ * exit USRSCAC *
+-----+ | *****
| |
+-----+ | *****
| Write | +->* Cost derivation *
| DAYS.CICCAC00 | * exit USRDCAC *
+-----+ *****
|
+-----+
| Process updates to |
| month-to-date cycles |
| MONTH.CICCACM0 |
+-----+
|
*****
* CICCSCF *
*****
|
/-----\ yes *****
< Detail time >----->* User detail TS *
\ span exit? / * exit USRXCSCF *
\ *****
|

```

```

+-----+
| Summarize CSF file |
| at days level.    |
+-----+
|
+-----+
| Write              |
| DAYS.CICCSF00     |
+-----+
|
*****
* CICCSU *
*****
|
/-----\ yes *****
< Detail time >----->* User detail TS *----->| 2-1 |
\ span exit? /      * exit USRXCSU *
\                      *****

```

Figure 9-4. Data Base Timespan Processing (Part 1 of 3)

```

+-----+ +-----+
| 2-1 |----->| Perform data |
+-----+      | manipulations |
+-----+      +-----+
|
+-----+ *****
| Summarize CSW file |<---->* User output *
| at days level.    |<--+ * exit USRSCSU *
+-----+          | *****
|
+-----+          | *****
| Write              | +-->* Cost derivation *
| DAYS.CICCSU00     | * exit USRDCSU *
+-----+          | *****
|
+-----+
| Process updates to |
| month-to-date cycles |
| MONTH.CICCSUM0    |
+-----+
|

```

```

*****
*  CICCAU  *
*****

      |
/-----\ yes *****
< Detail time >----->* User detail TS *
\ span exit? /        * exit USRXCAU  *
\          *****

      |
+-----+
| Perform data |
| manipulations |
+-----+

      |
+-----+ *****
| Summarize CSW file |<---->* User output *
| at days level.    |      * exit USRSCAU *
+-----+ *****

      |
+-----+
| Write            |
| DAYS.CICCAU00    |
+-----+

      |
+-----+
| Process updates to |      +-----+
| month-to-date cycles |----->| 3-1 |
| MONTH.CICCAUM0      |      +-----+
+-----+

```

Figure 9-4. Data Base Timespan Processing (Part 2 of 3)

```
+-----+          *****
| 3-1 |----->*  CICC SR/CICCSY  *
+-----+          *****

                        |
+-----+
| Sort WORK.CICCSY00 |
| to eliminate      |
| duplicates.       |
+-----+

                        |
+-----+
| Write              |
| DETAIL.CICCSY00   |
+-----+

                        |
+-----+
| Summarize CSY file |
| at days level.    |
+-----+

                        |
```

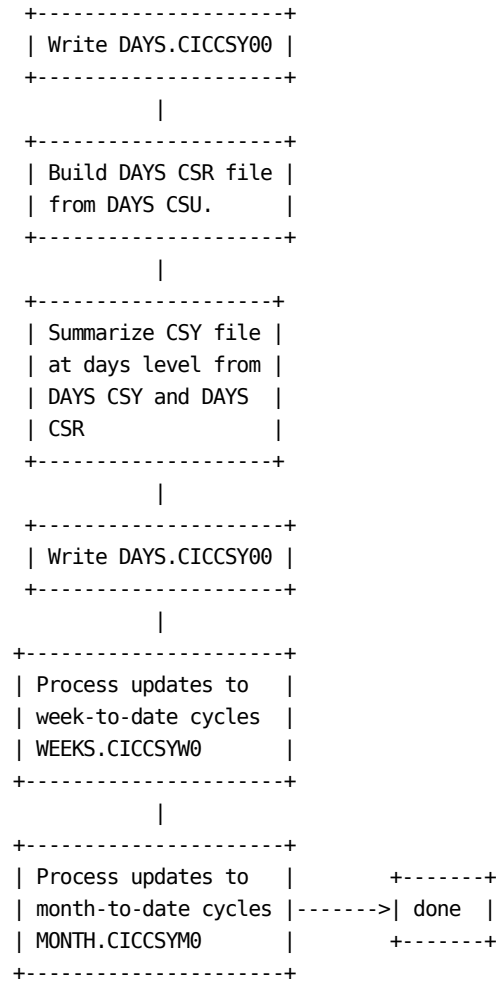



Figure 9-4. Data Base Timespan Processing (Part 3 of 3)

9.2.5 File Aging

The file aging phase deletes the oldest cycles of the DETAIL and DAYS cycles, replacing them with the next oldest and so forth until the new interim cycle (00) becomes the current cycle (01). This phase also replaces the existing week-to-date and month-to-date cycles (00) with update cycles W0 and M0, respectively.

No user exits are invoked during this phase.

Figure 9-5 outlines the flow of this phase.

```

+-----+
|  1-1  |
+-----+
|
+-----+
| Age cycle 00 through |
| specified cycle limit|
+-----+
|
+-----+
| Age cycle 00 through |
| specified file limit |
+-----+
|
+-----+
| Age (replace) cycle  |
| W0 over cycle 00     |
+-----+
|
+-----+
| Age (replace) cycle  |
| M0 over cycle 00     |
+-----+
|
+-----+
| Done  |
+-----+

```

Figure 9-5. File Aging Processing

9.2.6 MICSLOG Operational Report

In addition to the standard report that displays checkpoint and total record processing information, a CICS operational report is produced during step DAY040. The report is produced following the completion of data processing for each input ddname and is a breakdown by ddname.

Each report segment includes record counts in the following areas:

- o Records processed by type
- o CICS file records rejected by user exit
- o CICS file records written

Figures 9-6 and 9-7 give sample reports for CMF and ASG-TMON data.

```

16.17.24 CIC06120I
=====+
16.17.24 CIC06120I INPUT PROCESSING FOR DDNAME INPUTSMF
16.17.24 CIC06120I
16.18.10 CIC06150I TOTAL RECORDS INPUT                      7500
16.18.10 CIC06150I TYPE 110 (CMF) RECORDS INPUT            3000
16.18.10 CIC06150I CMF RECORDS REJECTED BY BASE              0
16.18.10 CIC06154I DICTIONARY RECORDS INPUT                  129
16.18.10 CIC06154I TRANSACTION RECORDS INPUT                47655
16.18.10 CIC06154I TRANSACTION RECORDS REJECTED BY USRSCMF EXIT 0
16.18.10 CIC06154I GLOBAL RECORDS INPUT                      1041
16.18.10 CIC06154I GLOBAL RECORDS REJECTED BY USRSCMF EXIT  0
16.18.10 CIC06154I ACCOUNTING DATA SECTIONS REJECTED BY SELCODE 0
16.18.10 CIC06154I ACCOUNTING RECORDS INPUT                  0
16.18.10 CIC06154I EXCEPTION DATA SECTIONS REJECTED BY SELCODE 0
16.18.10 CIC06154I EXCEPTION RECORDS INPUT                    5
16.18.10 CIC06154I CICCDC00 RECORDS REJECTED BY USRSCDC EXIT 0
16.18.10 CIC06154I CICCIN00 RECORDS REJECTED BY USRSCIN EXIT 0
16.18.10 CIC06154I CICC SW00 RECORDS REJECTED BY USRSCSW EXIT 0
16.18.10 CIC06154I CICC SY00 RECORDS REJECTED BY USRSCSY EXIT 0
16.18.10 CIC06154I CICCDC00 RECORDS WRITTEN                  816
16.18.10 CIC06154I CICCIN00 RECORDS WRITTEN                   40
16.18.10 CIC06154I CICC SW00 RECORDS WRITTEN                47655
16.18.10 CIC06154I CICC SY00 RECORDS WRITTEN                1041
16.18.10 CIC06222I CICS JOURNAL RECORD INPUT                  0
16.18.10 CIC06222I PHYSICAL STATISTICS RECORD INPUT          4000
16.18.10 CIC06222I INCOMPLETE STATISTICS RECORDS INPUT       0
16.18.10 CIC06222I LOGICAL STATISTICS RECORDS INPUT          27854
16.18.10 CIC06222I RECORDS REJECTED BY USRSCST EXIT          0
16.18.10 CIC06222I TYPE 8 STATISTICS RECORDS READ            21
16.18.10 CIC06222I TYPE 9 STATISTICS RECORDS READ            21
16.18.10 CIC06222I TYPE 27 STATISTICS RECORDS READ            21
16.18.10 CIC06222I TYPE 45 STATISTICS RECORDS READ            21
16.18.10 CIC06222I TYPE 48 STATISTICS RECORDS READ            21
16.18.10 CIC06222I TYPE 57 STATISTICS RECORDS READ            21

```

Figure 9-6. CMF Data Operation Report by DDname

```

16.17.24 CIC06120I
=====+
16.17.24 CIC06120I INPUT PROCESSING FOR DDNAME INPUTMON
16.17.24 CIC06120I
16.18.10 CIC06157I TOTAL RECORDS INPUT                      9197
16.18.10 CIC06157I TRANSACTION OR SYSTEM RECORDS IN INPUT  9197
16.18.10 CIC06157I TRANSACTION OR SYSTEM RECORDS REJECTED BY BASE    0
16.18.10 CIC06158I TRANSACTION RECORDS PROCESSED            9005
16.18.10 CIC06158I SYSTEM RECORDS PROCESSED                 192
16.18.10 CIC06158I FILE SEGMENTS IN TRANSACTION RECORDS PROCESSED 45025
16.18.10 CIC06158I MRO SEGMENTS IN TRANSACTION RECORDS PROCESSED    0
16.18.10 CIC06158I CICCIN00 RECORDS REJECTED BY USRSCIN EXIT        0
16.18.10 CIC06158I CICCSCF00 RECORDS REJECTED BY USRSCSF EXIT      0
16.18.10 CIC06158I CICCNR00 RECORDS REJECTED BY USRSCSF EXIT      0
16.18.10 CIC06158I CICCSCW00 RECORDS REJECTED BY USRSCSW EXIT      0
16.18.10 CIC06158I CICCIN00 RECORDS WRITTEN                   1216
16.18.10 CIC06158I CICCSCF00 RECORDS WRITTEN                 45025
16.18.10 CIC06158I CICCNR00 RECORDS WRITTEN                   0
16.18.10 CIC06158I CICCSCW00 RECORDS WRITTEN                 9005
16.18.10 CIC06223I CICCSCY00 RECORDS WRITTEN                 192

```

Figure 9-7. ASG-TMON Data Operation Report by DDname

9.3 Utility Programs

The CICS Analyzer provides utility programs to assist you with problem determination when errors are encountered during daily CICS processing. These utility programs are described in the following sections:

- 1 - Input List Program (CICILIST)
- 2 - Data Dictionary Print Program (CICCDCPT)
- 3 - CICS Standalone Processing Program (CICALONE)
- 4 - CICS GDG Generation Program (CICGDGNN)

9.3.1 Input List Program (CICILIST)

The utility program CICILIST selectively lists the data records from a CMF or ASG-TMON input file. The data is listed in the same format as that produced by the SAS LIST statement, which lists the data lines in the SAS log following a ruler indicating column positions. Each data line is also accompanied by its hex representation. This format is very helpful when input fields have to be examined to determine the cause of any error messages issued during the CA MICS daily CICS processing.

The JCL for CICILIST resides in prefix.MICS.CNTL(CICILIST). It is generated when you install the CA MICS Analyzer Option for CICS in a database unit. Before running this utility program, you must supply the correct input data set name and the following program control statements:

- o For the input file to be listed, a control statement is required that names the ddname of the input data source. The format is:

```
ddname xxx
```

where:

```
ddname = JCL ddname where the input file is defined.  
xxx     = A three-character input data source. Valid  
          values are CMF, MON, or TCE.
```

- o For each record to be listed, a control statement is required indicating the record number (one statement per record number). For example, to list record numbers 30 to 33, the following statements are required:

```
30  
31  
32  
33
```

Sample JCL is shown below to list two input records from CMF input. The record numbers are 1820 and 2050.

```
//SAMPLE JOB ..  
//CICILIST EXEC MICSNDBx  
//SYSIN DD DISP=SHR,DSN=sharedprefix.MICS.INCLLIB(CICILIST)  
//INDATA DD DISP=SHR,DSN=your.cmf110.data.file  
//CONTROL DD *  
INDATA CMF  
1820  
2050  
//
```

9.3.2 Data Dictionary Print Program (CICCDCPT)

The utility program CICCDCPT prints the CA MICS CICS Dictionary File (DETAIL.CICCDC01) using SAS PROC PRINT. The report is sorted by dictionary APPLID (CDCAPPL), CICSID, end timestamp (ENDTS), dictionary date of last reference (CDCLREF), and dictionary class of data (CDCCLASS). This information is useful when you are researching CA MICS daily CICS processing errors that are related to CMF data dictionary records.

The JCL for CICCDCPT resides in prefix.MICS.CNTL(CICCDCPT). It is generated when you install the CICS Analyzer in a database unit. To run this utility program, simply submit the JCL. There are no required program control statements.

9.3.3 CICS Standalone Processing Program (CICALONE)

The utility program CICALONE processes CMF or ASG-TMON data and creates work files that you can use to analyze CICS activity in detail without affecting the content of the CA MICS database. The utility can be used to analyze data from one or more regions, but only from one of the above input data sources.

The work files created by CICALONE are identical in content to the standard CICS files at the DETAIL timespan. They allow you to perform on-demand analysis of CICS at the transaction level outside of normal CA MICS operation. However, to run this utility program, you must have installed the CA MICS Analyzer Option for CICS in a unit.

Note: CICALONE does not have an automated interface to CA SMF Director. CICALONE is intended to execute independently of a unit database, so there is no way to programmatically identify eligible input files. Therefore, you must continue to manually define your SMF input using the INPUTSMF DD as described below.

The JCL for CICALONE is in prefix.MICS.CNTL(CICALONE). It is generated when you install the CA MICS Analyzer Option for CICS in a database unit. Before submitting this JCL to run the utility program, you must do the following:

1. Supply the name of your input data set.
2. Edit prefix.MICS.PARMS(CICALONE) to specify the SAS macros required by this program.
3. Supply your own SAS code to either analyze the work files or store the files in permanent SAS files. By default, CICALONE writes the files to the WORK DD, which is deleted upon completion of the job. To review the logic of CICALONE, see sharedprefix.MICS.SOURCE(CICALONE).

The following shows the generated JCL for CICALONE:

```
// JOB card
//CICALONE EXEC MICSSHRx
//WORK      DD UNIT=SYSDA,SPACE=(TRK,(500,200))
//INPUTSMF DD DSN=your input data set
//SYSIN DD DISP=SHR,DSN=sharedprefix.MICS.SOURCE(CICALONE)
```

You must update the INPUTSMF DD statement to supply the name of your input data set. If you are processing ASG-TMON data, you must change the input ddname to match the ddname that you

supply in the ddname macro in prefix.MICS.PARMS(CICALONE).

The macros defined in prefix.MICS.PARMS(CICALONE) are described below. Each macro contains a default definition. You must change the default definition if it does not match your processing requirement.

DDNAME - This defines the ddname associated with the input data. If the input is in SMF format, then code INPUTSMF. Otherwise, specify any other one-to-eight character ddname.

Default definition:

```
%MACRO DDNAME;
  INPUTSMF
%MEND;
```

MONFLGS - This identifies ASG-TMON input source. Set the appropriate flag to 1 based on the input. Only one input data source can be processed. For example, if the input is ASG-TMON, then set MONMON to 1, CMFMON, TCEMON to 0.

Default definition:

```
%MACRO MONFLGS;
  CMFMON = 1;      /*CMF or OMEGAMON          */
  MONMON = 0;      /*ASG-TMON                */
  TCEMON = 0;      /*ASG-TMON for CICS TS 2.0+ */
%MEND;
```

CICIPCV - This defines the input processing control variables required by this utility program. You must modify the default values to match your data center specifications. To process multiple regions, repeat the following control variables, where nn = 01 to 99, for each region. For example, to process 3 regions, specify values for MSYS01, MSYS02, MSYS03, MCIC01, MCIC02, MCIC03 and so on.

The control variables are as follows:

MSYSnn - The original (SMF) system ID. Specify "CICS" for non-SMF input.
 MCICnn - The CICS ID assigned by you.
 MMONnn - The monitor type. CMF or ASG-TMON.
 MGMTnn - The GMT offset. Specify the value as a

SAS time constant.
MREGnn - The VTAM APPLID.
MIVLnn - The CMF recording interval. Specify the value as a SAS time constant.
MSELnn - The record selection code. Always code "0".

Default definition:

```
%MACRO CICIPCV;  
  MSYS01 = 'PROD';          /* ORGSYSID */  
  MCIC01 = 'CIC1';          /* CICSID */  
  MMON01 = 'CMF';           /* MONITOR TYPE */  
  MGMT01 = '6:00:00'T;      /* GMT OFFSET */  
  MREG01 = 'CICA120';       /* APPLID */  
  MIVL01 = '0:15:00'T;      /* INTERVAL */  
  MSEL01 = '0';             /* SELCODE */  
%MEND;
```

Example for selecting two regions:

```
%MACRO CICIPCV;  
  MSYS01 = 'PROD';          /* ORGSYSID */  
  MCIC01 = 'CIC1';          /* CICSID */  
  MMON01 = 'CMF';           /* MONITOR TYPE */  
  MGMT01 = '6:00:00'T;      /* GMT OFFSET */  
  MREG01 = 'CICA120';       /* APPLID */  
  MIVL01 = '0:15:00'T;      /* INTERVAL */  
  MSEL01 = '0';             /* SELCODE */  
  
  MSYS02 = 'PROD';          /* ORGSYSID */  
  MCIC02 = 'CIC2';          /* CICSID */  
  MMON02 = 'CMF';           /* MONITOR TYPE */  
  MGMT02 = '6:00:00'T;      /* GMT OFFSET */  
  MREG02 = 'CICA100';       /* APPLID */  
  MIVL02 = '0:15:00'T;      /* INTERVAL */  
  MSEL02 = '0';             /* SELCODE */  
%MEND;
```

- REGIONS - This specifies the total number of regions being processed by CICALONE. It must match the number of regions defined in the CICIPCV macro. The default is 1.
- CICTRAN - This assigns a value to the TRANTYPE data element. This is a null macro by default. You can modify the macro to supply your logic or use the member CICRLRT in prefix.MICS.PARMS to assign TRANTYPE.

Example:

```
%MACRO CICTRAN;
  IF TRANCOD EQ 'CE' OR TRANCOD EQ 'CS' THEN
    TRANTYPE='X';
  ELSE IF TRANCOD EQ 'A' THEN TRANTYPE='S';
  ELSE IF TRANCOD EQ 'B' THEN TRANTYPE='L';
  ELSE TRANTYPE='M';
%MEND;
```

or

```
%MACRO CICTRAN;
  %INCLUDE PARM(CICRLRT);
%MEND;
```

CICACRT - This assigns values to the CICS account codes. By default, this macro calls the member CICACRT in sharedprefix.MICS.PARMS to assign CICACTX. You can override the CICACRT macro by supplying your own code.

Example:

```
%MACRO CICACRT;
  CICACTX1=SUBSTR(TRANCOD,1,2);
  CICACTX2=SUBSTR(TRANCOD,3,2);
%MEND;
```

CICAURT - This assigns values to the CICS application unit CICAPU. By default, this macro calls the member CICAURT in prefix.MICS.PARMS to assign CICAPU. You can override the CICAURT macro by supplying your own code.

Example:

```
%MACRO CICAURT;
  IF TRANCOD EQ : 'C' THEN CICAPU ='OVHD    ';
  ELSE CICAPU=TRANCOD;
%MEND;
```

USYSID - This assigns SYSID values. By default, this macro assigns SYSID based on the definitions in prefix.MICS.PARMS(SYSID). You can override the USYSID macro by supplying your own code.

Example:

```
%MACRO USYSID;  
  IF ORGSYSID EQ 'X090' THEN DO;  
    SYSID='X090';  
    MIPS=17.5;  
    CPUMODEL='3090';  
  END;  
%MEND;
```

- CMFINP - This defines the INFILE statement for the CMF input data set. Do not change this macro definition.
- MONINP - This defines the INFILE statement for ASG-TMON input data set. Do not change this macro definition.
- TCEINP - This defines the INFILE statement for ASG-TMON for CICS TS 2.0 and above input data set. Do not change this macro definition.

After you have reviewed and completed the macro definitions in `prefix.MICS.PARMS(CICALONE)`, you should either supply the code necessary to analyze the work files created by the CICALONE program, or store the work files in a permanent SAS database for subsequent analysis.

To add code to the CICALONE program to analyze the work files, you can simply concatenate your code to the `SYSIN DD` statement in the CICALONE JCL, as follows:

```
// JOB card  
//CICALONE EXEC MICSSHRx  
//WORK DD UNIT=SYSDA,SPACE=(TRK,(500,200))  
//INPUTSMF DD DSN=your input data set  
//SYSIN DD DISP=SHR,DSN=sharedprefix.MICS.SOURCE(CICALONE)  
// DD DISP=SHR,DSN=user.source(mycode)
```

To store the work files in a permanent SAS database for later analysis, you must add a `DD` statement to the CICALONE JCL for the SAS database. You must also store the work files in the SAS database. To store the files, you can either modify `sharedprefix.MICS.SOURCE(CICALONE)` to output the data directly to your SAS database during file creation, or you can concatenate your own logic (as shown in the above JCL) to save the work files after they have been created by CICALONE. The following work files are created by CICALONE:

USERCSW - detail CICS transaction file

USERCSY - detail CICS system file for CMF
USERCSM - detail CICS system file for ASG-TMON
USERCSF - detail CICS file segment file for ASG-TMON
USERCIN - detail CICS incident file

9.3.4 CICS GDG Generation Program (CICGDGGN)

The utility program CICGDGGN defines the generation data group (GDG) for storing the `DETAIL.CICCSU01` (CICS User Activity) file on tape as well as the IBM 110 STATISTICS records.

The creation of the `DETAIL.CICCSU01` file on tape is controlled via the `TAPECSU` keyword in `CICOPS`. The creation of the STATISTICS records on tape is controlled via the `STATCOPY` keyword in `CICOPS`.

TAPECSU

By default, the GDG has 7 generations, and is named `tapeprefix.tMICS.DETAIL.CICCSU`. `Tapeprefix` is the tape prefix that you have defined in `prefix.MICS.PARMS(JCLDEF)` and `tMICS` is the `MICSLEVEL` associated with the `tapeprefix` keyword. At this time, the only CA MICS file written to tape during the DAILY update is the `DETAIL.CICCSU01` file.

To customize the specifications of the `DETAIL.CICCSU01` file such as the number of generations, see sections 7.3.2.12.2/3/4 of this guide.

For more information on writing `DETAIL.CICCSU01` to tape during the CA MICS daily update, refer to section 7.3.2 of this guide for a discussion of `prefix.MICS.PARMS(CICOPS)`, section 10.1.4 for the comprehensive checklist to activate the `DETAIL.CICCSU` file on tape, and section 10.2.3 for information on the `USRTCSU` exit, which enables you to limit the amount of data written to the `DETAIL.CICCSU` tape file.

STATCOPY

For information on writing the IBM 110 STATISTICS records to tape, see section 7.3.2.4. For information on the use of the `USRSTAT` exit, see section 10.2.3.

Chapter 10: MODIFICATION

The CA MICS CICS Analyzer is tailored primarily through the use of options and parameters and sometimes through the use of exits. Most of the customization is done by following the standard procedures defined by the CA MICS Systems Modification Guide (SMG). However, for some types of modifications, such as activating CICS files in the DETAIL timespan, special procedures are required. This chapter does not reiterate the details already provided by the SMG. Instead, it provides checklists to show you how to customize the CICS Analyzer, and describes the CICS Analyzer exits that are available during daily update processing.

This section contains the following topics:

[10.1 Checklists](#) (see page 727)

[10.2 User Exits](#) (see page 773)

10.1 Checklists

This section provides checklists for tailoring the CICS Analyzer. The following topics are discussed:

- 1 - Alter Derived Element Support
- 2 - Add or Change Data Elements in a File
- 3 - Add or Modify CICS Account Codes
- 4 - Alter the Active Timespans for a File
- 5 - Activate Archive for a File
- 6 - Add a New File
- 7 - Add or Delete a CICS Region in Daily Processing
- 8 - Change the DDnames of Input Data
- 9 - Activate the CICS File/DBD Activity File (CICCSF)
- 10 - Change Retention Period of the CICCDC File
- 11 - Change the Number of Work Files for DAY040
- 12 - Process CMF User Clocks, Counters, and User Area
- 13 - Activate the CICS MRO Activity File (CICCMR)
- 14 - Convert to a Different Data Source
- 15 - Process Additional Non-USER Areas in CMF
- 16 - Enable Internal Step Restart
- 17 - Implement Incremental Update

10.1.1 Alter Derived Element Support

Derived elements, such as averages and percentages, can either be kept in the data base or computed when the data is required for reporting. Computing these elements only when needed can reduce the amount of DASD required for the data base.

CA MICS provides a global option for derived element support in `sharedprefix.MICS.PARMS(CPLXDEF)`. This option applies to all files in the CA MICS Data Base. You can override this option at the product level or at the file level. For example, you may wish to compute data elements at reporting time for the CICS Analyzer, but keep the computed data elements for all other CA MICS products in the data base. Or, you may wish to compute data elements for the CICS User Activity File at reporting time, but keep the computed data elements in the CA MICS Data Base for the rest of the CICS files.

Follow the checklist below to override the options in `sharedprefix.MICS.PARMS(CPLXDEF)`.

```
*****
*                                     *
*  CHECKLIST 1 - OVERRIDE DERIVED ELEMENT SUPPORT  *
*                                     *
*****
```

- ___ 1. Save a copy of `sharedprefix.MICS.GENLIB(CICGENIN)` in `sharedprefix.MICS.LOCALMOD.CNTL` for recovery purposes.
- ___ 2. To override derived element support at the product level, review the description of the `COMPOPT` statement in `sharedprefix.MICS.GENLIB(CICGENIN)` provided by Section 4.2.4.1.3 of the System Modification Guide (SMG). To override the option at the file level, review the description of the `FOPT` statement in `sharedprefix.MICS.GENLIB(CICGENIN)` provided by Section 4.2.4.2.2 of the SMG.
- ___ 3. Update the `COMPOPT` and/or `FOPT` statements in `sharedprefix.MICS.GENLIB(CICGENIN)`. Use `IEBUPDTE` to apply the change.
- ___ 4. Regenerate the CICS Analyzer by submitting the job in `sharedprefix.MICS.CNTL(CICCGEN)`.

10.1.2 Add or Change Data Elements in a File

You can add new data elements to any of the CIC Information Area files. You can also change the definitions associated with existing data elements, such as data element type, length, and label. In addition, you can activate or deactivate data elements in a specific timespan or in all timespans. Note that adding or changing data elements may affect your CA MICS Accounting and Chargeback files.

For each data element defined to the CA MICS CICS Analyzer, there is a corresponding NAME or NAMX statement in `sharedprefix.MICS.GENLIB(CICGENIN)`. NAME statement indicates that the data element is an essential data element, which means it is always kept in the CA MICS Data Base. NAMX statements denote nonessential data elements, which, depending upon the definition of the ESSENTIAL option in `sharedprefix.MICS.PARMS(CPLXDEF)`, may be excluded from the CA MICS Data Base.

Both NAME and NAMX statements define the data element name, the cluster code that associates the data element with a specific data source, the timespans in which the data element is active, and the 40-byte data element label.

For derived data elements, you can also supply their computations or expressions using the EXP statements. If the computations depend upon the presence of other data elements, then a DEPEND statement is also needed. Therefore, a NAME or NAMX statement, followed by DEPEND and one or more EXP statements are typically defined in `CICGENIN` for a derived data element.

The NAME and NAMX statements are preceded by a TYPE statement, which defines the data element type, and length and output formats in different timespans.

The TYPE, NAME, and NAMX statements are associated with a FILE statement, which defines the file name, the file type, the timespans in which the file is active, and the file label.

To add a data element or change the attributes associated with a data element, follow the checklist provided below.

```
*****
*
* CHECKLIST 2 - ADD OR CHANGE ELEMENTS IN A FILE *
*
*****
```

- ___ 1. Save a copy of sharedprefix.MICS.GENLIB(CICGENIN) in sharedprefix.MICS.LOCALMOD.CNTL for recovery purposes.
- ___ 2. Review the descriptions of data element definition statements provided by Section 4.2.4.3 of the System Modification Guide.
- ___ 3. Add or modify TYPE, NAME, NAMX, DEPEND, or EXP statements in sharedprefix.MICS.GENLIB(CICGENIN). Ensure these statements follow the appropriate FILE statements for the file in which you are adding or modifying the data elements. In addition, any changes or additions made to the CSU, CAU, or CAC files must also be made in the CSW file at the DETAIL timespan. The CSW file is a work file used during daily CICS Analyzer processing (DAY040 step of DAILY job) to build other files. Use IEBUPDTE to apply the change.
- ___ 4. Regenerate the CICS Analyzer by submitting the job in sharedprefix.MICS.CNTL(CICCGEN).

10.1.3 Add or Modify CICS Account Codes

You can add, delete, or change the definitions of the CICS account codes. However, changing the account code structure affects not only the content of the CA MICS Data Base, but also your ability to produce consistent reports, as the CICS files will be summarized differently afterwards. Your CA MICS Accounting and Chargeback files will also be affected. Therefore, you should implement the new account code structure when it will produce the least impact on your CA MICS system, such as during the testing phase or at the beginning of an accounting month.

Use Checklist 3.1 to alter the CICS account code structure (CICACCT). If you only wish to change the CICS Account Code Derivation Exit (CICACRT), follow Checklist 3.2.

