CA MICS® Resource Management

Analyzer Option for VM/CMS 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 9

1.1 Primary Areas of Application 12
1.2 Major Features 13
1.3 Reporting and Inquiry Facilities 15
1.4 Files Overview 22
1.5 Product Prerequisites 31
1.6 Benefits 31

Chapter 2: USAGE GUIDELINES 35

2.1 Operating System Concepts 36
2.2 Product Use 40
2.2.1 Data Analysis 40
2.2.2 Performance Analysis 44
2.2.3 Analyzer Reporting Options 54
2.2.4 Accounting and Chargeback Interface 54
2.2.5 Capacity Planner Interface 57

Chapter 3: REPORTS 59

3.1 Standard Analysis Reports 59
3.1.1 Running Standard Analysis Reports 60
3.1.2 VM Performance Overview Report 61
3.2 MICF Inquiries 64
3.2.1 VM MICF Inquiry Descriptions 65
3.2.2 How to Modify Standard MICF Inquiries 142
3.3 VM Management Objective Reports 148
3.3.1 Daily VM Reports 150
3.3.2 Weekly VM Reports 161
3.3.3 Monthly VM Reports 173
3.3.4 VM Management Objective Report Options 184

Chapter 4: EXCEPTIONS 191

4.1 Introduction 192
4.2 Standard Exceptions 194
4.3 Exception Descriptions 195

Chapter 5: FILES 207

5.1 VM Data Element Naming Conventions 209
5.2 VMC Information Area Files 213
5.2.1 VM User Resource Accounting File (VMCCMU) 214
5.2.2 VM Device Accounting File (VMCDAC) 219
5.2.3 VM RSCS Accounting File (VMCVRA) 227
5.2.4 VM Network Terminal Usage File (VMCNTU) 233
5.2.5 VM Network Data Transmissions File (VMCNDT) 238
5.2.6 VM Exception Activity File (VMCVXC) 243
5.3 VMX Information Area Files 248
5.3.1 VMX System Configuration File (VMXSCF) 253
5.3.2 VMX Processor Configuration File (VMXPCF) 259
5.3.3 VMX System Global Data File (VMXSGP) 263
5.3.4 VMX System Processor Data File (VMXSPP) 279
5.3.5 VMX Device Configuration File (VMXDCF) 289
5.3.6 VMX Device Activity File (VMXXDV) 294
5.3.7 VMX Device SEEKS Data File (VMXXDS) 302
5.3.8 VMX CPVOL Configuration Data File (VMXACF) 308
5.3.9 VMX CP Owned Volume Data File (VMXCPV) 312
5.3.10 VMX User Data Configuration File (VMXVXF) 318
5.3.11 VMX User Activity Data Files (VMXVXU and VMX_VU) 325
5.3.12 VMX User Transaction Data Files (VMXVXT and VMX_VT) 337
5.3.13 VMX Channel Path Contention File (VMXCHC) 348
5.3.14 VMX Block Paging Data File (VMXBPG) 352
5.3.15 VMX Shared Storage Configuration File (VMXXSF) 358
5.3.16 VMX Shared Storage Management File (VMXXSS) 362
5.3.17 VMX System Timer Pop File (VMXSTP) 367
5.3.18 VMX DASD Cache Activity File (VMXCDI) 372
5.3.19 VMX Paging Cache Activity File (VMXCDP) 379
5.3.20 VMX Logical Partition Activity File (VMXLPR) 385
5.3.21 VMX Logical Processor Activity File (VMXLCP) 391
5.3.22 VMX Scheduler Parameters File (VMXSCX) 396
5.3.23 VMX Channel Path Measurement File (VMXCPM) 400
5.3.24 VMX Shared Address Space File (VMXASC) 405
5.4 LNX Information Area Files 411
5.4.1 Linux Process Activity File (LNXLXP) 411

Chapter 6: DATA SOURCES 417

6.1 Data Collector Options and Operation 417
6.1.1 VM Account Data Options 418
6.1.2 VM Monitor Data Options 419

6.1.3 Velocity Monitor Data Options 423
6.1.4 CA VM:Account Operation 424
6.2 Processing Considerations 425
6.2.1 Supplying Data to CA MICS 425
6.2.2 Data Volume - VM Monitor 425
6.2.3 Dropping Input Records 426
6.2.4 Monitor Suspend Considerations and Handling 428
6.2.5 Consolidated Monitor Input 429
6.2.6 Configuration Record Requirements 430
6.3 Record Descriptions 432
6.3.1 VM Account Record Descriptions 432
6.3.2 VM Monitor Record Descriptions 439
6.3.3 CA VM:Account Record Descriptions 446
6.4 Operating System Details 447
6.4.1 Instruction Simulation 447
6.4.2 Shadow Table Maintenance 450
6.4.3 PR/SM and LPAR 452
6.4.4 Virtual Machine CPU Time vs. Linux CPU Time 459

Chapter 7: PARAMETERS 461

7.1 Complex Level Parameters 462
7.1.1 Account Codes 462
7.1.2 Component Generation Parameters (VMCGENIN) 472
7.2 Unit Level Parameters 476
7.2.1 Component Options (VMCOPS) 477
7.2.2 Component-Specific JCL Information (INPUTVMC) 555
7.2.3 Database Space Modeling (DBMODEL) 558
7.2.4 CA MICS JCL Parameters (JCLDEF) 561
7.2.5 CA MICS System Identification (SYSID) 562

Chapter 8: INSTALLATION 565

Chapter 9: PROCESSING 567

9.1 Processing Considerations 567
9.2 Analyzer Daily Update Flow 568
9.2.1 Input Raw Account Data 569
9.2.2 Account Post-Processing 573
9.2.3 Input Raw VM Monitor Data 574
9.2.4 VM Monitor Post-Processing 580
9.2.5 Consolidation 589

9.2.6 Database Time-Span Update 589
9.2.7 File Aging 590
9.3 DAY080 Physical Structure 590
9.4 Daily Update Physical Structure 591
9.5 Account Physical Structure 592
9.6 VM Monitor Processing Physical Structure 594

Chapter 10: MODIFICATION 597

10.1 Parameter Modification 597
10.1.1 Add or Delete a Data Source for a System 598
10.1.2 Add or Change Serial Numbers for a System 599
10.2 Standard User Exits 599
10.2.1 General Exits 606
10.2.2 Analyzer Input Exits 609
10.2.3 Analyzer Parameter-Related Exits 618
10.2.4 Analyzer Output Exits 621
10.2.5 Special Exits 630

Appendix A: MESSAGES 637

Messages 637

Chapter 1: OVERVIEW

The CA MICS Analyzer Option for VM/CMS helps you to manage VM and CMS systems by processing information it collects from multiple sources. CA MICS automatically integrates this information with other information it maintains. This is possible because it stores information in a format that is common to all CA MICS products.

The CA MICS Analyzer Option for VM/CMS provides the capability to interpret, consolidate, store, and report on VM operating system management data gathered from the following data sources:

- IBM z/VM Monitor
- IBM z/VM Account
- IBM Remote Spooling & Communications Subsystem (RSCS)
- IBM VM VTAM Communications Network Application (VCNA) and VM SNA Communications Services (VSCS)
- CA VM:Account
- Velocity Software ESAMON and zVPS Monitor Data

In this guide, unless otherwise specified, the term VM means the ESA feature of VM, whichever is appropriate. z/VM is the successor to VM/ESA.

The CA MICS Analyzer Option for VM/CMS tracks system problems and can be used to perform an in-depth audit of users of various VM and CMS facilities. In addition, it produces a series of management and analysis reports that can be tailored for all management levels. As one of the CA MICS Data Integration Applications, it processes raw data for analyzing and managing specific technologies.

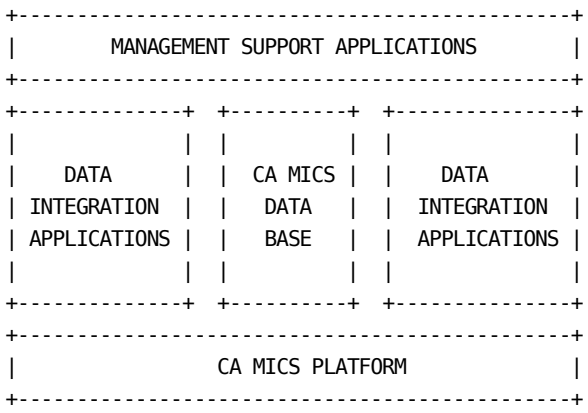


Figure 1-1. The CA MICS System

CA MICS is a comprehensive, flexible application system that applies standard management practices to the management of the IS organization. It uses integrated applications analogous to the integrated financial applications that are indispensable to corporate financial management.

This chapter includes these sections:

1 - Primary Areas of Application
2 - Major Features
3 - Reporting and Inquiry Facilities
4 - Files Overview
5 - Product Prerequisites
6 - Benefits

This section contains the following topics:

1.1 Primary Areas of Application (see page 12)
1.2 Major Features (see page 13)

1.3 Reporting and Inquiry Facilities (see page 15)
1.4 Files Overview (see page 22)
1.5 Product Prerequisites (see page 31)
1.6 Benefits (see page 31)

1.1 Primary Areas of Application

The CA MICS Analyzer Option for VM/CMS provides data and reporting capabilities that aid in the management of complex VM systems in the following areas:

- Performance Management - It provides information that allows you to examine and analyze specific VM and CMS session workloads to pinpoint system and operational problems. The ability to identify and correct these types of problems significantly improves both system performance and the level of service provided to system users.

- Operational Issues - You can track and monitor changes to the operating configuration as well as specific system events to ensure that resources are available and that defined procedures are being followed.

- Accounting and Chargeback - Analyzer files contain data in a form that can be used for basic billing of system resource usage. The data includes CPU time, I/O counts, and counts of records transmitted between systems or to remote work stations. A standard interface to CA MICS Accounting and Chargeback permits accounting from VM, the VM Remote Spooling and Communications Subsystem (RSCS), the VM/VTAM Communications Network Application (VCNA), and CMS batch.

- Capacity Planning - The Analyzer maintains a database of historical system usage and performance data that can be used for growth trend projections at both the system and organizational unit (for example, division or department) levels.

- System Availability and Reliability - Information provided by the product can be used to monitor such events as when and how often Control Program terminations occur.

- Management and Analytical Reporting - Users can group and summarize information directly from the database to create useful management and analytical reporting measures. They can also generate reports that compare actual activity against established objectives for availability, throughput, service and resource consumption. Detailed reports and technical inquiries monitor such things as swap/paging activity, channel utilization, and user consumption characteristics. In addition to these standard reports supplied with the system, users can easily create custom reports through various online

facilities. Additional tools allow resummarization when the summarized information in the database does not address specific reporting needs.

1.2 Major Features

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

- Report facilities
- Use of Monitor, Account, RSCS, and Network data
- General product flexibility
- Interface capabilities

Report Facilities

- 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 organization and are designed to assist in tracking VM service and performance, monitoring the operations configuration, and planning for future resource requirements.
- Standard Analysis Reports are usually run on need basis and provide concise information in the form of graphs, tables, charts, or plots. These reports provide information on various VM and CMS activity such as the number of channels that were concurrently busy over a user-specified period of time. They can be used to identify service level problems, analyze performance and availability, or examine throughput issues.
- Exception Analyzer Reports allow you to define, capture, and report on conditions encountered that are outside of the expected norm.
- Ad Hoc Reporting Facilities allow you to access information in the CA MICS database either interactively or in batch using the CA MICS Information Center Facility (MICF), a panel-oriented productivity tool.

Comprehensive Use of Monitor, Account, RSCS, and Network Data

The CA MICS Analyzer Option for VM/CMS combines information from several sources, which allows it to provide a complete and total picture of system use and activity. In addition, it provides the following functions:

- It consolidates data from multiple systems into an information database and then provides a common access method for reporting on VM activity.

- It interprets encoded values in the input data so that information is immediately usable in a logical form and without further conversion and/or translation.

- It supports cost center information classification based on organizational structure by allowing each organization to set its own account codes to link individual VM session activities with the responsible cost center or organizational unit. This account code structure can be used to report VM activity by user (engineering, payroll, manufacturing, etc.), or by any other relevant category.

- In a VM or z/VM environment, you can quickly track problems or potential problems related to host processor configuration operational status. Operational configuration consolidation done by the system provides information about any changes made during the day as well as resources that are available during IPL.

General Product Flexibility

- Usage guidelines for applying VM management information

- A wide range of standard exits that allow you to tailor the product to your enterprise's specific needs

- Data verification functions that increase data reliability and usability by detecting incorrect monitor input and preventing its inclusion in the database

- A data dictionary that describes the information found in the VMC, VMX, and LNX Information Areas' files and how the information was derived

Interface Capabilities

The CA MICS Analyzer Option for VM/CMS integrates VM activity

and utilization data that is collected by various monitoring functions into the CA MICS database. This process creates the VMC (derived from CP account data), VMX (derived from z/VM and Velocity Software's ESAMON monitor data), and LNX (derived from Velocity Software's zVPS monitor data), information areas, all of which interface with a number of other CA MICS products including:

- CA MICS Accounting and Chargeback - The data provided to Accounting and Chargeback supports accounting for consumed resources such as CPU time, I/O counts, and counts of records transmitted between systems or to remote work stations. This accounting process is accomplished using information from VM, VM Remote Spooling and Communications Subsystem (RSCS), VM/VTAM Communications Network Application (VCNA), Velocity Software's zVPS for Linux, and CMS batch.

- CA MICS Capacity Planner - The CA MICS Analyzer Option for VM/CMS maintains a database of detailed and summarized VM usage and performance data that the CA MICS Capacity Planner can use to make growth trend projections. These projections can be made at all information levels including the system level, a specific virtual machine grouping level, or the organizational structure level.

1.3 Reporting and Inquiry Facilities

The CA MICS Analyzer Option for VM/CMS 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 the MICF and standard SAS language interface to create ad hoc reports.

The standard reports and online inquiries provide concise, comprehensive information to help you manage the VM environment. Each type of report and reporting facility is described on the following pages.

Management Reports

The management reports provide a concise graphical or tabular representation of the site'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 of these timespans (daily, weekly, or monthly).

The management objective reports described in the chart below are delivered with the CA MICS Analyzer Option for VM/CMS.

Report	Function
VM Resource Consumption	Quantifies the total VM resource consumption in seconds per hour averaged across all processor engines in the supporting computing system. The management objective to be tracked is the average CPU time (in seconds) consumed per day.
VM Service	Quantifies the total VM service provided for each computing system. The management objective to be tracked is an absolute measure of the number of VM Q1 drops that occurred within a user-specified number of seconds. This is represented by a chart depicting the average Q1 service interval time.
VM Availability	Shows the availability of a VM system for the identified time period and computing system. The management objective to be tracked is either the number of minutes of uptime per hour (daily and weekly) or the percentage uptime for the total time available in user-defined shifts (zones) in each month.
VM Throughput	Quantifies total VM throughput in terms of the number of scheduled interactions processed on an hourly basis for the identified time period and computing system over a day, week, or several months. The management objective to be tracked is an average of the number of scheduled interactions processed per hour.

Standard Analysis Reports

You can also produce a standard analysis report on VM activity. This report provides a one page summary for a user-specified time period that quantifies system utilization, storage utilization, and scheduler/dispatch activity measures. Derived from detail data, this report is designed to produce a snapshot of key system performance measures.

Standard analysis reports include a series of predefined MICF inquiries. These inquiries are run on an as-needed basis and can serve as templates for designing your own inquiries. The predefined MICF inquiries described in the following chart are delivered with the product.

o VM or z/VM

Inquiry	Function
VM IUCV/VMCF User Summary	Produces a horizontal bar chart, in descending order, of IUCV/ VMCF service users by name, for the top n services during a user-defined reporting period. You can use this report to analyze which services are experiencing heavy usage.
VM Channel Busy Report	Produces a tabular listing of channel activity by channel path identifier for a user-specified time period, quantifying the average percent busy and average percent of total system channel activity for each CHPID. You can use this listing to analyze which CHPIDs are being heavily used.
VM Demand Scan Summary	Produces a tabular listing, in time sequence, of demand scan activity summarized by a user-specified number of minutes and a user-specified time period. Numbers in the Emergency Pass section of the report indicate a possible storage-constrained environment.

VM Device Ranking Report	Produces a tabular ranking of devices, in descending order of utilization, by subchannel identifier or device number (for example, summarized across subchannels) for a user-specified time period. This report provides analysts with the device activity information necessary to monitor which devices are being used and at what rates.
VM IUCV/VMCF Summary Report	Produces, in time sequence, a listing of overall system activity, summarized by a user-defined number of minutes, that covers such measures as system utilization, active users, and paging activity for a user-specified period of time. You can use this report to monitor gross system activity, to perform post-analysis of poor system performance periods, or to compare activity at different times of day, for different days of the week, etc.
VM User Consumption Report	Produces a tabular listing of summarized user resource consumption, for a user-specified time period, ranked by CPU time consumed, average working set size, or virtual I/O rate.
VM User State Summary	Provides a tabular listing, in time sequence, of user states (various wait states, running, etc.) for a user-specified time period and summarized by a user-defined number of minutes. You can use this report to monitor the states in which users are spending their time and to ascertain whether users are often waiting on a particular resource such as the CPU or I/O.
VM Concurrent Channel Busy	Produces, for a user-defined time period, a horizontal bar chart in which each bar represents the number of times during the reporting period that n number of channels were concurrently busy. You can use this chart to analyze whether the I/O workload is being spread over the channel subsystem.

VM Percent Channel Busy	Provides a horizontal bar chart in which each bar represents the percent busy for a given Channel Path ID (CHPID) for a user-specified period of time. The chart displays only those channel paths that experienced some activity. You can use this report to analyze the channel subsystem for performance tuning and capacity control purposes.
VM Cache Usage Report	Produces a tabular listing of cache usage on 3880-23 and 3990-3 control units, in ascending VOLSER order, for a user-specified period of time. You can use this report to analyze cache usage based on volume of activity directed to cache VOLSERs, whether transfers to and from DASD are being avoided by using cache, and the percentage of bytes read from cache, from DASD bypassing cache, and from DASD due to read misses.
VM CP-Owned Volume Summary	Produces a tabular listing of activity on CP-owned VOLSERs for the duration of the user-specified period. You can use this report to analyze page and spool activity on CP-owned volumes, including average and maximum slots used, I/O activity, I/O queuing measures, interference from user I/O, etc.
VM Device Utilization	Produces a tabular listing of activity for selected devices. It allows analysis of individual devices, selectable by address, VOLSER, device type, or device class. It is useful for determining when during the day, activity is being directed to the device, and when device performance is degraded.
VM Cache Usage Detail Report	Produces a tabular report of cache usage on 3880-23 and 3990-3 control units by selected VOLSERs. It allows analysis of cache use by individual device. Separate reports are produced for 3880-23 and 3990-3 controllers.

User Consumption Report (By User)	Provides a tabular report of summarized resource consumption ordered by time. It allows analysis of individual virtual machines. It is useful for tracking changes in user activity that may adversely impact overall system performance.
LPAR Utilization Report	Provides a tabular report of processor usage for logical partitions for a user-defined time period. It is useful for analyzing the distribution of processor resources between dispatch time and management overhead among logical partitions in a PR/SM environment. The report also includes LPAR configurations.

Exception Analyzer 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 VM/CMS is delivered with several exception tests, and you can easily add your own tests. 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.

The Exception Analyzer Reports described in the following chart are delivered with the product.

Report	Function
Exception Management Overview	Provides a concise, high-level summary of 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 specified management area (for example, performance or availability), and the number and severity of each exception type that occurred during an hour. Typically, first-level managers who are responsible for defined management areas use this inquiry.
Severity Level Exception	Provides an hourly summary of exceptions encountered for given severity levels, allowing quick assessment of the types of exceptions in each level. It gives first-level managers, performance analysts, and system programmers an integrated report of problems that may have affected the site in any given hour.
Detail Exception	Provides a detailed list of detected exceptions in order of occurrence. You may produce this report online using various selection criteria, including a date and time range, severity level, management area, etc. You can use this report as background detail for effective analysis of the reported exceptions.

Ad Hoc Reporting Facilities

Because the specific types of information needed to manage a large IS organization change daily, CA MICS provides interactive capabilities that allow fast response to complex requests. These capabilities are provided by:

- The CA MICS Information Center Facility (MICF) - MICF is an ISPF-based 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 dynamically builds an inquiry program that executes in either interactive or batch mode. Then, depending on your specific requirements, MICF either prints the inquiry report, displays the results at your terminal, or catalogs the results for later viewing.

o The Standard SAS Language - Because an advanced analysis language enhances the staff's analytical capabilities and improves the quality of decision-making, CA MICS uses SAS, a powerful fourth generation language. SAS allows easy data manipulation and statistical analysis and provides coding facilities for programmers who are conducting extensive analysis or designing new reports for IS business applications.

The CA MICS Workstation Facility (MWF) provides an online environment that supports SAS in both interactive and batch modes. CA MICS uses the SAS system as a data manager and as an advanced analysis language.

1.4 Files Overview

The CA MICS Analyzer Option for VM/CMS uses a variety of source records to build its information areas:

- The VMC Information Area includes account data from all levels of VM systems and monitor data from VM systems.
- The VMX Information Area includes monitor data from VM/ESA and z/VM.

Both information areas maintain system and user measurement data on service, availability, load, and resource consumption. In addition, the VMC Information Area includes exception data (VMCVXC) from all levels of VM systems.

This section defines the use and content of each CA MICS file in the VMC and VMX information areas.

```
+-----------------------+
|  VMC Information Area  |
+-----------------------+
```

o VM User Resource Accounting File (VMCCMU)

The VMCCMU file quantifies the resources used by virtual machine. The information in this file is derived from VM Account record types 01 and C1.

- VM Device Accounting File (VMCDAC)

  The VMCDAC file identifies the allocation of dedicated devices and temporary disk virtual machine (user). The file is created from CP account record types 02, 03, C2, and C3.

- VM RSCS Accounting File (VMCVRA)

  The VMCVRA file quantifies RSCS use attributable to a virtual machine. The information in this file is derived from CP account record type C0 as created by RSCS.

- VM Network Terminal Usage File (VCNA) (VMCNTU)

  The VMCNTU file quantifies VCNA activity attributable to virtual machines. The information in this file is derived from CP account record type 07.

- VM Network Data Transmission File (VMCNDT)

  The VMCNDT file quantifies data moved over virtual devices such as IUCV, VCTC, and Guest LAN's attributable to virtual machines. The information in this file is derived from CP account record type 0C.

- VM Exception Activity File (VMCVXC)

  The VMCVXC file quantifies exception activity within the VM system. The information in this file is derived from CP account record types 04, 05, 06, 08, and VM and Monitor domain 1, records 2, 3, 10, 11, and 12.

```
+------------------------+
|  VMX Information Area  |
+------------------------+
```

- VM System Configuration File (VMXSCF)

  The VMXSCF file identifies system hardware and software configurations for the VM system(s) represented in the source data. The information in this file is derived from VM Monitor domain 1, records 1, 4, 5, 7, 9 and 17.

- VM Processor Configuration File (VMXPCF)

  The VMXPCF file identifies CPU-specific configuration information for the VM system(s) represented in the source data. This file is created from the VM Monitor domain 1, record 5 and from domain 5, records 1, 2, 4, 5, 6 and 7.

- VM System Global Data File (VMXSGP)

  The VMXSGP file quantifies VM system-level utilization, storage, virtual machine states, and communication activity. The information in this file is derived from VM Monitor domain 0, records 1-19; domain 1, records 5, 7 and 17; domain 3, records 1-3, 9, 10, and 14; domain 4, records 2 and 3; and domain 5, record 3.

- VM System Processor Data File (VMXSPP)

  The VMXSPP file quantifies VM and CPU-level utilization, storage, virtual machine states, and communication activity. This file is created from VM Monitor domain 0, records 1, 2, 4, 5, 11 and 13; domain 3, record 2; and domain 5, record 3.

- VM Device Configuration File (VMXDCF)

  The VMXDCF file identifies I/O configuration information for devices and control units. The information in this file is derived from VM Monitor domain 1, record 6; and domain 6, records 1, 2, and 5-9.

- VM Device Activity File (VMXXDV)

  The VMXXDV file quantifies I/O activity and utilization by device. This file is created from VM Monitor domain 1, record 6; and domain 6, record 3.

- VM Device SEEKs Data File (VMXXDS)

  The VMXXDS file quantifies, by device, DASD arm movement and frequency of access. The information in this file is derived from VM Monitor domain 7, record 1.

- VM CPVOL Configuration Data File (VMXACF)

  The VMXACF file identifies VM System volume configuration. This file is created from VM Monitor domain 1, record 8; and domain 3, record 7.

- VM CP Owned Volume Data File (VMXCPV)

  The VMXCPV file quantifies the activity on the VM system volumes. The information in this file is derived from VM Monitor domain 1, record 8; domain 3, records 4, 7, and 11.

- VM User Data Configuration File (VMXVXF)

  The VMXVXF file identifies virtual machines (users) logged on and the configuration of the virtual machine. The information in this file is derived from VM Monitor domain 1, record 15; and domain 4, records 1, 2, and 5-7.

- VM User Activity Sample Data File (VMXVXU)

  The VMXVXU file quantifies resource consumption of each virtual machine based on sampling. This file is mainly created from VM Monitor domain 4, records 2 and 3; and domain 3, record 10. VM Monitor domain 1, record 15, and domain 4, record 1 contribute to the user configuration information.

- VM User Activity Event Data File (VMX_VU)

  The VMX_VU file quantifies resource consumption of each virtual machine at interaction end. This file is mainly created from VM Monitor domain 4, record 9. VM Monitor domain 1, record 15, and domain 4, record 1 contribute to the user configuration information.

- VM User Transaction Sample Data File (VMXVXT)

  The VMXVXT file quantifies, based on sampling, interactions among users, and between users and systems services using IUCV and VMCF. The information in this file is mainly derived from VM Monitor domain 4, record 4. VM Monitor domain 1, record 15, and domain 4, record 1 contribute to the user configuration information.

- VM User Transaction Event Data File (VMX_VT)

  The VMX_VT file quantifies, based on recording individual events, interactions among users, and between users and systems services using IUCV and VMCF. This file is mainly created from VM Monitor domain 4, records 8 and 10. VM Monitor domain 1, record 15, and domain 4, record 1 contribute to the user configuration information.

- VM Channel Path Contention File (VMXCHC)

  The VMXCHC file quantifies utilization, busy conditions, and queue lengths of the physical channels. The information in this file is derived from VM Monitor domain 0, record 9.

- VM Block Paging Data File (VMXBPG)

  The VMXBPG file provides VM block paging information. This file is created from VM Monitor domain 3, record 8.

- VM Shared Storage Configuration File (VMXXSF)

  The VMXXSF file identifies VM shared system configurations. The information in this file is derived from VM Monitor domain 3, records 3, 5 and 6.

- VM Shared Storage Management File (VMXXSS)

  The VMXXSS file quantifies VM shared system activity. The information in this file is derived from VM Monitor domain 3, record 3.

- VM System Timer Pop File (VMXSTP)

  The VMXSTP file identifies system load as reported by the VM scheduler at the scheduler interval. This file is created from VM Monitor domain 2, record 8.

- VM DASD Cache Activity File (VMXCDI)

  The VMXCDI file quantifies DASD cache activity for 3880-X3 and 3990-3 subsystems. The information in this file is mainly derived from VM Monitor domain 6, record 4. VM Monitor domain 1, record 6 contributes to the device configuration information of the file.

- VM Paging Cache Activity File (VMXCDP)

  The VMXCDP file quantifies DASD cache activity for 3880-X1 controllers. This file is mainly created from VM Monitor domain 6, record 4. VM Monitor domain 1, record 6 contributes to the device configuration information of the file.

- VM Logical Partition Activity File (VMXLPR)

  The VMXLPR file quantifies activity for each logical partition in the central electronic complex when VM is running in a logical partition. This file is created from VM Monitor domain 0, records 15-17.

- VM LPAR Logical CPU Activity File (VMXLCP)

  The VMXLCP file quantifies activity for each logical CPU in the central electronic complex when VM is running in a logical partition. This file is created from VM Monitor domain 0, records 15-17.

o VM Scheduler Parameters File (VMXSCX)

The VMXSCX file identifies the settings of scheduler tuning commands such as SET SRM IAB, SET SRM DSPSLICE, SET SRM LDUBUF, and SET SRM STORBUF, etc., for system resource management. This file is created from VM Monitor domain 1, record 16, and domain 2, record 7.

Figures 1-2 through 1-4 illustrate the structures of the VMC and VMX Information Areas.

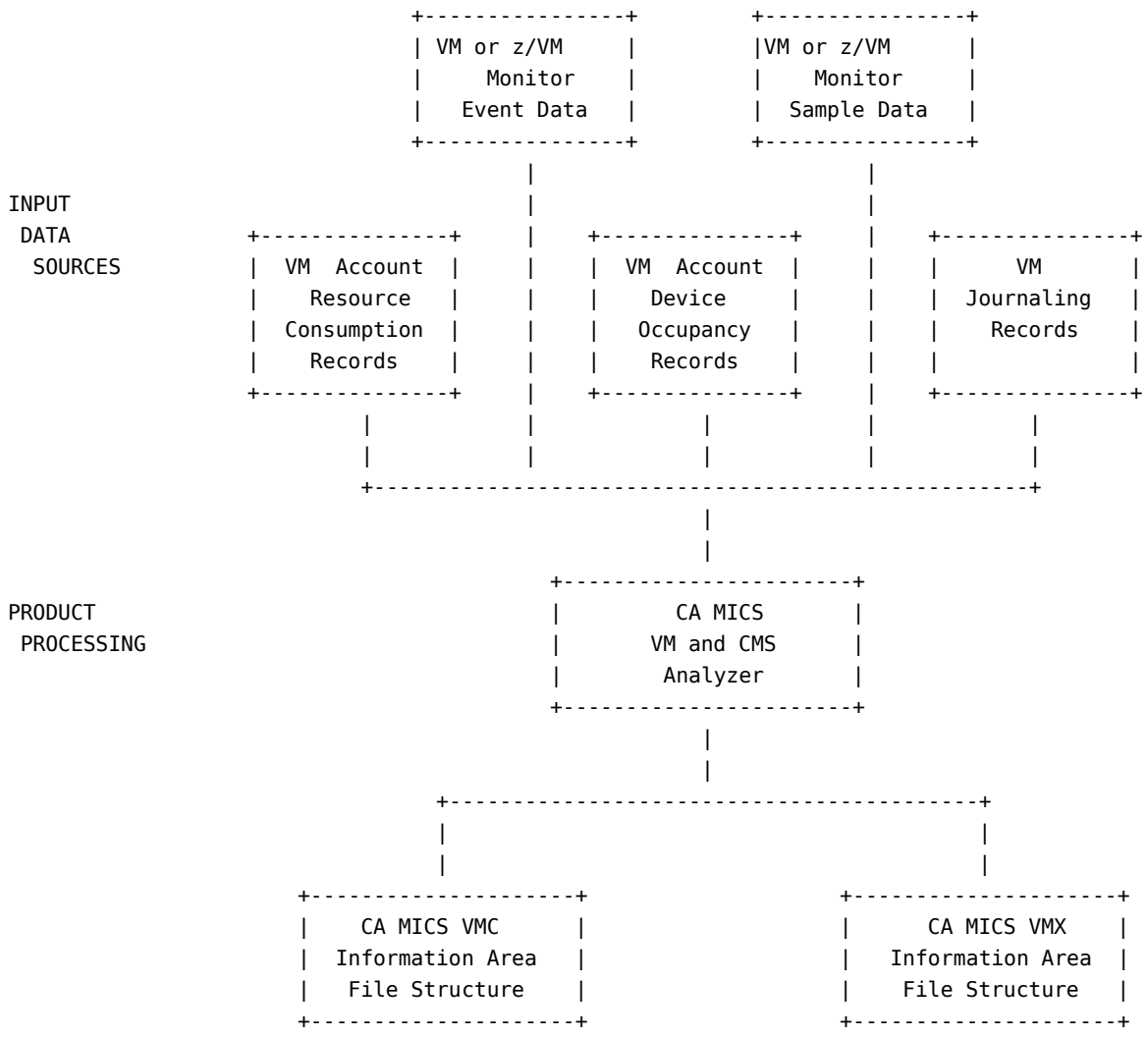


Figure 1-2. File Structure (Part 1 of 3)

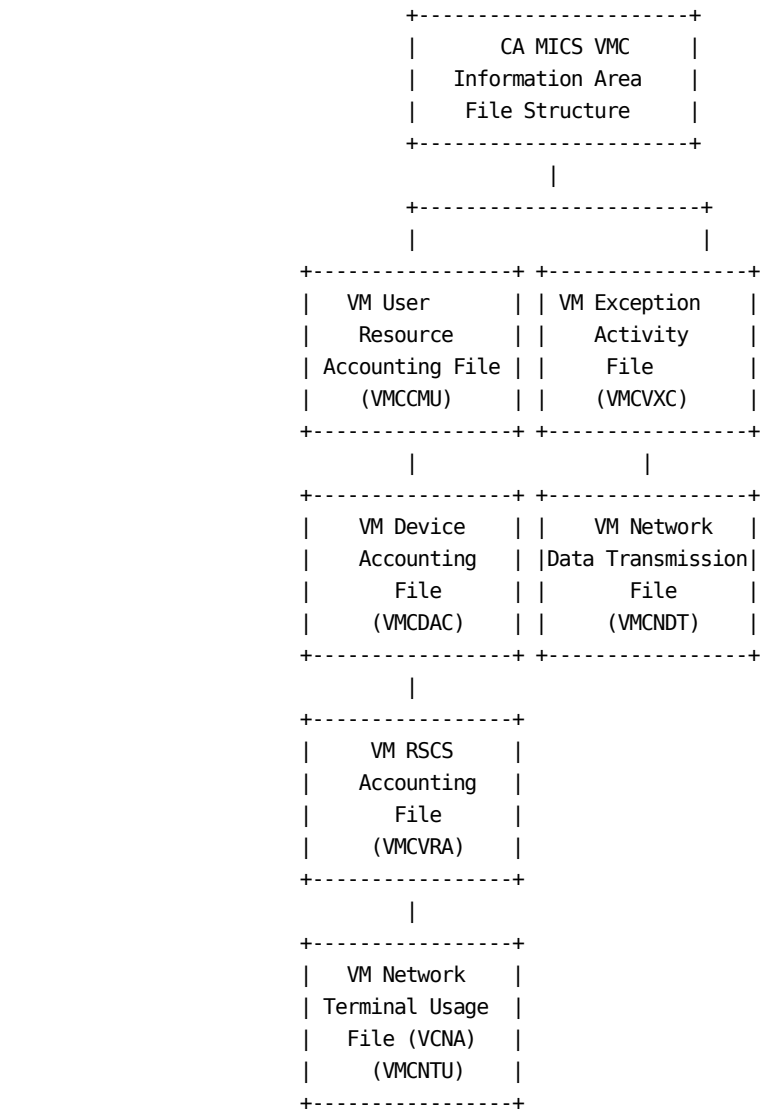


Figure 1-3. File Structure (Part 2 of 3)

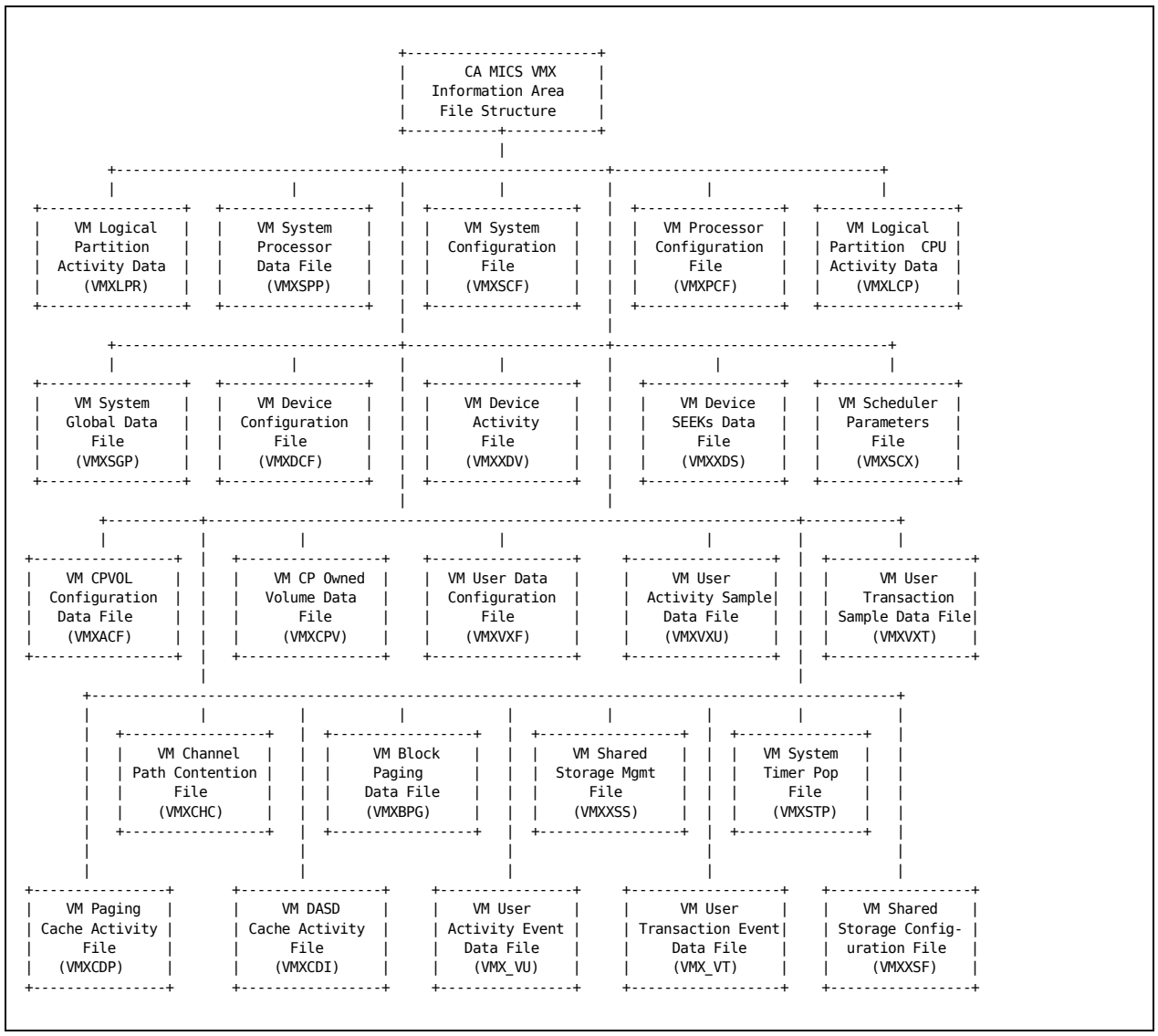


Figure 1-4. File Structure (Part 3 of 3)

1.5 Product Prerequisites

The CA MICS Analyzer Option for VM/CMS runs in a CA MICS environment operating under z/OS.

You must have one or more of the following monitors operational at any release level:

- IBM's VM/Enterprise Systems Architecture ESA Feature
- IBM's z/VM Systems Architecture

You must also have the CA VM Data Transfer Program or an equivalent facility to move VM data from a VM environment to the z/OS CA MICS environment. For VM monitor data, you must use the CA VM Data Transfer Program to decompose and move the data from VM to z/OS for CA MICS processing.

1.6 Benefits

The CA MICS Analyzer Option for VM/CMS contributes to the overall benefits that CA MICS provides. Those benefits fall into the following broad categories.

- Improves day-to-day I/S management
- Reduces risk to the enterprise
- Controls and/or helps reduce costs
- Improves return on the I/S investment
- Increases productivity
- Improves planning

Each of these benefits is described in detail below.

Improves Day-to-day I/S Management

- Allows managers to track usage and plan corrective actions to ensure smooth operations

- Allows you to identify problems more quickly and accurately, and to assess or quantify a problem's impact in terms of the root cause(s) and resolution alternatives

- Allows you to measure and track the service levels that are being delivered for VM and CMS workloads.

Reduces Risk to this Enterprise

- Allows I/S to define exceptional conditions and assess their impact

- Provides auditing and control capabilities for VM and CMS activity

- Allows you to archive and back-up system data automatically so that you can retrieve historical information easily for recovery and evaluation purposes

Controls and/or Helps Reduce Costs

- Ensures that individual users are charged only for the resources they use

- Allows you to identify unused resources and substantially reduce their associated costs

- Using the CA MICS Analyzer Option for VM/CMS in place of other VM reporting products eliminates the costs and problems associated with maintaining separate products.

Improves Return on the I/S Investment

- Allows managers to gauge the quality of service being provided

- Provides a basis for chargeback of operating system and hardware resource usage either through CA MICS Accounting and Chargeback or a user-developed costing process

- Helps you to understand, report on, control, and reduce VM and CMS expenses

Increases Productivity

- Allows less experienced users to interpret report results and to use database information
- Provides the opportunity to learn about the characteristics and uses of VM data via the comprehensive data dictionary and report documentation
- Reduces the task of coordinating information from multiple sources so that you can report or analyze information at the enterprise rather than the system level
- Reduces the large quantities of VM data to a manageable and easily used format.

Improves Planning

- Allows managers to anticipate problems before they affect the system
- Provides the raw data and a common frame of reference for other analyses such as capacity and hardware planning, workload balancing, and performance management
- Allows you to use summarized historical data for trend analysis

Chapter 2: USAGE GUIDELINES

The CA MICS Analyzer Option for VM/CMS allows consistent analysis of VM and CMS activity across all operating systems. The following sections describe special issues associated with using the product and the interfaces with other CA MICS products. The first section, Operating System Concepts, is an introduction to the features of the VM operating system.

This section contains the following topics:

2.1 Operating System Concepts (see page 36)
2.2 Product Use (see page 40)

2.1 Operating System Concepts

The VM operating system provides a simulation of hardware facilities for operating systems and applications to use. Operating systems such as MVS, VSE, and VM that run natively can run in a virtual machine under VM. VM provides virtual machine, time sharing, networking, and diagnostic support for IBM OS390 and z/OS systems.

Since its inception in the early 1970s, the VM operating system has rapidly increased its popularity and services among the IBM user community. Running on machines from the smallest IBM 9370 to the largest IBM ES/9000, the VM interface addresses the diverse needs of its user groups.

VM's interactive CMS subsystem has also served to increase the use of the VM operating system. CMS's simplicity of use and numerous products that operate under it -- SAS, PROFS, FORTRAN, APL, and CICS, to name a few -- have resulted in the expansion of the VM user community.

THE ENVIRONMENT PROVIDED BY VM

VM has two principal components: CP and CMS. CP (Control Program) runs in a real machine and can simulate concurrent operation of multiple virtual machines. CP is primarily a real machine resource manager. CMS (Conversational Monitor System) provides the online interface to the system. CMS operates only in a virtual machine, and together with CP, provides a time-sharing system suitable for program development, problem solving, and general time-sharing work. Some of other VM components are:

```
RSCS - for inter-network data transfer
PVM  - for networking VM terminals
VSCS - for networking VM terminals using VTAM
```

VM also provides services that handles communication between multiple systems using TSAF (Transparent System Access Facility) and APPC/VM. APPC/VM VTAM (AVS) lets application programs using APPC/VM or CPI Communications communicate with programs anywhere in a SNA network.

VM provides an environment in which several copies of the same operating system or of different operating systems can run concurrently. These systems all have a unique structure internally and perform different functions (or similar functions in a different manner). The environment provided

by VM comprises the following categories:

HARDWARE INTERFACE

CPU - The operating system can use privileged instructions to control the CPU.

storage - The operating system can manage storage allocated to it in any way, including setting storage keys and resetting reference bits.

I/O - The operating system can construct channel programs and initiate I/O instructions to read and write on any recording medium.

SERVICES

VM controls the real environment (CPUs, storage, and I/O devices). To do this, it runs in supervisor state. This allows VM to issue any instruction that is valid for the model of processor on which it is running. To maintain the integrity of its own code, to protect the code in one virtual machine from damage by another, and to prevent unauthorized modification of the real environment by virtual machines, VM runs virtual machines in real problem state. This means that when the virtual machine issues an instruction that is valid only in supervisor state, the hardware intercepts the instruction and passes control to VM. VM then provides any management services required for the instruction to execute properly in the real environment. This is necessary because the operating system in a virtual machine behaves as though it owns the real environment and issues supervisor or privileged instructions.

Paging - VM provides real storage management for virtual machines that do not have real storage dedicated to them as a part of system generation. VM implements demand and block paging and manages data caching in real storage and expanded storage. VM allows pages in real storage to be nonpageable (locked), eliminating the need to page the locked frames. VM also allows part of real storage to be reserved for use by a virtual machine so that it is shielded from storage contention by other virtual machines.

I/O - VM initiates all I/O operations in the real

machine. VM provides real device management services through the translation of virtual device addresses to real device addresses, and of storage references in channel programs into real addresses. VM allows DASD to be partitioned for use by several virtual machines, and provides for relocation of the cylinder or block reference in the virtual channel program to the real cylinder or block on the device. VM also provides interfaces for synchronous and asynchronous I/O that reduce the work required within a virtual machine to perform I/O. VM will also simulate channel-to-channel adapters (CTCA) so that two virtual machines can communicate with each other. No real CTCA is required for this.

Spool - VM can simulate unit record devices for a virtual machine, providing read access to the VM spool via a virtual reader device and providing write access to the spool via a virtual punch or print device. These virtual unit record devices do not have to have a real counterpart on the system. A facility is also provided that allows the recording of virtual console traffic to a spool file for subsequent reference.

Communications - VM provides the ability for virtual machines to set up communications between with each other. This is accomplished by means of VMCF (virtual machine communication facility) and IUCV (interuser communications vehicle). IUCV is also used by VM to provide services such as access to VTAM or a security package, and to allow a virtual machine to trap messages that would otherwise be displayed on its virtual console.

Assists:

VMA - VMA (virtual machine assist) is a microcode enhancement for VM that reduces or eliminates the need for CP to provide instruction simulation for virtual machines.

PMA - PMA (preferred machine assist) assist allows a MVS guest single preferred to run in real supervisor mode, eliminating the requirement for VM to handle I/O initiated by the guest to any device on a channel that has been set aside for that purpose. The only time VM intercepts the guest is when I/O is done to some device managed by CP or when some

other CP service is required.

SIE - SIE (start interpretive execution) is used by CP to allow a virtual machine access to the real hardware facilities. SIE eliminates much of the CP processing on simulating most privilege instructions and certain interrupts for a virtual machine by performing these functions by way of hardware.

SIE - SIE Assist provides additional performance Assist enhancements for a virtual machine that has exclusive control of a real device by eliminating the need for CP to intercept the I/O for integrity checking.

MHPGS - MHPGS (multiple high performance guest support) provides VM with the ability to have more than one guest that has virtual storage mapped directly to virtual storage. The guest which is given control of real page zero is called the V=R (virtual=real) guest and the other guests that have this virtual-to-real storage mapping are called V=F (virtual=fixed) guests. VM/ESA preferred machine recovery is available only for the V=R guest.

IMPLICATIONS FOR MEASUREMENT

VM provides measurement data in two basic forms: account data and monitor data. The account data is formatted in 80-byte card images and provides information suitable for accounting and chargeback. The monitor data is formatted as variable length records and provides information suitable for performance analysis and capacity management and planning.

VM views the virtual machine at the simulated hardware level. It does not know the unique internal structures, code paths, or algorithms used by the software running in the virtual machine. Therefore, VM can provide measurements of system use only to the level of resource use that it understands: the virtual machine. A transaction within the virtual machine is not recognized by VM as a transaction in the same way that the operating system within the virtual machine recognizes it. This means that VM cannot measure this activity as a transaction in a manner that provides meaning as a measurement of virtual transaction resource consumption or elapsed time.

VM measures transactions from internal perspective: a transaction is a piece of work from the time that a virtual machine is dispatched until the time it is dropped from the dispatch queue. VM provides measurements of resource consumption by the virtual machine, and measurements of resources required to manage the virtual machine within VM. VM also provides overall system measurements of real device activity, storage use and management, scheduler activity, and system performance.

2.2 Product Use

The CA MICS Analyzer Option for VM/CMS provides a large amount of information about your VM system. This information can be used for performance management, capacity planning, and installation accounting.

The following sections describe some of the data analysis that you can perform:

1 - Data Analysis
2 - Performance Analysis
3 - Analyzer Reporting Options
4 - Accounting and Chargeback Interface
5 - Capacity Planner Interface

2.2.1 Data Analysis

The data available in the CA MICS Analyzer Option for VM/CMS can be used for performance analysis, capacity management, and accounting and chargeback.

```
PERFORMANCE ANALYSIS

For VM, performance data is provided by the following
files:

  VMXSGP - System Performance data, providing scheduling
           information, page rates, simultaneous CHPID busy,
           summarized IUCV data, wait states, and CPU busy

  VMXSPP - System Performance data at the CPU Level

  VMXXDV - Device Activity, providing device utilization

  VMXCPV - CP Owned Volume Activity, providing page and spool
           activity, and I/O queuing data

  VMXVXU - User Activity, providing user resource consumption

  VMXCDI - DASD Cache Activity, providing effectiveness of
           use measurements for cached DASD

  VMXCDP - Paging Cache Activity, providing effectiveness of
           use measurements for cached DASD

  LNXLXP - Interval view inside of Linux guests, providing
           resource consumption, page fault activity, and
           memory size for all processes executed.
```

CAPACITY MANAGEMENT

Capacity data is available in the VMXSGP and VMXXDV files. The information available includes the following:

- CPU utilization
- Storage utilization
- Logged and active users
- Page and spool slots utilization
- Transaction counts
- I/O rate
- VMDBK waits
- Expanded storage utilization
- Channel path busy
- Device utilization

You can run your own capacity studies using the CA MICS Analyzer Option for VM/CMS. You can also use the CA MICS Capacity Planner to analyze VM data. For more information, see Section 2.2.5.

```
CONFIGURATION ANALYSIS

System configuration changes can be traced through the
CA MICS Analyzer Option for VM/CMS files.

The following configuration files are available:

  VMXSCF - System Configuration File

  VMXPCF - Processor Configuration File

  VMXDCF - I/O Subsystem Configuration File

  VMXACF - CP Owned Volume Configuration File

  VMXXSF - Shared System Configuration File

  VMXVXF - Logged On Users Configuration File

  VMXSCX - Scheduler Parameters File


ACCOUNTING AND CHARGEBACK

Support for accounting and chargeback is provided in five
files:

  VMCCMU - Virtual Machine Resource Use File

  VMCDAC - Attached Device Use File

  VMCVRA - RSCS Use File

  VMCNTU - VCNA Network Use File

  VMCNDT - Network Data Transmission File

These files provide information about the CPU, spool, and I/O
use of virtual machines, use of attached devices such as
temporary disk and tape, traffic through RSCS, including
destination and file type, and data traffic through VCNA or
VSCS for virtual machines accessing VM using VTAM.

In addition, CA MICS Accounting and Chargeback provides a
complete accounting system for your VM operation.  For
further information on the interface for accounting, see
section 2.2.4.
```

EXCEPTION TRACKING

You can also track exceptional conditions that are reported in the account or monitor data. The VM Exception File (VMCVXC) provides this information for all levels of VM systems. It includes password violations from the journal data and monitor suspensions.

2.2.2 Performance Analysis

The VM control program views the real machine as consisting of four subsystems: CPU, storage, paging, and I/O. The performance of the total system and of individual virtual machines is determined by the utilization of these subsystems. Understanding significant measurements of these subsystems and the relationships among them is the first step in analyzing VM performance.

VM controls the subsystems of the real machine, distributing them to the workload as required. The workload in a VM system consists of the applications and operation systems of the virtual machines that are logged on to real machines at any given moment. To evaluate VM performance, it is first important to characterize the workload of the virtual machines and to understand the basic elements of CPU utilization.

WORKLOAD DISTRIBUTION

A workload profile can be obtained by looking at a count of active users by hour (Figure 2-1). An active user is a virtual machine that has consumed some CPU time during the measurement interval. (The default interval is 60 seconds, and the intervals are summarized to show the count by hour.) The active user count shows the workload that existed during the measurement interval.

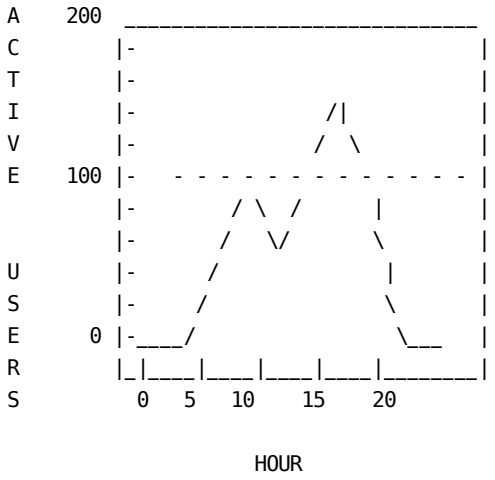


Figure 2-1. Active User Count by Hour

Typically, there will be periods of very few active users (where the system resource consumption is very low). Alternatively, there may be periods of high demand, with many active users. As the workload increases (periods of high demand begin to rise), the count of active users can be plotted against other variables to determine where bottlenecks occur, as we will see in the following sections.

```
CPU UTILIZATION

The utilization of the CPU is quantified by how busy the CPU
was during the measurement interval.  In a VM system, the CPU
utilization measurement known as "total CPU" is made up of
two elements:  virtual time and overhead time. Virtual time
is that portion of the total time that is directly consumed
by the virtual machines(s).  Overhead time consists of the
time that is not attributable to any virtual machine (such as
I/O simulation, interrupt processing, and time consumed on
behalf of a virtual machine).  The portion of the interval
that the CPU was not performing any work is known as a "wait"
state.
```

Figure 2-2 shows an example of total and virtual CPU utilization plotted against active users. The figure shows the load capacity of the system. Saturation occurs at the point at which the workload is increasing. This occurs because the overhead that is needed to support the workload consumes those CPU cycles which otherwise would have been consumed by the virtual machine(s).

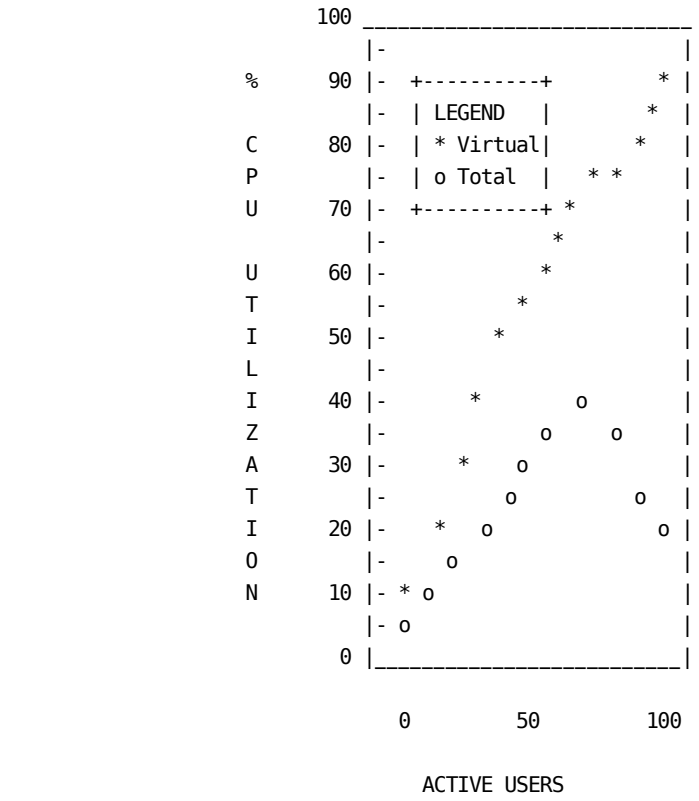


Figure 2-2. Total and Virtual CPU Utilization vs. Active Users

Once you have determined your VM system's workload and its load capacity, you can go on to pinpoint specific subsystems that are bottlenecked. You can do this by producing plots of contention indicators for the four subsystems (CPU, storage, paging, and I/O) against workload. This will show not only which resource bottlenecks first, but can aid you in determining where you can obtain the most improvement for the cost.

This section includes sample plots for two of the subsystems. Plots for the other subsystems can be generated using the guidelines discussed here. Note that the detection of subsystem bottlenecks should be based both on factual analysis (as we will show in the following sections) and on familiarity with normally acceptable conditions.

CPU CONTENTION

Contention for a system resource such as the CPU can be quantified as the percentage of active users who are waiting for the processor plotted against total CPU utilization (Figure 2-3). This figure indicates that throughout most of the range of CPU utilization, the percentage of active users waiting for the CPU was relatively low (less than 5 percent) and would generally be considered acceptable.

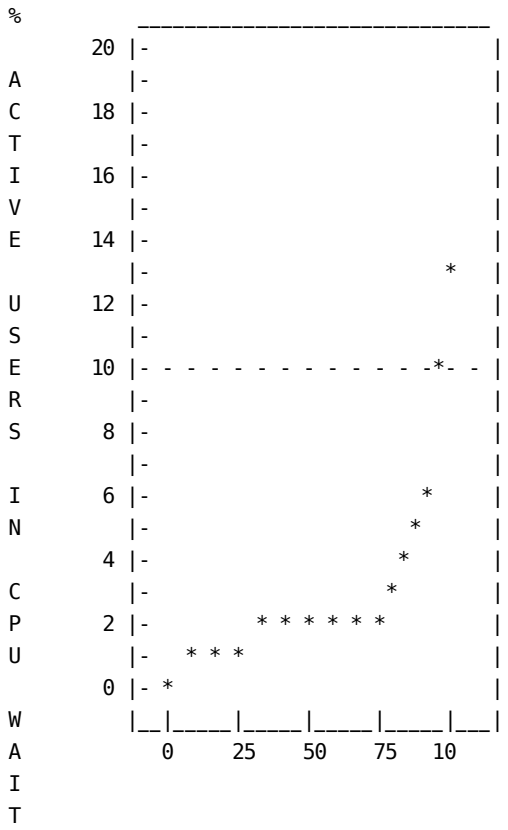


Figure 2-3. Percent Active Users Waiting for CPU vs. Total CPU Utilization

At the high end of utilization, as greater proportions of active users are waiting for this resource, it is clearly evident that the CPU has become a bottleneck. As the load increases in a CPU-bound environment, most users will be

found waiting for the CPU as opposed to the other resources.

VM favors the interactive users and will continue to deliver acceptable response. Therefore, when the CPU cannot satisfy the demand, the performance degradation will be felt most by non-interactive users.

STORAGE CONTENTION

At low system utilization levels, there is usually an adequate quantity of main storage to satisfy the workload. As the workload increases, however, storage contention occurs.

Sufficient main storage is essential in a VM system so the good and consistent interactive response can be delivered. If the available main storage is insufficient to satisfy the demand of the workload, due to paging, virtual machines will incur a wait before being dispatched.

Virtual machines are classified as eligible when they are waiting. The percentage of active users in storage wait is the measurement that quantifies the contention for main storage. Plotting this measurement against the total number of active users can aid in determining whether a storage bottleneck exists.

PAGING CONTENTION

The paging resource in VM is treated as an extension of storage. When sufficient storage to handle the workload exists, paging is typically inactive. However, as the workload increases, paging activity also increases. The speed of the primary paging devices, the number of paging slots available, and the "robustness" of the paging resource can have a tremendous impact on virtual machine performance.

A paging bottleneck is likely to be revealed by the number of active users who are in page wait. This can be observed by plotting the page rate against the users who are waiting for pages. Figures 2-4 illustrates this. Again, the saturation point occurs where the workload (number of active users) is increasing and the paging rate is level or decreasing.

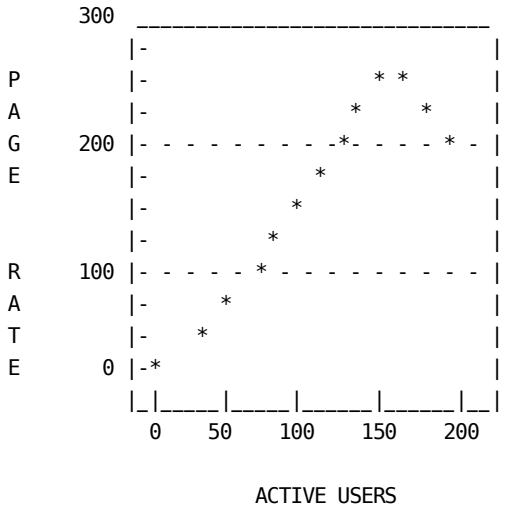


Figure 2-4. Paging Rate vs. Active Users Waiting for Pages

In addition, systems with mixed paging devices should be examined to determine when workload levels cause the paging to overflow to slower devices. When the paging overflows from fixed-head devices to moving-head devices, there may be interference with the normal disk I/O speed that users experience. A paging bottleneck in a VM system is typically accompanied by inconsistent and painfully slow response times.

I/O CONTENTION

The I/O resource processes virtual-machine-initiated I/O. As a result of this, the I/O rate is sometimes used as a measurement of virtual machine throughput. As the workload increases, I/O bottlenecks can occur due to channel, control unit, and device saturation. In addition, configuration imbalances also can cause I/O contention.

Plotting the I/O rate against workload can show the saturation point at which the I/O rate decreases as the workload increases. Plotting the percentage of active users in I/O wait against the active user count also can aid in detecting I/O contention.

RELATIONSHIPS AMONG RESOURCES

The supply of, or demand for, one subsystem can affect the utilization of others. Removing the bottleneck for one system by eliminating contention or decreasing the workload can lead to an unanticipated bottleneck somewhere else. Latent demand that is suddenly manifested can cause subsystem resource consumption at unexpected rates (i.e., the removal of a storage bottleneck can lead to a significant increase in CPU utilization).

Response Indicator

Collecting and analyzing true response time measurements of a VM system requires a lot of overhead. The time virtual machines spends in queue and in the eligible list waiting to get into queue provides a good measurement of the responsiveness of the system as a whole.

The measurements discussed here are normal measurements of a VM system. However, it must be understood that these measurements are based on internal scheduler variables and algorithms, which means that a transaction as viewed by the scheduler will probably not be the same as a transaction as viewed by an individual sitting at a terminal. The scheduler views a transaction as the work required for a virtual machine to voluntarily drop from queue (complete the task at hand). The end user views a transaction as the time from the pressing of the ENTER key to the completed response at the terminal. The scheduler may be required to perform several of its transactions to complete one end user transaction.

It is important to note that scheduler algorithms change from release to release of VM, which is how VM performance has been improved. There have been major revisions in each release which means that the measured system responsiveness may change from release to release of VM, and should be re-measured as each new release is installed.

Finally, if your VM system uses VTAM or PVM to communicate with terminals (local or remote), additional software is in the response path between the end user and VM. This elongates the actual response time in a manner that cannot be measured by the scheduler.

The Next Step

Performing basic performance analysis of a VM system can be summarized as finding the answers to two questions:

1. Is the system performing well based on past history?

2. Which of the four subsystems -- CPU, storage, paging, or I/O -- seems to be bottlenecked?

Once you have identified a bottleneck, the next course of action is to perform a more detailed analysis of the suspect resource.

2.2.3 Analyzer Reporting Options

The CA MICS Analyzer Option for VM/CMS provides several reporting options for data center management. These reports include system, application, user group, and user reports that quantify the utilization and performance of VM service, key users, and user-defined applications. They provide summarized information reported over several months. In addition, they permit management to evaluate and track VM performance on a month-to-month basis. See Chapter 3, Reports, for a complete description of the reports that are available with this product.

2.2.4 Accounting and Chargeback Interface

In today's computer environment, the ability to account for VM/CMS usage has become an industry-wide concern. The primary reason for this is a change in VM/CMS usage.

Traditionally, sites used VM/CMS as a test vehicle in which the system cost could be absorbed as an overhead item. Today, however, VM/CMS is being used as the basis for large corporate information centers. CMS usage is widespread in these centers and must be chargeable to departmental budgets. This has resulted in major new accounting responsibilities.

The CA MICS Analyzer Option for VM/CMS addresses the accounting of VM/CMS resource usage by interfacing with CA MICS Accounting and Chargeback. This combination provides the ability to utilize charging methodologies that are unique to the VM/CMS environment. For example, it allows the recognition of foreground mode and batch machine usage.

IDENTIFICATION OF SERVICE VIRTUAL MACHINES

You can detect a virtual machine that is providing some sort of general system service. Virtual machines such as RSCS, SMART, DIRMAINT, and VTAM fall into this category. This allows a site to group these virtual machines for identification for system resource use to be prorated across some or all system users.

DIFFERENTIAL CHARGING FOR BATCH USAGE

CMS batch is a special case of disconnected use. Some sites have implemented a batch structure, where one or more virtual machines are set up exclusively to perform large compilations and other lengthy tasks. These machines are usually prioritized so that they execute in a background mode. A batch machine generates C1, C2, and C3 accounting records. If a site wants to discount those resources consumed by the batch machine, it can simply use the CMUXMODE to detect batch machine usage.

```
VM CHARGING ELEMENTS

When CA MICS Accounting and Chargeback produces invoices for
data center users, those customers who have used VM system
resources are charged according to site-defined algorithms.
The algorithms are based on VM data elements from the CA MICS
Analyzer Option for VM/CMS files of the CA MICS database.  To
simplify the charging process, the following default charging
elements are distributed with Accounting and Chargeback:

     VM ACCOUNTING

       VM Resource Accounting

           VM CPU Charges
             VM Instructions Executed
             VM Total CPU Time
             VM Virtual CPU Time
             VM Overhead CPU Time
             VM Session Time

           VM Device Charges
             VM Tape Allocations
             VM DASD Allocations
             VM Temporary Disk Allocations
             VM Comm Allocations
             VM Unit Record Allocations
             VM Total Allocations
             VM Non-spooled SIOs
             VM Tape Occupancy Time
             VM DASD Occupancy Time
             VM Temporary Disk Occupancy Time
             VM Comm Occupancy Time
             VM Unit Record Occupancy Time
             VM DASD (Perm and TDISK) Space
             VM Device Connect Time

           VM Input/Output Charges
             VM Cards Read
             VM Cards Punched
             VM Lines Printed

       VM VTAM Terminal Usage
           VM VTAM Input Request Units
           VM VTAM Input Characters
           VM VTAM Output Request Units
           VM VTAM Output Characters
           VM VTAM Console Output Lines
           VM VTAM Copy Requests
```

```
VM RSCS Resources
    VM RSCS Records in File

VM Linux Resources
    Linux Virtual CPU Time
    Linux Total CPU Time
```

2.2.5 Capacity Planner Interface

These elements of the CA MICS Analyzer Option for VM/CMS are useful for on-demand analyses using the CA MICS Capacity Planner:

```
CPUMODEL  CPU Model Identification
DURATION  Duration of the Interval Reported
SGPCPUTM  Total CPU Time Charged to Users
SGPPRBTM  Total CPU Time Spent in Emulation Mode
SGPSYSTM  Total CPU Time Charged to the System
SGPSUPTM  Total CP CPU Time Charged to Users
SGPTOTTM  Total CPU Time Consumed
SGPPCFSU  CP Free Storage Utilization
SGPPCPIX  Percent Page-Ins From Expanded Storage
SGPPCPOX  Percent Page-Outs to Expanded Storage
SGPPCPRB  Percent CPU Virtual Time
SGPPCSUP  Percent CPU Supervisor Time
SGPPCSYS  Percent CPU System Time
SGPPCTOT  Percent CPU Total Time
SGPPCWAT  Percent CPU Wait Time
SGPPCXTU  Percent CP Expanded Storage in Use
SGPXPGIN  Number of Fast Path Pagins Performed
SGPPGFCT  Count of Page Faults on this CPU
SGPPGRIO  Count of Read Operations for Paging
SGPTVRAT  Total to Virtual Ratio (System)
SGPTVRVM  Total to Virtual Ratio (Users)
SGPPGRTA  Page Rate to Auxiliary Storage
SGPPGRTE  Page Rate
SGPPGRTX  Page Rate to Expanded Storage
SGPSTGUT  Storage Utilization
SGPXTCCP  Expanded Storage Configured to CP
SGPXTSIZ  Expanded Storage Installed
SGPCDLDG  Count of Loading Users in Dispatch Lst
SGPCTXAV  Expanded Stor. Available but Not Used
SGPC1DLD  Count of Loading Users in Q1, Q2, & Q3
SGPC1DSP  Q1, Q2, and Q3 Users in Dispatch List
```

```
SGPC1ELD  Count of Loading Users in E1, E2, & E3
SGPC1ELG  E1, E2, and E3 Users in Eligible List
SGPC2DLD  Count of Loading Users in Q2, and Q3
SGPC2DSP  Q2 and Q3 Users in Dispatch List
SGPC2ELD  Count of Loading Users in E2, and E3
SGPC2ELG  E2, and E3 Users in Eligible List
SGPC3DLD  Count of Loading Users in Q3
SGPC3DSP  Q3 Users in Dispatch List
SGPC3ELD  Count of Loading Users in E3
SGPC3ELG  E3 Users in Eligible List
```

Chapter 3: REPORTS

The CA MICS Analyzer Option for VM/CMS provides both online and batch reports. These reports include overall system analysis reports and guest level reports suitable for the analysis of individual virtual machine performance. The guest level reports are useful for tracking and analyzing the performance of LINUX virtual machines deployed on a VM system running z/VM.