```
*****
*                                     *
*  CHECKLIST 3.1 - ADD OR MODIFY CICS ACCOUNT CODE  *
*                                     *
*****
```

- ___ 1. Save a copy of sharedprefix.MICS.PARMS members CICACCT and CICACRT (if necessary) in sharedprefix.MICS.LOCALMOD.CNTL for recovery purposes.
- ___ 2. Modify sharedprefix.MICS.PARMS(CICACCT) to add, delete, or modify an existing account code definition. See Section 7.2.2 of this guide for detailed description of CICACCT.
- ___ 3. Submit sharedprefix.MICS.CNTL(CICCGEN).
- ___ 4. Modify the CICS Account Code Derivation Routine in sharedprefix.MICS.PARMS(CICACRT) to derive the account codes. Changes to this member take effect immediately, as this code is included in the next DAILY execution.
- ___ 5. If the Multisystem Accounting Exit (CICMSAC) is active, then examine prefix.MICS.PARMS(CICMSAC) to ensure that the exit is correct. Modify this exit as necessary.

Note: the CICMSAC exit is active if the keyword MSACCOUNT is specified in prefix.MICS.PARMS(CICOPS).

- ___ 6. Retrofit the 00 cycles of the CICCISU and CICCAC files in the WEEKS, MONTHS, and YEARS (if active) time-spans if you added or deleted an account code level or shortened the account code length. Use the "System Modifications Guide" section 6.3 to retrofit the database.

```
*****
*
* CHECKLIST 3.2 - MODIFY ACCOUNT CODE DERIVATION EXIT *
*
*****
```

- ___ 1. Modify the CICS Account Code Derivation Routine in sharedprefix.MICS.PARMS(CICACRT) to derive the account codes. Changes to this member take effect immediately, as this code is included in the next DAILY execution. No generation is required afterwards.

10.1.4 Alter the Active Timespans for a File

For each CICS information area file, there is a corresponding FILE statement in sharedprefix.MICS.GENLIB(CICGENIN). The FILE statement defines the file name, the file type, the file status (active or inactive) in each timespan, and the file status for archive (audit and history collection).

```
*****
*
* Note: If incremental update is active, be sure to
* review the important considerations in Section
* 10.1.4.1 of this guide before applying the
* TAPEfff option or USRXfff exits.
*
*****
```

This section includes three checklists:

- o Checklist 4.1, steps for activating the DETAIL timespan for the CICCSCU file on tape.
- o Checklist 4.2, steps for activating the DETAIL timespans for the CICCAC, CICCACU, and CICCSCU files on disk.
- o Checklist 4.3, steps for altering active timespans.

Notes: Do NOT activate the CICS Incident File (CICICIN) in higher timespans because its data is only meaningful in the DETAIL timespan.

Do NOT change in any way the file status for the CICS Dictionary File (CICCDC). This is a special file required to process CMF data and is not used for reporting purposes.

To capture DETAIL timespan data for the CICS User Application Count (CICCAC), CICS Application Unit Activity (CICCACU), or CICS User Activity (CICCSCU) files, you must code a USRXfff exit, where fff is the three-character file ID. Refer to Section 10.2.3 for more information about these exits.

To activate the CICS File/DBD Activity File (CICCSF), refer to Section 10.1.9.

To activate the CICS MRO Activity File (CICCMR), refer to Section 10.1.13.

```
*****
```

```
*
* CHECKLIST 4.1 - ACTIVATE DETAIL TIMESPAN
* of CICCSCU file on CA MICS GDG tape
*
* If incremental update is active, be sure to
* review the considerations in Section 10.1.4.1
* of this guide.
*
*****
```

- ___ 1. Review the descriptions of the FILE statement as provided by Section 4.2.4.2 of the System Modification Guide (SMG).

- ___ 2. To activate and deactivate data elements in the DETAIL timespan, update the NAME/NAMX statements in sharedprefix.MICS.GENLIB(CICGENIN).

The TAPECSU feature uses the KEEP list (macro) generated from the CSWCSU file. Any file tailoring (activation/deactivation of elements) should be done to the CSWCSU file in sharedprefix.MICS.GENLIB(CICGENIN).

Note: Deactivating elements in the CSWCSU file makes them unavailable to other files as well, such as CICCAG, CICCAC, and CICCSCU at DAYS and higher timespans.

Several types of data elements are not meaningful in the DETAIL timespan. Among these are elements related to averages, minimums, and maximums. Elements that are based on TRANTYPE, such as response time distributions, are also meaningless, because they are not calculated during detail input processing. Turning off these elements in the DETAIL file will save DASD space.

- ___ 3. Skip this step if you did not perform Step 2. Otherwise, submit sharedprefix.MICS.CNTL(CICCGEN).

```
*****
* Repeat the following steps for each database *
* unit containing the CICS Analyzer.
*****
```

- ___ 4. Edit prefix.MICS.PARMS(CICOPS) and add the following statement anywhere in the member:

TAPECSU

- ___ 5. Submit prefix.MICS.PARMS(CICPGEN).
- ___ 6. Edit prefix.MICS.PARMS(JCLGENU) so that it reads:

CICGDGGN DAILY
- ___ 7. Submit prefix.MICS.CNTL(JCLGENU) to regenerate the JCL for DAILY and CICGDGGN.
- ___ 8. Submit prefix.MICS.CNTL(CICGDGGN) to define the tape GDG. See Section 9.3.4 for additional documentation regarding the CICGDGGN job.
- ___ 9. (Optional) Edit prefix.MICS.USER.SOURCE(#CICEXIT) and define the USRTCSU exit if you want to limit the amount of output that will be written to the CICCSU file.

```
*****
*
* CHECKLIST 4.2 - ACTIVATE DETAIL TIMESPANS
* FOR CICCAC, CICCAG, and CICCSU files on disk
*
* If incremental update is active, be sure to
* review the considerations in Section 10.1.4.1
* of this guide.
*
*****
```

- ___ 1. Make a backup copy of the prefix.MICS.USER.SOURCE(#CICEXIT) member for each CICS unit to be modified.
- ___ 2. Review the descriptions of the FILE statement as provided by Section 4.2.4.2 of the System Modification Guide (SMG).
- ___ 3. Update the FILE statement in sharedprefix.MICS.GENLIB(CICGENIN) for each file to be activated in the DETAIL timespan. Change the file status from N to Y. Use IEBUPDTE to apply the change. See Section 6.2.2.2 in the SMG for an example.

Note: In order to activate the CICCAC and/or CICCAG files in the DETAIL timespan, they MUST also be active in the DAYS timespan. Additionally, any

elements you require MUST be active in the DAYS timespan of the CICC SW and or CICCAC/CICCAU files even if you only require them at the DETAIL timespan.

- ___ 4. Update the NAME/NAMX statements in the CICGENIN member of sharedprefix.MICS.GENLIB to activate or deactivate data elements in the DETAIL timespan. Elements that are only in the CICC SW file may be added to the CICCAU and/or CICCAC files. To do this, review how the element is defined in the CICC SW file and then define it identically in the CICCAU and/or CICCAC file. The only difference in the definitions should be the data element name. For example, CSUINCH is named CACINCH if it is being added to the CICCAC file. Again, any element required in the CICCAC and/or CICCAU files MUST be active in the DAYS timespan of the CICC SW file and the requested output files CICCAC/CICCAU even if you only require them at the DETAIL timespan.

Several types of data elements are not meaningful in the DETAIL timespan. Among these are elements related to averages, minimums, and maximums. Elements that are based on TRANTYPE, such as response time distributions, are also meaningless, because they are not calculated during detail input processing. Turning off these elements in the DETAIL file will save DASD space requirements.

If you activate an element in the CICC SW file, you MUST activate the element in the CSWCSU file as well.

If you deactivate an element in the CICC SW file AND that element is also inactive or does not exist in the CICCAU and CICCAC files, you should deactivate the element in the CSWCSU file to save work space and DAY040 run time.

- ___ 5. Submit sharedprefix.MICS.CNTL(CICCGEN).
- ___ 6. If the CICCAC file is being activated in the DETAIL timespan, then make the following updates to the _USRXCAC macro in prefix.MICS.USER.SOURCE(#CICEXIT):

Note: See examples under Step 8 below for additional coding techniques.

```
DATA &CICX..CICCAC00(%CACFILE(OP=FILEOPTS,TS=DETAIL));
```



```
SET %%WRKMERG(WRK=CSW,CCC=CIC,R=CACREN);
RUN;
```

- ___ 7. If the CICCAU file is being activated in the DETAIL timespan, then make the following updates to the _USRXCAU macro in prefix.MICS.USER.SOURCE(#CICEXIT):

Note: See examples under Step 8 below for additional coding techniques.

```
DATA &CICX..CICCAU00(%%CAUFILE(OP=FILEOPTS,TS=DETAIL));
SET %%WRKMERG(WRK=CSW,CCC=CIC,R=CAUREN);
RUN;
```

- ___ 8. If the CICCSU file is being activated in the DETAIL timespan, then make the following updates to the _USRXCSU macro in prefix.MICS.USER.SOURCE(#CICEXIT):

```

/*****
/****          EXAMPLE ONE          ****
/****          ****
/**** Write DETAIL.CICCSU to disk in compressed format ****
/**** (if file is generated with compression on)          ****
/**** using the generated KEEP macro (created from          ****
/**** sharedprefix.MICS.GENLIB(CICGENIN) CSWCSU file        ****
/****          ****
/****          ****

```

```
DATA &CICX..CICCSU00 (KEEP=%%CSWKEEP(TS=DETAIL)
                      COMPRESS=%%INTRST(CCC=CIC));
SET %%WRKMERG(WRK=CSW,CCC=CIC);
RUN;
```

```

/*****
/****          EXAMPLE TWO          ****
/****          ****
/**** Write DETAIL.CICCSU to disk FORCING to compressed ****
/**** format regardless of compression option, and still****
/**** using the generated KEEP macro from CSWCSU file.    ****
/**** If compression is not desired, omit the COMPRESS=    ****
/**** option keyword, or change YES to NO.                 ****
/****          ****
/****          ****

```

```
DATA &CICX..CICCSU00 (KEEP=%%CSWKEEP(TS=DETAIL)
                      COMPRESS=YES) ;
SET %%WRKMERG(WRK=CSW,CCC=CIC);
RUN;
```

```

/*****
/****          EXAMPLE THREE          ****
/****          ****

```

```

/****                                     ****/
/**** Write DETAIL.CICCSU to disk and provide your own ****/
/**** KEEP list, again using file compression. ****/
/**** Any elements you specify MUST be active in the ****/
/**** CSWCSU file or DAY040 willabend. ****/
/****                                     ****/
/**** If compression is not desired, omit the COMPRESS= ****/
/**** option keyword, or change YES to NO. ****/
/****                                     ****/

```

```

DATA &CICX..CICCSU00
  (KEEP = CICSID USERID TRANCODE STARTTS ENDTS CSUTRSTM
    CSUCPUTM CSUFCOPS COMPRESS=YES );
SET %%WRKMERG(WRK=CSW,CCC=CIC);
RUN;

```

```

*****
*      Repeat the following steps for each database *
*      unit containing the CICS Analyzer.           *
*****

```

- ___ 9. Review the definitions in prefix.MICS.PARMS(DBMODEL).
Modify the cycle retention number to match the file status in each timespan. For example, at least one cycle is retained for the DETAIL timespan for the files you activated. See Section 7.3.10 of this guide for details on DBMODEL definitions.

- ___ 10. Submit the job in prefix.MICS.CNTL(CYCLEGEN).

```

*****
*                                     *
* CHECKLIST 4.3 - ALTER ACTIVE TIMESPANS FOR FILE *
*                                     *
*****

```

- ___ 1. Save a copy of sharedprefix.MICS.GENLIB(CICGENIN) in sharedprefix.MICS.LOCALMOD.CNTL for recovery purposes.
- ___ 2. Review the description of the FILE statement in Section 4.2.4.2 of the System Modification Guide.
- ___ 3. Update the FILE statement in sharedprefix.MICS.GENLIB(CICGENIN) for each file to be modified. Use IEBUPDTE to apply the change.
- ___ 4. Regenerate the CICS Analyzer by submitting the job in

```
sharedprefix.MICS.CNTL(CICCGEN).
```

```
*****  
*      Repeat the following steps for each database *  
*      unit containing the CICS Analyzer.           *  
*****
```

- ___ 5. Review the definitions in prefix.MICS.PARMS(DBMODEL).
Modify the cycle retention number to match the
file status in each timespan. For example, zero
cycle is retained for all inactive timespans. See
Section 7.3.10 of this guide for details on DBMODEL
definitions.
- ___ 6. Submit the job in prefix.MICS.CNTL(CYCLEGEN).

10.1.4.1 Incremental Update Considerations

If you are using the TAPEfff option or USRXfff exits to collect detail level data, and incremental update is active, you may need to make a few simple but very important adjustments to your exits or options to accommodate the way that incremental update processes data.

This section discusses the following:

- o TAPEfff option considerations with incremental update
- o USRXfff exit considerations with incremental update
 - DETAIL tape files created in USRXfff exits
 - DASD files created in USRXfff exits

TAPEfff OPTION CONSIDERATIONS WITH INCREMENTAL UPDATE

The TAPEfff option allows the creation of DETAIL timespan tape files for certain high volume files.

Without incremental update, a single DETAIL timespan tape file is created for TAPEfff activated files during the DAILY run. The general practice is to use GDGs so that the (+0) tape volume contains yesterday's data, (-1) contains the prior day's data, and so on.

With incremental update activated, a tape is created with each execution of an incremental update as well as a final tape during the DAILY update run. For example, three incremental updates followed by a final daily update run result in a total of four tapes created, one for each TAPEfff activated file. Each tape contains only the detail data encountered during the individual incremental update or daily run. If you are employing GDGs you will want to consider increasing the number of generations to encompass the desired time range of history. For example, if previously 7 "daily" generations were maintained and you run 4 increments per day, you may wish to expand the generations kept to 28 (7 times 4 updates per day).

The use of fewer tapes in a single update under the incremental feature expedites processing. In addition, due to the "special study" nature of the data, separation into multiple pieces representing period time slices should allow usage to be more direct. That is, the data is already subset to the period contained within the update.

You also may consider creating a standalone job to consolidate these individual tapes into a single one after

the DAILY job completes if you prefer to have a single tape data set contain all transactions for the time range.

USRXfff EXIT CONSIDERATIONS WITH INCREMENTAL UPDATE

The USRXfff exits allow client supplied code to force the creation of SAS files containing DETAIL level data. These exits were provided for specific high volume transaction based data that are populated at the DAYS and higher timespans in a summarized form in CA MICS. Due to the high volume of data involved, these exits may be used to create tape files, although nothing other than space considerations precludes creation of files on DASD with these exits.

Many of the products that provide USRXfff exits have implemented standardized approaches to collect DETAIL level data, thus removing the need for USRXfff exit coding. Any user code employed in USRXfff exits should be examined to ensure that the data and the approach are still required. Generally, transaction data at this level is employed for specific tuning projects or special studies centered on subsystem or application related issues and is needed only for occasional and isolated time periods.

If you are using incremental update and find that the USRXfff exits you have coded are still required for your operation, be sure to review the considerations below.

DETAIL TAPE FILES CREATED IN USRXfff EXITS

When using the exits to write to tape without incremental update activated, a single tape file is created from the user written code in each activated USRXfff exit. The general practice is to use generation data groups (GDGs) to retain history. In this case, it is assumed that CA MICS units are updated once a day with yesterday's data. Without incremental update the generation 0 (+0) tape data set would contain data from the previous CA MICS update (yesterday), (-1) contains the prior run (two days ago), and so on.

With incremental update, a tape file is again created from the user written code in each activated USRXfff exit with each execution of an update. However, rather than running CA MICS updates once per day, incremental update implies multiple update runs daily. For example, three incremental updates followed by a final daily update run

result in a total of four tapes in one day, one created from each activated USRXfff exit. Each tape contains only the detail data encountered during the individual incremental update or daily run. If you are employing GDGs you will want to evaluate increasing the number of generations to encompass the desired time range of history. For example, if previously 7 "daily" generations were maintained and you ran 4 increments per day, you may wish to expand the generations kept to 28 (7 times 4 updates per day).

The use of fewer tapes in a single update under the incremental feature expedites processing. In addition, in keeping with the "special study" nature of the data, separation of the data into multiple pieces representing the narrower period slices should simplify usage. That is, the data is already subset to the period contained within the update.

You also may consider creating a standalone job to consolidate these individual tapes into a single one after the DAILY job completes if you prefer to have a single tape data set contain all transactions for the time range.

DASD FILES CREATED IN USRXfff EXITS

Either of two basic approaches may be employed when using the exits to write to DASD files. Data can be written to detail data set(s) of CA MICS units. Or it can be written to data sets allocated externally to the CA MICS units (such as a user data set that has been added to the Daily JCL stream). In either scenario the SAS file written by the exit will be replaced by new data with each execution of the update step.

If the exit is writing to the CA MICS Detail data set and the recommended approach has been taken with the exit then no change to outputs will occur under incremental update. The recommended approach employs indirect coding to reference the database (uses "&iiiit" not "DETAIL" in the data step) and an active detail status (file status of "Y" at the detail timespan in cccGENIN and cccCGEN has been executed). When these conditions are in place the file created by the exit will be treated as other CA MICS files in incremental update mode and at the conclusion of the Daily job stream a single file containing data from all increments will be available in the detail unit database. If the file status is not active, only data

from the last incremental execution will be present.

An alternative to selective exit coding is to employ a generation data set (GDG) structure for the external DASD file. In this scenario GDG entries to cover the incremental updates would be defined and each incremental run would write to a new GDG entry which would contain data for the increment. Each series of incremental runs followed by a DAILY job execution would create a new series of GDG data sets. While this approach allows you to capture all data on an ongoing basis, use of the unit detail data set as described above is more efficient and is the recommended approach.

10.1.5 Activate Archive for a File

For each CIC information area file, there is a corresponding FILE statement in sharedprefix.MICS.GENLIB(CICGENIN). The FILE statement defines the name, the type, the status of a file in each timespan, and the file status for each of the three possible archive files: the weekly audit, the weekly history, and the monthly history.

The weekly audit archive is written each week during the execution of the WEEKLY job or of the standalone Weekly Processing - Database Archive Audit (AUDIT) job. By default, up to 10 cycles of the DETAIL timespan for a given file are copied from the CA MICS database to tape. Since many files are inactive in the DETAIL timespan or have far fewer than 10 active cycles in the DETAIL timespan, one can archive cycles from the DAYS timespan instead of cycles from the DETAIL timespan.

Weekly history archive files are written each week by the WEEK300 step of the WEEKLY job or by the standalone Weekly Processing - Database Archive (HISTW) job. When weekly history archive is active for a particular CA MICS file, the just completed WEEKS cycle of the CA MICS file is copied to the output history tape for that file. At the same time, the last generation of the weekly history archive for the file is read so that its previous weekly cycles can be written to new tape generation along with its just-completed WEEKS cycle. The number of cycles of data that are written to the new tape generation is limited by a predefined retention period.

Monthly history archive files are written each month by the MONTH300 step of the MONTHLY job or by the standalone Monthly Processing - Database Archive (HISTM) job. When monthly history archive is active for a particular CA MICS file, the just-completed MONTHS cycle of the CA MICS file is copied to the output history tape for that file. At the same time, the last generation of the monthly history archive for the file is read so that its previous monthly cycles can be written to new tape generation along with its just completed MONTHS cycle. The number of cycles of data that are written to the new tape generation is limited by a predefined retention period.

The checklist below provides directions on how to activate archiving for a file.

```
*****
*
*   CHECKLIST 5 - ACTIVATE ARCHIVE FOR A FILE   *
*
```


*

- ___ 1. Save a copy of sharedprefix.MICS.GENLIB(CICGENIN) in sharedprefix.MICS.LOCALMOD.CNTL for recovery purposes.

- ___ 2. Decide which files are to be archived and for each of these files, decide which will have a weekly audit, a weekly history, or a monthly history archive. At the same time, decide whether the archiving will be performed during the WEEKLY and MONTHLY jobs or during the standalone AUDIT, HISTW, and HISTM jobs. For more information, see the PIOM.

- ___ 3. Review the description of the FILE statement in the CA MICS System Modification Guide.

- ___ 4. In sharedprefix.MICS.GENLIB(CICGENIN), update the FILE statement for each file to be modified. For weekly archive audit, the file should be active in either the DETAIL or DAYS timespan. For weekly archive history, the file should be active in the WEEKS timespan. For monthly archive history, the file should be active in the MONTHS timespan.

At the same time, activate the archive files by specifying "Y" in the appropriate positions on the FILE statement for each file as described in the documentation reference in step 3 above.

- ___ 5. Regenerate the CICS Analyzer by submitting the job in sharedprefix.MICS.CNTL(CICCGEN).

* Repeat the following steps for each database *
* unit containing the CICS Analyzer. *

- ___ 6. Review the definitions in prefix.MICS.PARMS(DBMODEL). Modify the cycle retention number to match the file status in each timespan and the archive cutoff values indicating the number of cycles of data to be kept in the weekly and monthly archive history files. For details, see section 7.3.10 of this guide for details on DBMODEL definitions.

- ___ 7. Submit the job in prefix.MICS.CNTL(CYCLEGEN).

- ___ 8. Modify prefix.MICS.PARMS(JCLGENU) so that it contains a single line that reads:

WEEKLY MONTHLY AUDIT HISTW HISTM

- ___ 9. Submit the job in prefix.MICS.CNTL(JCLGENU).
- ___ 10. Submit the job in prefix.MICS.CNTL(GDGSGEN).

10.1.6 Add a New File

To add a new file to retain information that is not normally kept in the CA MICS Data Base, follow the checklist provided below.

```
*****
*
*          CHECKLIST 6 - ADD A NEW FILE          *
*
*****
```

- ___ 1. Save a copy of the following members in
sharedprefix.MICS.LOCALMOD.CNTL for recovery
purposes:

sharedprefix.MICS.GENLIB: CICGENIN, COMPTDEF
sharedprefix.MICS.HOLD.PARMS: COMPTDEF
sharedprefix.MICS.SOURCE: #CICEXIT
- ___ 2. Review the file and data element definition statement
descriptions in Sections 4.2.4.2 and 4.2.4.3 of the
System Modification Guide (SMG).
- ___ 3. Add the new file to the CICGENIN member of your
sharedprefix.MICS.GENLIB using the file and data
element definition statements. Use IEBUPDTE to apply
the change.
- ___ 4. Regenerate the CICS Analyzer by submitting the job in
sharedprefix.MICS.CNTL(CICCGEN).
- ___ 5. Define SAS DATA statement and KEEP list for cycle 00
of the new file in exit USRSDKP in
sharedprefix.MICS.SOURCE(#CICEXIT). See Section
10.2.3 of this guide for an example.
- ___ 6. Insert your code for creating the new file in the
appropriate exit in
sharedprefix.MICS.SOURCE(#CICEXIT). For example, to
output an observation during transaction processing
time, use the USRSCSW exit. See Section 10.2.3 of
this guide for an example.

```
*****
*          Repeat steps 7 through 11 for each data      *
*          base unit containing the CICS Analyzer.      *
*****
```

___ 7. Review the definitions in prefix.MICS.PARMS(DBMODEL).
Modify the cycle retention number to match the file
status in each timespan and the status for audit and
history archive. See Section 7.3.10 of this guide
for details on DBMODEL definitions.

___ 8. Submit the job in prefix.MICS.CNTL(CYCLEGEN).

```
*****
*   Perform steps 9 through 11 if file archive   *
*   is active.                                   *
*****
```

___ 9. Modify prefix.MICS.PARMS(JCLGENU) so that it contains
a single line that reads:

GDGSGEN

___ 10. Submit the job in prefix.MICS.CNTL(JCLGENU).

___ 11. Submit the job in prefix.MICS.CNTL(GDGSGEN).

10.1.7 Add or Delete a CICS Region in Daily Processing

To add a CICS region for daily processing in a unit, follow Checklist 7.1 provided below. To delete a CICS region, follow Checklist 7.2.

```
*****
*
* CHECKLIST 7.1 - ADD A CICS REGION FOR PROCESSING *
*
*****
```

- ___ 1. If the data is produced by a new CICS monitor that has not been processed by the CA MICS Analyzer Option for CICS (for example, you are converting from CMF to ASG-TMON), then check the OPTION statement in sharedprefix.MICS.GENLIB(CICGENIN) for the appropriate data source definition. See Section 7.2.1 of this guide for OPTION statement syntax.
- ___ 2. If sharedprefix.MICS.GENLIB(CICGENIN) is modified in step 1, then submit the job in sharedprefix.MICS.CNTL(CICCGEN).
- ___ 3. If the new CICS region produces data containing SMF system identification that has not been defined to CA MICS, then add the new system ID to prefix.MICS.PARMS(SYSID). This step is not necessary if data to be processed is CMF journal. See Section 2.3.2.2 of the PIOM for a detailed description of the SYSID member.
- ___ 4. If new system IDs are added in step 3, then submit the job in prefix.MICS.CNTL(BASPGEN).
- ___ 5. For each new CICS region to be processed by the CA MICS Analyzer Option for CICS, define a CICOPTS statement in prefix.MICS.PARMS(CICOPS). See Section 7.3.2 of this guide for detailed CICOPTS statement syntax.
- ___ 6. Modify prefix.MICS.PARMS(INPUTCIC) to add the input DD statements for the new regions. Do not modify this member if you coded SMF as the ddname in step 5. See Section 7.3.4 of this guide for a description of INPUTCIC.
- ___ 7. If you coded SMF as the ddname in step 5, then review or modify prefix.MICS.PARMS(INPUTSMF) to add the

input DD statements for the new regions. See Section 7.3.3 of this guide for a description of INPUTSMF.

- ___ 8. If you coded SMF as the ddname in step 5, then review or modify prefix.MICS.PARMS(JCLDEF) to ensure SMFRECORDING CICS is specified. See Section 2.3.3.2.1.1 of the PIOM for more details.
- ___ 9. Submit the job in prefix.MICS.CNTL(CICPGEN).
- ___ 10. If you added new JCL in step 6 or 7, or turned on SMFRECORDING in step 8, then code the following line in prefix.MICS.PARMS(JCLGENU):

DAILY

- ___ 11. If step 10 is performed, then submit the job in prefix.MICS.CNTL(JCLGENU) to regenerate the DAILY JCL.

```
*****
*
* CHECKLIST 7.2 - DELETE A REGION FROM PROCESSING *
*
*****
```

- ___ 1. Review input data sources defined by the OPTION statement in sharedprefix.MICS.GENLIB(CICGENIN). Deactivate a data source if applicable. For example, change OPTION CMF to OPTION NOCMF.
- ___ 2. If sharedprefix.MICS.GENLIB(CICGENIN) is updated in step 1, then submit the job in sharedprefix.MICS.CNTL(CICCGEN).
- ___ 3. Remove CICOPTS statements of affected regions from prefix.MICS.PARMS(CICOPS).
- ___ 4. Remove input JCL for affected regions from prefix.MICS.PARMS(INPUTCIC), if applicable.
- ___ 5. Remove input JCL for affected regions from prefix.MICS.PARMS(INPUTSMF), if applicable.
- ___ 6. Turn off SMFRECORDING for CICS in prefix.MICS.PARMS(JCLDEF), if applicable.
- ___ 7. Submit the job in prefix.MICS.CNTL(CICPGEN).

- ___ 8. If you changed JCL in step 4 or 5, or turned off SMFRECORDING in step 6, then code the following line in prefix.MICS.PARMS(JCLGENU):

DAILY

- ___ 9. If step 8 is performed, submit the job in prefix.MICS.CNTL(JCLGENU) to regenerate the DAILY JCL.

10.1.8 Change the DDnames of Input Data

To change DDnames of the input CICS data in a unit, follow the checklist provided below.

```
*****
*                                     *
*  CHECKLIST 8 - CHANGE DDNAMES OF INPUT DATA  *
*                                     *
*****
```

- ___ 1. For each affected region, update the CICOPTS statements in prefix.MICS.PARMS(CICOPS) to reflect the new DDnames. See Section 7.3.2 of this guide for CICOPTS statement syntax.
- ___ 2. Modify prefix.MICS.PARMS(INPUTCIC) to change the DDnames or remove JCL for DDnames that have been renamed. Do not code the DDname INPUTSMF in this member. See Section 7.3.4 of this guide for a description of INPUTCIC.
- ___ 3. If you coded SMF as the DDname in step 1, then add DDname INPUTSMF in prefix.MICS.PARMS(INPUTSMF). See Section 7.3.3 of this guide for a description of INPUTSMF.
- ___ 4. If you coded SMF as the DDname in step 1, then review or modify prefix.MICS.PARMS(JCLDEF) to ensure SMFRECORDING CICS is specified. See Section 2.3.3.2.1.1 of the PIOM for more details.
- ___ 5. Submit the job in prefix.MICS.CNTL(CICPGEN).
- ___ 6. Modify prefix.MICS.PARMS(JCLGENU) to contain one line that reads:

DAILY
- ___ 7. Submit the job in prefix.MICS.CNTL(JCLGENU) to regenerate the DAILY JCL.

10.1.9 Activate the CICS File/DBD Activity File (CICCSF)

The CICS File/DBD Activity File (CICCSF) is an optional file created from the file segments in The Monitor's task accounting detail records. It provides file access statistics by file ID. The CICCSF file can be voluminous if you have defined a large number of file segments for The Monitor. Therefore, to limit the DASD requirement of the CA MICS Data Base, the CICCSF file is distributed as an inactive file. To activate this file, follow the checklist provided below.