This section contains the following topics:

3.1 Standard Analysis Reports (see page 59)
3.2 MICF Inquiries (see page 64)
3.3 VM Management Objective Reports (see page 148)

3.1 Standard Analysis Reports

Standard analysis reports describe overall operations of the VM system. They are run in batch mode. The VM Performance Summary Report is described in the following sections:

1 - Running Standard Analysis Reports
2 - VM Performance Overview Report

3.1.1 Running Standard Analysis Reports

The CA MICS Analyzer Option for VM/CMS provides batch reporting. The following JCL can be used to produce VM batch reports:

```
//REPORT EXEC MICSSHRx
//SYSIN  DD   *
 %VMCRPT(REPORT=01,BEGIN=hh:mm:ss,END=hh:mm:ss,
         PERSIZE=minutes,CYCLE=nn,
         ID=optional title line)
/*
```

REPORT specifies the report to be run; nn is the report number as listed in Section 3.1.2. For example, to run the VM/ESA Performance Overview Report, specify REPORT=01. The default value for this parameter is 0

BEGIN and END specify the time period for the report. If multiple days are present in the cycle, all days containing this time period are reported. The default values for BEGIN and END are 09:00:00 and 17:00:00 respectively.

PERSIZE specifies the length of the reporting period in minutes. This specification allows the DETAIL data to be summarized to a level greater than the monitor recording interval. The default value for PERSIZE is 60 minutes.

CYCLE specifies the DETAIL cycle that is reported. The default value is 01.

ID provides an extra title line for the printed report. There is no default value for ID.

3.1.2 VM Performance Overview Report

The VM Performance Overview Report provides a high level summary of system performance data. This report provides information pertaining to system utilization, storage use, and scheduler activity. For most variables, the MIN, MAX, MEAN, and percentile values are provided. The report can be produced for any period greater than or equal to the monitor recording interval.

To produce VM Performance Overview Report, create and submit the following JCL:

```
//REPORT   EXEC  MICSSHRx
//SYSIN     DD    *
   %VMCRPT(REPORT=01,BEGIN=hh:mm:ss,END=hh:mm:ss,
           PERSIZE=minutes,CYCLE=nn,ID='report title')
/*
```

See Section 3.1.1 for descriptions of the parameters for the VMCRPT macro.

The following JCL example produces a report from cycle 01 of the VMXSGP file in the DETAIL timespan, with the data ranging from 9 a.m. to 5 p.m. The length of the reporting period is 480 minutes (that is, 8 hours), thus exactly one report would be produced.

```
//REPORT EXEC MICSSHRV
//SYSIN  DD      *
   %VMCRPT(REPORT=1,BEGIN=09:00:00,END=17:00:00,
       PERSIZE=480,CYCLE=01,ID=CA MICS I/S MANAGEMENT);
/*
```

VMCRPT01 22JUN93

CA MICS ANALYZER FOR VM/CMS - Z/VM PERFORMANCE OVERVIEW REPORT

SYSID:VM01 0B5435 9021-720 REPORTING PERIOD ENDING: 17:00:00 ON Wednesday, May 26, yyyy DURATION: 7852 SECS

CA MICS I/S MANAGEMENT

SYSTEM UTILIZATION

	TOTAL % CPU	%CP SYSTEM	%CP USERID	%PRB	%WAIT	TOTAL-TO-VIRTUAL SYSTEM	TOTAL-TO-VIRTUAL USERID	VIRTUAL MACHINES LOGGED ON
MEAN	52.29	3.94	12.70	35.65	39.37	1.48	1.36	1004
MIN	33.30	3.03	9.58	20.66	15.67	1.26	1.21	978
25th	47.62	3.59	11.69	31.89	35.18	1.43	1.33	986
50th	52.30	3.87	12.76	34.70	39.40	1.49	1.38	1008
75th	55.59	4.33	13.47	38.17	44.31	1.53	1.40	1023
95th	67.65	4.64	15.21	49.26	48.81	1.57	1.43	1029
MAX	73.43	4.77	16.17	58.12	57.94	1.61	1.46	1042

STORAGE

CONFIGURATION		FREE EXTENDS EXTENTS	FREE EXTENDS TOT PGS	AVAIL - LIST EMPTY	AVAIL - LIST AVAIL	% STORAGE UTILIZED	PAGING RATES EXP	PAGING RATES AUX	% UTILIZED PAGE	% UTILIZED SPOOL	% UTILIZED DUMP
FREE	806K	MEAN 0	3793	11	303	14	391	196	65	42	5
DPA	361274K	MIN 0	3647	0	53	7	173	88	61	39	5
NUCLEUS	1824K	25th 0	3793	4	190	12	331	168	62	41	5
V=R FREE	0K	50th 0	3800	5	275	14	389	198	66	41	5
V=R	0K	75th 0	3807	10	374	15	454	225	67	44	5
TOTAL	389120K	95th 0	3824	28	671	18	517	278	70	44	5
		MAX 0	3841	117	767	22	592	304	70	45	5
EXPANDED	327680K										

SCHEDULER/DISPATCHER

	VMDBKS(STATES) DORMANT	VMDBKS(STATES) ELIGIBLE	VMDBKS(STATES) DISPATCH	VMDBKS(LOADING) Q1	VMDBKS(LOADING) Q2	VMDBKS(LOADING) Q3	VMDBKS(FORCED) Q1	VMDBKS(FORCED) Q2	VMDBKS(FORCED) Q3	QDISP USERS ADDED	E1 ETS		TRANSACTIONS NUMBER	TRANSACTIONS RESPONSE
MEAN	973	0	38	0	0	0	0	0	0	914	0.71	TRIV	183849	0.12
MIN	945	0	24	0	0	0	0	0	0	160	0.59	N-TRIV	133160	1.22
25th	962	0	32	0	0	0	0	0	0	554	0.67	QDISP	49816	0.66
50th	971	0	38	0	0	0	0	0	0	588	0.71			
75th	986	0	44	0	0	0	0	0	0	631	0.75			
95th	1003	0	56	1	1	0	0	0	0	2624	0.81			
MAX	1009	0	64	2	1	1	0	0	0	2905	0.85			

Figure 3-1. VM Performance Overview Report

```
Report Variables, System Utilization section:

 TOTAL % CPU: Total percent CPU busy for all processors in the
              complex for the reporting period

 %CP:         Percent system state CPU time that is consumed by the
              system (that is, not attributable to a virtual
              machine) and percent system state CPU time
              consumed on behalf of all virtual machines

 %PRB:        Percent CPU time that is consumed while the processor is
              in interpretive mode (SIE)

 %WAIT:       Percent CPU wait time

 TOTAL TO VIRTUAL - SYSTEM: The ratio of all CPU time consumed
              divided by the CPU time that is consumed in SIE

 TOTAL TO VIRTUAL - USERID: The ratio of all CPU time consumed
              by virtual machines that is divided by the CPU time
              consumed in SIE

 VIRTUAL MACHINES LOGGED ON: Users that are logged on to the system
              during the reporting period

 Report Variables, Storage section:

 CONFIGURATION: This chart provides the average size of all
              named areas for the reporting period

 FREE EXTENDS: The rate of free storage extend operations per
              second for the period and the total number of
              pages extended

 AVAIL LIST:  The number of times the available list was empty
              during the reporting period and the average
              number of page frames in the available list

 % STORAGE UTILIZED: The percentage of DPA frames that
              are not in the available list

 PAGING RATES: The paging rate to expanded storage and to
              auxiliary storage

 % UTILIZED:  The percent of page, spool, and dump slots on
              CP-owned volumes that are in use

 Report Variables, Scheduler/Dispatcher section:
```

VMDBKS(STATES): The number of VMDBKs that are in the dormant, eligible, and dispatch lists

VMDBKS(LOADING): The number of VMDBKs by queue that are classified as loading users

VMDBKS(FORCED): The number of VMDBKs by queue that were forced into queue due to being behind schedule

QDISP USERS ADDED: The number of QUICKDISP users added to Q0 during the reporting period

E1 ETS: The E1 elapsed time slice. While variations for this value are normal, large variations in show a change in workload characteristics.

TRANSACTIONS: The number of transactions (as viewed by the scheduler) for the period. The RESPONSE is the average elapsed time for the transaction type, derived by dividing the elapsed time for the transaction type by the count of transactions for the type.

3.2 MICF Inquiries

The CA MICS Analyzer Option for VM/CMS provides a set of prepared report and color graphics inquiries categorized for analysis by service, reliability, performance, and workload. These inquiries provide an immediate source of useful reporting and color graphics presentations, and are implemented using the CA MICS Information Center Facility (MICF).

You may use Execution Time Data Selection panels to tailor the output from these inquiries. In addition, you may develop customized reports for your site with a minimum effort by modifying copies of the distributed inquiries.

The distributed inquiry names have been designed for easy identification. Each name identifies the information area being reported on, the type of output it has (i.e., color graphics, printer graphics, or printed reports), and the timespan from which it is produced (that is, DETAIL, DAILY, WEEKLY, MONTHLY). It also has its own unique identifier to distinguish it from other unique inquiries of the same type.

Consider, for example, the MICF inquiry VMXCX1:

- VMX indicates that the inquiry comes from the CA MICS VMX Information Area.
- C indicates that the output is color graphics.
- X indicates that the timespan is DETAIL.
- 1 is an identifier that distinguishes the inquiry from other VMX Information Area MICF inquiries that also have color graphics output and are produced from the DAYS timespan.

MICF is described in detail in the MICF User Guide. The list of distributed MICF inquiries, instructions on how to modify them, and descriptions of the output that they produce are provided in the following sections:

1 - VM MICF Inquiry Descriptions
2 - How to Modify Standard MICF Inquiries

3.2.1 VM MICF Inquiry Descriptions

All distributed MICF inquiries are categorized by analysis group. The CA MICS VMX information area provides MICF inquiries within two analysis groups: workload and performance.

The VM inquiries are created from DETAIL timespan files in the VMX information area, and are typically designed to allow for analysis of activity over a user-defined period of time during the day. Additionally, the tabular reports produced here are designed for running in batch, and most produce 132-character-wide reports. However, as with all MICF inquiries, these reports can be run in foreground and the output viewed online. Figure 3-2 lists the MICF inquiries contained within each of these groups:

1 - VM Workload Analysis Inquiries
2 - VM Performance Analysis Inquiries

VMX Information Area

MICF Distributed Inquiries

Report Class	Report Name	Report ID: Color Graphics	Report ID: Printer Graphics	Report ID: Tabular Report	Database File
Workload	Detail VMX IUCV/VMCF User Summary	VMXCX3	VMXPX3		VMXVXT
	Detail VMX Channel Busy Report			VMXTX2	VMXCHC
	Detail VMX Demand Scan Summary			VMXTX4	VMXSPP
	Detail VMX Device Ranking Report			VMXTX5	VMXXDV
	Detail VMX IUCV/VMCF Summary Report			VMXTX6	VMXSGP
	Detail VMX System Summary Report			VMXTX7	VMXSGP
	Detail VMX User Consumption Report			VMXTX8	VMXVXU
	Detail VMX User States Summary			VMXTX9	VMXSGP
	Detail VMX User Consumption Report (By User)			VMXTXC	VMXVXU
	Detail VMX LPAR Utilization Report			VMXTXD	
Performance	Detail VMX Concurrent Channel Busy	VMXCX1	VMXPX1		VMXSGP
	Detail VMX Percent Channel Busy	VMXCX2	VMXPX2		VMXCHC
	Detail VMX DASD Cache Usage Report			VMXTX1	VMXCDI
	Detail VMX CP Owned Volume Summary			VMXTX3	VMXCPV
	Detail VMX Device Utilization Report (By Device)			VMXTXA	VMXXDV
	Detail VMX DASD Cache Usage Report (By Device)			VMXTXB	VMXCDI
	VM Channel Path Activity Report			VMXLPA	
	VM Channel Path Load Analysis			VMXLS1	
	VM FICON Channel Path Load Analysis			VMXLS2	
	VM HiperSocket Channel Activity Report			VMXLS3	

Figure 3-2. VMX Information Area MICF Inquiry List

3.2.1.1 VM Workload Analysis Inquiries

Workload analysis inquiries indicate the level of system-wide activity. The inquiries in this category are available for user-defined reporting periods, defined as execution-time parameters. For example, you might want to produce a report on Demand Scan activity covering the period from 8:45 a.m to 9:30 a.m. The MICF inquiry will select the detail data covering the period of time requested, summarize the information, and produce a report. The workload analysis inquiries are explained in greater detail in the following sections:

1 - Detail VMX IUCV/VMCF User Summary
2 - Detail VMX Channel Busy Report
3 - Detail VMX Demand Scan Summary
4 - Detail VMX Device Ranking Report
5 - Detail VMX IUCV/VMCF Summary Report
6 - Detail VMX System Summary Report
7 - Detail VMX User Consumption Report **
8 - Detail VMX User States Summary
9 - Detail VMX User Consumption Report (By User) **
10 - VM LPAR Utilization Report

** Suitable for LINUX guest reporting.

3.2.1.1.1 Detail VMX IUCV/VMCF User Summary

This report produces a graph of IUCV/VMCF service use by name for the top n services used during the selected reporting period, where n is defined by the user, and defaults to 20.

INTENDED USE:

For analysis of what services are being heavily used. Such a report might be used to identify services that should be optimized, or services that are being used inappropriately.

FORMAT:

A horizontal bar chart in descending order of IUCV/VMCF Service used.

An example of this report is shown in Figure 3-3.

INQUIRY ID:

Color Graphics: VMXCX3
Printer Graphics: VMXPX3

DATA SOURCE (File/Timespan):

VMXVXT01 - DETAIL

DATA ELEMENTS USED:

VXUSVMID - Last SVM or CP Service Called via IUCV

CALCULATIONS:

None

USAGE CONSIDERATIONS:

None

```
                                IUCV/VMCF User Summary (Sample Data) - Top 10 Services                                1
                                          Product Manufacturing Corporation

INQUIRY:  VMXPX3                                                                                    RUN DATE:  3FEByy

                                                System Identifier=VM Day of Month=21

                                                        BAR CHART OF FREQCY

                        VXUSVMID   Last SVM or CP Service Called via IUCV                                FREQCY
                                   |
                        MAIL       |*************************************************************       30760
                                   |
                        DATABASE   |***********************                                            12134
                                   |
                        CALENDAR   |****************                                                    8120
                                   |
                        ONLINE     |*                                                                    585
                                   |
                        SYSTEM     |*                                                                    574
                                   |
                        SYSRES     |                                                                     230
                                   |
                        TAPES      |                                                                     216
                                   |
                        USER01     |                                                                      83
                                   |
                        USER02     |                                                                      82
                                   |
                        TAPEOPER   |                                                                      72
                                   |
                                   --------+-------+-------+-------+-------+-------+-------+------
                                         4000    8000    12000   16000   20000   24000   28000

                                                          Frequency
```

Figure 3-3. Detail VMX IUCV/VMCF User Summary

3.2.1.1.2 Detail VMX Channel Busy Report

This report produces a tabular listing of channel activity by CHPID for the duration of the period specified in the run-time execution parameters. The percent busy and the percent of system values represent an average for the period covered by report.

INTENDED USE:

Allows analysis of CHPIDs that are being heavily used. Such a report might be used to detect channels that are being over-utilized or under-utilized, and to detect the proportionate share of all channel activity being handled by a given channel.

FORMAT:

A tabular report in ascending order of CHPID, where each column represents a different measurement associated with a given CHPID.

An example of this report is shown in Figure 3-4.

INQUIRY ID:

Tabular Report: VMXTX2

DATA SOURCE (File/Timespan):

VMXCHC01 - DETAIL

DATA ELEMENTS USED:

CHCHPID - Channel Path ID
VMXHFSAM - High Frequency Sample Count
CHCPCBSY - Percent CHPID Busy

CALCULATIONS:

SYSBUSY, the percentage of all channel activity handled by a given channel, is obtained by adding up the summarized percent channel busy values for all CHPIDs, and placing this value in a temporary element named BUSYTOT. The percentage workload held by each channel path is then computed as follows:

SYSBUSY = CHCPCBSY / BUSYTOT*100;

USAGE CONSIDERATIONS:

The Channel Path Contention Data File (VMXCHC) is created from the 0/09 monitor records. With the introduction of z990 processor, the 0/09 no longer contains valid channel path information. It is recommended that the 0/18 or 0/20 monitor records be used for valid channel path information.

See report VMXLS1 (VM Channel Path Load Analysis).

```
                                Detail VMX Channel Busy Report (Channel Path Order)                                1
                                        Product Manufacturing Corporation

INQUIRY:  VMXTX2                                                                                  RUN DATE:  1FEByy

------------------------------------------------ SYSID=VM YEAR=yy MONTH=12 DAY=21 ------------------------------------------------

                                 Channel          Sample      Percent    Percent
                                 Path ID          Count       Busy       of System

                                   00               864       0.00 %       0.00 %
                                   01               864       0.00 %       0.00 %
                                   02               864       0.00 %       0.00 %
                                   03               864       6.13 %       8.93 %
                                   04               864       0.00 %       0.00 %
                                   05               864       0.00 %       0.00 %
                                   06               864       0.00 %       0.00 %
                                   07               864       2.77 %       4.04 %
                                   08               864       0.46 %       0.67 %
                                   09               864       0.81 %       1.18 %
                                   0A               864       4.74 %       6.91 %
                                   0B               864       2.66 %       3.87 %
                                   0C               864       2.89 %       4.21 %
                                   0D               864       3.47 %       5.05 %
                                   0E               864       3.81 %       5.56 %
                                   0F               864       3.00 %       4.38 %
                                   10               864       0.92 %       1.34 %
                                   11               864       0.00 %       0.00 %
                                   12               864       0.00 %       0.00 %
                                   13               864       0.00 %       0.00 %
                                   14               864       0.00 %       0.00 %
                                   15               864       0.00 %       0.00 %
                                   16               864       0.00 %       0.00 %
                                   17               864       0.00 %       0.00 %
                                   18               864       0.00 %       0.00 %
                                   19               864       0.00 %       0.00 %
                                   1A               864       0.00 %       0.00 %
                                   1B               864       1.85 %       2.69 %
                                   1C               864       0.00 %       0.00 %
                                   1D               864       0.00 %       0.00 %
                                   1E               864       0.00 %       0.00 %
                                   1F               864       7.29 %      10.62 %
                                   20               864       1.73 %       2.52 %
                                   21               864       2.77 %       4.04 %
                                   22               864       3.00 %       4.38 %
```

Figure 3-4. Detail VMX Channel Busy Report

3.2.1.1.3 Detail VMX Demand Scan Summary

This report produces a tabular listing of demand scan activity in time sequence. The information is summarized by the number of minutes specified by the user as a run-time execution parameter. Activity is displayed over the period of time specified by the user, again as run-time execution parameters.

INTENDED USE:

When CP needs additional storage to continue processing, it takes frames from the CP-maintained list of available frames. If CP cannot find a frame in the available list, it must take frames away from the virtual systems to replenish the available list. CP invokes a demand scan function to replenish the available list when it reaches a point called the low threshold. The function removes selected frames from user-owned frame lists, and places them on the available list. The demand scan function places frames on the available list until the list reaches a value called the high threshold. The demand scan function consists of two passes of various VMDBKs, and a special emergency pass if the two regular passes do not successfully replenish the list.

Within each pass, the function examines VMDBKs in an order that will hopefully have the least impact on a given user and on the system as a whole. For example, in the first pass of the function, long-term dormant users are examined first, followed by short-term dormant users. Next, the shared frame owned list (including system VMDBK, named saved systems, discontiguous saved segments, and shared address spaces) is examined for frames to remove, followed by the eligible list, and finally the dispatch list. With each succeeding pass, the function becomes less selective about which frames to take from users.

The VM Monitor records count indicating at which point in the demand scan function the available list was successfully replenished. For example, over a five-minute period, during an average interval, the Demand Scan function might complete after scanning the short-term dormant list in Pass 1, 90.0 times. This report indicates the average number of times per monitor interval that the demand scan function completed within a given phase of the function.

The report is intended to show how far into the Demand

Scan function it is necessary to proceed before the available frame list is replenished. If numbers start

appearing in the Emergency Pass part of the report, it can be indicative of a storage-constrained environment.

FORMAT:

A tabular report in chronological order, where each line represents a summarization of DETAIL data across a user-specified period of time. The columns within each Demand Scan Pass represent the average number of times per monitor interval that the Demand Scan function completed in that phase. They are ordered from left to right in the same order that the system algorithm proceeds through the various lists of VMDBKs. The following abbreviations are used for column headings with each pass:

LTD	Long Term Dormant VMDBKs
STD	Short Term Dormant VMDBKs
SYS/SHAR	System VMDBK, Shared Storage VMDBKs (NSSs and DCSSs), and Shared Address Spaces
ELST	Eligible List VMDBKs
DLST	Dispatch List VMDBKs

The final three columns of the report show the following averages for the reporting interval (these are not totals):

SCAN NCMP - The average number of times per monitor interval that the Demand Scan Function did not complete successfully.

ALIST EMPTY - The average number of times per monitor interval that the available frame list was empty.

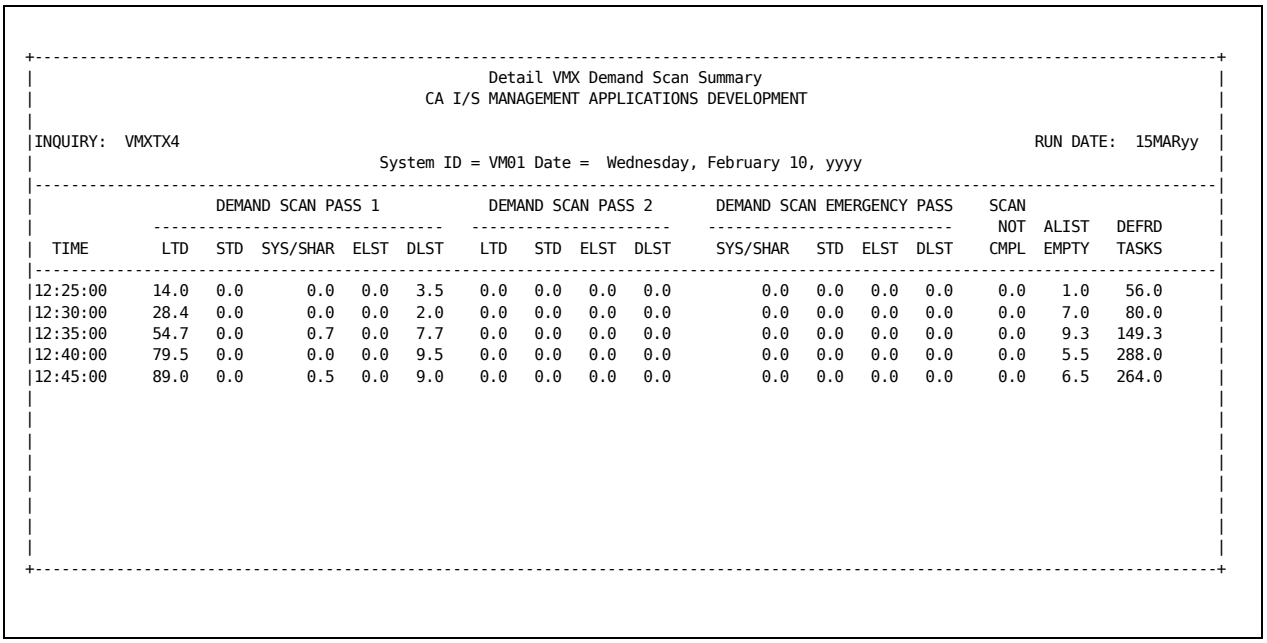
DFRD TASKS - The average number of tasks per monitor interval that were deferred while waiting for a page of real storage. This represents the average number of requests for real frames that have not been satisfied yet because no frames are available.

An example of this report is shown in Figure 3-5.

INQUIRY ID:

Tabular Report: VMXTX4

```
DATA SOURCE (File/Timespan):

   VMXSPP01 - DETAIL

DATA ELEMENTS USED:

   SPPLD1CT - DScan Pass 1 Complete Long Term Dormant
   SPPDM1CT - DScan Pass 1 Complete Short Term Dormant
   SPPSY1CT - DScan Pass 1 Complete SYSTEM VMDBK (* note)
   SPPSS1CT - DScan Pass 1 Complete Shared Frame Owned List
   SPPEL1CT - DScan Pass 1 Complete Eligible List
   SPPDL1CT - DScan Pass 1 Complete Dispatch List
   SPPLD2CT - DScan Pass 2 Complete Long Term Dormant
   SPPDM2CT - DScan Pass 2 Complete Short Term Dormant
   SPPEL2CT - DScan Pass 2 Complete Eligible List
   SPPDL2CT - DScan Pass 2 Complete Dispatch List
   SPPSSECT - DScan Pass E Complete Shared Frame Owned List
   SPPDMECT - DScan Pass E Complete Short Term Dormant
   SPPELECT - DScan Pass E Complete Eligible List
   SPPSYECT - DScan Pass E Complete SYSTEM VMDBK (* note)
   SPPDLECT - DScan Pass E Complete Dispatch List
   SPPNOCMP - Demand Scan Not Completed Successfully
   SPPALMPT - Times Available List Became Empty
   SPPFRQWT - Deferred Tasks Waiting for A Frame

   Note:  SPPSY1CT and SPPSYECT are only applicable to
          systems prior to VM/ESA Release 2.

 CALCULATIONS:

   The various elements to be reported on are summarized
   across processors, using CA MICS summarization, to provide
   system-wide totals.  CA MICS summarization is then invoked
   a second time to summarize across the user-specified time
   period, if necessary.  Finally, prior to reporting, each
   element is averaged by dividing it by the number of
   monitor intervals.

USAGE CONSIDERATIONS:

   To obtain a detail report, with one line per monitor
   interval, specify 0 for the Summarization Period in the
   run-time execution parameters.
```



Detail VMX Demand Scan Summary
CA I/S MANAGEMENT APPLICATIONS DEVELOPMENT

INQUIRY: VMXTX4 RUN DATE: 15MARyy

System ID = VM01 Date = Wednesday, February 10, yyyy

	DEMAND SCAN PASS 1					DEMAND SCAN PASS 2				DEMAND SCAN EMERGENCY PASS				SCAN		
TIME	LTD	STD	SYS/SHAR	ELST	DLST	LTD	STD	ELST	DLST	SYS/SHAR	STD	ELST	DLST	NOT CMPL	ALIST EMPTY	DEFRD TASKS
12:25:00	14.0	0.0	0.0	0.0	3.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	56.0
12:30:00	28.4	0.0	0.0	0.0	2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.0	80.0
12:35:00	54.7	0.0	0.7	0.0	7.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	9.3	149.3
12:40:00	79.5	0.0	0.0	0.0	9.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.5	288.0
12:45:00	89.0	0.0	0.5	0.0	9.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.5	264.0

Figure 3-5. Detail VMX Demand Scan Summary

3.2.1.1.4 Detail VMX Device Ranking Report

This report produces a tabular ranking of devices by either subchannel identifier or device number (that is, summarized across subchannels), in descending order of utilization. The devices reported here include those that are owned by this system, and those that are shared among multiple systems within a Cross System Extensions (CSE) complex, if defined. A run-time execution parameter is used to specify whether the report should be by subchannel ID or device number. The information is summarized across the period of time specified by the user, again as run-time execution parameters.

INTENDED USE:

To provide an analyst with the necessary information about device activity for the purpose of monitoring what devices are being used, and at what rates. Such a report might be used when exploring periods of poor system performance to detect a DASD bottleneck. Another use might be for examining what devices are being under-utilized so that activity can be balanced.

FORMAT:

A tabular report in order by either subchannel identifier or device number.

The first column of the report contains either the device number or the device address, depending on which run-time option was selected. The next three columns contain the device type, control unit model, and the volume serial number, if applicable. The remaining columns of the report are described below:

I/O COUNT - The number of start and resume subchannels issued to this device over the reporting period.

I/O %BUSY - The percentage of the time during the reporting period that this device was busy with I/O operations.

I/O RATE - The number of start and resume subchannel operations per second for this device during the period of the report.

I/O MSEC - The average number of milliseconds

consumed per I/O operation.

PENDING TIME - The sum, for all I/O requests to this device during the reporting period, of the time intervals between acceptance of the start function at the subchannel and acceptance of the first command associated with the initiation of channel program execution at the device. The time is shown in seconds.

CONNECT TIME - The time, in seconds, that this device was logically connected to the channel path during the reporting period for the purposes of searching and transferring data between it and the channel subsystem.

DISCONNECT TIME - The time, in seconds, that this device was logically disconnected from the channel path during the reporting period due to positioning functions such as Seek and Set Sector operations, and delays for path reconnection.

ACTONLY TIME - The sum of time that the subchannel was device-active but not subchannel-active at the end of an I/O operation or chain of I/O operations initiated by START SUBCHANNEL. The time is measured in seconds, and this value includes all I/O requests to this device during the reporting period.

SAMPLES - The number of measurement facility samples that took place during the reporting period.

An example of this report is shown in Figure 3-6.

```
INQUIRY ID:

   Tabular Report:  VMXTX5

DATA SOURCE (File/Timespan):

   VMXXDV01 - DETAIL

DATA ELEMENTS USED:

   VMXDVSID - Host Subchannel ID
   VMXDVDEV - Device Number
   DEVTYPE  - Device Type
   CUTYPE   - Control Unit Type
```

```
   VOLSER   - Volume Serial Number
   VMXMSSCH - Count of SSCH and RSCH for This Device
   XDVPCBSY - Percent Device Busy
   XDVFPNTM - Device Function Pending Time
   XDVDCNTM - Device Connect Time
   XDVDDCTM - Device Disconnect Time
   XDVDAOTM - Device-Active-Only Time
   VMXMSAMP - Measurement Facility Samples
   XDVSHRD  - CSE Shared Volume Indicator

 CALCULATIONS:

   The detail records for the user-specified reporting period
   are extracted from the XDV files, and summarized, using
   CA MICS summarization, and across subchannels, if that was
   requested.  CA MICS summarization is then invoked a second
   time to summarize across the reporting period, producing a
   single record per device or subchannel ID.  The following
   calculations are then performed for the report items that
   do not come directly from the XDV file:

      /* DEVICE I/O RATE ELEMENT - IORATE */
      IORATE = VMXMSSCH / DURATION;

      /* MILLISECONDS PER I/O - MSEC */
      MSEC = MAX(((SUM(XDVFPNTM,XDVDCNTM,XDVDDCTM)/VMXMSSCH)*
             1000),0);

   If no I/O activity occurred during a reporting period
   (VMXMSSCH EQ ZERO) for a device, the data element MSEC is
   set to zero.


USAGE CONSIDERATIONS:

   Keep in mind that in VM, a single device may have as many
   as eight subchannel paths associated with it.  Thus, if
   this report is produced by subchannel ID, the same device
   may appear several times in the report.
```

VMX Device Ranking Report By Percent Device Utilized
Product Manufacturing Corporation

INQUIRY: VMXTX5 RUN DATE: 18JUNyy

System ID = VM01 Date = Thursday, May 27, yyyy

DEVICE				I/O				TIMES				
NUMBER	TYPE	CONTROL UNIT	VOLSER	COUNT	%BUSY	RATE	MSEC	PENDING	CONNECT	DISCONN	ACTONLY	SAMPLES
043F	3380K	3880-23	VMBX74	41,219	57.58	11.54	49.91	48.956	250.322	1758.074	0.000	41,219
043C	3380K	3880-23	VMBX71	40,822	56.30	11.43	49.28	162.109	614.331	1235.437	0.000	40,822
0430	3380K	3880-23	VMBX62	71,144	49.84	19.91	25.03	316.606	360.169	1104.109	0.000	71,144
0C50	CTCA	3088-04		5,328	48.32	1.49	323.89	0.325	6.215	1719.164	0.000	5,328
0431	3380K	3880-23	VMBX63	23,194	34.98	6.49	53.89	96.945	136.036	1016.852	0.000	23,194
098C	3380K	3990-2	JES3Q1	18,483	32.65	5.18	63.08	523.655	305.197	337.116	0.000	18,483
07E7	3380D	3880-3	VMCX11	31,362	32.48	8.78	36.99	27.471	730.958	401.687	0.000	31,362
043A	3380K	3880-23	VMBX70	19,817	26.96	5.55	48.60	110.146	226.872	626.114	0.000	19,817
086A	3380E	3880-3	VMOS1E	9,026	25.44	2.53	100.65	25.047	872.984	10.446	0.000	9,026
03B9	3380K	3880-23	VMBX56	25,717	18.09	7.22	25.06	105.506	151.098	387.757	0.000	25,717
0644	3380D	3880-23	VMBPG4	16,393	17.35	4.59	37.80	4.250	336.024	279.369	0.000	16,393
03FD	3390-2	3990-3	TVS105	13,425	17.02	3.76	45.29	64.849	539.754	3.375	0.000	13,425
0646	3380D	3880-23	VMBPG5	23,443	16.87	6.57	25.70	6.090	214.187	382.128	0.000	23,443
03B6	3380K	3880-23	VMBX54	19,760	15.18	5.55	27.36	90.796	120.406	329.482	0.000	19,760
03BA	3380K	3880-23	VMBX57	20,055	14.88	5.63	26.42	83.804	140.338	305.811	0.000	20,055
03F7	3390-2	3990-3	SAR104	52,419	12.34	14.67	8.41	194.028	134.251	112.446	0.000	52,419
0641	3380D	3880-23	VMBX75	24,238	11.35	6.79	16.72	7.893	98.560	298.878	0.000	24,238
0592	3380D	3880-23	VMBX11	11,950	8.19	3.35	24.46	2.779	60.539	229.023	0.000	11,950
049A	3380K	3880-23	LIBP04	11,753	8.18	3.29	24.87	118.438	135.189	38.616	0.000	11,753
03D0	3390-2	3990-3	SAR111	26,480	7.93	7.41	10.70	45.725	68.923	168.669	0.000	26,480
0643	3380D	3880-23	VMBX76	6,096	5.87	1.71	34.37	5.461	98.202	105.867	0.000	6,096
0597	3380D	3880-23	VMBX25	10,758	5.81	3.01	19.29	2.404	59.667	145.402	0.000	10,758
0C47	CTCA	3088-04		87,252	3.97	24.43	1.63	4.234	66.146	71.527	0.000	87,252

Note: A volume with '@' by its VOLSER indicates a CSE shared device.

Figure 3-6. Detail VMX Device Ranking Report

3.2.1.1.5 Detail VMX IUCV/VMCF Summary Report

This report produces a tabular summary of CP system use of IUCV (Inter-User Communication Vehicle), and user usage of both IUCV and VMCF (VM Communication Facility), for the reporting period specified as run-time execution parameters.

INTENDED USE:

To provide an overview of the utilization of system services, such as *MSG, and *BLOCKIO, as well as user activity using IUCV and VMCF. The report can be utilized to monitor the data transfer activity, and to determine whether an inordinate number of transfer failures are occurring. Such failures may point to problem areas in need of further analysis.

FORMAT:

A tabular report in which the first column lists either a CP system service name that makes use of IUCV, 'USER IUCV', or 'USER VMCF'. The next two columns contain the number of successful data transmissions sent to and from a given system service, or users. The last column displays the number of failed transmissions that took place during the reporting period.

An example of this report is shown in Figure 3-7.

INQUIRY ID:

Tabular Report: VMXTX6

DATA SOURCE (File/Timespan):

VMXSGP01 - DETAIL

DATA ELEMENTS USED:

SGPSTxxx - Successful Data Transfers By IUCV to Service xxx
SGPSTUSR - Successful Data Transfers By IUCV to a User
SGPSTVMC - Successful Data Transfers By VMCF to a User
SGPSFxxx - Successful Data Transfers By IUCV from Service xxx
SGPSFUSR - Successful Data Transfers By IUCV from a User
SGPSFVMC - Successful Data Transfers By VMCF from a User
SGPUTxxx - Unsuccessful Data Transfers By IUCV to Service xxx

```
   SGPUTUSR - Unsuccessful Data Transfers By IUCV to a User
   SGPUTVMC - Unsuccessful Data Transfers By VMCF to a User


   Where service xxx has the following values:

        MSA - *MSGALL
        MSG - *MSG
        RPI - *RPI
        BLK - *BLOCKIO
        MON - *MONITOR
        SIG - *SIGNAL
        SPL = *SPL
        ACC = *ACCOUNT
        CRM = *CRM
        IDN = *IDENT
        LOG = *LOGREC
        SYM = *SYMPTOM

 CALCULATIONS:

   The detail records for the user specified reporting period
   are extracted from the XDV files and summarized, using
   CA MICS summarization, across the reporting period,
   producing one record per day per system, for the entire
   reporting period.  Records are then produced for each
   system service, and new data elements are created to
   contain the service name (SERVICE), the successful send
   tos (SENDTO), the successful send froms (SENDFROM), and
   the unsuccessful send tos (FAILED).  The items in
   parentheses are column heads in the report found in Figure
   3-17.  The names of these elements follow the standards
   established in the section DATA ELEMENTS USED, above.

USAGE CONSIDERATIONS:

   None
```

```
                    Detail VM IUCV/VMCF Summary Report
                     Product Manufacturing Corporation

INQUIRY:  VMXTX6                                              RUN DATE:  2FEByy
                           System ID = VM  ,Day = 21
                                    DATA TRANSFERS
                SERVICE       SENT TO  SENT FROM     FAILED

                *ACCOUNT            0      1,437          0
                *BLOCKIO          641        641          0
                *CRM                0          0          0
                *IDENT              0          0          0
                *LOGREC             0          5          0
                *MONITOR        2,406      2,406          0
                *MSG                0      5,905          0
                *MSGALL             0          0          0
                *RPI            5,933      6,944          0
                *SIGNAL        13,020     13,020          0
                *SPL                0     13,020          0
                *SYMPTOM            0          0          0
                USER IUCV     148,835     13,020          0
                USER VMCF      18,376     18,376          0

                SYSID/DAY     189,211     74,774          0


                Total         189,211     74,774          0
```

Figure 3-7. Detail VMX IUCV/VMCF Summary Report

3.2.1.1.6 Detail VMX System Summary Report

This report produces a tabular listing of overall system activity in time sequence. The information is summarized by the number of minutes specified by the user as a run-time execution parameter. Activity is displayed over the period of time specified by the user, again as run-time execution parameters.

INTENDED USE:

To monitor system activity at a gross level. It can be used for post-analysis of periods of poor system performance or for performing comparisons of activity at different times of a day, different days of a week, etc.

FORMAT:

A tabular report in chronological order, where each line represents a summarization of DETAIL data across a user-specified period of time. The time column represents the timestamp of the end of the summarized interval. The next five columns all represent percentages: percent wait time, percent CP system time, percent CPU supervisor time charged to users, percent user problem state time, and the combined system, supervisor, and problem state percent of time. The next two columns contain the total-to-virtual-ratios: first the ratio of total CPU time used to the CPU time spent in emulation mode; and second, the ratio of the total CPU time charged to users to the time spent in emulation mode.

The remaining columns of the report are described below:

MAIN UTIL - The percent of total pageable frames in the system that are in the available list.

STORAGE CORESAT - A storage saturation index, computed as shown below in the section on calculations. It indicates how many times storage must be expanded to eliminate paging.

CONTENTION RATIO - A ratio of contention for spots on the dispatch list, computed as shown below in the section on calculations. A value of 1 indicates no contention.

PAGE RATE TO ASTOR - The paging rate to auxiliary

storage

PAGE RATE TO XSTOR - The paging rate to expanded storage

USERS LOGGED ON - The number of logged on users

USERS TOT - The average number of users active in total time (emulation mode and/or CP Supervisor) per monitor interval

USERS ACTIVE VIRT - The average number of users active in emulation mode per monitor interval

USERS VF - The average number of users utilizing vector facility processing

PCT AREA USED PAGE - The percent of page slots used

PCT AREA USED SPOOL - The percent of spool slots used

An example of this report is shown in Figure 3-8.

INQUIRY ID:

Tabular Report: VMXTX7

DATA SOURCE (File/Timespan):

VMXSGP01 - DETAIL

DATA ELEMENTS USED:

SGPPCWAT - Percent CPU Wait Time
SGPPCSYS - Percent CPU Virtual Time
SGPPCSUP - Percent CPU Supervisor Time
SGPPCPRB - Percent CPU SYSTEM Time
SGPPCTOT - Percent CPU Total Time
SGPTVRAT - Total to Virtual Ratio (System)
SGPTVRVM - Total to Virtual Ratio (Users)
SGPSTGUT - Storage Utilization
SGPPGRTA - Page Rate to Auxiliary Storage
SGPPGRTX - Page Rate to Expanded Storage
SGPUSRLG - Number of Users Logged On
SGPACTIV - Active Users (Total Time)
SGPACTVV - Active Users (Emulation Mode)
SGPPCPSU - Page Slot Utilization
SGPPCSSU - Spool Slot Utilization

CALCULATIONS:

The various elements to be reported on are summarized across the user-specified time period, if necessary. The following calculations are then performed for the storage saturation element, CORESAT, and the contention ratio, CRATIO:

```
CORESAT = SGPDPASZ/(SGPESWSS+SGPRSVPG+SGPLOKFR+SGPRSHRF);

CRATIO = (SGPCDISP+SGPCELIG) / SGPCDISP;
```

Finally, prior to reporting, the "number of users" data elements are averaged by dividing by the number of monitor intervals

USAGE CONSIDERATIONS:

To obtain a detail report, with one line per monitor interval, specify zero for the summarization period in the run-time execution parameters.

```
                                VM System Summary Report for 5 Minute Intervals
                                      Product Manufacturing Corporation

INQUIRY:  VMXTX7                                                                                                    RUN DATE: 30JUNyy
                                                    System ID = VM  ,Day = 21
-------------------------------------------------------------------------------------------------------------------------------------
         <----      CPU USE       ----> Total/Virt                              Page  Rate <-----    Users     -----> Pct. Area
                  -- USER --       -  Ratio -  Main Storage Contention    -   To    - Logged      <-- Active --> - Used  -
  Time    Wait System  SUPV  Virt Total  SYS  USER  Util CORESAT   Ratio       ASTOR XSTOR   On        Tot       Virt PAGE SPOL
-------------------------------------------------------------------------------------------------------------------------------------
 8:45:00  80.4    0.5   1.3  17.7  19.5  1.1   1.1   485   0.2       1.0           5     7   61         54         53 20.0 11.9
 9:00:00  79.8    0.5   1.3  18.3  20.1  1.1   1.1   513   0.2       1.0          32    34   62         56         55 20.0 11.9
 9:15:00  73.4    0.8   1.9  23.9  26.5  1.1   1.1   468   0.2       1.0         115   171   61         57         58 20.1 11.9
 9:30:00  77.5    0.6   1.7  20.1  22.4  1.1   1.1   467   0.2       1.0          26    41   61         50         49 20.2 11.9
 9:45:00  77.1    0.5   1.6  20.8  22.9  1.1   1.1   502   0.2       1.0          21    36   63         56         55 20.3 11.9
10:00:00  78.5    0.9   1.6  18.9  21.5  1.1   1.1   507   0.2       1.0         398   349   64         60         60 21.6 11.9
10:15:00  76.6    0.6   1.9  20.8  23.4  1.1   1.1   439   0.2       1.0          64   116   64         60         60 21.8 11.9
10:30:00  78.8    0.8   1.6  18.7  21.1  1.1   1.1   473   0.2       1.0         107   205   63         57         55 21.8 11.9
10:45:00  58.7    1.0   2.7  37.5  41.2  1.1   1.1   484   0.2       1.0         147   161   64         58         58 21.4 11.9
11:00:00  68.8    1.0   2.1  28.0  31.2  1.1   1.1   475   0.2       1.0         170   249   65         61         61 21.8 11.9
11:15:00  71.8    0.6   1.5  26.0  28.1  1.1   1.1   460   0.2       1.0          46    87   65         61         60 21.5 11.9
11:30:00  73.4    0.7   1.8  23.9  26.5  1.1   1.1   478   0.2       1.0          42    80   64         59         58 21.6 11.9
11:45:00  78.2    0.8   1.9  19.0  21.7  1.1   1.1   515   0.2       1.0          49    92   64         57         56 22.0 11.9
```

Figure 3-8. Detail VMX System Summary Report

3.2.1.1.7 Detail VMX User Consumption Report

This report produces a tabular listing of summarized user resource consumption ranked by either CPU time consumed, average working set size, or virtual I/O rate. The ranking order is determined by a run-time execution parameter. The period of time covered by the report is also specified by the user, again as run-time execution parameters.

INTENDED USE:

To allow analysis of what sorts of workloads users are putting on the systems. It can be used, for example, to detect users who are putting an inordinate load on the system, providing the basis for further analysis. This report is useful for monitoring the resource consumption of individual LINUX guests running under z/VM.

FORMAT:

A tabular listing that contains one line of information for each user who was logged on to the system during the reporting period. In addition to the user ID, the following columns are shown on the report:

COUNT - The number of monitor intervals in which information was recorded for this user. For users running in V=V MP mode, there is a record for each virtual CPU associated with the user.

CPU USAGE TOTAL - The total CPU time charged to this user over the reporting period. This includes both time spent in emulation mode and time spent by CP on behalf of this user.

CPU USAGE VIRT - The total CPU time used by this user for productive work (in emulation mode) during the reporting period. It does not include the time spent by CP on behalf of this user.

CPU USAGE RATE - The rate of CPU usage for this user, derived as indicated below in the calculations section.

CPU USAGE VECTOR - The time during the reporting period that this user used the Vector Facility in interpretive execution mode.

WSS PAGES AVG - The average working set size, in

```
          pages, for this user during the reporting period.

          WSS PAGES MAX -  The maximum working set size, in
          pages, for this user during the reporting period.

          VIRTUAL I/O TOTAL - The total number of I/O
          operations to virtual devices for this user during
          the reporting period.

          VIRTUAL I/O RATE - The rate of Virtual I/O operations
          for this user, derived as indicated below in the
          calculations section.

          SPOOL READS and WRITES - The total number of spool
          reads and writes performed by this user during the
          reporting period.

          PAGE READS and WRITES - The total number of page
          reads and writes for this user during the reporting
          period.

      An example of this report is shown in Figure 3-9.

INQUIRY ID:

      Tabular Report:  VMXTX8

DATA SOURCE (File/Timespan):

      VMXVXU01 - DETAIL

DATA ELEMENTS USED:

      USER     - User Identification
      INTERVLS - Number of Recording Intervals
      VXUTTIME - Total CPU Time Used
      VXUVTIME - Virtual CPU Time Used
      VXUSPRIO - Spool Read I/Os Issued
      VXUSPWIO - Spool Write I/Os Issued
      VXUPGRIO - Page Read I/Os Issued
      VXUPGWIO - Page Write I/Os Issued

CALCULATIONS:

      The maximum working set size MAXWSS, is extracted from
      field VXUESWSS and then the various elements to be
      reported on are summarized across the user specified
      reporting period.  The following computations are then
      performed to obtain the CPU consumption rate (CPURATE),
```

the average working set size (AVGWSS), the Virtual I/O total (VIORATE), and the Virtual I/O rate (VIORATE):

```
CPURATE =  VXUTTIME / DURATION;

AVGWSS  =  VXUESWSS / INTERVLS;

VIOTOTAL = SUM(VXUCONIO, VXUDASIO, VXUVURIO, VXUCTCIO,
               VXUMSCIO);

VIORATE  = VIOTOTAL / DURATION;
```

USAGE CONSIDERATIONS:

None

Detail VM User Consumption Reported by CPU CONSUMPTION RATE
Product Manufacturing Corporation

INQUIRY: VMXTX8 RUN DATE: 2FEByy

System ID = VM ,Day = 21

	C P U U S A G E			WSS PAGES		VIRTUAL I/O		S P O O L		P A G E	
USERID	COUNT	TOTAL	VIRT	RATE	AVG	MAX	TOTAL	RATE	READS	WRITES	READS	WRITES
VTAM	72	566.84	380.05	0.131	1183	1187	19	0	0	0	0	0
PAYROLL	12	14.30	13.89	0.021	63	150	411	1	0	0	99	121
USER01	72	56.30	40.24	0.013	142	162	3,121	1	1	0	129	21
CONTRACT	72	54.15	46.73	0.013	58	188	5,651	1	386	0	315	761
DOCS	72	51.98	30.00	0.012	292	396	37,576	9	428	0	2,399	2,219
USER06	2	0.86	0.60	0.009	114	142	411	4	6	0	0	0
USER02	18	8.48	7.99	0.008	59	91	340	0	0	0	145	124
SECURITY	72	29.71	17.44	0.007	254	358	9,116	2	4	0	335	271
USER03	56	22.23	20.20	0.007	77	148	6,275	2	6	0	226	455
CALENDAR	72	28.48	11.57	0.007	364	557	19,519	5	5	0	337	322
USER04	24	9.03	8.12	0.006	53	160	1,371	1	29	0	263	263
USER05	72	25.50	13.55	0.006	40	64	14,351	3	8	0	84	115
MAIL	72	25.01	7.24	0.006	115	168	13,580	3	2,336	0	215	184
USER100	4	1.15	0.96	0.006	85	152	117	1	0	0	0	0

Figure 3-9. Detail VMX User Consumption Report

3.2.1.1.8 Detail VMX User States Summary

```
This report produces a tabular listing of user states
(various wait states, running, and so on) in time sequence.
The information is summarized by the number of minutes
specified by the user as a run-time execution parameter.
Activity is displayed over the period of time specified by
the user, again as run-time execution parameters.

INTENDED USE:

   This report is designed for monitoring the kinds of states
   in which users are spending their time.  It can be used to
   see whether users are often waiting for a particular
   resource (such as the CPU, or I/O).

FORMAT:

   A tabular report in chronological order, where each line
   represents a summarization of DETAIL data across a user-
   specified period of time.  The time column represents the
   timestamp of the end of the summarized interval.  The
   first set of columns report on the VMDBKs in the dispatch
   list, and what states they are in.  Each value is an
   average for a monitor interval:

        TIDL  - Number of VMDBKs in test idle but not SVM
                (Service Virtual Machine) wait

        SVM   - Number of VMDBKs in SVM (Service Virtual
                Machine) wait

        I/O   - Number of VMDBKs in I/O wait

        CF    - Number of VMDBKs in console function wait

        SIM   - Number of VMDBKs in simulation wait

        PAG   - Number of VMDBKs in page wait

        CPU   - Number of VMDBKs in CPU wait

        RUN   - Number of VMDBKs running on a CPU

        ASIO  - Number of VMDBKs in asynchronous I/O wait

        OTHER - Number of VMDBKs in an undefined state
```

```
        QDSP  - Number of Quickdisp VMDBKs in the dispatch
                list

        TOTAL - Total number of VMDBKs in the dispatch list


   The next two columns are for the eligible list:

        TOTAL - Total number of VMDBKs in the eligible list

        SVM   -  Number of VMDBKs on the eligible list that
                 are in SVM (Service Virtual Machine) wait


   The last two columns are for the dormant list:

        TOTAL - Total number of VMDBKs in the dormant list

        SVM   - Number of VMDBKs on the dormant list that are
                in SVM (Service Virtual Machine) wait


   An example of this report is shown in Figure 3-10.

INQUIRY ID:

   Tabular Report:  VMXTX9

DATA SOURCE (File/Timespan):

   VMXSGP01 - DETAIL

DATA ELEMENTS USED:

   SGPTIDL  - Times a VMDBK Found in Test Idle State
   SGPTSVM  - Times a VMDBK in Test Idle and SVM State
   SGPTIOWT - Times a VMDBK was Found in I/O Wait
   SGPTCFWT - Times a VMDBK was Found in CF Wait
   SGPTSMWT - Times a VMDBK was in Simulation Wait
   SGPTPGWT - Times a VMDBK was Found in Page Wait
   SGPTCPWT - Times a VMDBK was Found in CPU Wait
   SGPCPURN - Number of VMDBKs Running on a Processor
   SGPNASIO - Times VMDBK was in Asynchronous I/O Wait State
   SGPTOTHR - Times VMDBK was in Undefined State
   SGPNQDSP - Number of Quickdisp Users in DLIST
   SGPCDISP - Count of Users in Dispatch List
   SGPCELIG - Number of VMDBKs in the Eligible List
   SGPESVMW - Eligible List VMDBKs in SVM Wait
   SGPCDORM - Number of Users in the Dormant List
```

SGPDSVMW - Dormant List VMDBKs in SVM Wait

CALCULATIONS:

The various elements to be reported on are summarized across the user-specified time period, if necessary. All reported elements are then averaged by dividing by the number of monitor recording intervals.

USAGE CONSIDERATIONS:

To obtain a detail report, with one line per monitor interval, specify 0 for the summarization period in the run-time execution parameters.

Detail VM User States Summary - 5 Minute Intervals
Product Manufacturing Corporation

INQUIRY: VMXTX9 RUN DATE: 3AUGyy

System ID = VM ,Day = 21

TIME	D I S P A T C H L I S T											ELIGIBLE		DORMANT	
	TIDL	SVM	I/O	CF	SIM	PAG	CPU	RUN	ASIO	OTHER	QDISP	TOTAL	TOTAL	SVM	TOTAL	SVM
8:35:35	8	0	1	1	3	0	1	0	5	0	2	19	0	0	882	0
8:40:35	8	0	1	2	3	0	1	0	4	0	2	19	0	0	884	0
8:45:35	4	1	0	2	4	1	1	0	6	0	3	19	0	0	895	0
8:50:35	5	0	0	2	2	0	0	0	0	0	1	9	0	0	911	0
8:55:35	5	0	0	2	3	0	0	0	7	0	2	17	0	0	910	0
8:58:35	5	1	1	2	3	0	1	0	3	0	2	16	0	0	896	0
9:05:34	7	0	0	2	1	0	0	0	1	0	2	11	0	0	902	0
9:10:34	6	0	1	2	2	0	0	0	3	0	2	14	0	0	900	0
9:15:34	6	0	1	1	0	0	0	0	5	0	2	13	0	0	880	0
9:20:34	6	0	0	1	1	0	0	0	2	0	1	10	0	0	876	0
9:25:34	6	0	1	1	2	0	0	0	3	0	2	13	0	0	883	0
9:30:34	5	1	0	1	2	0	0	0	7	0	2	16	0	0	878	0
9:35:34	7	1	1	2	1	0	1	0	5	0	2	18	0	0	888	0
9:38:35	3	0	0	1	1	1	1	0	6	0	1	13	0	0	899	0
9:41:35	4	0	1	0	2	1	1	0	1	0	2	10	0	0	913	0

Figure 3-10. Detail VMX User States Summary

3.2.1.1.9 Detail VMX User Consumption Report (by User)

This report produces a tabular listing of summarized resource consumption ordered by time. The data is summarized over the reporting interval specified when the report is executed. The period of time covered by the report and the user or users reported are also specified as run-time execution parameters.

INTENDED USE:

To allow analysis of resource consumption by individual users during the day. It can be used, for example, to detect users who are putting an inordinate load on the system, providing the basis for further analysis. This report is useful for monitoring the resource consumption of individual LINUX guests running under z/VM.

FORMAT:

A tabular listing that contains one line of information for each report interval (from the execution parameters) for the user logged on to the system during the reporting period. In addition to the TIME, the following columns are shown on the report:

COUNT - The number of monitor intervals in which information was recorded for this user. For users running multiple virtual CPUs, the data for all CPUs is summarized together to provide data for the entire virtual machine.

CPU USAGE TOTAL - The total CPU time charged to this user over the reporting period. This includes both time spent in emulation mode and time spent by CP on behalf of this user.

CPU USAGE VIRT - The total CPU time used by this user for productive work (in emulation mode) during the reporting period. It does not include the time spent by CP on behalf of this user.

CPU USAGE RATE - The rate of CPU usage for this user, derived as indicated below in the calculations section.

WSS PAGES AVG - The average working set size, in pages, for this user during the reporting period.

```
        WSS PAGES MAX - The maximum working set size, in
        pages, for this user during the reporting period.

        VIRTUAL I/O TOTAL - The total number of I/O
        operations to virtual devices for this user during
        the reporting period.

        VIRTUAL I/O RATE - The rate of virtual I/O operations
        for this user, derived as indicated below in the
        calculations section.

        SPOOL READS and WRITES - The total number of spool
        reads and writes performed by this user during the
        reporting period.

        PAGE READS and WRITES - The total number of page
        reads and writes for this user during the reporting
        period.

   An example of this report is shown in Figure 3-11.

INQUIRY ID:

   Tabular Report:  VMXTXC

DATA SOURCE (File/Timespan):

   VMXVXU01 - DETAIL

DATA ELEMENTS USED:

   USER     - User Identification
   INTERVLS - Number of Recording Intervals
   VXUTTIME - Total CPU Time Used
   VXUVTIME - Virtual CPU Time Used
   VXUSPRIO - Spool Read I/Os Issued
   VXUSPWIO - Spool Write I/Os Issued
   VXUPGRIO - Page Read I/Os Issued
   VXUPGWIO - Page Write I/Os Issued


CALCULATIONS:

   The maximum working set size MAXWSS, is extracted from
   field VXUESWSS and then the various elements to be
   reported on are summarized across the user specified
   reporting period.  The following computations are then
   performed to obtain the CPU consumption rate (CPURATE),
   the average working set size (AVGWSS), the virtual I/O
```

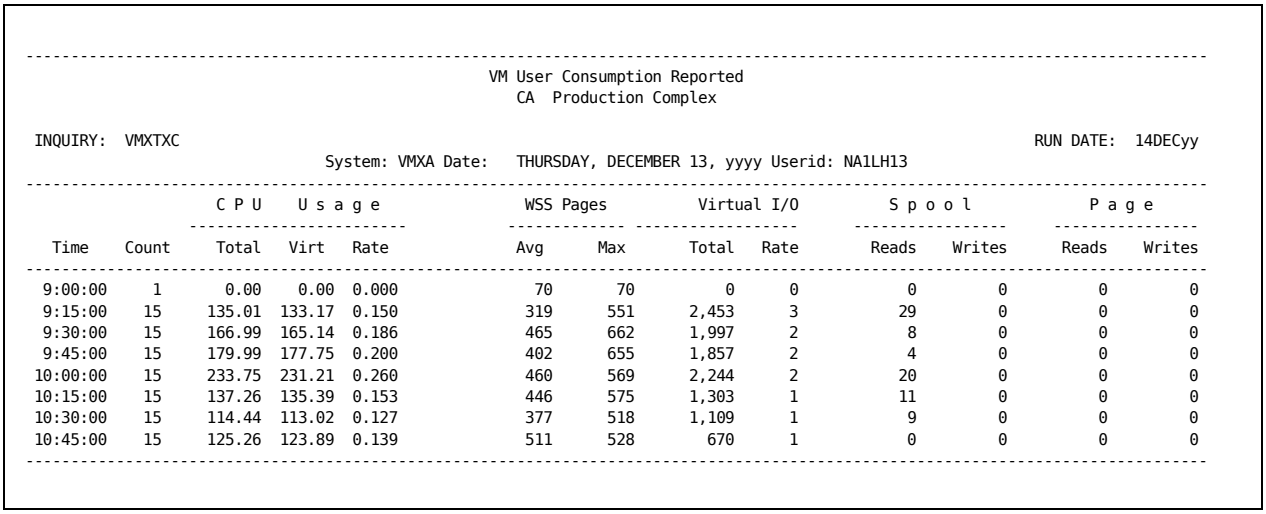
total (VIOTOTAL), and the virtual I/O rate (VIORATE):

```
CPURATE = VXUTTIME / DURATION;

AVGWSS = VXUESWSS / INTERVLS;

VIOTOTAL = SUM(VXUCONIO, VXUDASIO, VXUVURIO, VXUCTCIO,
               VXUMSCIO);

VIORATE = VIOTOTAL / DURATION;
```

USAGE CONSIDERATIONS:

For virtual machines that run with more than one virtual CPU (such as an MVS guest), data for all defined virtual CPUs are summarized to provide a picture of the whole virtual machine over the specified reporting interval. A virtual machine with more than one virtual CPU defined can absorb more CPU time than the duration of the reporting period because it can run on several real CPUs concurrently. This will be reflected in a CPU Usage Rate of greater than 1. The first sample report illustrates this case.

```
                                        VM User Consumption Reported
                                           CA  Production Complex

 INQUIRY:  VMXTXC                                                                                                RUN DATE:  14DECyy
                              System: VMXA Date:   THURSDAY, DECEMBER 13, yyyy Userid: MVS08
-------------------------------------------------------------------------------------------------------------------------------
                   C P U    U s a g e              WSS Pages          Virtual I/O         S p o o l            P a g e
                 ----------------------          ------------ ------------------      -----------------     ----------------
   Time    Count     Total    Virt   Rate             Avg      Max       Total   Rate       Reads    Writes      Reads    Writes
-------------------------------------------------------------------------------------------------------------------------------
  9:00:00    15     768.10  643.65  0.853               0        0    178,482    198           0         0          0         0
  9:15:00    15     870.01  746.64  0.967               0        0    177,910    198           0         0          0         0
  9:30:00    15     835.97  714.32  0.929               0        0    171,897    191           0         0          0         0
  9:45:00    15     849.65  752.05  0.944               0        0    145,845    162           0         0          0         0
 10:00:00    15     954.25  830.14  1.060               0        0    179,900    200           0         0          0         0
 10:15:00    15    1257.32 1064.64  1.397               0        0    301,663    335           0         0          0         0
 10:30:00    15    1213.69 1042.87  1.349               0        0    264,515    294           0         0          0         0
 10:45:00    15    1262.80 1085.79  1.403               0        0    277,434    308           0         0          0         0
 11:00:00    15    1182.99 1006.48  1.314               0        0    236,061    262           0         0          0         0
-------------------------------------------------------------------------------------------------------------------------------
```



VM User Consumption Reported
CA Production Complex

INQUIRY: VMXTXC RUN DATE: 14DECyy
System: VMXA Date: THURSDAY, DECEMBER 13, yyyy Userid: NA1LH13

		CPU Usage			WSS Pages		Virtual I/O		Spool		Page	
Time	Count	Total	Virt	Rate	Avg	Max	Total	Rate	Reads	Writes	Reads	Writes
9:00:00	1	0.00	0.00	0.000	70	70	0	0	0	0	0	0
9:15:00	15	135.01	133.17	0.150	319	551	2,453	3	29	0	0	0
9:30:00	15	166.99	165.14	0.186	465	662	1,997	2	8	0	0	0
9:45:00	15	179.99	177.75	0.200	402	655	1,857	2	4	0	0	0
10:00:00	15	233.75	231.21	0.260	460	569	2,244	2	20	0	0	0
10:15:00	15	137.26	135.39	0.153	446	575	1,303	1	11	0	0	0
10:30:00	15	114.44	113.02	0.127	377	518	1,109	1	9	0	0	0
10:45:00	15	125.26	123.89	0.139	511	528	670	1	0	0	0	0

Figure 3-11. Detail VMX User Consumption Report (by User)

3.2.1.1.10 Detail VMX LPAR Utilization Report

This report produces a tabular listing of the processor resource consumption for logical partitions defined in your PR/SM complex for a user-defined interval.

Run-time execution parameters for this report include:

- the database unit identifier or identifiers on which you want to report
- the range of file cycles on which you want to report (both online and archive cycles are available)
- the SYSID of the system on which you want to report
- the length of the interval period on which you want to report
- the period of time represented in the report

INTENDED USE:

Enables you to review CPU utilization for each LPAR defined in your PR/SM complex. Includes the configuration of your logical partitions.

FORMAT:

A tabular listing that contains one line of information for each logical partition defined in the interval. The following columns are shown on the report:

INQUIRY - The MICF inquiry identifier.

SYSID - The name of the system from which the data was collected.

DATE - The date when the data was captured in the CA MICS database.

RUN DATE - The date when the inquiry was submitted for execution.

END TIME - The end of the reporting interval.

PR/SM CPU TOT - The number of physical processors in the PR/SM complex, or the number of physical

processors in the physical partition if the system is running in physically partitioned mode.

SHR - The number of physical processors that are not dedicated to any logical partitions.

LOGICAL PARTITION
NO. - The PR/SM partition number. When there is an asterisk (*) in this column, this line of information represents LPAR management overhead that cannot be attributed to any of the logical partitions.

NAME - The name of the logical partition. In addition to the LPARs you define, CA MICS creates the LPAR named PHYSICAL to report the LPAR management time spent managing the physical configuration in the complex (available under VM/ESA Release 2 and above).

FLAGS
O (Owner) - If an "X" appears in this column, the logical partition is the one that runs the VM system from which the LPAR data was collected.

S (Shared) - If an "X" appears in this column, the LPAR shares the physical processors with other LPARs. If the column is blank, the physical processors are dedicated to the LPAR.

W (Wait Completion=On) - If an "X" appears in this column, the LPAR receives service from the shared physical processors based on the passage of a time interval (as opposed to receiving service based on the occurrence of an event).

C (Capped) - If an "X" appears in this column, the PR/SM capping function is enabled. The LPAR receives only up to its share of physical processor resources based on its processing weight.

LCPU CNT - The number of logical processors.

INTVLS - The number of monitor sample intervals.

WGHTS - The portion of shared processor resources allocated to this LPAR. Valid values range from 1 through 999.

DISPATCH TIME
TOTAL - The total physical processor time accumulated by this LPAR during the reporting interval. If WAIT COMPLETION=On, total dispatch time includes wait time.

AVERAGE - the average amount of physical processor time accumulated by this LPAR in a monitor interval.

% REAL CPU
AVG - The average percentage of the physical processor resources that were assigned to this LPAR during the reporting interval.

MIN - The minimum percentage of the physical processor resources that were assigned to this LPAR during the reporting interval.

MAX - The maximum percentage of the physical processor resources that were assigned to this LPAR during the reporting interval.

LPAR MANAGEMENT OVERHEAD
TOTAL - The amount of time used to manage this LPAR during the reporting interval. For the LPAR named PHYSICAL, this is the time spent managing the physical configuration of the processor complex.

% - The percentage of physical processor time that was spent managing this LPAR as opposed to doing work in the LPAR. For the LPAR named PHYSICAL, this is the percentage of time spent managing the physical configuration of the processor complex.

An example of this report is shown in Figure 3-12.

```
INQUIRY ID:

   Tabular Report:  VMXTXD

DATA SOURCE (File/Timespan):

   VMXLPR01 - DETAIL

DATA ELEMENTS USED:

   ENDTS     - End Timestamp
   LPRCAPP   - Intervals with Capping
   LPRCOLCT  - Intervals Running in Owner LPAR
   LPRCPUCT  - Number of Logical CPUs
   LPRLPAR   - Intervals When LPAR is Active
   LPRLPDTM  - LPAR Dispatch Time
   LPRLPMTM  - LPAR Management Time
   LPRPCMTM  - Percent LPAR Management Overhead
   LPRPCSSU  - LPAR Real CPU Utilization
   LPRPPWGT  - LPAR Processing Weight
   LPRSHRMD  - Intervals in Shared Mode
   LPRWAITF  - Intervals with Wait Completion On
   PRSMLPNM  - Logical Partition Name
   PRSMLPNO  - Logical Partition Number
   PRSMNOPP  - Total Physical Processors in Complex
   SYSID     - System Identifier
   VMCINTTS  - VM Interval Time Stamp


CALCULATIONS:

   The Flags in the report are derived as follows:

        O: the owner flag (OWNER) is set to "X" if the value
           of LPRCOLCT is greater than 0.

        S: the shared flag (SHRD) is set to "X" if LPRSHRMD
           is greater than 0.

        W: the wait completion flag (WAITF) is set to "X" if
           the value of LPRWAITF is greater than 0.

        C: the capping flag (CAPP) is set to "X" if the value
           of LPRCAPP is greater than 0.

   The number of dedicated physical processors (DCPU) is
   computed by summarizing the value of LPRCPUCT for the
   partition whose LPRSHRMD contains the value of 0.
```

The total number of shareable physical processors (RSCPU) is computed as:

RSCPU = PRSMNOPP-DCPU

The average LPAR dispatch time (AVGLPDTM) is computed as:

AVGLPDTM = LPRLPDTM/LPRLPAR;

VMX LPAR Utilization Report
CA I/S MANAGEMENT APPLICATIONS DEVELOPMENT

INQUIRY: VMXTXD RUN DATE: 31MARyy

SYSID = LGNT Date = 02/10/yy

End Time	PR/SM CPU Tot	PR/SM CPU Shr	Logical Partition No.	Name	Flags O	S	W	C	LCPU Cnt	Intvls	Wghts	Dispatch Time Total	Dispatch Time Average	%REAL CPU Avg	%REAL CPU Min	%REAL CPU Max	LPAR Management Overhead Total	LPAR Management Overhead %
12:25	6	6	1	SYD		X			6	1	400	0:04:37.36	0:04:37.36	77.23	77.23	77.23	0:00:02.27	0.63
			2	TEST		X			0	0	400	.	.	.	.	.	.	.
			3	VMB	X	X			6	1	599	0:01:15.84	0:01:15.84	21.12	21.12	21.12	0:00:04.44	1.24
			*	PHYSICAL						1							0:00:03.32	0.92
12:30	6	6	1	SYD		X			6	5	400	0:22:34.54	0:04:30.91	75.25	65.72	81.68	0:00:11.38	0.63
			2	TEST		X			0	0	400	.	.	.	.	.	.	.
			3	VMB	X	X			6	5	599	0:06:55.74	0:01:23.15	23.09	17.20	32.59	0:00:18.18	1.01
			*	PHYSICAL						5							0:00:15.15	0.84
12:35	6	6	1	SYD		X			6	3	400	0:22:30.34	0:07:30.11	75.01	71.99	78.21	0:00:14.19	0.79
			2	TEST		X			0	0	400	.	.	.	.	.	.	.
			3	VMB	X	X			6	3	599	0:06:43.69	0:02:14.56	22.42	19.67	24.53	0:00:22.60	1.26
			*	PHYSICAL						3							0:00:18.76	1.04

Notes: O - Owner S - Shared W - Wait Completion C - Capped
* - LPAR Management Overhead that Cannot be Attributed to Any Logical Partitions

Figure 3-12. Detail VMX LPAR Utilization Report

3.2.1.2 VM Performance Analysis Inquiries

Performance Analysis inquiries quantify key measurements that impact system performance, including channel activity, cache usage, and paging activity. The inquiries in this category are available for user-defined reporting periods specified as execution-time parameters. For example, it might be desirable to produce a report on CP-owned Volume Activity covering the period from 1:30 p.m to 2:45 p.m. The MICF inquiry will select the detail data covering the period of time requested, summarize the information, and then produce a report. The performance analysis inquiries are explained in greater detail in the following sections:

1 - Detail VMX Concurrent Channel Busy
2 - Detail VMX Percent Channel Busy
3 - Detail VMX DASD Cache Usage Report
4 - Detail VMX CP Owned Volume Summary
5 - Detail VMX Device Utilization Report (By Device)
6 - Detail VMX DASD Cache Usage Report (by Device)
7 - VM Channel Path Activity Report
8 - VM Channel Path Load Analysis
9 - VM FICON Channel Path Load Analysis
10- VM HiperSocket Channel Activity Report
11- Shared Address Space Analysis Report

3.2.1.2.1 Detail VMX Concurrent Channel Busy

This report produces a graph of concurrently busy channels during a user-defined reporting period. Each bar of the graph represents the number of times during the reporting period that n number of channels were concurrently busy, where n represents the actual number of channels.