```
*****
*                                     *
*   CHECKLIST 9 - ACTIVATE THE CICCSF FILE   *
*                                     *
*****
```

- ___ 1. Save a copy of sharedprefix.MICS.GENLIB(CICGENIN) in sharedprefix.MICS.LOCALMOD.CNTL for recovery purposes.
- ___ 2. Modify sharedprefix.MICS.GENLIB(CICGENIN). Code the following FILE statement for the CICCSF file if the file will only be turned on in the DAYS and MONTHS time spans.

```
FILE CSF 00 1 N E N Y N N N CICS File/DBD Activity File
```

Code the following FILE statement for the CICCSF file if the file will be turned on in the DETAIL, DAYS, and MONTHS time spans:

```
FILE CSF 00 1 Y E N Y N N N CICS File/DBD Activity File
```

- ___ 3. Submit the job in sharedprefix.MICS.CNTL(CICCCGEN).
- ___ 4. To help reduce the volume of the CICCSF file, use the _USRSCSF exit in prefix.MICS.USER.SOURCE(#CICEXIT) to keep only those files that are of a critical nature. One way to do this is by examining the CSFFID data element for a specific file ID. If CSFFID is not for a file that should be kept in the data base, then skip the record by setting the variable SKIP_REC to one. This will prevent the observation from being written to the CICCSF file.

For example :

```
MACRO _USRSCSF
  IF CSFFID=: 'DFH' THEN SKIP_REC=ONE;
  %

___ 5. If the CICCSCF file is being activated in the DETAIL
      timespan, then make the following updates to the
      _USRXCSF macro in prefix.MICS.USER.SOURCE(#CICEXIT):

DATA &CICX..CICCSF00 (KEEP=%CSFFILE(OP=FILEOPTS,TS=DETAIL));
  SET %CSFMERG;
RUN;

___ 6. Review the definitions in prefix.MICS.PARMS(DBMODEL).
      Modify the cycle retention number to match the file
      status in each timespan. For example, at least one
      cycle is retained for the DETAIL timespan for the
      files you activated. See Section 7.3.10 of this
      guide for details on DBMODEL definitions.

___ 7. Submit the job in prefix.MICS.CNTL(CYCLEGEND).
```

10.1.10 Change Retention Period of the CICCDC File

The CICS Dictionary File (CICCDC) contains the data dictionary records that are required by the CICS Analyzer to input CMF data. It exists at the DETAIL timespan in cycle 01 only. As new data dictionary records are encountered during input, the latest copy is kept in storage and output to the CICCDC file at end of input processing. The number of days that the data dictionary records are kept in the CICCDC file is dependent upon a threshold value that you specify.

To change the retention period of the dictionary records in the CICCDC01 file, simply change the CICCDCX statement in prefix.MICS.PARMS(CICTHRSH). No generations are required afterwards, as this member is input during the daily processing.

See Section 7.3.8 of this guide for more information on CICS processing thresholds.

10.1.11 Change the Number of Work Files for DAY040

To change the number of work file used by the detail transaction processing in DAY040, follow the checklist provided below for each unit.

```
*****
*
* CHECKLIST 11 - CHANGE NUMBER OF WORK FILES
*
*****
```

See Section 7.3.4 of this guide for a detailed description of the statement syntax.

```
*****
*
* CHANGE NUMBER OF WORK FILES
*
*****
```

- ___ 1. Update the WORK statement in prefix.MICS.PARMS(cccOPS), where (ccc) is the component identifier, to specify the number of work data sets required. Below is an example:

```
WORK n STORCLAS=MICSTEMP SPACE=(XXX,(pppp,ssss))
```

where:

- n - is the number of WORK data sets.
- STORCLAS - specifies a storage class for a new data set. The name can have up to eight characters.
- SPACE - specifies how much disk space to provide for a new data set being allocated.
- XXX - is TRK or CYL.
- pppp - is the primary allocation.
- ssss - is the secondary allocation.

You should specify the minimum number of WORK data sets to meet your work space requirements. As a start, try incrementing the number gradually beginning from the default.

- ___ 2. If this is the first time you are implementing multiple work files for this product, then continue with Step 2. If you are just changing the number currently in use, or simply the space definitions,

then proceed to Step 3 of this checklist.

Browse `sharedprefix.MICS.PROTOLIB(DYcccnnn)` and `sharedprefix.MICS.PROTOLIB(cccINCR)`, where (nnn) is the job step number and (ccc) is the product ID for this product, checking for the presence of the WORK symbolic on the EXEC statement to determine if you have previously modified this product to increase the allocation of SAS WORK space.

- ___ 2a. If you find a WORK symbolic, simply divide the primary and secondary allocation values from the WORK symbolic by the number of work files specified above (value of n on the WORK statement coded in Step 1).

Coding the resulting values will yield the same aggregate space allocation you have been using with a single WORK file. To double your available WORK space, carry out the division, double the results and use the values in the WORK definition above.

- ___ 2b. If you did not find a WORK symbolic in PROTOLIB, examine `prefix.MICS.PARMS(JCLDEF)` for each CA MICS unit containing this product. Find the WORKSPACE keyword. The space allocation specified is used for a single SAS WORK file. Perform the same division as described in the previous paragraph to determine the quantity that will yield equivalent total allocation with multiple WORK files. Then adjust the values upward to meet your needs.

- ___ 3. Submit the job in `prefix.MICS.CNTL(cccPGEN)`.
- ___ 4. If you specified RESTART YES in the product's cccOPS, you are done. Otherwise, you must do Steps 5, 6, and 7.
- ___ 5. Edit `prefix.MICS.PARMS(JCLGENU)` so that it contains a single line that reads:

DAILY

or, if incremental update is enabled for this product in this unit database, specify:

DAILY INCRccc

where ccc is the product ID.

- ___ 6. Submit the job in prefix.MICS.CNTL(JCLGENU). Ensure that there are no error messages in MICSLOG or SYSTSPRT, that the MICSLOG contains the normal termination message, BAS10999I, and that the job completes with a condition code of zero.
- ___ 7. The following operational job(s) have changed:

DAILY

INCRccc (if incremental update is enabled)

If your site has implemented the operational CA MICS processes in a scheduling product, the JCL may have to be refreshed in that product. See the scheduling product's administrator for the exact processes involved in updating that product's representation of the CA MICS jobs.

10.1.12 Process CMF User Clocks, Counters, and User Area

The CICS Monitoring Facility (CMF) allows application programs to add data fields to the transaction records in the performance class and accounting class. CMF calls these data fields user clocks and counters (it supports 256) and the user area (it allows one user area per ENTRYNAME defined by the DFHMCT TYPE=EMP macro).

The CA MICS CICS Analyzer supports user clocks, user counters, and the user area from the performance class only. Data from the accounting class is accessible during the daily input processing step in DAY040; however, you must provide the necessary SAS code in an exit routine to create observations or files based on this data.

By default, the CA MICS CICS Analyzer supports nine user clocks, 50 user counters, and one user area (stored in two SAS variables due to the limit of 200-byte length for character variables), though the support is shipped inactive. The ENTRYNAME for these fields must be USER (the default name supplied by CICS). If you have specified an ENTRYNAME other than USER and you would like to process that data through CA MICS, refer to Section 10.1.15 of this guide for more information.

If you choose to activate this support and the defaults do not meet your installation's needs, you can define and activate up to 99 user clocks and user counters.

If your site has more than one CICS region, be certain to get the needed information from each region before activating additional user clocks and counters in the CICS Analyzer. If you activate clocks or counters or a user area for one CICS region, the data elements are carried for the other regions and may contain missing values if the other regions have no user clocks or counters.

To add or activate additional user clocks and user counters, follow Checklist 12.

```

*****
*
* CHECKLIST 12 - ADD USER CLOCKS AND USER COUNTERS *
*
*****

```

- ___ 1. Determine the number of user clocks in each CICS region processed by the CICS Analyzer by doing the following:
 - a. Ask your CICS system programmer.
 - or-
 - b. If you are running CMF, review the DFHMCT TYPE=EMP macro specifications and make note of the PERFORM= statement. The number of clocks is equal to the largest number specified in the SCLOCK statement. If there are no SCLOCK statements, no clocks have been defined.
- ___ 2. Determine the number of user counters in each CICS region processed by the CA MICS CICS Analyzer by doing the following:
 - a. Ask your CICS systems programmer.
 - or-
 - b. If you are running CMF, review the DFHMCT TYPE=EMP macro specifications and make note of the PERFORM= statement. The number of counters is equal to the largest number specified in the MLTCNT(x,y) statement.
- ___ 3. Modify or add NAME (or NAMX if you using ESSENTIAL=ALL support) statements to provide the number of clocks and counters you need in sharedprefix.MICS.GENLIB(CICGENIN). For each clock, you must activate or add NAME statements for data element CICPCKnn. For each counter, activate or add NAME statements for CICPCTnn, where nn ranges from 01 to 99. Note that these data elements are defined in THREE files and must be activated in the CICCSW, and, optionally, CICCUS or CICCAS files. See Section 4.2.4 of the System Modification Guide (SMG) for details on modifying NAMX or NAME statements.
- ___ 4. The user clocks and user counters are defined as common data elements, as they exist in multiple CICS files in the CA MICS Data Base. Therefore, for each NAME (or NAMX) statement that you added in step 3, you must add the corresponding ALIAS, TYPE, and NAME

statements in `sharedprefix.MICS.GENLIB($DEGENIN)`.

In addition, you can change the format and length of the clocks and counters by changing the TYPE statement in the \$DEGENIN member. See Section 4.2.4.4 of the SMG for details.

- ___ 5. Submit `sharedprefix.MICS.CNTL(CICCGEN)`.

10.1.13 Activate the CICS MRO Activity File (CICCMR)

The CICS MRO Activity File (CICCMR) is an optional file created from the MRO segments in The Monitor's detail transaction records. It provides MRO activity statistics by APPLID. The CICCMR file can be voluminous if you have defined a large number of MRO segments for The Monitor. Therefore, to limit the DASD requirement of the CA MICS Data Base, the CICCMR file is distributed as an inactive file. To activate this file, follow the checklist provided below.

```
*****
*
*   CHECKLIST 13 - ACTIVATE THE CICCMR FILE   *
*
*****
```

- ___ 1. Save a copy of sharedprefix.MICS.GENLIB(CICGENIN) in sharedprefix.MICS.LOCALMOD.CNTL for recovery purposes.
- ___ 2. Modify sharedprefix.MICS.GENLIB(CICGENIN). Code the following FILE statement for the CICCMR file if the file will only be turned on in the DAYS and MONTHS timespans.

```
FILE CMR 00 1 N E N Y N N N CICS MRO Activity File
```

Code the following FILE statement for the CICCMR file if the file will be turned on in the DETAIL, DAYS, and MONTHS timespans:

```
FILE CMR 00 1 Y E N Y N N N CICS MRO Activity File
```

- ___ 3. Submit the job in sharedprefix.MICS.CNTL(CICCGEN).
- ___ 4. To help reduce the volume of the CICCMR file, use the _USRSCMR exit in prefix.MICS.USER.SOURCE(#CICEXIT) to keep only those MRO segments that are of a critical nature. One way to do this is by examining CMRAPPLD for a specific APPLID value. If CMRAPPLD is not for a region that should be kept in the data base, then skip the record by setting the variable SKIP_REC to 1. This will prevent the observation from being written to the CICCMR file.

For example :

```
MACRO _USRSCMR
```

```
IF CMRAPPLD='CIC1' THEN SKIP_REC=ONE;  
%
```

- ___ 5. If the CICCMR file is being activated in the DETAIL timespan, then make the following updates to the _USRXCMR macro in prefix.MICS.SOURCE(#CICEXIT):

```
DATA &CICX..CICCMR00 (KEEP=%CMRFILE(OP=FILEOPTS,TS=DETAIL));  
SET %%CMRMERG;  
RUN;
```

- ___ 6. Review the definitions in prefix.MICS.PARMS(DBMODEL). Modify the cycle retention number to match the file status in each timespan. For example, at least one cycle is retained for the DETAIL timespan for the files you activated. See Section 7.3.10 of this guide for details on DBMODEL definitions.

- ___ 7. Submit the job in prefix.MICS.CNTL(CYCLEGEND).

10.1.14 Convert to a Different Data Source

The CA MICS Analyzer Option for CICS supports two data source definitions: CMF (this includes CICS statistics processing for CICS 3.1.1 and CMF look-alike records produced by OMEGAMON for CICS) and ASG-TMON for CICS TS (TCE).

When you switch from one input data source to another, you must change the appropriate options in CA MICS. The following checklist shows what steps must be taken to accomplish this.

```
*****
*                                     *
*   CHECKLIST 14 - CONVERT TO A DIFFERENT DATA   *
*   SOURCE                                           *
*                                     *
*****
```

- ___ 1. Make a backup copy of all complex and unit level libraries in the complex.
- ___ 2. Edit sharedprefix.MICS.GENLIB(CICGENIN) and specify the correct parameters for the OPTION statement. Turn off the parameters that are unrelated to your data source. For example, for CMF processing, specify the following:


```
OPTION CMF NOMON NORESP NOOMEGU
```

See section 7.2.1 of this guide for a complete description of the OPTION statement.
- ___ 3. Update the FILE statement in the CICGENIN member of sharedprefix.MICS.GENLIB to turn on/off files that are related to your data source. Currently, only two CICS files are unique to a specific data source. They are the CICCFS (CICS File/DBD Activity) and CICCMT (CICS MRO Activity) files, both of which are created from ASG-TMON for CICS TS (TCE) only and are turned off as shipped. You may activate them as needed using the instructions provided in sections 10.1.9 (CICCFS) and 10.1.13 (CICCMT) of this guide.
- ___ 4. Submit sharedprefix.MICS.CNTL(CICCGEN).
- ___ 5. Copy the appropriate decompression modules for the new input data source to sharedprefix.MICS.LOAD.

```
*****
*   Repeat the following steps for each database *
*   unit containing the CA MICS Analyzer Option *
*   for CICS.                                   *
*****
```

- ___ 6. Edit prefix.MICS.PARMS(CICOPS) and update the CICOPTS statements to reflect the new data source. See section 7.3.2 for a complete description of the CICOPTS statement.
- ___ 7. Edit prefix.MICS.PARMS(INPUTCIC) to specify the JCL for the data source. If the input file is SMF, leave only comments in this member. Do not delete the member. See section 7.3.4 for a complete description of prefix.MICS.PARMS(INPUTCIC).
- ___ 8. If the input is SMF, edit prefix.MICS.PARMS(JCLDEF) to specify SMFRECORDING for CICS.
- ___ 9. Edit prefix.MICS.PARMS(INPUTSMF) to specify the JCL if the input file is SMF. You must use the ddname INPUTSMF when coding the JCL for this member. See section 7.3.3 for a complete description of prefix.MICS.PARMS(INPUTSMF).
- ___ 10. Submit prefix.MICS.CNTL(CICPGEN).
- ___ 11. Edit prefix.MICS.PARMS(JCLGENU) so that it reads:

DAILY
- ___ 12. Submit prefix.MICS.CNTL(JCLGENU) to regenerate the DAILY JCL.
- ___ 13. Review the definitions in prefix.MICS.PARMS(DBMODEL). Modify the cycle retention number to match the file status in each timespan.
- ___ 14. Submit prefix.MICS.CNTL(CYCLEGEN) if DBMODEL was modified in Step 13.

10.1.15 Process Additional Non-USER Areas in CMF

The CICS Monitoring Facility (CMF) allows application programs to add additional fields, known as user area, to the performance class data. Currently, the CA MICS Analyzer Option for CICS supports user areas with the ENTRYNAMES of USER, DBCTL, as well as, OMEGAMON user areas OMEGBSC, OMEGDLI, and OMEGDB2.

For other user areas that you have added to CMF, you can follow Checklist 15 to access the user areas during the input processing step in DAY040; however, you must provide the necessary SAS code in an exit routine to save the field in a CA MICS data element if you wish to retain the data in the CA MICS database.

```
*****
*
* CHECKLIST 15 - Process Non-USER CMF User Areas *
*
*****
```

- ___ 1. Determine the name of the user areas that you wish to process by doing the following:
 - a. Ask your CICS system programmer.
 - or-
 - b. Review the MICSLOG from DAY040. For each user field that was skipped during input processing, the warning message CIC06128 is issued. You can determine from the message text the name of the user field (OWNER=), and the type of field (area, clock, or counter).
- ___ 2. Edit sharedprefix.MICS.SOURCE(CICDEPEL) to add the names of the user areas that you wish to process. The CICDEPEL module has reserved nine DEFINEID statements with IDs 3902 through 3910 for additional user area definitions. For each user area you wish to add, simply un-comment the line and change the word RESERVED on the DEFINEID statement to the name of your user area. Each name specified by the DEFINEID statement corresponds to a SAS variable containing the content of the user area. You may reference the SAS variable in a detail exit during DAY040 processing.

For example, to add user areas SCREEN and FUNCTION, edit the statements in module CICDEPEL as follows:

From:

```
* DEFINEID    3902 RESERVED
* DEFINEID    3903 RESERVED
```

To:

```
DEFINEID    3902 SCREEN
DEFINEID    3903 FUNCTION
```

If you need to add more than nine user areas, insert additional DEFINED statements (via user modification) as needed and increment each field ID (39xx) by 1. For example, you may insert four additional user areas with IDs of 3911 through 3914.

```
*****
* Repeat Steps 3 and 5 for each unit database that contains *
* the CA MICS Analyzer Option for CICS.                      *
*****
```

- ___ 3. Code the USRCDCT exit in member #CICEXIT in prefix.MICS.USER.SOURCE. This adds the user area IDs defined in Step 2 to the CDADID data dictionary array. Because this exit is called within SELECT logic, the exit code must contain WHEN clauses instead of IF-THEN logic. There must be a WHEN clause for each user area defined in Step 2.

For example, to add the SCREEN and FUNCTION user areas from Step 2, you must code the USRCDCT exit as follows:

```
%MACRO USRCDCT;
  WHEN('SCREEN')  CDADID(CDAINDEX)=PUT(902,PIB2.);
  WHEN('FUNCTION') CDADID(CDAINDEX)=PUT(903,PIB2.);
%MEND USRCDCT;
```

Note: The name in quotes on the WHEN clause must match the name on the DEFINED statement from Step 2. The three-digit field ID in the PUT function must match the last three digits of the same DEFINEID statements from Step 2.

- ___ 4. To access a user area in a detail exit (such as CICACRT, CICRLRT, CICAURT, or _USRSCSW) during DAY040

processing, you must code a LENGTH statement for the user area before referencing it in the exit. For example, to reference the SAS variables SCREEN and FUNCTION, you may code the following in the CICACRT exit:

```
LENGTH SCREEN FUNCTION $ 8;
CICACT5=SCREEN;
CICACT6=FUNCTION;
```

Note the maximum length of a character user area is 200 bytes. In the above example, the content of SAS variables SCREEN and FUNCTION are stored in the CA MICS data elements CICACT5 and CICACT6, respectively. You may save the content of the user areas to any CA MICS data elements, either pre-defined or user-defined through system modifications, as required for your processing needs.

- ___ 5. Code the USRSINT exit in the #CICEXIT member of prefix.MICS.USER.SOURCE. This exit is a SAS macro that is used to initialize the user areas to blanks, prior to reading the data for each new CICS region. To initialize the user areas SCREEN and FUNCTION, code the macro as follows:

```
%MACRO USRSINT;
  SCREEN=BLANKS;
  FUNCTION=BLANKS;
%MEND USRSINT;
```

Coding this macro prevents data in user areas for a CICS region from being written to CA MICS data elements for regions that do not have those user areas defined.

10.1.16 Enable Internal Step Restart

To enable the internal step restart in the CA MICS CICS Analyzer, follow the checklist provided below:

See Section 7.3.2 of this guide for a detailed description of the statement syntax.

```
*****
*                                     *
*           ENABLE INTERNAL STEP RESTART           *
*                                     *
*****
```

- ___ 1. Edit prefix.MICS.PARMS(cccOPS), where (ccc) is the component identifier, and specify:

RESTART YES

For additional information on related topic, review the documentation for this product on WORK, RESTARTWORK, and RESTARTCKPT parameters to override default data set allocation parameters.

- ___ 2. Submit the job in prefix.MICS.CNTL(cccPGEN).

- ___ 3. Edit prefix.MICS.PARMS(JCLGENU) so that it contains a single line that reads:

DAILY

or, if incremental update is enabled for this product in this unit database, specify:

DAILY INCRccc

where ccc is the product ID.

- ___ 4. Submit the job in prefix.MICS.CNTL(JCLGENU). Ensure that there are no error messages in MICSLOG or SYSTSPRT, that the MICSLOG contains the normal termination message, BAS10999I, and that the job completes with a condition code of zero.

- ___ 5. The following operational job(s) have changed:

DAILY

INCRccc (if incremental update is enabled)

If your site has implemented the operational CA MICS processes in a scheduling product, the JCL may have to be refreshed in that product. See the scheduling product's administrator for the exact processes involved in updating that product's representation of the CA MICS jobs.

10.1.17 Implement Incremental Update

To implement incremental update in the CA MICS CICS Analyzer, follow the checklist provided below:

See Section 7.3.2 of this guide for a detailed description of the statement syntax.

```
*****
*                                                                 *
*  Note:  If you are using the TAPEfff option or USRXfff       *
*          exits, be sure to review the important              *
*          considerations in Section 10.1.4.1 of this guide     *
*          before activating incremental update.                *
*                                                                 *
*****
```

```
*****
*                                                                 *
*          IMPLEMENT INCREMENTAL UPDATE                        *
*                                                                 *
*****
```

- ___ 1. Edit prefix.MICS.PARMS(cccOPS), where (ccc) is the component identifier:
 - o Specify the following:

INCRUPDATE YES
 - o If you want to store the incremental update database files on tape between incremental updates, specify this:

INCRDB TAPE #gdgs
 - o If you want to allocate the incremental update database files during the first incremental update of the day and delete these data sets at the end of the DAILY job step, specify this:

INCRDB DYNAM
 - o If you specified INCRDB TAPE or INCRDB DYNAM, then you must also specify this:

INCRDETAIL data_set_allocation_parameters
INCRDAYS data_set_allocation_parameters

- o If you want the incremental update job for this product to get input measurement data from the output of the SPLITSMF job, specify this:

```
INCRSPLIT USE data_set_allocation_parameters
```

- o For additional information on related topic, review the documentation for this product on INCRCKPT, INCRDETAIL, INCRDAYS, or INCRSPLIT parameters to override default data set allocation parameters.

___ 2. Submit the job in prefix.MICS.CNTL(cccPGEN).

___ 3. Edit prefix.MICS.PARMS(JCLGENU) so that it contains two or more lines reading:

```
DAILY
INCRccc cccIUALC cccIUGDG
```

___ 4. Submit the job in prefix.MICS.CNTL(JCLGENU). Ensure that there are no error messages in MICSLOG or SYSTSPRT, that the MICSLOG contains the normal termination message, BAS10999I, and that the job completes with a condition code of zero.

___ 5. Edit the job in prefix.MICS.CNTL(cccIUALC).

- o Inspect and/or specify data set allocation parameters for the incremental update database and checkpoint files. If you specified INCRDB TAPE or INCRDB DYNAM, the cccIUALC job will only allocate the incremental update checkpoint data set.
- o Submit the job. Ensure that there are no error messages in MICSLOG or SASLOG, and that the job completes with a condition code of zero.

___ 6. If you specified INCRDB TAPE, submit the job in prefix.MICS.CNTL(cccIUGDG) to define generation group indexes for the incremental update DETAIL and DAYS tape data sets. Examine SASLOG, MICSLOG, and SYSPRINT to verify that the generation group indexes were correctly defined.

Note: You may see error messages for the DLTX (or DELETE) statements. This is not a problem. cccIUGDG deletes each index prior to defining it, and an error

message is issued if the index does not yet exist (e.g., if this is the first time you ran the cccIUGDG job).

- ___ 7. The following operational job(s) have changed:

DAILY INCRccc

If your site has implemented the operational CA MICS processes in a scheduling product, the JCL may have to be refreshed in that product. See the scheduling product's administrator for the exact processes involved in updating that product's representation of the CA MICS jobs.

- ___ 8. Implement operational procedures for gathering input measurement data and executing incremental updates (INCRccc) during the day.

You may also need to modify operational procedures for the DAILY job to ensure that processing is limited to input measurement data that has not been input to one of the day's incremental update executions.

10.2 User Exits

Before implementing an exit, determine whether the required modification can be accomplished by means of parameter modification or file tailoring.

At times, additional system customization is needed to tailor CA MICS to your data center. There are many user exit points provided within CA MICS. Each of these points corresponds to a position in the CA MICS logic where user modification:

- o is likely to occur, based on our experience
- o presents only a slight possibility of compromising the CA MICS database or operational integrity
- o is relatively easy to document and understand

Using the CA MICS exit facilities to augment CA MICS processing logic is the safest method of system modification. Generally, you have four methods with which to augment, change, and enhance CA MICS.

- o CA MICS standard option and parameter definitions

The diverse installation and definition options provide most users with adequate provisions for customizing CA MICS.

- o Modification through user exits

Often there may be site-dependent requirements that are not adequately addressed through the standard options and parameters. In these cases, one or more of the user exits provided with CA MICS may be used to insert user-written routines that satisfy the requirements.

- o Extension through user-written components

Although most users do not initially consider this a way of modifying a standard CA MICS product, the development of user-written components may be the most advantageous method. For example, often it may be better to write a user component to handle additional SMF data than to attempt to modify the logic of the CA MICS CICS Analyzer.

- o System code modification

Finally, if all else fails, you may be able to satisfy your site's requirements by implementing a source code change.

GENERAL EXIT CONSIDERATIONS

You should carefully explore whether or not it is necessary to develop a user exit routine. To this end, first ensure that the requirement cannot be satisfied through one of the standard options or definitions. For assistance, contact Technical Support at <http://ca.com/support>.

The design, coding, testing, and implementation of CA MICS user exit routines should be done carefully, since errors may result and corrupt the data. You should be meticulous in the definition and validation of the exit routines to ensure that system integrity and performance is not adversely affected.

If an exit must be used, refer to Chapter 9 of this guide for a description of the location of the exit in the DAILY update. For a better understanding of CA MICS user exits, see Section 4.3 of the SMG, User Exit Facilities.

Product exits are classified according to their purpose. They include product input, output, parameter-related, and accounting exits. Input exits are invoked as the raw input records are read. Output exits are invoked just before CA MICS writes an observation to the CA MICS database.

The CICS Analyzer provides the input exit `_USRSEL`. You can use this exit to select or exclude input data that meets special site-defined criteria, add data elements, or modify record fields as they are read.

One output exit routine is provided for each product file. You can use this output exit routine to selectively block the writing of specified records, alter data elements in the records, or produce additional records for the target file or a user-defined file.

The product parameter-related exits are also user exits, but they are defined as part of the standard CA MICS installation process. They are normally associated with other parameters or options. For example, the Account Code Derivation Routine is a user exit routine for completing values for account codes that were defined in the separate but related CICACRT member of prefix.MICS.PARMS. CICACRT is the Account Code Definition parameter member. These exits are identified in

this chapter, but are described in Chapter 7 of this guide.

The user exits for the CICS Analyzer are discussed in detail in the following sections:

- 1 - Input Exits
- 2 - Parameter-Related Exits
- 3 - Output Exits

10.2.1 Input Exits

This section provides a description of the input exit that is shared by CICS with the other installed CA MICS components.

The exit description includes the name and title, a description of its purpose, when it is invoked, and whether it has an interface to the CA MICS Accounting and Chargeback Option. It also shows which data elements are available, the special considerations to note, and a sample user exit.

```
+-----+
| U S R S E L |   Input Record Selection Exit (USRSEL)
+-----+
```

DESCRIPTION: The _USRSEL exit lets you prevent records from being processed by the CA MICS Analyzer Option for CICS. It is similarly invoked in input format routines for other CA MICS components. It can be defined at the complex level in sharedprefix.MICS.SOURCE(#BASEXIT) or at the unit level in prefix.MICS.USER.SOURCE(#BASEXIT).

INVOCATION: For the CA MICS Analyzer Option for CICS, the _USRSEL exit gains control during the input format routine. It is invoked for each input record processed from CMF and ASG-TMON. Note that each CMF physical record can contain multiple logical transaction records. The _USRSEL exit is invoked once per CMF physical record.

ACCOUNTING INTERFACE: No interface is provided.

USES: This exit can be used to exclude records from being processed by the CA MICS Analyzer Option for CICS during the daily input step. To exclude a record, set SKIP_REC to one. For example, you can code the following to skip records that contain the SMF system ID TEST:

```
MACRO _USRSEL;
  IF ROUTINE EQ 'DYCICFMT' THEN DO;
    IF ORGSYSID EQ 'TEST' THEN SKIP_REC=1;
  END;
%
```

The CA MICS Analyzer Option for CICS provides the appropriate logic to delete the record based on the value of SKIP_REC after the _USRSEL exit is invoked.

ELEMENTS AVAILABLE: ORGSYSID - Original System ID

CODING RESTRICTIONS: Do not refer to any CA MICS-assigned data elements, such as CICSID and CICTx, when setting SKIP_REC in this exit. Extra records will be deleted as a result. For additional coding restrictions, see section 4.2.5 of the System Modification Guide.

```
+-----+
| U S R S C I D | Input Record Set CICSID Exit (USRSCID)
+-----+
```

DESCRIPTION: The %USRSCID exit lets you set the CICSID when a record's ORGSYSID/APPLID has not been defined in CICOPS. Normally, a record whose ORGSYSID/APPLID has not been defined will be discarded, and message CIC06011E will be written to the MICSLOG. By implementing the USRSCID exit, you can set the 4-byte CICSID and the record will be processed. The CICSID value supplied by the exit, in combination with the ORGSYSID, must be unique.

The daily update run will be terminated, and a MICSLOG error message will be issued if the USRSCID exit code generates a CICSID value that:

- Duplicates a ORGSYSID/CICSID combination already specified in CICOPS with a CICOPTS statement.
Error Message - CIC06173E
- Duplicates a ORGSYSID/CICSID already generated by this exit for a different APPLID.
Error Message - CIC06173E
- Duplicates a CICS Transaction Gateway CTGID specified in CICOPS with a GATEWAY APPLID statement.
Error Message - CIC07085E

The intent of this exit is to avoid discarding data for new CICS regions that were not yet defined in CICOPS. Due to additional overhead incurred in the assignment of CICSIDs through the use of this exit, it is strongly recommended that you update CICOPS and run a CICPGEN subsequent to any DAILY run that encounters data that is not defined in CICOPS. When undefined data is encountered and this exit is used to assign the CICSID, message CIC06174W will be issued, with the last line providing a sample CICOPTS statement:

```
CIC06174 USER EXIT USRSCID PROVIDED CICSID=%CICSID
CIC06174 FOR ORGSYSID=%ORGSYSID, APPLID=%CSUAPPL
```

```
CIC06174  PLEASE UPDATE CICOPS ACCORDINGLY
CIC06174  AND RUN CICPGEN BEFORE NEXT DAILY RUN
CIC06174  ...SAMPLE CICOPTS, BE SURE TO SET SELCODE...
CIC06174  CICOPTS %ORGSYSID %CICSID SMF CMF 0 %CSUAPPL ??
```

INVOCATION: The %USRSCID exit is passed control only when a record is encountered that does not find a matching ORGSYSID/APPLID in CICOPS.

ACCOUNTING INTERFACE: No interface is provided.

USES: This exit can be used to process records whose ORGSYSID/APPLID has not been defined in CICOPS. Normally, a subset of CSUAPPL would be used to set CICSID. For example:

```
%MACRO USRSCID;
  CICSID = SUBSTR(CSUAPPL,5,4) /* set CICSID to last 4 chars*/
%MEND USRSCID;
```

ELEMENTS AVAILABLE: ORGSYSID - Original System ID
 CSUAPPL - Original APPLID
 CICSID - CICS ID
 SELCODE - Select Code

CODING RESTRICTIONS: The value assigned to CICSID by the exit for a given ORGSYSID must be unique. If the ORGSYSID/CICSID combination was previously defined in CICOPS, or by the exit for a different APPLID, the run is terminated. The value assigned to CICSID must contain only alphanumeric characters or a dash (-), with a length of 1 to 4 bytes. CICSID is initialized to blanks prior to invocation of the exit. If CICSID contains blanks upon return from the exit, CIC06011E will be issued.

It is also necessary to set SELCODE if other than the default value of 0 (zero) is required. For valid SELCODE values, see section 7.3.2, CICS Processing Options (CICOPS).

For additional coding restrictions, see section 4.2.5 of the System Modification Guide.

```
+-----+
|  U S R S K I P  | User Data Skip Exit
+-----+
```

DESCRIPTION: The USRSKIP exit enables you to skip CMF records during DAY040 processing. The exit is invoked

immediately after reading the SMF record header. You may use the exit to determine whether or not the physical record, which may contain multiple CICS transaction records, should be excluded from input processing. This exit can also be used to change the value of ORGSYSID if so desired.

INVOCATION: The USRSKIP exit is invoked immediately after reading the SMF record header for CMF input.

ACCOUNTING INTERFACE: No interface is provided.

USES: Use this exit to exclude data from DAY040 input processing or to change the value of ORGSYSID.

ELEMENTS AVAILABLE: Temporary variables including SMFRTYP (SMF record type), SMFTIME (SMF record end timestamp), ORGSYSID (SMF system ID), SMFLPS (SMF section header length), and INPRECS (Physical input record number).

CODING RESTRICTIONS: Set SKIP_REC to 1 to exclude records from DAY040 processing. The following example shows how to change the value of ORGSYSID, as well as reject a record based on the value of ORGSYSID:

```
%MACRO USRSKIP;  
    IF ORGSYSID EQ 'TSTX' THEN ORGSYSID EQ 'TSTA';  
    ELSE IF ORGSYSID EQ 'TSTG' THEN SKIP_REC=1;  
%MEND USRSKIP;
```

10.2.2 Parameter-Related Exits

This section identifies the CICS parameter-related exits that you define at product installation time. Detailed descriptions of these exits are presented in Chapter 7 of this guide.

```
+-----+
| C I C A C R T | CICS Account Code Derivation Exit
+-----+
```

DESCRIPTION: The CICACRT exit derives the value of the account code elements defined in `sharedprefix.MICS.PARMS(CICACCT)`. It is a complex-level exit (it resides in the same library as CICACCT) and its definition applies to all data base units in which the CICS Analyzer is installed.

INVOCATION: The CICACRT exit gains control during the input format routine of the daily update process step DAY040. It is invoked once for each detail transaction record processed by the CICS Analyzer.

ACCOUNTING INTERFACE: No interface is provided.

USES: This exit sets the value of the CICACTX variables.

ELEMENTS AVAILABLE: All DETAIL timespan elements of the files are available.

CODING RESTRICTIONS: Refer to Section 7.2.3 of this guide.

SPECIAL NOTES: The exit code is contained in `prefix.MICS.PARMS(CICACRT)`.

SAMPLE USER EXIT: See Section 7.2.3 for a sample exit.

```
+-----+
| C I C A U R T | CICS Application Unit Definition
+-----+
```

DESCRIPTION: The CICAURT exit derives the value of the CICS Application Unit ID (CICAPU).

INVOCATION: The CICAURT exit gains control during the input format routine of the daily update process step DAY040. It is invoked once for each detail transaction record

processed by the CICS Analyzer.

ACCOUNTING INTERFACE: No interface is provided.

USES: This exit sets the value of the CICAPU data element.

ELEMENTS AVAILABLE: All DETAIL timespan elements of the files are available.

CODING RESTRICTIONS: Refer to Section 7.3.5 (CICAPU) of this guide.

SPECIAL NOTES: The exit code is contained in prefix.MICS.PARMS(CICAURT).

SAMPLE USER EXIT: See Section 7.3.6 (CICAURT) for a sample exit.

```
+-----+  
| C I C R L R T | CICS Relative Longevity Derivation Routine  
+-----+
```

DESCRIPTION: The CICRLRT exit derives the data element TRANTYPE to identify the transaction type.

INVOCATION: The CICRLRT exit gains control during the input format routine of the daily update process step DAY040. It is invoked once for each detail transaction record processed by the CICS Analyzer.

ACCOUNTING INTERFACE: No interface is provided.

USES: This exit sets the value of the TRANTYPE data element.

ELEMENTS AVAILABLE: All DETAIL timespan elements of the files are available.

CODING RESTRICTIONS: Refer to Section 7.3.7 of this guide.

SPECIAL NOTES: The exit code is contained in prefix.MICS.PARMS(CICRLRT).

SAMPLE USER EXIT: See Section 7.3.7 for a sample exit.

```
+-----+  
| C I C M S A C | CICS Multisystem Account Derivation Exit  
+-----+
```

DESCRIPTION: The CICMSAC exit allows you to override the account code values after they are derived in the input processing step. It is used to identify transactions that have been routed to other CICS regions for processing in an MRO or ISC environment.

INVOCATION: The CICMSAC exit is invoked after the input processing has completed and the detail transaction work files have been created in DAY040. It is invoked when you specify the MSACCOUNT keyword in prefix.MICS.PARMS(CICOPS).

ACCOUNTING INTERFACE: No interface is provided.

USES: This exit sets the value of the CICTx variables.

ELEMENTS AVAILABLE: All DETAIL timespan elements of the file are available.

CODING RESTRICTIONS: Refer to Section 7.3.9 of this guide.

SPECIAL NOTES: The exit code is contained in prefix.MICS.PARMS(CICMSAC).

SAMPLE USER EXIT: See Section 7.3.9 for a sample exit.

10.2.3 Output Exits

Output exits are invoked just prior to adding an observation to the DETAIL timespan of a file. Use these exits to modify the value of elements or prevent selected observations from being added to a file. In addition, the CA MICS Analyzer Option for CICS provides two special exits to output detail data to a user-defined data set. These exits specify user data set names and the length, formats, and labels associated with the data set.

```
+-----+
|  U S R S f f f |  CICS File Selection Exit
+-----+
```

DESCRIPTION: The USRSfff exit lets you modify or select observations immediately prior to output. For the CICS Information Area, the files available (fff) are CSY, CSW, CSU, CSF, CAU, CAC, CIN, and CDC. For the CTG Information Area, the file available is GSA.

INVOCATION: These exits are invoked in the information area processing phase immediately before output of the file. See Chapter 9 for more information.

ACCOUNTING INTERFACE: No interface is provided.

USES: Use this exit to modify elements or to exclude observations from further processing. For files CSY, CSF, CIN and GSA, the USRSfff exit is called early enough to modify elements or delete observations at the DETAIL level.

One cannot use _USRSfff to make changes in the DETAIL timespan for any of these files: CAC, CAU, or CSU. However, since the CSW work file feeds each of these files and the _USRSCSW exit is applied to DETAIL level observations early, we suggest that any modifications to elements or any deletion of DETAIL level observations be made in the _USRSCSW exit. See the SPECIAL NOTES for details.

ELEMENTS AVAILABLE: All elements in the file are available.

CODING RESTRICTIONS: See section 4.3.1 of the System Modification Guide.

SPECIAL NOTES:

1. To implement this exit at the complex level, code it in `sharedprefix.MICS.SOURCE(#CICEXIT)`.

2. To implement this exit at the unit level, code it in `prefix.MICS.USER.SOURCE(#CICEXIT)`.
3. The `USRSCSW` exit is called as the `CICCSW` work file is being defined. This is the best place to insert user code that sets values for elements. Also you can code an expression that sets `SKIP_REC` to 1 so that a particular observation will not be written to the CSW work file. Observations not written to CSW will not be available to the CAC, CSU, or CAU files in any timespans. In the following example, no observations with `CICSID` other than `CIC1` will reach the CAC, CSU, and CAU files.

```
MACRO _USRSCSW
  IF CICSID NE "CIC1" THEN SKIP_REC=ONE;
  %
```

4. For files CAC, CSU, and CAU, the `_USRSfff` macro can be used to modify or define `DETAIL` level element values just before they are summarized to the `DAYS` level.
5. Since the `_USRDCAC` and `_USRDCSU` exits are called just before output to the `DAYS` timespan for the CAC and CSU files, observations can be omitted from the `DAYS` timespan for the CAC and CSU files by inserting code into `_USRDCAC` or `_USRDCSU` as in the following examples:

```
MACRO _USRDCAC
  IF CICSID NE "SYS1" THEN SKIP_REC=ONE;
  _USRUCAC
  _USRJCA
  %
```

```
MACRO _USRDCSU
  IF CICSID NE "SYS1" THEN SKIP_REC=ONE;
  _USRUCSU
  _USRJCS
  %
```

A `_USRdfff` call like this has no impact on the `DETAIL` file itself whether it is on DASD or on tape. The `_USRdfff` exit is described below.

6. If you take the `TAPECSU` option for the CSU file, you can use the `USRTCSU` exit to change element values or to prevent observations before being written to the `TAPECSU.CICCSU01` file. `USRTCSU` is described below.


```
+-----+
|  U S R D f f f |  CICS User-written Cost Derivation Exit
+-----+
```

DESCRIPTION: The USRDfff exit lets you derive the fffcOST data element. The files available (fff) are CSU and CAC.

INVOCATION: These exits are invoked in the information area processing phase immediately before output of the file. See Chapter 9 for more information.

ACCOUNTING INTERFACE: No interface is provided.

USES: The exit allows elements to be passed to the routine computing the value of fffcOST.

ELEMENTS AVAILABLE: All elements in the file are available.

CODING RESTRICTIONS: See section 4.3.1 of the System Modification Guide.

```
+-----+
|  U S R X f f f |  CICS Detail Data Exit
+-----+
```

DESCRIPTION: The USRXfff exit creates a DETAIL timespan file for files that do not support the DETAIL timespan by default (due to the large volume of transaction data at most sites). The files available (fff) are CICS CSU, CAU, CAC, and CSF.

INVOCATION: This exit is invoked in the code that processes detail CICS transaction data after a sort of the detail data into the DETAIL CICfff file sequence and before the data is summarized to the DAYS level.

ACCOUNTING INTERFACE: No interface is provided.

USES: Use this exit to create a DETAIL timespan file. If you choose to create a DETAIL timespan file for the CICCSU file on tape rather than on DASD, use the USRTCSU exit. See section 10.1.4 for coding examples.

ELEMENTS AVAILABLE: All elements in the file are available.

CODING RESTRICTIONS: See section 4.3.1 of the System Modification Guide.

```
+-----+
|  U S R S D K P  |  User Data Set Keep Exit
+-----+
```

DESCRIPTION: The USRSDKP exit defines user data sets to create during CICS input file processing.

INVOCATION: The exit is invoked as part of the DATA statement that defines the data sets to be created during the initial processing of the CICS input files.

ACCOUNTING INTERFACE: No interface is provided.

USES: Use this exit to create additional SAS data sets during the initial processing of CICS input files. For example, you can use it to define your own output data sets for the accounting or exception class of the CICS monitoring data.

ELEMENTS AVAILABLE: All elements in the file are available.

CODING RESTRICTIONS: To code the exit, specify the SAS data set name to be created. For example, to create data sets named CICEXP and CICAUT in the DETAIL database, code the following:

```
MACRO _USRSDKP
  &CICX..CICEXP(KEEP=variable names)
  &CICX..CICAUT(KEEP=variable names)
%
```

Do not include a semicolon (;) at the end of the data set names.

To output observations to the files defined by the _USRSDKP exit, you can use any of the detail exits available during input processing, such as _USRSCSW, CICRLRT, and CICAUT.

```
+-----+
|  U S R S L F L  |  User Data Length, Format, Label Exit
+-----+
```

DESCRIPTION: The USRSLFL exit defines the length, formats, and labels associated with a user data set created during the initial processing of CICS input files. It is used in conjunction with the USRSDKP exit described above.

INVOCATION: The exit is invoked during the code that processes the CICS input files.

ACCOUNTING INTERFACE: No interface is provided.

USES: Use this exit to specify the length, formats, and labels for variables associated with a user-defined data set.

ELEMENTS AVAILABLE: All elements in the file are available.

CODING RESTRICTIONS: To code the exit, specify the LENGTH, FORMAT, and LABEL statements for the variables in the KEEP list as defined by the USRSDKP exit. For example:

```
MACRO _USRSLFL
LENGTH var1 var2 var3 8;
FORMAT var1 var2 var3 8.2;
LABEL var1 = "Exception Label 1"
      var2 = "Exception Label 2"
      var3 = "Exception Label 3";
%
```

```
+-----+
| U S R S C M F | CMF Accounting & Exception Data Processing
+-----+
```

DESCRIPTION: The USRSCMF exit serves two purposes. It can build CICS files in the CA MICS database using the CMF accounting and exception class transaction records. It can also be used to output performance class data to SAS files that you have defined in the USRSDKP exit.

INVOCATION: The USRSCMF exit is invoked for each transaction record input from the CMF accounting or exception class. The exit is invoked after the fields from these records have been processed by the input format routine. As a result, do not code a SAS INPUT statement in this exit.

This exit is also invoked for each transaction and global record input from the CMF performance class, which lets you manipulate the fields read from the CMF records and optionally output them to your own SAS file.

ACCOUNTING INTERFACE: No interface is provided.

USES: Use this exit to build your own CICS files or to add observations to the CA MICS CICS files using CMF accounting, exception, and performance class data.

ELEMENTS AVAILABLE: All data fields in the accounting, exception, and performance class data. See `sharedprefix.MICS.SOURCE(CICDEPEL)` for the corresponding SAS variable names.

CODING RESTRICTIONS: Do not code SAS INPUT statement in this exit, because the exit is invoked after each field in the record has been read. Since this exit is invoked for all classes of CMF data, you must check the temporary variable `CMFRTYPE` to determine the record type. The following shows the value of `CMFRTYPE` for each CMF record type:

Record Type	CMFRTYPE
-----	-
Accounting class transaction record	A
Exception class transaction record	E
Performance class global record	G
Performance class transaction record	T

The following example shows how to write to the `CICCIN` file when you have identified that a file is waiting for a VSAM buffer.

```
MACRO _USRSCMF
  SELECT (CMFRTYPE);
  WHEN ('E') DO;
    IF FCVSBWNM NE ' ' THEN DO;
      CINCODE=91;
      CINTXT=FCVSBWNM || ' WAITING FOR A VSAM BUFFER';
      OUTCICFL='OUTCIN';
      LINK OUTCIN;
    END;
  END;
  OTHERWISE;
END;
%
```

```
+-----+
| U S R S I N T | User Specified Initialization Exit
+-----+
```

DESCRIPTION: The `USRSINT` exit lets you initialize CA MICS data elements that become unique to specific regions as a result of CMF customization performed at your site. This initialization logic prevents data propagation that can occur under certain conditions in CA MICS. For example, to help reduce system overhead in your production regions, you tailor

CMF to exclude a set of monitoring fields. However, the test region continues to collect all CMF fields. When the data from the test region is processed in CA MICS followed by data from the production regions, you noticed elements in the production regions for the excluded fields contain values from the test region. To resolve this problem, you must initialize data elements for the excluded fields in the USRSINT exit.

INVOCATION: The exit is invoked during the initialization module CICINIT that is executed for fields that are normally unique to each CICS input source.

ACCOUNTING INTERFACE: No interface is provided.

USES: Use this exit to set a CA MICS data element to either missing or blanks based on the data element format. The purpose of the exit is to prevent data propagation for fields that have been excluded from selected regions as a result of CMF customization.

ELEMENTS AVAILABLE: All elements in the CICCSU and CICCSY files at the DETAIL timespan are available. You can browse sharedprefix.MICS.GENLIB(CICGENIN) for the list of data elements.

CODING RESTRICTIONS: To code the exit, assign data elements to either missing or blanks based on element format. The data elements coded in this exit should be populated directly from input fields. They should not be elements that are derived from either CICS source modules or the %fffDERV macro, since the latter derivations override any code specified here. An example of the exit is shown below:

```
%MACRO USRSINT;
  CSUSIOCT=.;
  CSUFACTY=' ';
%MEND USRSINT;
```

```
+-----+
|  U S R S T A T  |
+-----+
```

DESCRIPTION: The USRSTAT exit lets you selectively copy CICS statistics records to a sequential file on tape during the daily summarization process. The sequential file can then be used as input for statistics report programs, as IBM's DFHSTUP, without processing the entire SMF data set.

INVOCATION: The USRSTAT exit is invoked prior to output of the sequential file.

ACCOUNTING INTERFACE: No interface is provided.

USES: Use this exit to select the statistics record types or the CICS regions for which data is to be written to the sequential file.

ELEMENTS AVAILABLE: All data fields in the input records processed by the CA MICS Analyzer Option for CICS, plus the temporary numeric variable RECNUM, which indicates the statistics record type.

CODING RESTRICTIONS: Verify that the STATCOPY keyword is specified in your prefix.MICS.PARMS(CICOPS) and that CICPGEN has been executed before coding this exit.

Set SKIP_REC to 1 to exclude records from being written to the sequential file.

For example, code the following to limit output written to the sequential file to include data about storage manager task subpools (type 8), storage manager dynamic storage areas (type 9), and loader statistics (type 27):

```
%MACRO USRSTAT;  
  IF RECNUM = 8 OR RECNUM = 9 OR RECNUM = 27  
    THEN SKIP_REC=0;  
    ELSE SKIP_REC=1;  
%MEND USRSTAT;
```

```
+-----+  
| U S R T C S U | TAPECSU Data Selection Exit  
+-----+
```

DESCRIPTION: The USRTCSU exit selects data by CICSID or other criteria before the DETAIL CICCUSU01 file is written to tape during the daily summarization process.

INVOCATION: The USRTCSU exit is invoked prior to output of the TAPECSU.CICCUSU01 file in SOURCE(#CSUDSUM).

ACCOUNTING INTERFACE: No interface is provided.

USES: Use this exit to limit the amount of data that is written to the TAPECSU.CICCUSU01 file.

ELEMENTS AVAILABLE: All elements in the DETAIL CICC01 file.

CODING RESTRICTIONS: Verify that the TAPECSU keyword is specified in your prefix.MICS.PARMS(CICOPS) and that CICIPGEN has been executed before coding this exit.

For example, code the following to limit output written to the TAPECSU.CICC01 file by excluding data about all CICS regions other than CIC1 and CICIP.

```
%MACRO USRTCSU;
  IF CICSID NE 'CIC1' AND CICSID NE 'CICIP' THEN
    SKIP_REC=1;
%MEND USRTCSU;
```

```
+-----+
| U S R U J f f | Accounting & Chargeback Exits
| C I C J f f f |
+-----+
```

DESCRIPTION: These exits provide the linkage into the CA MICS Accounting and Chargeback Option. The files available (fff) are the CICC01 and CICCAC files.

INVOCATION: These exits are invoked in the daily summarization #fffDSUM routines, where fff is CSU or CAC, because the data is summarized at the DAYS timespan.

ACCOUNTING INTERFACE: These exits provide the interface to the CA MICS Accounting and Chargeback Option.

USES: The exit allows elements to be passed to the CA MICS Accounting and Chargeback Option.

ELEMENTS AVAILABLE: All elements in the file are available.

CODING RESTRICTIONS: See section 4.3.1 of the System Modification Guide.

Appendix A: MESSAGES

Messages

This appendix lists all messages generated by the CA MICS Analyzer Option for CICS. Some messages are generated during the processing of the control statements, while others are caused by various conditions in the data. The messages are listed in ascending numerical sequence and include full text of the message, the type, the reason for the message, appropriate user action, and applicable document references.

The following type codes are used to categorize the messages:

1. Information: A note to the user that documents a CICS option or potentially important feature in the data.
2. Warning: A condition of either the data or the control statements that does not affect the operation of the CA MICS Analyzer Option for CICS, but can lead to unexpected results.
3. Error: A problem has been encountered with a control statement that will prevent a successful run of the CA MICS Analyzer Option for CICS. Execution is stopped after all control statements are processed.
4. Termination: A critical problem has been encountered with the data that requires the CA MICS Analyzer Option for CICS to terminate processing immediately.

```
+-----+
| C I C 0 2 0 0 1 |
+-----+
```

TEXT: No data has passed through initial selection.
Check file and selection criteria validity.
Run terminates with a User 998 ABEND code.

REASON: The selection criteria specified for this MICF inquiry excluded all data.

TYPE: Error

ACTION: Review data selection criteria; rerun inquiry.

REFERENCES: Chapter 3

```
+-----+
| C I C 0 5 0 1 0 |
+-----+
```

TEXT: INPUTCIC---> %LINE
DDNAME INPUTSMF NOT VALID

TYPE: Error

REASON: INPUTSMF is a reserved word and cannot be used as a ddname for a non-SMF input data set.

ACTION: Review and correct the ddname specification in members CICOPS and INPUTCIC of prefix.MICS.PARMS. Rerun job.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

```
+-----+
| C I C 0 5 0 1 1 |
+-----+
```

TEXT: INPUTCIC---> %LINE
DDNAME %INCDDNME LENGTH NOT 1 TO 8

TYPE: Error

REASON: Invalid ddname is specified in prefix.MICS.PARMS(INPUTCIC). The ddname must be one to eight characters in length.

ACTION: Review and correct the ddname specification in members CICOPS and INPUTCIC of prefix.MICS.PARMS. Rerun job.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

```
+-----+  
| C I C 0 5 0 1 2 |  
+-----+
```

TEXT: INPUTCIC---> %LINE
DDNAME %INCDDNME PREVIOUSLY SPECIFIED

TYPE: Error

REASON: A ddname is specified twice in prefix.MICS.PARMS(INPUTCIC).

ACTION: Eliminate the duplicate ddname from prefix.MICS.PARMS(INPUTCIC). Rerun job.

REFERENCES: Section 7.3.4

```
+-----+  
| C I C 0 5 0 1 3 |  
+-----+
```

TEXT: THE NUMBER OF DDNAMES SPECIFIED IN INPUTCIC EXCEEDS CHECKPOINT LIMIT. THE MAXIMUM IS %CKPTCNT ENTRIES.

TYPE: Error

REASON: The number of ddnames defined in prefix.MICS.PARMS(INPUTCIC) has exceeded the maximum allowed.

ACTION: Reduce the number of ddnames by concatenating multiple input data sets to one ddname. Modify ddname specifications in members CICOPS and INPUTCIC of prefix.MICS.PARMS. Rerun job.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

```
+-----+
```

```
| C I C 0 5 0 2 0 |
+-----+
```

TEXT: CICOPS---> %LINE
KEYWORD NOT RECOGNIZED. KEYWORD=%KEYWRD01

TYPE: Error

REASON: An invalid keyword is specified in
prefix.MICS.PARMS(CICOPS). Valid keywords are
MSACCOUNT, WORK, RESP, TRANSLATE, CICOPTS, and
TAPECSU.

ACTION: Correct the keyword specification in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 2 1 |
+-----+
```

TEXT: MULTIPLE MSACCOUNT STATEMENTS

TYPE: Warning

REASON: Multiple MSACCOUNT statements are specified in
prefix.MICS.PARMS(CICOPS); only one is used.

ACTION: Warning only. No action required.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 2 2 |
+-----+
```

TEXT: MULTIPLE WORK STATEMENTS

TYPE: Error

REASON: Multiple WORK statements are specified in
prefix.MICS.PARMS(CICOPS), but one is allowed.

ACTION: Remove all but one WORK statement from
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 2 3 |  
+-----+
```

TEXT: CICOPS---> %LINE
 WORK FILE PAIRS NOT NUMERIC

TYPE: Error

REASON: Invalid WORK statement is found in
 prefix.MICS.PARMS(CICOPS). The work file pair
 parameter must be a number from 1 to 9.

ACTION: Correct the WORK statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 2 4 |  
+-----+
```

TEXT: CICOPS---> %LINE
 WORK FILE PAIRS NOT 1 TO 9

TYPE: Error

REASON: Invalid WORK statement is found in
 prefix.MICS.PARMS(CICOPS). The work file pair
 parameter must be a number from 1 to 9.

ACTION: Correct the WORK statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 2 5 |  
+-----+
```

TEXT: MULTIPLE RESP STATEMENTS

TYPE: Error

REASON: Multiple RESP statements are specified in
 CICOPS, but only one is permitted.

ACTION: Remove all but one RESP statement from
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 2 6 |  
+-----+
```

TEXT: CICOPS---> %LINE
LESS THAN SEVEN RESPONSE VALUES SPECIFIED

TYPE: Error

REASON: Invalid RESP statement is specified in
prefix.MICS.PARMS(CICOPS). The statement must
define seven response values.

ACTION: Correct the RESP statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 2 7 |  
+-----+
```

TEXT: CICOPS---> %LINE
RESPONSE VALUE NUMBER %WRKINX2 IS NOT NUMERIC

TYPE: Error

REASON: Invalid RESP statement is specified in
prefix.MICS.PARMS(CICOPS). The statement must
define seven numeric response values.

ACTION: Correct the RESP statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 2 8 |  
+-----+
```

TEXT: CICOPS---> %LINE
RESPONSE VALUE NUMBER %WRKINX2 IS > 3600

TYPE: Error

REASON: Invalid RESP statement is specified in
prefix.MICS.PARMS(CICOPS). The response value
cannot exceed 3600 seconds.

ACTION: Correct the RESP statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 2 9 |  
+-----+
```

TEXT: CICOPS---> %LINE
RESPONSE VALUES NOT ASCENDING - SEE RESPONSE
VALUE %WRKINX2

TYPE: Error

REASON: Invalid RESP statement is specified in
prefix.MICS.PARMS(CICOPS). The statement must
define seven numeric response values in
ascending order.