INTENDED USE:

Allows analysis of whether the I/O workload is being spread over the channel subsystem. This report might be used to identify periods of poor system performance due to the system's inability to use more than one or two channels at a time.

FORMAT:

A horizontal bar chart of concurrent channel activity. Each bar represents the number of times during the reporting period that the channel subsystem was sampled, and the associated number of channels that were concurrently active. Only non-zero states are shown.

An example of this report is shown in Figure 3-13.

```
INQUIRY ID:

   Color Graphics:  VMXCX1
   Printer Graphics:  VMXPX1

DATA SOURCE (File/Timespan):

   VMXSGP01 - DETAIL

DATA ELEMENTS USED:

   SGPCBZxx - CHPIDs Simultaneously Busy

 CALCULATIONS:

   The information is summarized across the user specified
   time period, and a separate record is output for each non-
   zero value of SGPCBZxx.  Data element CHNSBUSY is created
   from the value of xx in SGPCBZxx, and element BUSYCNT
   contains the corresponding value of SGPCBZxx.

USAGE CONSIDERATIONS:

   The Channel Path Contention Data File (VMXCHC) is created
   from the 0/09 monitor records.  With the introduction of
   z990 processor, the 0/09 no longer contains valid channel
   path information.  It is recommended that the 0/18 or 0/20
   monitor records be used for valid channel path
   information.

   See report VMXLS1 (VM Channel Path Load Analysis).
```

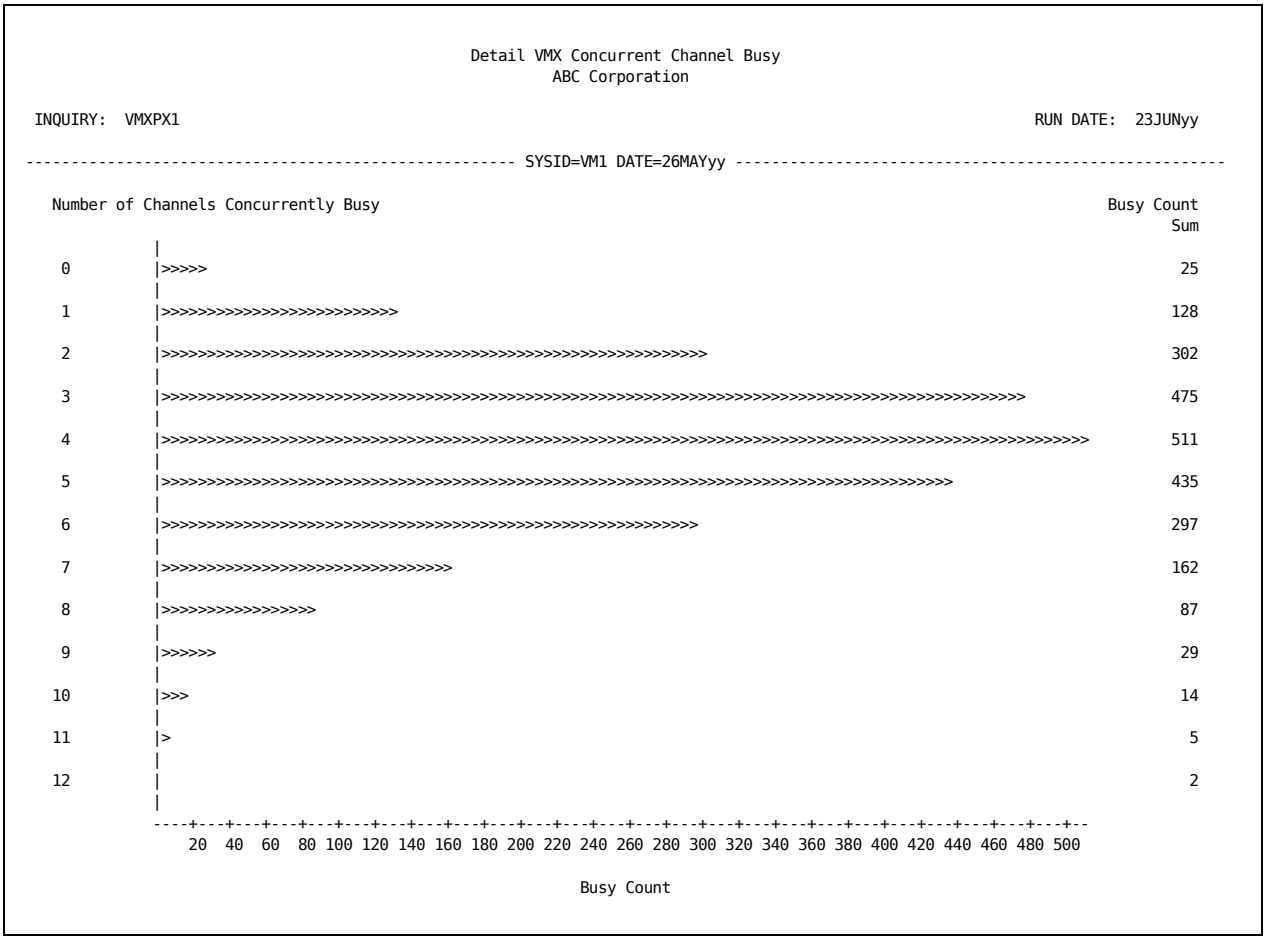


Figure 3-13. Detail VMX Concurrent Channel Busy

3.2.1.2.2 Detail VMX Percent Channel Busy

This report produces a graph of the average percentage of channels busy during a user-defined reporting period. Each bar of the graph represents the percentage of channels busy for a given channel path ID.

INTENDED USE:

Allows analysis of the channel subsystem for purposes of performance tuning and capacity control. This report might be used to identify periods of poor system performance due to an overloaded channel, or might be monitored over time to see how channel workload might be shifting.

FORMAT:

A horizontal bar chart of the percentages of channels busy for each CHPID (Channel Path ID). Only channel paths that had some activity are displayed on the graph.

An example of this report is shown in Figure 3-14.

INQUIRY ID:

Color Graphics: VMXCX2
Printer Graphics: VMXPX2

DATA SOURCE (File/Timespan):

VMXCHC01 - DETAIL

DATA ELEMENTS USED:

CHPID - Channel Path ID
CHCPCBSY - Percent CHPID Busy

CALCULATIONS:

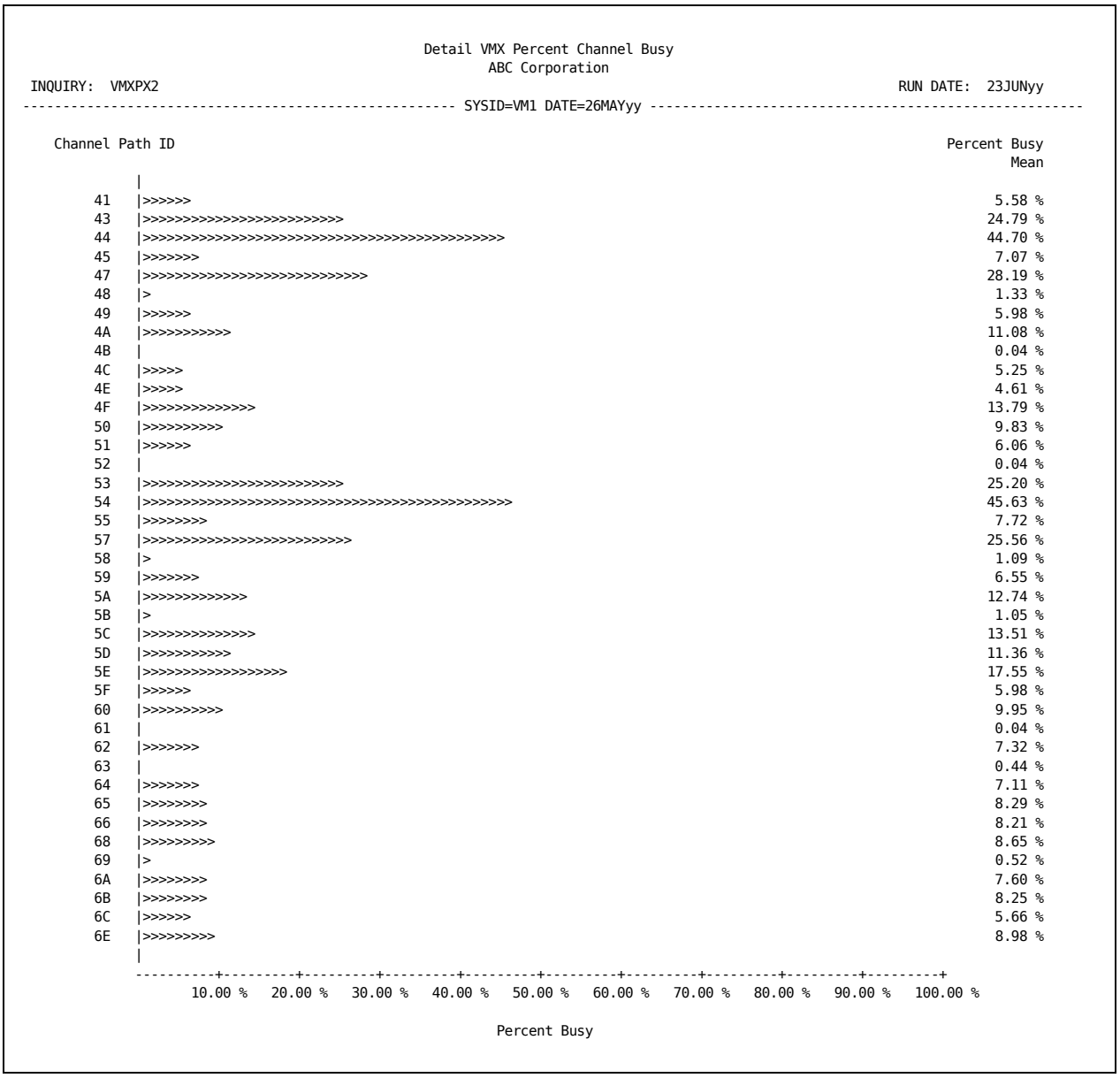
The information is summarized across the user-specified time period, and only active CHPIDs are selected. No other calculations are performed.

USAGE CONSIDERATIONS:

The Channel Path Contention Data File (VMXCHC) is created from the 0/09 monitor records. With the introduction of

z990 processor, the 0/09 no longer contains valid channel path information. It is recommended that the 0/18 or 0/20 monitor records be used for valid channel path information.

See report VMXLS1 (VM Channel Path Load Analysis).



```
                                        Detail VMX Percent Channel Busy
                                                ABC Corporation
 INQUIRY:  VMXPX2                                                                                                RUN DATE:  23JUNyy
------------------------------------------------------ SYSID=VM1 DATE=26MAYyy ------------------------------------------------------

    Channel Path ID                                                                                                  Percent Busy
                                                                                                                            Mean
            |
        41  |>>>>>>                                                                                                      5.58 %
        43  |>>>>>>>>>>>>>>>>>>>>>>>>>                                                                                  24.79 %
        44  |>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>  44.70 %
        45  |>>>>>>>                                                                                                     7.07 %
        47  |>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>  28.19 %
        48  |>                                                                                                         1.33 %
        49  |>>>>>>                                                                                                     5.98 %
        4A  |>>>>>>>>>>>                                                                                               11.08 %
        4B  |                                                                                                          0.04 %
        4C  |>>>>>                                                                                                     5.25 %
        4E  |>>>>>                                                                                                     4.61 %
        4F  |>>>>>>>>>>>>>>>                                                                                           13.79 %
        50  |>>>>>>>>>>                                                                                                9.83 %
        51  |>>>>>>                                                                                                    6.06 %
        52  |                                                                                                          0.04 %
        53  |>>>>>>>>>>>>>>>>>>>>>>>>>>                                                                                25.20 %
        54  |>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>  45.63 %
        55  |>>>>>>>>                                                                                                   7.72 %
        57  |>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>  25.56 %
        58  |>                                                                                                         1.09 %
        59  |>>>>>>>                                                                                                    6.55 %
        5A  |>>>>>>>>>>>>>>                                                                                            12.74 %
        5B  |>                                                                                                         1.05 %
        5C  |>>>>>>>>>>>>>>>                                                                                           13.51 %
        5D  |>>>>>>>>>>>                                                                                               11.36 %
        5E  |>>>>>>>>>>>>>>>>>>>                                                                                      17.55 %
        5F  |>>>>>>                                                                                                    5.98 %
        60  |>>>>>>>>>>                                                                                               9.95 %
        61  |                                                                                                          0.04 %
        62  |>>>>>>>                                                                                                   7.32 %
        63  |                                                                                                          0.44 %
        64  |>>>>>>>                                                                                                   7.11 %
        65  |>>>>>>>>                                                                                                  8.29 %
        66  |>>>>>>>>                                                                                                  8.21 %
        68  |>>>>>>>>>                                                                                                 8.65 %
        69  |>                                                                                                         0.52 %
        6A  |>>>>>>>>                                                                                                  7.60 %
        6B  |>>>>>>>>                                                                                                  8.25 %
        6C  |>>>>>>                                                                                                    5.66 %
        6E  |>>>>>>>>>                                                                                                 8.98 %
            |
            ----------+---------+---------+---------+---------+---------+---------+---------+---------+---------+
                   10.00 %   20.00 %   30.00 %   40.00 %   50.00 %   60.00 %   70.00 %   80.00 %   90.00 %   100.00 %

                                                    Percent Busy
```

Figure 3-14. Detail VMX Percent Channel Busy

3.2.1.2.3 Detail VMX DASD Cache Usage Report

This report produces a tabular listing of cache usage on 3880-23 and 3990-3 control units by VOLSER for the duration of the period specified in the run-time execution parameters. The report is in two formats: one for 3880-23 controllers, and one for 3990-3 controllers. These are identified by the report titles.

INTENDED USE:

For analysis of cache usage in terms of how much activity is directed to cache VOLSERs; whether transfers to and from DASD are being avoided by using cache; and what percentage of bytes read are from cache, from DASD bypassing cache, and from DASD due to read misses. This report can be used to detect when cache is being utilized inefficiently, and for monitoring changes to the cache environment.

FORMAT 3880-23 Control Units:

A tabular report in ascending order of device address where each column represents a different measurement associated with a given VOLSER:

READ HIT RATIO - The ratio of the sum of all "search and read normal" and "search and read sequential" requests that were satisfied from cache to the total number of read requests to the volume.

WRITE/READ RATIO - The ratio of the sum of all "write normal" and "write sequential" requests to the sum of all "search and read normal" and "search and read sequential" requests.

READ REQUESTS - The sum of all "search and read normal" and "search and read sequential" requests to the volume.

WRITE REQUESTS - the sum of all "write normal" and "write sequential" requests.

PCT OF DASD READ CACHE - The percentage of all bytes transferred from this VOLSER that were transferred from cache to the channel without DASD access.

PCT OF DASD READ DASD - The percentage of all bytes transferred from this VOLSER that were transferred

from DASD to cache to the channel due to read misses.

PCT OF DASD READ BYPASS - The percentage of all bytes transferred from this VOLSER that were transferred from DASD to the channel, bypassing cache access.

FORMAT 3990-3 Control Units:

A tabular report in ascending order of device address where each column represents a different measurement associated with a given VOLSER:

OVERALL READ HIT RATIO - The ratio of the sum of all basic and extended function read requests that were satisfied from cache to the total number of read requests for the volume.

OVERALL WRITE/READ RATIO - The ratio of the sum of all write requests to the sum of all read requests for the volume.

BASIC READ HIT RATIO - The ratio of basic read (search and read normal, and search and read sequential) requests satisfied from cache to basic read requests for the volume.

BASIC READ WRITE RATIO - The ratio of basic read requests and basic write requests for the volume.

DASD F/W HIT RATIO - The ratio of DASD Fast Write requests that replaced a record in cache to the total of all DASD Fast Write requests for the volume.

DASD F/W OVERFLOW RATE - The rate at which DASD Fast Write requests had to be directed to DASD because cache was not available due to capacity constraints.

CACHE F/W READ HIT RATIO - The ratio of Cache Fast Write read requests satisfied from cache to the number of Cache Fast Write read requests made.

CACHE F/W WRITE HIT RATIO - The ratio of Cache Fast Write write requests satisfied from cache to the number of Cache Fast Write write requests made.

An example of the report for FORMAT 3880-23 Control Units is shown in Figure 3-15.

```
   An example of the report for FORMAT 3990-3 Control Units
   is shown in Figure 3-16.

INQUIRY ID:

   Tabular Report:  VMXTX1

DATA SOURCE (File/Timespan):

   VMXCDI01 - DETAIL

DATA ELEMENTS USED:

   VMXDVSID - Host Subchannel ID
   CDISRNRQ - Search and Read Normal Requests
   CDISRSRQ - Search and Read Sequential Requests
   CDISRNRH - Search and Read Normal Request Hits
   CDISRSRH - Search and Read Sequential Request Hits
   CDIWNRQ  - Write Normal Requests
   CDIWSQRQ - Write Sequential Requests
   CDIDACBX - Bytes Transferred From DASD To Channel
   CDICACBX - Bytes Transferred From Cache To Channel
   CDIDCCBX - Bytes Xfr From DASD To Cache and Channel
   CDIDFWNO - DASD Fast Write Normal Requests
   CDIDFWSO - DASD Fast Write Sequential Requests
   CDIDFWRH - DASD Fast Write Normal Request Hits
   CDIFWSRH - DASD Fast Write Sequential Request Hits
   CDISRCRQ - Cache Fast Write Read Requests
   CDISRCRH - Cache Fast Write Read Request Hits
   CDIDFWRH - DASD Fast Write Normal Request Hits
   CDIFWSRH - DASD Fast Write Sequential Request Hits

CALCULATIONS:

   A number of elements are calculated in arriving at this
   report, as follows:

   READREQ = CDISRNRQ + CDISRSRQ;

   WRITEREQ = CDIWNRQ + CDIWSQRQ;

   IF READREQ > 0 THEN DO;
      RHRATIO = 100 * ((CDISRNRH + CDISRSRH) / READREQ);
      WRRATIO = 100 * (WRITEREQ / READREQ);
   END;
   ELSE DO;
      RHRATIO = 0;
      WRRATIO = 0;
   END;
```

```
   BYTEXFRD = SUM(CDIDACBX,CDICACBX,CDIDCCBX);
   IF BYTEXFRD > 0 THEN DO;
      CACHE  = 100 * (CDICACBX / BYTEXFRD);
      DASD   = 100 * (CDIDCCBX / BYTEXFRD);
      BYPASS = 100 * (CDIDACBX / BYTESFRD);
   END;
   ELSE DO;
      CACHE = 0; DASD = 0; BYPASS = 0;
   END;
```

USAGE CONSIDERATIONS: None

```
                                        Detail VM Cache Usage Report
                                          CA  Production Complex

 INQUIRY:  VMXTX1                                                                                                RUN DATE:  14DECyy
                          System: VMXA Date: 13DECyy Control Unit Type: 3880-23  Time: 12:00
-------------------------------------------------------------------------------------------------------------------------------------
                                      Read  Write                        Percent Of Read I/Os
                                Device   Hit   Read       Read      Write -------------------
                             VOLSER Number  Ratio  Ratio   Requests  Requests  Cache   DASD Bypass
-------------------------------------------------------------------------------------------------------------------------------------
                             MVSC1A  0977   97.79  87.99      6,605     5,812  98.30   1.67   0.02
                             MVSDL1  0976   71.43   0.00         28         0  68.38  30.78   0.84
                             MVS006  0960   95.48   0.58      9,588        56  94.40   5.49   0.11
                             MVS007  0962   98.44  10.74     26,484     2,845  99.92   0.07   0.00
                             MVS042  0964   84.54   2.06     22,767       468  81.89  13.65   4.46
                             MVS043  0975   89.72   1.72     17,828       307  79.51  17.86   2.63
                             MVS961  0961   86.16   7.56      4,284       324  75.30  21.64   3.06
                             MVS963  0963   96.57  11.97     95,654    11,453  94.49   5.35   0.16
                             MVS965  0965   91.29  83.71      1,136       951  75.29  20.12   4.59
                             MVS966  0966   85.33   7.33      1,500       110  83.29  14.78   1.93
                             MVS967  0967   86.93  27.37      8,136     2,227  87.07  12.01   0.91
                             MVS970  0970   95.71   0.03      6,670         2  77.96  21.26   0.78
                             MVS971  0971   50.35   1.23      5,295        65  88.20  10.83   0.97
                             MVS972  0972   92.00  39.74      9,925     3,944  89.74  10.08   0.18
                             MVS973  0973   95.13   0.83     58,255       486  91.04   8.71   0.25
                             MXARS1  0974   75.88   0.00        879         0  65.93  31.21   2.86
-------------------------------------------------------------------------------------------------------------------------------------
```

Figure 3-15. Detail VM 3880-23 Cache Usage Report

```
-------------------------------------------------------------------------------------------------------------------------------
                                        Detail VM Cache Usage Report
                                           CA  Production Complex

 INQUIRY:  VMXTX1                                                                                    RUN DATE:  14DECyy
                        System: VMXA Date: 13DECyy Control Unit Type: 3990-3   Time: 12:00
-------------------------------------------------------------------------------------------------------------------------------
                             <---  Overall ---><----  Basic ---> <-- DASD F/W  --> <-- Cache F/W -->
                    Device Read Hit  R/Write Read Hit  R/Write Read Hit Overflow Read Hit Write Hit
              VOLSER  Number   Ratio    Ratio    Ratio    Ratio    Ratio    Rate     Ratio    Ratio
-------------------------------------------------------------------------------------------------------------------------------
              HSM008   08C8    73.15     0.00    73.15     0.00     0.00     0.00     0.00     0.00
              LRG009   08C9    99.66   116.10    99.66   116.10     0.00     0.00     0.00     0.00
              LRG010   08CA    92.49     5.38    92.49     5.38     0.00     0.00     0.00     0.00
              LRG011   08CB    76.77    22.63    76.77    22.63     0.00     0.00     0.00     0.00
              LRG012   08CC    88.46    13.53    88.46    13.53     0.00     0.00     0.00     0.00
              LRG013   08CD    87.28    12.03    87.28    12.03     0.00     0.00     0.00     0.00
              LRG014   08DF    92.98     3.54    92.98     3.54     0.00     0.00     0.00     0.00
              LRG015   08DE    82.71    11.74    82.71    11.74     0.00     0.00     0.00     0.00
              SYS069   08FE     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00
              TMP072   08DD    94.59     1.10    94.59     1.10     0.00     0.00     0.00     0.00
              TSU020   08CE    84.19    30.27    84.19    30.27     0.00     0.00     0.00     0.00
              TSU021   08CF    88.98     4.00    88.98     4.00     0.00     0.00     0.00     0.00
              XADMP1   08EE    83.71     0.33    83.71     0.33     0.00     0.00     0.00     0.00
              XAP002   08E0    64.41     0.21    64.41     0.21     0.00     0.00     0.00     0.00
              XAP003   08E1    29.23     0.30    29.23     0.30     0.00     0.00     0.00     0.00


-------------------------------------------------------------------------------------------------------------------------------
```

Figure 3-16. Detail VMX 3990-3 Cache Usage Report

3.2.1.2.4 Detail VMX CP Owned Volume Summary

This report produces a tabular listing of activity on CP-owned VOLSERs for the duration of the period specified in the run-time execution parameters. It includes those volumes that are owned by this system, and those that are shared by multiple systems in a Cross System Extensions (CSE) complex, if CSE is in use. In VM, it additionally includes those volumes that contain mapped minidisks.

INTENDED USE:

Allows analysis of page and spool activity on CP-owned volumes, including such items as average and maximum slots used, I/O activity, I/O queuing measures, and interference from user I/O. This report can be used to detect when page or spools spaces are approaching capacity, and bottlenecks due to queuing of I/O or interference from I/O activity for non-system (user) activity.

FORMAT:

A tabular report in VOLSER order where each column represents a different measurement associated with a given VOLSER:

PAGE SLOTS USED AVG - The average number of page slots in use on all paging areas defined for this volume during the reporting period

PAGE SLOTS USED % - The percentage of available page slots on the volume that the average used figure above represents

PAGE SLOTS USED MAX - The maximum number of page slots that were in use on all paging areas defined for this volume during the reporting period

PAGE SLOTS USED % - The percentage of available page slots on the volume that the maximum used figure above represents

PAGE SLOTS USED TIME - The time of day when the maximum used figure occurred

PAGE I/O READS - The total number of paging read operations to this VOLSER during the reporting period

PAGE I/O WRITES - The total number of paging write operations to this VOLSER during the reporting period

SPOOL SLOTS USED AVG - The average number of spool slots in use on all spool areas defined for this volume during the reporting period

SPOOL SLOTS USED % - The percentage of available spool slots on the volume that the average used figure above represents

SPOOL SLOTS USED MAX - The maximum number of spool slots that were in use on all spool areas defined for this volume during the reporting period

SPOOL SLOTS USED % - The percentage of available spool slots on the volume that the maximum used figure above represents

SPOOL SLOTS USED TIME - The time of day when the maximum used figure occurred

SPOOL I/O READS - The total number of spool read operations to this VOLSER during the reporting period

SPOOL I/O WRITES - The total number of spool write operations to this VOLSER during the reporting period

I/O QUEUED AVG - The average number of page/spool I/Os queued to this VOLSER during the reporting period

I/O QUEUED MAX - The maximum number of page/spool I/Os queued to this VOLSER during the reporting period

I/O QUEUED TIME - The time of day when the maximum queued figure occurred

USER I/O INTERFERENCE AVG - The average number of times per monitor interval that user I/O interfered with paging and/or spooling during the reporting period

USER I/O INTERFERENCE MAX - The maximum number of times during one monitor interval that user I/O interfered with paging and/or spooling during the reporting period

USER I/O INTERFERENCE TIME - The time of day when the maximum interference figure occurred

An example of this report is provided in Figure 3-17.

INQUIRY ID:

Tabular Report: VMXTX3

DATA SOURCE (File/Timespan):

VMXCPV01 - DETAIL

DATA ELEMENTS USED:

VOLSER - Volume Serial Number
CPVPAGSL - Page Slots in Use on This Volume
CPVPCPSU - Percent Spool Slots Used
CPVPSLOT - Total Page Slots on This Volume
CPVPAGRD - Total Page Reads
CPVPAGWR - Total Page Writes
CPVSPLSL - Spool Slots in Use on This Volume
CPVPCSSU - Percent Page Slots Used
CPVSSLOT - Total Spool Slots on This Volume
CPVSPLRD - Total Spool Reads
CPVSPLWR - Total Spool Writes
CPVIOQCT - Page/Spool I/Os Queued or Active
CPVUSRIO - User I/O Interference Count
CPVPAGAR - Number of Page Areas on Volume
CPVSPLAR - Number of Spool Areas on Volume
CPVSHRD - CSE Shared Volume Indicator
CPVDUMPV - Dump Only Volume Indicator

```
CALCULATIONS:

   As a first step, the maximum values and timestamps are
   pulled out of the detail data, and put into a separate
   file.  As part of this process, the following calculations
   are performed for maximum percentage of page/spool used:

        MAXPCPAG = 100 * MAXPAGSL / MAXPSLOT;
          where MAXPAGSL is the maximum value of CPVPAGSL
          encountered, and MAXPSLOT was the value of CPVPSLOT
          at the time

        MAXPCSPL = 100 * MAXSPLSL / MAXSSLOT;
          where MAXSPLSL is the maximum value of CPVSPLSL
          encountered, and MAXSSLOT was the value of CPVSSLOT
          at the time

   The detail data is the read again, and summarized into
   another file.  Then the following calculations are
   performed:

        AVGPAGSL = CPVPAGSL / INTERVLS;

        AVGSPLSL = CPVSPLSL / INTERVLS;

        AVGIOQCT = CPVIOQCT / INTERVLS;

        AVGUSRIO = CPVUSRIO / INTERVLS;
```

USAGE CONSIDERATIONS:

None

```
                                        Detail VM CP Owned Volume Summary
                                        Product Manufacturing Corporation

INQUIRY:  VMXTX3                                                
```

3.2.1.2.5 Detail VMX Device Utilization Report (by device)

This report produces a tabular listing of activity for selected devices for the duration of the period specified in the run-time execution parameters. The devices reported are selectable using run-time parameters based on device number, VOLSER, device class, or device type. The selection criteria are combined to produce "or" logic so that if device number 3FA is requested and device type 3380 is requested, all 3380 devices will be reported and device number 3FA will be reported whether it is a 3380 or not.

INTENDED USE:

Allows analysis of device activity over time, including such items as I/O activity, I/O queuing measures, and minidisks allocated. This report can be used to detect anomalies in the activity on a device that might be a cause of poor system response time.

FORMAT:

A tabular report in TIME order where each row represents a different measurement interval for the device:

I/O COUNT - The number of I/O operations performed by the device over the reporting interval.

%BUSY - The average utilization for the device over the interval.

CONNECT - The device connect time (the time that the device was connected to the channel to transfer data) expressed in seconds.

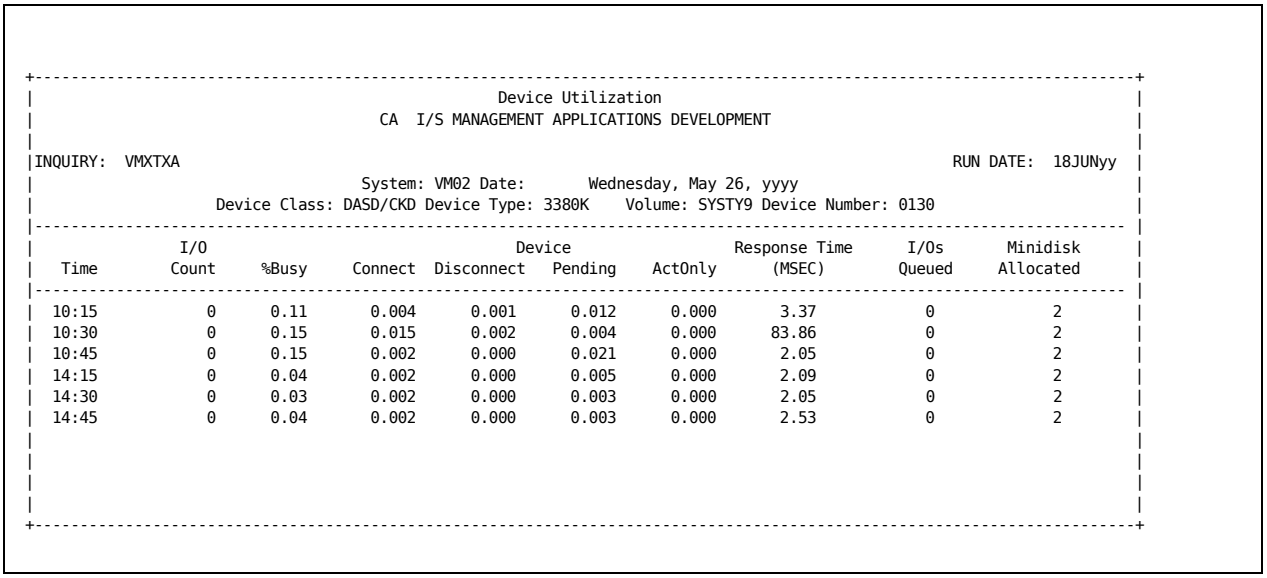
DISCONNECT - The device disconnect time (the time that the device was disconnected from the channel to prepare for data transfer) expressed in seconds.

PENDING - the device function pending time (the time between the initiation of a start operation and the execution of the first command in the channel program) expressed in seconds.

ACTONLY - the device-active-only time (the time that the subchannel is device-active but not subchannel active at the end of an I/O operation or chain of I/O operations initiated by START SUBCHANNEL) expressed in seconds.

```
        RESPONSE TIME - The average number of milliseconds
        required for an I/O operation to complete.

        I/Os QUEUED - The average number of operations queued
        by VM for the device.

        MINIDISKS ALLOCATED - the number of minidisks that
        are allocated on the device (for DASD).  This is
        equal to the number of minidisks in use on the volume
        plus links to those minidisks.  For example, MAINT
        190 may represent 100 allocations if 100 logged on
        users are linked to that minidisk.


   An example of this report is provided in Figure 3-18.

INQUIRY ID:

   Tabular Report:  VMXTXA

DATA SOURCE (File/Timespan):

   VMXXDV01 - DETAIL

DATA ELEMENTS USED:

   VOLSER   - Volume Serial Number
   VMXDVDEV - Device Number
   DEVTYPE  - Device Type
   DEVCLASS - Device Class
   XDVAVSSC - Average Start Subchannel Count
   XDVPCBSY - Device Percent Busy
   XDVAVDCN - Average Device Connect Time
   XDVAVDDC - Average Device Disconnect Time
   XDVAVFPN - Average Device Function Pending Time
   XDVAVDAO - Average Device-Active-Only Time
   XDVAVRES - Average Device Response Time
   XDVAVQUE - Average I/Os Queued
   XDVAVMDA - Average Minidisks Allocated

CALCULATIONS:

   The detail data is summarized using standard CA MICS
   summarization to the reporting interval specified with the
   run-time parameters.

USAGE CONSIDERATIONS:
```

None



```
                                          Device Utilization
                                  CA  I/S MANAGEMENT APPLICATIONS DEVELOPMENT

INQUIRY:  VMXTXA                                                                                    RUN DATE:  18JUNyy
                                System: VM02 Date:       Wednesday, May 26, yyyy
                    Device Class: DASD/CKD Device Type: 3380K    Volume: SYSTY9 Device Number: 0130
                I/O                                  Device                 Response Time      I/Os       Minidisk
   Time        Count     %Busy     Connect  Disconnect   Pending    ActOnly      (MSEC)        Queued     Allocated
 10:15            0      0.11       0.004      0.001      0.012      0.000       3.37            0             2
 10:30            0      0.15       0.015      0.002      0.004      0.000      83.86            0             2
 10:45            0      0.15       0.002      0.000      0.021      0.000       2.05            0             2
 14:15            0      0.04       0.002      0.000      0.005      0.000       2.09            0             2
 14:30            0      0.03       0.002      0.000      0.003      0.000       2.05            0             2
 14:45            0      0.04       0.002      0.000      0.003      0.000       2.53            0             2
```

Figure 3-18. Detail VMX Device Utilization Report (by device)

3.2.1.2.6 Detail VMX DASD Cache Usage Report (by Device)

This report produces a tabular listing of cache usage on 3880-23 and 3990-3 control units by VOLSER for the duration of the period specified in the run-time execution parameters. The report is in two formats: one for 3880-23 controllers, and one for 3990-3 controllers. These are identified by the report titles.

INTENDED USE:

For analysis of cache usage in terms of how much activity is directed to cache VOLSERs; whether transfers to and from DASD are being avoided by using cache; and what percentage of bytes read are from cache, from DASD bypassing cache, and from DASD due to read misses. This report can be used to detect when cache is being utilized inefficiently, and for monitoring changes to the cache environment.

FORMAT 3880-23 Control Units:

A tabular report, in ascending order, of device address where each column represents a different measurement associated with a given VOLSER:

READ HIT RATIO - The ratio of the sum of all "search and read normal" and "search and read sequential" requests that were satisfied from cache to the total number of read requests to the volume.

WRITE/READ RATIO - The ratio of the sum of all "write normal" and "write sequential" requests to the sum of all "search and read normal" and "search and read sequential" requests.

READ REQUESTS - The sum of all "search and read normal" and "search and read sequential" requests to the volume.

WRITE REQUESTS - the sum of all "write normal" and "write sequential" requests.

PCT OF DASD READ CACHE - The percentage of all bytes transferred from this VOLSER that were transferred from cache to the channel without DASD access.

PCT OF DASD READ DASD - The percentage of all bytes transferred from this VOLSER that were transferred

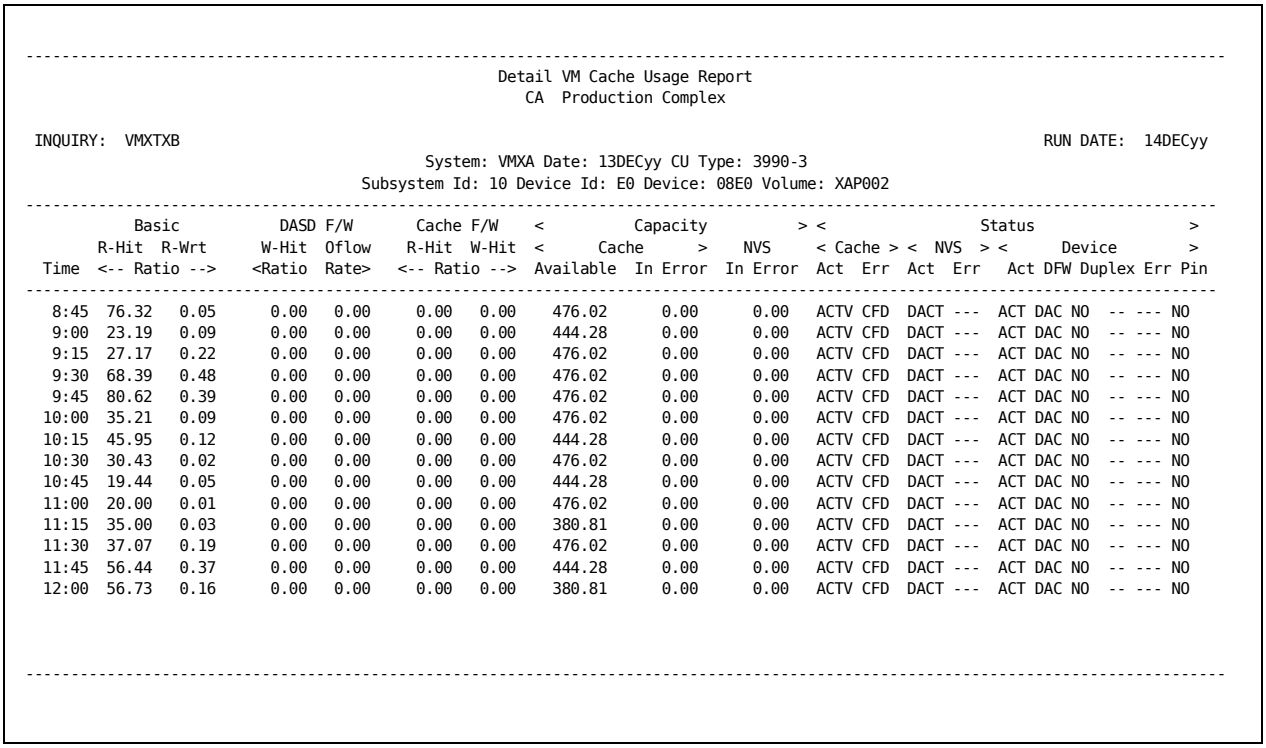
```
        from DASD to cache to the channel due to read misses.

        PCT OF DASD READ BYPASS - The percentage of all bytes
        transferred from this VOLSER that were transferred
        from DASD to the channel, bypassing cache access.


   An example of this report is shown in Figures 3-19 and
   3-20.

INQUIRY ID:

   Tabular Report:  VMXTXB

DATA SOURCE (File/Timespan):

   VMXCDI01 - DETAIL

DATA ELEMENTS USED:

   VMXDVSID - Host Subchannel ID
   CDISDID  - Storage Director Identification
   CDICUTYP - Control Unit Type
   CDIOCSTA - Overall Caching Status
   CDICAPAV - Available Subsystem Storage Capacity
   CDICAPOF - Offline Subsystem Storage Capacity
   CDISRNRQ - Search and Read Normal Requests
   CDISRSRQ - Search and Read Sequential Requests
   CDISRNRH - Search and Read Normal Request Hits
   CDISRSRH - Search and Read Sequential Request Hits
   CDIWNRQ  - Write Normal Requests
   CDIWSQRQ - Write Sequential Requests
   CDIDACBX - Bytes Transferred From DASD To Channel
   CDICACBX - Bytes Transferred From Cache To Channel
   CDIDCCBX - Bytes Xfr From DASD To Cache and Channel

CALCULATIONS:

   A number of elements are calculated to produce this
   report. They are as follows:

   READREQ = CDISRNRQ + CDISRSRQ;

   WRITEREQ = CDIWNRQ + CDIWSQRQ;

   IF READREQ > 0 THEN DO;
      RHRATIO = 100 * ((CDISRNRH + CDISRSRH) / READREQ);
      WRRATIO = 100 * (WRITEREQ / READREQ);
   END;
```

```
   ELSE DO;
      RHRATIO = 0;
      WRRATIO = 0;
   END;

   BYTEXFRD = SUM(CDIDACBX,CDICACBX,CDIDCCBX);
   IF BYTEXFRD > 0 THEN DO;
      CACHE  = 100 * (CDICACBX / BYTEXFRD);
      DASD   = 100 * (CDIDCCBX / BYTEXFRD);
      BYPASS = 100 * (CDIDACBX / BYTESFRD);
   END;
   ELSE DO;
      CACHE = 0; DASD = 0; BYPASS = 0;
   END;
```

USAGE CONSIDERATIONS: None

```
------------------------------------------------------------------------------------------------------------------------
                                          VM Cache Usage Detail Report
                                             CA  Production Complex

 INQUIRY:  VMXTXB                                                                                    RUN DATE:  14DECyy
                                    System: VMXA Date: 13DECyy CU Type: 3880-23
                                 Stg Dir Id: 0E Device Id: 77 Device: 0977 Volume: MVSC1A
------------------------------------------------------------------------------------------------------------------------
                         Read  Write                      Percent Of Read I/Os                  Overall
                          Hit   Read       Read      Write --------------------  Capacity       Cache Stat
                 Time  Ratio  Ratio   Requests   Requests  Cache   DASD Bypass Available   <------------------>
------------------------------------------------------------------------------------------------------------------------
                 8:45  86.27   0.00         51          0  77.83  22.17   0.00   414.87    ACT --- --- CU0 CU1
                 9:00  84.08   1.20        333          4  76.17  23.83   0.00   287.22    ACT --- --- CU0 CU1
                 9:15  94.06  59.58        522        311  93.84   6.12   0.05   319.13    ACT --- --- CU0 CU1
                 9:30  97.44  90.74        821        745  97.85   2.14   0.01   414.87    ACT --- --- CU0 CU1
                 9:45  99.29  98.40        564        555  99.80   0.14   0.06   351.05    ACT --- --- CU0 CU1
                10:00 100.00 100.00        488        488 100.00   0.00   0.00   287.22    ACT --- --- CU0 CU1
                10:15  99.06  92.94        425        395  99.77   0.20   0.02   382.96    ACT --- --- CU0 CU1
                10:30  99.08  93.58        436        408  99.84   0.16   0.00   351.05    ACT --- --- CU0 CU1
                10:45 100.00  95.69        418        400 100.00   0.00   0.00   351.05    ACT --- --- CU0 CU1
                11:00  99.04  97.50        519        506  99.70   0.30   0.00   319.13    ACT --- --- CU0 CU1
                11:15  99.78  99.56        457        455  99.99   0.01   0.00   287.22    ACT --- --- CU0 CU1
                11:30  97.42  90.96        542        493  96.78   3.09   0.13   351.05    ACT --- --- CU0 CU1
                11:45 100.00 106.36        393        418 100.00   0.00   0.00   319.13    ACT --- --- CU0 CU1
                12:00  99.69  99.69        636        634  99.89   0.11   0.00   382.96    ACT --- --- CU0 CU1


------------------------------------------------------------------------------------------------------------------------
```

Figure 3-19. Detail VMX 3880-23 Cache Usage Report (by Device)



```
                                                  Detail VM Cache Usage Report
                                                     CA  Production Complex

 INQUIRY:  VMXTXB                                                                                                                                RUN DATE:  14DECyy
                                                System: VMXA Date: 13DECyy CU Type: 3990-3
                                             Subsystem Id: 10 Device Id: E0 Device: 08E0 Volume: XAP002
-------------------------------------------------------------------------------------------------------------------------------------------------------------
          Basic           DASD F/W       Cache F/W    <          Capacity          > <                Status                >
       R-Hit  R-Wrt      W-Hit  Oflow    R-Hit  W-Hit  <      Cache      >    NVS     < Cache > <  NVS  > <      Device        >
  Time  <-- Ratio -->    <Ratio  Rate>   <-- Ratio -->  Available  In Error  In Error  Act  Err  Act  Err   Act DFW Duplex Err Pin
-------------------------------------------------------------------------------------------------------------------------------------------------------------
   8:45  76.32   0.05      0.00   0.00     0.00   0.00    476.02      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
   9:00  23.19   0.09      0.00   0.00     0.00   0.00    444.28      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
   9:15  27.17   0.22      0.00   0.00     0.00   0.00    476.02      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
   9:30  68.39   0.48      0.00   0.00     0.00   0.00    476.02      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
   9:45  80.62   0.39      0.00   0.00     0.00   0.00    476.02      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
  10:00  35.21   0.09      0.00   0.00     0.00   0.00    476.02      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
  10:15  45.95   0.12      0.00   0.00     0.00   0.00    444.28      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
  10:30  30.43   0.02      0.00   0.00     0.00   0.00    476.02      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
  10:45  19.44   0.05      0.00   0.00     0.00   0.00    444.28      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
  11:00  20.00   0.01      0.00   0.00     0.00   0.00    476.02      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
  11:15  35.00   0.03      0.00   0.00     0.00   0.00    380.81      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
  11:30  37.07   0.19      0.00   0.00     0.00   0.00    476.02      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
  11:45  56.44   0.37      0.00   0.00     0.00   0.00    444.28      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO
  12:00  56.73   0.16      0.00   0.00     0.00   0.00    380.81      0.00      0.00   ACTV CFD  DACT ---  ACT DAC NO  -- --- NO



-------------------------------------------------------------------------------------------------------------------------------------------------------------
```

Figure 3-20. Detail VMX 3990-3 Cache Usage Report (by Device)

3.2.1.2.7 VM Channel Path Activity Report

The VM Channel Path Activity report provides information about channel path usage. Channel paths provide the interface that allows for control units and I/O devices to be physically connected to the I/O subsystem.

INTENDED USE:

The report is designed to help analyze each channel path by path ID. For each ID, the type of path and the percent path usage is reported. If a path is offline, this is reported in place of the percent usage. The report can help detect bottlenecks in the I/O subsystem caused by excessive path utilization.

For systems using HiperSockets, detailed traffic information is provided in a separate section, at the end of the report.

FORMAT:

A tabular listing that contains one line of information for each active channel configured during the reporting period. In addition to the channel path ID, the following columns are shown on the report:

CHANNEL PATH ID - Hexadecimal number of the channel path identifier.

TYPE - Type of channel path.

SHR Y/N - An indication of whether a channel path is defined as shared between one or more logical partitions. YES indicates the channel path is shared.

UTILIZATION(%) PART - The channel path utilization percentage for the individual partition.

UTILIZATION(%) TOTAL - The channel path utilization percentage for the CPC.

UTILIZATION(%) BUS - Percentage of bus cycles, the bus has been found busy for this channel in relation to the theoretical limit.

READ (M/B SEC) PART - Transfer data rates from the control unit to the channel for this partition.

READ (M/B SEC) TOTAL - Transfer data rates from the control unit to the channel for the CPC.

WRITE (M/B SEC) PART - Transfer data rates from the channel to the control unit for this partition.

WRITE (M/B SEC) TOTAL - Transfer data rates from the channel to the control unit for the CPC.

FICON OPERATIONS RATE - FICON operations rate.

FICON OPERATIONS ACTIVE - Active FICON operations rate

FICON OPERATIONS DEFER - Deferred FICON operations rate.

ZHPF OPERATIONS RATE - zHPF operations rate.

ZHPF OPERATIONS ACTIVE - Active zHPF operations rate.

ZHPF OPERATIONS DEFER - Deferred Zhpf operations rate.

An example of this report is shown in Figure 3-21.

INQUIRY ID:

VMXLPA

DATA SOURCE (File/Timespan):

VMXCPM01 - DAYS

DATA ELEMENTS USED:

CHANNEL - Physical Channel Address
CPMTYPE - VM Channel Path Type
CPMCMG - VM Channel Measurement Group
CPMECPMF - VM Ext Channel Data Supported
CPMCPMF - VM Channel Data Enabled
CPMSHRCT - VM Channel Path Shared
CPMPCPBY - VM Percent Channel Busy - LPAR
CPMPCPSY - VM Percent Channel Busy - CPC
CPMPCBUS - VM Percent Bus Busy
CPMPSPR - VM FICON Read MB/Sec - LPAR
CPMPSTR - VM FICON Read MB/Sec - CPC
CPMPSPW - VM FICON Write MB/Sec - LPAR
CPMPSTW - VM FICON Write MB/Sec - CPC

CPMPSHPW - VM HiperSockets Write MB/Sec - LPAR
CPMPSPMS - VM HiperSockets Message Sent Rate - LPAR
CPMPSHTW - VM HiperSockets Write MB/Sec - CPC
CPMPSTMS - VM HiperSockets Message Sent Rate - CPC
CPMHPMSZ - VM HiperSockets Avg Message Size - LPAR
CPMHTMSZ - VM HiperSockets Avg Message Size - CPC
CPMPSPUS - VM HiperSockets Message Failed Rate - LPAR
CPMPSTUB - VM HiperSockets Receive Failed Rate - CP
CPMPSPUB - VM HiperSockets Receive Failed Rate - LPAR

CALCULATIONS:
None

USAGE CONSIDERATIONS:

None

C H A N N E L P A T H A C T I V I T Y R E P O R T

Page nnn|

z/VM System ID SYSA Start 20JULyy:23:59:45.92 Interval 1:00:00.00
Release V5R4 End 21JULyy:00:59:45.92
CPMF: Extended Mode CSSID: 0

DETAILS FOR ALL CHANNELS

Channel Path			Utilization(%)			Read(MB/Sec)		Write(MB/Sec)		Ficon Operations			zHPF Operations			
ID	Type	SHR	Part	Total	Bus	Part	Total	Part	Total	Rate	Active	Defer	Rate	Active	Defer	
05	FCS	Y	32.54	32.54	8.76	35.81	35.81	34.24	34.24	894.03	6.11	0.00	4.76	1.64	0.00	
06	FCS	Y	32.62	32.62	8.80	35.66	35.66	34.76	34.76	894.98	6.08	0.00	4.74	1.63	0.00	
07	FCS	Y	32.66	32.66	8.79	35.62	35.62	34.67	34.67	893.64	6.23	0.00	4.84	1.66	0.00	
08	FCS	Y	32.72	32.72	8.80	35.69	35.69	34.69	34.69	893.89	6.27	0.00	4.83	1.61	0.00	
0B	FCS	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	|
0C	FCS	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	|
10	OSX	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00				
11	OSX	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00				
12	OSX	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00				
13	OSX	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00				
15	FCPTP		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
17	FCPTP		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
19	FCS	Y	0.00	0.30	0.00	0.00	0.00	0.00	0.00	44.03	1.25	0.00	0.00	0.00	0.00	
1A	FC	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00				
1B	FC	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00				
1D	CFP	Y	0.00	0.00												
1E	CFP	Y	0.00	0.00												
24	FCS	Y	0.00	1.11	0.02	0.00	0.01	0.00	0.18	155.45	1.02	0.00	0.00	0.00	0.00	
25	FCS	Y	0.00	1.09	0.02	0.00	0.01	0.00	0.18	155.49	1.02	0.00	0.00	0.00	0.00	
2C	OSE	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00				
30	OSA/3270	Y	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00				
3D	OSD	Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00				
FB	IQD	Y	See Hipersocket Section of Report													
FC	IQD	Y	See Hipersocket Section of Report													
FD	IQD	Y	See Hipersocket Section of Report													
FE	IQD	Y	See Hipersocket Section of Report													

DETAILS FOR HIPERSOCKETS

Channel Path			Write(B/Sec)		Message Rate		Message Size		Send Fail	Receive Fail	
ID	Type	SHR	Part	Total	Part	Total	Part	Total	Part	Part	Total
FB	IQD	Y	0	N/A	0	0	-------	-------	0.00	0.00	0.00
FC	IQD	Y	0	N/A	0	0	-------	-------	0.00	0.00	0.00
FD	IQD	Y	0	N/A	0	0	-------	-------	0.00	0.00	0.00
FE	IQD	Y	0	N/A	0	0	-------	-------	0.00	0.00	0.00

Figure 3-21. VM Channel Path Activity Report

3.2.1.2.8 VM Channel Path Load Analysis

The VM Channel Path Load Analysis report provides information about all active channels. Only channels for which "busy" conditions occur are included. Channel paths provide the interface that allows for control units and I/O devices to be physically connected to the I/O subsystem.

INTENDED USE:

The report is designed to help analyze each channel path by path ID. For each ID, the type of path and the relative frequency with which the channel busy percentage was found in either of 10% ranges of 0-10, 11-20,..., 91-100. The report can help detect bottlenecks in the I/O subsystem caused by excessive path utilization.

Only FICON and ESCON channels are reported.

FORMAT:

A tabular listing that contains one line of information for each active channel configured during the reporting period. Report is ordered by descending %BUSY sequence (highest to lowest) and CHPID. In addition to the channel path ID, the following columns are shown on the report:

CHPID - Hexadecimal number of the channel path identifier.

CHAN-GROUP DESCR - Description of the channel model group as found in the Store-Channel-Path description.

CHAN-GROUP QUAL - The hexadecimal channel model group qualifier.

%BUSY CUR - Current percent channel busy calculated for the last interval of the hour.

%BUSY AVE - Average percent channel busy for the whole period since the last MONITOR RESET.

For FICON channels, the percent busy is calculated as the percentage of actual work unit utilization compared to the maximum possible value.

CHANNEL BUSY & DISTRIBUTION 0-10 thru 91-100 - is the channel load distribution for each active channel. It is the relative frequency with which percent channel

```
        busy was found to be in the range.


   An example of this report is shown in Figure 3-22.

INQUIRY ID:

   VMXLS1

DATA SOURCE (File/Timespan):

   VMXCPM01 - DETAIL

DATA ELEMENTS USED:

   CHANNEL  - Physical Channel Address
   CPMTYPE  - VM Channel Path Type
   CPMCMG   - VM Channel Measurement Group
   CPMCMMQ  - VM Channel Model Group Qualifier
   CPMPCPSY - VM Percent Channel Busy - CPC

 CALCULATIONS:

   For each channel within the hour, PC0_10 thru PC91_100
   fields are accumulated to tell how many times Percent
   Channel Busy - CPC (CPMPCBSY) fell within the specified
   ranges.

       IF CPMPCBSY LE nn THEN PCll_hh+1;

       where ll = low range percent and hh = high range
                  percent  (e.g. PC11_20)

   For the percentage of observations falling into a
   particular range of percent values, e.g. 30-39%, (PCTnn)
   of total CPMPCBSY during the whole period (time from last
   MONITOR RESET to current), the following calculation is
   used:

       PCTnn=(PCll_hh/TOTSAMP*100)

       where PCTnn is the percent range and TOTSAMP is the
       accumulated value of INTERVLS for each active channel
       within the hour

USAGE CONSIDERATIONS:

   None
```

```
                              C H A N N E L   P A T H   L O A D   A N A L Y S I S   R E P O R T                              Page nnn
   z/VM                          System ID SYSA               Start 20JULyy:23:52:45.92
   Release  V5R4                                                End 20JULyy:23:59:45.92
                                                                    CPMF: Extended Mode      CSSID:  0
| CHPID   Chan-Group    <% Busy>   <-------------- Channel % Busy Distributon ------->                                        |
| (Hex)   Descr Qual    Cur  Ave   0-10 11-20 21-30 31-40 41-50 51-60 61-70 71-80 81-90 91-100                               |
|  40     FCS     0      36   38    0     0     0     72    28    0     0     0     0     0                                  |
|  27     FCS     0      34   38    0     0     0     72    28    0     0     0     0     0                                  |
|  2B     FCS     0      22   24    0     14    86    0     0     0     0     0     0     0                                  |
|  4C     FCS     0      22   24    0     14    86    0     0     0     0     0     0     0                                  |
|  31     FCS     0      20   20    0     28    72    0     0     0     0     0     0     0                                  |
|  4F     FCS     0      20   20    0     28    72    0     0     0     0     0     0     0                                  |
|  48     FCS     0      14   14    0     100   0     0     0     0     0     0     0     0                                  |
|  41     FCS     0      12   16    0     72    28    0     0     0     0     0     0     0                                  |
|  43     FCS     0      12   14    0     100   0     0     0     0     0     0     0     0                                  |
|  2D     FCS     0      10    6    100   0     0     0     0     0     0     0     0     0                                  |
|  21     FCS     0      10    8    86    14    0     0     0     0     0     0     0     0                                  |
|  22     FCS     0      10    8    86    14    0     0     0     0     0     0     0     0                                  |
|  23     FCS     0      10    8    86    14    0     0     0     0     0     0     0     0                                  |
|  24     FCS     0      10    8    86    14    0     0     0     0     0     0     0     0                                  |
|  29     FCS     0      10    6    100   0     0     0     0     0     0     0     0     0                                  |
|  4A     FCS     0      10    6    100   0     0     0     0     0     0     0     0     0                                  |
|  4E     FCS     0      10    6    100   0     0     0     0     0     0     0     0     0                                  |
|  51     FCS     0       8    4    100   0     0     0     0     0     0     0     0     0                                  |
|  53     FCS     0       8    4    100   0     0     0     0     0     0     0     0     0                                  |
|  20     FCS     0       6   12    28    72    0     0     0     0     0     0     0     0                                  |
|  4B     FCS     0       4    4    100   0     0     0     0     0     0     0     0     0                                  |
|  4D     FCS     0       4    4    100   0     0     0     0     0     0     0     0     0                                  |
|  42     FCS     0       4    4    100   0     0     0     0     0     0     0     0     0                                  |
|  44     FCS     0       4    4    100   0     0     0     0     0     0     0     0     0                                  |
|  14     OSD     0       2    2    100   0     0     0     0     0     0     0     0     0                                  |
|  AA     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
|  AB     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
|  AC     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
|  AD     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
|  AE     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
|  AF     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
|  A0     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
|  A1     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
|  A2     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
|  A3     SCHP    0       0    0    100   0     0     0     0     0     0     0     0     0                                  |
```

Figure 3-22. VM Channel Path Load Analysis

3.2.1.2.9 VM FICON Channel Path Load Analysis

The VM FICON Channel Path Load Analysis report provides information about FICON channels. Only channels for which "busy" conditions occur are included. Channel paths provide the interface that allows for control units and I/O devices to be physically connected to the I/O subsystem.

INTENDED USE:

The report is designed to help analyze FICON channels by path ID. For each ID, percent channel utilization for the partition and total for the system can be analyzed to help detect bottlenecks in the I/O subsystem caused by excessive path utilization.

FORMAT:

A tabular listing that contains one line of information for each active channel configured during the reporting period. Report is ordered by channel path ID. In addition to the channel path ID, the following columns are shown on the report:

CHPID - Hexadecimal number of the channel path identifier.

SHRD - Indicator for the channel which can be either YES for channels shared with other logical partitions, or NO for channels dedicated to this logical partition.

FICON CHANNEL UTILIZATION %: The following fields show the relative utilization of resources compared to the maximum value, in percent.

TOTAL FOR SYSTEM:

BUS CYCLE T_BCY - The bus cycles utilization for the whole system.

WORK UNITS T_WUN - The work units utilization for the whole system.

DATA UNITS WRITE T_DUW - The data units write utilization for the whole system.

DATA UNITS READ T_DUR - The data units read utilization for the whole system.

```
        OWN PARTITION:

        WORK UNITS L_WUN - The work units utilization for the
        own logical partition.

        DATA UNITS WRITE L_DUW - The data units write
        utilization for the own logical partition.

        DATA UNIT READ L_DUR - The data units read
        utilization for the own logical partition.

        TOTAL DATA TRANSFER RATE (BYTES/S):

        WRITES/S - The total number of bytes written per
        second for the whole system.

        READ/S - The total number of bytes read per second
        for the whole system.

   An example of this report is shown in Figure 3-23.

INQUIRY ID:

   VMXLS2

DATA SOURCE (File/Timespan):

   VMXCPM01 - DETAIL

DATA ELEMENTS USED:

   CHANNEL  - Physical Channel Address
   CPMTYPE  - VM Channel Path Type
   CPMCMG   - VM Channel Measurement Group
   CPMCMMQ  - VM Channel Model Group Qualifier
   CPMPCPSY - VM Percent Channel Busy - CPC
   CPMCBCCT - VM Total Bus Cycles Count
   CPMSHRCT - VM Channel Path Shared
   CPMCPINT - VM Channel Path Interval Time
   CPMCWUT  - VM FICON Channel Work Unit Count - CPC
   CPMTDUW  - VM FICON Write Data Unit Count - CPC
   CPMTDUR  - VM FICON Read Data Unit Count - CPC
   CPMCWUL  - VM FICON Channel Work Unit Count - LPAR
   CPMLDUW  - VM FICON Write Data Unit Count - LPAR
   CPMLDUR  - VM FICON Read Data Unit Count - LPAR
   CPMMXIBC - VM Max FICON Internal Bus Cycles Per Second
   CPMMXUNP - VM Max FICON Channel Work Units Per Second
   CPMMXUNW - VM Max FICON Write Data Units Per Second
```

```
   CPMMXUNR - VM Max FICON Read Data Units Per Second
   CPMDUS   - VM FICON Data Unit Size

 CALCULATIONS:

   Percent average total internal bus cycle utilization:

    T_BCY=(CPMCBCCT*100)/CPMMXIBC/CPMCPINT;

   Percent average total channel work units utilization:

    T_WUN=(CPMCWUT*100)/CPMMXUNP/CPMCPINT;

   Percent average total data unit writes utilization:

    T_DUW=(CPMTDUW*100)/CPMMXUNW/CPMCPINT;

   Percent average total data unit reads utilization:

    T_DUR=(CPMTDUR*100)/CPMMXUNR/CPMCPINT;

   Percent average logical channel work unit utilization
   for the partition:

    L_WUN=(CPMCWUL*100)/CPMMXUNP/CPMCPINT;

   Percent average logical channel data unit writes
   utilization for the partition:

    L_DUW=(CPMLDUW*100)/CPMMXUNW/CPMCPINT;

   Percent average logical channel data unit reads
   utilization for the partition:

    L_DUR=(CPMLDUR*100)/CPMMXUNR/CPMCPINT;

   Average total bytes written per second:

    WRITES=ROUND((CPMTDUW*CPMDUS)/CPMCPINT);

   Average total bytes read per second:

    READS=ROUND((CPMTDUR*CPMDUS)/CPMCPINT);

USAGE CONSIDERATIONS:

   None
```

```
|                          F I C O N  C H A N N E L  L O A D  A N A L Y S I S  R E P O R T
|                                                                                                                  Page 001     |
|    z/VM                          System ID SYSA               Start 20JULyy:23:52:45.92                                     |
|    Release  V5R4                                                End 20JULyy:23:59:45.92                                     |
|                                                                     CPMF: Extended Mode      CSSID:  0                      |
|------------------------------------------------------------------------------------------------------------------------------|
|                    <------------------ FICON Channel Utilization % ----------------->                                       |
|                    <--------- Total for System --------->  <---- Own Partition ----->  <--- Total  Data --->               |
|Channel               Bus       Work      <- Data Units ->   Work     <- Data Units ->   <-- Transfer Rate -->               |
|Path                Cycle       Units      Write       Read  Units     Write       Read   <---- (Bytes/s) ---->              |
|ID        Shared    T_Bcy      T_WUn      T_DUW      T_DUR  L_WUn     L_DUW      L_DUR  Writes/s      Reads/s               |
|0A          YES         0          0          0          0      0         0          0       725         7085                 |
|0B          YES         0          0          0          0      0         0          0       436            0                 |
|0C          YES         0          0          0          0      0         0          0       284            0                 |
|0D          YES         0          0          0          0      0         0          0         0            0                 |
|00          YES         0          0          0          0      0         0          0         0         1877                 |
|01          YES         0          0          0          0      0         0          0     52744        11940                 |
|02          YES         0          0          0          0      0         0          0      7957        10867                 |
|03          YES         0          0          0          0      0         0          0         0         1877                 |
|04          YES         0          0          0          0      0         0          0         0         1877                 |
|05          YES         0          0          0          0      0         0          0         0            0                 |
|06          YES         0          0          0          0      0         0          0         0          269                 |
|07          YES         0          0          0          0      0         0          0         0            0                 |
|08          YES         0          0          0          0      0         0          0         0          269                 |
|09          YES         0          0          0          0      0         0          0         0         1877                 |
|10          YES         0          0          0          0      0         0          0         0         1877                 |
|11          YES         0          0          0          0      0         0          0         0         1877                 |
|12          YES         0          0          0          0      0         0          0         0         1877                 |
|13          YES         0          1          0          0      0         0          0    100670       283112                 |
|14          YES         0          2          0          0      0         0          0         0         1876                 |
|15          YES         0          0          0          0      0         0          0         0            0                 |
|16          YES         0          0          0          0      0         0          0         0            0                 |
|17          YES         0          0          0          0      0         0          0        10         7245                 |
|18          YES         0          0          0          0      0         0          0         0            0                 |
|2A          YES         0          0          0          0      0         0          0         0            0                 |
|2B          YES         5         23          2         14      1         0          0   3323764     28138817                 |
|2C          YES         0          0          0          0      0         0          0         0            0                 |
|2D          YES         1          7          1          1      1         0          0   1522803      2455957                 |
|2E          YES         0          0          0          0      0         0          0         0           27                 |
|20          YES         1         12          1          1      0         0          0   1441894      1462888                 |
|21          YES         1          7          1          0      0         0          0   2481329       524871                 |
|22          YES         1          7          1          0      0         0          0   2523201       498375                 |
```

Figure 3-23. VM FICON Channel Path Load Analysis

3.2.1.2.10 VM HiperSocket Channel Activity Report

The VM HiperSockets Channel Activity report provides information about HiperSocket channels. Only channels for which "busy" conditions occur are included. HiperSockets is a program that has the ability not only to directly communicate with a program running within the same logical partition (LP), but also to communicate across any logical partition within the same CPC.

INTENDED USE:

The report is designed to help analyze HiperSocket channels by path ID. For each ID, HiperSocket activity per second for the partition and total for the system can be analyzed to help detect bottlenecks in the I/O subsystem caused by excessive path utilization.

FORMAT:

A tabular listing that contains one line of information for each active channel configured during the reporting period. Report is ordered by channel path ID. In addition to the channel path ID, the following columns are shown on the report:

CHPID - Hexadecimal number of the channel path identifier.

SHRD - Indicator for the channel which can be either YES for channels shared with other logical partitions, or NO for channels dedicated to this logical partition.

HIPERSOCKET ACTIVITY/SEC:

TOTAL FOR SYSTEM:

TRANSFERRED T_MSGS - The number of messages sent per second for the whole system.

TRANSFERRED T_DUNITS - The number of data units per second for the whole system.

FAILED T_NOBUFF - The number of sends per second that failed due to unavailable receive buffers, for the whole system.

OWN PARTITION:

TRANSFERRED L_MSGS - The number of messages sent per second by programs in the own partition.

TRANSFERRED L_DUNITS - The number of data units per second for the own partition.

FAILED L_NOBUFF - The number of sends per second that failed due to unavailable receive buffers, for the own partition.

FAILED L_OTHER - The number of sends per second that failed due to other problems, for the own partition.