ACTION: Correct the RESP statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 3 0 |  
+-----+
```

TEXT: CICOPS---> %LINE
MORE THAN 7 RESPONSE VALUES SPECIFIED

TYPE: Error

REASON: Invalid RESP statement is specified in
prefix.MICS.PARMS(CICOPS). The statement
contains more than seven response values.

ACTION: Correct the RESP statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 3 1 |  
+-----+
```

TEXT: CICOPS---> %LINE
 OLD AND NEW ID MUST BE SPECIFIED

TYPE: Error

REASON: Invalid TRANSLATE statement is specified in
 prefix.MICS.PARMS(CICOPS). Neither an old nor
 new transaction ID is specified in this
 TRANSLATE statement.

ACTION: Correct the TRANSLATE statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 3 2 |  
+-----+
```

TEXT: INVALID SPECIAL CHARACTERS IN CICSID:
 %KEYWRD03
 THIS WILL MAKE CICSREL AND CICSMMVER
 BLANK IN THE CICCXY FILE.

TYPE: Warning

REASON: Regions defined as monitor type 'MON' in
 CICOPS cannot contain special characters. We
 use the CICSID for 'MON' regions as a macro
 variable in order to populate the CICSREL and
 CICSMMVER data elements in the CICCXY File. If
 they contain special characters, the macro
 variable definition will fail and CICSREL and
 CICSMMVER will not get populated.

ACTION: Change the CICSID in the CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). Rerun CICPGEN.

REFERENCES: Section 7.3.2


```
+-----+  
| C I C 0 5 0 3 3 |  
+-----+
```

TEXT: CICOPS---> %LINE
 LENGTH OF OLD ID GREATER THAN 8 CHARACTERS.
 ID=%KEYWRD02

TYPE: Error

REASON: Invalid TRANSLATE statement is specified in
 prefix.MICS.PARMS(CICOPS). The original
 transaction ID cannot be longer than eight
 characters.

ACTION: Correct the TRANSLATE statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 3 4 |  
+-----+
```

TEXT: CICOPS---> %LINE
 LENGTH OF NEW ID GREATER THAN 8 CHARACTERS.
 ID=%KEYWRD03

TYPE: Error

REASON: Invalid TRANSLATE statement is specified in
 prefix.MICS.PARMS(CICOPS). The new
 transaction ID cannot be longer than eight
 characters.

ACTION: Correct the TRANSLATE statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 3 5 |  
+-----+
```

TEXT: CICOPS---> %LINE
 OLD ID WAS PREVIOUSLY SPECIFIED

TYPE: Error

REASON: Invalid TRANSLATE statement is found in
prefix.MICS.PARMS(CICOPS). The original ID
has been specified in a prior TRANSLATE
statement.

ACTION: Remove or correct the TRANSLATE statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 3 6 |
+-----+
```

TEXT: CICOPS---> %LINE
OLD ID IS NOT AN ATTENTION IDENTIFIER SO
LENGTH MUST BE 8 CHARACTERS ID=%KEYWRD02

TYPE: Error

REASON: Invalid TRANSLATE statement is specified in
prefix.MICS.PARMS(CICOPS). The length of the
original ID must be eight characters, unless
the original ID is one of the following
identifiers: PA1, PA2, PA3, PF1-PF24, ENTER,
LPA, or OPID.

ACTION: Correct the TRANSLATE statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 3 8 |
+-----+
```

TEXT: CICOPS---> %LINE
OLD ID IS NOT AN ATTENTION IDENTIFIER - IT
MUST BE IN HEXADECIMAL FORMAT ID=%KEYWRD02

TYPE: Error

REASON: Invalid TRANSLATE statement is specified in
prefix.MICS.PARMS(CICOPS). The original ID
must be specified in hexadecimal format,

unless it is one of the following identifiers:
PA1, PA2, PA3, PF1-PF24, ENTER, LPA, or OPID.

ACTION: Correct the TRANSLATE statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 3 9 |  
+-----+
```

TEXT: TRANSLATION LIMIT OF 300 HAS BEEN EXCEEDED

TYPE: Error

REASON: More than 300 TRANSLATE statements have been
specified in prefix.MICS.PARMS(CICOPS).

ACTION: Specify a maximum of 300 TRANSLATE statements
in prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 4 0 |  
+-----+
```

TEXT: CICOPS---> %LINE
SYSID NOT 1 TO 4 CHARACTERS IN LENGTH.
SYSID=%KEYWRD02

TYPE: Error

REASON: Invalid SYSID value is specified in the
CICOPTS statement in
prefix.MICS.PARMS(CICOPS). The SYSID must be
one to four characters in length.

ACTION: Correct the CICOPTS statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 4 1 |  
+-----+
```

TEXT: CICOPS---> %LINE
 CICSID NOT 1 TO 4 CHARACTERS IN LENGTH.
 CICSID=%KEYWRD03

TYPE: Error

REASON: Invalid CICSID value is specified in the
 CICOPTS statement prefix.MICS.PARMS(CICOPS).
 The CICSID must be one to four characters in
 length.

ACTION: Correct the CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 4 2 |
+-----+
```

TEXT: CICOPS---> %LINE
 MONITOR TYPE MUST BE CMF OR MON.
 MON=%KEYWRD05

TYPE: Error

REASON: Invalid monitor source is specified in the
 CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). CA MICS currently
 supports CMF and ASG-TMON. They are specified
 by the words CMF and MON.

ACTION: Correct the CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 4 3 |
+-----+
```

TEXT: CICOPS---> %LINE
 SYSID/CICSID PREVIOUSLY DEFINED.
 SYSID=%KEYWRD02 CICSID=%KEYWRD03

TYPE: Error

REASON: Duplicate SYSID/CICSID combinations are defined in prefix.MICS.PARMS(CICOPS).

ACTION: Correct the CICOPTS statement in prefix.MICS.PARMS(CICOPS). Ensure each SYSID/CICSID combination is unique. Rerun job.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 4 4 |
+-----+
```

TEXT: CICOPS---> %LINE
DDNAME NOT 1 TO 8 CHARACTERS IN LENGTH.
DDNAME=%KEYWRD04

TYPE: Error

REASON: Invalid ddname is specified in CICOPTS statement in prefix.MICS.PARMS(CICOPS). The ddname must be one to eight characters in length.

ACTION: Review and correct the ddname specifications in members CICOPS and INPUTCIC of prefix.MICS.PARMS. Rerun job.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

```
+-----+
| C I C 0 5 0 4 5 |
+-----+
```

TEXT: CICOPS---> %LINE
WHEN A GENERIC SYSID IS SPECIFIED, DDNAME MUST BE SMF OR MON

TYPE: Error

REASON: Invalid ddname specified in CICOPTS statement in prefix.MICS.PARMS(CICOPS). When four question marks (????) are used to specify generic SYSID, the associated ddname must be either SMF or MON.

ACTION: Review and correct the ddname specification in members CICOPS, INPUTCIC, and INPUTSMF of prefix.MICS.PARMS. Rerun job.

REFERENCES: Section 7.3.2, 7.3.3, and 7.3.4

```
+-----+
| C I C 0 5 0 4 6 |
+-----+
```

TEXT: CICOPS---> %LINE
DDNAME NOT PERMITTED. DDNAME=%KEYWRD04

TYPE: Error

REASON: Invalid ddname specified in CICOPTS statement in prefix.MICS.PARMS(CICOPS). INPUTSMF and INPUTMON are both reserved words that cannot be used to specify the ddname.

ACTION: Review and correct the ddname specifications in members CICOPS, INPUTCIC, and INPUTSMF of prefix.MICS.PARMS. Rerun job.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

```
+-----+
| C I C 0 5 0 4 7 |
+-----+
```

TEXT: CICOPS---> %LINE
DDNAME NOT PERMITTED FOR MONITOR.
MONITOR=%KEYWRD05 DDNAME=%KEYWRD04

TYPE: Error

REASON: Invalid ddname specified in CICOPTS statement in prefix.MICS.PARMS(CICOPS). When the monitor type is CMF, the ddname cannot be MON.

ACTION: Review and correct ddname specification in members CICOPS, INPUTSMF, and INPUTCIC of prefix.MICS.PARMS. Rerun job.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

```
+-----+
| C I C 0 5 0 4 8 |
+-----+
```

TEXT: CICOPS---> %LINE
 NULL APPLID/SYSIDNT NOT ALLOWED FOR
 DDNAME=%KEYWRD04

TYPE: Error

REASON: Invalid APPLID or system identification
 (sysidnt) is specified in CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). When the ddname is
 SMF or MON, the associated APPLID or sysidnt
 cannot be NULL.

ACTION: Correct the CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). For ddname SMF,
 specify the eight-byte APPLID. For ddname
 MON, specify the four-byte sysidnt.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 4 9 |
+-----+
```

TEXT: CICOPS---> %LINE
 DDNAME|APPLID/SYSIDNT NOT UNIQUE.
 DDNAME=%KEYWRD04 APPLID/SYSIDNT=%KEYWRD07

TYPE: Error

REASON: Multiple ddname/applid or ddname/sysidnt
 combinations are specified in CICOPTS
 statements in prefix.MICS.PARMS(CICOPS).

ACTION: Correct the CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). For CMF
 processing, ensure the ddname/applid
 combination is unique. For ASG-TMON
 processing, ensure the ddname/sysidnt
 combination is unique. Rerun job.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 5 0 |
+-----+
```

+-----+

TEXT: CICOPS---> %LINE
 GMT OFFSET IS NOT NUMERIC. VALUE=%KEYWRD06

TYPE: Error

REASON: Invalid GMT offset is specified in CICOPTS
 statement in prefix.MICS.PARMS(CICOPS). The
 GMT offset must be numeric in the range of -13
 to +13.

ACTION: Correct the CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

+-----+

| C I C 0 5 0 5 1 |

+-----+

TEXT: CICOPS---> %LINE
 GMT OFFSET NOT IN RANGE -13 TO +13 HOURS.
 OFFSET=%VALUE

TYPE: Error

REASON: Invalid GMT offset is specified in CICOPTS
 statement in prefix.MICS.PARMS(CICOPS). The
 GMT offset must be numeric in the range of -13
 to +13.

ACTION: Correct the CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

+-----+

| C I C 0 5 0 5 2 |

+-----+

TEXT: CICOPS---> %LINE
 LESS THAN 7 PARAMETERS SPECIFIED AFTER CICOPTS

TYPE: Error

REASON: One or more parameters are missing from the
 CICOPTS statement in

prefix.MICS.PARMS(CICOPS). Because the parameters are positional, it is not possible to determine which parameters have been omitted.

ACTION: Correct the CICOPTS statement in prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 5 3 |  
+-----+
```

TEXT: CICOPS---> %LINE
INTERVAL COLLECTION TIME IS NOT NUMERIC.
INTERVAL=%KEYWRD08

TYPE: Error

REASON: Invalid sample interval is specified in the CICOPTS statement in prefix.MICS.PARMS(CICOPS). The sample interval time is in minutes and must be numeric in the range 1 to 999.

ACTION: Correct the CICOPTS statement in prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 5 4 |  
+-----+
```

TEXT: CICOPS---> %LINE
INTERVAL COLLECTION TIME NOT IN RANGE 1-999
MINUTES. INTERVAL=%VALUE

TYPE: Error

REASON: Invalid sample interval is specified in the CICOPTS statement in prefix.MICS.PARMS(CICOPS). The sample interval time is in minutes and must be numeric in the range 1 to 999.

ACTION: Correct the CICOPTS statement in

prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 5 5 |
+-----+
```

TEXT: CICOPS---> %LINE
 THE COMBINED NUMBER OF CICOPTS AND GATEWAY
 APPLID STATEMENTS EXCEED THE CHECKPOINT LIMIT.
 THE MAXIMUM IS %CKPTCNT ENTRIES.

TYPE: Error

REASON: The combined total number of CICOPTS
 statements and GATEWAY APPLID statements
 exceed the maximum allowed, which is displayed
 with the %CKPTCNT variable.

ACTION: Either reduce the number of CICOPTS and/or
 GATEWAY APPLID statements in
 prefix.MICS.PARMS(CICOPS), or increase the
 maximum number of checkpoint entries allowed.
 Then rerun job.

REFERENCES: Section 7.3.2 - CICOPTS statement
 Section 7.3.6 - GATEWAY APPLID statement
 PIOM Section 2.3.2.4 - Increase Max Checkpoint

```
+-----+
| C I C 0 5 0 5 6 |
+-----+
```

TEXT: NO INPUTCIC STATEMENT FOR DDNAME %INCDDNME IN
 CICOPS

TYPE: Error

REASON: The ddname specified in the CICOPTS statement
 in prefix.MICS.PARMS(CICOPS) has no
 corresponding match in
 prefix.MICS.PARMS(INPUTCIC).

ACTION: Correct the ddname specification in members
 CICOPS and INPUTCIC of prefix.MICS.PARMS.
 Rerun job.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

```
+-----+  
| C I C 0 5 0 5 7 |  
+-----+
```

TEXT: ERROR(S) ENCOUNTERED DURING UNIT GENERATION
(PGEN) - EXECUTION ABNORMALLY TERMINATED

TYPE: Error

REASON: At least one error condition was specified
while processing CICPGEN parameter statements.

ACTION: Correct the errors as indicated by the error
messages in MICSLOG. Rerun job.

REFERENCES: Section 7.3.1

```
+-----+  
| C I C 0 5 0 5 8 |  
+-----+
```

TEXT: CICOPS---> %LINE
%MON IS NOT SUPPORTED

TYPE: Error

REASON: OPTIONS statement for PAII or CMF15 is
specified in prefix.MICS.PARMS(CICOPS).
CA MICS support for PAII and CICS 1.5 have
been dropped.

ACTION: Remove OPTIONS statement for PAII and CMF15
from prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 5 9 |  
+-----+
```

TEXT: NULL APPLID OR BLANK SYSIDNT IS SPECIFIED.
NULL APPLID OR SYSIDNT WILL BE GENERATED.

TYPE: Warning

REASON: NULL applid or blank sysidnt is specified in

prefix.MICS.PARMS(CICOPS).

ACTION: Review prefix.MICS.PARMS(CICOPS) to verify that NULL applid or blank sysidnt is the intended specification.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 6 0 |  
+-----+
```

TEXT: CICOPS---> %LINE
ALL CICOPTS STATEMENTS ENCOUNTERED.
CONVERSION MAY HAVE BEEN EXECUTED.

TYPE: Information

REASON: Prefix.MICS.PARMS(CICOPS) does not contain any OPTION/CMFOPS/LANOPS statement. This indicates that the CICOPS conversion utility program CICOPSCV has already been executed or that it is not needed.

ACTION: None.

REFERENCES: N/A

```
+-----+  
| C I C 0 5 0 6 1 |  
+-----+
```

TEXT: ERROR(S) ENCOUNTERED DURING CICOPS
CONVERSION - EXECUTION ABNORMALLY TERMINATED

TYPE: Error

REASON: One or more errors encountered while executing the CICOPS conversion utility program CICOPSCV.

ACTION: Correct errors as indicated by messages in the MICSLOG. Rerun job.

REFERENCES: N/A

```
+-----+
```

| C I C 0 5 0 6 2 |
+-----+

TEXT: CMF INPUT PROCESSOR ACTIVE

TYPE: Information

REASON: CMF has been specified as the source of the
CICS monitor data.

ACTION: None.

REFERENCES: Section 7.3.2

+-----+
| C I C 0 5 0 6 3 |
+-----+

TEXT: MONITOR INPUT PROCESSOR ACTIVE

TYPE: Information

REASON: ASG-TMON has been specified as the source of
the CICS monitor data.

ACTION: None.

REFERENCES: Section 7.3.2

+-----+
| C I C 0 5 0 6 5 |
+-----+

TEXT: USING %WORKCT WORK FILE PAIRS

TYPE: Information

REASON: The indicated number of WORK file pairs will
be used during DAY040 processing.

ACTION: None.

REFERENCES: Section 7.3.2

+-----+
| C I C 0 5 0 6 6 |
+-----+

TEXT: MSACCOUNT EXIT ACTIVE

TYPE: Information

REASON: The Multisystem Accounting Exit is active.

ACTION: None.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 6 7 |  
+-----+
```

TEXT: CICOPS---> %LINE
DDNAME ASSOCIATED WITH MORE THAN ONE MONITOR.
DDNAME=%KEYWRD04 MONITORS=%KEYWRD05 %OLDMON

TYPE: Error

REASON: Invalid CICOPTS statement is specified in
prefix.MICS.PARMS(CICOPS). The ddname in
error is specified for more than one type of
monitor. Different types of monitor data
cannot be concatenated to the same input DD.

ACTION: Review and correct the ddname specification in
members CICOPS, INPUTCIC, and INPUTSMF in
prefix.MICS.PARMS. Rerun job.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

```
+-----+  
| C I C 0 5 0 6 8 |  
+-----+
```

TEXT: NO RESPONSE DISTRIBUTION THRESHOLD STATEMENT

TYPE: Error

REASON: No RESP statement is specified in
prefix.MICS.PARMS(CICOPS). One is required.

ACTION: Add a valid RESP statement to
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 6 9 |  
+-----+
```

TEXT: CICOPS---> %LINE
 GMT OFFSET MUST BE 0 WHEN MONITOR IS %KEYWRD05

TYPE: Error

REASON: Invalid GMT offset is specified in the CICOPTS
 statement in prefix.MICS.PARMS(CICOPS). When
 input is ASG-TMON, the GMT offset must be 0.

ACTION: Correct the CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 7 0 |  
+-----+
```

TEXT: DDNAME %CURDDN CAN ONLY APPEAR ONCE
 BECAUSE IT SPECIFIES APPLID/SYSIDNT = NULL

TYPE: Error

REASON: Invalid CICOPTS statement is specified in
 prefix.MICS.PARMS(CICOPS). When NULL applid
 or system identification (sysidnt) is
 specified for a CICS region, the associated
 ddname must be unique.

ACTION: Correct the CICOPTS statement in
 prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 5 0 7 1 |  
+-----+
```

TEXT: NO CICOPS STATEMENT FOR INPUTCIC DDNAME
 %CURDDN

TYPE: Warning

REASON: Prefix.MICS.PARMS(INPUTCIC) contains input
ddname that is not defined in
prefix.MICS.PARMS(CICOPS).

ACTION: Review prefix.MICS.PARMS(INPUTCIC) and remove
any JCL that causes unnecessary data
allocation during DAY040 processing.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

```
+-----+
| C I C 0 5 0 7 2 |
+-----+
```

TEXT: CICOPS---> %LINE
INVALID SELCODE VALUE = %KEYWORD09
VALID VALUES ARE 0, 2, 4, AND 6

TYPE: Error

REASON: Invalid selection code for CMF processing is
specified in the CICOPTS statement in
prefix.MICS.PARMS(CICOPS). Valid selection
codes are 0, 2, 4, and 6.

ACTION: Correct the CICOPTS statement in
prefix.MICS.PARMS(CICOPS). Rerun job.

REFERENCES: Section 7.3.2

```
+-----+
| C I C 0 5 0 7 3 |
+-----+
```

TEXT: %PGM PROGRAM HAS ALREADY BEEN RUN.
JOB ABORTED.

TYPE: Error

REASON: Job has already been executed. It must not
run more than once. This is to prevent any
loss of data that was saved by the previous
job run.

ACTION: None.

REFERENCES: N/A


```
+-----+  
| C I C 0 5 0 7 4 |  
+-----+
```

TEXT: DATA DICTIONARY CONVERSION PROGRAM HAS NOT
BEEN RUN. JOB ABORTED.

TYPE: Error

REASON: The data dictionary conversion program
CICDCCV must run first, prior to the
execution of the dictionary generation job
CICDCGN.

ACTION: Follow the CIC4300 checklist and run the
CICDCCV and CICDCGN jobs in the correct
sequence.

REFERENCES: N/A

```
+-----+  
| C I C 0 5 0 7 5 |  
+-----+
```

TEXT: APPLID IS NOT DEFINED FOR CICSID "%CICSID" IN
PARMS(CICOPS).
PLEASE SUPPLY USING PSP.PARMS(CICDCGN).

TYPE: Error

REASON: The APPLID for this CICSID is not defined in
prefix.MICS.PARMS(CICOPS). The APPLID is
required by the data dictionary generation
program CICDCGN to generate the data
dictionary records in the DETAIL.CICDC01
file.

ACTION: Specify the APPLID for the CICS region in
sharedprefix.MICS.PSP.PARMS(CICDCGN). Rerun
job.

REFERENCES: N/A

```
+-----+  
| C I C 0 5 0 7 6 |  
+-----+
```

TEXT: DATA DICTIONARY RECORDS GENERATED FOR
CICSID "%CICSID", APPLID "%CDCAPPL",
CLASS "%CDCCLASS".

TYPE: Information

REASON: The data dictionary generation program
CICDCGN has generated data dictionary records
for the specified regions.

ACTION: None.

REFERENCES: N/A

```
+-----+  
| C I C 0 5 0 7 7 |  
+-----+
```

TEXT: %PGM PROGRAM ENDED SUCCESSFULLY.

TYPE: Information

REASON: The job has ended successfully.

ACTION: None.

REFERENCES: N/A

```
+-----+  
| C I C 0 5 0 7 9 |  
+-----+
```

TEXT: GMT OFFSET %GMT ENCOUNTERED FOR MONITOR.
CHANGED TO ZERO BECAUSE GMT MUST BE ZERO FOR
MONITOR.

TYPE: Warning

REASON: GMT offset must be zero for monitor
processing.

ACTION: None.

REFERENCES: N/A

```
+-----+  
| C I C 0 5 0 8 0 |  
+-----+
```

+-----+

TEXT: DETAIL.CICCDC01 CONTAINS ZERO OBSERVATION.
 DATA DICTIONARY CONVERSION IS NOT PERFORMED.

TYPE: Warning

REASON: Data dictionary conversion is not performed
 because file DETAIL.CICCDC01 is empty, which
 is normal if your data source is ASG-TMON.

ACTION: None.

REFERENCES: N/A

+-----+

| C I C 0 5 0 8 3 |

+-----+

TEXT: STATCOPY OPTION IS ACTIVE. CICS/ESA
 STATISTICS RECORDS WILL BE COPIED TO A
 SEQUENTIAL FILE ON TAPE

TYPE: Information

REASON: STATCOPY option was specified in
 prefix.MICS.PARMS(CICOPS).

ACTION: No action required.

REFERENCES: Section 7.3.2

+-----+

| C I C 0 5 0 8 4 |

+-----+

TEXT: ALLOCATION FAILURE, DD=%DDNAME, DSN=%DSNAME,
 RETURN CODE WAS %SVCC %INFOC
 RERUN CICPGEN ANOTHER TIME TO REMOVE UNDEFINED
 REGIONS FROM THE TMON SUSPEND FILE
 PROCESSING WILL CONTINUE

TYPE: Information

REASON: CICPGEN attempted to allocate the DETAIL
 timespan but the dynamic allocation failed.
 This was an attempt to remove any obsolete
 observations from the DETAIL.CICCSX01 file,

for any TMON regions that are no longer defined in prefix.MICS.PARMS(CICOPS). The allocation return codes are listed in the error message.

ACTION: See the JES message log for more information. If the data set was in use, rerun the CICPGEN at another time. If unable to determine the cause, contact CA Technical Support for assistance.

REFERENCES: IBM's Authorized Assembler Programming Guide for an explanation of the error codes.

```
+-----+  
| C I C 0 5 0 8 5 |  
+-----+
```

TEXT: "VERIFY RELEASE ABORT" IS ACTIVE. THE FIRST RECORD FROM AN UNSUPPORTED RELEASE WILL CAUSE THE DAY040 OR THE INCR040 STEP TO ABEND WITH A U0998.

TYPE: Informational

REASON: CICPGEN issues this message when the "VERIFY RELEASE ABORT" option is taken in the prefix.MICS.PARMS(CICOPS). With this option, the DAY040 or the INCR040 will fail on the first encounter with a type 110 record or with an ASG-TMON (TCE) record not supported in CA MICS.

ACTION: No action is required here unless you would like to have the DAY040 or the INCR040 step continue processing without taking the ABEND. In this case, code

VERIFY RELEASE NOABORT

in prefix.MICS.PARMS(CICOPS) and run prefix.MICS.CNTL(CICPGEN) in each CA MICS unit that processes CICS input.

REFERENCES: Section 7.3.2.13

```
+-----+  
| C I C 0 5 0 8 6 |  
+-----+
```

TEXT: "VERIFY RELEASE NOABORT" IS ACTIVE. IF A RECORD FROM AN UNSUPPORTED RELEASE IS READ, THE DAY040 OR INCR040 WILL CONTINUE PROCESSING.

TYPE: Informational

REASON: CICPGEN issues this message when the "VERIFY RELEASE NOABORT" option is taken in the prefix.MICS.PARMS(CICOPS). With this option, the DAY040 or the INCR040 will continue processing even though unsupported type 110 or ASG-TMON (TCE) records are encountered.

ACTION: No action is required here unless you would like to have the DAY040 or the INCR040 step ABEND on the first encounter with an unsupported input record. In this case, code

VERIFY RELEASE ABORT

in prefix.MICS.PARMS(CICOPS) and run prefix.MICS.CNTL(CICPGEN) in each unit that processes CICS input.

REFERENCES: Section 7.3.2.13

```
+-----+  
| C I C 0 5 0 8 7 |  
+-----+
```

TEXT: VERIFY RELEASE OPTION INCORRECT. PLEASE CORRECT IT OR DELETE IT AND RETRY THE CICPGEN JOB.

TYPE: Error

REASON: CICPGEN issues this error message when the

VERIFY RELEASE ABORT/NOABORT

option has been coded incorrectly in the prefix.MICS.PARMS(CICOPS) member prior to running the prefix.MICS.CNTL(CICPGEN).

ACTION: Correct the VERIFY RELEASE statement with the ABORT or NOABORT option. Otherwise, delete this statement altogether and retry the

CICPGEN. If there is no VERIFY RELEASE statement in the CICOPS member, the default option is NOABORT.

REFERENCES: Section 7.3.2.13

```
+-----+  
| C I C 0 5 0 9 0 |  
+-----+
```

TEXT: MONITOR/ESA INPUT PROCESSOR ACTIVE

TYPE: Information

REASON: ASG-TMON has been specified as the source of the CICS monitor data.

ACTION: None.

REFERENCES: Section 7.3.2

```
+-----+  
| C I C 0 6 0 1 1 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID APPLID=%CSUAPPL
INPUT NOT DEFINED IN P.PARMS(CICOPS)
INPUT RECORD REJECTED

TYPE: Error

REASON: The CICS region is not defined in unit prefix
OR the CICS has been defined but CICPGEN
either was not run or was run but did not
complete successfully.

ACTION: Define the region in prefix.MICS.PARMS(CICOPS)
and submit prefix.MICS.CNTL(CICPGEN) if data
is to be processed by CA MICS. Otherwise,
increase the CIC06011 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition.

REFERENCES: Sections 7.3.1, 7.3.2, 7.3.8, and 9.1.2.

```
+-----+  
| C I C 0 6 0 1 2 |  
+-----+
```

+-----+

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID
MORE THAN ONE PRODUCT SECTION - FIELD
SMFNPS AT COLUMN %PTR IS > 1
INPUT RECORD REJECTED

TYPE: Error

REASON: Unexpected CMF record format.

ACTION: Increase the CIC06012 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart DAY040.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

+-----+

| C I C 0 6 0 1 3 |

+-----+

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID APPLID=%CSUAPPL
INPUT RECORD LENGTH LESS THAN 85
RECORD MUST CONTAIN AT LEAST AN SMF HEADER,
PRODUCT SECTION, AND ONE SECTION
HEADER/DESCRIPTOR RECORD LENGTH IS %CICLEN
INPUT RECORD REJECTED

TYPE: Error

REASON: Truncated CMF record.

ACTION: Increase the CIC06013 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart DAY040.

Dump the record in error by using the job
CICILIST in prefix.MICS.CNTL. Consult your
CICS systems programmer to determine origin of
truncation.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

+-----+

```
| C I C 0 6 0 1 4 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
ATTEMPTING TO PROCESS SECTION HEADER AT
COLUMN %PTR
RECORD IS NOT LONG ENOUGH TO CONTAIN SECTION
HEADER
INPUT RECORD REJECTED

TYPE: Error

REASON: Truncated CMF record.

ACTION: Increase the CIC06014 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart DAY040.

Dump the record in error by using the job
CICILIST in prefix.MICS.CNTL. Consult your
CICS systems programmer to determine origin of
truncation.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+
| C I C 0 6 0 1 5 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
RECORD CLASS NOT 2 (ACCOUNTING), 3
(PERFORMANCE), OR 4 (EXCEPTION)
RECORD CLASS IS %MNSEGCL AT COLUMN %PTR
INPUT SECTION REJECTED

TYPE: Error

REASON: Invalid record class.

ACTION: Increase the CIC06015 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart DAY040.