An example of this report is shown in Figure 3-24.

```
INQUIRY ID:

   VMXLS3

DATA SOURCE (File/Timespan):

   VMXCPM01 - DETAIL

DATA ELEMENTS USED:

   CHANNEL  - Physical Channel Address
   CPMTYPE  - VM Channel Path Type
   CPMCMG   - VM Channel Measurement Group
   CPMPSTMS - VM HiperSockets Message Sent Rate - CPC
   CPMPSTUB - VM HiperSockets Receive Failed Rate - CPC
   CPMPSPMS - VM HiperSockets Message Sent Rate - LPAR
   CPMPSPUB - VM HiperSockets Receive Failed Rate - LPAR
   CPMPSPUS - VM HiperSockets Message Failed Rate - LPAR
   CPMHDUST - VM HiperSockets Data Units Sent - CPC
   CPMHDUSL - VM HiperSockets Data Units Sent - LPAR

 CALCULATIONS:
   Average total data units sent per second for the CPC:

   T_DUNITS

   ROUND(CPMHDUST/CPMCPINT);

   Average data units sent per second for the LPAR:

   L_DUNITS

   ROUND(CPMHDUSL/CPMCPINT);
```

USAGE CONSIDERATIONS:

None

```
+-------------------------------------------------------------------------------------------------------------------------------+
|                               C H A N N E L   P A T H   L O A D   A N A L Y S I S   R E P O R T                               |
|                                                                                                                      Page nnn|
|    z/VM                       System ID SYSC              Start 20JULyy:23:59:45.92                                       |
|    Release  V5R4                                            End 21JULyy:00:59:45.92                                       |
|                                                                 CPMF: Extended Mode      CSSID:  0                            |
|-------------------------------------------------------------------------------------------------------------------------------|
|            <---------------------- HiperSocket Activity/Sec ---------------------->                                           |
|            <-- Total for System -->       <----------- Own Partition ------------>                                           |
|Channel     <--Transferred-->     Failed   <--Transferred-->   <----- Failed ----->                                           |
|ID      Shrd T_Msgs   T_DUnits   T_NoBuff   L_Msgs   L_DUnits   L_NoBuff     L_Other                                          |
|FB       NO    0.63      15033       1.94     0.59      15033       1.94        1.94                                          |
|FE       NO    0.00     ------       0.00     0.00       0.00       0.00        0.00                                          |
|FF       NO    0.00     ------       0.00     0.00       0.00       0.00        0.00                                          |
+-------------------------------------------------------------------------------------------------------------------------------+
|                               C H A N N E L   P A T H   L O A D   A N A L Y S I S   R E P O R T                               |
|                                                                                                                      Page nnn|
|    z/VM                       System ID SYSC              Start 21JUL10:00:59:45.92                                       |
|    Release  V5R4                                            End 21JUL10:01:59:45.92                                       |
|                                                                 CPMF: Extended Mode      CSSID:  0                            |
|-------------------------------------------------------------------------------------------------------------------------------|
|            <---------------------- HiperSocket Activity/Sec ---------------------->                                           |
|            <-- Total for System -->       <----------- Own Partition ------------>                                           |
|Channel     <--Transferred-->     Failed   <--Transferred-->   <----- Failed ----->                                           |
|ID      Shrd T_Msgs   T_DUnits   T_NoBuff   L_Msgs   L_DUnits   L_NoBuff     L_Other                                          |
|FB       NO    0.63      15033       1.94     0.59      15033       1.94        1.94                                          |
|FE       NO    0.00     ------       0.00     0.00       0.00       0.00        0.00                                          |
|FF       NO    0.00     ------       0.00     0.00       0.00       0.00        0.00                                          |
+-------------------------------------------------------------------------------------------------------------------------------+
|                               C H A N N E L   P A T H   L O A D   A N A L Y S I S   R E P O R T                               |
|                                                                                                                      Page nnn|
|    z/VM                       System ID SYSC              Start 21JUL10:01:59:45.92                                       |
|    Release  V5R4                                            End 21JUL10:02:59:45.92                                       |
|                                                                 CPMF: Extended Mode      CSSID:  0                            |
|-------------------------------------------------------------------------------------------------------------------------------|
|            <---------------------- HiperSocket Activity/Sec ---------------------->                                           |
|            <-- Total for System -->       <----------- Own Partition ------------>                                           |
|Channel     <--Transferred-->     Failed   <--Transferred-->   <----- Failed ----->                                           |
|ID      Shrd T_Msgs   T_DUnits   T_NoBuff   L_Msgs   L_DUnits   L_NoBuff     L_Other                                          |
|FB       NO    0.63      15033       1.94     0.59      15033       1.94        1.94                                          |
|FE       NO    0.00     ------       0.00     0.00       0.00       0.00        0.00                                          |
|FF       NO    0.00     ------       0.00     0.00       0.00       0.00        0.00                                          |
+-------------------------------------------------------------------------------------------------------------------------------+
|                               C H A N N E L   P A T H   L O A D   A N A L Y S I S   R E P O R T                               |
|                                                                                                                      Page nnn|
|    z/VM                       System ID SYSC              Start 21JUL10:02:59:45.92                                       |
|    Release  V5R4                                            End 21JUL10:03:59:45.92                                       |
|                                                                 CPMF: Extended Mode      CSSID:  0                            |
|-------------------------------------------------------------------------------------------------------------------------------|
|            <---------------------- HiperSocket Activity/Sec ---------------------->                                           |
|            <-- Total for System -->       <----------- Own Partition ------------>                                           |
|Channel     <--Transferred-->     Failed   <--Transferred-->   <----- Failed ----->                                           |
|ID      Shrd T_Msgs   T_DUnits   T_NoBuff   L_Msgs   L_DUnits   L_NoBuff     L_Other                                          |
|FB       NO    0.63      15033       1.94     0.59      15033       1.94        1.94                                          |
|FE       NO    0.00     ------       0.00     0.00       0.00       0.00        0.00                                          |
|FF       NO    0.00     ------       0.00     0.00       0.00       0.00        0.00                                          |
```

Figure 3-24. VM HiperSocket Channel Activity Report

3.2.1.2.11 Shared Address Space Analysis Report

The Shared Address Space Analysis report provides information on DASD page activity, main storage activity including expanded storage, pages stolen, pages locked, and resident pages available for all shared address spaces on a z/VM system.

INTENDED USE:

The report is designed to help analyze how much physical memory the virtual machine is using by tracking residency data for each virtual machine. This can help predict when more physical memory needs to be brought online. This information is essential to tracking the memory residency of VDISK address spaces.

FORMAT:

A tabular listing that contains one line of information for each address space active during the reporting period. The report is ordered by ascending Owner sequence (lowest to highest). In addition to the Owner, the following columns are shown on the report:

Address Space Name - Address space identifier

Size - Defined address space size

Under Address Space Storage:

Resident:

LT 2GB - number of pages less than 2GB that were resident

GT 2GB - number of pages greater than 2GB that were resident

Locked:

LT 2GB - number of pages residing in storage less than 2GB locked by the address space

GT 2GB - number of pages residing in storage greater than 2GB locked by the address space

```
      Paged Out:

         DASD -  number of pages read from DASD into main
                 storage

         XSTOR - number of pages read from expanded storage
                 into main storage

      Under Pages Per Second:

         Stolen - number of page frames taken from the
                  address space per second

      DASD:

         IN -  number of pages written from main storage to
               DASD per second

         OUT - number of pages written from DASD to main
               storage per second

      Expanded Storage:

         IN -  number of pages written to expanded storage
               from main storage per second

         OUT - number of pages read from expanded storage
               into main storage per second

         Mig - number of pages migrated from DASD to
               expanded storage per second


   An example of this report is shown in Figure 3-26.

INQUIRY ID:

   VMXLS4

DATA SOURCE (File/Timespan):

   VMXASC01 - DETAIL

DATA ELEMENTS USED:

   USER     - User Identification
   ASCNAME  - Address Space Name
   ASCDEFSZ - Defined Address Space Size
```

ASCTRSAF - Resident Pages <2G
ASCCTPRG - Resident Pages >2G
ASCPGLOK - Pages Locked <2G
ASCCPLKA - Pages Locked >2G
ASCCTPGR - Pages Read from DASD
ASCCTXRD - Pages Written to DASD
ASCPGSRT - Pages Stolen per Second
ASCPGWRT - Pages Written to DASD per Second
ASCPGRRT - Pages Read from DASD per Second
ASCPGWXS - Pages Written to XSTOR per Second
ASCPGRXS - Pages Read from XSTOR per Second
ASCPSMGD - Pgs Migrated from DASD to XSTOR per Sec'd

USAGE CONSIDERATIONS:

None

```
                                  S H A R E D   A D D R E S S   S P A C E   A N A L Y S I S


   z/VM                        System ID SYSA               Start DDMMMYY:HH:MM:SS.XX                                  Inter
   Release  VnRn                                              End DDMMMYY:HH:MM:SS.XX

                                                  DETAILS FOR ALL OWNERS
                                          <---------- Address Space Pages ---------->    <------------ Pages per Second -
                                          <- Resident ->  <-- Locked -->  <- Paged Out->          <--- DASD --->  <- Expand
   Owner       Address Space Name         Size  LT 2GB  GT 2GB  LT 2GB  GT 2GB    DASD   XSTOR    Stolen      In     Out      In
-------- ------------------------  -----------  ------  ------  ------  ------  ------  ------    ------  ------  ------  ------  -
 CCIVM    BASE                        33554431     181       0       1       0       0       1         0       0       0       0
 CCIVM    CCIDS                     2147483647       0       0       0       0       0       0         0       4       0       0
 SYSTEM   FULL$TRACK$CACHE$1        2147483647       0       0       0       0       0       0         1       0       5       0
 SYSTEM   ISFCDATASPACE             2147483647      67       0      48       0       0       0         0       0       0       0
 SYSTEM   PTRM0000                  4294967295    6492       0       0       0       0       0         0       2       0       0
 SYSTEM   REAL                      1610612735       0       0       0       0       0       0         0       0       3       0
 SYSTEM   SYSTEM                    2147483647      50       0       0       0       0       0         1       0       0       0
 SYSTEM   VDISK$SYSLD40A$0200$0001  1611661311       0       0       0       0       0       0         0       6       0       0
 SYSTEM   VDISK$SYSLSS27$0200$0004   210763775       0       0       0       0       0       0         0       0       0       0
 SYSTEM   VDISK$SYSLW40A$01F0$0002   263192575       3       0       0       0       0       0         2       0       0       0
 SYSTEM   VDISK$SYDLW40A$01F1$0003   105906175       5       0       0       0       0       0         0       3       0       0
 SYSTEM   VIRTUAL$FREE$STORAGE      2147483647       0       0       0       0       0       0         0       8       0       0
```

Figure 3-26. Shared Address Space Analysis Report

3.2.2 How to Modify Standard MICF Inquiries

Figures 3-41 through 3-50 are sample MICF inquiry modification panels. See the CA MICS MICF User Guide, for tailoring MICF inquiries.

```
/-------------------------------------------------------------------------------\
|---------------------------  Data Base Inquiries  ----------------------------|
|Command ===>                                                 Scroll ===> PAGE |
|                                                                               |
|Catalog     ===> SHARED  (PRIVATE/SHARED)  Catalog Group         ===> *        |
|Select Mode ===> EXECUTE (EXECUTE/MODIFY)  Name (subset display) ===> ________ |
|                                                                               |
|Line Cmds: S Select  I Insert   R Repeat  C Copy  D Delete                     |
|                                                                               |
|Cmd   Name                  Title                        Date     Time         |
| -   ------  ---------------------------------------  ---------  -----        |
| -   VMXCX1  Detail VMX Concurrent Channel Busy        03/05/08   17:00        |
| -   VMXCX2  Detail VMX Percent Channel Busy           03/05/08   17:05        |
| C   VMXCX3  Detail VMX IUCV/VMCF User Summary         03/05/08   17:10        |
|****************************** BOTTOM OF DATA *********************************|
\-------------------------------------------------------------------------------/
```

Figure 3-41. Copying an Inquiry to a Private Catalog

```
 /-------------------------------------------------------------------------------\
 |---------------------------  Inquiry Repeat/Copy  -------------- Copy Inquiry|
 |Command ===>                                                                   |
 |                                                                               |
 |Catalog:  SHARED                                                               |
 |                                                                               |
 |     Inquiry Name   ===> VMXCX3                                                |
 |             Title  ===> Detail VMX IUCV/VMCF User Summary                     |
 |             Author ===> Your Name                                             |
 |                                                                               |
 |Catalog Group Name  ===> AVAIL                                                 |
 |              Title ===> Availability Analysis                                 |
 |                                                                               |
 |                                                                               |
 |                                                                               |
 |Press ENTER to add the new inquiry to your private inquiry catalog.            |
 |or                                                                             |
 |Press END to cancel.                                                           |
 \-------------------------------------------------------------------------------/
```

Figure 3-42. Adding an Inquiry to a Private Library

```
/-------------------------------------------------------------------------------\
|---------------------------  Data Base Inquiries  ----------------------------|
|Command ===>                                                  Scroll ===> PAGE |
|                                                                               |
|Catalog     ===> PRIVATE (PRIVATE/SHARED)  Catalog Group         ===> *        |
|Select Mode ===> MODIFY  (EXECUTE/MODIFY)  Name (subset display) ===> ________ |
|                                                                               |
|Line Cmds: S Select  I Insert   R Repeat  C Copy  D Delete                     |
|                                                                               |
|Cmd   Name                   Title                        Date     Time        |
| -   ------  ----------------------------------------  ---------  -----        |
| S   VMXCX3  Detail VMX IUCV/VMCF User Summary         03/05/08   17:05        |
|****************************** BOTTOM OF DATA *********************************|
\-------------------------------------------------------------------------------/
```

Figure 3-43. Selecting an Inquiry for Modification

```
/-------------------------------------------------------------------------------\
|-------------------------  MICF Inquiry Step Display  -------------------------|
|Command ===>                                                  Scroll ===> CSR  |
|                                                                               |
|Modifying CA MICS Inquiry: VMXCX3 - Detail VMX IUCV/VMCF User Summary          |
|                                                                               |
|Execute the Inquiry in ===> BATCH      (BATCH/FOREGROUND) Mode ===> N (Y/N)    |
|                                                                               |
|Line Cmds: S Select  I Insert  D Delete  M Move  R Repeat  C Copy              |
|                                                                               |
|Cmd                         Inquiry Step                          Input File   |
| -   ------------------------------------------------------------  ----------- |
| _   MICS File Selection                                          DAYS VXT     |
| _   Common Data Selection - Select Previous Day                  FILE1        |
| _   Execution-Time Data Selection - Select SYSID                 FILE1        |
| _   Data Sequencing - by SYSID                                   FILE1        |
| _   Data Element Derivation - Compute DOWNTIME                   FILE1        |
| S   Color Vertical Bar Chart - DOWNTIME by HOUR                  FILE1        |
|****************************** BOTTOM OF DATA *********************************|
\-------------------------------------------------------------------------------/
```

Figure 3-44. Selecting an Inquiry Step for Modification

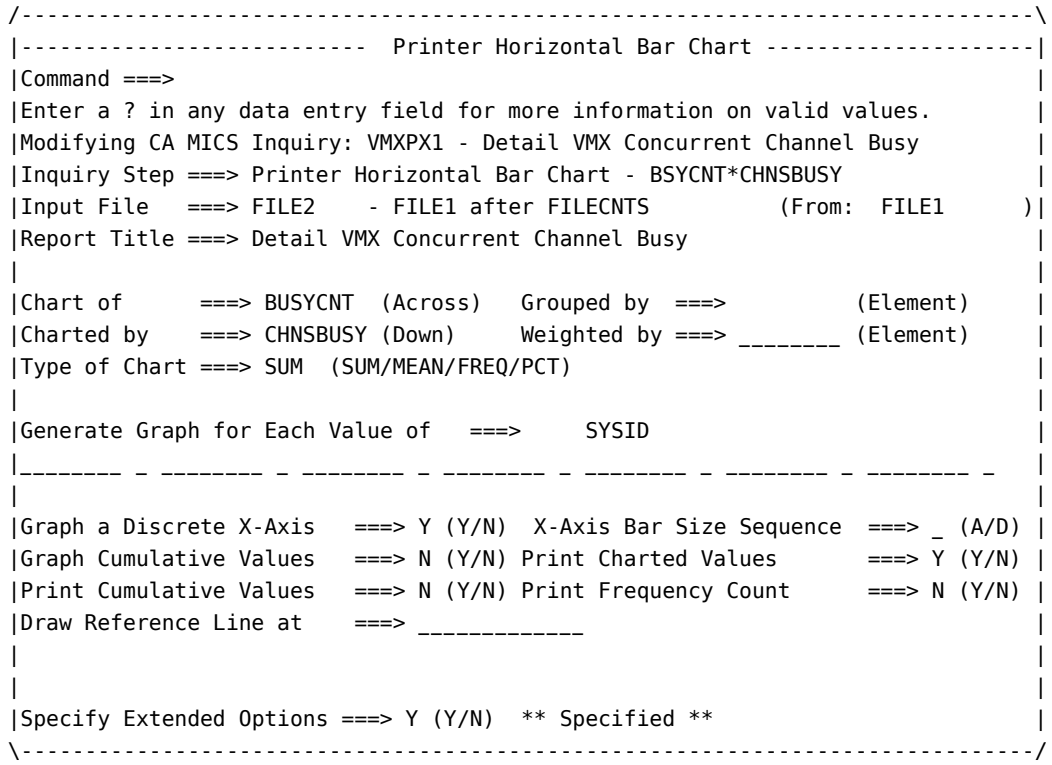


Figure 3-45. Specifying Extended Options

```
/-------------------------------------------------------------------------------\
|--------------------------  Extended Options Menu  --------------------------|
|Command ===>                                                                   |
|                                                                               |
|Inquiry Step:  Color Vertical Bar Chart - DOWNTIME by HOUR                     |
|Input File:    File1 - Intermediate File Number 1 (From:  DAYS PRF)            |
|                                                                               |
|Specify Data Element Labels and Formats      ===> N (Y/N/R)  ** Specified **   |
|Specify Subheading Text and Data Formats     ===> Y (Y/N/R)  ** Specified **   |
|Specify Axis Scale                           ===> N (Y/N/R)  ** Specified **   |
|Specify Reference Lines                      ===> N (Y/N/R)                    |
|Specify Chart Group Values                   ===> N (Y/N/R)                    |
|                                                                               |
|                                                                               |
|                                                                               |
|                                                                               |
|Begin Chart With Record Number ===> ________   Stop After ===> _______ Records |
\-------------------------------------------------------------------------------/
```

Figure 3-46. Selecting a Specify Option

```
/-------------------------------------------------------------------------------\
|-------------------------Subheading Text and Data Formats----------------------|
|Command ===>                                                                   |
|                                                                               |
|Inquiry Step:  Color Vertical Bar Chart - DOWNTIME by HOUR                     |
|Input File:    File1 - Intermediate File Number 1 (From:  DAYS PRF)            |
|                                                                               |
|  Element           Subheading Text                   Output Format            |
|  ________  ________________________________________  _______________          |
|  SYSID     S                                        $CHAR4.                  |
|                                                                               |
|                                                                               |
\-------------------------------------------------------------------------------/
```

Figure 3-47. Specifying Subheading Text in Extended Options

```
/-------------------------------------------------------------------------------\
|------------------------  MICF Inquiry Step Display  -------------------------|
|Command ===> V                                                Scroll ===> CSR  |
|                                                                               |
|Modifying CA MICS Inquiry: VMCCD3 - Daily VM Availability Report               |
|Inquiry validation is required.  Enter VALIDATE to validate the inquiry.       |
|Execute the Inquiry in ===> BATCH      (BATCH/FOREGROUND) Mode ===> N (Y/N)    |
|                                                                               |
|Line Cmds: S Select  I Insert  D Delete  M Move  R Repeat  C Copy              |
|                                                                               |
|Cmd                         Inquiry Step                          Input File   |
| -   ------------------------------------------------------------  ----------- |
| _   MICS File Selection                                          DAYS PRF     |
| _   Common Data Selection - Select Previous Day                  FILE1        |
| _   Execution-Time Data Selection - Select SYSID                 FILE1        |
| _   Data Sequencing - by SYSID                                   FILE1        |
| _   Data Element Derivation - Compute DOWNTIME                   FILE1        |
| _   Color Vertical Bar Chart - DOWNTIME by HOUR                  FILE1        |
|****************************** BOTTOM OF DATA ********************************|
\-------------------------------------------------------------------------------/
```

Figure 3-48. Validating a Modified Inquiry

```
/-------------------------------------------------------------------------------\
|----------------------  Active Inquiry Save Confirmation  ------ Confirm Save. |
|Command ===>                                                                   |
|                                                                               |
|                                                                               |
|   Active MICF Inquiry:  VMXCX3 - Detail VMX IUCV/VMCF User Summary            |
|                Author:           Your Name                                    |
|                                                                               |
|                                                                               |
|   Do you wish to save the current, active MICF Inquiry? ===> YES (YES/NO)     |
|                                                                               |
|                                                                               |
|                                                                               |
|                                                                               |
|Instructions:                                                                  |
|                                                                               |
|   Press ENTER  key to save the current, active MICF Inquiry.                  |
|                                                                               |
|   Reply NO and press ENTER key to delete the current, active inquiry.         |
\-------------------------------------------------------------------------------/
```

Figure 3-49. Saving a Modified Inquiry

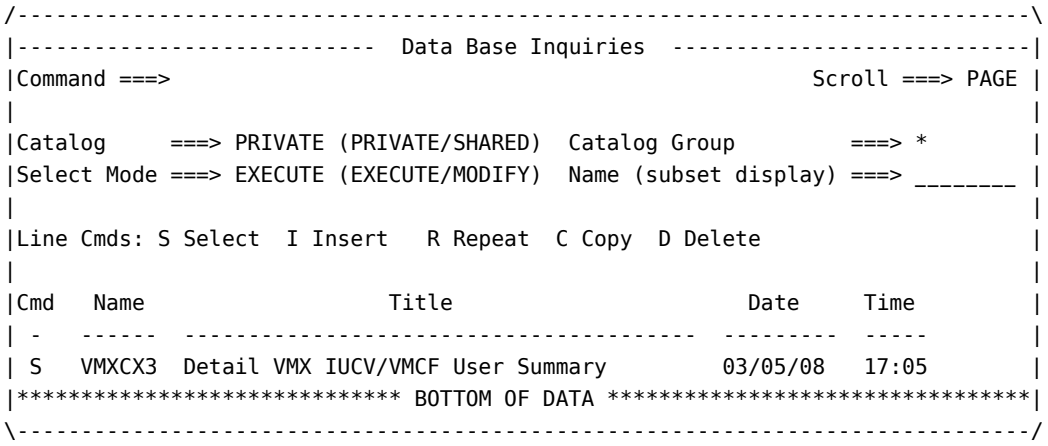


Figure 3-50. Selecting a Modified Inquiry for Execution

3.3 VM Management Objective Reports

This section identifies and describes the Management Objective Reports that are provided for the VMC and VMX Information Areas. The reports are produced on a daily, weekly, and monthly basis and they are described in that order within this section.

User-defined options are provided to allow you to specify data selection (time, zones, etc.), define the applicable objective values, and specify the scales of the graphic report representations. Each of the options is described along with the report to which it pertains and is fully explained in Section 3.3.4, VM Management Objective Report Options.

Options available for producing the VM Management Objective Reports are shown below:

Hourly Selection Range for Daily Reports: The start and end hours of the day that are to be included in the Daily VM Reports. We recommend that the hours be set to include the entire day (i.e., 00 23).

Hourly Selection Range for Weekly Reports: The start and end hours of the day that are to be included in the Weekly VM Reports. We recommend that the hours be set to include only the prime-time hours (e.g., 08 16).

Total Response Objective: The average VM Q1 service time for all Q1 responses in the reporting period

Hourly VM Availability: The target number of uptime minutes per hour, generally set to 60, measured during the interval spanned by the data

Zone Selection for Monthly Reporting: The zone or range of zones to be included in the Monthly VM Reports. Normally, all zones should be reported; therefore, the range should be 1 to 9. The VMC reports could be limited to only the range of zones required.

Monthly System Availability: The percentage of actual time during each zone in the month that the VM monitor shows VM is actually available

Monthly VM CPU Time Consumption Limit: The target and maximum average CPU time used by each processor engine per wall clock hour (by zone) during the month. For

example, the target of 2700 CPU seconds per hour would be specified as 2700 and 3600.

Monthly VM Throughput Rate: The target and maximum number of Q1 drops processed per zone within month by VM. The target amount is used for a reference line on the graph and the maximum is used to set the maximum value on the graph's vertical axis.

The VM Report Options, which are defined as SAS MACRO variables, are contained in the source member #VMCMOBJ in the MICS.SOURCE library. The SAS source code for the VM Management Objective Reports is contained in the following three members in sharedprefix.MICS.SOURCE:

DYVMCMBO - Daily VM Management Objective Reports
WKVMCMBO - Weekly VM Management Objective Reports
MNVMCMBO - Monthly VM Management Objective Reports

The following sections present more specific information on the VM reports:

1 - Daily VM Reports
2 - Weekly VM Reports
3 - Monthly VM Reports
4 - VM Management Objective Report Options

3.3.1 Daily VM Reports

The daily objective reports provide management a way to determine whether the primary VM processing objectives are being attained. One-page charts illustrate the day's activity by hour in reference to the established hourly objectives for response, throughput, resource consumption, and availability.

An hour-of-the-day selection facility is provided so that these daily reports may reflect either the entire 24 hours or a user-specified range.

Hourly Selection Range for Daily Reports: The start and end hours of the day that are to be included in the Daily VMC Reports. We recommend that the hours be set to include the entire day (i.e., 00 23).

&VMCDLHR and &VMCDHHR defines the low and high hours of the day for selection of the VM activity to be reported in the daily reports. Therefore, to set the hours to include the entire day, code the following:

```
%LET VMCDLHR = 00;
%LET VMCDHHR = 23;
```

For a full description of the user options, see Section 3.3.4.1, VM Report Options.

The following sections display and explain the Daily VM Management Objective Reports:

1 - Report Format Descriptions
2 - Daily VM Resource Consumption Report
3 - Daily VM Service Report
4 - Daily VM Availability Report
5 - Daily VM Throughput Report

3.3.1.1 Report Format Descriptions

The Daily Management Objective Reports are produced as a standard process in the CA MICS DAILY job stream. The SAS CHART procedure is used to produce the desired bar chart reports.

The heading lines contain the installation name (e.g., XYZ Manufacturing, Inc.), the frequency identification (e.g., DAILY VM MANAGEMENT OBJECTIVES) the objective being reported (e.g., VM RESOURCE CONSUMPTION WITHIN 2700 CPU SECONDS/HOUR), and the computing system (e.g., SYSID code) on which the VM processing was measured.

The format of the bar chart illustrates the hour of the day along the horizontal axis and the measurement being reported (e.g., availability), along the vertical axis. The day being reported is identified below the hours of the day, just under the horizontal axis.

The vertical axis is used to chart the measure being reported and it is with this axis that the management objective for this category is defined. The vertical axis is first established through the SAS CHART AXIS option which simply identifies the highest value to be represented on the vertical axis. The management objective is then illustrated through the SAS CHART REF option and formats a dashed line across the page to illustrate the management objective.

Consider the charting of the hourly objective for Q1 service (response time; see Section 3.3.1.3, Daily VM Service Report). The title describes the management objective as having average Q1 service within three seconds for all Q1 activity (usually equal to VM trivial command response time). The measure that will be charted is the average Q1 service time, shown for each hour of the day. The vertical axis, therefore, is set up for a maximum value of three seconds, and the vertical reference line is printed at the three-second level. This reference line then clearly depicts how the objective was satisfied or missed for each hour of the day.

A special note should be made at this time to point out that the location of the reference line on the vertical axis may be rounded for purposes of scaling the axis. Each vertical line may have a unit value of .10 seconds, so a reference line specification of 1.75 seconds does not fall directly on a vertical axis coordinate. In this case, the 1.75-second line is shown as a line at 1.80 seconds.

The member DYVMCMBO in the sharedprefix.MICS.SOURCE library contains the SAS statements for producing the charts. The member #VMCMOBJ contains the SAS statements used by DYVMCMBO. The #VMCMOBJ member contains the specifications for defining the objectives to be displayed and may be modified to reflect your installation's unique management objectives. For defining your installation objectives, see Section 3.3.4, VM Management Objective Report Options.

The remainder of this section explains and illustrates each of the standard Daily VM Management Objective Reports.

3.3.1.2 Daily VM Resource Consumption Report

This report quantifies the total VM CPU resource consumption in seconds per hour averaged across all processor engines in the supporting computing system. This is but another way to look at percentage resource consumption. That is, if the average number of CPU seconds consumed by each processor is 1800 seconds (out of a possible 3600 seconds), then 50% of available processor capacity is being utilized.

The management objective to be tracked for evaluating the rate of resource consumption for VM is the average CPU time (in seconds) consumed per hour.

The only user option that applies to this report is the Daily VM CPU Consumption Amount, which is the target CPU consumption (expressed in seconds) that used to set a reference line on the graph.

&VMCSAX defines the CPU consumption chart axis. &VMCSREF defines the target CPU consumption amount. Therefore, to set a target CPU (average) consumption amount of 2700 CPU seconds for any given hour, code:

```
%LET VMCSAX = 3600;
%LET VMCSREF = 2700;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-51 illustrates the Daily VM Resource Consumption Report based on a management objective of consuming less than 2700 CPU seconds (on average) during any given hour. The sample report shown here quantifies the seconds of CPU consumption with the horizontal axis representing the hours of the day and the vertical axis charting the average seconds of CPU time consumed per processing engine.

The management objective for VM resource consumption is shown by the dashed horizontal reference line printed across the chart. In Figure 3-51, the reference line is shown at the 2700 CPU seconds consumed level of the vertical axis. Therefore, any hour for which the charted bar exceeds this reference line signals that the objective to contain resource consumption was missed for that hour.

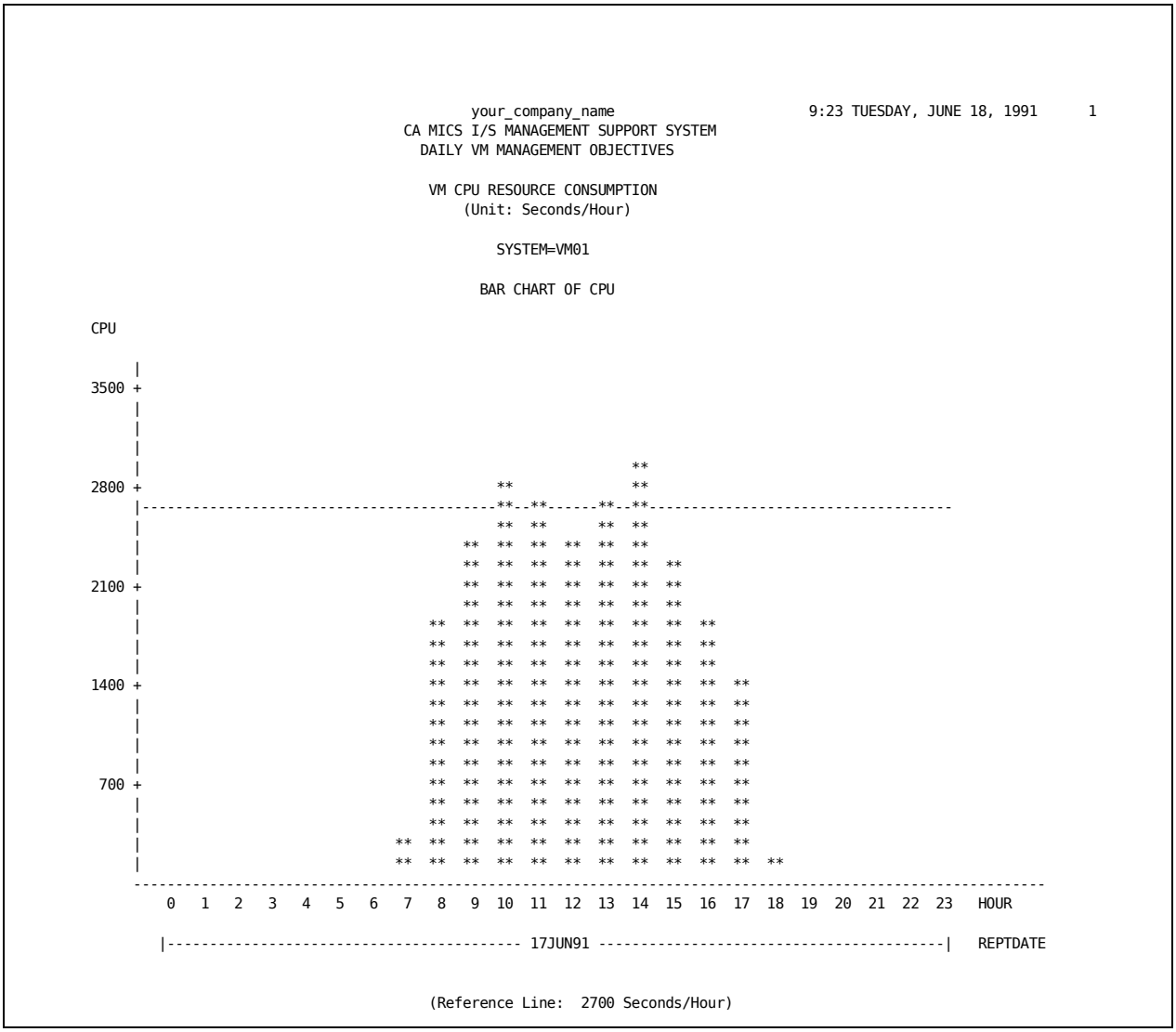


Figure 3-51. Daily VM Resource Consumption Report

3.3.1.3 Daily VM Service Report

This report quantifies the total VM service provided on an hourly basis for the identified day and computing system.

The management objective to be tracked for evaluating the speed and consistency for all VM activity is an absolute measure of the number of VM Q1 drops that occurred within a specified number of seconds.

The only user option that applies to this report is the Total Service Objective, which is the overall average of VM users that entered queue 1 and were subsequently dropped within the installation's service objective (e.g., average Q1 service within three seconds).

&VMCDSECX defines the Q1 service time chart axis. &VMCSEC defines the target Q1 service time. Therefore, to define, a target Q1 service of one second or less, code:

```
%LET VMCDSECX = 3.0;
%LET VMCSEC = 1.0;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-52 illustrates the Daily VM Service Report based on a management objective of maintaining an average VM Q1 service interval time of less than one second. The sample report shown here quantifies the service provided with the horizontal reference line representing the hours of the day and the vertical reference line charting the average VM Q1 service time in seconds, as identified in the title.

The management objective for total VM service is shown by the dashed horizontal reference line printed across the chart. In Figure 3-52, the reference line is shown at the three-second level of the vertical axis. Therefore, any hour for which the vertical bar exceeds this reference line signals that the VM service objective was missed for that hour.

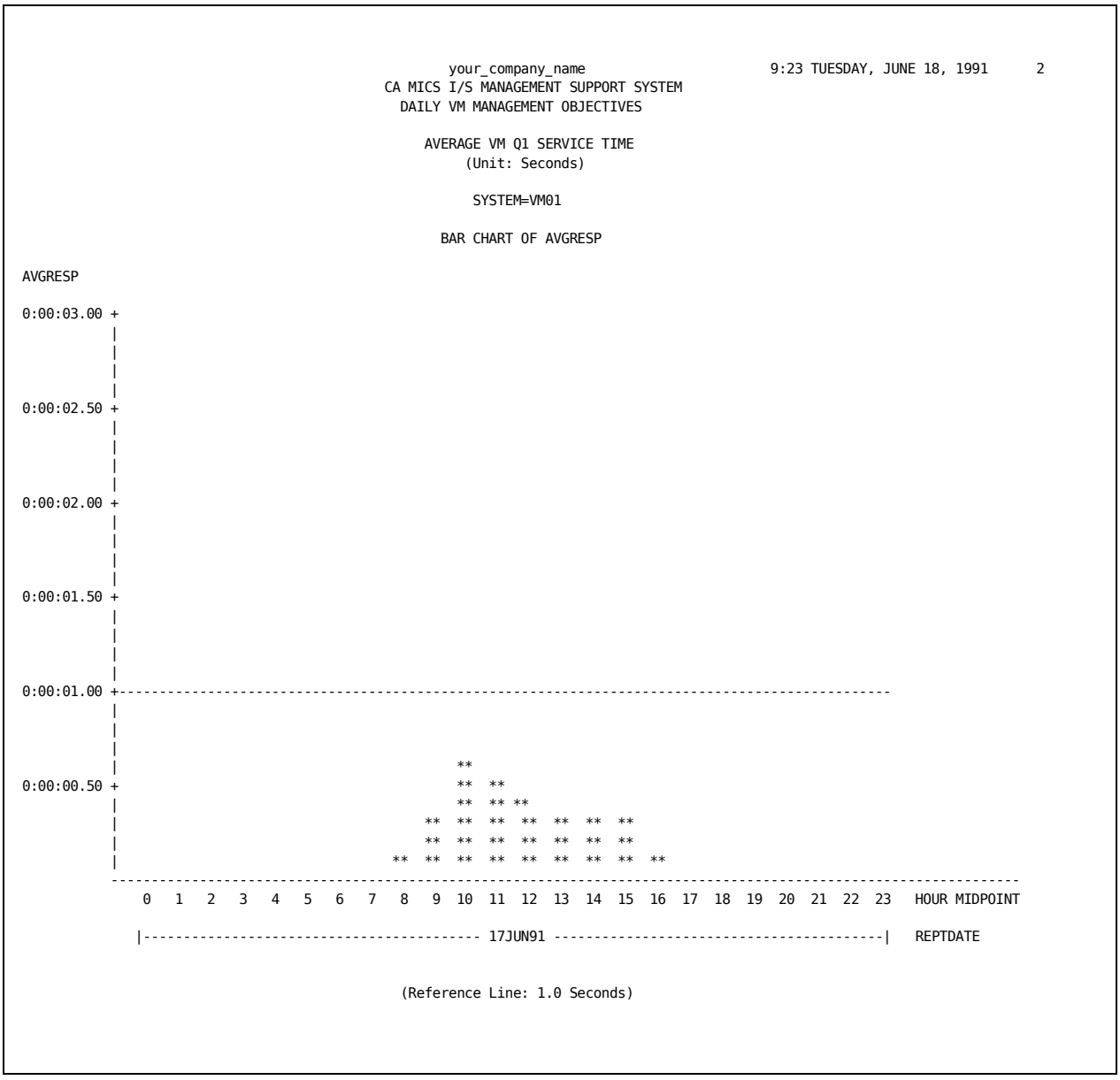


Figure 3-52. Daily VM Average Q1 Service Report

3.3.1.4 Daily VM Availability Report

This report shows the availability of a VM system on an hourly basis for the identified day and computing system.

The management objective to be tracked for evaluating the availability of the VM system is a measure of the number of minutes that the VM SCP was active within an hour of the day.

The only user option that applies to this report is the Hourly VM Uptime Limit which is the target number of uptime minutes per hour, generally set to 60.

&VMCDUTX sets the vertical axis of the hourly uptime chart. &VMCUT defines the hourly uptime target for daily and weekly reporting. Therefore, to define an hourly uptime of 60 minutes, code:

```
%LET VMCDUTX = 60;
%LET VMCUT = 60;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-53 illustrates the Daily VM Availability Report based on a management objective of no outage time (60 minutes uptime) per hour. The sample report shown here quantifies the system available time with the horizontal axis representing the hours of the day and the vertical axis charting the number of uptime minutes incurred.

The management objective is to incur no outage time. Therefore, any time a charted bar does not reach the 60 minute uptime reference line, the VM availability objective was missed for that hour. In Figure 3-53, the reference line is shown at the 60-minute level on the vertical axis.

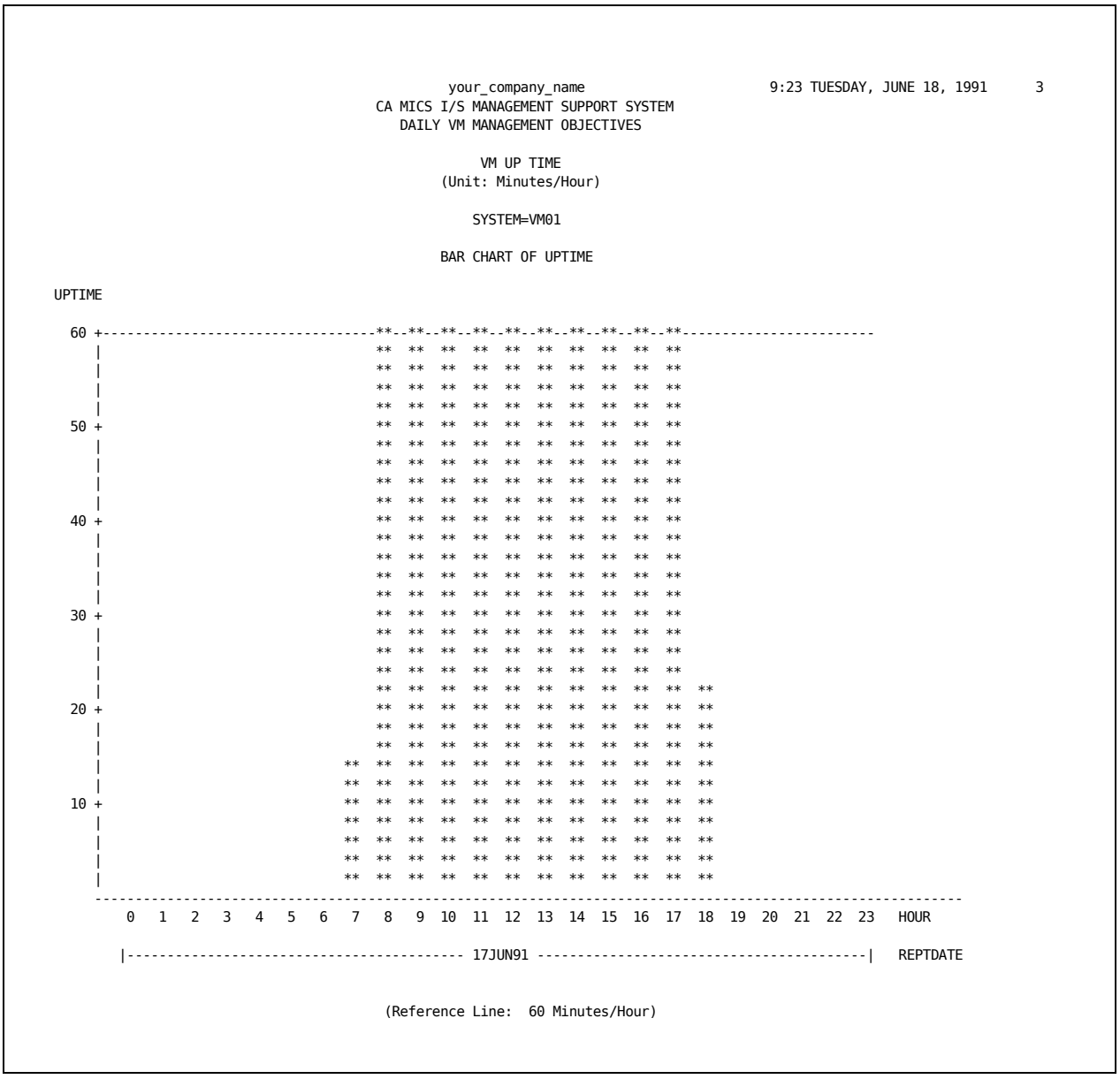


Figure 3-53. Daily VM Availability Report

3.3.1.5 Daily VM Throughput Report

This report quantifies the total VM throughput in terms of the number of Q1 drops processed on an hourly basis for the identified day and computing system.

The management objective to be tracked for evaluating the VM throughput rate is a measure of the number of Q1 drops processed per hour.

The only user option that applies to this report is the Daily VM Command Rate, which is the target and maximum number of Q1 drops processed per hour by VM. The target number is used for a reference line on the graph and the maximum number is used to set the maximum value on the graph's vertical axis.

&VMCCAX and &VMCCREF define the vertical axis maximum scaling and reference line for hourly Q1 drop rate execution in the daily reports. Therefore, to define a maximum of 250,000 Q1 drops processed per hour and a target of 100,000 per hour, code:

```
&LET VMCCAX = 250000;
&LET VMCCREF = 100000;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-54 illustrates the Daily VM Throughput Report based on a management objective of processing less than 100,000 Q1 drops in any given hour. The sample report shown here quantifies the number of Q1 drops processed with the horizontal axis representing the hours of the day and the vertical axis charting the Q1 drops processed.

The management objective for VM throughput is shown by the dashed horizontal reference line printed across the chart. In Figure 3-54, the reference line is shown at the 100,000 Q1 drops processed level of the vertical axis. Therefore, any hour for which the charted bar exceeds this reference line signals that the objective to contain throughput was missed for that hour.

The vertical axis is set with a maximum value of 250,000 queue 1 drops processed level to ensure uniform scaling from report to report. If the value is exceeded, SAS uses the highest value encountered to set this vertical axis maximum value.

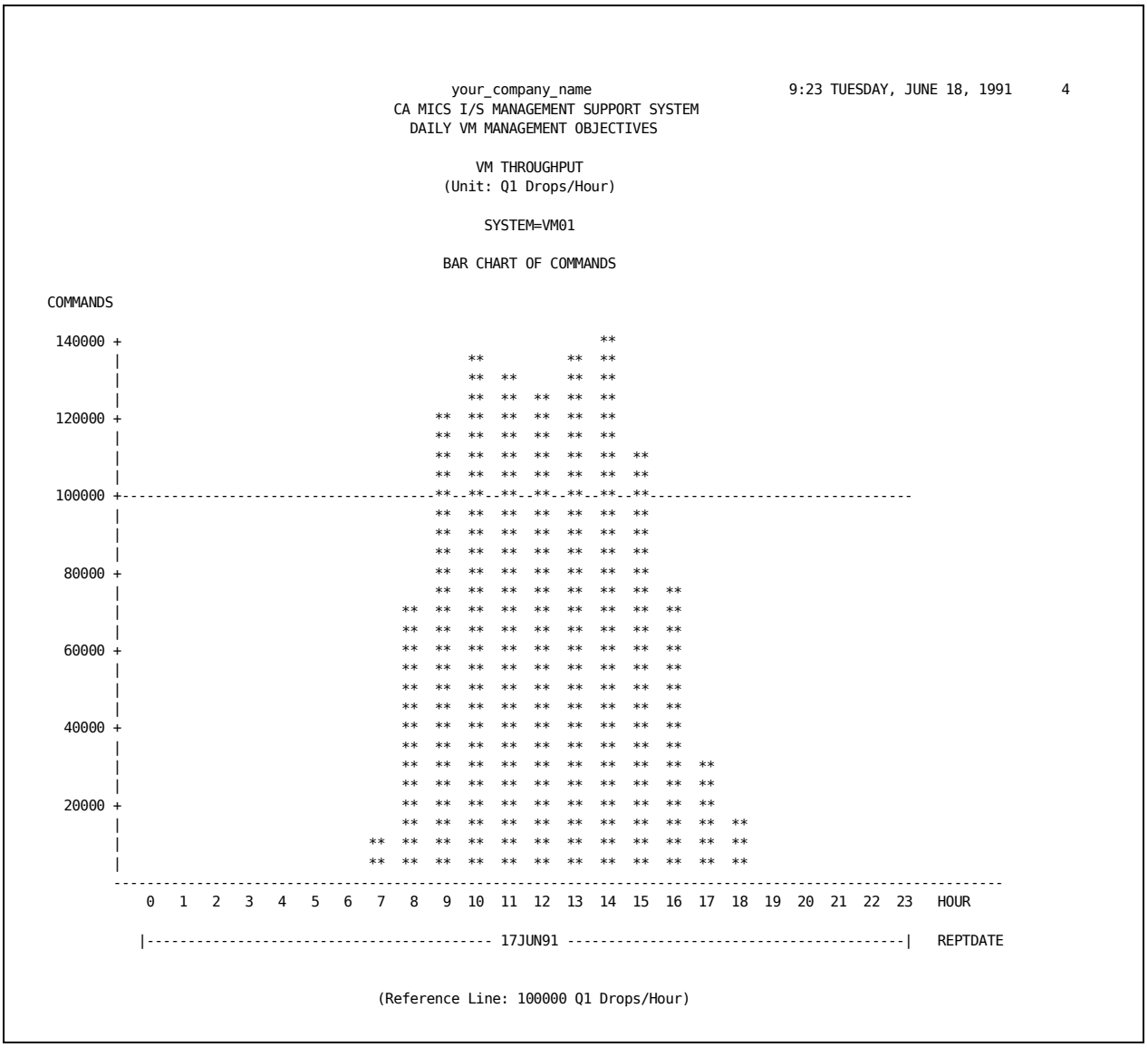


Figure 3-54. Daily VM Throughput Report

3.3.2 Weekly VM Reports

The weekly objective reports provide management with a way to determine whether the primary VM processing objectives are being attained. One-page charts illustrate the day's activity by prime-time hour within day of the week in reference to the established hourly objectives for response, command load, resource consumption, and availability.

An hour-of-the-day selection facility is provided, so that the weekly reports may reflect the entire 24 hours of each day, or any user-specified range.

The only user option that applies to this report is the Hourly Selection Range for Weekly Reports, which is the start and end hours of the day that are to be included in the Weekly VM Reports. It is suggested that the hours be set to include the prime time period of the day (e.g., 08 18).

&VMCWLHR and &VMCWHHR defines the low and high hours of the day for selection of the VM activity to be reported in the weekly reports. Therefore, to define hour 08 as the lowest hour to be selected and hour 18 as the highest, code:

```
%LET VMCWLHR = 08;
%LET VMCWHHR = 18;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

The following sections display and explain the Weekly VM Management Objective Reports:

1 - Report Format Descriptions
2 - Weekly VM Resource Consumption Report
3 - Weekly VM Service Report
4 - Weekly VM Availability Report
5 - Weekly VM Throughput Report - Commands

3.3.2.1 Report Format Descriptions

The Weekly Management Objective Reports are produced as a standard process in the CA MICS Weekly Job Stream. The SAS CHART procedure is used to produce the desired bar chart reports.

The heading lines contain the installation name (e.g., XYZ Manufacturing, Inc.), the frequency identification (e.g., WEEKLY VM MANAGEMENT OBJECTIVES), the objective being reported (e.g., VM RESOURCE CONSUMPTION WITHIN 2700 CPU SECONDS/HOUR), and the computing system (e.g., SYSID code) on which the VM processing was measured.

The format of the bar chart illustrates the hour and the day along the horizontal axis and the measurement being reported (e.g., availability), along the vertical axis. The day being reported is identified below the hours of the day, just under the horizontal axis.

The vertical axis is used to chart the measure being reported and it is with this axis that the management objective for this category is defined. The vertical axis is first established through the SAS CHART AXIS option which simply identifies the highest value to be represented on the vertical axis. The management objective is then illustrated through the SAS CHART REF option and formats a dashed line across the page to illustrate the management objective.

Consider the charting of the hourly objective for Q1 service (response time; see Section 3.3.2.3, Weekly VM Service Report). The title would describe the management objective as having average Q1 service within three seconds for all Q1 activity (usually equal to VM trivial command response time). The measure that would be charted is the average Q1 service time, shown for each hour of each day. Therefore, the vertical axis would be set up for a maximum value of three seconds, and the vertical reference line would be printed at the three-second level. This reference line then clearly depicts how the objective was satisfied or missed for each hour of each day.

A special note should be made at this time to point out that the location of the reference line on the vertical axis may be rounded for purposes of scaling the axis. Each vertical line may have a unit value of .10 seconds, so a reference line specification of 1.75 seconds would not fall directly on a vertical axis coordinate. In this case, the 1.75-second line would be shown as a line at 1.80 seconds.

The member WKVMCMBO in the sharedprefix.MICS.SOURCE library contains the SAS statements for producing the charts. The member #VMCMOBJ contains the SAS MACROs used by WKVMCMBO. The #VMCMOBJ member contains the specifications for defining the objectives to be displayed and may be modified to reflect your installation's unique management objectives. For defining your installation's objectives, see Section 3.3.4, VM Management Objective Report Options.

The remainder of this section explains and illustrates each of the standard Weekly VM Management Objective Reports.

3.3.2.2 Weekly VM Resource Consumption Report

This report quantifies the total VM CPU resource consumption in seconds per hour averaged across all processor engines in the supporting computing system for the week being reported. This is but another way to look at percentage resource consumption. That is, if the average number of CPU seconds consumed by each processor is 1800 seconds (out of a possible 3600 seconds), then approximately 50% of available processor capacity is being utilized.

The management objective to be tracked for evaluating the rate of resource consumption for VM is the average CPU time (in seconds) consumed per hour.

The only user option that applies to this report is the Weekly VM CPU Consumption Amount, which is the target CPU consumption (expressed in seconds) is used to set a reference line on the graph.

&VMCWSAX defines the CPU consumption chart axis. &VMCWSRF defines the target CPU consumption amount. Therefore, to define an objective (reference line) of 2700 CPU seconds consumed out of every hour (or 3600 CPU seconds), code:

```
%LET VMCWSAX = 3600;
%LET VMCWSRF = 2700;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-55 illustrates the Weekly VM Resource Consumption Report based on a management objective of consuming less than 2700 CPU seconds (on average) during any given hour. The sample report shown here quantifies the seconds of CPU consumption with the horizontal axis representing the hours of the day and the vertical axis charting the average seconds of CPU time consumed per processing engine.

The management objective for VM resource consumption is shown by the dashed horizontal reference line printed across the chart. In Figure 3-55, the reference line is shown at the 2700 CPU seconds consumed level of the vertical axis. Therefore, any hour for which the charted bar exceeds this reference line signals that the objective to contain resource consumption was missed for that hour.

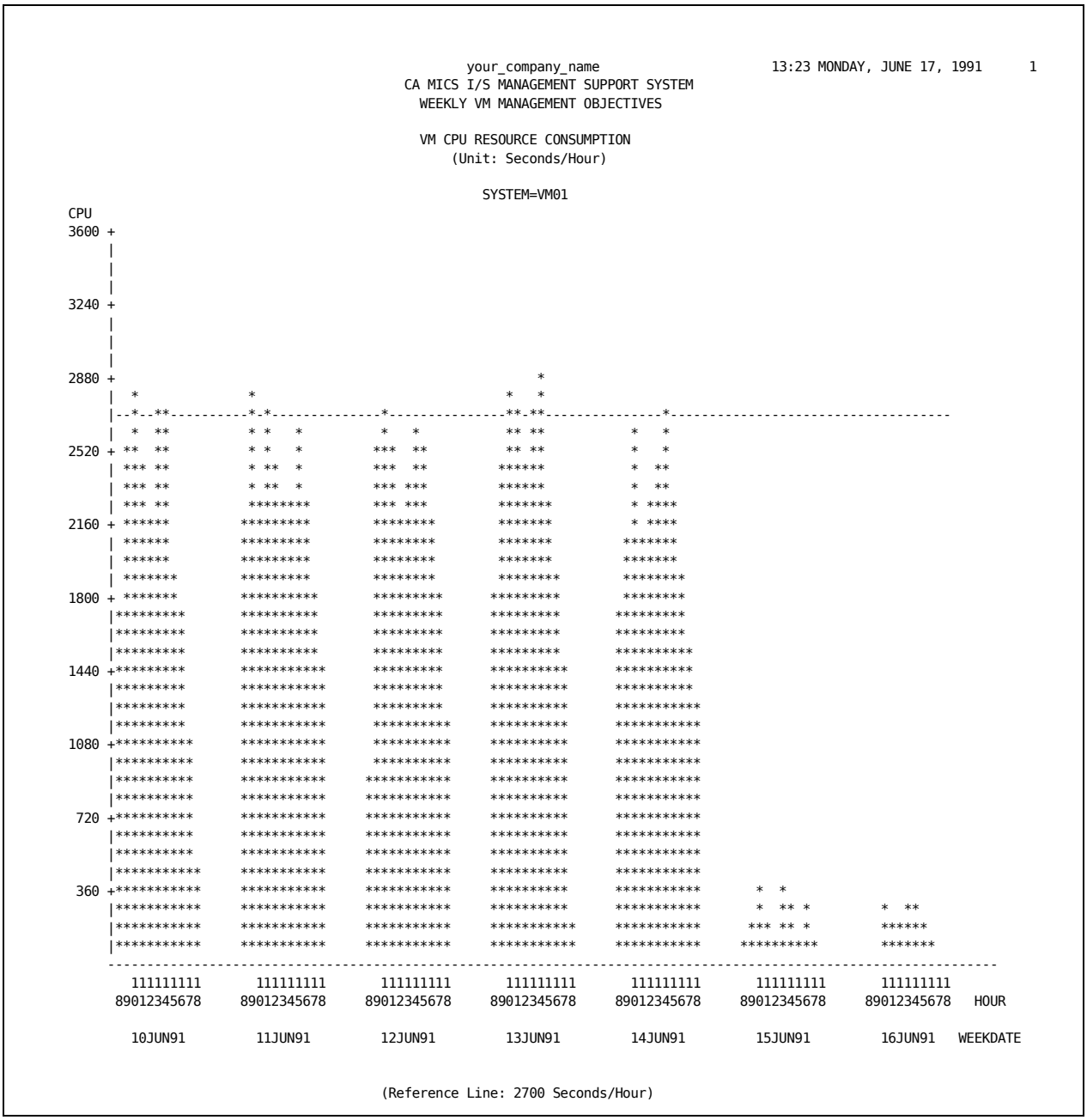


Figure 3-55. Weekly VM Resource Consumption Report Format - CPU Time

3.3.2.3 Weekly VM Service Report

This report quantifies the total VM service provided on an hourly basis for each day of a week, for each computing system.

The management objective to be tracked for evaluating the speed and consistency for all VM activity is an absolute measure of the number of VM queue 1 drops that occurred within a specified number of seconds.

The only user option that applies to this report is the Total Service Objective, which is the overall average of VM users that entered queue one and were subsequently dropped within the installation's service objective (e.g., average Q1 service within three seconds).

&VMCWSECX defines the vertical axis of the Q1 service time report. &VMCWSEC defines the target Q1 service time. Therefore, to define an average Q1 service of 1 second or less, code:

```
%LET VMCWSECX = 3.0;
%LET VMCWSEC = 1.0;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-56 illustrates the Weekly VM Service Report based on a management objective of maintaining an average VM Q1 service interval time of less than one second. The sample report shown here quantifies the service provided with the horizontal reference line representing the hours of each day in the report, and the vertical reference line charting the average VM Q1 service time in seconds, as identified in the title.

The management objective for total VM service is shown by the dashed horizontal reference line printed across the chart. In Figure 3-56, the reference line is shown at the three-second level of the vertical axis. Therefore, any hour for which the charted bar exceeds this reference line signals that the VM service objective was missed for that hour of that day.

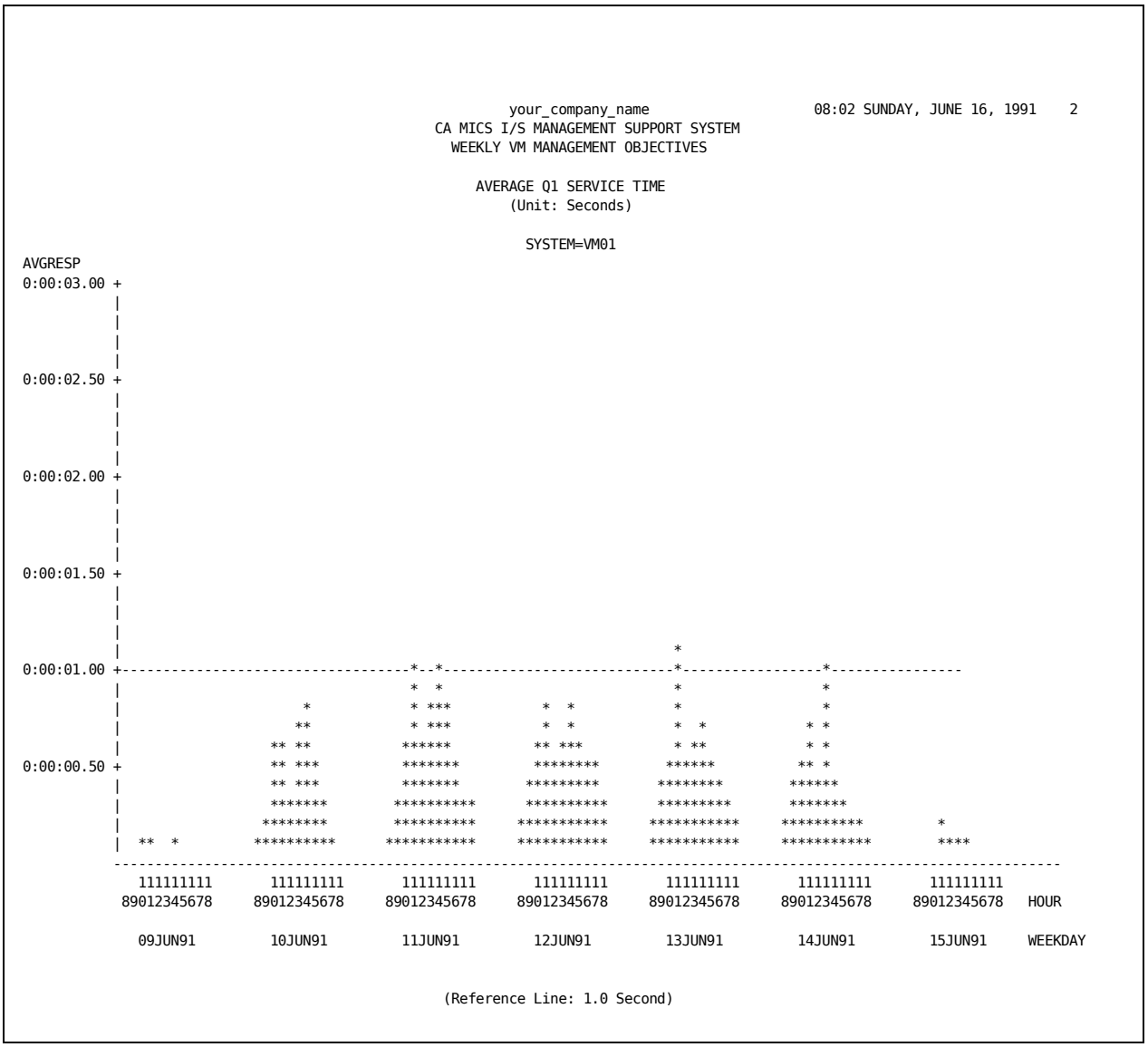


Figure 3-56. Weekly VM Average Q1 Service Times

3.3.2.4 Weekly VM Availability Report

This report shows the availability of a VM system on an hourly basis for the identified day and computing system.

The management objective to be tracked for evaluating the availability of the VM system is a measure of the number of minutes that the VM SCP was active within an hour of the day.

The only user option that applies to this report is the Hourly VM Uptime Limit, which is the target number of uptime minutes per hour, generally set to 60.

&VMCWUTX sets the vertical axis of the hourly uptime chart. &VMCWUT defines the hourly uptime target for daily and weekly reporting. Therefore, to define a target hourly uptime of 60 minutes per hour, code:

```
%LET VMCWUTX = 60;
%LET VMCWUT = 60;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-57 illustrates the Weekly VM Availability Report based on a management objective of no outage time (60 minutes uptime) per hour. The sample report shown here quantifies the system available time with the horizontal axis representing the hours of the seven days and the vertical axis charting the number of uptime minutes incurred.

The management objective is to not incur any outage time. Therefore, any time a charted bar fails to reach the 60 minute reference line, the VM availability objective was missed for that hour in the week.

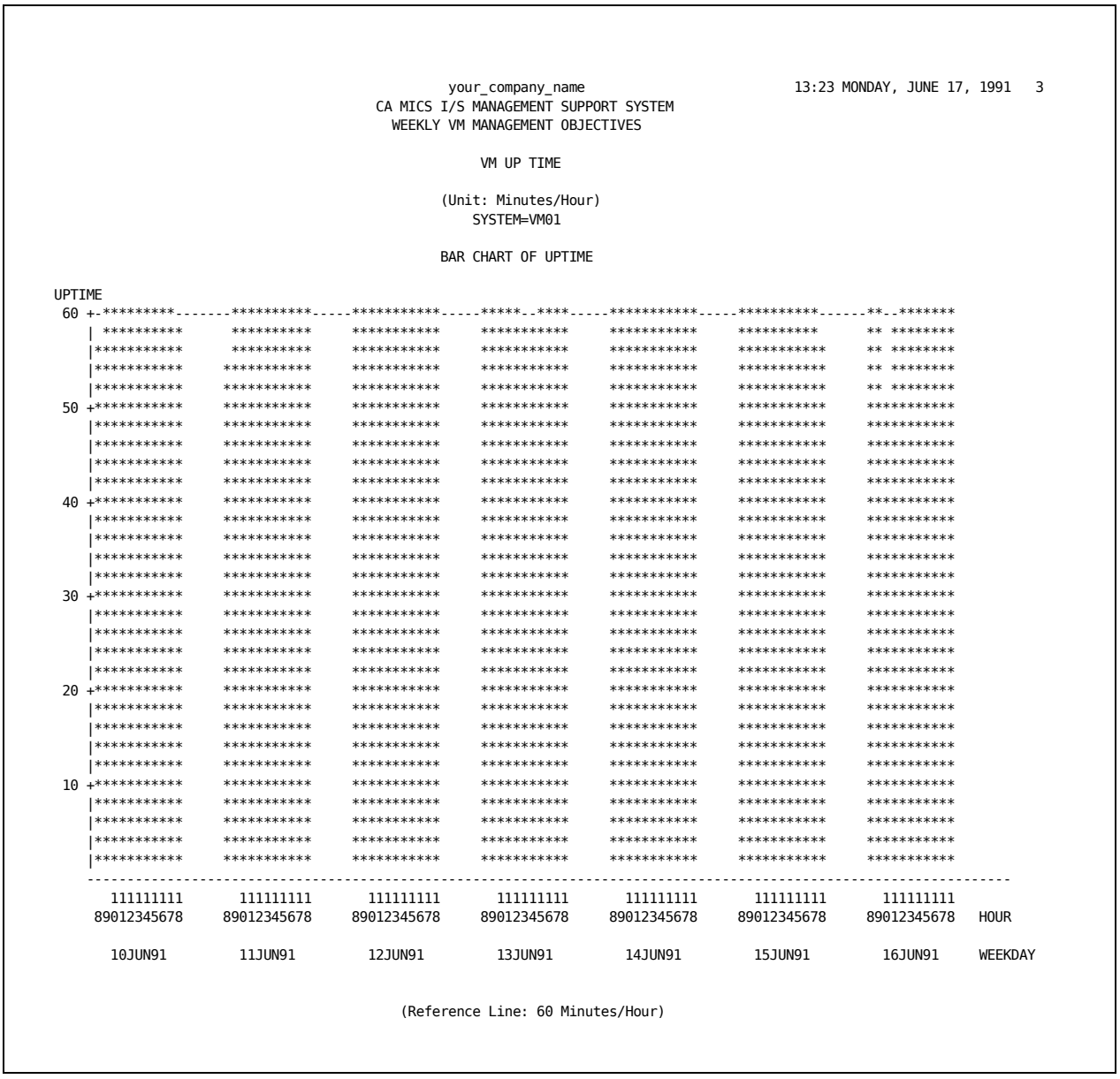


Figure 3-57. Weekly VM Availability Report

3.3.2.5 Weekly VM Throughput Report - Commands

This report quantifies the total VM throughput in terms of the number of Q1 drops processed on an hourly basis for the identified day and computing system over a week.

The management objective to be tracked for evaluating the VM throughput rate is a measure of the number of Q1 drops processed per hour. Figure 3-58 illustrates the Weekly VM Throughput Report based on a management objective of processing less than 100,000 Q1 drops in any given hour.

The following user options apply to this report:

Weekly VM Command Rate: The target and maximum number of Q1 drops processed per hour by VM. The target amount is used for a reference line on the graph and the maximum is used to set the maximum value on the graph's vertical axis.

```
&VMCWCAX and &VMCWCRF define the vertical axis maximum
     scaling and reference line for hourly Q1 drop rate
     execution in the weekly reports.

%LET VMCWCAX = 140000;
%LET VMCWCRF = 100000;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-58 quantifies the number of Q1 drops processed with the horizontal axis representing the hours of each day and the vertical axis charting the Q1 drops processed.

The management objective for VM throughput is shown by the dashed horizontal reference line printed across the chart. In Figure 3-58, the reference line is shown at the 100,000 Q1 drops processed level of the vertical axis. Therefore, any hour for which the charted bar exceeds this reference line signals that the objective to contain throughput was missed for that hour of that day.

The vertical axis is set with a maximum value of 140,000 Q1 drops processed level to ensure uniform scaling from report to report. If the value is exceeded, SAS uses the highest value encountered to set this vertical axis maximum value.

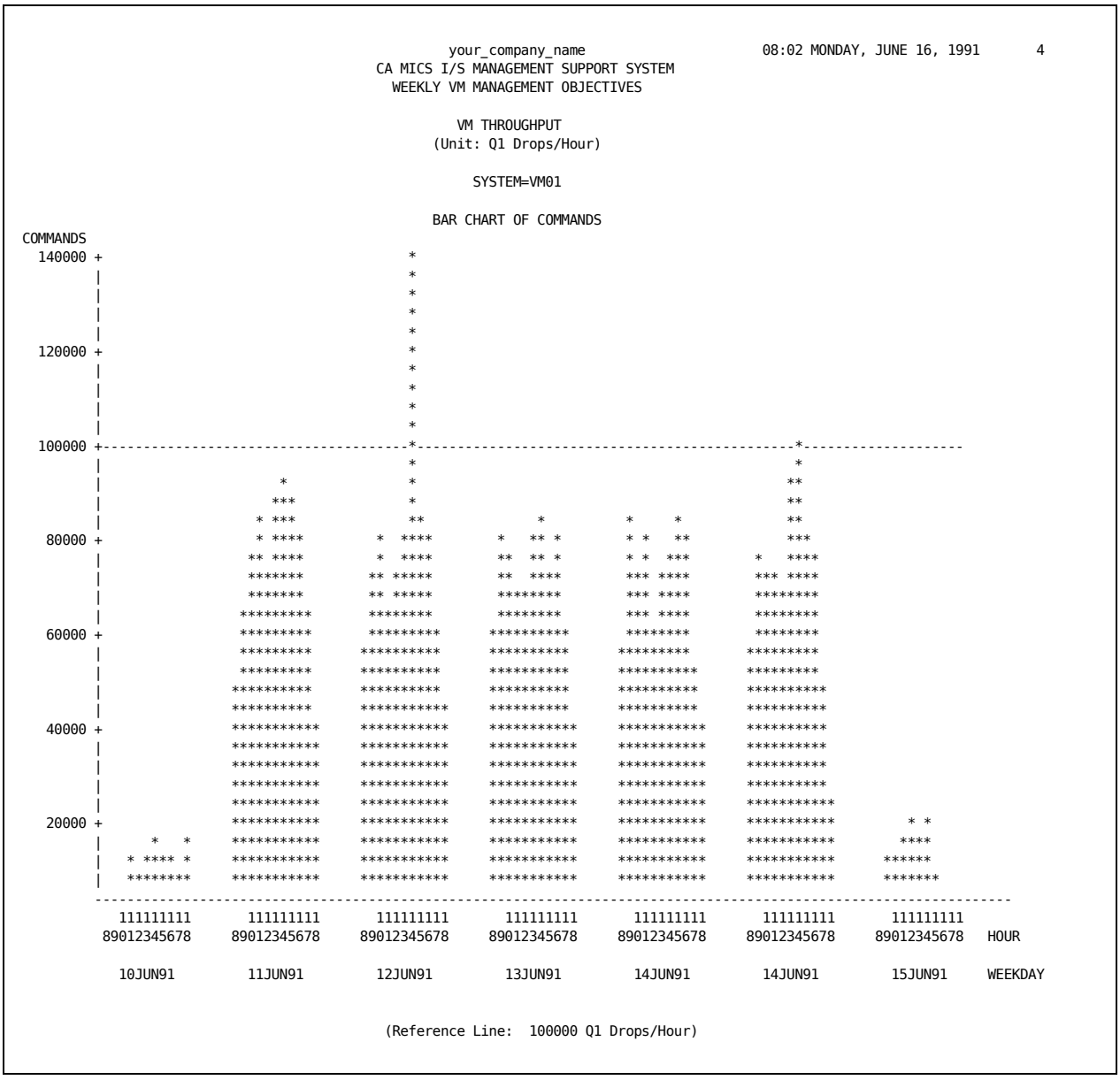


Figure 3-58. Weekly VM Throughput Report

3.3.3 Monthly VM Reports

The monthly objective reports provide management with a way to determine whether the primary VM processing objectives are being attained. One-page charts illustrate six months' activity by zone within the month in reference to the established zone objectives for response, command load, resource consumption, and availability.

A zone selection facility is provided, so that these monthly reports may reflect any user-specified range of zones.

The following user options apply to this report:

Monthly Report Zone Selection: The low and high zone numbers to be selected for the monthly reports are specified in these MACROs. It is suggested that the zones be set to include the non-holiday work week (usually zones 1 through 3).

```
&VMCLOZN and &VMCHIZN define the low and high zones of
     the month for selection of the VM activity to be
     reported in the monthly reports.

%LET VMCLOZN = 1;
%LET VMCHIZN = 3;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

The following sections display and explain the Monthly VM Management Objective Reports:

```
1 - Report Format Descriptions
2 - Monthly VM Resource Consumption Report
3 - Monthly VM Service Report
4 - Monthly VM Availability Report
5 - Monthly VM Throughput Report
```

3.3.3.1 Report Format Descriptions

The Monthly Management Objective Reports are produced as a standard process in the CA MICS Monthly Job Stream. The SAS CHART procedure is used to produce the desired bar-chart reports.

The heading lines contain the installation name (e.g., XYZ Manufacturing, Inc.), the frequency identification (e.g., MONTHLY VM MANAGEMENT OBJECTIVES), the objective being reported (e.g., VM RESOURCE CONSUMPTION WITHIN 2700 CPU SECONDS/HOUR), and the computing system (e.g., SYSID code) on which the VM processing was measured.

The format of the bar chart illustrates the zone and the month along the horizontal axis and the measurement being reported (e.g., availability) along the vertical axis. The month being reported is identified below the zones of the month just under the horizontal axis.

The vertical axis is used to chart the measure being reported and it is with this axis that the management objective for this category is defined. The vertical axis is first established through the SAS CHART AXIS option which simply identifies the highest value to be represented on the vertical axis. The management objective is then illustrated through the SAS CHART REF option and formats a dashed line across the page to illustrate the management objective.

Consider the charting of the hourly objective for Q1 service (response timel; see Section 3.3.3.3, Monthly VM Service Report). The title would describe the management objective as having average Q1 service within three seconds for all Q1 activity (usually equal to VM trivial command response time). The measure that would be charted is the average Q1 service time, shown for each selected zone. The vertical axis, therefore, would be set up for a maximum value of three seconds, and the vertical reference line would be printed at the three-second level. This reference line then clearly depicts how the objective was satisfied or missed for each zone of each month.

A special note should be made at this time to point out that the location of the reference line on the vertical axis may be rounded for purposes of scaling the axis. Each vertical line may have a unit value of .10 seconds, so a reference line specification of 1.75 seconds would not fall directly on a vertical axis coordinate. In this case, the 1.75-second line would be shown as a line at 1.80 seconds.

The member MNVMCMBO in the sharedprefix.MICS.SOURCE library contains the SAS statements for producing the charts. The member #VMCMOBJ contains the SAS MACROs used by MNVMCMBO. The #VMCMOBJ member contains the specifications for defining the objectives to be displayed and may be modified to reflect your installation's unique management objectives. For defining your installation objectives see Section 3.3.4.1, VM Report Options.

The remainder of this section explains and illustrates each of the standard Monthly VM Management Objective Reports.

3.3.3.2 Monthly VM Resource Consumption Report

This report quantifies the total VM CPU resource consumption in seconds per hour averaged across all processor engines and within zone on the supporting computing system for a period of five months. This is but another way to look at percentage resource consumption. That is, if the average number of CPU seconds consumed by each processor is 1800 seconds (out of a possible 3600 seconds), then approximately 50% of available processor capacity is being utilized.

The management objective to be tracked for evaluating the rate of resource consumption for VM is the average CPU time (in seconds) consumed per hour across a zone.

The only user option that applies to this report is the Monthly VM CPU Consumption Amount, which is the target CPU consumption (expressed in seconds) is used to set a reference line on the graph.

&VMCMSAX defines the CPU consumption chart axis. &VMCMSRF defines the target CPU consumption percentage. Therefore, to define a target CPU consumption of 2700 CPU seconds per hour, code:

```
%LET VMCMSAX = 3600;
%LET VMCMSRF = 2700;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-59 illustrates the Monthly VM Resource Consumption Report based on a management objective of consuming less than 2700 CPU seconds (on average) per hour during any given zone. The sample report shown here quantifies the average seconds of CPU consumption per hour within zone, with the horizontal axis representing the zones of each month and the vertical axis charting the average seconds of CPU time consumed per processing engine across the hours falling within a given zone.

The management objective for VM resource consumption is shown by the dashed horizontal reference line printed across the chart. In Figure 3-59, the reference line is shown at the 2700 CPU seconds consumed level of the vertical axis. Therefore, any zone for which the charted bar exceeds this reference line signals that the objective to contain resource consumption was missed for that zone.

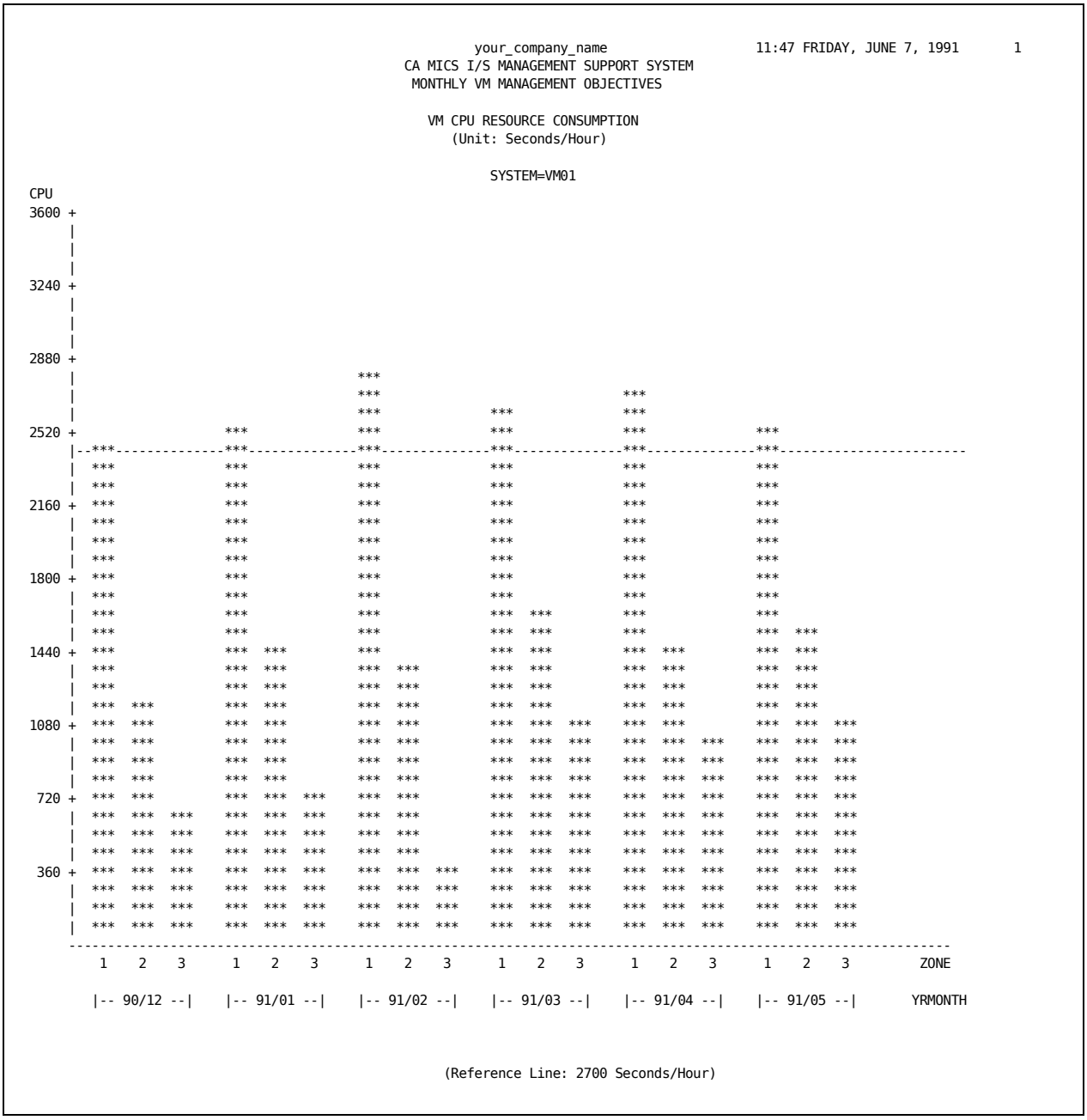


Figure 3-59. Monthly VM Resource Consumption

3.3.3.3 Monthly VM Service Report

This report quantifies the total VM service provided on a zone basis for a period of months, for each computing system.

The management objective to be tracked for evaluating the speed and consistency for all VM activity is an absolute measure of the number of VM Q1 drops that occurred within a specified number of seconds.

The only user option that applies to this report is the Total Service Objective, which is the average over all VM users that entered queue one and were subsequently dropped within the installation's service objective (e.g., average Q1 service within three seconds).

&VMCMSECX defines the vertical axis of the Q1 service time report. &VMCMSEC defines the target Q1 service time. Therefore, to measure the number of VM Q1 drops that occur within each three-second period, code:

```
%LET VMCMSECX = 3.0;
%LET VMCMSEC = 1.0;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-60 illustrates the Monthly VM Service Report based on a management objective of maintaining an average VM Q1 service interval time of less than one second. The sample report shown here quantifies the service provided with the horizontal axis representing the zones of each month in the report and the vertical axis charting the average VM Q1 service time in seconds, as identified in the title.

The management objective for total VM service is shown by the dashed horizontal reference line printed across the chart. In Figure 3-60 , the reference line is shown at the three-second level of the vertical axis. Therefore, any zone for which the vertical bar exceeds this reference line signals that the VM service objective was missed for that zone of that month.

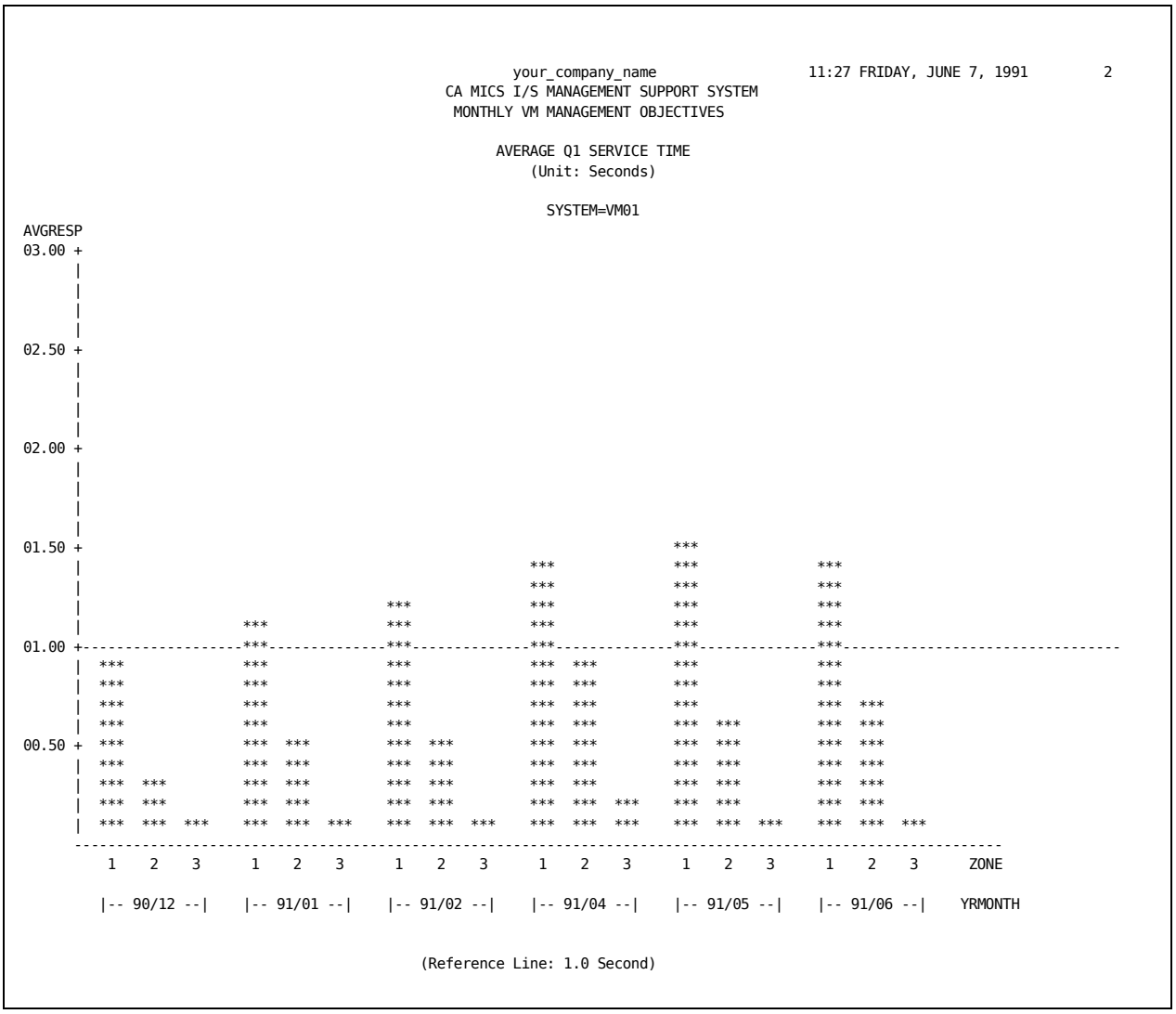


Figure 3-60. Monthly VM Average Q1 Service

3.3.3.4 Monthly VM Availability Report

This report shows the availability of a VM system on an hourly basis for the identified day and computing system.

The management objective to be tracked for evaluating the availability of the VM system is a percentage of the total time available in the zone of the month represented by the uptime measured for the VM system.

The only user option that applies to this report is the Monthly VM Up Time Percentage, which is the target uptime percent value, generally set to 100, and charted on an axis of 100.

&VMCMPCT defines the uptime percentage chart axis. &VMCRPCT defines the uptime percentage target. Therefore, to define a target uptime of 100 percent and chart it on an axis of 100, code:

```
%LET VMCMPCT = 100;
%LET VMCRPCT = 100;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-62 illustrates the Monthly VM Availability Report based on a management objective of no outage time (100% uptime) per zone. The sample report shown here quantifies the system available time with the horizontal axis representing the zones of the five months and the vertical axis charting the percent of uptime.

The management objective is to not incur any outage time. Therefore, any time a charted bar fails to reach the 100 percent reference line, the VM availability objective was missed for that zone in the month.

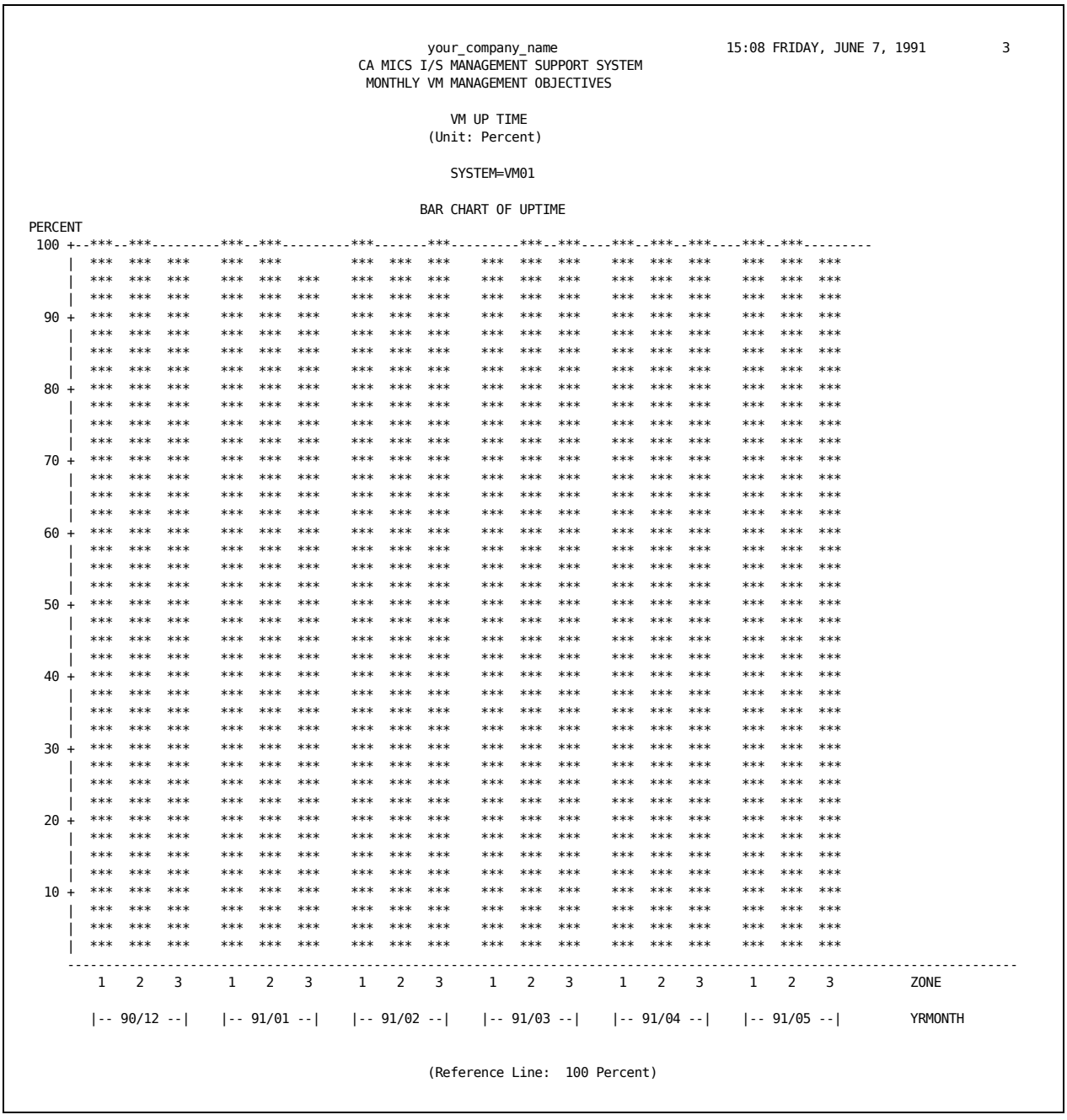


Figure 3-61. Monthly VM Availability Report

3.3.3.5 Monthly VM Throughput Report

This report quantifies the total VM throughput in terms of the number of Q1 drops processed on a zone basis for the identified day and computing system over six months.

The management objective to be tracked for evaluating the VM throughput rate is an average of the number of Q1 drops processed per hour.

The only user option that applies to this report is the Monthly VM Command Rate, which is the target and maximum average number of Q1 drops processed per hour, across each zone, by VM. The target amount is used for a reference line on the graph and the maximum average is used to set the maximum value on the graph's vertical axis.

&VMCMCAX and &VMCMCRF define the vertical axis maximum scaling and reference line for hourly Q1 drop rate execution in the monthly reports. Therefore, to define an average of less than 100,000 Q1 drops per hour with a maximum measurement of 140,000, code:

```
%LET VMCMCAX = 140000;
%LET VMCMCRF = 100000;
```

A full description of the user options is provided in Section 3.3.4.1, VM Report Options.

Figure 3-62 illustrates the Monthly VM Throughput Report based on a management objective of averaging less than 100,000 Q1 drops per hour in any given zone. The sample report shown here quantifies the number of Q1 drops processed with the horizontal axis representing the zones of each month and the vertical axis charting the Q1 drops processed.

The management objective for VM throughput is shown by the dashed horizontal reference line printed across the chart. In Figure 3-62, the reference line is shown at the 100,000 Q1 drops processed level of the vertical axis. Therefore, any hour for which the vertical bar exceeds this reference line signals that the objective to contain throughput was missed for that zone of that month.

The vertical axis is set with a maximum value of 140,000 Q1 drops processed level to ensure uniform scaling from report to report. If the value is exceeded, SAS uses the highest value encountered to set this vertical axis maximum value.

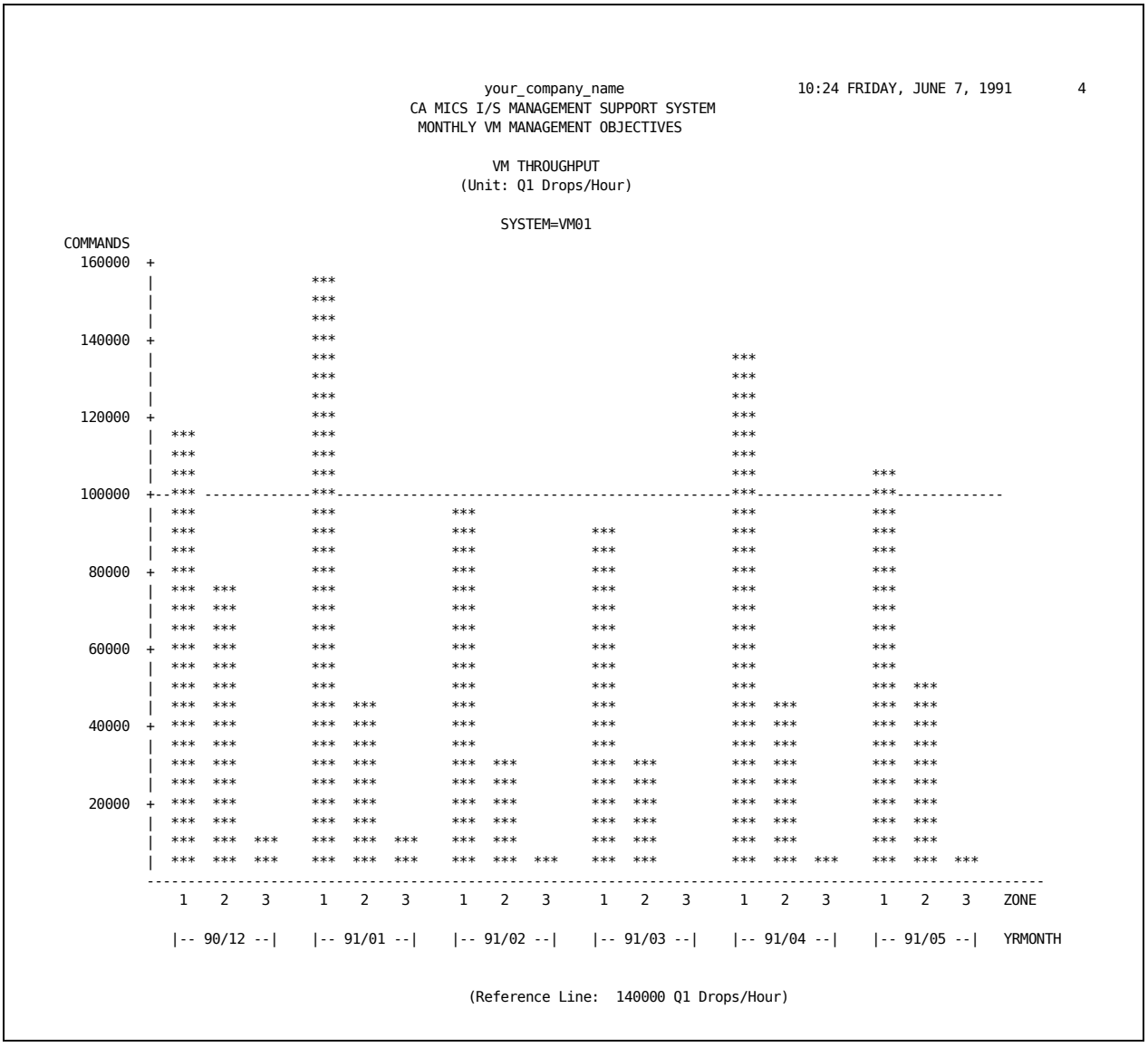


Figure 3-62. Monthly VM Throughput Report Format - Q1 Drops

3.3.4 VM Management Objective Report Options

The Management Objective Reports assume that a set of objectives related to the operation of VM have been determined. Often this may not be the case. This section describes how the management objectives may be specified in the reporting process once they have been defined by management. First, however, a discussion of how the objectives may be established is provided.

Numbers that will be used as management objectives should not be defined without some analysis of the installation's historical VM performance.

The CA MICS Database provides a useful information base for analyzing historical performance. We suggest that statistical analysis procedures be used to study past performance.

The same data sources mentioned above may be used to evaluate the measures required in determining specific management objectives. Specifically, the SAS procedures UNIVARIATE and FREQ may be used to provide a reasonable statistical analysis of an individual measure, providing the analyst with mean, standard deviation, and percentile values. It is our opinion that the percentile values may be the most applicable to the process of determining the management objective's target values.

The database files that should be used in analyzing VM activity for purposes of establishing the VM Report Options are listed according to the reports to which they apply:

```
Daily   - &VMCD..VMXSGP01
Weekly  - &VMCD..VMXSGP01-10
            (selects last week only)
Monthly - &VMCD..VMXSGP01-06
```

In each case, the data source is the VM Performance Data File, which contains data from the System Global Data File, which contains VM performance data.

The last input essential in defining management objectives for VM rests with the perspective of VM users as to what minimum level of service, availability, and throughput they require.

The integration of this information with data processing management's own priorities and requirements will establish

the base for defining the management objectives for VM.

The following sections describe specifically how an installation is able to specify the numerical values used in these reports:

1 - VM Report Options
2 - Report Options MACRO Example

3.3.4.1 VM Report Options

The following options are provided for defining VM Management Objective Reports. The names of the SAS MACRO variables used for option definition are also given.

DATA SELECTION MACRO DEFINITIONS:

Hourly Selection Range for Daily Reports: The start and end hours of the day that are to be included in the Daily VMC Reports. We suggest that the hours be set to include the entire day (i.e., 00 23).

&VMCDLHR and &VMCDHHR define the low and high hours of the day for selection of the VM activity to be reported in the daily reports.

Hourly Selection Range for Weekly Reports: The start and end hours of the day that are to be included in the Weekly VMC Reports. We suggest that the hours be set to include only the prime-time hours (e.g., 08 16).

&VMCWLHR and &VMCWHHR define the low and high hours of the days for selection of the VM activity to be reported in the weekly reports.

Zone Selection for Monthly Reporting: The zone or range of zones that are to be included in the Monthly VM Reports. Normally all zones should be reported and therefore the range should be 1 to 9. The VM Reports could be limited to only the range of zones required.

&VMCLOZN and &VMCHIZN define the low and high zones in the months input for selection of the VM activity to be reported in the monthly reports.

```
Q1 SERVICE MACRO DEFINITIONS:

Average Q1 Service Objective:  The average service time of
all VM Q1 requests that were satisfied within a specified
number of seconds to reflect the installation's response
objective for total VM activity (e.g., maintain the average
at a level below three seconds)

    &VMCDSECX and &VMCSEC define maximum scaling for vertical
    axis and the reference line for the average Q1 service
    time in the daily reports.

    &VMCWSECX and &VMCWSEC define maximum scaling for
    vertical axis and the reference line for the average Q1
    service time in the weekly reports.

    &VMCMSECX and &VMCMSEC define maximum scaling for
    vertical axis and the reference line for the average Q1
    service time in the monthly reports.


AVAILABILITY MACRO DEFINITIONS:

Hourly VM Outage Limits:  The target number of uptime minutes
per hour, generally set to 60, and the downtime objective per
hour, which is generally set to 0 minutes

    &VMCDUTX and &VMCUT define maximum scaling for the
    vertical axis and the hourly uptime target reference line
    for daily reports.

    &VMCWUTX and &VMCWUT define maximum scaling for the
    vertical axis and the hourly uptime target reference line
    for weekly reports.

Monthly Availability Percentage:  The percent of uptime VM is
targeted to be available in any given month for the selected
zones

    &VMCMPCT and &VMCRPCT define maximum scaling for the
    vertical axis and the reference line for the percent of
    VM uptime by zone in the monthly reports.
```

```
THROUGHPUT MACRO DEFINITIONS:

Hourly VM Q1 Drop Rate:  The target and maximum number of Q1
drops processed per hour by VM.  The target amount is used
for a reference line on the graph and the maximum is used to
set the maximum value on the graph's vertical axis.

    &VMCCAX and &VMCCREF define maximum scaling for the
    vertical axis and the reference line for hourly command
    rate execution in the daily reports.

    &VMCWCAX and &VMCWCRF define maximum scaling for the
    vertical axis and the reference line for commands
    executed by hour in the weekly reports.

    &VMCMCAX and &VMCMCRF define maximum scaling for the
    vertical axis and the reference line for commands
    executed by zone in the monthly reports.


CPU RESOURCE CONSUMPTION MACRO DEFINITIONS:

VM CPU Consumption Amount:      The target and maximum CPU
seconds per hour for VM.  The target amount is used for a
reference line on the graph and the maximum is used to set
the maximum value on the graph's vertical axis.

    &VMCSAX and &VMCSREF define maximum scaling for the
    vertical axis and the reference line for CPU consumption
    by hour in the daily reports.

    &VMCWSAX and &VMCWSRF define maximum scaling for the
    vertical axis and the reference line for CPU consumption
    by hour in the weekly reports.

    &VMCMSAX and &VMCMSRF define maximum scaling for the
    vertical axis and the reference line for CPU consumption
    by zone in the monthly reports.

The VM Report Options, which are defined as SAS MACRO
variables, are contained in the source member #VMCMOBJ in the
sharedprefix.MICS.SOURCE library.  The next section shows the
member and describes how to modify the options.
```

3.3.4.2 Report Options MACRO Example

Once the management objectives for VM have been established, they can easily be incorporated into the CA MICS VM and CMS Analyzer reports.

The specification of the management objectives to be used in the reporting process are defined in the #VMCMOBJ member contained in the sharedprefix.MICS.SOURCE library. This member consists of a series of SAS MACRO variables which define values for the required objectives.

The #VMCMOBJ member is distributed with a set of default values. It is the responsibility of the CA MICS System Administrator to initially tailor the values and to maintain them for subsequent installation changes and, of course, for CA MICS system maintenance itself.

Prior to discussing the specification of the objectives, several points should be explained. All values input by the user for defining objectives are specified as SAS MACRO variables for ease of incorporation into the reporting process. A brief explanation of SAS MACRO variables is provided. For instance, the objective for average Q1 service time might be defined as:

```
%LET VMCSEC = 3;
```

A SAS MACRO variable is defined by the word %LET followed in order by the name used to reference the variable, an equal (=) sign, the value that is to be substituted any time the variable is used in a SAS program, and the semicolon; which signifies the end of the variable definition. For example, the name of the variable in the above statement is &VMCSEC. Any time that &VMCSEC is referenced in the CA MICS code, the value 3 is substituted.

In changing the objectives used in the reports, you are simply changing the appropriate MACRO variable's value to reflect what should be used in the reports for your installation.

You must ensure that the word %LET, the CA MICS MACRO variable name, and the ending semicolon ARE NOT ALTERED IN ANY WAY, and the only item changed is the value following the equal sign.

The #VMCMOBJ member is organized to enable you to quickly identify the values that require changing and is illustrated in Figure 3-63.

```
                    /* AVERAGE Q1 RESPONSE SPECIFICATIONS */
 %LET VMCDSECX = ? ; /* DAILY   : AXIS                     */
 %LET VMCSEC   = ? ; /*         : REFERENCE                */
 %LET VMCWSECX = ? ; /* WEEKLY  : AXIS                     */
 %LET VMCWSEC  = ? ; /*         : REFERENCE                */
 %LET VMCMSECX = ? ; /* MONTHLY : AXIS                     */
 %LET VMCMSEC  = ? ; /*         : REFERENCE                */
                    /* UPTIME SPECIFICATIONS (IN MINUTES) */
 %LET VMCDUTX  = ? ; /* DAILY   : AXIS                     */
 %LET VMCUT    = ? ; /*         : REFERENCE                */
 %LET VMCWUTX  = ? ; /* WEEKLY  : AXIS                     */
 %LET VMCWUT   = ? ; /*         : REFERENCE                */
                    /* CPU CONSUMPTION SPECIFICATION      */
 %LET VMCSAX   = ? ; /* DAILY   : AXIS                     */
 %LET VMCSREF  = ? ; /*         : REFERENCE                */
 %LET VMCWSAX  = ? ; /* WEEKLY  : AXIS                     */
 %LET VMCWSRF  = ? ; /*         : REFERENCE                */
 %LET VMCMSAX  = ? ; /* MONTHLY : AXIS                     */
 %LET VMCMSRF  = ? ; /*         : REFERENCE                */
                    /* THROUGHPUT SPECIFICATION           */
 %LET VMCCAX   = ? ; /* DAILY   : AXIS                     */
 %LET VMCCREF  = ? ; /*         : REFERENCE                */
 %LET VMCWCAX  = ? ; /* WEEKLY  : AXIS                     */
 %LET VMCWCRF  = ? ; /*         : REFERENCE                */
 %LET VMCMCAX  = ? ; /* MONTHLY : AXIS                     */
 %LET VMCMCRF  = ? ; /*         : REFERENCE                */
                    /* TIME SELECTION SPECIFICATION       */
 %LET VMCDLHR  = ? ; /* DAILY   : START HOUR               */
 %LET VMCDHHR  = ? ; /*         : END HOUR                 */
 %LET VMCWLHR  = ? ; /* WEEKLY  : START HOUR               */
 %LET VMCWHHR  = ? ; /*         : END HOUR                 */
 %LET VMCLOZN  = ? ; /* MONTHLY : START ZONE               */
 %LET VMCHIZN  = ? ; /*         : END ZONE                 */
                    /* AVAILABILITY SPECIFICATIONS        */
 %LET VMCMPCT  = ? ; /* MONTHLY : TARGET PERCENT           */
 %LET VMCRPCT  = ? ; /*         : REFERENCE                */
```

Figure 3-63. VM Report Options in #VMCMOBJ

Figure 3-64 shows how the VM options have been defined in #VMCMOBJ in the distributed version of the member.

```
                    /* AVERAGE Q1 RESPONSE SPECIFICATIONS */
 %LET VMCDSECX = 0.8;  /* DAILY   : AXIS                   */
 %LET VMCSEC   = 0.1;  /*         : REFERENCE              */
 %LET VMCWSECX = 0.8;  /* WEEKLY  : AXIS                   */
 %LET VMCWSEC  = 0.1;  /*         : REFERENCE              */
 %LET VMCMSECX = 0.8;  /* MONTHLY : AXIS                   */
 %LET VMCMSEC  = 0.1;  /*         : REFERENCE              */
                    /* UPTIME SPECIFICATIONS (IN MINUTES) */
 %LET VMCDUTX  = 60 ;  /* DAILY   : AXIS                   */
 %LET VMCUT    = 60 ;  /*         : REFERENCE              */
 %LET VMCWUTX  = 60 ;  /* WEEKLY  : AXIS                   */
 %LET VMCWUT   = 60 ;  /*         : REFERENCE              */
                    /* CPU CONSUMPTION SPECIFICATION      */
 %LET VMCSAX  = 3600;  /* DAILY   : AXIS                   */
 %LET VMCSREF = 2700;  /*         : REFERENCE              */
 %LET VMCWSAX = 3600;  /* WEEKLY  : AXIS                   */
 %LET VMCWSRF = 2700;  /*         : REFERENCE              */
 %LET VMCMSAX = 3600;  /* MONTHLY : AXIS                   */
 %LET VMCMSRF = 2700;  /*         : REFERENCE              */
                    /* THROUGHPUT SPECIFICATION           */
 %LET VMCCAX  = 50000; /* DAILY   : AXIS                   */
 %LET VMCCREF = 20000; /*         : REFERENCE              */
 %LET VMCWCAX = 50000; /* WEEKLY  : AXIS                   */
 %LET VMCWCRF = 20000; /*         : REFERENCE              */
 %LET VMCMCAX = 50000; /* MONTHLY : AXIS                   */
 %LET VMCMCRF = 20000; /*         : REFERENCE              */
                    /* TIME SELECTION SPECIFICATION       */
 %LET VMCDLHR =  7;    /* DAILY   : START HOUR             */
 %LET VMCDHHR = 17;    /*         : END HOUR               */
 %LET VMCWLHR =  7;    /* WEEKLY  : START HOUR             */
 %LET VMCWHHR = 17;    /*         : END HOUR               */
 %LET VMCLOZN =  1;    /* MONTHLY : START ZONE             */
 %LET VMCHIZN =  3;    /*         : END ZONE               */
                    /* AVAILABILITY SPECIFICATIONS        */
 %LET VMCMPCT = 100;   /* MONTHLY : TARGET PERCENT         */
 %LET VMCRPCT = 100;   /*         : REFERENCE              */
```

Figure 3-64. Sample #VMCMOBJ Contents

Chapter 4: EXCEPTIONS

Exception reports provide a concise, integrated, and itemized list of the problems that can impact a site in terms of availability, service, workload, security, and performance.

Chapter 2 of the Standard Reports Guide describes the philosophy of and requirement for exception reporting, a general description of the process, the methodology for qualifying exceptions, an explanation of the operational flow, a description of the reports, and a plan for the using the reports.

This chapter describes the specific exceptions related to the Analyzer, including a list of the exceptions provided and the more detailed descriptions required to explain the exception definition process.

This section contains the following topics:

4.1 Introduction (see page 192)
4.2 Standard Exceptions (see page 194)
4.3 Exception Descriptions (see page 195)

4.1 Introduction

The exception reporting process reads data from the available monitoring sources (e.g., VMC, SMF, CA TSO/MON PM), compares this activity against predefined thresholds, and provides an integrated list of exceptions for potential problem areas.

Figure 4-1 illustrates the operation of the exception reporting process. The user-defined values comprise the part of the system that tailors and modifies the exceptions to uniquely address a site's requirements.

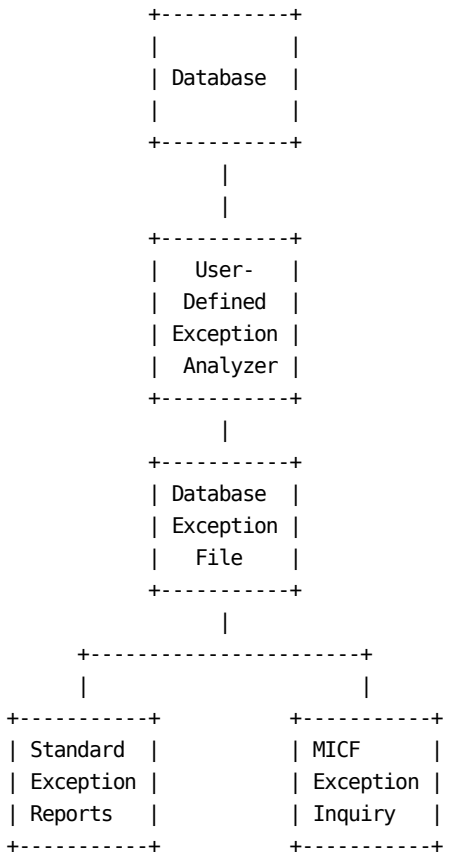


Figure 4-1. Exception Reporting Operational Flow

On a daily basis, the exception files contained in the database are processed to detect the defined exception conditions. The Exception Analyzer performs this task by using individual test routines to identify the exceptions. Each test routine requires user-defined values that tailor

the exception test to the site requirements.

An exception analysis completely defines the tests made to determine the exception condition and the criteria that define and classify the exception for reporting. The user may easily modify, or in some cases even add, values to the exception criteria to better meet a site's exception reporting requirements.

These tests are called Exception Test Routines and are stored in the source library prefix.MICS.USER.SOURCE in the member DYVMCEXC. An exception group exists for each file that is to be processed by the Exception Analyzer and consists of the exception data that is to be processed for a product.

Figure 4-2 lists the range of Exception Numbers that may be defined for each data group and file. The exception code is organized sequentially by exception numbers within the DYVMCEXC program and consists of standard identification definitions (e.g., severity level), exception-dependent criteria (e.g., amount of CPU time used), or standard selection facilities (e.g., selection of prime-time hours only). Modification of existing exception analysis can be done easily. The exception name and number delineate the start of its definition within DYVMCEXC. The value to be modified can then be located and changed directly within the SAS code.

Exception Number Range	Database File Name
08050-08055	Domain 1 VM Monitor records
08056-08061	DETAIL.VMXSGP01
08062-08063	DETAIL.VMXSCF01
08064	Domain 1 Record 11 VM monitor records
08065-08066	DETAIL.VMXLPR01
08067-08068	Domain 1 VM Monitor Event records

Figure 4-2. Exception Number Table

4.2 Standard Exceptions

This section lists the standard exceptions provided for the CA MICS Analyzer Option for VM/CMS. Figure 4-3 lists the standard VMC exceptions by number, severity level, management area code, and descriptive title.

Number	Severity	Management Area	Exception Description
08004	Warning	Security	User Linked Unowned Protected Minidisk
08005	Warning	Security	Limit of Invalid Link Passwords Reached
08006	Warning	Security	User Disconnected from the System
08050	Impacting	Performance	VM Monitor Event Profile
08051	Impacting	Performance	VM Monitor Event Collection Altered
08052	Critical	Performance	VM Monitor Event Suspension, Records Lost
08054	Impacting	Performance	VM Monitor Sample Profile
08055	Impacting	Performance	VM Monitor Sample Collection Altered
08056	Warning	Availability	VM Guest Survival Storage Exhausted
08057	Warning	Performance	VM Available Frame List Empty
08058	Warning	Performance	VM Demand Scan Not Complete
08059	Critical	Availability	VM Page Slots > 90% Full
08060	Critical	Availability	VM Spool Slots > 90% Full
08061	Warning	Performance	VM Excessive Absolute Shares in Dispatch
08062	Critical	Availability	VM CP ABENDed at Last Termination
08063	Impacting	Availability	VM System IPL Detected
08064	Critical	Performance	VM Monitor Data Truncated, Records Lost
08065	Warning	Performance	Excessive LPAR Attributable Management Time
08066	Warning	Performance	Excessive LPAR System Management Time
08067	Impacting	Performance	CPU Capability is Changed
08068	Impacting	Performance	Memory Configuration Change

Figure 4-3. Exception List

4.3 Exception Descriptions

This section provides an in-depth description of each standard exception available in the exception reporting process. The exceptions are organized by number and appear sequentially, beginning with 08050.

The description provides the title, number, purpose, rationale, and definition of each exception, to help you understand the meaning and use of the information.

EXC08004: User Linked Unowned Protected Minidisk

```
+-----------------+
| E X C 0 8 0 0 4 |
+-----------------+

FILE:        VMC Exception Activity File (VMCVXC01)

PURPOSE:     Identifies when a user has successfully linked
             to a minidisk that is protected and not owned by
             that user.

RATIONALE:   An installation may want to identify the
             situation where many users are linking to a
             sensitive minidisk because the password has
             become common knowledge.

DEFINITION:  This exception is detected when a type 05 VM
             Account record is detected.
```

```
+-----------------+
| E X C 0 8 0 0 5 |
+-----------------+

FILE:        VMC Exception Activity File (VMCVXC01)

PURPOSE:     Identifies when an attempt is made by one
             user to link to a minidisk owned by another user
             using an invalid userid.
```

```
RATIONALE:   An installation can set a threshold of invalid
             passwords, allowing for some mistyped passwords,
             before an accounting record is written.  This
             exception can identify those users attempting to
             violate security by linking to a protected
             minidisk.

DEFINITION:  This exception is detected when a type 06 VM
             Account record is detected.


+-----------------+
| E X C 0 8 0 0 6 |
+-----------------+

FILE:        VMC Exception Activity File (VMCVXC01)

PURPOSE:     Indicates when a virtual machine has entered
             a disconnected state; that is, when a virtual
             machine begins running without a terminal.

RATIONALE:   An installation can monitor this activity to
             determine the frequency with which users are
             disconnecting rather than logging off the VM
             system.

DEFINITION:  This exception is detected when a type 08 VM
             Account record is detected.


+-----------------+
| E X C 0 8 0 5 0 |
+-----------------+

FILE:        VMC Exception Activity File (VMCVXC01)

PURPOSE:     Identifies the domains for which the VM
             monitoring facility was producing 'event' type
             measurement data.
```

RATIONALE: It is important to know which domains were being monitored by means of event collection (rather than, or in addition to, sampling activity) because this determines which CA MICS files will contain measurement information. If certain CA MICS files do not contain data, this exception provides a useful starting point for determining whether the lack of raw monitor data is responsible for the empty CA MICS files.

DEFINITION: This exception is detected when a domain 1, record 1, VM Monitor record (Event Profile) is read by the CA MICS VM and CMS Analyzer's daily format routine.

```
+-----------------+
| E X C 0 8 0 5 1 |
+-----------------+
```

FILE: VMC Exception Activity File (VMCVXC01)

PURPOSE: Identifies when a Monitor Event Command was issued, altering monitor collection characteristics.

RATIONALE: Altering the types of event data being collected by the VM monitor can have a significant impact on the kinds of information contained in various CA MICS files. This exception can be useful when trying to understand why certain information is, or is not, appearing in CA MICS files.

DEFINITION: This exception is detected when a domain 1, record 2, VM Monitor record (Event Alteration Command) is read by the CA MICS VM and CMS Analyzer's daily format routine.

```
+-----------------+
| E X C 0 8 0 5 2 |
+-----------------+
```

FILE: VMC Exception Activity File (VMCVXC01)

PURPOSE: Identifies those periods when VM monitoring of event activity was suspended, causing a loss of data.

RATIONALE: Monitor suspension records indicate periods when VMA was unable to save event-type monitor data. Monitoring may be suspended because the event portion of the DCSS (Discontiguous Saved Segment) is full, or the IUCV (Inter-User Communication Vehicle) path to the receiving virtual machine has been quiesced.

DEFINITION: This exception is detected when a domain 1, record 3, VM Monitor record (Suspension Record) is read by the CA MICS VM and CMS Analyzer's daily format routine.

```
+-----------------+
| E X C 0 8 0 5 4 |
+-----------------+
```

FILE: VMC Exception Activity File (VMCVXC01)

PURPOSE: Identifies the domains for which the VM Monitoring facility was producing 'sample' type measurement data.

RATIONALE: It is important to know which domains were being monitored by means of sampling (rather than, or in addition to, monitoring event activity) because this determines which CA MICS files will contain measurement information. If certain CA MICS files do not contain data, this exception provides a useful starting point for determining whether the lack of raw monitor data is responsible for the empty CA MICS files.

```
DEFINITION:  This exception is detected when a domain 1,
             record 9, VM Monitor record (Sample Profile) is
             read by the CA MICS VM and CMS Analyzer's daily
             format routine.
```

```
+-----------------+
| E X C 0 8 0 5 5 |
+-----------------+
```

```
FILE:        VMC Exception Activity File (VMCVXC01)

PURPOSE:     Identifies when a Monitor Sample Command was
             issued, altering monitor collection
             characteristics.

RATIONALE:   Altering the types of sampled data being
             collected by the VM monitor can significantly
             impact the information contained in CA MICS
             files.  This exception can be useful when trying
             to understand why certain information is, or is
             not, appearing in CA MICS files.

DEFINITION:  This exception is detected when a domain 1,
             record 10, VM Monitor record (Sample Alteration
             Command) is read by the CA MICS VM and CMS
             Analyzer's daily format routine.
```

```
+-----------------+
| E X C 0 8 0 5 6 |
+-----------------+
```

```
FILE:        VMX System Global Data File (VMXSGP01)

PURPOSE:     Identifies those occurrences when the storage
             set aside for recovery of a preferred virtual
             machine has been completely used, and it was
             necessary to allocate V=R free storage or
             storage for VMDBK control blocks outside this
             area.
```

RATIONALE: Preferred virtual machine recovery is available for systems generated with a V=R area. This recovery attempts to allow a guest in the V=R area to resume execution when VM terminates and subsequently restarts successfully (system bounce) after a system incident. When guest survival storage is exhausted, preferred virtual machine recovery is no longer possible in the event of a system bounce.

DEFINITION: This exception is detected when element SGPGSEXH (Guest Survival Storage Exhausted) has a non-zero value.

```
+----------------+
| E X C 0 8 0 5 7 |
+----------------+
```

FILE: VMX System Global Data File (VMXSGP01)

PURPOSE: Indicates when the list of frames available for CP to use is found empty.

RATIONALE: When CP needs additional storage to continue processing, it takes frames from the CP-maintained list of available frames. If CP cannot find a frame in the available list, it must take frames away from the virtual system to replenish the available list. CP invokes a demand scan function to replenish the available list when it reaches a point called the low threshold. This ordinarily will prevent the list from becoming empty. However, in a severely storage-constrained environment, if the available list becomes totally empty, CP must suspend activities in order to allow the replenishment functions to proceed. This can result in severe performance degradation.

DEFINITION: This exception is detected when a non-zero value is found in data element SGPALMPT.

```
+----------------+
| E X C 0 8 0 5 8 |
+----------------+

FILE:        VMX System Global Data File (VMXSGP01)

PURPOSE:     Identifies the number of times that the CP
             Demand Scan function, used to replenish the
             available list, did not successfully complete
             during the measurement interval.

RATIONALE:   When the list of available frames maintained by
             CP reaches a low point, called the low
             threshold, CP invokes the demand scan function
             to remove selected frames from user-owned frame
             lists and place them on the available list.  The
             demand scan function places frames on the
             available list until the list reaches a value
             called the high threshold.  The demand scan
             function consists of two passes of various
             VMDBKs, and a special emergency pass if the two
             regular passes do not successfully replenish the
             list.  With each succeeding pass, the function
             becomes less selective about which frames to
             take from users.  However, there may be times
             when the demand scan runs to completion, yet was
             unable to get the available list back to the
             high threshold.  In this instance, system
             performance may be severely degraded as overhead
             for maintaining the available list rises.

DEFINITION:  This exception is detected when a non-zero value
             in encountered in data element SGPNOCMP.



+----------------+
| E X C 0 8 0 5 9 |
+----------------+

FILE:        VMX System Global Data File (VMXSGP01)

PURPOSE:     Identifies those times when the number of page
             slots in use was over 90 percent of the total
             page slots available to the system.
```

```
RATIONALE:   Any time the number of page slots in use reaches
             this level, the system is in danger of running
             out of paging space.  The 90 percent threshold
             is set within CP and cannot be altered by the
             user.

DEFINITION:  This exception is detected when a non-zero value
             is encountered in element SGPPGFUL.
```

```
+-----------------+
| E X C 0 8 0 6 0 |
+-----------------+

FILE:        VMX System Global Data File (VMXSGP01)

PURPOSE:     Identifies those times when the number of spool
             slots in use was over 90 percent of the total
             spool slots available to the system.

RATIONALE:   Any time the number of spool slots in use
             reaches this level, the system is in danger of
             running out of spool space.  The 90 percent
             threshold is set within CP and cannot be altered
             by the user.

DEFINITION:  This exception is detected when a non-zero value
             is encountered in element SGPSPFUL.
```

```
+-----------------+
| E X C 0 8 0 6 1 |
+-----------------+

FILE:        VMX System Global Data File

PURPOSE:     Identifies those periods of time when the total
             absolute share of all VMDBKs in the Dispatch
             List exceeds a specified limit.
```

```
RATIONALE:   An installation can set a threshold for absolute
             shares in the dispatch list.  A high total
             absolute share restricts the system's ability to
             equitably distribute resources, as each absolute
             share user is guaranteed a normalized share of
             system resources.

DEFINITION:  This exception is detected when the element
             SGPABSDL exceeds the installation threshold.
```

```
+----------------+
| E X C 0 8 0 6 2 |
+----------------+

FILE:        VMX System Configuration File (VMXSCF01)

PURPOSE:     Indicates that CP terminated abnormally.

RATIONALE:   It is important to identify when CP ends
             abnormally, and what abend code occurred at that
             termination, so that you can maintain system
             availability.

DEFINITION:  This exception is detected when a non-blank
             value is found in element SCFABNCD (ABEND Code
             of Last System Termination).
```

```
+----------------+
| E X C 0 8 0 6 3 |
+----------------+

FILE:        VMX System Configuration File

PURPOSE:     Indicates when the VM system was IPLed.

RATIONALE:   It is important to identify when the system has
             been IPLed because IPLs impact system
             availability.

DEFINITION:  This exception is detected when the date and
             time of the last system IPL was within 24 hours
             of the last VM Monitor initialization.
```

```
+----------------+
| E X C 0 8 0 6 4 |
+----------------+

FILE:        VMC Exception Activity File (VMCVXC01)

PURPOSE:     Identifies those periods when the VM monitoring
             facility only stored part of the monitor records
             due to lack of space in the monitor saved
             segment.

RATIONALE:   Monitor data truncation indicates periods when
             the monitor saved segment lacks space to store
             all of sample or event records for active
             domains.  Note that some counts, such as the
             number of active users (SGPACTIV), may be
             distorted due to the loss of data.

DEFINITION:  This exception is detected when the target
             message type of the IUCV parameter list in the
             monitor control record contains a value of
             'SP' or 'EP', indicating insufficent space in
             the monitor saved segment.


+----------------+
| E X C 0 8 0 6 5 |
+----------------+

FILE:        VMX Logical Partition Activity File (VMXLPR01)

PURPOSE:     Identifies those occurrences when the percentage
             of CPU time spent managing a logical partition
             within a PR/SM complex exceeds the installation
             threshold.

RATIONALE:   It is important to know when PR/SM spent too
             much time managing a logical partition because
             high LPAR overhead may indicate an abnormality
             in the PR/SM environment.  An installation can
             set a threshold for LPRPCMTM (Percent LPAR
             Management Overhead).

DEFINITION:  This exception is detected when the value
             of the variable LPRPCMTM (Percent of LPAR
             Management Overhead) in a logical partition
             exceeds the installation threshold.
```

```
+----------------+
| E X C 0 8 0 6 6 |
+----------------+

FILE:        VMX Logical Partition Activity File (VMXLPR01)

PURPOSE:     Identifies those occurrences when the
             percentage of CPU time spent managing the
             physical configuration within a PR/SM
             environment exceeds the installation threshold.

RATIONALE:   It is important to know when PR/SM spent too
             much time managing the physical configuration
             because high system management overhead may
             indicate an abnormality in the PR/SM
             environment.  An installation can set a
             threshold for LPRPCMTM (Percent LPAR Management
             Overhead).

DEFINITION:  This exception is detected when the value of the
             variable LPRPCMTM (Percent of LPAR Management
             Overhead) in the named partition, PHYSICAL,
             exceeds the installation threshold.

+----------------+
| E X C 0 8 0 6 7 |
+----------------+

FILE:        VMC Exception Activity File (VMCVXC01)

PURPOSE:     Provides notification of a change in CPU
             capability due to a processor being added to the
             configuration and/or a processor speed change.
             Processor speed can change if the CPC uses power
             saving and cycle steering capabilities, such as
             those introduced with the IBM zEnterprise z196.

RATIONALE:   Knowing the CPU capability can provide a
             capacity measurement for billing purposes.

DEFINITION:  This exception is detected when an 0D accounting
             record is read by the daily format routine of
             the CA MICS Analyzer Option for VM/CMS, and New
             CPU Capability (SCFNWCAP) has a non-zero value.
```

```
+----------------+
| E X C 0 8 0 6 8 |
+----------------+

FILE:        VMC Exception Activity File (VMCVXC01)

PURPOSE:     Provides notification of a memory configuration
             change in a partition's storage configuration.

RATIONALE:   Tells you the reserved and standby amounts of
             central storage available to the partition when
             the partition's memory configuration changed.

DEFINITION:  This exception is detected when a Domain 1, Type
             21 Memory Configuration Change monitor record is
             read by the CA MICS Analyzer Option for VM/CMS.
             This record is generated when a configuration
             change event changes a partition's storage
             configuration.
```

Chapter 5: FILES

The CA MICS Analyzer Option for VM/CMS organizes its data logically into the VMC and VMX Information Areas. The VMC Information Area contains account data from all levels of VM systems, while the VMX Information Area contains performance data from VM systems. VM and user resource utilization and activity are quantified. Additional information includes device and channel activity and event-generated data such as device SEEKs.

The table below identifies the individual Data Base files and categorizes them by the Information Area to which they belong. For each file in an information area, the following information is provided:

XDWMYT - This entry defines the Time-Spans in which the file is supported. A nonblank indicates that the file is supported. A '.' indicates that the file is not supported. The Time-Spans indicated by "XDWMYT" are as follows:

- X - DETAIL
- D - DAYS
- W - WEEKS
- M - MONTHS
- Y - YEARS
- T - TABLES AREA

File - The name used to access this file.

File Name - The descriptive label for the file.

NOTE: The Time-Spans 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.