Dump the record in error by using the job

CICILIST in prefix.MICS.CNTL. Consult your CICS systems programmer to determine origin of invalid record class.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+  
| C I C 0 6 0 1 6 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
RECORD TYPE WITHIN CLASS NOT 0 (DICTIONARY),
1 (TRANSACTION), OR 2 (GLOBAL)
RECORD TYPE WITHIN CLASS IS %MCTSSDID AT
COLUMN %PTR
INPUT SECTION REJECTED

TYPE: Error

REASON: Invalid record type within class.

ACTION: Increase the CIC06016 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart DAY040.

Dump the record in error by using the job
CICILIST in prefix.MICS.CNTL. Consult your
CICS systems programmer to determine origin of
invalid record type within class.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+  
| C I C 0 6 0 1 7 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
NO DATA RECORDS FOR DATA SECTION
NUMBER OF DATA RECORDS IS %MCTSSDRN AT
COLUMN %PTR
INPUT SECTION REJECTED

TYPE: Error

REASON: No data to process in a data section.

ACTION: Increase the CIC06017 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart DAY040.

Dump the record in error by using the job
CICILIST in prefix.MICS.CNTL. Consult your
CICS systems programmer to determine the
origin of invalid data section.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+
| C I C 0 6 0 1 8 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
DATA SECTION LENGTH IS LESS THAN SIZE OF
SECTION HEADER AND DESCRIPTOR
DATA SECTION LENGTH IS %SECTLENG AT
COLUMN %PTR
INPUT SECTION REJECTED

TYPE: Error

REASON: Corrupted CMF data section length.

ACTION: Increase the CIC06018 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart DAY040.

Dump the record in error by using the job
CICILIST in prefix.MICS.CNTL. Consult your
CICS systems programmer to determine origin of
invalid data section.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+
| C I C 0 6 0 1 9 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
UNABLE TO DETERMINE CICS RELEASE OR RELEASE IS
NOT SUPPORTED
PRODUCT SECTION LENGTH IS %SMFLPS AT COLUMN
%PTR
INPUT RECORD REJECTED

TYPE: Error

REASON: Invalid or unknown CMF record structure.

ACTION: Increase the CIC06019 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart DAY040.

Dump the record in error by using the job
CICILIST in prefix.MICS.CNTL. Consult your
CICS systems programmer to determine if record
structure is invalid or unknown (new release
of SMF may alter length of product section).

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+  
| C I C 0 6 0 2 0 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR: CICS RELEASE %RELEASE
IS NOT SUPPORTED. INPUT RECORD IS %INPRECS.
ALL SUBSEQUENT INPUT DATA FOR CICS REGION
APPLID: %CSUAPPL EXECUTING ON SYSTEM:
%ORGSYSID USING THIS RELEASE WILL BE BYPASSED.

TYPE: Error

REASON: The CICS release is either no longer supported
or is not yet supported by CA MICS.

ACTION: Check the sharedprefix.MICS.PC.TEXT members
with names beginning with @CIC. Read the
members for those product changes that have
not yet been applied to your complex to
determine whether your CICS release level
could be supported by unapplied maintenance.
If you do not find any product change that
provides the release support, contact CA

Technical Support.

If your input is from an old release of CICS, check CA MICS product changes that you have already applied to determine whether one of them has invalidated your input. If such a problem occurs, contact CA Technical Support.

REFERENCES: sharedprefix.MICS.PC.TEXT members.

```
+-----+  
| C I C 0 6 0 2 1 |  
+-----+
```

TEXT: 20 CICS REGIONS WITH UNSUPPORTED RELEASES HAVE BEEN DETECTED. NO FURTHER CIC06020 MESSAGES WILL BE ISSUED.

TYPE: Informational

REASON: The CICS release is either no longer supported or is not yet supported by CA MICS.

ACTION: See CIC06020 ACTION section.

REFERENCES: See CIC06020 REFERENCES section.

```
+-----+  
| C I C 0 6 0 2 2 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
P.PARMS(CICRLRT) EXIT IN ERROR - TRANTYPE =
%TRANTYPE
TRANTYPE MUST = S, M, L, C, OR X
TRANSACTION RECORD REJECTED

TYPE: Error

REASON: Invalid TRANTYPE.

ACTION: Update prefix.MICS.PARMS(CICRLRT) to correct TRANTYPE assignment. Restart DAY040.

REFERENCES: Section 7.3.7

```
+-----+  
| C I C 0 6 0 2 3 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
ZERO DATA ELEMENT LENGTH IN DICTIONARY AT
COLUMN %PTR
DATA ELEMENT OWNER - %CMODNAME
TYPE - %CMODTYPE
DATA ID - %CMODIDNT
LENGTH - %CMODLENG
CONNECTOR - %CMODCONN
NAME - %CMODHEAD
INPUT RECORD REJECTED

TYPE: Error

REASON: Invalid CMF data dictionary encountered in
input data.

ACTION: See CIC06020 ACTION section.

REFERENCES: See CIC06020 REFERENCES section.

```
+-----+  
| C I C 0 6 0 2 4 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
NO DATA DICTIONARY INFORMATION
INPUT RECORD REJECTED

TYPE: Error

REASON: Data dictionary information required for input
processing is not available for this region.

ACTION: If measurement data for the region is not
required, increase CIC06024 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. This causes all records to be
rejected until a dictionary record is
encountered.

If data for the region is required, you must obtain the dictionary records and place them first in the input data.

For CICS/ESA release 3.1.1 and higher, IBM provides a utility program, DFHMNDUP, which creates and writes dictionary records to a sequential file. You can use this utility to generate the missing dictionary records. Concatenate this file ahead of your normal input file to DAY040. For more information on DFHMNDUP, see the CICS/ESA Operations Guide.

REFERENCES: Sections 7.3.8 and 9.1.2
 IBM CICS Customization Guide
 IBM CICS/ESA Operations Guide

```
+-----+
| C I C 0 6 0 2 5 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
 SYSID=%ORGSYSID CICSID=%CICSID
 APPLID=%CSUAPPL
 INVALID DATA CONNECTOR IN CICS DATA
 DICTIONARY
 CONNECTOR = %CMODCONN FOR DATA CLASS =
 %CLASS AT COLUMN %PTR
 CONNECTOR MUST BE < 21 FOR CLASS 2, < 357
 FOR CLASS 3, AND < 31 FOR CLASS 4
 INPUT RECORD REJECTED

TYPE: Error

REASON: Number of connectors exceeds capacity of
 internal data dictionary array structure.

ACTION: The capacity of the data dictionary array
 structure can be increased by changing the
 number assigned to the CDACONS macro variable
 in sharedprefix.MICS.SOURCE(\$CICMSTR). This
 assignment is made in the following statement
 on the line with sequence number 00097000:

```
%LET CDACONS = 356;
```

For example, if you need to process 370
 fields, you would change this line to

%LET CDACONS = 370;

REFERENCES: None

```
+-----+  
| C I C 0 6 0 2 7 |  
+-----+
```

TEXT: INPUT DICTIONARY RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID APPLID=%CSUAPPL
CONTAINS A TYPE C (CHARACTER) DATA ELEMENT
WHOSE LENGTH AT COLUMN %PTR EXCEEDS 1000
DATA ELEMENT LENGTH IS SET TO 1000 FOR THE
FOLLOWING DATA ELEMENT:

DATA ELEMENT OWNER	- %CMODNAME
TYPE	- %CMODTYPE
DATA ID	- %CMODIDNT
LENGTH	- %CMODLENG
CONNECTOR	- %CMODCONN
NAME	- %CMODHEAD

PROCESSING CONTINUES

TYPE: Warning

REASON: Data dictionary record encountered in input
with a type C element length greater than
1000. The element's length is set to maximum
value for a SAS character field.

ACTION: None.

REFERENCES: Sections 7.3.8, 9.1.3.2, 9.2.2, and 10.2.1
IBM CICS Customization Guide

```
+-----+  
| C I C 0 6 0 3 0 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
INVALID DATA ELEMENT LENGTH IN DATA
DICTIONARY AT COLUMN %PTR
WHEN DATA ELEMENT TYPE IS S (CLOCK, FLAG,
AND COUNT) ELEMENT LENGTH MUST BE 8

DATA ELEMENT OWNER	- %CMODNAME
TYPE	- %CMODTYPE
DATA ID	- %CMODIDNT
LENGTH	- %CMODLENG

CONNECTOR - %CMODCONN
 NAME - %CMODHEAD
 INPUT RECORD REJECTED

TYPE: Error

REASON: Invalid data dictionary record encountered in
 input data.

ACTION: See CIC06020 ACTION section.

REFERENCES: See CIC06020 REFERENCES section.

```
+-----+
| C I C 0 6 0 3 1 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
 SYSID=%ORGSYSID CICSID=%CICSID
 APPLID=%CSUAPPL
 INVALID DATA ELEMENT LENGTH IN DATA
 DICTIONARY AT COLUMN %PTR
 WHEN DATA ELEMENT TYPE IS T (STCK - TIME
 STAMP) ELEMENT LENGTH MUST BE 8
 DATA ELEMENT OWNER - %CMODNAME
 TYPE - %CMODTYPE
 DATA ID - %CMODIDNT
 LENGTH - %CMODLENG
 CONNECTOR - %CMODCONN
 NAME - %CMODHEAD
 INPUT RECORD REJECTED

TYPE: Error

REASON: Invalid data dictionary record encountered in
 input data.

ACTION: See CIC06020 ACTION section.

REFERENCES: See CIC06020 REFERENCES section.

```
+-----+
| C I C 0 6 0 3 2 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
 SYSID=%ORGSYSID CICSID=%CICSID
 APPLID=%CSUAPPL
 INVALID DATA ELEMENT TYPE IN DATA DICTIONARY

AT COLUMN %PTR
 DATA ELEMENT TYPE MUST BE A, P, C, S, OR T
 DATA ELEMENT OWNER - %CMODNAME
 TYPE - %CMODTYPE
 DATA ID - %CMODIDNT
 LENGTH - %CMODLENG
 CONNECTOR - %CMODCONN
 NAME - %CMODHEAD
 INPUT RECORD REJECTED

TYPE: Error

REASON: Invalid data dictionary record encountered in input data.

ACTION: See CIC06020 ACTION section.

REFERENCES: See CIC06020 REFERENCES section.

```
+-----+
| C I C 0 6 0 3 3 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
 SYSID=%ORGSYSID CICSID=%CICSID
 APPLID=%CSUAPPL
 LENGTH OF DATA RECORD IS INSUFFICIENT TO
 CONTAIN ALL RELATED DATA ELEMENTS DEFINED IN
 DATA DICTIONARY
 DATA RECORD BEGINS AT COLUMN %PTR
 RELATED DATA ELEMENTS ARE:

TYPE: Error

REASON: Sum of data element lengths described in data dictionary exceeds data record length. The dictionary record does not describe the data record.

ACTION: If CICS maintenance has been applied that alters the CMF record, the MCT may require assembly. (Data record format maintenance does not take effect until the MCT is assembled. Dictionary record content is altered by SMP processing.)

If assembly of the MCT is the problem, its update will eliminate this error condition for tomorrow's data. The current data will

require use of data dictionary records that precede introduction of the maintenance.

If there is dictionary information for this region in the DETAIL.CICCDC01 file and it reflects the current data record format, delete the input data dictionary records.

If there is no dictionary information for this region in the DETAIL.CICCDC01 file, increase the CIC06033 threshold limit in prefix.MICS.PARMS(CICTHRSH) to tolerate condition (with loss of measurement data).

REFERENCES: Section 7.3.8
IBM CICS Customization Guide

```
+-----+
| C I C 0 6 0 3 4 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
DICTIONARY ARRAY UPPER BOUNDARY LESS THAN
LOWER
LOWER BOUNDARY IS %CDASTART
UPPER BOUNDARY IS %CDAEND
INPUT RECORD REJECTED

TYPE: Error

REASON: Multidimensional data dictionary array
emulation not properly defined to
sharedprefix.MICS.ASM(@\$CICDEP).

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+
| C I C 0 6 0 3 5 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
CONNECTOR NOT WITHIN ARRAY BOUNDARY
CONNECTOR IS %CDAVAR
LOWER BOUNDARY IS %CDASTART
UPPER BOUNDARY IS %CDAEND
INPUT RECORD REJECTED

TYPE: Error

REASON: Data element connector value is not compatible with multidimensional data dictionary array emulation (its value exceeds the number of rows allotted to its class of data).

ACTION: If the number of occurrences is insignificant, it may be regarded as a transient error that can be tolerated. Increase the CIC06035 threshold value in prefix.MICS.PARMS(CICTHRSH).

If the number of occurrences is significant, it may be caused by a repetitive storage violation (corrupted connector value) or additional data elements (user clocks/counters or new data variables). Consult your CICS systems programmer to determine the source of corruption.

If the problem is the result of a significant number of data elements added to the CMF record, contact CA Technical Support.

REFERENCES: Section 7.3.8
IBM CICS Customization Guide

```
+-----+  
| C I C 0 6 0 3 9 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS,
SYSID=%ORGSYSID CICSID=%CICSID
APPLID=%CSUAPPL
ATTEMPTING TO PROCESS SECTION AT COLUMN %PTR
RECORD IS NOT LONG ENOUGH TO CONTAIN SECTION
INPUT RECORD REJECTED

TYPE: Error

REASON: Truncated CMF record.

ACTION: Increase CIC06039 threshold value in prefix.MICS.PARMS(CICTHRSH) to tolerate condition. Restart DAY040.

Dump the record in error by using the job CICILIST in prefix.MICS.CNTL. Consult your CICS systems programmer to determine the

origin of error.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+  
| C I C 0 6 0 4 0 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
DECOMPRESSION FAILED
INPUT RECORD REJECTED

TYPE: Error

REASON: Decompression routine indicated failure.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 0 4 1 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
MONITOR RECORD TYPE NOT D OR S
INPUT RECORD REJECTED

TYPE: Error

REASON: Invalid or unknown ASG-TMON record type.

ACTION: Check ASG-TMON input data set to ensure that
it is created by the TMV608 dump program.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 0 4 2 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
RECORD LENGTH INVALID
RECORD LENGTH IS %CICLEN
INPUT RECORD REJECTED

TYPE: Error

REASON: ASG-TMON record length is less than expected
minimum length for type of record.

ACTION: Increase CIC06042 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart DAY040.

Dump the record in error by using the job
CICILIST in prefix.MICS.CNTL. Consult your
CICS systems programmer to determine the
origin of error.

REFERENCES: Sections 7.3.8 and 9.1.3.2

```
+-----+  
| C I C 0 6 0 4 3 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
DICTIONARY ARRAY LOWER BOUNDARY IS NOT VALID
BOUNDARY IS %CDASTART
INPUT RECORD REJECTED

TYPE: Error

REASON: Multidimensional data dictionary array
emulation not properly defined to
sharedprefix.MICS.ASM(@\$CICDEP).

ACTION: Contact CA Technical Support. This is a
CA MICS CICS component internal error that may
be the result of a user modification.

REFERENCES: Section 7.3.9

```
+-----+  
| C I C 0 6 0 4 4 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
INPUT DATA CLASS PASSED IN CURRENTCLASS
VARIABLE NOT DEFINED TO \$CICDEP
INVALID DATA CLASS IS %CDAVAR
INPUT RECORD REJECTED

TYPE: Error

REASON: Data class (record type) is not known to the data element processor (@\$CICDEP).

The data class may be new or it may have been corrupted by a storage violation.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 0 4 5 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
CONNECTOR LENGTH PASSED IN CONNECTORLENGTH
VARIABLE IS ZERO OR NEGATIVE
INPUT RECORD REJECTED

TYPE: Error

REASON: Invalid connector length. The connector length may have been corrupted by a storage violation.

ACTION: Increase CIC06045 threshold value in prefix.MICS.PARMS(CICTHRSH) to tolerate condition. Restart DAY040.

Dump the record in error by using the job CICILIST in prefix.MICS.CNTL. Consult your CICS systems programmer to determine the origin of error.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+  
| C I C 0 6 0 4 6 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
CONNECTOR COUNT PASSED IN CONNECTORCOUNT
VARIABLE IS ZERO OR NEGATIVE
INPUT RECORD REJECTED

TYPE: Error

REASON: Invalid connector count. The connector count may have been corrupted by a storage violation.

ACTION: Increase CIC06046 threshold value in prefix.MICS.PARMS(CICTHRSH) to tolerate condition. Restart DAY040.

Dump the record in error by using the job CICILIST in prefix.MICS.CNTL. Consult your CICS systems programmer to determine the origin of error.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+  
| C I C 0 6 0 4 7 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
DICTIONARY ARRAY UPPER BOUNDARY IS NOT VALID
BOUNDARY IS %CDAEND
INPUT RECORD REJECTED

TYPE: Error

REASON: Multidimensional data dictionary array
emulation not properly defined to
sharedprefix.MICS.ASM(@\$CICDEP).

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 0 4 8 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
NO DATA DICTIONARY ENTRY FOR CONNECTOR %CDAVAR
INPUT RECORD REJECTED

TYPE: Error

REASON: Connector does not have a data dictionary
entry. Unknown data element in data record.
Unable to process data record.

If the number of occurrences is insignificant, it may be regarded as a transient error that can be tolerated.

If the number of occurrences is significant, it may be due to a repetitive storage violation, a corrupted data dictionary entry, or an old dictionary record used as input.

Check all DAY040 input data sets to ensure all DD statements are correct. For example, if the IBM utility DFHMNDUP was used to create dictionary records to resolve a previous problem, those dictionary records may not match currently collected data, and should be removed.

ACTION: To bypass the affected data, increase CIC06048 value in prefix.MICS.PARMS(CICTHRSH). Restart DAY040.

REFERENCES: Sections 7.3.8 and 9.1.3.2
IBM CICS Customization Guide

```
+-----+
| C I C 0 6 0 4 9 |
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
ERROR DETECTED DURING \$CICDEP INITIALIZATION
INPUT RECORD REJECTED

TYPE: Error

REASON: @\$CICDEP initialization failed. Unable to perform data element processing.

ACTION: See preceding message in MICSLOG to determine reason for failure.

REFERENCES: N/A

```
+-----+
| C I C 0 6 0 5 2 |
+-----+
```


TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
OFFSET TABLE EXTENSION FOUND WHEN
INAPPROPRIATE
DATA ELEMENT(S) NOT PROCESSED
INPUT PROCESSING CONTINUES

TYPE: Error

REASON: The data element description in
sharedprefix.MICS.SOURCE(CICDEPEL) does not
match data element type in data dictionary.
Not able to process data element.

ACTION: If error is a result of user modification,
then correct the data element description in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise, dump the record in error
using the job CICILIST in prefix.MICS.CNTL,
and then contact CA Technical Support.

REFERENCES: Sections 9.1.3.1 and 9.3.1

```
+-----+  
| C I C 0 6 0 5 3 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
OFFSET TABLE EXTENSION NOT FOUND WHEN REQUIRED
DATA ELEMENT(S) NOT PROCESSED
INPUT PROCESSING CONTINUES

TYPE: Error

REASON: The data element description in
sharedprefix.MICS.SOURCE(CICDEPEL) does not
match data element type in data dictionary.
Unable to process data element.

ACTION: If error is a result of user modification,
then correct the data element description in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise, dump the record in error
using the job CICILIST in prefix.MICS.CNTL,
and then contact CA Technical Support.

REFERENCES: Sections 9.1.3.1 and 9.3.1

```
+-----+  
| C I C 0 6 0 5 4 |
```

+-----+

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
ONE OR MORE INVALID PACKED FIELDS ENCOUNTERED
DATA ELEMENT(S) NOT PROCESSED
INPUT PROCESSING CONTINUES

TYPE: Error

REASON: Invalid data found for a data element whose
data type is packed. Not able to process data
element. Probably corrupted input data.

ACTION: Increase the CIC06054 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart job.

If the number of occurrences is insignificant,
error may be considered transient and
tolerable.

If the number of occurrences is significant,
dump the record in error using the job in
prefix.MICS.CNTL(CICILIST). Consult your CICS
systems programmer to determine the cause.

REFERENCES: Sections 7.3.8 and 9.3.1
IBM CICS Customization Guide

+-----+

| C I C 0 6 0 5 6 |

+-----+

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
RECORD LENGTH INVALID SINCE LESS THAN 40
RECORD LENGTH IS %CICLEN
INPUT RECORD REJECTED

TYPE: Error

REASON: Input record is less than 40 bytes, which is
the minimum length required to process the SMF
header.

ACTION: If the total number of errors is small, the
condition may be caused by bad input data.
Increase the CIC06056 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart job.

If the error resulted from incorrect JCL specification, then review and correct prefix.MICS.PARMS(INPUTSMF) and regenerate DAILY JCL using JCLGENU. To do this, first specify the word DAILY in prefix.MICS.PARMS(JCLGENU); then submit the job JCLGENU in prefix.MICS.CNTL.

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 0 5 7 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
RECORD LENGTH INVALID SINCE LESS THAN LENGTH
OF HEADER AND PRODUCT SECTION
HEADER AND PRODUCT SECTION LENGTH IS %HEADPROD
RECORD LENGTH IS %CICLEN
INPUT RECORD REJECTED

TYPE: Error

REASON: The input record is too short to contain a
valid SMF header and product section.

ACTION: If the total number of errors is small, the
condition may be caused by bad input data.
Increase the CIC06057 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart job.

If the error resulted from incorrect JCL specification, then review and correct prefix.MICS.PARMS(INPUTSMF) and regenerate DAILY JCL using JCLGENU. To do this, first specify the word DAILY in prefix.MICS.PARMS(JCLGENU); then submit the job JCLGENU in prefix.MICS.CNTL.

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 0 5 8 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
RECORD SUBTYPE IS INVALID

SUBTYPE SHOULD BE 0, 1, OR 2
SUBTYPE IS %SMFSTY
INPUT RECORD REJECTED

TYPE: Error

REASON: The input record contains a subtype that is invalid for CICS release 3.1.1 or higher. Valid subtypes are 0 (CICS journal), 1 (CICS Monitoring), and 2 (CICS Statistics).

ACTION: If the total number of errors is small, the condition may be caused by bad input data. Increase the CIC06058 threshold value in prefix.MICS.PARMS(CICTHRSH) to tolerate condition. Restart job.

If the error resulted from incorrect JCL specification, then review and correct prefix.MICS.PARMS(INPUTSMF) and regenerate DAILY JCL using JCLGENU. To do this, first specify the word DAILY in prefix.MICS.PARMS(JCLGENU); then submit the job JCLGENU in prefix.MICS.CNTL.

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 0 5 9 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR STAT RECORD %INPREC
SYSID=%ORGSYSID CICS=%CICSID APPLID=%CSUAPPL
CALCULATED RECORD LENGTH EXCEEDS ACTUAL RECORD
LENGTH,
ACTUAL LENGTH IS %CUMLNTH
INPUT RECORD REJECTED

TYPE: Error

REASON: The input record length is too short for the CA MICS Analyzer Option for CICS to read the next logical record in the CICS statistics record (for CICS release 3.1.1 and higher).

ACTION: If the total number of errors is small, the condition may be caused by bad input data. Increase the CIC06059 threshold value in prefix.MICS.PARMS(CICTHRSH) to tolerate

condition. Restart job.

If the error resulted from incorrect JCL specification, then review and correct prefix.MICS.PARMS(INPUTSMF) and regenerate DAILY JCL using JCLGENU. To do this, first specify the word DAILY in prefix.MICS.PARMS(JCLGENU); then submit the job JCLGENU in prefix.MICS.CNTL.

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 0 6 0 |  
+-----+
```

TEXT: INPUT PROCESSING ERROR FOR RECORD %INPRECS
SYSID=%ORGSYSID CICS=%CICSID APPLID=%CSUAPPL
INVALID RECORD CLASS
RECORD CLASS MUST BE 1, 3, OR 4,
RECORD CLASS IS %SMFMNCL
INPUT RECORD REJECTED

TYPE: Error

REASON: For CICS release 3.1.1 and higher, the CMF
input record contains an invalid class. Valid
classes are 1 - dictionary; 3 - performance;
4 - exception.

ACTION: If the total number of errors is small, the
condition may be caused by bad input data.
Increase the CIC06060 threshold value in
prefix.MICS.PARMS(CICTHRSH) to tolerate
condition. Restart job.

If the error resulted from incorrect JCL specification, then review and correct prefix.MICS.PARMS(INPUTSMF) and regenerate DAILY JCL using JCLGENU. To do this, first specify the word DAILY in prefix.MICS.PARMS(JCLGENU); then submit the job JCLGENU in prefix.MICS.CNTL.

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 0 6 1 |  
+-----+
```

TEXT: INPUT PROCESSING TERMINATED DUE TO A CICS
RELEASE NOT SUPPORTED BY MICS. RETURN CODE
998 IS BEING ISSUED. IF YOU NEED TO HAVE THIS
STEP COMPLETE SUCCESSFULLY, GO INTO
prefix.MICS.PARMS(CICOPS).
ENTER A LINE CONTAINING "VERIFY RELEASE
NOABORT" AND RUN prefix.MICS.PARMS(CICPGEN).
DELETE ANY LINE CONTAINING CIC06020 FROM THE
prefix.MICS.PARMS(CICTHRSH).
RESTART THE JOB STEP THAT FAILED (DAY040 OR
INCR040).

TYPE: Warning

REASON: During execution of the DAY040 step of the
DAILY or of the INCR040 step of the
Incremental Update, a U0998 ABEND is issued
when the first record from a CICS release not
supported in CA MICS is encountered.

ACTION: CA MICS should be upgraded to support the
new release of CICS. This applies to both
SMF type 110 records and to ASG-TMON (TCE)
records.

REFERENCES: Section 7.3.2.13

```
+-----+
| C I C 0 6 0 6 2 |
+-----+
```

TEXT: INPUT PROCESSING COMPLETE EVEN THOUGH RECORDS
FOR AN UNSUPPORTED CICS RELEASE WERE DETECTED.
PLEASE UPGRADE CA MICS TO SUPPORT THE NEW
RELEASE AND PROCESS THE REJECTED RECORDS AFTER
THE CA MICS UPGRADE.

TYPE: Warning

REASON: The DAY040 step of the DAILY or the INCR040
step of the Incremental Update job, will
continue even though it encounters type 110 of
ASG-TMON (TCE) records that are not yet
supported by CA MICS. Any records that are
supported are processed as usual. Those that
are not supported are skipped.

ACTION: CA MICS should be upgraded to support the

new releases of CICS or of the ASG-TMON (TCE) product. Once CA MICS has been upgraded, records from any CICSID that were skipped on previous updates should be forced into the CA MICS data base.

REFERENCES: Section 7.3.2.13

```
+-----+  
| C I C 0 6 1 0 1 |  
+-----+
```

TEXT: INVALID PARM(CICTHRSH): %KEYWRD01
 KEYWORD MUST BE CICCCEX, CICMSGTH,
 OR CIC06EXX
 THRESHOLD IGNORED

TYPE: Warning

REASON: An invalid keyword was specified in
 prefix.MICS.PARMS(CICTHRSH).

ACTION: Correct keyword.

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 1 0 2 |  
+-----+
```

TEXT: INVALID PARM(CICTHRSH): %KEYWRD01
 ERROR NUMBER NOT NUMERIC
 THRESHOLD IGNORED

TYPE: Warning

REASON: An invalid CIC060nn error tolerance statement
 was specified in prefix.MICS.PARMS(CICTHRSH).
 Valid statements are CIC06001 through
 CIC06099.

ACTION: Correct statement.

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 1 0 3 |  
+-----+
```

TEXT: INVALID PARM(CICTHRSH): %KEYWRD01
 ERROR NUMBER MUST BE 1 THRU 99
 THRESHOLD IGNORED

TYPE: Warning

REASON: An invalid CIC060nn error tolerance statement
 was specified in prefix.MICS.PARMS(CICTHRSH).
 Valid statements are CIC06001 through
 CIC06099.

ACTION: Correct statement.

REFERENCES: Section 7.3.8

```
+-----+
| C I C 0 6 1 0 4 |
+-----+
```

TEXT: INVALID PARM(CICTHRSH): %KEYWRD01 %KEYWRD02
 THRESHOLD VALUE NOT NUMERIC
 THRESHOLD IGNORED

TYPE: Warning

REASON: Non-numeric threshold value was specified by
 the CIC060nn error tolerance statements in
 prefix.MICS.PARMS(CICTHRSH). Threshold values
 must be numeric.

ACTION: Correct statement.

REFERENCES: Section 7.3.8

```
+-----+
| C I C 0 6 1 0 5 |
+-----+
```

TEXT: INVALID PARM(CICTHRSH): %KEYWRD01 %KEYWRD02
 THRESHOLD VALUE GREATER THAN 999999999
 THRESHOLD IGNORED

TYPE: Warning

REASON: Invalid threshold value was specified by the
 CIC060nn error tolerance statement in
 prefix.MICS.PARMS(CICTHRSH). Threshold values

cannot exceed 999999999.

ACTION: Correct statement.