```
+--------------------------------------+------------------+
| Linux                                | Date Generated:  |
| Information Area                     |Tue, May 12, 2009 |
+--------------------------------------+------------------+
|                                                         |
| XDWMYT  File    File Name                               |
|                                                         |
| XD.M..  LNXLXP  LINUX PROCESS ACTIVITY FILE             |
+--------------------------------------+------------------+
| Accounting and Exception             | Date Generated:  |
| Information Area                     |Tue, May 12, 2009 |
+--------------------------------------+------------------+
|                                                         |
| XDWMYT  File    File Name                               |
|                                                         |
| X..MY.  VMCCMU  VM USER RESOURCE ACCOUNTING FILE        |
| X..MY.  VMCDAC  VM DEVICE ACCOUNTING FILE               |
| X..M..  VMCNDT  VM NETWORK DATA TRANSMISSIONS FILE      |
| X..MY.  VMCNTU  VM NETWORK TERMINAL USAGE FILE          |
| X..MY.  VMCVRA  VM RSCS ACCOUNTING FILE                 |
| X.....  VMCVXC  VMC EXCEPTION ACTIVITY FILE             |
+--------------------------------------+------------------+
| VM/XA and VM/ESA System              | Date Generated:  |
| Information Area                     |Tue, May 12, 2009 |
+--------------------------------------+------------------+
|                                                         |
| XDWMYT  File    File Name                               |
|                                                         |
| X.....  VMX_VT  VMX USER TRANSACTION EVENT DATA FILE    |
| X.....  VMX_VU  VMX USER ACTIVITY EVENT DATA FILE       |
| X.....  VMXACF  VMX CPVOL CONFIGURATION DATA FILE       |
| X.....  VMXBPG  VMX BLOCK PAGING DATA FILE              |
| X..M..  VMXCDI  VMX DASD CACHE ACTIVITY FILE            |
| X..M..  VMXCDP  VMX PAGING CACHE ACTIVITY FILE          |
| X.....  VMXCHC  VMX CHANNEL PATH CONTENTION FILE        |
| X.....  VMXCPV  VMX CP OWNED VOLUME DATA FILE           |
| X.....  VMXDCF  VMX DEVICE CONFIGURATION FILE           |
| X..M..  VMXLCP  VMX LOGICAL PROCESSOR ACTIVITY FILE     |
| X..M..  VMXLPR  VMX LOGICAL PARTITION ACTIVITY FILE     |
| X.....  VMXPCF  VMX PROCESSOR CONFIGURATION FILE        |
| X.....  VMXSCF  VMX SYSTEM CONFIGURATION FILE           |
| X.....  VMXSCX  VMX SCHEDULER PARAMETERS FILE           |
| XDWM..  VMXSGP  VMX SYSTEM GLOBAL DATA FILE             |
| X.....  VMXSGS  VMX SYSTEM GLOBAL SUSPEND FILE          |
| X..M..  VMXSPP  VMX SYSTEM PROCESSOR DATA FILE          |
| X.....  VMXSPS  VMX SYSTEM PROCESSOR SUSPEND FILE       |
| X.....  VMXSTP  VMX SYSTEM TIMER POP FILE               |
```

```
| X.....  VMXVSF  VMX SHARED ACTIVITY SUSPEND FILE          |
| X.....  VMXVXF  VMX USER DATA CONFIGURATION FILE          |
| X.....  VMXVXT  VMX USER TRANSACTION SAMPLE DATA FILE     |
| XD.M..  VMXVXU  VMX USER ACTIVITY SAMPLE DATA FILE        |
| X.....  VMXXDS  VMX DEVICE SEEKS DATA FILE                |
| X..M..  VMXXDV  VMX DEVICE ACTIVITY FILE                  |
| X.....  VMXXSF  VMX SHARED STORAGE CONFIGURATION          |
| X.....  VMXXSS  VMX SHARED STORAGE MANAGEMENT FILE        |
|                                                           |
+-----------------------------------------------------------+
```

For a complete understanding of the CA MICS database, see the chapter on Database Structures in the Database Structure and Content Guide. This chapter presents the data files within the Analyzer as they are seen in the online database structure. The files may also be located offline in either audit or history archive files. The PIOM discusses these other structures.

This section contains the following topics:

5.1 VM Data Element Naming Conventions (see page 209)
5.2 VMC Information Area Files (see page 213)
5.3 VMX Information Area Files (see page 248)
5.4 LNX Information Area Files (see page 411)

5.1 VM Data Element Naming Conventions

The naming convention for data elements in the CA MICS database depends on whether they are standard or common data elements.

STANDARD 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 gives the three-character prefix with which the standard data element names begin and the file in which they are contained.

File Name	File	Names Begin With
VM User Resource Accounting	VMCCMU	CMU
VM Device Accounting	VMCDAC	DAC
VM RSCS Accounting	VMCVRA	VRA
VM Network Terminal Usage	VMCNTU	NTU
VM Network Data Transmissions	VMCNDT	NDT

Figure 5-1. VMC Information Area

File Name	File	Names Begin With
VMX System Configuration	VMXSCF	SCF
VMX Processor Configuration	VMXPCF	PCF
VMX System Global Data	VMXSGP	SGP
VMX System Processor Data	VMXSPP	SPP
VMX Device Configuration	VMXDCF	XDV
VMX Device Activity	VMXXDV	XDV
VMX Device SEEKs Data	VMXXDS	XDS
VMX CPVOL Configuration Data	VMXACF	CPV
VMX CP Owned Volume Data	VMXCPV	CPV
VMX User Data Configuration	VMXVXF	VXF
VMX User Activity Sample Data	VMXVXU	VXU
VMX User Activity Event Data	VMX_VU	VXU
VMX User Transaction Sample Data	VMXVXT	VXU
VMX User Transaction Event Data	VMX_VT	VXU
VMX Channel Path Contention	VMXCHC	CHC
VMX Block Paging Data	VMXBPG	BPG
VMX Shared Storage Configuration	VMXXSF	XSS
VMX Shared Storage Management	VMXXSS	XSS
VMX System Timer Pop	VMXSTP	STP
VMX DASD Cache Activity	VMXCDI	CDI
VMX Paging Cache Activity	VMXCDP	CDP
VMX Logical Partition Activity	VMXLPR	LPR
VMX Logical Processor Activity	VMXLCP	LCP
VMX Scheduler Parameters	VMXSCX	SCX

Figure 5-2. VMX Information Area

COMMON DATA ELEMENTS

Common data elements do not use a data element prefix. They have a common definition across database information areas or across files within an information area. Figure 5-3 lists the common data elements that are used by the CA MICS Analyzer Option for VM/CMS.

Variable	Description
CHANNEL	number of a channel attached to the CPU
CPUMODEL	CPU model identification (2084-312, etc.)
CPUSER	CPU/processor serial number
DAY	day of the week (1-7)
DAYNAME	name of the day (SUN, MON, etc.)
DBFILE	information area / file
DEVADDR	address of a device attached to the CPU
DEVCLASS	generic device class (DASD/CKD, TAPE, etc.)
DEVTYPE	specific device type (3380, 3211, etc.)
DURATION	length in seconds of a recording interval
ENDTS	time of day at which an observation ends
HOUR	hour of the day (0-23)
INFOAREA	information area
INTERVLS	number of intervals comprising an observation
MGMTAREA	management area
MICSVER	CA MICS version code
MONTH	month of the year (1-12 or 1-13)
SAMPLES	number of samples
SEVERITY	severity level
STARTTS	time of day at which an observation begins
SYSID	identifier of the system generating data
SYSOUT	sysout class
TERMADDR	CMS terminal address
TIMESPAN	time period
USER	virtual machine to which the record applies
VMLEVEL	VM system release level
VMCACNT	CP account number
VMCSGRPN	VM security group name
VMCINTTS	interval end timestamp
VMCINTVL	interval length
VMCRCODE	account data record type
VMXAGE	observation age
VMXCPUAD	processor address
VMXDVSID	host subchannel ID number
VMXHFSAM	high frequency sample count
VMXMOD	processor model family (2084, 2094, etc.)
VMXMODID	processor model (312, 1C7, etc.)

VMXMSSCH	count of SSCH and RSCH for the device
VMXSTCHG	status change
VOLSER	volume serial number of a tape or disk
WEEK	week of the year(1-52)
YEAR	year of the century
ZONE	time period to which the observation belongs

Figure 5-3. Common Data Elements

5.2 VMC Information Area Files

This section identifies each file in the VMC Information Area and defines its level of summarization and data sequencing as the files appear in the applicable time-span levels. It also lists the data elements contained in each file.

The files maintained in the VMC Information Area, and discussed in the following sections, include:

1 - VM User Resource Accounting File (VMCCMU)
2 - VM Device Accounting File (VMCDAC)
3 - VM RSCS Accounting File (VMCVRA)
4 - VM Network Terminal Use File (VMCNTU)
5 - VM Network Data Transmission File (VMCNDT)
6 - VM Exception Activity File (VMCVXC)

5.2.1 VM User Resource Accounting File (VMCCMU)

The VM User Resource Accounting File identifies the resources consumed by each virtual machine (user). Resources identified include CPU utilization, paging, I/O, and unit record utilization.

This file is created from VM Account data, types 01 and C1 (resource utilization for foreground and background).

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.2.1.1 VMCCMU 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	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	DAY	HOUR	ENDTS
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	ZONE		
YEARS	SYSID	VMCACT1	VMCACT2	VMCACT3	USER
	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 5-4. VMCCMU Sort Sequence and Data Granularity

5.2.1.2 VMCCMU 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	DAY	- Day of Month
X.....E	HOUR	- Hour of Day
X..M..E	MONTH	- Month of Year
X..MY.E	SYSID	- System Identifier
X..MY.E	USER	- User Identification
X..MY.E	VMCACT1	- DIVISION

```
 X..MY.E  VMCACT2  - REGION
 X..MY.E  VMCACT3  - PROJECT
 X.....E  WEEK     - Week of Year
 X..MY.E  YEAR     - Year of Century
 X..MY.E  ZONE     - Time Zone

Common Data Elements

 X..MY.E  CPUMODEL - CPU Model Identification
 X..MY.E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAYNAME  - Name of Day of Week
 X..MY.E  DURATION - Recording Interval Time
 X..MY.E  ENDTS    - End Time Stamp
 X..MY.E  MICSVER  - CA MICS Version Number
 X..MY.E  STARTTS  - Start Time Stamp
 X..MY.E  VMCACNT  - VM Account Number
 X..MY.E  VMCRCODE - VM Account Record Code
 X.....E  VMXCPUAD - Processor Address

Retained Data Elements

 X..MY.E  CMUXMODE - Virtual Machine Execution Mode

Accumulated Data Elements

 X..MY.E  CMUACSES - Number of Accounting Sessions
 X..MY.E  CMUCEOTM - Supervisor Eligible ICF CPU Time
 X..MY.E  CMUCETTM - Total Eligible ICF CPU Time
 X..MY.E  CMUCEVTM - Virtual Eligible ICF CPU Time
 X..MY.E  CMUCFOTM - Supervisor ICF CPU Time
 X..MY.E  CMUCFTTM - Total ICF CPU Time
 X..MY.E  CMUCFVTM - Virtual ICF CPU Time
 X..MY.E  CMUCOST  - Processing Charges
 X..MY.E  CMUCPCT  - Defined CP CPU Count
 X..MY.E  CMUCPOTM - Supervisor CP CPU Time
 X..MY.E  CMUCPTTM - Total CP CPU Time
 X..MY.E  CMUCPUNI - Instructions Executed
 X..MY.E  CMUCPVTM - Virtual CP CPU Time
 X..MY.E  CMUCRD   - Cards Read
 X..MY.E  CMUICFCT - Defined ICF CPU Count
 X..MY.E  CMUIEOTM - Supervisor Eligible   zIIP CPU Time
 X..MY.E  CMUIETTM - Total Eligible zIIP CPU Time
 X..MY.E  CMUIEVTM - Virtual Eligible zIIP CPU Time
 X..MY.E  CMUIFACT - Defined zAAP CPU Count
 X..MY.E  CMUIFLCT - Defined IFL CPU Count
 X..MY.E  CMUIPOTM - Supervisor zIIP CPU Time
 X..MY.E  CMUIPTTM - Total zIIP CPU Time
 X..MY.E  CMUIPVTM - Virtual zIIP CPU Time
 X..MY.E  CMULEOTM - Supervisor Eligible IFL CPU Time
```

```
 X..MY.E  CMULETTM - Total Eligible IFL CPU Time
 X..MY.E  CMULEVTM - Virtual Eligible IFL CPU Time
 X..MY.E  CMULXOTM - Supervisor IFL CPU Time
 X..MY.E  CMULXTTM - Total IFL CPU Time
 X..MY.E  CMULXVTM - Virtual IFL CPU Time
 X..MY.E  CMUNRSES - Number of Sessions
 X..MY.E  CMUOTIME - Supervisor CPU Time
 X..MY.   CMUPAGR  - Page Reads
 X..MY.   CMUPAGW  - Page Writes
 X..MY.E  CMUPRT   - Lines Printed
 X..MY.E  CMUPUN   - Cards Punched
 X..MY.E  CMUSESTM - Session Length
 X..MY.E  CMUSIO   - Non-Spooled SIO Instructions
 X..MY.E  CMUSIOX  - Successful Non-Spooled SIO Instructions
 X..MY.E  CMUTTIME - Total CPU Time
 X..MY.E  CMUVTIME - Virtual CPU Time
 X..MY.E  CMUZEOTM - Supervisor Eligible zAAP CPU Time
 X..MY.E  CMUZETTM - Total Eligible zAAP CPU Time
 X..MY.E  CMUZEVTM - Virtual Eligible zAAP CPU Time
 X..MY.E  CMUZIPCT - Defined zIIP CPU Count
 X..MY.E  CMUZPOTM - Supervisor zAAP CPU Time
 X..MY.E  CMUZPTTM - Total zAAP CPU Time
 X..MY.E  CMUZPVTM - Virtual zAAP CPU Time

Derived Data Elements

 X..MY.E  CMUAVCTM - Virtual Machine Avg Session Length
```

5.2.1.3 VMCCMU Usage Considerations

Special considerations or techniques related to using the VMCCMU File are provided below:

- The information in the VMCCMU File comes from VM Account data. This is event data. Type 01 records are created when a user logs off, when the account number for the virtual machine is changed, and at midnight (if the virtual machine is logged on). Since there may be more than one observation for a user per session in the DETAIL time-span, the element CMUNRSES must be examined to determine whether an observation represents a session start. CMUNRSES will be one for a new session.

o The element CMUSESTM is the amount of time the virtual machine was logged on to VM. This may not be equal to terminal connect time, because it is possible to run a virtual machine disconnected (without a terminal). This is typical of service machines such as RSCS and CMSBATCH.

5.2.1.4 VMCCMU Retrieval Examples

Determine how many sessions user 'JONES' had yesterday:

```
%LET BY = SYSID USER;
PROC SORT DATA=&VMCX..VMCCMU01 OUT=CMU;
  BY &BY; RUN;
DATA _NULL_;
  SET CMU END=LAST;
  BY &BY;
  RETAIN SESSCNT 0;
IF USER = 'JONES' THEN SESSCNT+CMUNRSES;
IF LAST THEN PUT "USER JONES LOGGED ON " SESSCNT " TIMES.";
RUN;
```

5.2.2 VM Device Accounting File (VMCDAC)

The VM Device Accounting File identifies the allocation of dedicated devices and temporary disk virtual machine (user).

This file is created from VM Account data types 02, C2, 03, and C3 records (dedicated device, foreground and CMS Batch, temporary disk, foreground and CMS Batch).

At the DETAIL time-span, this file contains one observation per virtual device allocated by a virtual machine. Each observation contains the variable VMCRCODE, which can be used to determine which of the following Account record types provided the data:

```
Type:  02 - Dedicated Device (Foreground)
       C2 - Dedicated Device (Background)
       03 - Temporary Disk   (Foreground)
       C3 - Temporary Disk   (Background)
```

A dedicated device can be a tape drive, disk drive, terminal, channel-to-channel adapter, or any other device that is allocated to a virtual machine for its exclusive use.

A temporary disk is one that is allocated from the VM T-DISK pool defined to the Control Program (CP) using the CP DEFINE command.

VM supports both count-key-data (CKD) and fixed-block architecture (FBA) DASD.

CKD devices are organized using cylinders, tracks, and records. Data on CKD devices is referenced by cylinder, head, record (CCHHR). The architectural boundary of CKD DASD that is used by VM is the cylinder. Minidisks are provided to virtual machines in cylinder increments.

FBA devices are organized into blocks (currently 512 bytes in length) and the data is referenced by logical block number. Minidisks are provided to virtual machines in block increments.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.2.2.1 VMCDAC 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	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	DAY	HOUR	DEVCLASS
	DEVTYPE	ENDTS			
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	ZONE	DEVCLASS	DEVTYPE
YEARS	SYSID	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	ZONE	DEVCLASS	DEVTYPE	
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. VMCDAC Sort Sequence and Data Granularity

5.2.2.2 VMCDAC 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	DAY	- Day of Month
X..MY.E	DEVCLASS	- Device Class
X..MY.E	DEVTYPE	- Device Type
X.....E	HOUR	- Hour of Day
X..M..E	MONTH	- Month of Year
X..MY.E	SYSID	- System Identifier
X..MY.E	USER	- User Identification

```
 X..MY.E  VMCACT1  - DIVISION
 X..MY.E  VMCACT2  - REGION
 X..MY.E  VMCACT3  - PROJECT
 X.....E  WEEK     - Week of Year
 X..MY.E  YEAR     - Year of Century
 X..MY.E  ZONE     - Time Zone

Common Data Elements

 X..MY.E  CPUMODEL - CPU Model Identification
 X..MY.E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAYNAME  - Name of Day of Week
 X..MY.E  DURATION - Recording Interval Time
 X..MY.E  ENDTS    - End Time Stamp
 X..MY.E  MICSVER  - CA MICS Version Number
 X..MY.E  STARTTS  - Start Time Stamp
 X..MY.E  VMCACNT  - VM Account Number
 X..MY.E  VMCRCODE - VM Account Record Code

Retained Data Elements

 X..MY.E  DACATYPE - Device Allocation Type
 X..MY.E  DACFET   - Device Feature
 X..MY.E  DACMDL   - Device Model

Accumulated Data Elements

 X..MY.E  DACACALC - Number of Accounting Allocations
 X..MY.E  DACCONTM - Device Connect Time
 X..MY.E  DACCOST  - Processing Charges
 X..MY.E  DACMBYTS - DASD Megabytes Occupied
 X..MY.E  DACNRALC - Number of Allocations
 X..MY.E  DACSPACE - DASD Cylinders or Blocks Allocated
```

5.2.2.3 VMCDAC Usage Considerations

Special considerations or techniques related to using the VMCDAC File are provided below:

DATA DUPLICATION CONSIDERATIONS

The VMCDAC File is created from VM Account device records. When a device record is created, there is no identification of the real or virtual device being accounted for. If a virtual machine has two devices that have the same characteristics (e.g., two five-cylinder, 3380 temporary disks) that have been attached to the virtual machine for the same amount of time, the account records produced for these two devices will be exactly the same.

It is possible for data to appear to be duplicated, when in fact it is not. Therefore, no attempt to drop duplicate data in this file is made. This means that, unlike other files, VMCDAC is exposed to duplicate data due to the same data read more than once in the same daily update. If the SORT of the VMCCMU File shows duplicate observations dropped, the VMCDAC File is very likely to contain duplicate data. This will impact accounting if your installation is using the VMCDAC File for cost recovery.

VIRTUAL DEVICE DATA

The VMCDAC File contains one observation per virtual device that is attached, dialed, or dedicated to a virtual machine. If your installation creates account data on an interval basis (with a mechanism that uses the CP ACNT CLOSE ALL command) or has a means of changing the account number for a virtual machine dynamically, the number of observations will increase by the number of times the ACNT command is issued and/or the virtual machine account number is changed while the virtual machine is logged on.

CHARGEBACK CONSIDERATIONS

If your installation wants to use the CA MICS Accounting and Chargeback Product to charge for device usage, certain issues should be considered when using the VMCDAC File.

Some installations may not want to charge for dedicated DASD or may want to apply rates for dedicated DASD that are different from those applied to temporary disk. To determine whether an observation is for dedicated DASD, check the elements DEVCLASS and DACATYPE by using the following type of algorithm:

```
IF DEVCLASS EQ :'DASD' THEN DO;         /* DASD ?      */
  IF DACATYPE EQ :'PERM' THEN DO;       /* DEDICATED ? */
```

Different charges by device type can be made as follows:

```
DTYPE = SUBSTR(DEVTYPE,1,4);
SELECT (DTYPE);
  WHEN ('3380') THEN ...
  WHEN ('3350') THEN ...
  WHEN ('3480') THEN ...
  WHEN ('3420') THEN ...
  OTHERWISE NOP;
END;
```

5.2.2.4 VMCDAC Retrieval Examples

Determine how many tape drives user 'JONES' allocated yesterday:

```
%LET BY = SYSID USER;
PROC SORT DATA=&VMCX..VMCDAC01 OUT=DAC;
  BY &BY; RUN;
DATA _NULL_;
  SET DAC END=LAST;
  BY &BY;
  RETAIN CARTCNT TAPECNT 0;
IF USER = 'JONES' AND DEVCLASS = 'TAPE' THEN DO;
  IF DEVTYPE = '3480' THEN CARTCNT+DACNRALC;
  ELSE TAPECNT+DACNRALC;
END;
IF LAST THEN PUT "USER JONES ALLOCATED " CARTCNT " 3480
   TAPE " "DRIVES AND " TAPECNT " OTHER TAPE DRIVES.";
RUN;
```

5.2.3 VM RSCS Accounting File (VMCVRA)

The VM RSCS Accounting File identifies the amount of data transferred between nodes in a network such as NJE and lines (or cards) read from or written to remote entry stations on behalf of a user.

This file is created from VM Account record, type C0, as created by RSCS.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.2.3.1 VMCVRA 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	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	DAY	HOUR	ENDTS
	DEVCLASS				
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	ZONE	DEVCLASS	
YEARS	SYSID	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	ZONE	DEVCLASS		
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. VMCVRA Sort Sequence and Data Granularity

5.2.3.2 VMCVRA 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	DAY	- Day of Month
X..MY.E	DEVCLASS	- Device Class
X.....E	HOUR	- Hour of Day
X..M..E	MONTH	- Month of Year
X..MY.E	SYSID	- System Identifier
X..MY.E	USER	- User Identification

```
 X..MY.E  VMCACT1  - DIVISION
 X..MY.E  VMCACT2  - REGION
 X..MY.E  VMCACT3  - PROJECT
 X.....E  WEEK     - Week of Year
 X..MY.E  YEAR     - Year of Century
 X..MY.E  ZONE     - Time Zone

Common Data Elements

 X..MY.E  CPUMODEL - CPU Model Identification
 X..MY.E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAYNAME  - Name of Day of Week
 X..MY.E  DEVTYPE  - Device Type
 X..MY.E  ENDTS    - End Time Stamp
 X..MY.E  MICSVER  - CA MICS Version Number
 X..MY.E  SYSOUT   - SYSOUT Class
 X..MY.E  VMCACNT  - VM Account Number
 X..MY.E  VMCRCODE - VM Account Record Code

Retained Data Elements

 X..MY.E  VRALCLNN - Local Node Name
 X..MY.E  VRALSPL  - Original Spool ID
 X..MY.E  VRALSTID - Last (Destination) Userid
 X..MY.E  VRALSTNN - Last (Destination) Node Name
 X..MY.E  VRANUSER - Network Machine Userid
 X..MY.E  VRAORGID - Original Userid
 X..MY.E  VRAORGNN - Original Node Name
 X..MY.E  VRAOSPL  - Original Spool ID
 X..MY.E  VRASBSYS - Accounting Subsysid
 X..MY.E  VRATRCDE - Transmission Code

Accumulated Data Elements

 X..MY.E  VRACOST  - Processing Charges
 X..MY.E  VRANLR   - Number of Records Processed
```

5.2.3.3 VMCVRA Usage Considerations

Special considerations or techniques related to using the VMCVRA File are provided below:

- RSCS does not provide an account number in the account record it creates for data transmission activity, because none is provided by the CP. If your installation uses the VMCVRA File for accounting, the algorithm for building the account structure must rely on USER or some other element.

- If your installation is using SMF type 26 records to charge for VM-originated print on your MVS system, you may be exposed to duplicate charging if both the BATSPL and VMCVRA Files are used for cost recovery. One solution to this problem is to use each file to recover only resources used on the system that generated the account record (C0 VM Account or type 26 SMF). Alternately, either records for data sent to or received from your MVS system can be disregarded when the VMCVRA is processed for accounting.

- The VMCVRA File contains the element VRALCLNN (Local Node Name), which is set in the exit %VMC0EXIT. Together with the elements VRAORGNN (Origin Node Name) and VRALSTNN (Last or Destination Node Name), an analysis of node-to-node spooled data traffic through your VM and CMS Analyzer can be made.

- The VMCVRA File provides information about the use of your RSCS virtual machine(s) and the workload placed on RSCS by users in your spool network. The information available allows you to apportion the workload between users of the network based on print or punch records sent or received, and the origin or destination for the file.

  The elements USER and VMCACNT are set to the value of VRAORGID (originating userid). VMCACNT is defaulted to USER because RSCS does not supply an account number. It may be desirable to modify VMCACNT or USER based on your installation's requirements for charging for RSCS activity in the file exit for the VMCVRA File located in prefix.MICS.USER.SOURCE(#VMCEXIT).

5.2.3.4 VMCVRA Retrieval Examples

Determine how many jobs were sent to MVS (node MVSSYS1) yesterday:

```
DATA _NULL_;
  SET &VMCX..VMCVRA01 END=LAST;
  RETAIN DESTNODE 'MVSSYS1 ' COUNT 0;
IF VRALSTNN = DESTNODE AND VRALSTNN = 'JOB' THEN COUNT+1;
IF LAST THEN PUT ">>> " COUNT " JOBS SENT TO " DESTNODE;
RUN;
```

5.2.4 VM Network Terminal Usage File (VMCNTU)

The VM User Network Terminal Usage File identifies the amount of data passed between the CP and VCNA, and passed between VCNA and VTAM.

This file is created from VM Account record, type 07, VCNA data.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.2.4.1 VMCNTU 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	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	DAY	HOUR	ENDTS
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	ZONE		
YEARS	SYSID	VMCACT1	VMCACT2	VMCACT3	USER
	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. VMCNTU Sort Sequence and Data Granularity

5.2.4.2 VMCNTU 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	DAY	- Day of Month
X.....E	HOUR	- Hour of Day
X..M..E	MONTH	- Month of Year
X..MY.E	SYSID	- System Identifier
X..MY.E	USER	- User Identification
X..MY.E	VMCACT1	- DIVISION

```
 X..MY.E  VMCACT2  - REGION
 X..MY.E  VMCACT3  - PROJECT
 X.....E  WEEK     - Week of Year
 X..MY.E  YEAR     - Year of Century
 X..MY.E  ZONE     - Time Zone

Common Data Elements

 X..MY.E  CPUMODEL - CPU Model Identification
 X..MY.E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAYNAME  - Name of Day of Week
 X..MY.E  DURATION - Recording Interval Time
 X..MY.E  ENDTS    - End Time Stamp
 X..MY.E  MICSVER  - CA MICS Version Number
 X..MY.E  STARTTS  - Start Time Stamp
 X..MY.E  TERMID   - TSO Terminal ID
 X..MY.E  VMCACNT  - VM Account Number
 X..MY.E  VMCRCODE - VM Account Record Code

Accumulated Data Elements

 X..MY.E  NTUCOST  - Processing Charges
 X..MY.E  NTUNRCHR - Number of Characters Received
 X..MY.E  NTUNRCHS - Number of Characters Sent
 X..MY.E  NTUNRCOL - Number of Console Output Lines
 X..MY.E  NTUNRCPY - Number of Copy Requests
 X..MY.E  NTUNRRUR - Number of Request Units Received
 X..MY.E  NTUNRRUS - Number of Request Units Sent
```

5.2.4.3 VMCNTU Usage Considerations

When using the VMCNTU File, note that VCNA Account records are not created in the same way that other VM Account resource records are produced. VCNA records are only produced when a user disconnects or logs off a terminal that is controlled by VTAM and VCNA. If your installation is using a mechanism that creates Account records on a periodic basis (using the CP ACNT CLOSE ALL command), VCNA records will not be among the records generated. Also, if the user has the capability of changing account numbers without logging off, a VCNA Account record will not be created when the account number change occurs.

5.2.4.4 VMCNTU Retrieval Examples

Determine how many request units were processed yesterday:

```
DATA _NULL_;
  SET &VMCX..VMCNTU01 END=LAST;
  RETAIN OCOUNT ICOUNT 0;
OCOUNT+NTUNRRUS;
ICOUNT+NTUNRRUR;
IF LAST THEN PUT "RU'S SENT    :" OCOUNT /
                 "RU'S RECEIVED:" ICOUNT;
RUN;
```

5.2.5 VM Network Data Transmissions File (VMCNDT)

The VM Network Data Transmissions File identifies the amount of data moved over virtual devices such as IUCV, VCTC, and Guest LAN's.

This file is created from VM Account record, type 0C, Network Data Transmissions.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.2.5.1 VMCNDT 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	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	DAY	HOUR	ENDTS
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	VMCACT1	VMCACT2	VMCACT3	USER
	YEAR	MONTH	ZONE		
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 5-8. VMCNDT Sort Sequence and Data Granularity

5.2.5.2 VMCNDT 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	DAY	- Day of Month
X.....E	HOUR	- Hour of Day
X..M..E	MONTH	- Month of Year
X..M..E	SYSID	- System Identifier
X..M..E	USER	- User Identification
X..M..E	VMCACT1	- DIVISION
X..M..E	VMCACT2	- REGION

```
 X..M..E  VMCACT3  - PROJECT
 X.....E  WEEK     - Week of Year
 X..M..E  YEAR     - Year of Century
 X..M..E  ZONE     - Time Zone

Common Data Elements

 X..M..E  CPUMODEL - CPU Model Identification
 X..M..E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAYNAME  - Name of Day of Week
 X..M..E  ENDTS    - End Time Stamp
 X..M..E  MICSVER  - CA MICS Version Number
 X..M..E  TERMID   - TSO Terminal ID
 X..M..E  VMCACNT  - VM Account Number
 X..M..E  VMCRCODE - VM Account Record Code

Retained Data Elements

 X..M..E  NDTCNRID - Connected Remote VM Userid
 X..M..E  NDTIPADD - Default IP Address of Network Adapter
 X..M..E  NDTLANNM - LAN Name
 X..M..E  NDTLANOW - LAN Owner
 X..M..E  NDTLCTCA - VDEV Address of Local CTCA
 X..M..E  NDTNETTP - Network Device Type
 X..M..E  NDTNICTP - Network Data Descriptor Type
 X..M..E  NDTRCTCA - VDEV Address of Remote CTCA
 X..M..E  NDTURCTC - Owner of Remote CTCA Userid
 X..M..E  NDTVNICA - Base VDEV Address of the Virtual NIC

Accumulated Data Elements

 X..M..E  NDTBRCTC - No. of Bytes Received from Remote CTCA
 X..M..E  NDTBRECD - No. of Bytes Received through a NIC
 X..M..E  NDTBRUSR - No. of Bytes Received from Remote User
 X..M..E  NDTBSCTC - Number of Bytes Sent to Remote CTCA
 X..M..E  NDTBSENT - Number of Bytes Sent through a NIC
 X..M..E  NDTBSUSR - Number of Bytes Sent to Remote User
```

5.2.5.3 VMCNDT Usage Considerations

The contents of the VMCNDT depends on the type of Network device being used. Each observation provides information about the amount of data moved over a virtual device, VDEV addresses, and the partner user ID for virtual devices.

The network devices which are accounted for are Virtual Channel to Channel adapters, IUCV connections, and Virtual Network Interface Cards (NIC) sometimes known as Guest LANS.

Element NDTNETTP (Network Device Type) identifies the network device for which the data is being collected.

For NIC, the count of data sent and received is further broken down into four sub-categories defined by data element NDTNICTP (Network NIC Type).

5.2.5.4 VMCNDT Data Retrieval Examples

List of network data transmissions using NIC.

```
DATA NIC;
  SET DATA=&VMCX..VMCNDT01;
  IF NDTNETTP='VIRTUAL NIC';
PROC PRINT ;
  VAR USER STARTTS ENDTS DURATION NDTNETTP NDTNICTP
  NDTIPADD NDTVNICA NDTLANNM NDTLANOW NDTBRECD NDTBSENT;
  TITLE "Network Transmissions using NIC";
RUN;
```

List of network data transmissions using Channel to Channel Adaptors.

```
DATA NIC;
  SET DATA=&VMCX..VMCNDT01;
  IF NDTNETTP='CTCA';
PROC PRINT ;
  VAR USER STARTTS ENDTS DURATION NDTNETTP NDTLCTCA
      NDTRCTCA NDTURCTC NDTBRCTC NDTBSCTC;
  TITLE "Network Transmissions using CTCA";
RUN;
```

```
List of network data transmissions with an IUCV or APPC/VM
Connection.

  DATA NIC;
    SET DATA=&VMCX..VMCNDT01;
    IF NDTNETTP='IUCV OR APPC/VM';
  PROC PRINT ;
    VAR USER STARTTS ENDTS DURATION NDTNETTP NDTCNRID
        NDTBRUSR NDTBSUSR;
    TITLE "Network Transmissions with an IUCV or APPC/VM
           Connection";
  RUN;
```

5.2.6 VM Exception Activity File (VMCVXC)

The VM Exception Activity File quantifies performance and management exceptions that occur in the VM system.

The VM Exception Activity File is created from VM Accounting data, VM Monitor data. Accounting record exceptions are generated from types 04, 05, 06, 08, and 0D VM Accounting records. VM Monitor data exceptions are generated from domain 1, records 2, 3, 10, 11, and 12.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.2.6.1 VMCVXC 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	SEVERITY	INFOAREA	MGMTAREA	VXCCODE
	VXCDESC1	YEAR	MONTH	DAY	HOUR
	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 5-9. VMCVXC Sort Sequence and Data Granularity

5.2.6.2 VMCVXC 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	DAY	- Day of Month
X.....E	HOUR	- Hour of Day
X.....E	INFOAREA	- Information Area
X.....E	MGMTAREA	- Management Area
X.....E	MONTH	- Month of Year
X.....E	SEVERITY	- Severity Level

```
 X.....E  SYSID    - System Identifier
 X.....E  VXCCODE  - Exception Code
 X.....E  VXCDESC1 - Exception Description Text
 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  DBFILE   - Information Area/File
 X.....E  ENDTS    - End Time Stamp
 X.....E  INTERVLS - Number of Recording Intervals
 X.....E  MICSVER  - CA MICS Version Number
 X.....E  TIMESPAN - Time-Period
 X.....E  VMCRCODE - VM Account Record Code

Retained Data Elements

 X.....E  VXCDESC2 - Exception Values Text

Accumulated Data Elements

 X.....   VXCCOUNT - Count of Exception Occurrences
```

5.2.6.3 VMCVXC Usage Considerations

When using the VMCVXC File, note that the VM Journal records which record invalid LINK attempts and invalid LOGON attempts are produced only when the installation-specified limit(s) are reached, so that not all failures will be reported.

5.2.6.4 VMCVXC Retrieval Examples

List all attempted LINKs that failed yesterday:

```
DATA;
 SET &VMCX..VMCVXC01;
 IF VXCCODE = "08005";
PROC PRINT;
 TITLE  REPORT OF FAILED LINK ATTEMPTS;
```

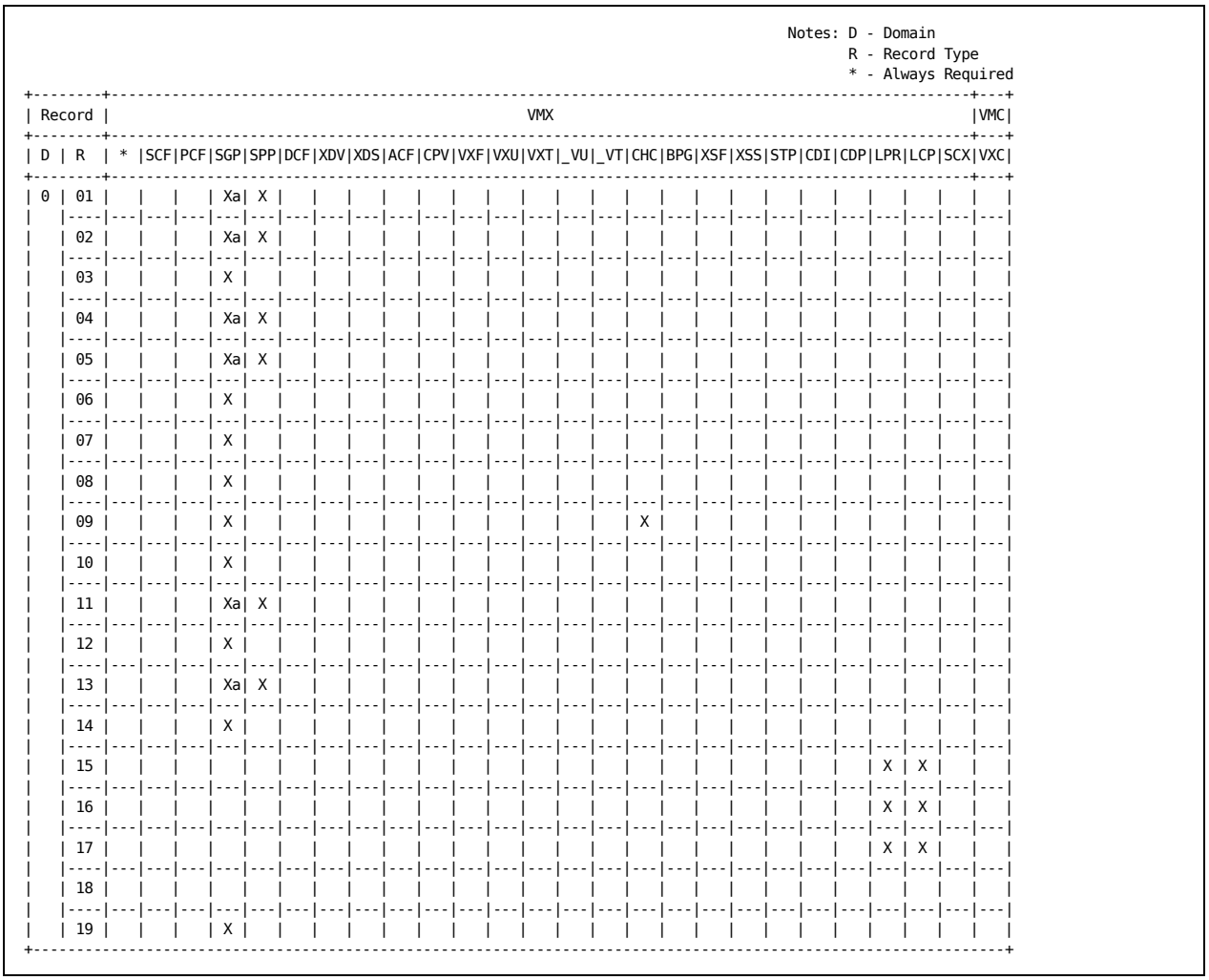
5.3 VMX Information Area Files

This section identifies each file in the VMX 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 maintained in the VMX Information Area include:

1 - VMX System Configuration File (VMXSCF)
2 - VMX Processor Configuration File (VMXPCF)
3 - VMX System Global Data File (VMXSGP)
4 - VMX System Processor Data File (VMXSPP)
5 - VMX Device Configuration File (VMXDCF)
6 - VMX Device Activity File (VMXXDV)
7 - VMX Device SEEKs Data File (VMXXDS)
8 - VMX CPVOL Configuration Data File (VMXACF)
9 - VMX CP Owned Volume Data File (VMXCPV)
10 - VMX User Data Configuration File (VMXVXF)
11 - VMX User Activity Data Files (VMXVXU and VMX_VU)
12 - VMX User Transaction Data Files (VMXVXT and VMX_VT)
13 - VMX Channel Path Contention File (VMXCHC)
14 - VMX Block Paging Data File (VMXBPG)
15 - VMX Shared Storage Configuration File (VMXXSF)
16 - VMX Shared Storage Management File (VMXXSS)
17 - VMX System Timer Pop File (VMXSTP)
18 - VMX DASD Cache Activity File (VMXCDI)
19 - VMX Paging Cache Activity File (VMXCDP)
20 - VMX Logical Partition Activity File (VMXLPR)
21 - VMX Logical Processor Activity File (VMXLCP)
22 - VMX Scheduler Parameters File (VMXSCX)
23 - VMX Channel Path Measurement File (VMXCPM)

The following table is a cross reference list that shows the dependency of each file in the VMX Information Area on VM/ESA and z/VM monitor records.

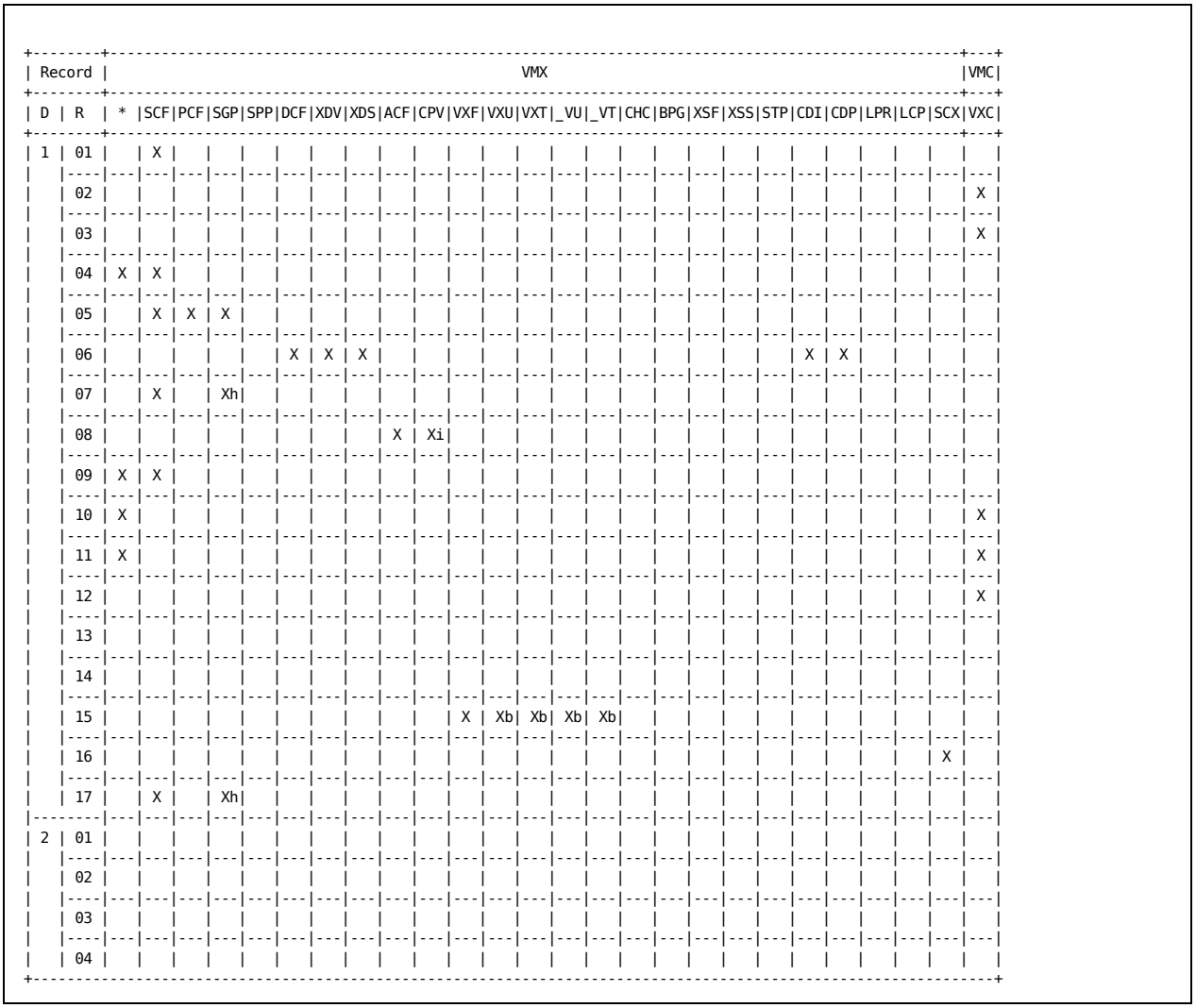


Notes: D - Domain
R - Record Type
* - Always Required

Record			VMX																										VMC
D	R	*	SCF	PCF	SGP	SPP	DCF	XDV	XDS	ACF	CPV	VXF	VXU	VXT	_VU	_VT	CHC	BPG	XSF	XSS	STP	CDI	CDP	LPR	LCP	SCX	VXC
0	01				Xa	X																							
	02				Xa	X																							
	03				X																								
	04				Xa	X																							
	05				Xa	X																							
	06				X																								
	07				X																								
	08				X																								
	09				X												X												
	10				X																								
	11				Xa	X																							
	12				X																								
	13				Xa	X																							
	14				X																								
	15																									X	X		
	16																									X	X		
	17																									X	X		
	18																												
	19				X																								

Xa: propagated from VMXSPP

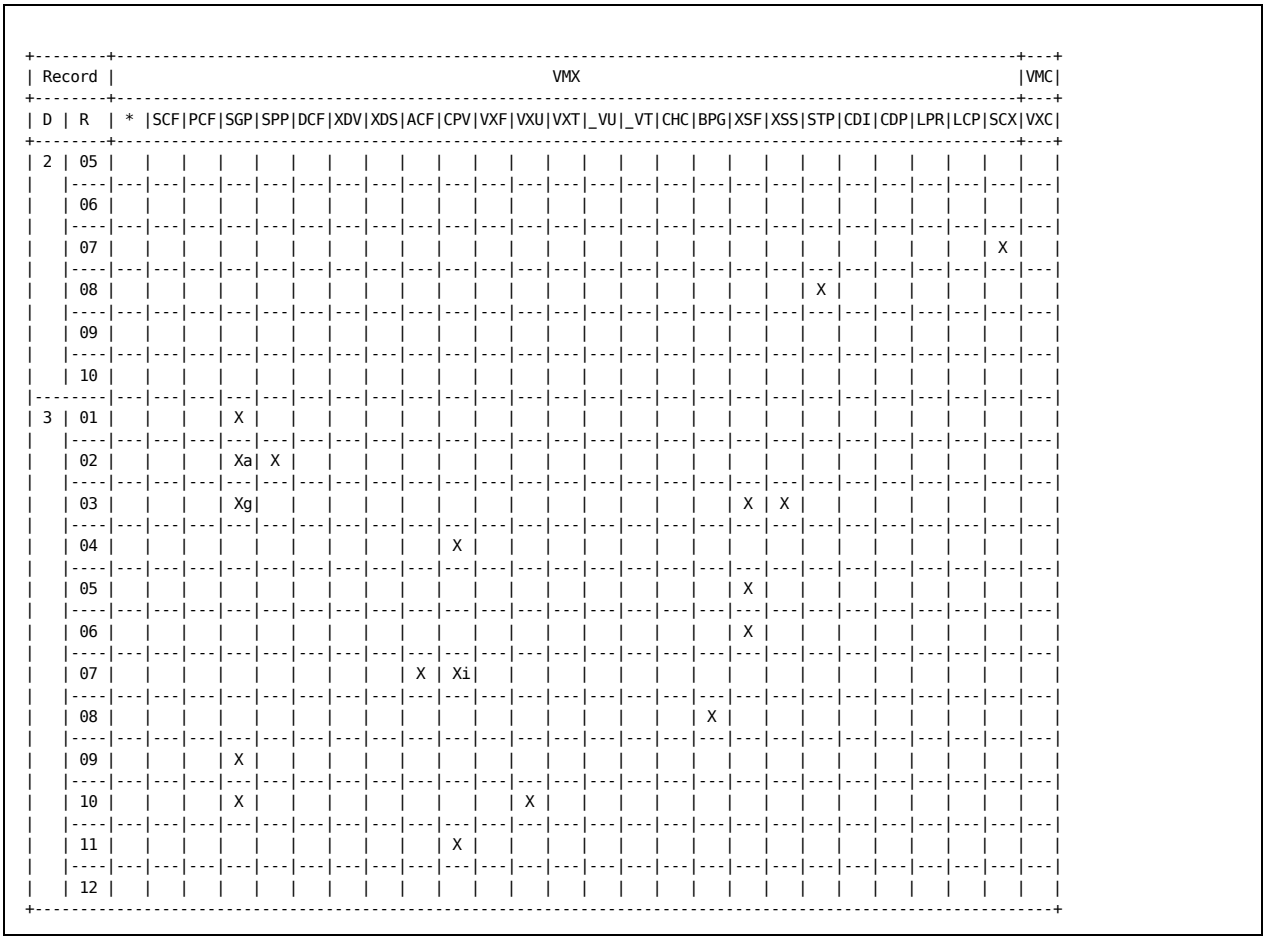
Figure 5-19a. VMX Files and Monitor Records Cross Reference List (Part 1 of 5)



Record	VMX																										VMC
D	R	*	SCF	PCF	SGP	SPP	DCF	XDV	XDS	ACF	CPV	VXF	VXU	VXT	_VU	_VT	CHC	BPG	XSF	XSS	STP	CDI	CDP	LPR	LCP	SCX	VXC
1	01		X																								
	02																										X
	03																										X
	04	X	X																								
	05		X	X	X																						
	06					X	X	X														X	X				
	07		X		Xh																						
	08								X	Xi																	
	09	X	X																								
	10	X																									X
	11	X																									X
	12																										X
	13																										
	14																										
	15											X	Xb	Xb	Xb	Xb											
	16																									X	
	17		X		Xh																						
2	01																										
	02																										
	03																										
	04																										

Xb: contributed to user configuration data for systems prior to VM/ESA Release 2.0
Xh: propagated from VMXSCF
Xi: contributed to paging/spooling configuration data

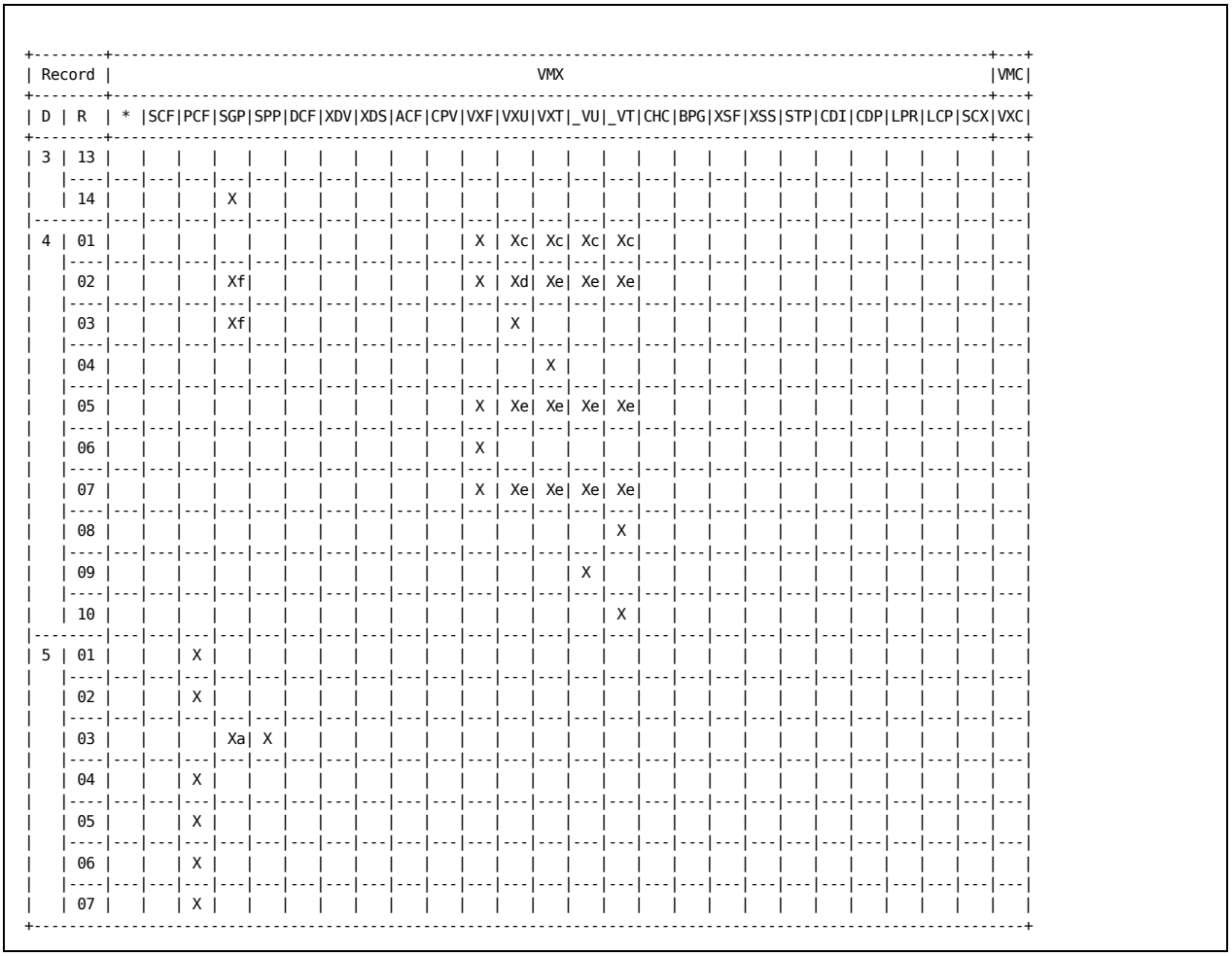
Figure 5-19b. VMX Files and Monitor Records Cross Reference List (Part 2 of 5)



Record			VMX																								VMC
D	R	*	SCF	PCF	SGP	SPP	DCF	XDV	XDS	ACF	CPV	VXF	VXU	VXT	_VU	_VT	CHC	BPG	XSF	XSS	STP	CDI	CDP	LPR	LCP	SCX	VXC
2	05																										
	06																										
	07																									X	
	08																				X						
	09																										
	10																										
3	01			X																							
	02			Xa	X																						
	03			Xg															X	X							
	04										X																
	05																		X								
	06																		X								
	07									X	Xi																
	08																	X									
	09			X																							
	10			X								X															
	11										X																
	12																										

Xa: propagated from VMXSPP
Xg: propagated from VMXXSS
Xi: contributed to paging/spooling configuration data

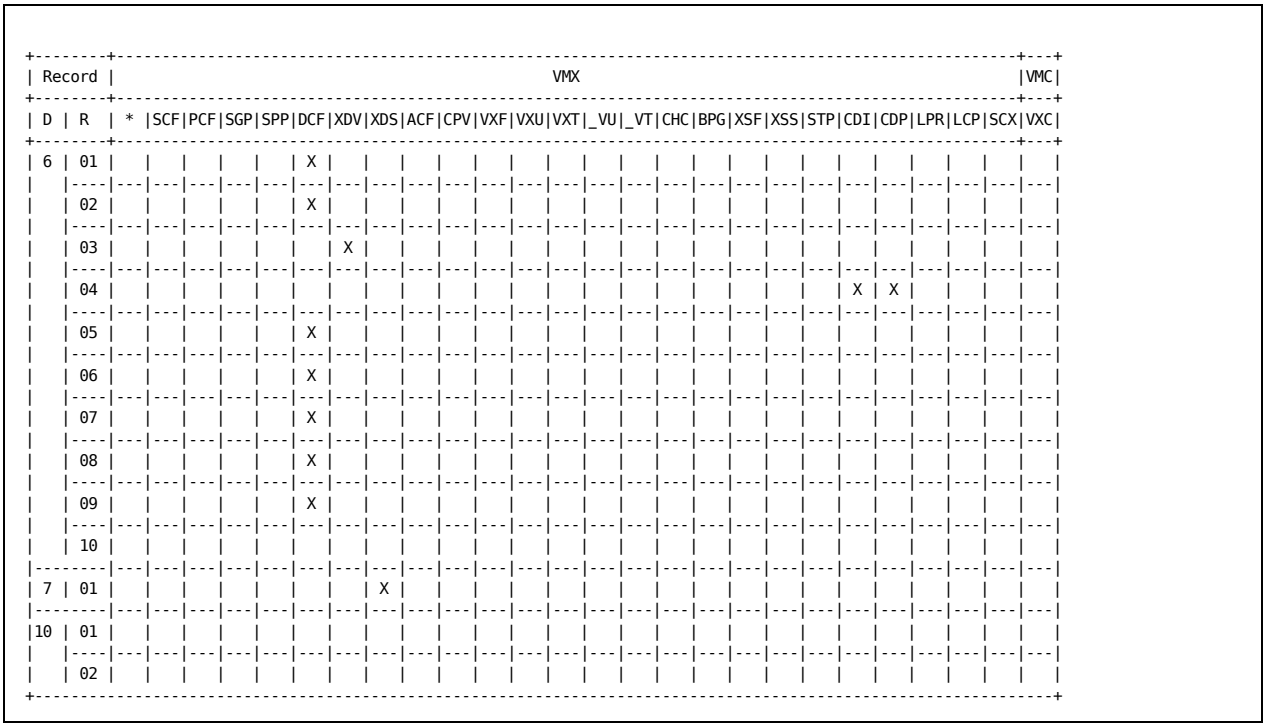
Figure 5-19c. VMX Files and Monitor Records Cross Reference List (Part 3 of 5)



Record	VMX																										VMC
D	R	*	SCF	PCF	SGP	SPP	DCF	XDV	XDS	ACF	CPV	VXF	VXU	VXT	_VU	_VT	CHC	BPG	XSF	XSS	STP	CDI	CDP	LPR	LCP	SCX	VXC
3	13																										
	14				X																						
4	01											X	Xc	Xc	Xc	Xc											
	02				Xf							X	Xd	Xe	Xe	Xe											
	03				Xf								X														
	04													X													
	05											X	Xe	Xe	Xe	Xe											
	06											X															
	07											X	Xe	Xe	Xe	Xe											
	08															X											
	09														X												
	10															X											
5	01		X																								
	02		X																								
	03			Xa	X																						
	04		X																								
	05		X																								
	06		X																								
	07		X																								

Xa: propagated from VMXSPP
Xc: contributed to user configuration data for systems prior to VM/ESA Release 2.0 (optional)
Xd: user logoff data are kept in VMXVXU only for VM/ESA Release 2.0 and later (optional)
Xe: assist in detecting user re-initialization for systems prior to VM/ESA Release 2.0 (optional)
Xf: propagated from VMXVXU

Figure 5-19d. VMX Files and Monitor Records Cross Reference List (Part 4 of 5)



Record	VMX																														VMC
D	R	*	SCF	PCF	SGP	SPP	DCF	XDV	XDS	ACF	CPV	VXF	VXU	VXT	_VU	_VT	CHC	BPG	XSF	XSS	STP	CDI	CDP	LPR	LCP	SCX	VXC				
6	01						X																								
	02							X																								
	03								X																							
	04																						X	X							
	05						X																								
	06						X																								
	07						X																								
	08						X																								
	09						X																								
	10																														
7	01							X																							
10	01																														
	02																														

Figure 5-19e. VMX Files and Monitor Records Cross Reference List (Part 5 of 5)

5.3.1 VMX System Configuration File (VMXSCF)

The VMX System Configuration File contains data that identifies system hardware and software configurations, such as storage, allocation, and vector configuration information.

This file is created from VM/ESA Monitor data, domain 1, records 1, 4, 5, 7, 9, and 17.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.1.1 VMXSCF 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 ENDTS	YEAR	MONTH	DAY	HOUR
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 5-19. VMXSCF Sort Sequence and Data Granularity

5.3.1.2 VMXSCF 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	HOUR	- Hour of Day
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  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  ENDTS    - End Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXAGE   - Observation Age
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier
```

```
Retained Data Elements

 X.....E  SCFABNCD - Abend Code of Last System Termination
 X.....E  SCFADMF  - Asynchronous Data Mover is Installed
 X.....E  SCFCKVOL - Checkpoint Volume Serial Number
 X.....E  SCFCPCAP - Current CPU Capability
 X.....E  SCFCPEID - VM System, Version, Release, Level
 X.....E  SCFCPMF  - Channel Path Measurement Facility Ind
 X.....E  SCFCTCPU - Count of Online CPUs
 X.....E  SCFCURDT - Data at Monitor Initialization
 X.....E  SCFECFTM - Event CONFIG Time Limit
 X.....E  SCFECPMF - Ext Channel Path Measurement Facility In
 X.....E  SCFESA   - Hardware is ESA Capable
 X.....E  SCFFNOTI - Count Uninitialized FRMTEs >= 2G
 X.....E  SCFFTLEN - Length of the Frame Table
 X.....E  SCFGOFFL - Count of Off-line Frames > 2G
 X.....E  SCFGSTOR - System Generation Storage Size
 X.....E  SCFGS2GB - System Generation Storage Size > 2GB
 X.....E  SCFIPLTM - Time of Day at Last System IPL
 X.....E  SCFLGFRM - Count of > 2G Useable Frames
 X.....E  SCFL2ADJ - LPAR Capability Adjustment Factor (L-2)
 X.....E  SCFL2CAP - Level-2 Configuration Capability
 X.....E  SCFL2LNM - Name of the Level-2 Configuration
 X.....E  SCFL2LPR - LPAR Number of L-2 Configuration
 X.....E  SCFMPADD - Address of Resident MP Modules
 X.....E  SCFNPGFR - Count of Non-Pageable Frames
 X.....E  SCFNSCAP - New Capability of Secondary CPU
 X.....E  SCFNUCAD - Address of Resident CP Nucleus
 X.....E  SCFNWCAP - New CPU Capability
 X.....E  SCFOPGFR - Count of Offline Frames
 X.....E  SCFPAGAD - Address of Pageable Nucleus
 X.....E  SCFPCM   - Processor Configuration Mode
 X.....E  SCFPGBFR - Count of Pageable Frames
 X.....E  SCFPGSGR - Preferred Guest Storage Granularity
 X.....E  SCFPPMAP - Program Product Bit Map
 X.....E  SCFRESAD - Address of Resident Executable Nucleus
 X.....E  SCFRSTOR - Real Storage At System Initialization
 X.....E  SCFRS2GB - Real Storage At System Init > 2GB
 X.....E  SCFRVRST - Reserved Real Storage Size
 X.....E  SCFSBRST - Standby Real Storage Size
 X.....E  SCFSCFTM - Sample CONFIG Time Limit
 X.....E  SCFSCMEX - Number of Extra Pages Requested
 X.....E  SCFSCMNO - Total SCMBKs in Contiguous Storage
 X.....E  SCFSDCAP - Current Capability of Secondary CPU
 X.....E  SCFSDVMP - Subchannel Device-Active-Only Indicator
 X.....E  SCFSTIOP - Store I/O Processor Utilization Data Ind
 X.....E  SCFSTLEN - Length of SXS Page Management Table
 X.....E  SCFSXSZE - System Execution Space Size
 X.....E  SCFSYMOD - STSI Machine Model No. of Configuration
```

```
 X.....E  SCFSYPOM - Plant of Manufacture for Configuration
 X.....E  SCFSYSEQ - Sequence Code of Configuration
 X.....E  SCFSYSID - Configuration File System Identifier
 X.....E  SCFSYSTP - Machine Type No. of the Configuration
 X.....E  SCFTRCFR - Trace Table Size for Master CPU
 X.....E  SCFTRCF2 - Trace Table Size per Alternate CPU
 X.....E  SCFTRMTM - Time of Day at Last System Termination
 X.....E  SCFWMVOL - Warmstart Volume Serial Number
 X.....E  SCFXIOMB - Ext'd I/O-Measurement-Block Indicator
 X.....E  SCFXTCCP - Expanded Storage Configured to CP
 X.....E  SCFXTSIZ - Expanded Storage Installed
 X.....E  SCFZONDF - TOD Clock Time Zone Differential
 X.....E  SCFZVM   - Hardware is z/VM Capable
 X.....E  SCF370   - Hardware has Interpretive Exec of S/370

Accumulated Data Elements

 X.....E  SCFL2CFG - No. L-2 Logical CPUs in Configure State
 X.....E  SCFL2CNT - Total Level-2 Logical CPUs Configured
 X.....E  SCFL2DED - No. L-2 Logical CPUs using L-1 Ded'd CPU
 X.....E  SCFL2RVD - No. L-2 Logical CPUs in Reserved State
 X.....E  SCFL2SBY - No. L-2 Logical CPUs in Standby State
 X.....E  SCFL2SHR - No. L-2 Logical CPUs using L-1 Shr'd CPU
```

5.3.1.3 VMXSCF Usage Considerations

The VMXSCF File is a spin file that contains one observation per set of monitor initialization data read (CP MONITOR START command issued). Only one cycle exists in the detail time span. This file contains observations from several runs of the DAILY job. Each cycle is identified by the element VMXAGE. VMXAGE is 1 for the current cycle, 2 for the previous cycle, etc. The VMXAGE element can be used to match a particular VMXSCF observation with a cycle of the VMXSGP file.

5.3.1.4 VMXSCF Data Retrieval Examples

List a history of IPLs for all systems in the unit.

```
%LET BY = SYSID SCFIPLTM;
PROC SORT DATA=&VMXX..VMXSCF01 OUT=IPL;
  BY &BY; RUN;
DATA IPL(KEEP=SYSID SCFIPLTM SCFTRMTM SCFABNCD);
  SET IPL; BY &BY;
  IF FIRST.SCFIPLTM;
  IF SCFABNCD =:' ' THEN SCFABNCD='*none*';
RUN;
PROC PRINT SPLIT= '*';
  ID SYSID;
  LABEL SCFIPLTM='Time of IPL'
        SCFABNCD='Abend Prior*To IPL'
        SCFTRMTM='Time of Abend*Prior to IPL';
  FORMAT SCFIPLTM SCFTRMTM DATETIME16.;
  TITLE "History of IPL Activity";
RUN;
```

5.3.2 VMX Processor Configuration File (VMXPCF)

The VMX Processor Configuration File contains CPU-specific configuration information, such as CPU serial number, model, vector configuration, and cryptographic facility.

This file is created from VM/ESA Monitor data, domain 1, record 5 and domain 5, records 1, 2, 4, 5, 6, and 7.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

```
1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples
```

5.3.2.1 VMXPCF 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 HOUR	VMXCPUAD ENDTS	YEAR	MONTH	DAY
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 5-20. VMXPCF Sort Sequence and Data Granularity

5.3.2.2 VMXPCF 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	HOUR	- Hour of Day
X.....E	MONTH	- Month of Year
X.....E	SYSID	- System Identifier
X.....E	VMXCPUAD	- Processor Address
X.....E	WEEK	- Week of Year
X.....E	YEAR	- Year of Century

```
 X.....E  ZONE     - Time Zone

Common Data Elements

 X.....E  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  ENDTS    - End Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXAGE   - Observation Age
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier
 X.....E  VMXRTYPE - Real CPU Type of Logical Processor
 X.....E  VMXSTCHG - Status Change

Retained Data Elements

 X.....E  PCFCRPID - Crypto Unit ID
 X.....E  PCFCRPTO - Crypto Facility Online Indicator
 X.....E  PCFPROTY - Host CPU Usage Type Identifier
 X.....E  PCFUSDED - Userid of the Dedicated Processor
```

5.3.2.3 VMXPCF Usage Considerations

The VMXPCF File is a spin file that contains one observation per set of monitor initialization data read (CP MONITOR START command issued) and one observation for each time a CPU is brought online or taken offline while the monitor is running. Only one cycle exists in the detail timespan. This file contains observations from several runs of the DAILY job. Each cycle is identified by the element VMXAGE. VMXAGE is 1 for the current cycle, 2 for the previous cycle, etc. The VMXAGE element can be used to match a particular VMXPCF observation with a cycle of the VMXPCF file.

5.3.2.4 VMXPCF Data Retrieval Examples

List occurrences of changes in the CPU configuration that are shown in the VMXPCF file.

```
DATA CPU;
  SET &VMXX..VMXPCF01;
  IF VMXSTCHG NE 'IC'; /* DROP INITIAL CONFIG DATA */
RUN;
PROC PRINT;
  VAR SYSID VMXCPUAD VMXSTCHG ENDTS;
  TITLE "History of CPU Configuration Changes";
RUN;
```

5.3.3 VMX System Global Data File (VMXSGP)

The VMX System Global Data File contains VM/ESA system level data quantifying processor counts and utilization, expanded storage use, real storage use, auxiliary storage use, channel path utilization, scheduler processing, system-wide virtual machine states, address space activity, and communication activity.