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 1 0 6 |  
+-----+
```

TEXT: INVALID PARM(CICTHRSH): %KEYWRD01 %KEYWRD02
%KEYWRD03
TOO MANY THRESHOLD PARAMETERS
THRESHOLD IGNORED

TYPE: Warning

REASON: An invalid CIC060nn error tolerance statement
was specified in prefix.MICS.PARMS(CICTHRSH).

ACTION: Correct statement.

REFERENCES: Sections 7.3.8

```
+-----+  
| C I C 0 6 1 0 7 |  
+-----+
```

TEXT: PARM(CICTHRSH) THRESHOLD IN EFFECT: %KEYWRD01
%KEYWRD02

TYPE: Information

REASON: CICTHRSH statement has passed validation and
is now in effect.

ACTION: None.

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 1 0 8 |  
+-----+
```

TEXT: A MAXIMUM OF %CICMSGNM ITERATIONS OF ANY
CIC060XX ERROR MESSAGE WILL BE LISTED
FOR EACH INPUT DDNAME

TYPE: Information

REASON: CICTHRSH parameter CICMSGTH is in effect.

ACTION: None.

REFERENCES: Section 7.3.8

```
+-----+
| C I C 0 6 1 2 0 |
+-----+
```

TEXT: +=====+

INPUT PROCESSING FOR DDNAME %INDDNME

+=====+

TYPE: Information

REASON: Indicates the start of MICSLOG input processing for the DDNAME indicated by variable %INDDNME.

ACTION: None.

REFERENCES: N/A

```
+-----+
| C I C 0 6 1 2 1 |
+-----+
```

TEXT: ERROR THRESHOLD EXCEEDED

USER ABEND 998 ISSUED

TYPE: Information

REASON: Number of occurrences of a CIC060nn error condition exceeds the error tolerance threshold value set in prefix.MICS.PARMS(CICTHRSH) or its default.

ACTION: Correct error condition as indicated by other error messages in the MICSLOG.

REFERENCES: N/A

```
+-----+
```

```
| C I C 0 6 1 2 5 |  
+-----+
```

TEXT: INPUT PROCESSING WARNING
DATA DICTIONARY CONTAINS ONE OR MORE DATA IDS
NOT DEFINED IN SP.SOURCE(CICDEPEL)
DATA RECORD CONTAINS A NEW ELEMENT OR DATA
DICTIONARY HAS BECOME CORRUPTED
INPUT PROCESSING CONTINUES

TYPE: Warning

REASON: Input record contains a data field not defined
in the data element list CICDEPEL. Unable to
relate input data field to a SAS data element.
As a result, the input field is ignored.

ACTION: If a new input data field, add an entry to
sharedprefix.MICS.SOURCE(CICDEPEL).

REFERENCES: Section 9.1.3.1

```
+-----+  
| C I C 0 6 1 2 6 |  
+-----+
```

TEXT: INPUT PROCESSING WARNING
ONE OR MORE INPUT DATA FIELDS NOT PROCESSED -
NO CORRESPONDING SAS VARIABLE.
SEE MESSAGE CIC06221 ABOVE FOR DETAILS
INPUT PROCESSING CONTINUES

TYPE: Warning

REASON: sharedprefix.MICS.SOURCE(CICDEPEL) contains an
entry whose data element name is unknown to
SAS. This is normal if the data element is
turned off (as shipped or deactivated by you)
in sharedprefix.MICS.GENLIB(CICGENIN).

ACTION: If access to data is desired, element must be
defined to SAS. If element is in
sharedprefix.MICS.GENLIB(CICGENIN), it must be
activated. If not, add a LENGTH statement for
the data element in the _USRSLFL exit in
prefix.MICS.USER.SOURCE(#BASEXIT).

REFERENCES: Section 9.1.3.1

```
+-----+  
| C I C 0 6 1 2 7 |  
+-----+
```

TEXT: INPUT PROCESSING WARNING
DATA DICTIONARY CONTAINS A DATA TYPE NOT
DEFINED IN @\$CICDEP
DATA DICTIONARY DEFINES A NEW ELEMENT OR
HAS BECOME CORRUPTED
INPUT PROCESSING CONTINUES

TYPE: Warning

REASON: Data dictionary record contains a data type
that is not defined in load module @\$CICDEP.

ACTION: Print the DETAIL.CICCDC01 file by submitting
the job in prefix.MICS.CNTL(CICCDCPT); then
contact CA Technical Support.

REFERENCES: Section 9.2.1

```
+-----+  
| C I C 0 6 1 2 8 |  
+-----+
```

TEXT: INPUT PROCESSING WARNING FOR DICTIONARY
RECORD FOR CICSID=%CICSID APPLID=%CSUAPPL
USER %ITEM WITH OWNER=%CMDNAME ENCOUNTERED
ONLY OWNER=USER IS SUPPORTED
THIS %ITEM WILL BE IGNORED WHEN IT IS
ENCOUNTERED IN TRANSACTION DATA

TYPE: Warning

REASON: Data dictionary record contains a user area
that has a label other than USER. CA MICS
only supports a user area with the label of
USER. This user area is ignored.

ACTION: None.

REFERENCES: Section 10.1.12

```
+-----+  
| C I C 0 6 1 3 3 |  
+-----+
```

TEXT: CONNECTOR - %CMODCONN
 DATA ID - %CMODIDNT
 TYPE - %CMDTYPE
 LENGTH - %CMODLENG

TYPE: Information

REASON: This message is issued in conjunction with
 CIC06033 to display the data dictionary entry
 in error.

ACTION: See CIC06033.

REFERENCES: See CIC06033.

```
+-----+
| C I C 0 6 1 3 4 |
+-----+
```

TEXT: COMPUTED RECORD LENGTH IS %CMODLENG
 RECORD LENGTH IS %MCTSSDRL
 INPUT RECORD REJECTED

TYPE: Information

REASON: This message is issued in conjunction with
 CIC06033 to display the length of record in
 error.

ACTION: See CIC06033.

REFERENCES: See CIC06033.

```
+-----+
| C I C 0 6 1 5 0 |
+-----+
```

TEXT: INPUT PROCESSING SUMMARY FOR DDNAME %INDDNME
 TYPE 110 CICS RECORDS %XMONRECS
 TYPE 111 TRANSACTION GATEWAY RECORDS %XGTWRECS
 NON-COMPONENT SMF RECORD TYPES %XEXCRECS
 =====
 TOTAL SMF RECORDS INPUT %XINPRECS
 +=====+

SMF TYPE 110 Record Processing Details
 TYPE 110 RECORDS IN INPUT %XMONRECS
 110 RECORDS REJECTED BY _USRSEL EXIT %XBASREJ

REASON: Displays input statistics for the DDNAME indicated by the variable \$INDDNME when CMF processing is active for the unit. When CMF is active, this message is followed by message CIC06154I, which provides additional SMF type 110 record processing information.

Variable %XMONRECS displays the total number of SMF type 110 records found in the input. This includes CICS statistics and journal records.

Variable %XGTWRECS displays the total number of SMF type 111 CICS Transaction Gateway records found in the input.

Variable %XEXCRECS displays the total number of non-component records found in the input.

Variable %XINPRECS displays the total of 110, 111, and non-component records.

The final variable, %XBASREJ, displays the total number of SMF type 110 records, if any, that were discarded using the _USRSEL exit.

TYPE: Information

ACTION: None.

REFERENCES: N/A

```
+-----+
| C I C 0 6 1 5 1 |
+-----+
```

TEXT:	INPUT PROCESSING ERROR	NUMBER OF OCCURRENCES
	-----	-----

TYPE: Information

REASON: Display a summary of CIC060nn error conditions encountered during input processing.

ACTION: Resolve all error conditions as indicated by the CIC060nn error messages in the MICSLOG. If errors cannot be resolved, bypass error conditions by specifying error tolerance thresholds in prefix.MICS.PARMS(CICTHRSH).

REFERENCES: Section 7.3.8

```
+-----+  
| C I C 0 6 1 5 2 |  
+-----+
```

TEXT: %ERRMSG %ERRCOUNT

TYPE: Information

REASON: This message is issued in conjunction with
CIC06151 to display a summary of CIC060nn
error conditions encountered during input
processing.

ACTION: Resolve all error conditions as indicated by
the CIC060nn error messages in the MICSLOG.
If errors cannot be resolved, bypass error
conditions by specifying error tolerance
thresholds in prefix.MICS.PARMS(CICTHRSH).

REFERENCES: N/A

```
+-----+  
| C I C 0 6 1 5 3 |  
+-----+
```

TEXT: =====

INPUT PROCESSING STATISTICS FOR SYSID=%SYSID
CICSID=%CICSID %WRKREGID=%REGID

TYPE: Information

REASON: Input processing statistics for a CICS region.

ACTION: None.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 1 5 4 |  
+-----+
```

TEXT: DICTIONARY RECORDS INPUT
 %WRXDICT
TRANSACTION RECORDS INPUT
 %WRXTRNCT

TRANSACTION RECORDS REJECTED BY USRSCMF EXIT
 %dWRXTREJ
 GLOBAL RECORDS INPUT
 %dWRXGBLCT
 GLOBAL RECORDS REJECTED BY USRSCMF EXIT
 %dWRXGREJ
 ACCOUNTING DATA SECTIONS REJECTED BY SELCODE
 %dWRJXAREJ
 ACCOUNTING RECORDS INPUT
 %dWRXACTCT
 EXCEPTION DATA SECTIONS REJECTED BY SELCODE
 %dWRJXEREJ
 EXCEPTION RECORDS INPUT
 %dWRXEXCCT
 CICCDC00 RECORDS REJECTED BY USRSCDC EXIT
 %dWRXCDCRJ
 CICCIN00 RECORDS REJECTED BY USRSCIN EXIT
 %dWRXCINRJ
 CICC SW00 RECORDS REJECTED BY USRSCSW EXIT
 %dWRXC SWRJ
 CICC SY00 RECORDS REJECTED BY USRSCSY EXIT
 %dWRXC SYRJ
 CICCDC00 RECORDS WRITTEN
 %dWRXCDCCT
 CICCIN00 RECORDS WRITTEN
 %dWRXCINCT
 CICC SW00 RECORDS WRITTEN
 %dWRXC SWCT
 CICC SY00 RECORDS WRITTEN
 %dWRXC SYCT

TYPE: Information

REASON: Display input processing statistics:

DICTIONARY RECORDS INPUT - the total number of CMF data dictionary records (type 0) processed. This includes data dictionary records for the accounting (class 2), performance (class 3), and exception (class 4) data.

TRANSACTION RECORDS INPUT - the total number of CMF performance class transaction records processed (class 3, type 1). This includes the transaction records rejected by the USRSCMF exit and other CICS user exits.

TRANSACTION RECORDS REJECTED BY USRSCMF EXIT -

the total number of CMF performance class transaction records rejected by the USRSCMF exit.

GLOBAL RECORDS INPUT - the total number of CMF performance class global records processed (class 3, type 2). This includes the global records rejected by the USRSCMF exit and other CICS user exits.

GLOBAL RECORDS REJECTED BY USRSCMF EXIT - the total number of CMF performance class global records rejected by the USRSCMF exit.

ACCOUNTING DATA SECTIONS REJECTED BY SELCODE - the total number of accounting data sections rejected due to selcode 0 and 4. Selcode is an optional keyword that you can specify in CICOPTS statements in prefix.MICS.PARMS(CICOPS). If this keyword is not specified, all accounting data sections encountered in input are rejected. Note this count is the number of data sections, not transaction records. This number may be lower than the actual transaction count, as each CMF data section can contain multiple transaction records.

ACCOUNTING RECORDS INPUT - the total number of accounting class transaction records processed (class 2, type 1).

EXCEPTION DATA SECTIONS REJECTED BY SELCODE - the total number of exception data sections rejected due to selcode 0 and 2. Selcode is an optional keyword that you can specify in CICOPTS statements in prefix.MICS.PARMS(CICOPS). If this keyword is not specified, all exception data sections encountered in input are rejected.

Note: This count is the number of data sections, not transaction records. This number may be lower than the actual transaction count, as each CMF data section can contain multiple transaction records.

EXCEPTION RECORDS INPUT - the total number of exception class transaction records processed

(class 4, type 1).

CICCDC00 RECORDS REJECTED BY USRSCDC EXIT - the total number of CICCDC observations deleted by the USRSCDC exit. The CICCDC file is created from the CMF data dictionary records.

CICCSY00 RECORDS WRITTEN - the total number of observations written to the CICCSY00 WORK file, which is created from CMF performance class global records. The file contains one observation for each global record that has been processed successfully.

ACTION: None.

REFERENCES: Section 9.1.2

```
+-----+
| C I C 0 6 1 5 7 |
+-----+
```

TEXT: TOTAL RECORDS INPUT
 %XINPRECS
 TRANSACTION OR SYSTEM RECORDS IN INPUT
 %XMONRECS
 TRANSACTION OR SYSTEM RECORDS REJECTED BY BASE
 %XUSRREJ

TYPE: Information

REASON: Display input statistics:

TOTAL RECORDS INPUT - the total number of input records processed by DAY040.

TRANSACTION OR SYSTEM RECORDS IN INPUT - the total number of ASG-TMON transaction and system records found in input.

TRANSACTION OR SYSTEM RECORDS REJECTED BY BASE - the total number of ASG-TMON transaction and system records rejected by USRSEL exit.

ACTION: None.

REFERENCES: N/A

```

+-----+
| C I C 0 6 1 5 8 |
+-----+

```

TEXT: TRANSACTION RECORDS PROCESSED
 %dWRXTRNCT
 SYSTEM RECORDS PROCESSED
 %dWRXSYSCT
 FILE SEGMENTS IN TRANSACTION RECORDS PROCESSED
 %dWRXFILCT
 MRO SEGMENTS IN TRANSACTION RECORDS PROCESSED
 %dWRXMROCT
 CICCIN00 RECORDS REJECTED BY USRSCIN EXIT
 %dWRXCINRJ
 CICC_SF00 RECORDS REJECTED BY USRSCSF EXIT
 %dWRXC_SFRJ
 CICC_MR00 RECORDS REJECTED BY USRSCMR EXIT
 %dWRXC_MR RJ
 CICC_SW00 RECORDS REJECTED BY USRSCSW EXIT
 %dWRXC_SW RJ
 CICCIN00 RECORDS WRITTEN
 %dWRXCINCT
 CICC_SF00 RECORDS WRITTEN
 %dWRXC_SFCT
 CICC_MR00 RECORDS WRITTEN
 %dWRXC_MRCT
 CICC_SW00 RECORDS WRITTEN
 %dWRXC_SWCT

TYPE: Information

REASON: Display input statistics:

TRANSACTION RECORDS PROCESSED - the total number of ASG-TMON transaction records processed.

SYSTEM RECORDS PROCESSED - the total number of ASG-TMON system records processed.

FILE SEGMENTS IN TRANSACTION RECORDS PROCESSED - the total number of file segments processed in ASG-TMON transaction records. This count is higher than the number of transaction records as each transaction record typically contain multiple file segments.

MRO SEGMENTS IN TRANSACTION RECORDS PROCESSED - the total number of MRO segments processed

in ASG-TMON transaction records. This count is higher than the number of transaction records as each transaction record typically contain multiple MRO segments.

CICCSIN00 RECORDS REJECTED BY USRSCIN EXIT - the total number of CICCSIN observations deleted by the USRSCIN exit. The CICCSIN file is created from ASG-TMON transaction records to track incidents such as transaction abends.

CICCSF00 RECORDS REJECTED BY USRSCSF EXIT - the total number of CICCSF observations deleted by the USRSCSF exit. The CICCSF file is created from the file segments in the ASG-TMON transaction records.

CICCMR00 RECORDS REJECTED BY USRSCMR EXIT - the total number of CICCMR observations deleted by the USRSCMR exit. The CICCMR file is created from the MRO segments in the ASG-TMON transaction records.

CICCSW00 RECORDS REJECTED BY USRSCSW EXIT - the total number of CICCSW observations deleted by the USRSCSW exit. The CICCSW file is created from ASG-TMON transaction records.

CICCSIN00 RECORDS WRITTEN - the total number of observations written to the CICCSIN00 work file. There is one observation per incident.

CICCSF00 RECORDS WRITTEN - the total number of observations written to the CICCSF00 work file, which contains one observation per file segment processed from a ASG-TMON transaction record.

CICCMR00 RECORDS WRITTEN - the total number of observations written to the CICCMR00 work file, which contains one observation per MRO segment processed from a ASG-TMON transaction record.

CICCSW00 RECORDS WRITTEN - the total number of observations written to the CICCSW00 work file, which contains one observation per ASG-TMON transaction record.

ACTION: None.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 1 5 9 |  
+-----+
```

TEXT: CICCXY00 RECORDS REJECTED BY USRSCSY EXIT
%WRXCSYRJ
CICCXY00 RECORDS WRITTEN
%WRXCSYCT

TYPE: Information

REASON: Display input statistics:

CICCXY00 RECORDS REJECTED BY USRSCSY EXIT -
the total number of CICCXY observations
deleted by the USRSCSY exit. The CICCXY file
is created from ASG-TMON system records.

CICCXY00 RECORDS WRITTEN - the total number of
observations written to the CICCXY00 work
file, which contains one observation per
ASG-TMON system record.

ACTION: None.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 1 6 0 |  
+-----+
```

TEXT: %CICXXUT: GETMAIN FAILED FOR RECORD AREA.
INCREASE REGION SIZE
USER ABEND 998 IS BEING ISSUED

TYPE: Error

REASON: Insufficient virtual area available to region.

ACTION: Increase region size and restart job.

REFERENCES: N/A

```
+-----+
```

| C I C 0 6 1 6 1 |
+-----+

TEXT: DECOMPRESSION ROUTINE LOAD FAILED
 - NOT ABLE TO PROCESS DATA FOR DDNAME %INDDNME
 SEE INSTRUCTIONS REGARDING
 DECOMPRESSION LOAD MODULE REQUIREMENT FOR THIS
 DATA SOURCE
 USER ABEND 998 IS BEING ISSUED

TYPE: Error

REASON: Decompression program not available to DAY040.

ACTION: Copy the decompression program to either
 sharedprefix.MICS.LOAD or a library that is
 part of the DAY040 STEPLIB concatenation, or
 ensure that the program is accessible via the
 LINKLST.

 For ASG-TMON, the required decompression
 program is TMV611 (before Version 8.0).

REFERENCES: Section 6.1

+-----+
| C I C 0 6 1 6 2 |
+-----+

TEXT: DDNAME %INDDNME PROCESSED BY %CICXXXUT NOT IN
 SAS FILE TABLE
 USER ABEND 998 IS BEING ISSUED

TYPE: Error

REASON: The ddname to be processed is not defined to
 SAS.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

+-----+
| C I C 0 6 1 6 3 |
+-----+

TEXT: CA MICS CIC PRODUCT INTERNAL ERROR
 %CICXXXUT UNABLE TO ADDRESS DCB FOR DDNAME

%INDDNME
USER ABEND 998 IS BEING ISSUED

TYPE: Error

REASON: DCB address not found in SAS file table.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

+-----+
| C I C 0 6 1 6 4 |
+-----+

TEXT: CA MICS CIC PRODUCT INTERNAL ERROR
CONNECTOR DISPOSITION NOT SPECIFIED
USER ABEND 998 IS BEING ISSUED

TYPE: Error

REASON: Connector disposition must be CONNS or NOCONNS
for CICXXXUT modules.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

+-----+
| C I C 0 6 1 6 5 |
+-----+

TEXT: CA MICS CIC PRODUCT INTERNAL ERROR CONNECTOR
VALUE EXCEEDS DICTIONARY ARRAY SIZE USER ABEND
998 IS BEING ISSUED

TYPE: Error

REASON: Connector value is greater than the data
dictionary array defined in load module
@\$CICDEP.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

+-----+

```
| C I C 0 6 1 6 9 |
+-----+
```

TEXT: CA MICS CIC PRODUCT INTERNAL ERROR
%CICXXXUT HAS ISSUED MESSAGE %CICUTFBK
USER ABEND 998 IS BEING ISSUED

TYPE: Error

REASON: Unexpected return code from CICxxxUT module.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+
| C I C 0 6 1 7 0 |
+-----+
```

TEXT: USRSCID ARRAY FULL, ARRAY LIMIT=%NBRUNT
ADD UNDEFINED ENTRIES TO CICOPS
RUN CICPGEN, THEN RERUN DAILY

TYPE: Error

REASON: The number of CICSID entries added by the exit
has exceeded the number of available slots in
the CICSID format tables.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+
| C I C 0 6 1 7 1 |
+-----+
```

TEXT: USER EXIT USRSCID SET CICID TO A VALUE THAT
IS NOT ALPHANUMERIC OR -, OR LENGTH HAS A
LENGTH GREATER THAN 4.
CICSID=%CICSID

TYPE: Error

REASON: The CICSID value provided by the exit is not
formatted correctly.

ACTION: The DAILY job terminates with an abend U998 in
step DAY040. Correct the exit code to ensure

that CICSID values are correctly formatted and rerun the failing DAILY job.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 1 7 2 |  
+-----+
```

TEXT: USER EXIT USRSCID SET SELCODE TO A VALUE
 OTHER THAT 0, 4, OR 6'
 SELCODE=%SELCODE'

TYPE: Error

REASON: The SELCODE value provided by the exit is not
 valid.

ACTION: The DAILY job terminates with an abend U998 in
 step DAY040. Correct the exit code to ensure
 that SELCODE values are correct and rerun the
 failing DAILY job.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 1 7 3 |  
+-----+
```

TEXT: USER EXIT USRSCID SET CICSID TO A VALUE THAT
 IN COMBINATION WITH ORGSYSID, HAS BEEN
 PREVIOUSLY DEFINED SYSID=%ORGSYSID
 CICSID=%CICSID APPLID=%CSUAPPL

TYPE: Error

REASON: The CICSID value provided by the exit already
 exists in the PARMs CICOPS member. Each CICSID
 value must be unique to each CICS region
 definition.

ACTION: The DAILY job terminates with an abend U998 in
 step DAY040. Correct the exit code to ensure
 that each CICSID value is unique, and that no
 CICSID value supplied by the exit exists in
 the CICOPS member. Rerun the failing DAILY
 job.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 1 7 4 |  
+-----+
```

TEXT: USER EXIT USRSCID PROVIDED CICSID=%CICSID
 FOR ORGSYSID=%ORGSYSID, APPLID=%CSUAPPL
 PLEASE UPDATE CICSOPS ACCORDINGLY
 AND RUN CICPGEN BEFORE NEXT DAILY RUN

 ...SAMPLE CICOPTS, BE SURE TO SET SELCODE...
 CICOPTS %ORGSYSID %CICSID SMF CMF 0 %CSUAPPL
 ??

TYPE: Warning

REASON: Self explanatory.

ACTION: The CICSID value provided by the exit should
 be added to the PARMS CICOPS member and a
 CICPGEN job should be executed before
 continuing to process additional data from
 this CICS region.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 1 7 5 |  
+-----+
```

TEXT: DATA DICTIONARY RECORDS FOUND FOR ORGSYSID=
 %ORGSYSID, APPLID=%CSUAPPL
 PLEASE IDENTIFY THIS REGION BY INSERTING A
 CICOPTS ENTRY IN CICOPS AND RUNNING CICPGEN.

TYPE: Warning

REASON: Self explanatory.

ACTION: The CICSID value provided by the exit should
 be added to the PARMS CICOPS member and a
 CICPGEN job should be executed before
 continuing to process additional data from
 this CICS region.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 2 1 0 |  
+-----+
```

TEXT: STATEMENT IN ERROR: %CDAVAR
COMMAND IN ABOVE STATEMENT IS INVALID

TYPE: Error

REASON: An invalid control statement was found in
sharedprefix.MICS.SOURCE(CICDEPEL). Valid
control statements are CONTROLVAR, DEFINEID,
and DEFINEIDCOUNT.

ACTION: If error is a result of user modification,
then correct control statement in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise contact CA Technical Support.

REFERENCES: Section 9.1.3.1

```
+-----+  
| C I C 0 6 2 1 1 |  
+-----+
```

TEXT: STATEMENT IN ERROR: %CDAVAR
INFORMATION IN ABOVE STATEMENT IS INSUFFICIENT

TYPE: Error

REASON: An invalid control statement was found in
sharedprefix.MICS.SOURCE(CICDEPEL).

ACTION: If error is a result of user modification,
then correct control statement in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise contact CA Technical Support.

REFERENCES: Section 9.1.3.1

```
+-----+  
| C I C 0 6 2 1 2 |  
+-----+
```

TEXT: STATEMENT IN ERROR: %CDAVAR
CONTROL VARIABLE NAME IN ABOVE STATEMENT IS
NOT DEFINED TO SAS

TYPE: Error

REASON: A control statement in
sharedprefix.MICS.SOURCE(CICDEPEL) referenced
a control variable name that was not defined
to SAS.

ACTION: If error is a result of user modification,
then correct control statement in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise contact CA Technical Support.

REFERENCES: Section 9.1.3.1

```
+-----+
| C I C 0 6 2 1 3 |
+-----+
```

TEXT: STATEMENT IN ERROR: %CDAVAR
THE DATA ID VALUE IS MISSING IN THE ABOVE
DEFINEID STATEMENT

TYPE: Error

REASON: An invalid DEFINEID statement is found in
sharedprefix.MICS.SOURCE(CICDEPEL). It does
not specify the data element ID.

ACTION: If error is a result of user modification,
then correct control statement in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise contact CA Technical Support.

REFERENCES: Section 9.1.3.1

```
+-----+
| C I C 0 6 2 1 4 |
+-----+
```

TEXT: MAXIMUM OF TEN DATA CLASS DEFINITIONS EXCEEDED

TYPE: Error

REASON: The limit of ten data classes as defined in
sharedprefix.MICS.ASM(@\$CICDEP) has been
exceeded.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 2 1 5 |  
+-----+
```

TEXT: STATEMENT IN ERROR: %CDAVAR
SPECIFIED SAS NAME IN ABOVE DEFINEID STATEMENT
IS TOO LONG

TYPE: Error

REASON: An invalid DEFINEID statement is found in
sharedprefix.MICS.SOURCE(CICDEPEL). It
contains a SAS data element name that is
longer than eight characters.

ACTION: If error is a result of user modification,
then correct control statement in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise contact CA Technical Support.

REFERENCES: Section 9.1.3.1

```
+-----+  
| C I C 0 6 2 1 6 |  
+-----+
```

TEXT: REQUIRED SAS CONTROL VARIABLE(S) NOT DEFINED
UNDEFINED CONTROL VARIABLES ARE: %CDAVAR

TYPE: Error

REASON: Not all required control variables for load
module @\$CICDEP have been specified in
sharedprefix.MICS.SOURCE(CICDEPEL).

ACTION: If error is a result of user modification,
then correct control statement in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise contact CA Technical Support.

REFERENCES: Section 9.1.3.1

```
+-----+  
| C I C 0 6 2 1 7 |  
+-----+
```

+-----+

TEXT: SPECIFIED SAS CONTROL VARIABLE(S) NOT FOUND
UNDEFINED SAS VARIABLES ARE: %CDAVAR

TYPE: Error

REASON: Sharedprefix.MICS.SOURCE(CICDEPEL) control
variable refers to a SAS element that is not
defined to SAS.

ACTION: If error is a result of user modification,
then correct control statement in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise contact CA Technical Support.

REFERENCES: Section 9.1.3.1

+-----+

| C I C 0 6 2 1 9 |

+-----+

TEXT: STATEMENT IN ERROR: %CDAVAR
NUMERIC FIELD IN ABOVE STATEMENT IS
TOO LARGE

TYPE: Error

REASON: Numeric value cannot be converted to binary by
load module @\$CICDEP.

ACTION: If error is a result of user modification,
then correct control statement in
sharedprefix.MICS.SOURCE(CICDEPEL) and restart
job. Otherwise contact CA Technical Support.

REFERENCES: Section 9.1.3.1

+-----+

| C I C 0 6 2 2 0 |

+-----+

TEXT: @\$CICDEP UNABLE TO LOCATE CICDEPEL IN SAS FILE
TABLE

TYPE: Error

REASON: Unexpected failure in CICDEPEL processing.

Possibly caused by a dynamic allocation
failure of sharedprefix.MICS.SOURCE(CICDEPEL)
to ddname CICDEPEL.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 2 2 1 |  
+-----+
```

TEXT: VARIABLE IN THE FOLLOWING STATEMENT NOT
DEFINED TO SAS
STATEMENT: %CDAVAR

TYPE: Warning

REASON: Data element defined by the DEFINEID statement
in sharedprefix.MICS.SOURCE(CICDEPEL) is
unknown to SAS. This condition is normal if
the data element is turned off in
sharedprefix.MICS.GENLIB(CICGENIN).

ACTION: If movement of data from the input field to a
SAS data element is desired, define the field
to SAS. If the data element is described in
sharedprefix.MICS.GENLIB(CICGENIN), then
activate the data element. Otherwise, specify
a LENGTH statement for the element in the
USRSLFL exit. This allows you to reference
the element during CICACRT and other CICS user
exits.

REFERENCES: Section 9.1.2

```
+-----+  
| C I C 0 6 2 2 2 |  
+-----+
```

TEXT: CICS JOURNAL RECORD INPUT
%XJRNOTKP
PHYSICAL STATISTICS RECORDS INPUT
%XSTPHSCT
INCOMPLETE STATISTICS RECORDS INPUT
%XSTINCMP
REQUESTED STATISTICS RECORDS READ
%XSTREQST

LOGICAL STATISTICS RECORDS READ
%dRXSTLCT
RECORDS REJECTED BY USRSCST EXIT
%dRXCSTRJ
TYPE 2 STATISTICS RECORDS READ
%dRXST2
TYPE 6 STATISTICS RECORDS READ
%dRXST6
TYPE 8 STATISTICS RECORDS READ
%dRXST8
TYPE 9 STATISTICS RECORDS READ
%dRXST9
TYPE 10 STATISTICS RECORDS READ
%dRXST10
TYPE 27 STATISTICS RECORDS READ
%dRXST27
TYPE 45 STATISTICS RECORDS READ
%dRXST45
TYPE 48 STATISTICS RECORDS READ
%dRXST48
TYPE 56 STATISTICS RECORDS READ
%dRXST56
TYPE 57 STATISTICS RECORDS READ
%dRXST57

TYPE: Information

REASON: Display input processing statistics for CICS
release 3.1.1 and higher:

CICS JOURNAL RECORD INPUT - the total number
of CICS journal records (type 110, subtype 0)
found in SMF input. This is the number of
physical records.

PHYSICAL STATISTICS RECORDS INPUT - the total
number of CICS statistics records (type 110,
subtype 3) found in input. This is the number
of physical records.

INCOMPLETE STATISTICS RECORDS INPUT - the
total number of CICS statistics records (type
110, subtype 3) found with a flag indicating
an incomplete record. These records were not
processed by the CA MICS Analyzer Option for
CICS.

REQUESTED STATISTICS RECORDS INPUT - the total
number of "requested" statistics records found

in input. These records were produced by issuing an EXEC CICS PERFORM STATISTICS command. The records were not processed by the CA MICS Analyzer Option for CICS.

LOGICAL STATISTICS RECORDS READ - the total number of CICS statistics records read from input. This is the number of logical records read (each physical record can contain multiple logical records).

RECORDS REJECTED BY USRSCST EXIT - the total number of CICS statistics records rejected by the USRSCST exit. This is the number of logical records.

TYPE 2 STATISTICS RECORDS READ - the total number of storage manager DSA statistics records read. This is the number of logical records. Type 2 record is available at CICS/ESA 4.1 and higher releases.

TYPE 6 STATISTICS RECORDS READ - the total number of storage manager task subpool statistics records read. This is the number of logical records. Type 6 record is available at CICS/ESA 4.1 and higher releases.

TYPE 8 STATISTICS RECORDS READ - the total number of storage manager task subpool statistics records read. This is the number of logical records. Type 8 record is available in CICS/ESA Version 3 releases only.

TYPE 9 STATISTICS RECORDS READ - the total number of storage manager DSA statistics records read. This is the number of logical records. Type 9 record is available in CICS/ESA Version 3 releases only.

TYPE 10 STATISTICS RECORDS READ - the total number of transaction manager statistics records read. This is the number of logical records. Type 10 record is available at CICS/ESA 4.1 and higher releases.

TYPE 27 STATISTICS RECORDS READ - the total number of loader global statistics records read. This is the number of logical records.

TYPE 45 STATISTICS RECORDS READ - the total number of transient data global statistics records read. This is the number of logical records.

TYPE 48 STATISTICS RECORDS READ - the total number of temporary storage global statistics records read. This is the number of logical records.

TYPE 56 STATISTICS RECORDS READ - the total number of dispatcher statistics records read. This is the number of logical records. Type 56 record is available at CICS/ESA 4.1 and higher releases.

TYPE 57 STATISTICS RECORDS READ - the total number of dispatcher statistics records read. This is the number of logical records. Type 57 record is available in CICS/ESA Version 3 releases only.

ACTION: None

REFERENCES: N/A

```
+-----+
| C I C 0 6 2 2 3 |
+-----+
```

TEXT: CICC000 RECORDS REJECTED BY USRSCSY (STAT
REC) %WRXCSYRJ
CICC000 RECORDS WRITTEN (CREATED FROM STAT
REC) %WRXCSYCT

TYPE: Information

REASON: Display input processing statistics.

ACTION: None

REFERENCES: N/A

```
+-----+
| C I C 0 6 3 1 1 |
+-----+
```

TEXT: MEMORY REQUEST MUST BE GET OR PUT

TYPE: Error

REASON: Invalid function request is issued for
CICVMAM. Function request must be GET or PUT.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 3 1 2 |  
+-----+
```

TEXT: CICVMAM KEY PARAMETER IS INVALID

TYPE: Error

REASON: Key parameter must be 14 bytes long.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 3 1 3 |  
+-----+
```

TEXT: CICVMAM APPL ID PARAMETER IS INVALID

TYPE: Error

REASON: APPLID parameter must be eight bytes long.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 3 1 4 |  
+-----+
```

TEXT: CICVMAM ORGSYSID PARAMETER IS INVALID

TYPE: Error

REASON: CICSID parameter must be four bytes long.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 3 1 5 |  
+-----+
```

TEXT: CICVMAM RELEASE PARAMETER IS INVALID

TYPE: Error

REASON: Release parameter must be four bytes long.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 3 1 7 |  
+-----+
```

TEXT: CICVMAM ARRAY ELEMENT SIZE PARAMETER
IS INVALID

TYPE: Error

REASON: Array size (number of elements) specified in
first call to CICVMAM does not match array
size parameter value in current call.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 3 1 8 |  
+-----+
```

TEXT: CICVMAM DATA ID ARRAY IS INVALID

TYPE: Error

REASON: Data ID array element length must be three
bytes in length.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 3 1 9 |  
+-----+
```

TEXT: CICVMAM TYPE ARRAY IS INVALID

TYPE: Error

REASON: Type array element length must be one byte in length.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 3 3 0 |  
+-----+
```

TEXT: CICVMAM UNABLE TO INITIALIZE.
\$USERCOM+200 CONTAINS INCORRECT CODE.

TYPE: Error

REASON: Offset in SAS user communication area utilized by load module CICVMAM has been modified by another module.

ACTION: Contact CA Technical Support.

REFERENCES: N/A

```
+-----+  
| C I C 0 6 3 3 1 |  
+-----+
```

TEXT: NO MEMORY FOR ADDITION OF ARRAY ENTRY.
REGION MUST BE INCREASED.

TYPE: Error

REASON: GETMAIN failed during data dictionary array

processing in load module CICVMAM.

ACTION: Increase region size and restart job.

REFERENCES: Section 9.1.3.1

```
+-----+  
| C I C 0 6 4 0 1 |  
+-----+
```

TEXT: FILE %FILE IS REQUIRED FOR CA MICS ACCOUNTING
BUT IS INACTIVE.
DAY040 STEP IS ABORTED.

TYPE: Error

REASON: A CA MICS Analyzer Option for CICS file that
is used by the CA MICS Accounting and
Chargeback Option has been deactivated.

ACTION: Reactivate the file or change the data source
for your CA MICS Accounting and Chargeback
files.

REFERENCES: Section 10.1

```
+-----+  
| C I C 0 6 9 0 0 |  
+-----+
```

TEXT: =====
CMF DATA DICTIONARY FILE CICCDC01 RECORDS NOT
REFERENCED AFTER %CICCDCTH DAYS WILL EXPIRE

%CCDCDL RECORDS HAVE EXPIRED AND WERE DELETED
=====

TYPE: Information

REASON: Display the data dictionary record expiration
interval and the number of records that have
expired due to the interval specification.
The expiration interval is defined in
prefix.MICS.PARMS(CICTHRSH).

ACTION: None.

REFERENCES: Sections 7.3.8, 9.1.2, and 9.1.3.2

```
+-----+  
| C I C 0 6 9 1 0 |  
+-----+
```

```
TEXT: =====  
INTERNAL ERROR - CICCDC RECORD DOES NOT HAVE  
A CORRESPONDING CDA ENTRY  
CDA KEY IS %CDAKEY  
USER ABEND 998 ISSUED  
=====
```

TYPE: Error

REASON: Internal error in load module @\$CICDEP.

ACTION: Contact CA Technical Support.

REFERENCES: Section 9.3.2

```
+-----+  
| C I C 0 6 9 2 0 |  
+-----+
```

```
TEXT: =====  
%CICDCDL CICCDC RECORDS WERE DELETED DUE TO  
INVALID DATA - SEE CORRESPONDING CIC060XX  
MESSAGE(S)  
=====
```

TYPE: Information

REASON: Invalid data dictionary records have been
deleted. See CIC060xx messages for
explanations of error.

ACTION: None

REFERENCES: N/A

```
+-----+  
| C I C 0 6 9 3 0 |  
+-----+
```

```
TEXT: =====  
CICCDC01 FILE WAS NOT CONVERTED TO CIC4300  
FORMAT  
USER ABEND 998 ISSUED  
=====
```

```

=====
TYPE:      Error

REASON:    The DETAIL.CICCDC01 file was not converted to
            CIC4300 format during the installation of
            CIC4300.

ACTION:    Convert the DETAIL.CICCDC01 file by submitting
            sharedprefix.MICS.PSP.CNTL(CICCDCCV).

REFERENCES: N/A

```

```

+-----+
| C I C 0 6 9 9 9 |
+-----+

```

```

TEXT:      =====

TYPE:      Information

REASON:    This is a header for messages in MICSLOG.

ACTION:    None.

REFERENCES: N/A

```

```

+-----+
| C I C 0 7 0 3 0 |
+-----+

```

```

TEXT:      INVALID CONTROL STATEMENT: %KEYWRD01
            DDNAME LENGTH GREATER THAN 8 CHARACTERS
            CONTROL STATEMENT BYPASSED

TYPE:      Error

REASON:    Invalid ddname is specified in
            prefix.MICS.CNTL(CICILIST). Length of ddname
            must not exceed eight characters.

ACTION:    Specify the correct ddname in
            prefix.MICS.CNTL(CICILIST) and rerun job.

REFERENCES: Section 9.3.1

```

```

+-----+

```


| C I C 0 7 0 3 1 |
+-----+

TEXT: INVALID CONTROL STATEMENT: %KEYWRD01
 DDNAME BEGINS WITH A NUMERIC VALUE
 CONTROL STATEMENT BYPASSED

TYPE: Error

REASON: Invalid ddname is specified in
 prefix.MICS.CNTL(CICILIST).

ACTION: Correct ddname in prefix.MICS.CNTL(CICILIST)
 and rerun job.

REFERENCES: Section 9.3.1

+-----+
| C I C 0 7 0 3 2 |
+-----+

TEXT: INVALID CONTROL STATEMENT: %KEYWRD01 %KEYWRD02
 DATA SOURCE NOT CMF, MON, OR TCE CONTROL
 STATEMENT BYPASSED

TYPE: Error

REASON: Invalid data source is specified in
 prefix.MICS.CNTL(CICILIST). Valid data
 sources are CMF, MON, and TCE.

ACTION: Specify the correct monitor data source in
 prefix.MICS.CNTL(CICILIST) and rerun job.

REFERENCES: Section 9.3.1

+-----+
| C I C 0 7 0 3 3 |
+-----+

TEXT: INVALID CONTROL STATEMENT: %KEYWRD03
 RELATIVE RECORD NUMBER NOT NUMERIC
 CONTROL STATEMENT BYPASSED

TYPE: Error

REASON: Invalid relative record number is specified
 for job CICILIST. It must be a number between

1 and 999999999.

ACTION: Specify the correct relative record number in
prefix.MICS.CNTL(CICILIST) and rerun job.

REFERENCES: Section 9.3.1

```
+-----+
| C I C 0 7 0 3 4 |
+-----+
```

TEXT: INVALID CONTROL STATEMENT: %KEYWRD03
RELATIVE RECORD NUMBER LT 1 OR GT 999999999
CONTROL STATEMENT BYPASSED

TYPE: Error

REASON: Invalid relative record number is specified
for job CICILIST. It must be a number between
1 and 999999999.

ACTION: Specify the correct relative record number in
prefix.MICS.CNTL(CICILIST) and rerun job.

REFERENCES: Section 9.3.1

```
+-----+
| C I C 0 7 0 3 5 |
+-----+
```

TEXT: PROCESSING REQUEST FOR:
%DDN %MON

TYPE: Information

REASON: Input processing message that shows the input
ddname and ASG-TMON source specified for the
CICILIST job.

ACTION: None.

REFERENCES: Section 9.3.1

```
+-----+
| C I C 0 7 0 3 6 |
+-----+
```

TEXT: %RECNUM

TYPE: Information

REASON: This message is issued in conjunction with message CIC07035 to show the relative record numbers specified for job CICILIST.

ACTION: None.

REFERENCES: Section 9.3.1

```
+-----+  
| C I C 0 7 0 3 7 |  
+-----+
```

TEXT: NO CONTROL STATEMENTS

TYPE: Error

REASON: No print requests found in CONTROL DD.

ACTION: Add print requests to CONTROL DD in prefix.MICS.CNTL(CICILIST).

REFERENCES: Section 9.3.1

```
+-----+  
| C I C 0 7 0 3 8 |  
+-----+
```

TEXT: FILE CONTAINS LESS THAN %RECNUM RECORDS

TYPE: Information

REASON: End of file was reached before the relative record number was read.

ACTION: Specify the correct relative record number in prefix.MICS.CNTL(CICILIST) and check the JCL to ensure that the correct input data set name is specified. Rerun job.

REFERENCES: Section 9.3.1

```
+-----+  
| C I C 0 7 0 3 9 |  
+-----+
```

TEXT: SEE DUMP OF REQUESTED RECORDS IN SASLOG

TYPE: Information

REASON: The output listing of the CICILIST job is located in the SAS log. This is normal since the record listing is produced by the SAS LIST statement.

ACTION: None.

REFERENCES: Section 9.3.1

```
+-----+  
| C I C 0 7 0 5 0 |  
+-----+
```

TEXT: SMF TYPE 111 CICS TRANSACTION GATEWAY
PROCESSING ACTIVE

REASON: This message indicates that a valid GATEWAY
ACTIVE statement was found in CICOPS.

TYPE: Information

ACTION: None.

REFERENCES: Section 7.3.2.6

```
+-----+  
| C I C 0 7 0 5 1 |  
+-----+
```

TEXT: CICOPS---> %LINE
"GATEWAY" KEYWORD MUST BE FOLLOWED BY "ACTIVE"
OR "APPLID" KEYWORDS
--> %KEYWRD02 FOUND INSTEAD

REASON: The GATEWAY statement only has two forms:
GATEWAY ACTIVE and GATEWAY APPLID arg1 arg2.
The KEYWRD02 variable was found instead of the
expected ACTIVE or APPLID.

TYPE: Error

ACTION: Correct the error and rerun job.

REFERENCES: Section 7.3.2.6

```
+-----+  
| C I C 0 7 0 5 2 |  
+-----+
```

TEXT: CICOPS---> %LINE
 GATEWAY APPLID NAME NOT 1 TO 8 CHARACTERS IN
 LENGTH
 --> %KEYWRD03

REASON: The first argument for the GATEWAY APPLID
 statement must be the APPLID of a CICS
 Transaction Gateway address space. It must be
 from 1 to 8 bytes in length. Variable KEYWRD03
 shows the incorrect entry.

TYPE: Error

ACTION: Correct the APPLID name and rerun job.

REFERENCES: Section 7.3.2.6

```
+-----+  
| C I C 0 7 0 5 3 |  
+-----+
```

TEXT: CICOPS---> %LINE
 GATEWAY ID (CTGID) NOT 1 TO 4 CHARACTERS IN
 LENGTH
 --> %KEYWRD04

REASON: The second argument for the GATEWAY APPLID
 statement is the CTGID you want to define for
 the CICS Transaction Gateway APPLID. It must
 be from 1 to 4 bytes in length. Variable
 KEYWRD04 shows the incorrect entry.

TYPE: Error

ACTION: Correct the CTGID name and rerun job.

REFERENCES: Section 7.3.2.6

```
+-----+  
| C I C 0 7 0 5 4 |  
+-----+
```

TEXT: GATEWAY APPLID STATEMENT(S) FOUND BUT NO
GATEWAY ACTIVE STATEMENT
ADD A "GATEWAY ACTIVE" STATEMENT TO CICOPS TO
ACTIVATE GATEWAY PROCESSING

REASON: You must have a GATEWAY ACTIVE statement in
order to add GATEWAY APPLID statements to
prefix.MICS.PARMS(CICOPS).

TYPE: Error

ACTION: Add a GATEWAY ACTIVE statement and rerun job.

REFERENCES: Section 7.3.2.6

```
+-----+  
| C I C 0 7 0 5 5 |  
+-----+
```

TEXT: "GATEWAY ACTIVE" STATEMENT FOUND, BUT NO
"GATEWAY APPLID" STATEMENT(S)
ALL GATEWAY SMF TYPE 111 RECORDS WILL BE
DISCARDED DURING DAILY DAY040 STEP
DAY040 MICSLOG WILL CONTAIN LIST OF ALL
GATEWAY SYSID/APPLIDS ENCOUNTERED

REASON: The GATEWAY ACTIVE statement enables DAY040 to
process SMF type 111 CICS Transaction Gateway
records, but without GATEWAY APPLID
statements, no output to the CTGGSA file will
occur. The DAY040 MICSLOG, however, will list
all Gateway APPLIDs encountered in the raw
data to assist you in defining the GATEWAY
APPLID statements.

TYPE: Information

ACTION: None

REFERENCES: Section 7.3.2.6

```
+-----+  
| C I C 0 7 0 5 6 |  
+-----+
```

TEXT: CICOPS---> %LINE
GATEWAY CTGID VALUE ALREADY DEFINED AS A

CICSID IN A CICOPTS STATEMENT
CTGID VALUE: %KEYWRD04

REASON: The CTGID value specified on the GATEWAY
APPLID statement, displayed with variable
KEYWRD04, was already used as a CICSID value
in a previous CICOPTS statement.

TYPE: Error

ACTION: Select a different value for CTGID and rerun
job.

REFERENCES: Section 7.3.2.6

```
+-----+  
| C I C 0 7 0 5 7 |  
+-----+
```

TEXT: CICOPS---> %LINE
GATEWAY APPLID ALREADY DEFINED IN PREVIOUS
GATEWAY APPLID STATEMENT
DUPLICATED APPLID VALUE: %KEYWRD03

REASON: The APPLID value in the GATEWAY APPLID
statement, listed with variable KEYWRD03, was
already used in a previous GATEWAY APPLID
statement. Each GATEWAY APPLID statement must
define the CTGID for a unique CICS Transaction
Gateway APPLID value.

TYPE: Error

ACTION: Remove the redundant GATEWAY APPLID statement,
or change APPLID to a different value and
rerun job.

REFERENCES: Section 7.3.2.6

```
+-----+  
| C I C 0 7 0 5 8 |  
+-----+
```

TEXT: CICOPS---> %LINE
GATEWAY CTGID ALREADY DEFINED IN PREVIOUS
GATEWAY APPLID STATEMENT
DUPLICATE CTGID VALUE: %KEYWRD04

REASON: The CTGID value in the GATEWAY APPLID statement, listed with variable KEYWRD04, was already used in a previous GATEWAY APPLID statement. Each GATEWAY APPLID statement must define a unique CTGID value.

TYPE: Error

ACTION: Change the CTGID to a different value and rerun job.

REFERENCES: Section 7.3.2.6

```
+-----+
| C I C 0 7 0 5 9 |
+-----+
```

TEXT: CICOPS---> %LINE
CICOPTS CICSID VALUE ALREADY DEFINED AS A
CTGID IN A GATEWAY APPLID STATEMENT
CICSID VALUE: %KEYWRD03

REASON: The CICSID value in the CICOPTS statement, listed with variable KEYWRD03, was already used as a CTGID value in a previous GATEWAY APPLID statement. CICOPTS statements and GATEWAY APPLID statements must define unique CICSID and CTGID values.

TYPE: Error

ACTION: Change the CICSID value to a unique value, not used by any GATEWAY APPLID CTGID values, and rerun job.

REFERENCES: Section 7.3.2.1, 7.3.2.6

```
+-----+
| C I C 0 7 0 6 0 |
+-----+
```

TEXT: CICOPS---> %LINE
"GATEWAY ACTIVE" STATEMENT DOES NOT SUPPORT
ADDITIONAL ARGUMENTS
--> %KEYWRD03 FOUND

REASON: The GATEWAY ACTIVE statement was followed by an extraneous argument, listed with variable

KEYWRD03.

TYPE: Error

ACTION: Remove the extraneous arguments from the
GATEWAY ACTIVE statement and rerun job.

REFERENCES: Section 7.3.2.1

```
+-----+  
| C I C 0 7 0 6 1 |  
+-----+
```

TEXT: CICOPS---> %LINE
GATEWAY APPLID STATEMENT REQUIRES TWO
NON-BLANK ARGUMENTS
AT LEAST ONE ARGUMENT MISSING:
ARGUMENT 1 (APPLID) --> %KEYWRD03
ARGUMENT 2 (CTGID) --> %KEYWRD04

REASON: The GATEWAY APPLID statement requires two
arguments, but either one or none was found
(displayed by variables KEYWRD03 and
KEYWRD04).

TYPE: Error

ACTION: Correct the GATEWAY APPLID statement and rerun
job.

REFERENCES: Section 7.3.2.1

```
+-----+  
| C I C 0 7 0 6 2 |  
+-----+
```

TEXT: CICOPS---> %LINE
FIRST CHARACTER OF GATEWAY CTGID MUST BE ALPHA
(A-Z)
ILLEGAL FIRST CHARACTER FOUND
--> %KEYWRD04

REASON: The first character of the CTGID value on a
GATEWAY APPLID statement must begin with a
letter (A-Z). Some other character was found,
as shown with variable KEYWRD04.

TYPE: Error

ACTION: Correct the CTGID value and rerun job.

REFERENCES: Section 7.3.2.1

```
+-----+
| C I C 0 7 0 6 3 |
+-----+
```

TEXT: CICOPS---> %LINE
GATEWAY CTGID CAN ONLY CONTAIN ALPHANUMERIC
CHARACTERS (A-Z,0-9) OR UNDERSCORES (_)
ILLEGAL CHARACTER(S) FOUND
--> %KEYWRD04

REASON: The CTGID value on a GATEWAY APPLID statement
contained characters other than letters (A-Z),
numbers (0-9), and underscores (_), as shown
by variable KEYWRD04.

TYPE: Error

ACTION: Correct the CTGID value and rerun job.

REFERENCES: Section 7.3.2.1

```
+-----+
| C I C 0 7 0 6 4 |
+-----+
```

TEXT: CICOPS---> %LINE
ONLY ONE GATEWAY ACTIVE STATEMENT ALLOWED.
MULTIPLE GATEWAY ACTIVE STATEMENTS FOUND.

TYPE: Error

REASON: More than one GATEWAY ACTIVE statement was
found in prefix.MICS.PARMS(CICOPS). Only one
GATEWAY ACTIVE statement is allowed.

ACTION: Remove the redundant GATEWAY ACTIVE statement
and rerun job.

REFERENCES: Section 7.3.2.1

```
+-----+
| C I C 0 7 0 6 5 |
+-----+
```

+-----+

TEXT: INPUTSMF DD FOUND IN INPUTCIC, BUT IN CICOPS
THERE ARE NO CICOPTS STATEMENTS CALLING FOR
CMF DATA, NOR IS THERE A GATEWAY ACTIVE
STATEMENT

TYPE: Warning

REASON: Prefix.MICS.PARMS(INPUTCIC) contains a
//INPUTSMF DD statement, but there are no
CICOPTS statements indicating SMF data input,
nor is there a GATEWAY ACTIVE statement in
prefix.MICS.PARMS(CICOPS).

ACTION: Review prefix.MICS.PARMS(INPUTCIC) and remove
the //INPUTSMF DD statement to prevent
unnecessary data allocation during DAY040
processing.

REFERENCES: Sections 7.3.2, 7.3.3, and 7.3.4

+-----+

| C I C 0 7 0 8 0 |

+-----+

TEXT: +=====+
SMF TYPE 111 GATEWAY RECORD PROCESSING DETAILS

TYPE 111 GATEWAY RECORDS INPUT %XGTWRECS
RECORDS REJECTED BY USRSGSA EXIT %WRXGSARJ
RECORDS REJECTED DUE TO UNDEFINED GATEWAY
APPLID %GXTWRJCT

REASON: Display input statistics for the SMF type 111
CICS Transaction Gateway records.

TYPE: Information

ACTION: None

REFERENCES: N/A

+-----+

| C I C 0 7 0 8 1 |

+-----+

TEXT: +-----+

Rejected CICS Transaction Gateway
data---APPLIDs not defined in CICOPS
ORGSYSID APPLID RECORD COUNT

REASON: While processing SMF type 111 CICS Transaction Gateway records, Gateway APPLIDs were encountered that were not defined with GATEWAY APPLID statements in prefix.MICS.PARMS(CICOPS). MICSLOG Message CIC07082W follows this message and lists the APPLIDs of the rejected SMF type 111 records.

TYPE: Warning

ACTION: Code GATEWAY APPLID statements for the listed APPLIDs if processing is desired.

REFERENCES: Section 7.3.2.1

+-----+
| C I C 0 7 0 8 2 |
+-----+

TEXT: %RJRORG %RJRAPI %XRJRCNT

REASON: While processing SMF type 111 CICS Transaction Gateway records, Gateway APPLIDs were encountered that were not defined with GATEWAY APPLID statements in prefix.MICS.PARMS(CICOPS). MICSLOG Message CIC07081W preceded this message and provided a header for the three variables shown above:

RJRORG - Original SYSID
RJRAPI - CICS Transaction Gateway APPLID
XRJRCNT - Count of discarded SMF type 111 records

TYPE: Warning

ACTION: Code GATEWAY APPLID statements for the listed APPLIDs in prefix.MICS.PARMS(CICOPS) if processing is desired. Then run prefix.MICS.CNTL(CICPGEN).

REFERENCES: Section 7.3.2.1

+-----+

```
| C I C 0 7 0 8 3 |
+-----+
```

TEXT: More than 50 undefined Gateway APPLIDs encountered.
First fifty listed above.

REASON: More than 50 CICS Transaction Gateway APPLIDs were encountered that were not defined in prefix.MICS.PARMS(CICOPS). The first 50 were displayed with MICSLOG messages CIC07081W and CIC07082W.

TYPE: Warning

ACTION: Code GATEWAY APPLID statements for the CICS Transaction Gateway APPLIDs listed with messages CIC07081W and CIC07082W in prefix.MICS.PARMS(CICOPS) if processing is desired. Then run prefix.MICS.CNTL(CICPGEN). The next DAY040 step will list any additional undefined Gateway APPLIDs.

REFERENCES: Section 7.3.2.1

```
+-----+
| C I C 0 7 0 8 4 |
+-----+
```

TEXT: INPUT PROCESSING SUMMARY FOR DDNAME %INDDNME
TYPE 110 CICS RECORDS %XMONRECS
TYPE 111 TRANSACTION GATEWAY RECORDS %XGTWRECS
NON-COMPONENT SMF RECORD TYPES %EXCRECS
=====

TOTAL SMF RECORDS INPUT	%XINPRECS

+=====+

SMF TYPE 110 Record Processing Details
CMF Processing not active for this unit

REASON: Display input statistics for the DDNAME indicated by the variable \$INDDNME when CMF processing is not active for the unit. This message is followed by message CIC07080I, which provides additional SMF type 111 record processing information.

Variable %XMONRECS displays the total number of SMF type 110 records found in the input. This includes CICS statistics and journal

records.

Variable %XGTWRECS displays the total number of SMF type 111 CICS Transaction Gateway records found in the input.

Variable %EXCRECS displays the total number of non-component records found in the input.

Variable %XINPRECS displays the total of 110, 111, and non-component records.

TYPE: Information

ACTION: None

REFERENCES: N/A

```
+-----+
| C I C 0 7 0 8 5 |
+-----+
```

TEXT: User Exit USRSCID generated a CICSID value that was already defined as a Transaction Gateway ID (CTGID) with a GATEWAY APPLID statement in CICOPS
SYSID=%ORGSYSID CICSID=%CICSID APPLID=%CSUAPPL

REASON: The USRSCID exit, used to dynamically generate CICSID values for CICS regions not defined in prefix.MICS.PARMS(CICOPS), generated a CICSID value that has already been defined as a GATEWAY CTGID value via a GATEWAY APPLID statement in CICOPS.

The redundant CICSID value is displayed with variable CICSID.

TYPE: Error

ACTION: Either update the user code in the USRSCID exit to avoid coding CICSID values that match GATEWAY CTGID values, or update the GATEWAY APPLID statement with a CTGID value that will not be generated by the USRSCID exit.

REFERENCES: Section 7.3.2.1, 10.2.1

```
+-----+  
| C I C 0 7 0 8 6 |  
+-----+
```

TEXT: +=====+

REASON: Used to format MICSL0G message text.

TYPE: Information

ACTION: None

REFERENCES: N/A

```
+-----+  
| C I C 0 7 0 8 7 |  
+-----+
```

TEXT: +=====+

SMF TYPE 111 GATEWAY RECORD PROCESSING DETAILS
Gateway Processing not active for this unit

REASON: No details for SMF type 111 records processed
because GATEWAY processing is not active for
the unit.

TYPE: Information

ACTION: None

REFERENCES: N/A

Appendix B: DATA DICTIONARY

The Data Dictionary is only available at your site, where it has been customized to your configuration and your product change level.

To see the Data Dictionary at your site, follow the instructions under Document Browse in the Document Access Guide.