This file is created from VM/ESA Monitor Data, domain 0, records 1 through 14; domain 1, records 5, 7, and 17; domain 3, records 1, 2, 3, 9, 10, and 14; domain 4, records 2 and 3; and domain 5, record 3.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

```
1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples
```

5.3.3.1 VMXSGP 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	YEAR	MONTH	DAY	HOUR
	ENDTS				
DAYS	SYSID	YEAR	MONTH	DAY	HOUR
WEEKS	SYSID	YEAR	WEEK	ZONE	HOUR
MONTHS	SYSID	YEAR	MONTH	ZONE	
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 5-21. VMXSGP Sort Sequence and Data Granularity

5.3.3.2 VMXSGP 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: Fri, Jan 24, 2014

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

XDW...E	HOUR	- Hour of Day
XDWM..E	MONTH	- Month of Year
XDWM..E	SYSID	- System Identifier
XDW...E	WEEK	- Week of Year
XDWM..E	YEAR	- Year of Century
XDWM..E	ZONE	- Time Zone

```
Common Data Elements

 XDWM..E  CPUMODEL - CPU Model Identification
 XDWM..E  CPUSER   - CPU/Processor Serial Number
 XDW...E  DAY      - Day of Month
 XDW...E  DAYNAME  - Name of Day of Week
 XDWM..E  DURATION - Recording Interval Time
 XDWM..E  ENDTS    - End Time Stamp
 XDWM..E  INTERVLS - Number of Recording Intervals
 XDWM..E  STARTTS  - Start Time Stamp
 X.....E  VMCGMTTS - GMT TOD Time Stamp
 XDWM..E  VMCINTTS - VM Interval Time Stamp
 XDWM..E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMCZONDF - TOD Clock Time Zone Differential
 XDWM..E  VMLEVEL  - VM System Release Level
 XDWM..E  VMXHFSAM - High Frequency Sample Count
 XDWM..E  VMXMOD   - CPU Processor Model Family
 XDWM..E  VMXMODID - CPU Model Identifier

Retained Data Elements

 XDWM..E  SGPARTOD - Artificial TOD
 XDWM..E  SGPATOD2 - Artificial TOD2
 XDWM..E  SGPAVGAG - Average Age of Paging XSTORE Blocks
 XDWM..E  SGPCPUCA - CPU Capability in Current Configuration
 XDWM..E  SGPE2ETF - Elapsed Timeslice Factor for Q2 Users
 XDWM..E  SGPE3ETF - Elapsed Timeslice Factor for Q3 Users
 XDWM..E  SGPIOPHI - I/O Priority System Maximum
 XDWM..E  SGPIOPLO - I/O Priority System Minimum
 XDWM..E  SGPIOPQ  - I/O Priority Queueing Flag
 XDWM..E  SGPLGFRM - Number Pageable Frames Above 2GB
 XDWM..E  SGPL3ADJ - LPAR Capability Adjustment Factor (L-3)
 XDWM..E  SGPL3CAP - Level-3 Configuration Capability
 XDWM..E  SGPL3CPN - VM CP Providing L-3 Configuration
 XDWM..E  SGPL3NAM - Name of the Level-3 Configuration
 XDWM..E  SGPMDCBS - Minidisk Cache Bias
 XDWM..E  SGPMDCON - Minidisk Cache Flag
 XDWM..E  SGPMEMAD - Additional Central Storage Size
 XDWM..E  SGPNMCAP - Nominal Capability of CPU
 XDWM..E  SGPNOLVL - Number of VM Levels Running
 XDWM..E  SGPNPGFR - Count of Non-Pageable Frames
 XDWM..E  SGPOPGFR - Count of Offline Frames
 XDWM..E  SGPPA2GB - Number Pages Available in Main Above 2GB
 XDWM..E  SGPPB2GB - Number Pages Available in Main Below 2GB
 XDWM..E  SGPPGBFR - Count of Pageable Frames
 XDWM..E  SGPPGMSU - Min Number of Pages Set by User Command
 XDWM..E  SGPPGXSU - Max Number of Pages Set by User Command
 XDWM..E  SGPPW2GB - Deferred Tasks Waiting For Frames > 2GB
```

```
 XDWM..E  SGPSCMEX - Number of Extra Pages Requested
 XDWM..E  SGPSCMNO - Total SCMBKs in Contiguous Storage
 XDWM..E  SGPSDCAP - Current Capability of Secondary CPU
 XDWM..E  SGPSXSAD - Additional SXS Storage
 XDWM..E  SGPSYSLM - System Virtual Disk Allocation Limit
 XDWM..E  SGPUSRLM - User Virtual Disk Allocation Limit
 X.....E  SGPWTCPU - Mask of CPUs in a Wait State
 XDWM..E  SGPXMNSU - Minimum XSTORE Blks Set by User Command
 XDWM..E  SGPXSTBS - Bias for MDC Use of Expanded Storage
 XDWM..E  SGPXSTTC - Maximum XSTORE Blocks in Track Cache
 XDWM..E  SGPXTCCP - Expanded Storage Configured to CP
 XDWM..E  SGPXTSIZ - Expanded Storage Installed

Accumulated Data Elements

 XDWM..E  SGPABNCT - Count of Soft Abends Taken
 XDWM..E  SGPABSDL - Total Absolute CPU Shares of All VMDBKs
 XDWM..E  SGPACTIV - Active Users (Total Time)
 XDWM..E  SGPACTVV - Active Users (Emulation Mode)
 XDWM..E  SGPALMPT - Time Available List Became Empty
 XDWM..E  SGPASCFT - Allocated But Unused Pages for VSC
 XDWM..E  SGPASCRT - Pageable Page Address Space Reorgs
 XDWM..E  SGPASCUT - Storage Frames Used for Paging VSC
 XDWM..E  SGPAVAIL - Number of Frames In Available List
 X.....E  SGPAVDST - Number Active Virtual Disks in Storage
 XDWM..E  SGPAVGM1 - Average Modified Device Load for Paging
 XDWM..E  SGPAVGM2 - Avg Modified Device Load for Spooling
 XDWM..E  SGPAVLHT - Available List High Threshold
 XDWM..E  SGPAVLLT - Available List Low Threshold
 XDWM..E  SGPBPFLT - Page Faults Against Blocked Pages
 XDWM..E  SGPCBZ0  - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ1  - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ10 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ11 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ12 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ13 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ14 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ15 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ16 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ17 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ18 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ19 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ2  - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ20 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ21 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ22 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ23 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ24 - CHPIDs Simultaneously Busy
 XDWM..E  SGPCBZ25 - CHPIDs Simultaneously Busy
```

```
XDWM..E  SGPCBZ26 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ27 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ28 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ29 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ3  - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ30 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ31 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ32 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ33 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ34 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ35 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ36 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ37 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ38 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ39 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ4  - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ40 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ41 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ42 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ43 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ44 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ45 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ46 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ47 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ48 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ49 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ5  - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ50 - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ6  - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ7  - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ8  - CHPIDs Simultaneously Busy
XDWM..E  SGPCBZ9  - CHPIDs Simultaneously Busy
XDWM..E  SGPCCWAB - Times CCW Translation Aborted
XDWM..E  SGPCCWDN - Times CCW Translation Was Successful
XDWM..E  SGPCCWNE - Times CCW Translation Not Eligible
XDWM..E  SGPCCWWR - Times a Write Channel Pgm Was Translated
XDWM..E  SGPCDISP - Count of Users in Dispatch List
XDWM..E  SGPCDLDG - Count of Loading Users in Dispatch List
XDWM..E  SGPCDORM - Number of Users in the Dormant List
XDWM..E  SGPCELIG - Number of VMDBKs in the Eligible List
XDWM..E  SGPCPCF  - VMDBKs in Console Function Waiting for C
XDWM..E  SGPCPCW  - VMDBKs in CPU Wait for CP
XDWM..E  SGPCPLOK - Count of Frames Locked by CP LOCK
XDWM..E  SGPCPPAR - CP Expanded Storage Partition Size
XDWM..E  SGPCPRN  - VMDBKs Running on a Real CP
XDWM..E  SGPCPSW  - VMDBKs in Simulation Wait for CP
XDWM..E  SGPCPUBY - No. CPUs in Standby State
XDWM..E  SGPCPUCF - No. CPUs in Configured State
XDWM..E  SGPCPUCT - Total No. of All CPUs in Configuration
```

```
XDWM..E  SGPCPURG - Cache purges due to SSI Write Link
XDWM..E  SGPCPURN - Number of VMDBKs Running on a Processor
XDWM..E  SGPCPURV - No. CPUs in Reserved State
XDWM..E  SGPCPUTM - Total CPU Time Charged to Users
XDWM..E  SGPCPWT  - VMDBKs in Dispatch List Waiting for CP
XDWM..E  SGPCSCCT - Count of Clear Subchannels for all CPUs
XDWM..E  SGPCTCPU - Count of Online CPUs
XDWM..E  SGPCTXAV - Expanded Storage Available but Not Used
XDWM..E  SGPC1DLD - Count of Loading Users in Q1, Q2, and Q3
XDWM..E  SGPC1DSP - Q1, Q2, and Q3 Users in Dispatch List
XDWM..E  SGPC1ELD - Count of Loading Users in E1, E2, and E3
XDWM..E  SGPC1ELG - E1, E2, and E3 Users in Eligible List
XDWM..E  SGPC2DLD - Count of Loading Users in Q2, and Q3
XDWM..E  SGPC2DSP - Q2 and Q3 Users in Dispatch List
XDWM..E  SGPC2ELD - Count of Loading Users in E2, and E3
XDWM..E  SGPC2ELG - E2, and E3 Users in Eligible List
XDWM..E  SGPC3DLD - Count of Loading Users in Q3
XDWM..E  SGPC3DSP - Q3 Users in Dispatch List
XDWM..E  SGPC3ELD - Count of Loading Users in E3
XDWM..E  SGPC3ELG - E3 Users in Eligible List
XDWM..E  SGPDBKWT - Number of VMDBKs Waiting for CPU
XDWM..E  SGPDGUCT - Count of User Supplied DIAG Instructions
XDWM..E  SGPDIAGT - IBM DIAGNOSE Instructions Executed
XDWM..E  SGPDIAUS - Count of Dialed Users
XDWM..E  SGPDMPAV - Number of Dump Slots Available
XDWM..E  SGPDMPIU - Number of Dump Slots In Use
XDWM..E  SGPDSPLP - Times Dispatcher Long Path Taken
XDWM..E  SGPDSVMT - Number Times a PLDV was Empty
XDWM..E  SGPDSVMW - Dormant List VMDBKs in SVM Wait
XDWM..E  SGPDSVNE - Number Times a PLDV was Not Empty
XDWM..E  SGPDURTM - Total User Duration Time
XDWM..E  SGPEFRC1 - Times User Forced Into Q1
XDWM..E  SGPEFRC2 - Times User Forced Into Q2
XDWM..E  SGPEFRC3 - Times User Forced Into Q3
XDWM..E  SGPEQKAD - Times a QUICKDISP User Added to DLIST
XDWM..E  SGPESVMW - Eligible List VMDBKs in SVM Wait
XDWM..E  SGPE1ETS - Elapsed Timeslice for a Q1 User
XDWM..E  SGPFPICT - Fast Path Instruction Simulations
XDWM..E  SGPFRQWT - Deferred Tasks Waiting for a Frame
XDWM..E  SGPFSHVM - Fair Share Cache Inserters
XDWM..E  SGPFSRCT - Count of Formal Spin Requests
XDWM..E  SGPFSRFR - Free Storage Reserved Frames
XDWM..E  SGPFSRTM - Elapsed Spin Time
XDWM..E  SGPFSXFR - Frames Taken On Free Storage Extend Scan
XDWM..E  SGPFXRCP - No. zHPF DCW Translations Completed
XDWM..E  SGPFXRWT - No. zHPF DCW Translations Presented
XDWM..E  SGPGSEXH - Guest Survival Free Storage Exhausted
XDWM..E  SGPHAFCT - Times VMDBK was Dispatched with Affinity
XDWM..E  SGPHSCCT - Count of Halt Subchannels for all CPUs
```

```
XDWM..E  SGPICFCF - VMDBKs in Console Function Waiting for I
XDWM..E  SGPICFCW - VMDBKs in CPU Wait for ICF
XDWM..E  SGPICFRN - VMDBKs Running on a Real ICF
XDWM..E  SGPICFSW - VMDBKs in Simulation Wait for ICF
XDWM..E  SGPICFWT - VMDBKs in Dispatch List Waiting for ICF
XDWM..E  SGPIFACF - VMDBKs in Console Function Waiting for I
XDWM..E  SGPIFACW - VMDBKs in CPU Wait for IFA
XDWM..E  SGPIFARN - VMDBKs Running on a Real IFA
XDWM..E  SGPIFASW - VMDBKs in Simulation Wait for IFA
XDWM..E  SGPIFAWT - VMDBKs in Dispatch List Waiting for IFA
XDWM..E  SGPIFLCF - VMDBKs in Console Function Waiting for I
XDWM..E  SGPIFLCW - VMDBKs in CPU Wait for IFL
XDWM..E  SGPIFLRN - VMDBKs Running on a Real IFL
XDWM..E  SGPIFLSW - VMDBKs in Simulation Wait for IFL
XDWM..E  SGPIFLWT - VMDBKs in Dispatch List Waiting for IFL
XDWM..E  SGPIIPCF - VMDBKs in Console Function Waiting for I
XDWM..E  SGPIIPCW - VMDBKs in CPU Wait for IIP
XDWM..E  SGPIIPRN - VMDBKs Running on a Real IIP
XDWM..E  SGPIIPSW - VMDBKs in Simulation Wait for IIP
XDWM..E  SGPIIPWT - VMDBKs in Dispatch List Waiting for IIP
XDWM..E  SGPLLCNT - Count of Adds to the Limit List
XDWM..E  SGPLNKCT - Total Virtual Disk in Storage LINKS
XDWM..E  SGPLOKFR - Frames Locked by CP LOCK Command
XDWM..E  SGPL3CFG - No. L-3 Logical CPUs in Configure State
XDWM..E  SGPL3CNT - Total No. Logical CPUs in L-3 Config.
XDWM..E  SGPL3RVD - No. L-3 Logical CPUs in Reserved State
XDWM..E  SGPL3SBY - No. L-3 Logical CPUs in Standby State
XDWM..E  SGPMCHCT - Count of Machine Checks for all CPUs
XDWM..E  SGPMDCAC - MDC Maximum Cache Pages Authorized
XDWM..E  SGPMDCAU - Number of Users Doing Cache Insert
XDWM..E  SGPMDCCR - Pages Used as Track Cache Pages
XDWM..E  SGPMDCEX - Times Cache Insert Fair Share Exceeded
XDWM..E  SGPMDCIA - Read Requests Satisfied From Cache
XDWM..E  SGPMDCIB - Number of Invalidated Cache Blocks
XDWM..E  SGPMDCIS - Number of MDC Hash Table Inserts
XDWM..E  SGPMDCIT - Number of Cache Invalidation Requests
XDWM..E  SGPMDCLI - MDC Fair Share Insert Threshold
XDWM..E  SGPMDCNE - MDC Actual Cache Pages Allocated
XDWM..E  SGPMDCPG - Number of Main Storage Pages in MDC
XDWM..E  SGPMDCPN - Times Data Buffer Was Not Page Aligned
XDWM..E  SGPMDCPR - Number of PAGEINs
XDWM..E  SGPMDCPW - Number of PAGEOUTs
XDWM..E  SGPMDCPY - Times the Data Buffer Was Page Aligned
XDWM..E  SGPMDCQC - Insertions into the MDC Waiting Queue
XDWM..E  SGPMDCTR - Total Cache Read Requests
XDWM..E  SGPMDCXG - Count of XSTOR Blocks Acquired by MDC
XDWM..E  SGPMDCXR - Count of XSTOR Blocks Returned by MDC
XDWM..E  SGPMSTOR - Main Storage Available to VM
XDWM..E  SGPMS2GB - Main Storage Available to VM Above 2GB
```

```
XDWM..E  SGPNASIO - VMDBK in Dispatch List for Asynch I/O
XDWM..E  SGPNOCMP - Demand Scan Not Completed Successfully
XDWM..E  SGPNONPG - Pageable Frames Currently Not Pageable
XDWM..E  SGPNQDSP - Number of Quickdisp Users in DLIST
XDWM..E  SGPNSLOK - Pages Locked by Shared Storage
XDWM..E  SGPNSMIG - Pages Migrated for Shared Storage
XDWM..E  SGPNSNPS - Page Slots Taken by Shared Storage
XDWM..E  SGPNSPGR - Pages Read for Shared Storage
XDWM..E  SGPNSPGW - Pages Written for Shared Storage
XDWM..E  SGPNSPST - Pages Stolen from Shared Storage
XDWM..E  SGPNSXRD - Pages Read from XSTOR for Shared Storage
XDWM..E  SGPNSXWT - Pages Written to XSTOR for Shared Storag
XDWM..E  SGPNTCMP - Non-Trivial Trans. Run in Virtual MP
XDWM..E  SGPNTCUP - Non-Trivial Trans. Run in Virtual UP
XDWM..E  SGPNTRMP - Time for Non-Trivial Trans in Virtual MP
XDWM..E  SGPNTRUP - Time for Non-Trivial Trans in Virtual UP
XDWM..E  SGPNUCSZ - Size of the CP Nucleus
XDWM..E  SGPNUMAS - Number of Addr Spaces Defined for Users
XDWM..E  SGPNUMSA - Number of Shared Address Spaces Defined
XDWM..E  SGPPALOK - Pages Locked in Private Addr Spaces
XDWM..E  SGPPAMIG - Pages Migrated for Private Addr Spaces
XDWM..E  SGPPANPS - Pages Slots Taken by Private Addr Spaces
XDWM..E  SGPPAPGR - Pages Read for Private Addr Spaces
XDWM..E  SGPPAPGW - Pages Written for Private Addr Spaces
XDWM..E  SGPPAPST - Pages Stolen from Private Addr Spaces
XDWM..E  SGPPARSF - Resident Pages for Private Addr Spaces
XDWM..E  SGPPAXRD - Pages Read from XSTOR for Private Addr S
XDWM..E  SGPPAXWT - Pages Written to XSTOR for Private Addr
XDWM..E  SGPPGABL - Number of Pageable Frames in the System
XDWM..E  SGPPGDEL - Number of Main Storage Pages Deleted
XDWM..E  SGPPGDRD - Count of Page Tables Read from DASD
XDWM..E  SGPPGDWT - Count of Page Tables Paged to DASD
XDWM..E  SGPPGFCT - Count of Page Faults
XDWM..E  SGPPGFUL - Number of Times Paging Slots 90% Used
XDWM..E  SGPPGMRD - PGMBK Read from DASD in XSTOR Migration
XDWM..E  SGPPGMRX - PGMBK Read from XSTOR in XSTOR Migration
XDWM..E  SGPPGOUT - Number of PAGOUTs Performed
XDWM..E  SGPPGRIO - Count of Read Operations for Paging
XDWM..E  SGPPGSAV - Number of Page Slots Available
XDWM..E  SGPPGSEC - Page Seconds
XDWM..E  SGPPGSIU - Number of Page Slots In Use
XDWM..E  SGPPGSTL - Count of Times Steal Function Was Used
XDWM..E  SGPPGURM - Pages Unreferenced Prior to Migration
XDWM..E  SGPPGURS - Pages Unreferenced Prior to Steal
XDWM..E  SGPPGWIO - Count of Write Operations for Paging
XDWM..E  SGPPGXRD - Count of Page Tables Read from XSTOR
XDWM..E  SGPPGXWT - Count of Page Tables Paged to XSTOR
XDWM..E  SGPPINVA - Page Faults Resolved w/ PGIN Std Track
XDWM..E  SGPPIN4K - Number of 4K Aligned Pageins
```

```
XDWM..E  SGPPRBTM - Total CPU Time Spent in Emulation Mode
XDWM..E  SGPQDTCT - Transactions Added for Quickdisp VMDBKs
XDWM..E  SGPQDTTM - Time for Quickdisp VMDBKs
XDWM..E  SGPRDCT  - MDC Record Successful Read Accesses
XDWM..E  SGPRELDL - Total Relative CPU Shares of All VMDBKs
XDWM..E  SGPRELOC - Number VDISK Blks Alloc for Relocation
XDWM..E  SGPRGFSB - Regular Free Storage Blocks
XDWM..E  SGPRLOIB - Current Inbound Guest Relocations
XDWM..E  SGPRLOOB - Current Outbound Guest Relocations
XDWM..E  SGPRSCCT - Count of Resume Subchannels for all CPUs
XDWM..E  SGPRSHRF - Resident Shared Page Frames
XDWM..E  SGPRSVPG - Number of Reserved Pages
XDWM..E  SGPSALOK - Pages Locked by Shared Address Spaces
XDWM..E  SGPSAMIG - Pages Migrated for Shared Address Spaces
XDWM..E  SGPSANPS - Page Slots Taken by Shared Address Space
XDWM..E  SGPSAN3E - Number of Shared Address Spaces Active
XDWM..E  SGPSAPGR - Pages Read for Shared Address Spaces
XDWM..E  SGPSAPGW - Pages Written for Shared Address Spaces
XDWM..E  SGPSAPST - Pages Stolen from Shared Address Spaces
XDWM..E  SGPSARSF - Resident Pages for Shared Addr Spaces
XDWM..E  SGPSAVFR - Number of Frames Used for Saveareas
XDWM..E  SGPSAXRD - Total PAGINs for Shared Address Spaces
XDWM..E  SGPSAXWT - Total PAGOUTs for Shared Address Spaces
XDWM..E  SGPSCGAV - Available SCGBKs Allocated
XDWM..E  SGPSCMAV - Available SCMBKs Allocated
XDWM..E  SGPSFACC - Successful Data Transfers By *ACCOUNT
XDWM..E  SGPSFBLK - Successful Data Transfers By *BLOCKIO
XDWM..E  SGPSFCRM - Successful Data Transfers By *CRM
XDWM..E  SGPSFCRT - Number of Spool Files Created
XDWM..E  SGPSFIDN - Successful Data Transfers By *IDENT
XDWM..E  SGPSFLOG - Successful Data Transfers By *LOGREC
XDWM..E  SGPSFMON - Successful Data Transfers By *MONITOR
XDWM..E  SGPSFMSA - Successful Data Transfers By *MSGALL
XDWM..E  SGPSFMSG - Successful Data Transfers By *MSG
XDWM..E  SGPSFPUR - Number of Spool Files Purged
XDWM..E  SGPSFRPI - Successful Data Transfers By *RPI
XDWM..E  SGPSFSIG - Successful Data Transfers By *SIGNAL
XDWM..E  SGPSFSPL - Successful Data Transfers By *SPL
XDWM..E  SGPSFSYM - Successful Data Transfers By *SYMPTOM
XDWM..E  SGPSFUSR - Data Successfully Sent By IUCV From User
XDWM..E  SGPSFVMC - Data Successfully Sent By VMCF From User
XDWM..E  SGPSHRRD - Total PAGINs for Shared Pages
XDWM..E  SGPSIGCT - Count of SIGPs Issued to Another CPU
XDWM..E  SGPSIICT - Count of Instructions from SIE
XDWM..E  SGPSIMCT - Simulated Instructions Executed
XDWM..E  SGPSINCT - Count of Solicited Interrupts
XDWM..E  SGPSITCT - Count of SIE Interceptions
XDWM..E  SGPSNAUS - Count of Users Logged On Thru SNA
XDWM..E  SGPSPFUL - Number of Times Spooling Slots 90% Used
```

```
XDWM..E  SGPSPGIN - Number of Long Path PAGINs Performed
XDWM..E  SGPSPLTM - Elapsed Time in Formal Spin Locks
XDWM..E  SGPSPRCT - Spin Lock Request Count
XDWM..E  SGPSPRDG - Single Page Reads for a Guest
XDWM..E  SGPSPRDS - Single Page Reads for the System
XDWM..E  SGPSPRIO - Count of Read Operations for Spooling
XDWM..E  SGPSPSAV - Number of Spooling Slots Available
XDWM..E  SGPSPSIU - Number of Spooling Slots In Use
XDWM..E  SGPSPWIO - Count of Write Operations For Spooling
XDWM..E  SGPSSCCT - Count of Start Subchannels for all CPUs
XDWM..E  SGPSSMIG - Count of Shared System Migration Visits
XDWM..E  SGPSTACC - Successful Data Transfers to *ACCOUNT
XDWM..E  SGPSTBLK - Successful Data Transfers to *BLOCKIO
XDWM..E  SGPSTCRM - Successful Data Transfers to *CRM
XDWM..E  SGPSTIDN - Successful Data Transfers to *IDENT
XDWM..E  SGPSTKEX - CPEBK Deferrals on a Track (Writes)
XDWM..E  SGPSTKPF - CPEBK Deferrals on a Track (PF-Prty)
XDWM..E  SGPSTLOG - Successful Data Transfers to *LOGREC
XDWM..E  SGPSTMON - Successful Data Transfers to *MONITOR
XDWM..E  SGPSTMSA - Successful Data Transfers to *MSGALL
XDWM..E  SGPSTMSG - Successful Data Transfers to *MSG
XDWM..E  SGPSTOCN - Storage to Consider for Dispatch List
XDWM..E  SGPSTRPI - Successful Data Transfers to *RPI
XDWM..E  SGPSTSIG - Successful Data Transfers to *SIGNAL
XDWM..E  SGPSTSPL - Successful Data Transfers to *SPL
XDWM..E  SGPSTSYM - Successful Data Transfers to *SYMPTOM
XDWM..E  SGPSTUSR - Data Successfully Sent By IUCV To A User
XDWM..E  SGPSTVMC - Data Successfully Sent By VMCF To A User
XDWM..E  SGPSVAFR - Savearea Frames
XDWM..E  SGPSYSTM - Total CPU Time Charged to the System
XDWM..E  SGPTADCS - Discontiguous Shared Segments Active
XDWM..E  SGPTANSS - Number of Named Saved Systems Active
XDWM..E  SGPTCFWT - Times a VMDBK was Found in CF Wait
XDWM..E  SGPTCPWT - Times a VMDBK was Found in CPU Wait
XDWM..E  SGPTCTMP - Trivial Transactions Run in Virtual MP
XDWM..E  SGPTCTUP - Trivial Transactions Run in Virtual UP
XDWM..E  SGPTHRCT - Total of I/O Delays
XDWM..E  SGPTIDL  - Times a VMDBK Found in Test Idle State
XDWM..E  SGPTIOWT - Times a VMDBK was Found in I/O Wait
XDWM..E  SGPTLKCT - Count of Formal Spin Requests for TRQBK
XDWM..E  SGPTLKTM - Elapsed Spin Time for TRQBK
XDWM..E  SGPTOTHR - Times VMDBK was in Undefined State
XDWM..E  SGPTOTM1 - Sum of Modified Device Load for Paging
XDWM..E  SGPTOTM2 - Sum of Modified Device Load for Spooling
XDWM..E  SGPTPGWT - Times a VMDBK was Found in Page Wait
XDWM..E  SGPTRCFR - Trace Table Size for Master CPU
XDWM..E  SGPTRCF2 - Trace Table Size per Alternate CPU
XDWM..E  SGPTRVMP - Time for Trivial Trans. in Virtual MP
XDWM..E  SGPTRVUP - Time for Trivial Trans. in Virtual UP
```

```
XDWM..E  SGPTSMWT - Times a VMDBK was in Simulation Wait
XDWM..E  SGPTSVM  - Times a VMDBK in Test Idle and SVM State
XDWM..E  SGPUONLL - Count of Users on Limit List
XDWM..E  SGPUSICT - Count of Unsolicited Interrupts
XDWM..E  SGPUSRLG - Number of Users Logged On
XDWM..E  SGPUTACC - Data Not Transferred to *ACCOUNT
XDWM..E  SGPUTBLK - Data Not Transferred to *BLOCKIO
XDWM..E  SGPUTCRM - Data Not Transferred to *CRM
XDWM..E  SGPUTIDN - Data Not Transferred to *IDENT
XDWM..E  SGPUTLOG - Data Not Transferred to *LOGREC
XDWM..E  SGPUTMON - Data Not Transferred to *MONITOR
XDWM..E  SGPUTMSA - Data Not Transferred to *MSGALL
XDWM..E  SGPUTMSG - Data Not Transferred to *MSG
XDWM..E  SGPUTRPI - Data Not Transferred to *RPI
XDWM..E  SGPUTSIG - Data Not Transferred to *SIGNAL
XDWM..E  SGPUTSPL - Data Not Transferred to *SPL
XDWM..E  SGPUTSYM - Data Not Transferred to *SYMPTOM
XDWM..E  SGPUTUSR - Data Not Transferred By IUCV to User
XDWM..E  SGPUTVMC - Data Not Transferred By VMCF to User
XDWM..E  SGPUXETM - Elapsed Time When User Exits Were Active
XDWM..E  SGPVIOCT - Count of Virtual I/Os at Logoff
XDWM..E  SGPVMDPM - Number of VMDBKs Passed to Master CPU
XDWM..E  SGPVRFSB - VRFREE Free Storage Blocks
XDWM..E  SGPVRTAC - Total Virtual Disk Blocks Allocated
XDWM..E  SGPWATTM - Total System Wait Time
XDWM..E  SGPWSSDL - WSS Sum for All Users in Dispatch List
XDWM..E  SGPWSSD1 - WSS Sum for Q1-Q3 Users in Dispatch List
XDWM..E  SGPWSSD2 - WSS Sum for Q2 Q3 Users in Dispatch List
XDWM..E  SGPWSSD3 - WSS Sum for Q3 Users in Dispatch List
XDWM..E  SGPXAFR  - Count of Diagnose 98 XA mode Frames
XDWM..E  SGPXBATT - XSTOR Blocks Attached to Users
XDWM..E  SGPXBGET - Count of Expanded Storage Allocations
XDWM..E  SGPXBMAG - Total Age of All XSTOR Blocks Migrated
XDWM..E  SGPXBMAL - XSTOR Blocks Migrated/Allocated to Guest
XDWM..E  SGPXBMDM - XSTOR Blocks Migrated Dormant Users
XDWM..E  SGPXBMLT - Expanded Storage Migration Low Threshold
XDWM..E  SGPXBMND - XSTOR Blocks Migrated Non-Dormant Users
XDWM..E  SGPXBMSS - XSTOR Blocks Migrated From Shared System
XDWM..E  SGPXBMSY - XSTOR Blocks Migrated From SYSTEM
XDWM..E  SGPXBNIO - XSTOR Blocks Released Without I/O
XDWM..E  SGPXBREL - Count of Expanded Storage Releases
XDWM..E  SGPXFREE - Number of Free Storage Extend Frames
XDWM..E  SGPXMCCT - XSTOR Migration Circuits of Cyclic List
XDWM..E  SGPXMICT - Expanded Storage Migration Invocations
XDWM..E  SGPXNTCT - Count of External Interrupts Received
XDWM..E  SGPXPGIN - Number of Fast Path PAGINs Performed
XDWM..E  SGPXSSTL - Count of XSTORE Blocks Stolen
XDWM..E  SGPXTEND - Number of Unsatisfied Extend Requests
XDWM..E  SGPXTUCT - Count of User Exit Calls Made
```

```
 XDWM..E  SGP002TM - Total Elapsed time for All 0/02 Records

Minimum Data Elements

 XDWM..E  SGPMNACT - Minimum Active Users (Total Time)
 XDWM..E  SGPMNACV - Minimum Active Users (Emulation Mode)
 XDWM..E  SGPMNCDL - Minimum Users in Dispatch List
 XDWM..E  SGPMNCEL - Minimum Users in Eligible List
 XDWM..E  SGPMNCLD - Minimum Loading Users
 XDWM..   SGPMNCLI - Minimum MDC Fair Share Insert Threshold
 XDWM..   SGPMNCNE - Minimum MDC Cache Pages Allocated
 XDWM..E  SGPMNCPU - Minimum CPU Time Charged to Users
 XDWM..   SGPMNCXG - Minimum XSTOR Blocks Acquired by MDC
 XDWM..E  SGPMNDLP - Minimum Dispatcher Long Paths Taken
 XDWM..E  SGPMNLOG - Minimum Users Logged On
 XDWM..E  SGPMNLOK - Minimum Locked Frames
 XDWM..E  SGPMNLPP - Minimum Long Path PAGINs Performed
 XDWM..E  SGPMNPG  - Minimum Page Rate
 XDWM..E  SGPMNPGO - Minimum PAGOUTs Performed
 XDWM..E  SGPMNPGR - Minimum Read Operations for Paging
 XDWM..E  SGPMNPGW - Minimum Write Operations for Paging
 XDWM..E  SGPMNPRB - Minimum CPU Time in Emulation Mode
 XDWM..E  SGPMNRGF - Minimum Regular Free Storage Blocks
 XDWM..E  SGPMNSCT - Minimum I/O Rate
 XDWM..E  SGPMNSVA - Minimum Savearea Frames
 XDWM..E  SGPMNSYS - Minimum CPU Time Charged to the System
 XDWM..E  SGPMNTOT - Minimum Pct Total CPU Time
 XDWM..E  SGPMNWAT - Minimum System Wait Time

Maximum Data Elements

 XDWM..E  SGPMXACT - Maximum Active Users (Total Time)
 XDWM..E  SGPMXACV - Maximum Active Users (Emulation Mode)
 XDWM..   SGPMXCAC - Maximum MDC Max Cache Pages Authorized
 XDWM..   SGPMXCAU - Maximum Users Doing Cache Insert
 XDWM..E  SGPMXCDL - Maximum Users in Dispatch List
 XDWM..E  SGPMXCEL - Maximum Users in Eligible List
 XDWM..   SGPMXCIT - Maximum Cache Invalidation Requests
 XDWM..E  SGPMXCLD - Maximum Loading Users
 XDWM..   SGPMXCLI - Maximum MDC Fair Share Insert Threshold
 XDWM..   SGPMXCNE - Maximum MDC Cache Pages Allocated
 XDWM..   SGPMXCPR - Maximum Number of Pageins
 XDWM..E  SGPMXCPU - Maximum CPU Time Charged to Users
 XDWM..   SGPMXCPW - Maximum Number of Pageouts
 XDWM..   SGPMXCQC - Maximum Insertions to MDC Waiting Queue
 XDWM..   SGPMXCXG - Maximum XSTOR Blocks Acquired by MDC
 XDWM..   SGPMXCXR - Maximum XSTOR Blocks Returned by MDC
 XDWM..E  SGPMXDLP - Maximum Dispatcher Long Paths Taken
 XDWM..E  SGPMXLOG - Maximum Users Logged On
```

```
XDWM..E  SGPMXLOK - Maximum Locked Frames
XDWM..E  SGPMXLPP - Maximum Long Path PAGINs Performed
XDWM..E  SGPMXNTR - Max Non-Trivial Trans Response Time
XDWM..E  SGPMXPG  - Maximum Page Rate
XDWM..E  SGPMXPGO - Maximum PAGOUTs Performed
XDWM..E  SGPMXPGR - Maximum Read Operations for Paging
XDWM..E  SGPMXPGW - Maximum Write Operations for Paging
XDWM..E  SGPMXPRB - Maximum CPU Time in Emulation Mode
XDWM..E  SGPMXQDP - Max Quickdisp Trans Response Time
XDWM..E  SGPMXRGF - Maximum Regular Free Storage Blocks
XDWM..E  SGPMXSCT - Maximum I/O Rate
XDWM..E  SGPMXSVA - Maximum Savearea Frames
XDWM..E  SGPMXSYS - Maximum CPU Time Charged to the System
XDWM..E  SGPMXTOT - Maximum Pct Total CPU Time
XDWM..E  SGPMXTRV - Max Trivial Trans Response Time
XDWM..E  SGPMXWAT - Maximum System Wait Time

Derived Data Elements

XDWM..E  SGPAVACT - Average Active Users (Total Time)
XDWM..E  SGPAVACV - Average Active Users (Emulation Mode)
XDWM..E  SGPAVLOG - Average Users Logged On
XDWM..E  SGPAVMAG - Average XSTOR Block Age at Migration
XDWM..E  SGPAVMCT - Average XSTOR Blocks Migrated
XDWM..E  SGPAVSCT - Average I/O Rate
XDWM..E  SGPDPASZ - Dynamic Paging Area Size
XDWM..E  SGPESWSS - WSS Per Second for In Queue Users
XDWM..E  SGPMDCRM - Number of Cache Request Misses
XDWM..   SGPNTMRP - Avg Non-Triv MP Trans Response Time
XDWM..E  SGPNTRRP - Avg Non-Trivial Trans Response Time
XDWM..   SGPNTURP - Avg Non-Triv UP Trans Response Time
XDWM..E  SGPPCAIW - Percent of VMDBKs in Async I/O Wait
XDWM..E  SGPPCCFW - Percent of VMDBKs in CF Wait
XDWM..E  SGPPCCPW - Percent of VMDBKs in CPU Wait
XDWM..E  SGPPCDSW - Percent of Dormant VMDBKs in SVM Wait
XDWM..E  SGPPCESW - Percent of Eligible VMDBKs in SVM Wait
XDWM..E  SGPPCFPP - Percent Fast Path PAGINs
XDWM..E  SGPPCIAC - Percent of Allocated Cache Invalidated
XDWM..E  SGPPCIOW - Percent of VMDBKs in I/O Wait
XDWM..E  SGPPCITC - Percent of Available Cache Invalidated
XDWM..E  SGPPCMDC - Percent Available MDC Allocated
XDWM..E  SGPPCMDX - Percent XTOR Available to MDC
XDWM..E  SGPPCPGW - Percent of VMDBKs in Page Wait
XDWM..E  SGPPCPIX - Percent Page Ins from Expanded Storage
XDWM..E  SGPPCPOX - Percent Page Outs to Expanded Storage
XDWM..E  SGPPCPRB - Percent CPU Virtual Time
XDWM..E  SGPPCPSU - Page Slot Utilization
XDWM..E  SGPPCQSW - Percent of Dispatch VMDBKs in SVM Wait
XDWM..E  SGPPCSMW - Percent of VMDBKs in Simulation Wait
```

```
  XDWM..E  SGPPCSSU - Spool Slot Utilization
  XDWM..E  SGPPCSUP - Percent CPU Supervisor Time
  XDWM..E  SGPPCSVM - Percent of VMDBKs in SVM Wait
  XDWM..E  SGPPCSYS - Percent CPU SYSTEM Time
  XDWM..E  SGPPCTID - Percent of VMDBKs in Test Idle State
  XDWM..E  SGPPCTOT - Percent CPU Total Time
  XDWM..E  SGPPCUND - Percent of VMDBKs in Undefined State
  XDWM..E  SGPPCWAT - Percent CPU Wait Time
  XDWM..E  SGPPCXTU - Percent CP Expanded Storage In Use
  XDWM..E  SGPPGRTA - Page Rate to Auxiliary Storage
  XDWM..E  SGPPGRTE - Page Rate
  XDWM..E  SGPPGRTX - Page Rate to Expanded Storage
  XDWM..   SGPQDPRP - Avg QuickDisp Trans Response Time
  XDWM..E  SGPSTGUT - Storage Utilization
  XDWM..E  SGPSUPTM - Total CP CPU Time Charged to Users
  XDWM..E  SGPTOTTM - Total CPU Time Consumed
  XDWM..   SGPTRMRP - Avg Trivial MP Trans Response Time
  XDWM..   SGPTRURP - Avg Trivial UP Trans Response Time
  XDWM..E  SGPTRVRP - Avg Trivial Trans Response Time
  XDWM..E  SGPTVRAT - Total to Virtual Ratio (System)
  XDWM..E  SGPTVRVM - Total to Virtual Ratio (Users)
  XDWM..E  SGPXBMCT - Expanded Storage Blocks Migrated
_
```

5.3.3.3 VMXSGP Usage Considerations

```
This file contains information from data reported at the
system level, configuration data, and information summarized
from the processor level.  Special considerations or
techniques related to using the VMXSGP file are provided
below:

o  The VMXSGP file is constructed primarily from SYSTEM
   domain and STORAGE domain data, however the elements
   SGPESWSS (Sum of the Estimated Working Set Sizes of In
   Queue Users), SGPACTIV and SGPACTVV (Users Active in Total
   Time and Virtual Time) are derived from the VMXVXU file.
   If all virtual machines are not monitored in the USER
   domain, these elements will not contain accurate data.
   This will affect other elements in the VMXSGP file that
   are calculated using any of the above-named elements.
```

- The VMXSGP file contains data that is summed for all processors in the complex. This information comes from the VMXSPP file. The VMXSGP data element pertaining to CPU use are derived from the VMXSPP file. A separate element named SGP002TM is kept on the file that represents the duration over which these CPU times are accumulated. This is done so that the real sum of elapsed time over all processors is available rather than an approximation based on the DURATION element.

5.3.3.4 VMXSGP Retrieval Examples

Plot the use of CPU by SYSTEM and users.

```
PROC TIMEPLOT DATA=&VMXX..VMXSGP01;
  BY HOUR;
  ID ENDTS;
  PLOT SGPSYSTM='>' SGPCPUTM='*'
  / OVERLAY HILOC POS=60 AXIS=0 TO 60;
  TITLE "CPU Consumption by SYSTEM an Virtual Machines";
RUN;
```

5.3.4 VMX System Processor Data File (VMXSPP)

The VMX System Processor Data File contains VM/ESA data that quantifies processor counts and utilization, expanded storage use, real storage use, auxiliary storage use, channel path utilization, scheduler processing, system-wide virtual machine states, and communication activity for each CPU.

This file is created from VM/ESA Monitor Data, domain 0, records 1, 2, 4, 5, 11, and 13; domain 3, record 2; and domain 5, record 3.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.4.1 VMXSPP 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	VMXCPUAD	VMXRTYPE	YEAR	MONTH
	DAY	HOUR	ENDTS		
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	VMXCPUAD	VMXRTYPE	YEAR	MONTH
	ZONE				
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 5-22. VMXSPP Sort Sequence and Data Granularity

5.3.4.2 VMXSPP 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ---------------------------------------


Sequence/Summary Data Elements

 X.....E  HOUR     - Hour of Day
 X..M..E  MONTH    - Month of Year
 X..M..E  SYSID    - System Identifier
 X..M..E  VMXCPUAD - Processor Address
 X..M..E  VMXRTYPE - Real CPU Type of Logical Processor
 X.....E  WEEK     - Week of Year
 X..M..E  YEAR     - Year of Century
```

```
 X..M..E  ZONE     - Time Zone

Common Data Elements

 X..M..E  CPUMODEL - CPU Model Identification
 X..M..E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X..M..E  DURATION - Recording Interval Time
 X..M..E  ENDTS    - End Time Stamp
 X..M..E  INTERVLS - Number of Recording Intervals
 X..M..E  STARTTS  - Start Time Stamp
 X..M..E  VMCINTTS - VM Interval Time Stamp
 X..M..E  VMCINTVL - VM Monitor Interval Length
 X..M..E  VMLEVEL  - VM System Release Level
 X..M..E  VMXHFSAM - High Frequency Sample Count
 X..M..E  VMXMOD   - CPU Processor Model Family
 X..M..E  VMXMODID - CPU Model Identifier

Retained Data Elements

 X..M..E  SPPDSVMU - Maximum VMDBKs Allowed in this PLDV
 X..M..E  SPPPROTP - Host CPU Usage Type Identifier
 X..M..E  SPPSTATE - Real Processor Operating Status
 X..M..E  SPPUSDED - Userid of the Dedicated Processor
```

```
Accumulated Data Elements

 X..M..E  SPPABNCT - Count of Soft Abends Taken
 X..M..E  SPPALMPT - Times Available List Became Empty
 X..M..E  SPPALOFR - Frames Obtained From the Available List
 X..M..E  SPPCPUTM - Total CPU Time Charged to Users
 X..M..E  SPPCSCCT - Count of Clear Subchannels for this CPU
 X..M..E  SPPDGUCT - Count of User Supplied DIAG Instructions
 X..M..E  SPPDIAGT - IBM DIAGNOSE Instructions Executed
 X..M..E  SPPDLECT - DScan Pass E Complete Dispatch List
 X..M..E  SPPDLEFR - DScan Pass E Frames Taken Dispatch List
 X..M..E  SPPDL1CT - DScan Pass 1 Complete Dispatch List
 X..M..E  SPPDL1FR - DScan Pass 1 Frames Taken Dispatch List
 X..M..E  SPPDL2CT - DScan Pass 2 Complete Dispatch List
 X..M..E  SPPDL2FR - DScan Pass 2 Frames Taken Dispatch List
 X..M..E  SPPDMECT - DScan Pass E Complete Short Term Dormant
 X..M..E  SPPDMEFR - DScan Pass E Frames Taken STDORM List
 X..M..E  SPPDM1CT - DScan Pass 1 Complete Short Term Dormant
 X..M..E  SPPDM1FR - DScan Pass 1 Frames Taken STDORM List
 X..M..E  SPPDM2CT - DScan Pass 2 Complete Short Term Dormant
 X..M..E  SPPDM2FR - DScan Pass 2 Frames Taken STDORM List
 X..M..E  SPPDSPLP - Times Dispatcher Long Path Taken
 X..M..E  SPPDSVMO - Number VMDBKs in Master-Only PLDV
 X..M..E  SPPDSVMT - Number Times this PLDV was Empty
 X..M..E  SPPDSVNE - Number Times this PLDV was Not Empty
 X..M..E  SPPEFRC1 - Times User Forced Into Q1
 X..M..E  SPPEFRC2 - Times User Forced Into Q2
 X..M..E  SPPEFRC3 - Times User Forced Into Q3
 X..M..E  SPPELECT - DScan Pass E Complete Eligible List
 X..M..E  SPPELEFR - DScan Pass E Frames Taken Eligible List
 X..M..E  SPPEL1CT - DScan Pass 1 Complete Eligible List
 X..M..E  SPPEL1FR - DScan Pass 1 Frames Taken Eligible List
 X..M..E  SPPEL2CT - DScan Pass 2 Complete Eligible List
 X..M..E  SPPEL2FR - DScan Pass 2 Frames Taken Eligible List
 X..M..E  SPPEQKAD - Times a QUICKDISP User Added to DLIST
 X..M..E  SPPFPICT - Fast Path Instruction Simulations
 X..M..E  SPPFRERQ - Free Storage Frame Requests
 X..M..E  SPPFRQWT - Deferred Tasks Waiting for A Frame
 X..M..E  SPPFRRET - Free Storage Returns
 X..M..E  SPPFRRRQ - Free Storage Return Requests
 X..M..E  SPPFRTCT - Free Storage Frames Returned
 X..M..E  SPPFRWPN - Frame Replacement Writes Pending
 X..M..E  SPPFSPCT - FSTOR Requests Made from Reg Subpools
 X..M..E  SPPFSXFR - Frames Taken On Free Storage Extend Scan
 X..M..E  SPPHAFCT - Times VMDBK was Dispatched with Affinity
 X..M..E  SPPHSCCT - Count of Halt Subchannels for this CPU
 X..M..E  SPPLD1CT - DScan Pass 1 Complete Long Term Dormant
 X..M..E  SPPLD1FR - DScan Pass 1 Frames Taken LTDORM List
 X..M..E  SPPLD2CT - DScan Pass 2 Complete Long Term Dormant
```

```
X..M..E  SPPLD2FR - DScan Pass 2 Frames Taken LTDORM List
X..M..E  SPPMCHCT - Count of Machine Checks for this CPU
X..M..E  SPPMVB2G - Times Page Moved from Frame >=2GB to <2G
X..M..E  SPPNEWFP - New Fast Path Page Translations
X..M..E  SPPNEWNF - New Non-Fast Path Page Translations
X..M..E  SPPNOCMP - Demand Scan Not Completed Successfully
X..M..E  SPPPGDRD - Count of Page Tables Read from DASD
X..M..E  SPPPGDWT - Count of Page Tables Paged to DASD
X..M..E  SPPPGFCT - Count of Page Faults on this CPU
X..M..E  SPPPGMRD - PGMBK Read from DASD in XSTOR Migration
X..M..E  SPPPGMRX - PGMBK Read from XSTOR in XSTOR Migration
X..M..E  SPPPGOUT - Number of PAGOUTs Performed
X..M..E  SPPPGRIO - Count of Read Operations for Paging
X..M..E  SPPPGWIO - Count of Write Operations for Paging
X..M..E  SPPPGXRD - Count of Page Tables Read from XSTOR
X..M..E  SPPPGXWT - Count of Page Tables Paged to XSTOR
X..M..E  SPPPRBTM - Total CPU Time Spent in Emulation Mode
X..M..E  SPPPTRFP - Fast Path Page Translations
X..M..E  SPPPTRRD - Page Translations Resulting in a Read
X..M..E  SPPRELFR - Frames Returned By Any Release Function
X..M..E  SPPRETFR - Frames Returned to the Available List
X..M..E  SPPRLSCT - Release Functions Performed
X..M..E  SPPRSCCT - Count of Resume Subchannels for this CPU
X..M..E  SPPSFACC - Successful Data Transfers By *ACCOUNT
X..M..E  SPPSFBLK - Successful Data Transfers By *BLOCKIO
X..M..E  SPPSFCRM - Successful Data Transfers By *CRM
X..M..E  SPPSFIDN - Successful Data Transfers By *IDENT
X..M..E  SPPSFLOG - Successful Data Transfers By *LOGREC
X..M..E  SPPSFMON - Successful Data Transfers By *MONITOR
X..M..E  SPPSFMSA - Successful Data Transfers By *MSGALL
X..M..E  SPPSFMSG - Successful Data Transfers By *MSG
X..M..E  SPPSFRPI - Successful Data Transfers By *RPI
X..M..E  SPPSFSIG - Successful Data Transfers By *SIGNAL
X..M..E  SPPSFSPL - Successful Data Transfers By *SPL
X..M..E  SPPSFSYM - Successful Data Transfers By *SYMPTOM
X..M..E  SPPSFUSR - Data Successfully Sent By IUCV From User
X..M..E  SPPSFVMC - Data Successfully Sent By VMCF From User
X..M..E  SPPSHRRD - Total PAGINs for Shared Pages
X..M..E  SPPSIGCT - Count of SIGPs Issued to Another CPU
X..M..E  SPPSIICT - Count of Instructions from SIE
X..M..E  SPPSIMCT - Simulated Instructions Executed
X..M..E  SPPSINCT - Count of Solicited Interrupts
X..M..E  SPPSITCT - Count of SIE Interceptions on this CPU
X..M..E  SPPSPGIN - Number of Long Path PAGINs Performed
X..M..E  SPPSPLTM - Elapsed Time in Formal Spin Locks
X..M..E  SPPSPRCT - Spin Lock Request Count
X..M..E  SPPSPRIO - Count of Read Operations for Spooling
X..M..E  SPPSPWIO - Count of Write Operations For Spooling
X..M..E  SPPSSCCT - Count of Start Subchannels for this CPU
```

```
X..M..E  SPPSSECT - DScan Pass E Complete SFO List
X..M..E  SPPSSEFR - DScan Pass E Frames Taken SFO List
X..M..E  SPPSS1CT - DScan Pass 1 Complete SFO List
X..M..E  SPPSS1FR - DScan Pass 1 Frames Taken SFO List
X..M..E  SPPSS2FR - DScan Pass 2 Frames Taken NSS/DCSS
X..M..E  SPPSTACC - Successful Data Transfers to *ACCOUNT
X..M..E  SPPSTBLK - Successful Data Transfers to *BLOCKIO
X..M..E  SPPSTCRM - Successful Data Transfers to *CRM
X..M..E  SPPSTIDN - Successful Data Transfers to *IDENT
X..M..E  SPPSTLOG - Successful Data Transfers to *LOGREC
X..M..E  SPPSTMON - Successful Data Transfers to *MONITOR
X..M..E  SPPSTMSA - Successful Data Transfers to *MSGALL
X..M..E  SPPSTMSG - Successful Data Transfers to *MSG
X..M..E  SPPSTRPI - Successful Data Transfers to *RPI
X..M..E  SPPSTSIG - Successful Data Transfers to *SIGNAL
X..M..E  SPPSTSPL - Successful Data Transfers to *SPL
X..M..E  SPPSTSYM - Successful Data Transfers to *SYMPTOM
X..M..E  SPPSTUSR - Data Successfully Sent By IUCV To A User
X..M..E  SPPSTVMC - Data Successfully Sent By VMCF To A User
X..M..E  SPPSYSTM - Total CPU Time Charged to the System
X..M..E  SPPUSICT - Count of Unsolicited Interrupts
X..M..E  SPPUTACC - Data Not Transferred to *ACCOUNT
X..M..E  SPPUTBLK - Data Not Transferred to *BLOCKIO
X..M..E  SPPUTCRM - Data Not Transferred to *CRM
X..M..E  SPPUTIDN - Data Not Transferred to *IDENT
X..M..E  SPPUTLOG - Data Not Transferred to *LOGREC
X..M..E  SPPUTMON - Data Not Transferred to *MONITOR
X..M..E  SPPUTMSA - Data Not Transferred to *MSGALL
X..M..E  SPPUTMSG - Data Not Transferred to *MSG
X..M..E  SPPUTRPI - Data Not Transferred to *RPI
X..M..E  SPPUTSIG - Data Not Transferred to *SIGNAL
X..M..E  SPPUTSPL - Data Not Transferred to *SPL
X..M..E  SPPUTSYM - Data Not Transferred to *SYMPTOM
X..M..E  SPPUTUSR - Data Not Transferred By IUCV to User
X..M..E  SPPUTVMC - Data Not Transferred By VMCF to User
X..M..E  SPPVMDPM - Number of VMDBKs Passed to Master CPU
X..M..E  SPPVMS0  - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS1  - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS10 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS11 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS12 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS13 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS14 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS15 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS16 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS17 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS18 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS19 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS2  - Number of VMDBKs Stolen from this CPU
```

```
X..M..E  SPPVMS20 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS21 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS22 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS23 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS24 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS25 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS26 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS27 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS28 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS29 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS3  - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS30 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS31 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS32 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS33 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS34 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS35 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS36 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS37 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS38 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS39 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS4  - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS40 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS41 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS42 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS43 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS44 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS45 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS46 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS47 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS48 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS49 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS5  - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS50 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS51 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS52 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS53 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS54 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS55 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS56 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS57 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS58 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS59 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS6  - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS60 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS61 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS62 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS63 - Number of VMDBKs Stolen from this CPU
X..M..E  SPPVMS7  - Number of VMDBKs Stolen from this CPU
```

```
 X..M..E  SPPVMS8  - Number of VMDBKs Stolen from this CPU
 X..M..E  SPPVMS9  - Number of VMDBKs Stolen from this CPU
 X..M..E  SPPVRFRN - VRFree Req Not Made from Reg Subpools
 X..M..E  SPPVRFRQ - VR Free Storage Requests
 X..M..E  SPPVRFRR - VRFree Req Made from Reg Subpools
 X..M..E  SPPWATTM - Total System Wait Time
 X..M..E  SPPXNTCT - Count of External Interrupts Received
 X..M..E  SPPXPGIN - Number of Fast Path PAGINs Performed
 X..M..E  SPPXTUCT - Count of User Exit Calls Made
 X..M..E  SPP001TM - Duration for 0/01 Record
 X..M..E  SPP002TM - Duration for 0/02 Record
 X..M..E  SPP004TM - Duration for 0/04 Record
 X..M..E  SPP005TM - Duration for 0/05 Record
 X..M..E  SPP011TM - Duration for 0/11 Record
 X..M..E  SPP013TM - Duration for 0/13 Record
 X..M..E  SPP503TM - Duration for 5/03 Record

Derived Data Elements

 X..M..E  SPPPCCPU - Percent CPU Total User Time
 X..M..E  SPPPCPRB - Percent CPU Virtual Time
 X..M..E  SPPPCSUP - Percent CPU Supervisor Time
 X..M..E  SPPPCSYS - Percent CPU SYSTEM Time
 X..M..E  SPPPCTOT - Percent CPU Total Time
 X..M..E  SPPPCWAT - Percent CPU Wait Time
 X..M..E  SPPSUPTM - Total CP CPU Time Charged to Users
 X..M..E  SPPTOTTM - Total CPU Time Used
 X..M..E  SPPTVRAT - Total to Virtual Ratio
 X..M..E  SPPTVRVM - TV Ratio for Virtual Machines
```

5.3.4.3 VMXSPP Usage Considerations

```
The VMXSPP File contains data reported at the processor
level.  Due to the way that the data is created by VM/ESA,
the duration for each record contributing to an observation
is maintained in the file as SPPxxxTM (where xxx represents
the domain/record).  These durations may differ from one
another.  They are used in calculations instead of the
DURATION element to provide greater accuracy.  (The ENDTS
element is the highest timestamp in all the records
contributing to the observation.  DURATION is the difference
between the ENDTS value in two consecutive observations.)

The VMXSPP file contains one observation per CPU in the
processor complex per interval.  To see information for the
entire complex observations must be summed based on VMCINTTS.
```

5.3.4.4 VMXSPP Retrieval Examples

```
Plot the of number of tasks waiting for a page, available
list empty count and the number of times the demand scan did
not complete successfully.

   PROC TIMEPLOT DATA=&VMXX..VMXSPP01;
     BY SYSID VMXCPUAD;
     ID ENDTS;
     PLOT SPPFRQWT='>' SPPNOCMP='*' SPPALMPT='+'
     / OVERLAY HILOC POS=0;
     TITLE "Storage Contention Indices";
   RUN;
```

5.3.5 VMX Device Configuration File (VMXDCF)

```
The VMX Device Configuration File contains I/O device
configuration information, such as device and control unit
identification.

This file is created from VM/ESA Monitor data, domain 1,
record 6 and domain 6, records 1, 2, 5, 6, 7, 8, and 9.

The following sections describe the file's organization, list
the data elements maintained, and provide usage hints.

   1 - File Organization
   2 - Data Elements List
   3 - Usage Considerations
   4 - Retrieval Examples
```

5.3.5.1 VMXDCF 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 HOUR	VMXDVDEV ENDTS	YEAR	MONTH	DAY
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 5-23. VMXDCF Sort Sequence and Data Granularity

5.3.5.2 VMXDCF 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	HOUR	- Hour of Day
X.....E	MONTH	- Month of Year
X.....E	SYSID	- System Identifier
X.....E	VMXDVDEV	- Device Number
X.....E	WEEK	- Week of Year
X.....E	YEAR	- Year of Century
X.....E	ZONE	- Time Zone

```
Common Data Elements

 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  DEVCLASS - Device Class
 X.....E  DEVTYPE  - Device Type
 X.....E  ENDTS    - End Time Stamp
 X.....E  INTERVLS - Number of Recording Intervals
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXAGE   - Observation Age
 X.....E  VMXDVSID - Host Subchannel ID Number
 X.....E  VMXSTCHG - Status Change
 X.....E  VOLSER   - Volume Serial Number

Retained Data Elements

 X.....E  XDVCHPS  - Channel Path Identifiers
 X.....E  XDVCLS   - Device Class Code
 X.....E  XDVDEVID - Device Type Number
 X.....E  XDVDEVNO - Device Number of the Base PAV
 X.....E  XDVDVCID - Control Unit Identifier
 X.....E  XDVDVCMN - Control Unit Model Number
 X.....E  XDVDVFLG - Device Model Source Flag
 X.....E  XDVHSCID - Host Subchannel ID Identifier of PAV
 X.....E  XDVLPMSK - Device Logical Path Mask
 X.....E  XDVMODLN - Device Model Number
 X.....E  XDVOBRCO - OBR Code
 X.....E  XDVPAVIN - Parallel Access Volume Indicator
 X.....E  XDVRCUCO - Real Control Unit Code
 X.....E  XDVSHRD  - CSE Shared Volume Indicator
 X.....E  XDVTHRDY - Count of Times I/O was Delayed
 X.....E  XDVTYP   - Device Type Code

Accumulated Data Elements

 X.....E  XDVIORTE - TOD I/O Rate
```

5.3.5.3 VMXDCF Usage Considerations

Special considerations or techniques related to using the VMXDCF file are provided below:

- The VMXDCF is a spin file that contains one observation per subchannel at monitor initialization, and one observation for each time the status of a device changes while the monitor is running.

- Only one cycle exists in the detail time span. This file contains observations from several runs of the DAILY job. Each cycle is identified by the element VMXAGE. VMXAGE is 1 for the current cycle, 2 for the previous cycle, etc. The VMXAGE element can be used to match a particular VMXDCF observation with a cycle of the VMXXDV or VMXXDS files.

5.3.5.4 VMXDCF Retrieval Examples

List occurrences of changes in the device configuration that are shown in the VMXDCF file.

```
DATA DEVICE;
  SET &VMXX..VMXDCF01;
  IF VMXSTCHG NE 'IC';   /* DROP INITIAL CONFIG DATA */
RUN;
PROC PRINT;
  ID SYSID;
  VAR VMXDVSID VMXDVDEV DEVTYPE VMXSTCHG ENDTS;
  TITLE "I/O Subsystem Configuration Changes";
RUN;
```

5.3.6 VMX Device Activity File (VMXXDV)

The VMX Device Activity File contains VM/ESA data that quantifies I/O activity and utilization by device. Queue lengths and busy conditions are presented for both devices and control units. The file also contains data for devices eligible for Start Interpretive Execution (SIE) assist and how they perform in SIE assist.

This file, which is created from VM/ESA Monitor data (domain 1, type 6 and domain 6, type 3), provides device activity data on a per subchannel basis for any device in the I/O subsystem.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.6.1 VMXXDV 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 HOUR	VMXDVDEV ENDTS	YEAR	MONTH	DAY
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	VMXDVDEV	YEAR	MONTH	ZONE
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 5-24. VMXXDV Sort Sequence and Data Granularity

5.3.6.2 VMXXDV 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: Fri, Jan 24, 2014

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	HOUR	- Hour of Day
X..M..E	MONTH	- Month of Year
X..M..E	SYSID	- System Identifier
X..M..E	VMXDVDEV	- Device Number
X.....E	WEEK	- Week of Year
X..M..E	YEAR	- Year of Century

```
 X..M..E  ZONE     - Time Zone

Common Data Elements

 X..M..E  CPUMODEL - CPU Model Identification
 X..M..E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X..M..E  DEVCLASS - Device Class
 X..M..E  DEVTYPE  - Device Type
 X..M..E  DURATION - Recording Interval Time
 X..M..E  ENDTS    - End Time Stamp
 X..M..E  INTERVLS - Number of Recording Intervals
 X..M..E  STARTTS  - Start Time Stamp
 X.....E  VMCGMTTS - GMT TOD Time Stamp
 X..M..E  VMCINTTS - VM Interval Time Stamp
 X..M..E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMCZONDF - TOD Clock Time Zone Differential
 X..M..E  VMLEVEL  - VM System Release Level
 X..M..E  VMXDVSID - Host Subchannel ID Number
 X..M..E  VMXHFSAM - High Frequency Sample Count
 X..M..E  VMXMOD   - CPU Processor Model Family
 X..M..E  VMXMODID - CPU Model Identifier
 X..M..E  VMXMSAMP - Measurement Facility Samples
 X..M..E  VMXMSSCH - Count of SSCH and RSCH for This Device
 X..M..E  VOLSER   - Volume Serial Number

Retained Data Elements

 X..M..E  XDVAFLG  - Device Allocation Flag
 X..M..E  XDVCHPS  - Channel Path Identifiers
 X..M..E  XDVDEVID - Device Type Number
 X..M..E  XDVDEVNO - Device Number of the Base PAV
 X..M..E  XDVDVCID - Control Unit Identifier
 X..M..E  XDVDVCMN - Control Unit Model Number
 X..M..E  XDVDVFLG - Device Model Source Flag
 X..M..E  XDVFCXP  - zHPF Features Supported on CU
 X..M..E  XDVHSCID - Host Subchannel ID Identifier of PAV
 X..M..E  XDVIORTE - TOD I/O Rate
 X..M..E  XDVLPMSK - Device Logical Path Mask
 X..M..E  XDVMODLN - Device Model Number
 X..M..E  XDVOBRCO - OBR Code
 X..M..E  XDVPAVIN - Parallel Access Volume Indicator
 X..M..E  XDVRCUCO - Real Control Unit Code
 X..M..E  XDVSHRD  - CSE Shared Volume Indicator
 X..M..E  XDVSIEST - SIE ASSIST Status
 X..M..E  XDVSIETS - Last Transition Timestamp of SIE ASSIST
 X..M..E  XDVUSER  - Current Owning Userid
 X..M..E  XDVVDEIO - VDEV State in I/O Assist
```

```
 X..M..E  XDVVFCXM - Mask of CHPIDs Supporting zHPF
 X..M..E  XDVVMAXD - CHSC zHPF Maximum Data Count

Accumulated Data Elements

 X..M..E  XDVADBTM - Accumulated Device-Busy Time
 X..M..E  XDVCUQTM - Control Unit Queuing Time
 X..M..E  XDVDAOTM - Device-Active-Only Time
 X..M..E  XDVDCNTM - Device Connect Time
 X..M..E  XDVDDCTM - Device Disconnect Time
 X..M..E  XDVDRLCT - RDEV Lock Requests Deferred
 X..M..E  XDVFPNTM - Device Function Pending Time
 X..M..E  XDVICRTM - Initial-Command-Response Time
 X..M..E  XDVIRLCT - RDEV Lock Requests Granted Immediately
 X..M..E  XDVMDCIA - Count of SSCHs Avoided by MDC Read Hits
 X..M..E  XDVMDKCT - Count of Minidisks Defined on Volume
 X..M..E  XDVMSICT - Count of Missing Interrupts
 X..M..E  XDVQSSCH - Count of Queued SSCH Requests
 X..M..E  XDVRCCWI - Count of Reserve CCWs Held Immediately
 X..M..E  XDVRCCWP - Count of Reserve CCWs Held Pending
 X..M..E  XDVRCPTM - Reserve CCW Pending Time
 X..M..E  XDVRDCT  - Count of Read Channel Programs
 X..M..E  XDVRSSCH - Count of SSCHs Done For Recovery
 X..M..E  XDVRWXCT - zHPF WRITE Channel Programs
 X..M..E  XDVSIEIN - Times Device Went into SIE ASSIST
 X..M..E  XDVSIEIT - Total Time Spent Inside of SIE ASSIST
 X..M..E  XDVSIELT - Total Time Processing Leaving SIE ASSIST
 X..M..E  XDVSIELV - Times Device Started to Leave SIE ASSIST
 X..M..E  XDVSIEOT - Total Time Spent Outside of SIE ASSIST
 X..M..E  XDVSIEOU - Times Device out of SIE ASSIST
 X..M..E  XDVSINCT - Count of Solicited Interrupts
 X..M..E  XDVSKCT  - Seek I/O Count
 X..M..E  XDVSKSM  - Seek Displacement
 X..M..E  XDVTHRDY - Count of Times I/O was Delayed
 X..M..E  XDVUSICT - Count of Unsolicited Interrupts
 X..M..E  XDVVRXCT - zHPF READ Channel Programs
 X..M..E  XDVWRCT  - Count of Write Channel Programs

Derived Data Elements

 X..M..E  XDVAVDAO - Average Device-Active-Only Time
 X..M..E  XDVAVDCN - Average Device Connect Time
 X..M..E  XDVAVDDC - Average Device Disconnect Time
 X..M..E  XDVAVFPN - Average Device Function Pending Time
 X..M..E  XDVAVMDA - Average MDISKs Allocated
 X..M..E  XDVAVQUE - Average Requests Queued
 X..M..E  XDVAVRCP - Average Reserve CCW Pending Time
 X..M..E  XDVAVRES - Average Device Response Time
 X..M..E  XDVAVSER - Average Device Service Time
```

```
X..M..E  XDVAVSSC - Average SSCH Activity Rate
X..M..E  XDVPCBSY - Percent Device Busy
X..M..E  XDVPCDCN - Percent Device Connected Time
X..M..E  XDVPCRCV - Percent Device Recovery SSCH
```

5.3.6.3 VMXXDV Usage Considerations

Special considerations or techniques related to using the VMXXDV file are provided below:

- The contents of the VMXXDV file may not reflect activity for all devices in the I/O subsystem. This will be true if the monitor has been set to collect data for only specific devices, device types, or device classes.

- The VMXXDV file contains one observation per subchannel to a device. To see all activity for a device, the file must be summarized by VMXDVDEV.

5.3.6.4 VMXXDV Retrieval Examples

Print device activity for all 3380s in the VMXXDV file.

```
%VMCBRK(FILE=VMXXDV01,PERIOD=15);
  IF XDVDEVID = 3380;
RUN;
%LET BY = SYSID VMXDVDEV VMCINTTS;
PROC SORT DATA=VMXXDV;
  BY &BY; RUN;
DATA VMXXDV;
  SET VMXXDV; BY &BY;
  IF NOT FIRST.VMCINTTS THEN DO;
    DURATION = 0;
    VMXHFSAM = 0;
    INTERVLS = 0;
  END;
RUN;
%LET BY = SYSID VMXDVDEV DATE TIME;
%LET BREAK = TIME;
DATA VMXXDV;
  SET VMXXDV;
  %XDVSUM(DERV=YES);
RUN;
PROC PRINT LABEL;
  BY SYSID DATE VMXDVDEV VOLSER;
  ID TIME;
  VAR XDVDCNTM XDVDDCTM XDVFPNTM XDVPCBSY INTERVLS
      DURATION VMXHFSAM;
  TITLE "Device Activity for 3380's";
RUN;
```

5.3.7 VMX Device SEEKS Data File (VMXXDS)

The VMX Device SEEKS Data File contains VM/ESA Monitor data that quantifies DASD arm movement during I/O operations and the frequency of access at the cylinder and track levels.

This file is created from VM/ESA Monitor data, domain 7, record 1.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.7.1 VMXXDS 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 VMXDVDEV XDSTARGT YEAR MONTH DAY HOUR 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 5-25. VMXXDS Sort Sequence and Data Granularity

5.3.7.2 VMXXDS 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  DAY      - Day of Month
 X.....E  HOUR     - Hour of Day
 X.....E  MONTH    - Month of Year
 X.....E  SYSID    - System Identifier
 X.....E  VMXDVDEV - Device Number
 X.....E  WEEK     - Week of Year
 X.....E  XDSTARGT - SEEK Target Cylinder Address
 X.....E  YEAR     - Year of Century
```

```
 X.....E  ZONE     - Time Zone

Common Data Elements

 X.....E  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  DEVTYPE  - Device Type
 X.....E  ENDTS    - End Time Stamp
 X.....E  USER     - User Identification
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXDVSID - Host Subchannel ID Number
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier
 X.....E  VOLSER   - Volume Serial Number

Retained Data Elements

 X.....E  XDSCUU   - Owning Userid Minidisk Address
 X.....E  XDSECYL  - Seek End Cylinder Address
 X.....E  XDSHEAD  - SEEK Target Head Address
 X.....E  XDSOWNER - Minidisk Owner
 X.....E  XDSRWFLG - Read/Write Channel Program Flag
 X.....E  XDSSEEKD - SEEK Distance
 X.....E  XDSSKDIR - SEEK Direction
 X.....E  XDSSOURC - SEEK Starting Cylinder Address

Accumulated Data Elements

 X.....E  XDSNZSEK - Number of Non-Zero Distance SEEKs
 X.....E  XDSRDCT  - Count of Read Channel Programs
 X.....E  XDSSEEKS - Number of Seeks
 X.....E  XDSSKCT  - Seek I/O Count
 X.....E  XDSSKSM  - Seek Displacements
 X.....E  XDSWRCT  - Count of Write Channel Programs
```

5.3.7.3 VMXXDS Usage Considerations

Special considerations or techniques related to using the VMXXDS file are provided below:

- This file is delivered with all timespans disabled. To activate this file, modify VMCGENIN and DBMODEL. Due to the volume of data that can be produced by the SEEKS domain, take care to properly limit the devices for which data is collected or maintained in the database.

- The VMXXDS file contains one observation per seek for each device that is monitored for SEEK activity at the detail level. This file can become large very quickly if many devices are monitored for seek activity or if the devices monitored are particularly active.

- The VMXXDS file is organized by subchannel. If there is more than one path to a device, analysis of all subchannels to the device will be necessary to obtain a complete picture of the seeks activity.

5.3.7.4 VMXXDS Retrieval Examples

List the average seek distance and non-zero seek distance for a subchannel 0374 for an interval from 9:00 am to 10:00 am.

```
%VMCBRK(FILE=VMXXDS01,INTVL='00:01',
        START='09:00'T,END='10:00'T,PERIOD=5);
  IF VMXDVSID = 0374;
RUN;
DATA SEEK;
  SET VMXXDS; BY SYSID DATE TIME;
  IF FIRST.TIME THEN DO;
    ZSEEK  = 0;
    NZSEEK = 0;
    SEEKD  = 0;
  END;
  NZSEEK+ XDSNZSEK;         /* NON-ZERO DISTANCE SEEKS */
  ZSEEK + (NOT XDSNZSEK);   /* ZERO DISTANCE SEEKS     */
  SEEKD + XDSSEEKD;
  IF LAST.TIME THEN DO;
    SEEKS = SUM(NZSEEK,ZSEEK);
    IF NZSEEK THEN AVGNZ = SEEKD/NZSEEK;
    ELSE AVGNZ = 0;
    AVGSD = SEEKD/SEEKS;
    OUTPUT;
  END;
RUN;
PROC TIMEPLOT DATA=SEEK;
  BY SYSID DATE;
  ID TIME ;
  PLOT AVGNZ='>' AVGSD='+'
   / OVERLAY POS=0;
   LABEL AVGNZ='Average Non-Zero Seek Distance'
         AVGSD='Average Seek Distance';
RUN;
```

5.3.8 VMX CPVOL Configuration Data File (VMXACF)

The VMX CPVOL Configuration File (VMXACF) identifies system hardware and software configurations for VM/ESA systems.

It is derived from data in VM/ESA Monitor domain 3, record 7 and domain 1, record 8.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.8.1 VMXACF 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	VOLSER HOUR	CPVSTRAD ENDTS	YEAR	MONTH
DAYS	N/A				
WEEKS	N/A				
MONTHS	N/A				
YEARS	N/A				
TABLES	N/A				

Generation Date: Tue, May 12, yyyy

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 5-26. VMXACF Sort Sequence and Data Granularity

5.3.8.2 VMXACF 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.

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	CPVSTRAD	- Area Start Cylinder/Page
X.....E	HOUR	- Hour of Day
X.....E	MONTH	- Month of Year
X.....E	SYSID	- System Identifier
X.....E	VOLSER	- Volume Serial Number
X.....E	WEEK	- Week of Year
X.....E	YEAR	- Year of Century
X.....E	ZONE	- Time Zone

```
Common Data Elements

 X.....E  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  ENDTS    - End Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXAGE   - Observation Age
 X.....E  VMXDVSID - Host Subchannel ID Number
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier
 X.....E  VMXSTCHG - Status Change

Retained Data Elements

 X.....E  CPVATYPE - Allocation Type
 X.....E  CPVCYLAR - Cylinders/Pages In Area
 X.....E  CPVPGCYL - Pages Per Cylinder
```

5.3.8.3 VMXACF Usage Considerations

Special considerations or techniques related to using the VMXACF file are provided below:

- The VMXACF is a spin file that contains one observation per area on CP owned volumes at monitor initialization data, and one observation for each time a new area is brought online while the monitor is running.

- Only one cycle exists in the detail timespan. This file contains observations from several runs of the DAILY job. Each cycle is identified by the element VMXAGE. VMXAGE is 1 for the current cycle, 2 for the previous cycle, etc. The VMXAGE element can be used to match a particular VMXACF observation with a cycle of the VMXCPV file.

5.3.8.4 VMXACF Retrieval Examples

List the configuration of CP-owned volumes that are shown in the VMXACF file.

```
PROC SORT DATA=&VMXX..VMXACF01 OUT=CPVOLS;
  BY SYSID CPVATYPE VOLSER; RUN;
PROC PRINT NOOBS SPLIT='*';
  BY SYSID CPVATYPE;
  PAGEBY SYSID;
  SUM CPVCYLAR;
  ID VOLSER ;
  VAR CPVCYLAR;
  LABEL VOLSER='Volume*Serial*Number'
        SYSID='System'
        CPVCYLAR='Cylinders*in Area'
        CPVATYPE='Area Type' ;
  TITLE "CP Owned Volume Inventory";
RUN;
```

5.3.9 VMX CP Owned Volume Data File (VMXCPV)

The VMX CP Owned Volume Data File (VMXCPV) reports on the activity of VM/ESA system volumes.

It is derived from data in VM/ESA Monitor domain 3, records 4, 7, and 11; and domain 1, record 8.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.9.1 VMXCPV 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	VOLSER HOUR	VMXDVSID 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 5-27. VMXCPV Sort Sequence and Data Granularity

5.3.9.2 VMXCPV 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  HOUR     - Hour of Day
 X.....E  MONTH    - Month of Year
 X.....E  SYSID    - System Identifier
 X.....E  VMXDVSID - Host Subchannel ID Number
 X.....E  VOLSER   - Volume Serial Number
 X.....E  WEEK     - Week of Year
 X.....E  YEAR     - Year of Century
 X.....E  ZONE     - Time Zone
```

```
Common Data Elements

 X.....E  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  DURATION - Recording Interval Time
 X.....E  ENDTS    - End Time Stamp
 X.....E  INTERVLS - Number of Recording Intervals
 X.....E  STARTTS  - Start Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXDVDEV - Device Number
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier
 X.....E  VMXMSSCH - Count of SSCH and RSCH for This Device

Retained Data Elements

 X.....E  CPVCPVOL - CP Owned Volume Indicator
 X.....E  CPVDUMPV - Dump Only Volume Indicator
 X.....   CPVDVFLG - Volume Flag
 X.....E  CPVMDISK - Mapped Minidisk Volume Indicator
 X.....E  CPVPAGAR - Number of Page Areas on Volume
 X.....E  CPVPGCYL - Pages Per Cylinder
 X.....E  CPVSHRD  - CSE Shared Volume Indicator
 X.....E  CPVSPLAR - Number of Spool Areas on Volume
```

```
Accumulated Data Elements

 X.....E  CPVALOCD - Times volume Allocated
 X.....E  CPVCFS1  - Times Contiguous Free Space Found
 X.....E  CPVCFS10 - Times Contiguous Free Space Found
 X.....E  CPVCFS11 - Times Contiguous Free Space Found
 X.....E  CPVCFS12 - Times Contiguous Free Space Found
 X.....E  CPVCFS13 - Times Contiguous Free Space Found
 X.....E  CPVCFS14 - Times Contiguous Free Space Found
 X.....E  CPVCFS15 - Times Contiguous Free Space Found
 X.....E  CPVCFS16 - Times Contiguous Free Space Found
 X.....E  CPVCFS17 - Times Contiguous Free Space Found
 X.....E  CPVCFS18 - Times Contiguous Free Space Found
 X.....E  CPVCFS19 - Times Contiguous Free Space Found
 X.....E  CPVCFS2  - Times Contiguous Free Space Found
 X.....E  CPVCFS20 - Times Contiguous Free Space Found
 X.....E  CPVCFS3  - Times Contiguous Free Space Found
 X.....E  CPVCFS4  - Times Contiguous Free Space Found
 X.....E  CPVCFS5  - Times Contiguous Free Space Found
 X.....E  CPVCFS6  - Times Contiguous Free Space Found
 X.....E  CPVCFS7  - Times Contiguous Free Space Found
 X.....E  CPVCFS8  - Times Contiguous Free Space Found
 X.....E  CPVCFS9  - Times Contiguous Free Space Found
 X.....E  CPVDEVST - Device Service Time
 X.....E  CPVIOAXC - Page/Spool I/Os Added Without SSCH
 X.....E  CPVIOQCT - Page/Spool I/Os Queued or Active
 X.....E  CPVLOKAT - Times Volume Examined for Allocation
 X.....E  CPVMLOAD - Modified Device Load
 X.....E  CPVPAGRD - Total Page Reads
 X.....E  CPVPAGSL - Page Slots in Use on This Volume
 X.....E  CPVPAGWR - Total Page Writes
 X.....E  CPVPSLOT - Total Page Slots on This Volume
 X.....E  CPVSPLRD - Total Spool Reads
 X.....E  CPVSPLSL - Spool Slots in Use on This Volume
 X.....E  CPVSPLWR - Total Spool Writes
 X.....E  CPVSSLOT - Total Spool Slots on This Volume
 X.....E  CPVUSRIO - User I/O Interference Count

Maximum Data Elements

 X.....E  CPVMXIOQ - Maximum Page/Spool I/Os Queued
 X.....E  CPVMXPSL - Maximum Page Slots In Use
 X.....E  CPVMXSSL - Maximum Spool Slots In Use

Derived Data Elements

 X.....E  CPVPCPSU - Percent Page Slots Used
 X.....E  CPVPCSSU - Percent Spool Slots Used
```

5.3.9.3 VMXCPV Usage Considerations

Special considerations or techniques for using the VMXCPV File are shown below.

- The VMXCPV file contains one observation per subchannel to a device. Activity for all areas on the device are reported by VM/ESA at the volume level. To see all activity for a volume, the file must be summarized by VOLSER.

- The VMXXDV file contains additional information about CP owned volumes. The MONITOR ENABLE I/O VOL CPVOL command allows specification of data collection for CP-owned volumes as a group.

5.3.9.4 VMXCPV Retrieval Examples

Print a summary of I/Os queued and executed by VOLSER.

```
%VMCBRK(FILE=VMXCPV01,PERIOD=15);
RUN;
%LET BY = SYSID VOLSER VMCINTTS;
PROC SORT DATA=VMXCPV;
  BY &BY;
RUN;
DATA VMXCPV;
  SET VMXCPV; BY &BY;
  IF NOT FIRST.VMCINTTS THEN DO;
    INTERVLS = 0;
    DURATION = 0;
  END;
RUN;
%LET BY = SYSID VOLSER DATE TIME;
%LET BREAK = TIME;
DATA VMXCPV;
  SET VMXCPV;
  %CPVSUM(DERV=YES);
RUN;
PROC PRINT LABEL;
  BY SYSID VOLSER;
  ID VMCINTTS;
  VAR DURATION;
  SUM CPVIOQCT VMXMSSCH;
RUN;
```

5.3.10 VMX User Data Configuration File (VMXVXF)

The VMX User Data Configuration File (VMXVXF) identifies the virtual machines (users) that are logged on and the configuration of the virtual machine.

It is derived from data in VM/ESA Monitor domain 1, record 15 and domain 4, records 1, 2, 5, 6, and 7.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.10.1 VMXVXF 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	USER	VMXCPUAD	YEAR	MONTH
	DAY	HOUR	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 5-28. VMXVXF Sort Sequence and Data Granularity

5.3.10.2 VMXVXF 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  DAY      - Day of Month
 X.....E  HOUR     - Hour of Day
 X.....E  MONTH    - Month of Year
 X.....E  SYSID    - System Identifier
 X.....E  USER     - User Identification
 X.....E  VMXCPUAD - Processor Address
 X.....E  WEEK     - Week of Year
 X.....E  YEAR     - Year of Century
```

```
 X.....E  ZONE     - Time Zone

Common Data Elements

 X.....   CPUMODEL - CPU Model Identification
 X.....   CPUSER   - CPU/Processor Serial Number
 X.....   DAYNAME  - Name of Day of Week
 X.....E  ENDTS    - End Time Stamp
 X.....   INTERVLS - Number of Recording Intervals
 X.....   MICSVER  - CA MICS Version Number
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXAGE   - Observation Age
 X.....E  VMXCPUAF - CPU Affinity setting
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier
 X.....E  VMXSTCHG - Status Change
 X.....E  VMXVTYPE - Virtual CPU Type
```

```
Retained Data Elements

 X.....E  VXFABCP  - CP ABSOLUTE Share
 X.....E  VXFABICF - ICF ABSOLUTE Share
 X.....E  VXFABIFA - zAAP ABSOLUTE Share
 X.....E  VXFABIFL - IFL ABSOLUTE Share
 X.....E  VXFABSSH - Virtual Machine Absolute Share
 X.....E  VXFABZIP - zIIP ABSOLUTE Share
 X.....E  VXFBYUSR - BYUSER ID Logged On
 X.....E  VXFCPCT  - Guest Defined CP CPUs
 X.....E  VXFCPUA2 - New Virtual CPU Address
 X.....E  VXFCPUCT - Virtual CPUs Defined
 X.....E  VXFCTPVG - DPA Private Resident Pages >= 2GB Count
 X.....E  VXFDEFSZ - Defined Storage Size for 64-Bit Systems
 X.....E  VXFDMODE - Virtual Machine Mode
 X.....E  VXFICFCT - Guest Defined ICF CPUs
 X.....E  VXFIFACT - Guest Defined zAAP CPUs
 X.....E  VXFIFLCT - Guest Defined IFL CPUs
 X.....E  VXFIIPCT - Guest Defined zIIP CPUs
 X.....E  VXFLIMTH - CPU MAX Share Limit Option
 X.....E  VXFLMCP  - CP MAX SHARE LIMIT Option
 X.....E  VXFLMICF - ICF MAX SHARE LIMIT Option
 X.....E  VXFLMIFA - zAAP MAX SHARE LIMIT Option
 X.....E  VXFLMIFL - IFL MAX SHARE LIMIT Option
 X.....E  VXFLMZIP - zIIP MAX SHARE LIMIT Option
 X.....E  VXFLOGON - Virtual Machine Logged On Time
 X.....E  VXFMSCP  - CP MAX Share Type
 X.....E  VXFMSICF - ICF MAX Share Type
 X.....E  VXFMSIFA - zAAP MAX Share Type
 X.....E  VXFMSIFL - IFL MAX Share Type
 X.....E  VXFMSZIP - zIIP MAX Share Type
 X.....E  VXFMVB2G - Times Page Moved from >= 2GB to < 2GB
 X.....E  VXFMXCP  - CP MAX Share
 X.....E  VXFMXICF - ICF MAX Share
 X.....E  VXFMXIFA - zAAP MAX Share
 X.....E  VXFMXIFL - IFL MAX Share
 X.....E  VXFMXSHA - CPU MAX Share Type
 X.....E  VXFMXSHR - User's MAX Share of the System
 X.....E  VXFMXZIP - zIIP MAX Share
 X.....E  VXFNEWTY - New CPU Type
 X.....E  VXFNMCP  - CP Normal Share Type
 X.....E  VXFNMICF - ICF Normal Share Type
 X.....E  VXFNMIFA - zAAP Normal Share Type
 X.....E  VXFNMIFL - IFL Normal Share Type
 X.....E  VXFNMZIP - zIIP Normal Share Type
 X.....E  VXFOLDTY - Former CPU Type Before DEFINE CPU nn Typ
 X.....E  VXFRELSH - Virtual Machine Relative Share
 X.....E  VXFRSCP  - CP RELATIVE Share
 X.....E  VXFRSICF - ICF RELATIVE Share
```

```
X.....E  VXFRSIFA - zAAP RELATIVE Share
X.....E  VXFRSIFL - IFL RELATIVE Share
X.....E  VXFRSVPG - Number of Reserved Pages
X.....E  VXFRSZIP - zIIP RELATIVE Share
X.....E  VXFSSIZE - Virtual Machine Size
X.....E  VXFSTAT  - Virtual Machine Information
X.....E  VXFSTOP  - Virtual CPU is Soft Stopped
X.....E  VXFSTOPD - Virtual CPU is Stopped
X.....E  VXFSTYPE - Virtual Machine Storage Type
X.....E  VXFSVM   - Service Virtual Machine Flag
X.....E  VXFVMARC - Virtual Machine Architecture
```

5.3.10.3 VMXVXF Usage Considerations

Special considerations or techniques related to using the VMXVXF File are provided below:

- The VMXVXF File is a spin file that contains one observation per virtual machine logged on monitor initialization, and one observation for each time a new VMDBK is created in the system or removed from the system while the monitor is running. A VMDBK is created by logon or DEFINE CPU, and removed by LOGOFF or DETACH CPU.

- Only one cycle exists in the detail timespan. This file contains observations from several runs of the DAILY job. Each cycle is identified by the element VMXAGE. VMXAGE is 1 for the current cycle, 2 for the previous cycle, etc. The VMXAGE element can be used to match a particular VMXVXF observation with a cycle of the VMXVXU and VMXVXT Files.

5.3.10.4 VMXVXF Retrieval Examples

List userids in the VMXVXF that are XA mode or have some special performance preference.

```
DATA _NULL_;
SET &VMXX..VMXVXF01;
  LENGTH LINE $80.;
  FILE PRINT NOTITLES HEADER=HEADER;
  IF VMXSTCHG = 'IC' OR VMXSTCHG = 'LN';
  OK1 = VXFDMODE = '..1.....'B;          /* XA       */
  OK2 = VXFDMODE = '....1...'B;          /* V=R      */
  OK3 = VXFABSSH NE 0;                   /* ABSSHARE */
  IF NOT OK3 THEN OK4 = VXFRELSH NE 100; /* RELSHARE */
  OK5 = VXFSVM;                          /* SVM      */
  OK6 = VXFCPUCT > 0;                    /* VIRT MP  */
  IF OK1 OR OK2 OR OK3 OR OK4 OR OK5 OR OK6;
  IF OK1 THEN SUBSTR(LINE,1,2) = 'XA';
  IF OK2 THEN SUBSTR(LINE,4,3) = 'V=R';
  IF OK5 THEN SUBSTR(LINE,8,3) = 'SVM';
  IF OK6 THEN
    SUBSTR(LINE,12,8) = 'CPUs: ' || PUT(VXFCPUCT,2.);
  VXFABSSH = VXFABSSH*100;
  IF OK3 THEN DO;
    IF VXFABSSH < 0
      THEN SUBSTR(LINE,21,22) = 'Dedicated CPU';
      ELSE SUBSTR(LINE,21,22) =
        'Absolute Share: ' || PUT(VXFABSSH,5.2) || ' %';
  END;
  IF OK4 THEN SUBSTR(LINE,21,22) =
        'Relative Share: ' || PUT(VXFRELSH,5.);
  PUT @10 USER $CHAR8. +3 LINE $CHAR50.;
RETURN;
HEADER:
  PUT @5  'Userids Specially Configured on System '
          SYSID /
      @11 'Userid    Special Conditions' /;
RETURN;
RUN;
```

5.3.11 VMX User Activity Data Files (VMXVXU and VMX_VU)

The VMX User Activity Sample Data File (VMXVXU) quantifies the resource consumption of each virtual machine, based on sampling.

It is derived mainly from data in VM/ESA Monitor domain 4, record 3; domain 3, record 10; and optionally from domain 4, record 2 if user event data is collected in a VM/ESA Release 2 or later system. Additional user configuration information is derived from domain 1, record 15 and domain 4, record 1.

The VMX User Activity Event Data File (VMX_VU) quantifies the resource consumption of each virtual machine at interaction end.

It is derived from VM/ESA Monitor data, domain 4, record 9. Additional user configuration information is derived from domain 1, record 15 and domain 4, record 1.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.11.1 VMXVXU and VMX_VU 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	USER	VMXCPUAD	VMXVTYPE	YEAR
	MONTH	DAY	HOUR	ENDTS	
DAYS	SYSID	USER	VMXCPUAD	VMXVTYPE	YEAR
	MONTH	DAY	HOUR		
WEEKS	N/A				
MONTHS	SYSID	USER	VMXCPUAD	VMXVTYPE	YEAR
	MONTH	ZONE			
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 5-29. VMXVXU Sort Sequence and Data Granularity

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	USER	VMXCPUAD	VMXVTYPE	YEAR
	MONTH	DAY	HOUR	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 5-30. VMX_VU Sort Sequence and Data Granularity

5.3.11.2 VMXVXU and VMX_VU Element Lists

```
VMXVXU ELEMENTS

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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 XD....E  DAY      - Day of Month
 XD....E  HOUR     - Hour of Day
 XD.M..E  MONTH    - Month of Year
 XD.M..E  SYSID    - System Identifier
 XD.M..E  USER     - User Identification
```

```
 XD.M..E  VMXCPUAD - Processor Address
 XD.M..E  VMXVTYPE - Virtual CPU Type
 XD....E  WEEK     - Week of Year
 XD.M..E  YEAR     - Year of Century
 XD.M..E  ZONE     - Time Zone

Common Data Elements

 XD.M..   CPUMODEL - CPU Model Identification
 XD.M..   CPUSER   - CPU/Processor Serial Number
 XD....   DAYNAME  - Name of Day of Week
 XD.M..   DURATION - Recording Interval Time
 XD.M..E  ENDTS    - End Time Stamp
 XD.M..   INTERVLS - Number of Recording Intervals
 XD.M..   MICSVER  - CA MICS Version Number
 XD.M..   STARTTS  - Start Time Stamp
 X.....E  VMCACNT  - VM Account Number
 XD.M..E  VMCINTTS - VM Interval Time Stamp
 XD.M..E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMCSGRPN - Security Group Name
 XD.M..E  VMLEVEL  - VM System Release Level
 X.....E  VMXCPUAF - CPU Affinity setting
 XD.M..E  VMXMOD   - CPU Processor Model Family
 XD.M..E  VMXMODID - CPU Model Identifier
```

```
Retained Data Elements

 XD.M..E  VXUABSSH - Virtual Machine Absolute Share
 XD.M..E  VXUBLKL  - Number of Last XSTOR Block Attached
 XD.M..E  VXUBLK1  - Number of First XSTOR Block Attached
 XD.M..E  VXUCTPVG - DPA Private Resident Pages >= 2GB Count
 XD.M..E  VXUDEFSZ - Defined Storage Size for 64-Bit Systems
 XD.M..E  VXUDLIST - Scheduling Class
 XD.M..E  VXUDMODE - Virtual Machine Mode
 XD.M..E  VXULIMTH - CPU MAX Share Limit Option
 XD.M..E  VXUMCLS  - Data Measurement Class
 XD.M..E  VXUMVB2G - Times Page Moved from >= 2GB to < 2GB
 XD.M..E  VXUMXSHA - CPU MAX Share Type
 XD.M..E  VXUMXSHR - User's MAX Share of the System
 XD.M..E  VXURELSH - Virtual Machine Relative Share
 XD.M..E  VXUSLIST - Scheduling List
 XD.M..E  VXUSSIZE - Virtual Machine Size
 XD.M..E  VXUSTAT  - Virtual Machine Information
 XD.M..E  VXUSTOP  - Virtual CPU is Soft Stopped
 XD.M..E  VXUSTOPD - Virtual CPU is Stopped
 XD.M..E  VXUSTYPE - Virtual Storage Type
 XD.M..E  VXUSVM   - Service Virtual Machine Flag
 XD.M..E  VXUVMARC - Virtual Machine Architecture
 XD.M..E  VXUVMDBK - VMDBK Information
 XD.M..E  VXUVMXFR - Expanded Storage Attached
 XD.M..E  VXUXSIZE - Size of XSTOR Attached
```

```
Accumulated Data Elements

 XD.M..E  VXUBLKIO - Count of SSCHs Issued by *BLOCKIO
 XD.M..E  VXUCACHN - Number of Minidisk Inserts
 XD.M..E  VXUCONIO - Start Requests to Virtual Console
 XD.M..E  VXUCPXFR - CP Expanded Storage Frames Occupied
 XD.M..   VXUCRPDP - Times Redispatched for Crypto Op Excep
 XD.M..E  VXUCTASC - Number of Non-Base Addr Spaces Owned
 XD.M..E  VXUCTCIO - Start Requests to Virtual CTCA
 XD.M..E  VXUCTMIG - Pages Migrated to DASD from XSTOR
 XD.M..E  VXUCTSTA - Times Virtual CPU was Started via -SIGP
 XD.M..E  VXUCTSTO - Times Virtual CPU was Stopped via -SIGP
 XD.M..E  VXUCTXRD - Number of PAGINs
 XD.M..E  VXUCTXWT - Number of PAGOUTs
 XD.M..E  VXUDASIO - Start Requests to Virtual DASD
 XD.M..E  VXUESWSS - Estimated Working Set Size
 XD.M..E  VXUFLREO - Frame List Reorders
 XD.M..E  VXUIUCCT - IUCV Connections
 XD.M..E  VXUMDCIA - Count of SSCHs Avoided by MDC Read Hits
 XD.M..E  VXUMSCIO - Start Requests to Other Virtual Devices
 XD.M..E  VXUMXCON - Connections from MAXCONN Directory
 XD.M..E  VXUNPSLT - Non-Preferred Page Slots Occupied
 XD.M..E  VXUPGFLT - Page Faults
 XD.M..E  VXUPGLOK - Pages Locked
 XD.M..E  VXUPGRES - Pages Resident (for Private Addr Spaces)
 XD.M..E  VXUPGRIO - Page Read I/Os Issued
 XD.M..E  VXUPGSPL - Pages Spooled
 XD.M..E  VXUPGWIO - Page Write I/Os Issued
 XD.M..E  VXUPNDIO - Pending Spool/Page I/O Count
 XD.M..E  VXURCVCT - IUCV Messages On RECEIVE Queue
 XD.M..E  VXUREPCT - IUCV Messages On REPLY Queue
 XD.M..E  VXURESET - Referenced Resident Frames at Reset
 XD.M..E  VXURSVPG - Number of Reserved Pages
 XD.M..E  VXUSFIUC - Data Successfully Sent by IUCV From User
 XD.M..E  VXUSFVMC - Data Successfully Sent by VMCF From User
 XD.M..E  VXUSNDCT - IUCV Messages On SEND Queue
 XD.M..E  VXUSPRIO - Spool Read I/Os Issued
 XD.M..E  VXUSPWIO - Spool Write I/Os Issued
 XD.M..E  VXUSTEAL - Pages Stolen From Virtual Machine
 XD.M..E  VXUSTIUC - Data Successfully Sent by IUCV To User
 XD.M..E  VXUSTVMC - Data Successfully Sent by VMCF To User
 XD.M..E  VXUTHRCT - Total of I/O Delays
 XD.M..E  VXUTTIME - Total CPU Time Used
 XD.M..E  VXUTVPTM - Total Virtual CPU Time on Primary Proces
 XD.M..E  VXUTVSTM - Total Virtual CPU Time on Secondary
 XD.M..E  VXUUTIUC - Data Not Successfully Sent by IUCV
 XD.M..E  VXUUTVMC - Data Not Successfully Sent by VMCF
 XD.M..E  VXUVIOCT - Count of Virtual I/Os at Logoff
 XD.M..E  VXUVMCCT - Messages On VMCF Queue
```

```
 XD.M..E  VXUVSPTM - Virtual + Simulation CPU Time on Primary
 XD.M..E  VXUVSSTM - Virtual + Simulation CPU Time on Seconda
 XD.M..E  VXUVTIME - Virtual CPU Time Used
 XD.M..E  VXUVURIO - Start Requests to Virtual Unit Record
 XD.M..E  VXUX98CT - Count of Diagnose 98 Issued

VMX_VU ELEMENTS


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 &quot;XDWMYT&quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------
```

```
Sequence/Summary Data Elements

 X.....E  DAY      - Day of Month
 X.....E  HOUR     - Hour of Day
 X.....E  MONTH    - Month of Year
 X.....E  SYSID    - System Identifier
 X.....E  USER     - User Identification
 X.....E  VMXCPUAD - Processor Address
 X.....E  VMXVTYPE - Virtual CPU Type
 X.....E  WEEK     - Week of Year
 X.....E  YEAR     - Year of Century
 X.....E  ZONE     - Time Zone

Common Data Elements

 X.....   CPUMODEL - CPU Model Identification
 X.....   CPUSER   - CPU/Processor Serial Number
 X.....   DAYNAME  - Name of Day of Week
 X.....   DURATION - Recording Interval Time
 X.....E  ENDTS    - End Time Stamp
 X.....   INTERVLS - Number of Recording Intervals
 X.....   MICSVER  - CA MICS Version Number
 X.....   STARTTS  - Start Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXCPUAF - CPU Affinity setting
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier

Retained Data Elements

 X.....E  VXUABSSH - Virtual Machine Absolute Share
 X.....E  VXUBLKL  - Number of Last XSTOR Block Attached
 X.....E  VXUBLK1  - Number of First XSTOR Block Attached
 X.....E  VXUCTPVG - DPA Private Resident Pages >= 2GB Count
 X.....E  VXUDEFSZ - Defined Storage Size for 64-Bit Systems
 X.....E  VXUDMODE - Virtual Machine Mode
 X.....E  VXULIMTH - CPU MAX Share Limit Option
 X.....E  VXUMCLS  - Data Measurement Class
 X.....E  VXUMVB2G - Times Page Moved from >= 2GB to < 2GB
 X.....E  VXUMXSHA - CPU MAX Share Type
 X.....E  VXUMXSHR - User's MAX Share of the System
 X.....E  VXUPRXTS - Observation Proximity Timestamp
 X.....E  VXURELSH - Virtual Machine Relative Share
 X.....E  VXUSTAT  - Virtual Machine Information
 X.....E  VXUSTOP  - Virtual CPU is Soft Stopped
 X.....E  VXUSTOPD - Virtual CPU is Stopped
```

```
 X.....E  VXUSVM   - Service Virtual Machine Flag
 X.....E  VXUVMARC - Virtual Machine Architecture
 X.....E  VXUVMDBK - VMDBK Information
 X.....E  VXUXSIZE - Size of XSTOR Attached

Accumulated Data Elements

 X.....   VXUBLKIO - Count of SSCHs Issued by *BLOCKIO
 X.....E  VXUCACHN - Number of Minidisk Inserts
 X.....E  VXUCONIO - Start Requests to Virtual Console
 X.....E  VXUCPXFR - CP Expanded Storage Frame Occupied
 X.....   VXUCRPDP - Times Redispatched for Crypto Op Excep
 X.....E  VXUCTCIO - Start Requests to Virtual CTCA
 X.....E  VXUCTMIG - Pages Migrated to DASD from XSTOR
 X.....E  VXUCTSTA - Times Virtual CPU was Started via -SIGP
 X.....E  VXUCTSTO - Times Virtual CPU was Stopped via -SIGP
 X.....E  VXUCTXRD - Number of PAGINs
 X.....E  VXUCTXWT - Number of PAGOUTs
 X.....E  VXUDASIO - Start Requests to Virtual DASD
 X.....E  VXUESWSS - Estimated Working Set Size
 X.....   VXUMDCIA - Count of SSCHs Avoided by MDC Read Hits
 X.....E  VXUMSCIO - Start Requests to Other Virtual Devices
 X.....E  VXUNPSLT - Non-Preferred Page Slots Occupied
 X.....E  VXUPGFLT - Page Faults
 X.....E  VXUPGLOK - Pages Locked
 X.....E  VXUPGRES - Pages Resident
 X.....E  VXUPGRIO - Page Read I/Os Issued
 X.....E  VXUPGSPL - Pages Spooled
 X.....E  VXUPGWIO - Page Write I/Os Issued
 X.....E  VXUPNDIO - Pending Spool/Page I/O Count
 X.....E  VXURCVCT - IUCV Messages On RECEIVE Queue
 X.....E  VXUREPCT - IUCV Messages On REPLY Queue
 X.....E  VXURESET - Referenced Resident Frames at Reset
 X.....E  VXUSFIUC - Data Successfully Sent by IUCV From User
 X.....E  VXUSFVMC - Data Successfully Sent by VMCF From User
 X.....E  VXUSNDCT - IUCV Messages On SEND Queue
 X.....E  VXUSPRIO - Spool Read I/Os Issued
 X.....E  VXUSPWIO - Spool Write I/Os Issued
 X.....E  VXUSTEAL - Pages Stolen From Virtual Machine
 X.....E  VXUSTIUC - Data Successfully Sent by IUCV To User
 X.....E  VXUSTVMC - Data Successfully Sent by VMCF To User
 X.....E  VXUTTIME - Total CPU Time Used
 X.....E  VXUTVPTM - Total Virtual CPU Time on Primary Proces
 X.....E  VXUTVSTM - Total Virtual CPU Time on Secondary
 X.....E  VXUUTIUC - Data Not Successfully Sent by IUCV
 X.....E  VXUUTVMC - Data Not Successfully Sent by VMCF
 X.....E  VXUVIOCT - Count of Virtual I/Os at Logoff
 X.....E  VXUVMCCT - Messages On VMCF Queue
 X.....E  VXUVSPTM - Virtual + Simulation CPU Time on Primary
```

```
X.....E  VXUVSSTM - Virtual + Simulation CPU Time on Primary
X.....E  VXUVTIME - Virtual CPU Time Used
X.....E  VXUVURIO - Start Requests to Virtual Unit Record
X.....E  VXUX98CT - Count of Diagnose 98 Issued
```

5.3.11.3 VMXVXU and VMX_VU Usage Considerations

Special considerations or techniques for using the VMXVXU and VMX_VU Files are listed below:

- The VMXVXU File contains one observation per VMDBK for each userid that is monitored by sample data collection. If a virtual machine is a MP (has more than one VMDBK) data for each VMDBK must be summed to obtain total resource use for the interval.

- The VMX_VU File contains one observation for each interaction that is initiated by a userid that is monitored for event data collection. If a virtual machine is a MP (has more than one VMDBK) an observation will be produced only for the VMDBK that participated in the interaction.

- The VMXVXT and VMXVXU Files can be merged for analysis of interaction and wait states in conjunction with activity data. The elements USER, VMXCPUAD, and VMCINTTS should be used in the BY list to accomplish the merge.

- The VMX_VT and VMX_VU Files can be merged for analysis of resource consumption by interaction. The elements USER, VMXCPUAD and VXUPRXTS should be used in the BY list to accomplish the merge.

5.3.11.4 VMXVXU and VMX_VU Retrieval Examples

```
VMXVXU EXAMPLE

 Print CPU consumed for the top ten users on the system.

   DATA CPU(KEEP=SYSID USER DATE
       VXUTTIME VXUVTIME TVRATIO);
     SET &VMXX..VMXVXU01; BY SYSID USER;
     FORMAT DATE DATE7.;
     LABEL TVRATIO='Total to*Virtual Ratio'
     IF FIRST.USER THEN DO;
       Z1=0;Z2=0;
     END;
     Z1+VXUTTIME;
     Z2+VXUVTIME;
     IF LAST.USER;
     VXUTTIME = Z1;
     VXUVTIME = Z2;
     TVRATIO = VXUVTIME > 0;
     IF TVRATIO THEN TVRATIO = VXUTTIME/VXUVTIME;
     DATE = DATEPART(ENDTS);
   RETURN;
   PROC SORT;
     BY SYSID DATE DESCENDING VXUTTIME;
   RUN;
   DATA CPU;
     SET CPU;
     RANK+1;
     OUTPUT;
     IF RANK = 20 THEN STOP;
   RETURN;
   PROC PRINT  SPLIT='*';
     BY SYSID DATE;
     ID RANK;
     TITLE "Top 20 Users by CPU Consumption";
   RUN;

VMX_VU EXAMPLE

   Show interaction duration as a function of page faults.

   DATA MERGED(KEEP=SYSID USER
          VMXCPUAD ENDTS VXUPGFLT TRNTM);
     MERGE &VMXX..VMX_VU01 &VMXX..VMX_VT01;
     BY SYSID USER VMXCPUAD VXUPRXTS;
     LABEL TRNTM='Transaction Time';
     TRNTM = VXUSUSTS-VXUTRNTS;
   RUN;
```

```
PROC PLOT;
  PLOT TRNTM*VXUPGFLT /
  vaxis=0 to 25 by 5;
  TITLE
     "Interaction Duration as a Function of Page Faults";
RUN;
```

5.3.12 VMX User Transaction Data Files (VMXVXT and VMX_VT)

The VMX User Transaction Sample Data File (VMXVXT) records the interactions among users and between users and systems services using IUCV and VMCF, based on sampling.

It is derived mainly from data in VM/ESA Monitor domain 4, record 4; additional user configuration information is derived from domain 1, record 15 and domain 4, record 1.

The VMX User Transaction Event Data File (VMX_VT) records the interactions among users and between users and systems services using IUCV and VMCF, at interaction end.

It is derived from VM/ESA Monitor data, domain 4, records 8 and 10. User configuration information is derived from domain 1, record 15 and domain 4, record 1.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.12.1 VMXVXT and VMX_VT 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 MONTH	USER DAY	VMXCPUAD HOUR	VMXVTYPE ENDTS	YEAR
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 5-31. VMXVXT Sort Sequence and Data Granularity

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	USER	VMXCPUAD	VMXVTYPE	YEAR	
	MONTH	DAY	HOUR	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 5-32. VMX_VT Sort Sequence and Data Granularity

5.3.12.2 VMXVXT and VMX_VT Element Lists

```
VMXVXT ELEMENTS

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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  DAY      - Day of Month
 X.....E  HOUR     - Hour of Day
 X.....E  MONTH    - Month of Year
 X.....E  SYSID    - System Identifier
 X.....E  USER     - User Identification
 X.....E  VMXCPUAD - Processor Address
```

```
 X.....E  VMXVTYPE - Virtual CPU Type
 X.....E  WEEK     - Week of Year
 X.....E  YEAR     - Year of Century
 X.....E  ZONE     - Time Zone

Common Data Elements

 X.....   CPUMODEL - CPU Model Identification
 X.....   CPUSER   - CPU/Processor Serial Number
 X.....   DAYNAME  - Name of Day of Week
 X.....   DURATION - Recording Interval Time
 X.....E  ENDTS    - End Time Stamp
 X.....   INTERVLS - Number of Recording Intervals
 X.....   MICSVER  - CA MICS Version Number
 X.....   STARTTS  - Start Time Stamp
 X.....E  VMCACNT  - VM Account Number
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMCSGRPN - Security Group Name
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXCPUAF - CPU Affinity setting
 X.....E  VMXHFSAM - High Frequency Sample Count
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier

Retained Data Elements

 X.....E  VXUABSSH - Virtual Machine Absolute Share
 X.....E  VXUCKST  - Virtual Machine in Check Stop State
 X.....E  VXUDBASE - Base VMDBK Indicator
 X.....E  VXUDLDTS - Time When VMDBK Dropped From DLIST
 X.....E  VXUDMODE - Virtual Machine Mode
 X.....E  VXUDSCWT - Virtual Machine in Disconnected Wait Mod
 X.....E  VXUDSTAT - Dispatch Status
 X.....E  VXUINTRN - VMDBK Transaction Indicator
 X.....E  VXUMCLS  - Data Measurement Class
 X.....E  VXUOSTAT - Operating Status
 X.....E  VXURELSH - Virtual Machine Relative Share
 X.....E  VXURSTAT - Running Status
 X.....E  VXUSLEEP - Virtual Machine is Sleeping
 X.....E  VXUSTAT  - Virtual Machine Information
 X.....E  VXUSUSTS - Time When VMDBK Marked Suspended
 X.....E  VXUSVM   - Service Virtual Machine Flag
 X.....E  VXUSVMID - Last SVM or CP Service Called via IUCV
 X.....E  VXUTRNTS - Time When VMDBK Transaction Started
 X.....E  VXUVMDBK - VMDBK Information
 X.....E  VXUVMSTD - Virtual Machine is Stopped
 X.....E  VXUVMSTS - Virtual Machine in Stop State
```

```
Accumulated Data Elements

 X.....E  VXUDFRWK - Number Deferred Work Items for VMDBK
 X.....E  VXUDSVMW - Times VMDBK Found Dormant and in SVM Wai
 X.....E  VXUESVMW - Times VMDBK Found in ELIST and SVM Wait
 X.....E  VXUEXTRF - Times IUCV/VMCF External Interrupt Made
 X.....E  VXUE0CT  - Number of Times Found in E0
 X.....E  VXUE1CT  - Number of Times Found in E1
 X.....E  VXUE2CT  - Number of Times Found in E2
 X.....E  VXUE3CT  - Number of Times Found in E3
 X.....E  VXUHFSAM - High Frequency Samples Taken
 X.....   VXUMTSCT - Count of Expired Minor Time Slices
 X.....E  VXUNASIO - Times VMDBK in DispList for Asynch I/O
 X.....   VXUNDLWT - Times VMDBK in Dispatch List in CPU Wait
 X.....E  VXUNTIDL - Times VMDBK Denied Test Idle Time
 X.....E  VXUPGFLA - Times VMDBK Had Active Page Fault Req's
 X.....E  VXUPRMQ0 - VMDBK Pre-Empted from Q0 for E1 Stg
 X.....E  VXUPRMQ1 - VMDBK Pre-Empted from Q1 for E1 Stg
 X.....E  VXUPRMQ2 - VMDBK Pre-Empted from Q2 for E1 Stg
 X.....E  VXUPRMQ3 - VMDBK Pre-Empted from Q3 for E1 Stg
 X.....E  VXUQ0CT  - Number of Times Found in Q0
 X.....E  VXUQ1CT  - Number of Times Found in Q1
 X.....E  VXUQ2CT  - Number of Times Found in Q2
 X.....E  VXUQ3CT  - Number of Times Found in Q3
 X.....E  VXURMIQ0 - VMDBK Removed from Q0 at TIDL End
 X.....E  VXURMIQ1 - VMDBK Removed from Q1 at TIDL End
 X.....E  VXURMIQ2 - VMDBK Removed from Q2 at TIDL End
 X.....E  VXURMIQ3 - VMDBK Removed from Q3 at TIDL End
 X.....E  VXURMTQ0 - VMDBK Removed from Q0 at End of Time
 X.....E  VXURMTQ1 - VMDBK Removed from Q1 at End of Time
 X.....E  VXURMTQ2 - VMDBK Removed from Q2 at End of Time
 X.....E  VXURMTQ3 - VMDBK Removed from Q3 at End of Time
 X.....E  VXURMWQ0 - VMDBK Removed from Q0 for WSS Growth
 X.....E  VXURMWQ1 - VMDBK Removed from Q1 for WSS Growth
 X.....E  VXURMWQ2 - VMDBK Removed from Q2 for WSS Growth
 X.....E  VXURMWQ3 - VMDBK Removed from Q3 for WSS Growth
 X.....E  VXUSVMWT - Times in SVM Wait
 X.....E  VXUSVMW2 - Times in SVM Wait Backup
 X.....E  VXUTCFWT - Times VMDBK Found Console Function Wait
 X.....E  VXUTCPWT - Times VMDBK Found in CPU Wait
 X.....E  VXUTDORM - Times VMDBK Found in Dormant List
 X.....E  VXUTIDL  - Times VMDBK Found in TEST IDLE State
 X.....E  VXUTIOWT - Times VMDBK Found in I/O Wait
 X.....E  VXUTLOAD - Times VMDBK Was Classified as Loading
 X.....E  VXUTOTHR - Times VMDBK Found in Undefined State
 X.....E  VXUTPGWT - Times VMDBK Found in PAGE Wait
 X.....E  VXUTRUN  - Times VMDBK Found Running on a Real CPU
 X.....E  VXUTSMWT - Times VMDBK Found in Simulation Wait
 X.....E  VXUTSVM  - Times VMDBK TEST IDLE and SVM State
```

```
VMX_VT ELEMENTS

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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  DAY      - Day of Month
 X.....E  HOUR     - Hour of Day
 X.....E  MONTH    - Month of Year
 X.....E  SYSID    - System Identifier
 X.....E  USER     - User Identification
 X.....E  VMXCPUAD - Processor Address
 X.....E  VMXVTYPE - Virtual CPU Type
 X.....E  WEEK     - Week of Year
```

```
 X.....E  YEAR     - Year of Century
 X.....E  ZONE     - Time Zone

Common Data Elements

 X.....   CPUMODEL - CPU Model Identification
 X.....   CPUSER   - CPU/Processor Serial Number
 X.....   DAYNAME  - Name of Day of Week
 X.....   DURATION - Recording Interval Time
 X.....E  ENDTS    - End Time Stamp
 X.....   INTERVLS - Number of Recording Intervals
 X.....   MICSVER  - CA MICS Version Number
 X.....   STARTTS  - Start Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXCPUAF - CPU Affinity setting
 X.....E  VMXHFSAM - High Frequency Sample Count
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier

Retained Data Elements

 X.....E  VXUABSSH - Virtual Machine Absolute Share
 X.....E  VXUCKST  - Virtual Machine in Check Stop State
 X.....E  VXUDBASE - Base VMDBK Indicator
 X.....E  VXUDLDTS - Time When VMDBK Dropped From DLIST
 X.....E  VXUDMODE - Virtual Machine Mode
 X.....E  VXUDSCWT - Virtual Machine in Disconnected Wait Mod
 X.....E  VXUDSTAT - Dispatch Status
 X.....E  VXUINTRN - VMDBK Transaction Indicator
 X.....E  VXUMCLS  - Data Measurement Class
 X.....E  VXUOSTAT - Operating Status
 X.....E  VXUPRXTS - Observation Proximity Timestamp
 X.....E  VXURELSH - Virtual Machine Relative Share
 X.....E  VXURSTAT - Running Status
 X.....E  VXUSLEEP - Virtual Machine is Sleeping
 X.....E  VXUSTAT  - Virtual Machine Information
 X.....E  VXUSUSTS - Time When VMDBK Marked Suspended
 X.....E  VXUSVM   - Service Virtual Machine Flag
 X.....E  VXUSVMID - Last SVM or CP Service Called via IUCV
 X.....E  VXUTRIV  - Trivial Transaction Flag
 X.....E  VXUTRNTS - Time When VMDBK Transaction Started
 X.....E  VXUVMDBK - VMDBK Information
 X.....E  VXUVMSTD - Virtual Machine is Stopped
 X.....E  VXUVMSTS - Virtual Machine in Stop State
```

```
Accumulated Data Elements

 X.....E  VXUDFRWK - Number Deferred Work Items for VMDBK
 X.....E  VXUDSVMW - Times VMDBK Found Dormant and in SVM Wai
 X.....E  VXUESVMW - Times VMDBK Found in ELIST and SVM Wait
 X.....E  VXUEXTRF - Times IUCV/VMCF External Interrupt Made
 X.....E  VXUE0CT  - Number of Times Found in E0
 X.....E  VXUE1CT  - Number of Times Found in E1
 X.....E  VXUE2CT  - Number of Times Found in E2
 X.....E  VXUE3CT  - Number of Times Found in E3
 X.....E  VXUHFSAM - High Frequency Samples Taken
 X.....E  VXUNASIO - Times VMDBK in DispList for Asynch I/O
 X.....E  VXUNDLWT - Times VMDBK in Dispatch List in CPU Wait
 X.....E  VXUNTIDL - Times VMDBK Denied Test Idle Time
 X.....E  VXUNTRCT - Count of Non-Trivial Transactions
 X.....E  VXUPGFLA - Times VMDBK Had Active Page Fault Req's
 X.....E  VXUPRMQ0 - VMDBK Pre-Empted from Q0 for E1 Stg
 X.....E  VXUPRMQ1 - VMDBK Pre-Empted from Q1 for E1 Stg
 X.....E  VXUPRMQ2 - VMDBK Pre-Empted from Q2 for E1 Stg
 X.....E  VXUPRMQ3 - VMDBK Pre-Empted from Q3 for E1 Stg
 X.....E  VXUQ0CT  - Number of Times Found in Q0
 X.....E  VXUQ1CT  - Number of Times Found in Q1
 X.....E  VXUQ2CT  - Number of Times Found in Q2
 X.....E  VXUQ3CT  - Number of Times Found in Q3
 X.....E  VXURMIQ0 - VMDBK Removed from Q0 at TIDL End
 X.....E  VXURMIQ1 - VMDBK Removed from Q1 at TIDL End
 X.....E  VXURMIQ2 - VMDBK Removed from Q2 at TIDL End
 X.....E  VXURMIQ3 - VMDBK Removed from Q3 at TIDL End
 X.....E  VXURMTQ0 - VMDBK Removed from Q0 at End of Time
 X.....E  VXURMTQ1 - VMDBK Removed from Q1 at End of Time
 X.....E  VXURMTQ2 - VMDBK Removed from Q2 at End of Time
 X.....E  VXURMTQ3 - VMDBK Removed from Q3 at End of Time
 X.....E  VXURMWQ0 - VMDBK Removed from Q0 for WSS Growth
 X.....E  VXURMWQ1 - VMDBK Removed from Q1 for WSS Growth
 X.....E  VXURMWQ2 - VMDBK Removed from Q2 for WSS Growth
 X.....E  VXURMWQ3 - VMDBK Removed from Q3 for WSS Growth
 X.....E  VXUSVMWT - Times in SVM Wait
 X.....E  VXUSVMW2 - Times in SVM Wait Backup
 X.....E  VXUTCFWT - Times VMDBK Found Console Function Wait
 X.....E  VXUTCPWT - Times VMDBK Found in CPU Wait
 X.....E  VXUTDORM - Times VMDBK Found in Dormant List
 X.....E  VXUTIDL  - Times VMDBK Found in TEST IDLE State
 X.....E  VXUTIOWT - Times VMDBK Found in I/O Wait
 X.....E  VXUTLOAD - Times VMDBK Was Classified as Loading
 X.....E  VXUTOTHR - Times VMDBK Found in Undefined State
 X.....E  VXUTPGWT - Times VMDBK Found in PAGE Wait
 X.....E  VXUTRUN  - Times VMDBK Found Running on a Real CPU
 X.....E  VXUTRVCT - Count of Trivial Transactions
 X.....E  VXUTSMWT - Times VMDBK Found in Simulation Wait
```

```
X.....E  VXUTSVM  - Times VMDBK TEST IDLE and SVM State
```

5.3.12.3 VMXVXT Usage Considerations

Special considerations or techniques for using the VMXVXT and VMX_VT Files are provided below:

- The VMXVXT File contains one observation per VMDBK for each userid that is monitored by sample data collection. If a virtual machine is an MP (has more than one VMDBK) data for each VMDBK must be summed to obtain total resource use for the interval.

- The VMX_VT File contains one observation for each interaction that is initiated by a userid that is monitored for event data collection. If a virtual machine is an MP (has more than one VMDBK) an observation will be produced only for the VMDBK that participated in the interaction.

- The VMXVXT and VMXVXU Files can be merged for analysis of interaction and wait states in conjunction with activity data. To accomplish the merge, the elements USER, VMXCPUAD and VMCINTTS should be used in the BY list.

- The VMX_VT and VMX_VU Files can be merged for analysis of resource consumption by interaction. To accomplish the merge, the elements USER, VMXCPUAD and VXUPRXTS should be used in the BY list.

5.3.12.4 VMXVXT and VMX_VT Retrieval Examples

```
VMXVXT EXAMPLE

List users ranked by IUCV and VCMF activity as measured by
external interrupts generated as a part of using these
facilities.

  %VMCBRK(FILE=VMXVXT01,
          START='08:45'T,END='09:00'T,PERIOD=15);
  RUN;
  %LET BY = SYSID DATE USER TIME;
  %LET BREAK = TIME;
  PROC SORT DATA=VMXVXT;
    BY &BY;
  RUN;
  DATA VMXVXT;
    SET VMXVXT;
    %VXTSUM(DERV=YES);
  RUN;
  PROC SORT;
    BY SYSID DATE DESCENDING VXUEXTRF;
  RUN;
  DATA VMXVXT;
    SET VMXVXT;
    RANK+1;
    OUTPUT;
    IF RANK = 20 THEN STOP;
  RETURN;
  PROC PRINT  SPLIT='*'; BY SYSID DATE;
    ID RANK;
    VAR USER VXUEXTRF VXUSVMWT
             VXUE0CT VXUE1CT VXUE2CT VXUE3CT
             VXUQ0CT VXUQ1CT VXUQ2CT VXUQ3CT;
    TITLE "Top 20 Users by IUCV/VMCF Activity";
  RUN;


VMX_VT EXAMPLE

Plot the elapsed times between interaction start and
interaction end (time suspended) and between interaction end
and dispatch list drop.  The transaction start to end time
will be displayed as a negative number.

  %VMCBRK(FILE=VMX_VT01,
          START='08:30'T,END='08:45'T,PERIOD=15);
  IF USER =: 'AL';              /* SUBSET USERS          */
  DLD = VXUDLDTS - VXUSUSTS;    /* TIME TO DLIST DROP    */
```

```
TRN = VXUTRNTS - VXUSUSTS;   /* TIME FROM TRANS START */
RUN;
PROC TIMEPLOT;
  BY SYSID USER VMXCPUAD;
  PLOT TRN = '<' DLD = '>'
     / REF=0 JOINREF OVERLAY  AXIS=-2.5 TO .5 BY .05;
  ID VXUTRNTS VXUSVMID;
  LABEL DLD='Trans. End to DLIST Drop'
        TRN='Trans. Start to End';
RUN;
```

5.3.13 VMX Channel Path Contention File (VMXCHC)

The VMX Channel Path Contention File contains data that quantifies the utilization, busy conditions, and queue lengths of the physical channels connected to a VM/ESA system.

This file is created from VM/ESA Monitor Data, domain 0, record 9.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.13.1 VMXCHC 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 HOUR	CHCHPID ENDTS	YEAR	MONTH	DAY
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 5-33. VMXCHC Sort Sequence and Data Granularity

5.3.13.2 VMXCHC 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  CHCHPID  - Channel Path ID
 X.....E  HOUR     - Hour of Day
 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  CHANNEL  - Physical Channel Address
 X.....E  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  DURATION - Recording Interval Time
 X.....E  ENDTS    - End Time Stamp
 X.....E  INTERVLS - Number of Recording Intervals
 X.....E  STARTTS  - Start Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXHFSAM - High Frequency Sample Count
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier

Accumulated Data Elements

 X.....E  CHCBUSY  - Channel Path Busy

Derived Data Elements

 X.....E  CHCPCBSY - Percent CHPID Busy
```

5.3.13.3 VMXCHC Usage Considerations

The VMXCHC File contains up to one observation per CHPID, to a maximum of 256, per interval. If a system does not have 256 CHPIDs with devices attached, an option is available at the unit level to discard observations for these CHPIDs. See Section 10.1.9 for more information.

5.3.13.4 VMXCHC Retrieval Examples

Print average percent channel busy for CHPIDs 8, 9, and 10
for the period 9:00 AM to 10:00 AM, at 15-minute intervals.

```
%VMCBRK(FILE=VMXCHC01,
        START='08:00'T,END='10:00'T,PERIOD=15);
  IF 8 LE CHCHPID LE 10;
RUN;
%LET BY = SYSID DATE CHCHPID TIME;
%LET BREAK = TIME;
PROC SORT DATA=VMXCHC;
  BY &BY;
RUN;
DATA VMXCHC;
  SET VMXCHC;
  %CHCSUM(DERV=YES);
RUN;
PROC TIMEPLOT;
  PLOT CHCPCBSY='>' / AXIS=0 TO 20;
  ID TIME VMXHFSAM INTERVLS DURATION;
  BY SYSID DATE CHCHPID;
RUN;
```

5.3.14 VMX Block Paging Data File (VMXBPG)

The VMX Block Paging Data Data File (VMXBPG) provides VM/ESA
block paging information.

It is derived from data in VM/ESA Monitor domain 3, record 8.

The following sections describe the file's organization, list
the data elements maintained, and provide usage hints.

```
1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples
```

5.3.14.1 VMXBPG 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 ENDTS	YEAR	MONTH	DAY	HOUR
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 5-34. VMXBPG Sort Sequence and Data Granularity

5.3.14.2 VMXBPG 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  HOUR     - Hour of Day
 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  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  DURATION - Recording Interval Time
 X.....E  ENDTS    - End Time Stamp
 X.....E  STARTTS  - Start Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier
```

```
Accumulated Data Elements

 X.....E  BPGPMG1  - Number of Times a Block Was Migrated
 X.....E  BPGPMG10 - Number of Times a Block Was Migrated
 X.....E  BPGPMG11 - Number of Times a Block Was Migrated
 X.....E  BPGPMG12 - Number of Times a Block Was Migrated
 X.....E  BPGPMG13 - Number of Times a Block Was Migrated
 X.....E  BPGPMG14 - Number of Times a Block Was Migrated
 X.....E  BPGPMG15 - Number of Times a Block Was Migrated
 X.....E  BPGPMG16 - Number of Times a Block Was Migrated
 X.....E  BPGPMG17 - Number of Times a Block Was Migrated
 X.....E  BPGPMG18 - Number of Times a Block Was Migrated
 X.....E  BPGPMG19 - Number of Times a Block Was Migrated
 X.....E  BPGPMG2  - Number of Times a Block Was Migrated
 X.....E  BPGPMG20 - Number of Times a Block Was Migrated
 X.....E  BPGPMG3  - Number of Times a Block Was Migrated
 X.....E  BPGPMG4  - Number of Times a Block Was Migrated
 X.....E  BPGPMG5  - Number of Times a Block Was Migrated
 X.....E  BPGPMG6  - Number of Times a Block Was Migrated
 X.....E  BPGPMG7  - Number of Times a Block Was Migrated
 X.....E  BPGPMG8  - Number of Times a Block Was Migrated
 X.....E  BPGPMG9  - Number of Times a Block Was Migrated
 X.....E  BPGPRD1  - Number of Times a Block Was Read
 X.....E  BPGPRD10 - Number of Times a Block Was Read
 X.....E  BPGPRD11 - Number of Times a Block Was Read
 X.....E  BPGPRD12 - Number of Times a Block Was Read
 X.....E  BPGPRD13 - Number of Times a Block Was Read
 X.....E  BPGPRD14 - Number of Times a Block Was Read
 X.....E  BPGPRD15 - Number of Times a Block Was Read
 X.....E  BPGPRD16 - Number of Times a Block Was Read
 X.....E  BPGPRD17 - Number of Times a Block Was Read
 X.....E  BPGPRD18 - Number of Times a Block Was Read
 X.....E  BPGPRD19 - Number of Times a Block Was Read
 X.....E  BPGPRD2  - Number of Times a Block Was Read
 X.....E  BPGPRD20 - Number of Times a Block Was Read
 X.....E  BPGPRD3  - Number of Times a Block Was Read
 X.....E  BPGPRD4  - Number of Times a Block Was Read
 X.....E  BPGPRD5  - Number of Times a Block Was Read
 X.....E  BPGPRD6  - Number of Times a Block Was Read
 X.....E  BPGPRD7  - Number of Times a Block Was Read
 X.....E  BPGPRD8  - Number of Times a Block Was Read
 X.....E  BPGPRD9  - Number of Times a Block Was Read
 X.....E  BPGWRT1  - Number of Times a Block Was Written
 X.....E  BPGWRT10 - Number of Times a Block Was Written
 X.....E  BPGWRT11 - Number of Times a Block Was Written
 X.....E  BPGWRT12 - Number of Times a Block Was Written
 X.....E  BPGWRT13 - Number of Times a Block Was Written
 X.....E  BPGWRT14 - Number of Times a Block Was Written
 X.....E  BPGWRT15 - Number of Times a Block Was Written
```

```
X.....E  BPGWRT16 - Number of Times a Block Was Written
X.....E  BPGWRT17 - Number of Times a Block Was Written
X.....E  BPGWRT18 - Number of Times a Block Was Written
X.....E  BPGWRT19 - Number of Times a Block Was Written
X.....E  BPGWRT2  - Number of Times a Block Was Written
X.....E  BPGWRT20 - Number of Times a Block Was Written
X.....E  BPGWRT3  - Number of Times a Block Was Written
X.....E  BPGWRT4  - Number of Times a Block Was Written
X.....E  BPGWRT5  - Number of Times a Block Was Written
X.....E  BPGWRT6  - Number of Times a Block Was Written
X.....E  BPGWRT7  - Number of Times a Block Was Written
X.....E  BPGWRT8  - Number of Times a Block Was Written
X.....E  BPGWRT9  - Number of Times a Block Was Written
```

5.3.14.3 VMXBPG Usage Considerations

The VMXBPG File contains three groups of elements in each observation: blocked pages read, written, and migrated. The count in each element represents the frequency of occurrence for pages read, written, or migrated at each block size of 1-19. The twentieth element accounts for all occurrences of 20 and greater.

5.3.14.4 VMXBPG Retrieval Examples

Print the counts of blocked pages read by size for five minute intervals during the day.

```
DATA BLOCKS(KEEP=SYSID TIME DATE BPGPRD1-BPGPRD20);
  SET &VMXX..VMXBPG01;
  %VMCBRK(FILE=VMXBPG01,DATA=NO,PERIOD=5);
RUN;
DATA BLOCKS(KEEP=SYSID DATE TIME SIZE COUNT);
  SET BLOCKS;
  ARRAY BP(20) 8 BPGPRD1-BPGPRD20;
  DO SIZE = 1 TO 20;
    COUNT = BP(SIZE);
    OUTPUT;
  END;
RUN;
PROC CHART; BY SYSID DATE;
  HBAR SIZE /
       TYPE=SUM SUMVAR=COUNT GROUP=TIME DISCRETE;
  TITLE "Frequency of Blocks Read By Size";
RUN;
```

5.3.15 VMX Shared Storage Configuration File (VMXXSF)

The VMX Shared Storage Configuration File (VMXXSF) identifies VM/ESA shared system configurations.

It is derived from VM/ESA Monitor data, domain 3, records 3, 5, and 6.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.15.1 VMXXSF 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 HOUR XSSNAME ENDTS YEAR MONTH DAY
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 5-35. VMXXSF Sort Sequence and Data Granularity

5.3.15.2 VMXXSF 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  HOUR     - Hour of Day
 X.....E  MONTH    - Month of Year
 X.....E  SYSID    - System Identifier
 X.....E  WEEK     - Week of Year
 X.....E  XSSNAME  - Shared System or Segment Name
 X.....E  YEAR     - Year of Century
 X.....E  ZONE     - Time Zone
```

```
Common Data Elements

 X.....E  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  ENDTS    - End Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXAGE   - Observation Age
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier
 X.....E  VMXSTCHG - Status Change

Retained Data Elements

 X.....E  XSSCLASS - Spool File Class
 X.....E  XSSFCRTM - File Creation Time
 X.....E  XSSFTYPE - System Data File Type
 X.....E  XSSSPLID - Spool ID
```

5.3.15.3 VMXXSF Usage Considerations

The VMXXSF File is a spin file that contains one observation per saved system that is accessed while the monitor is active, and an observation for each time a saved system is either purged or saved. The VMXXSF File is created from activity data because no configuration data is provided for saved systems at monitor initialization. For this reason, all saved systems defined to VM/XA, VM/ESA, and z/VM may not appear in this file. To have all saved systems appear, an EXEC can be constructed that will list all saved systems using the class E QUERY NSS command, and access each saved system over the period of a monitor interval.

5.3.15.4 VMXXSF Retrieval Examples

List all saved systems in the VMXXSF File. File class and type from observations based on activity data will be changed from blanks to hyphens for readability.

```
DATA VMXXSS;
  SET &VMXX..VMXXSF01; BY SYSID XSSNAME;
  IF LAST.XSSNAME;
  /* INFORMATION FROM ACTIVITY DATA ? */
  IF VMXSTCHG = '--' AND XSSCLASS = ' ' THEN DO;
    XSSFTYPE = '   --   ';
    XSSCLASS = '-';
  END;
RUN;
PROC PRINT SPLIT='*'; BY SYSID;
  ID XSSNAME;
  VAR XSSFCRTM XSSFTYPE XSSSPLID XSSCLASS;
  LABEL XSSNAME='System Name'
        XSSFCRTM='File Creation Time'
        XSSSPLID=' Spool*File Id'
        XSSFTYPE='System Type'
        XSSCLASS='Class';
  TITLE "Saved Systems Defined";
RUN;
```

5.3.16 VMX Shared Storage Management File (VMXXSS)

The VMX Shared Storage Management File (VMXXSS) reports on VM/ESA shared system data.

It is derived from VM/ESA Monitor data, domain 3, record 3.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

```
1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples
```

5.3.16.1 VMXXSS 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 HOUR	XSSNAME ENDTS	YEAR	MONTH	DAY
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 5-36. VMXXSS Sort Sequence and Data Granularity

5.3.16.2 VMXXSS 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  HOUR     - Hour of Day
 X.....E  MONTH    - Month of Year
 X.....E  SYSID    - System Identifier
 X.....E  WEEK     - Week of Year
 X.....E  XSSNAME  - Shared System or Segment Name
 X.....E  YEAR     - Year of Century
 X.....E  ZONE     - Time Zone
```

```
Common Data Elements

 X.....E  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  DURATION - Recording Interval Time
 X.....E  ENDTS    - End Time Stamp
 X.....E  INTERVLS - Number of Recording Intervals
 X.....E  STARTTS  - Start Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier

Retained Data Elements

 X.....E  XSSFCRTM - File Creation Time
 X.....E  XSSSPLID - Spool ID

Accumulated Data Elements

 X.....E  XSSCTMIG - Pages Migrated from XSTOR to DASD
 X.....E  XSSCTPGR - Pages Read from DASD to Main Storage
 X.....E  XSSCTPGW - Pages Written from Main Storage to DASD
 X.....E  XSSCTSBK - Count of Expanded Storage Blocks
 X.....E  XSSCTXRD - Pages Read from XSTOR to Main Storage
 X.....E  XSSCTXWT - Pages Written from Main Storage to DASD
 X.....E  XSSNDSPG - Pages in DCSS/NSS with No Data Saved
 X.....E  XSSNPSLT - Paging Slots Occupied by this System
 X.....E  XSSNSHCT - Users of DCSS/NSS in Non-shared Mode
 X.....E  XSSPGLOK - Frames Locked by this System
 X.....E  XSSPTRFR - Count of Resident Pages
 X.....E  XSSPTRSH - Page Translations for Shared Pages
 X.....E  XSSSAVPG - Pages Saved in DCSS/NSS
 X.....E  XSSSHRCT - Users of DCSS/NSS in Shared Mode
 X.....E  XSSSTEAL - Page Frames Taken by From this System
```

5.3.16.3 VMXXSS Usage Considerations

The VMXXSS File contains information that can be used to determine the extent of use of saved systems. This can be used to determine how many users are accessing products that use shared segments.

5.3.16.4 VMXXSS Retrieval Examples

List all saved systems in the VMXXSS file and the average number of users accessing each system.

```
%VMCBRK(FILE=VMXXSS01,PERIOD=15);
RUN;
DATA SAVSYS(KEEP=SYSID DATE TIME
            XSSNAME INTERVLS AVGNSH AVGSHR);
  SET VMXXSS; BY SYSID DATE XSSNAME TIME;
  IF FIRST.TIME THEN DO;
    Z1=0;Z2=0;Z3=0;
  END;
  Z1+XSSSHRCT;
  Z2+XSSNSHCT;
  Z3+INTERVLS;
  IF LAST.TIME
  THEN DO;
    AVGSHR = Z1/Z3; AVGNSH = Z2/Z3;
    INTERVLS = Z3;
    OUTPUT;
  END;
RETURN;
PROC PRINT SPLIT='*';
  BY SYSID;
  ID XSSNAME;
  PAGEBY SYSID;
  LABEL AVGSHR='Avg. Users In* Shared Mode'
        AVGNSH=' Avg. Users In *Non-Shared Mode'
        DATE='Date' Time='Time'
        XSSNAME='Saved System*    Name'
        INTERVLS='Interval* Count';
   TITLE "Saved System Access Summary";
RUN;
```

5.3.17 VMX System Timer Pop File (VMXSTP)

The VMX System Timer Pop File identifies VM/ESA system load as reported by the scheduler at each scheduler reporting interval.

This file is created from VM/ESA Monitor Data, domain 2, record 8.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.17.1 VMXSTP 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 ENDTS	YEAR	MONTH	DAY	HOUR
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 5-37. VMXSTP Sort Sequence and Data Granularity

5.3.17.2 VMXSTP 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  HOUR     - Hour of Day
 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  CPUMODEL - CPU Model Identification
 X.....E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X.....E  ENDTS    - End Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 X.....E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXMOD   - CPU Processor Model Family
 X.....E  VMXMODID - CPU Model Identifier

Retained Data Elements

 X.....E  STPPGRDS - Pages Per Second Read by the System
 X.....E  STPRSCTM - System Timer Pop Reset Interval Size
 X.....E  STPSTEAL - Percent of Page Frames Stolen

Accumulated Data Elements

 X.....E  STPAVPRT - Average Paging Rate for All Users
 X.....E  STPAVPR1 - Average Paging Rate for E1 Users
 X.....E  STPAVWSS - Average Working Set Size for All Users
 X.....E  STPAVWS1 - Average Working Set Size for E1 Users
 X.....E  STPEXPN1 - Expansion Factor for all E1 Users
 X.....E  STPEXPN2 - Expansion Factor for all E2 Users
 X.....E  STPEXPN3 - Expansion Factor for all E3 Users
 X.....E  STPPGRTM - Average Time to Read a Page
 X.....E  STPSRWPG - System Resource Weight for Paging
 X.....E  STPSRWST - System Resource Weight for Storage

Minimum Data Elements

 X.....E  STPMNEX1 - Minimum Expansion Factor for E1 Users
 X.....E  STPMNEX2 - Minimum Expansion Factor for E2 Users
 X.....E  STPMNEX3 - Minimum Expansion Factor for E3 Users
 X.....E  STPMNPFS - Minimum Percent of Page Frames Stolen
 X.....E  STPMNPGR - Minimum Average Time to Read a Page
 X.....E  STPMNPGS - Minimum Pages Per Second Read by System
 X.....E  STPMNPRT - Minimum Average Page Rate
 X.....E  STPMNPR1 - Minimum Average E1 Page Rate
 X.....E  STPMNWPR - Minimum SRW for Paging
 X.....E  STPMNWSS - Minimum Average Working Set Size
 X.....E  STPMNWST - Minimum SRW for Storage
 X.....E  STPMNWS1 - Minimum Average E1 Working Set Size
```

```
Maximum Data Elements

 X.....E  STPMXEX1 - Maximum Expansion Factor for E1 Users
 X.....E  STPMXEX2 - Maximum Expansion Factor for E2 Users
 X.....E  STPMXEX3 - Maximum Expansion Factor for E3 Users
 X.....E  STPMXPFS - Maximum Percent of Page Frames Stolen
 X.....E  STPMXPGR - Maximum Average Time to Read a Page
 X.....E  STPMXPGS - Maximum Pages Per Second Read by System
 X.....E  STPMXPRT - Maximum Average Page Rate
 X.....E  STPMXPR1 - Maximum Average E1 Page Rate
 X.....E  STPMXWPR - Maximum SRW for Paging
 X.....E  STPMXWSS - Maximum Average Working Set Size
 X.....E  STPMXWST - Maximum SRW for Storage
 X.....E  STPMXWS1 - Maximum Average E1 Working Set Size
```

5.3.17.3 VMXSTP Usage Considerations

Special considerations or techniques related to using the VMXSTP File are provided below:

- The VMXSTP File provides a view of the system load as seen by the scheduler. This File can be used to provide a graphic image of the change in system load throughout the day.

- The VMXSTP File contains information produced by the scheduler based on the 'system timer pop' which is every 30 seconds. This data is the same information that is used to build the response the the INDICATE LOAD command.

- The VMXSTP File will contain only data if the SCHEDULER domain has been enabled for collection. See Section 6.3.3 for more information.

5.3.17.4 VMXSTP Retrieval Examples

Produce a plot of expansion factors for E1, E2, and E3 Users.

```
DATA EXPAN(KEEP=SYSID DATE TIME
                STPEXPN1 STPEXPN2 STPEXPN3);
  SET &VMXX..VMXSTP01;
  DATE=DATEPART(ENDTS);
  TIME=TIMEPART(ENDTS);
RETURN;
PROC TIMEPLOT;BY SYSID DATE;
  PLOT STPEXPN1='1' STPEXPN2='2' STPEXPN3='3' /
    OVERLAY AXIS=0 TO 6 ;
  ID TIME;
  FORMAT TIME TIME8. DATE DATE8.;
  TITLE "Expansion Factors Summary";
RUN;
```

5.3.18 VMX DASD Cache Activity File (VMXCDI)

The VMX DASD Cache Activity File (VMXCDI) reports on DASD cache activity for 3880-X3 and 3990 cache controllers at the monitor interval.

This file, which is created from VM/ESA Monitor data (domain 6, type 4 and domain 1, type 6), provides information useful for determining how effectively cache controllers are used.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.18.1 VMXCDI 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	VMXDVDEV	CDISDID	YEAR	MONTH
	DAY	HOUR	ENDTS		
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	VMXDVDEV	CDISDID	YEAR	MONTH
	ZONE				
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 5-38. VMXCDI Sort Sequence and Data Granularity

5.3.18.2 VMXCDI 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X..M..E  CDISDID  - Storage Director Identification
 X.....E  HOUR     - Hour of Day
 X..M..E  MONTH    - Month of Year
 X..M..E  SYSID    - System Identifier
 X..M..E  VMXDVDEV - Device Number
 X.....E  WEEK     - Week of Year
 X..M..E  YEAR     - Year of Century
 X..M..E  ZONE     - Time Zone
```

```
Common Data Elements

 X..M..E  CPUMODEL - CPU Model Identification
 X..M..E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X..M..E  DEVCLASS - Device Class
 X..M..E  DEVTYPE  - Device Type
 X..M..E  DURATION - Recording Interval Time
 X..M..E  ENDTS    - End Time Stamp
 X..M..E  INTERVLS - Number of Recording Intervals
 X..M..E  STARTTS  - Start Time Stamp
 X..M..E  VMCINTTS - VM Interval Time Stamp
 X..M..E  VMCINTVL - VM Monitor Interval Length
 X..M..E  VMLEVEL  - VM System Release Level
 X..M..E  VMXDVSID - Host Subchannel ID Number
 X..M..E  VMXMOD   - CPU Processor Model Family
 X..M..E  VMXMODID - CPU Model Identifier
 X..M..E  VOLSER   - Volume Serial Number

Retained Data Elements

 X..M..E  CDIADVST - Addressed Device Status
 X..M..E  CDICUTYP - Control Unit Type
 X..M..E  CDIDCSTA - Device Caching Status
 X..M..E  CDIDEVID - Device Identification
 X..M..E  CDIOCSTA - Overall Caching Status
 X..M..E  CDIONSTA - Overall Nonvolatile Storage Status
 X..M..E  CDISSID  - Subsystem Identifier
```

```
Accumulated Data Elements

 X..M..E  CDIBCIRQ - Bypass Cache I/O Requests
 X..M..E  CDICACBX - Bytes Transfered From Cache To Channel
 X..M..E  CDICADBX - Bytes Transfered From Cache To DASD
 X..M..E  CDICADXO - Cache to DASD Transfer Operations
 X..M..E  CDICAPAV - Available Subsystem Storage Capacity
 X..M..E  CDICAPBD - Bound Subsystem Storage Capacity
 X..M..E  CDICAPCF - Configured Subsystem Storage Capacity
 X..M..E  CDICAPOF - Offline Subsystem Storage Capacity
 X..M..E  CDICAPPN - Pinned Subsystem Storage Capacity
 X..M..E  CDICFWRH - Cache Fast Write I/O Request Hits
 X..M..E  CDICFWRQ - Cache Fast Write I/O Requests
 X..M..E  CDIDACBX - Bytes Transfered From DASD To Channel
 X..M..E  CDIDCABX - Bytes Transfered From DASD To Cache
 X..M..E  CDIDCAXO - DASD To Cache Transfer Operations
 X..M..E  CDIDCCBX - Bytes Xfr From DASD To Cache and Channel
 X..M..E  CDIDFWDO - DASD Fast Write to DASD Operations Force
 X..M..E  CDIDFWNO - DASD Fast Write Normal Write Operations
 X..M..E  CDIDFWRH - DASD Fast Write I/O Request Hits
 X..M..E  CDIDFWSO - DASD Fast Write Seq. Write Operations
 X..M..E  CDIFWSRH - DASD Fast Write Seq. I/O Request Hits
 X..M..E  CDIICLRQ - Inhibit Cache Loading Requests
 X..M..E  CDINVSCF - Configured Nonvolatile Storage Capacity
 X..M..E  CDINVSPN - Pinned Nonvolatile Storage Capacity
 X..M..E  CDISDSRQ - Sequential Detected Seq. Access Reads
 X..M..E  CDISQDCO - Sequential DASD To Cache Xfr Operations
 X..M..E  CDISRBND - Search and Read Requests To Bound Tracks
 X..M..E  CDISRCRH - Search or Read Cache Fast Write Req. Hit
 X..M..E  CDISRCRQ - Search or Read Cache Fast Write Req.
 X..M..E  CDISRNRH - Search and Read Normal Request Hits
 X..M..E  CDISRNRQ - Search and Read Normal Requests
 X..M..E  CDISRSRH - Search and Read Sequential Request Hits
 X..M..E  CDISRSRQ - Search and Read Sequential Requests
 X..M..E  CDIWBND  - Write Requests To Bound Tracks
 X..M..E  CDIWNRH  - Write Normal Request Hits
 X..M..E  CDIWNRQ  - Write Normal Requests
 X..M..E  CDIWSQRQ - Write Sequential Requests

Derived Data Elements

 X..M..E  CDIPCRBT - Percent Reads to Bound Tracks
 X..M..E  CDIPCSSB - Percent Subsystem Storage Bound
 X..M..E  CDIPCSSO - Percent Subsystem Storage Offline
 X..M..E  CDIPCWBT - Percent Writes to Bound Tracks
 X..M..E  CDIRNRAT - Search and Read Normal Hit Ratio
 X..M..E  CDIRSRAT - Search and Read Sequential Hit Ratio
 X..M..E  CDIWNRAT - Write Normal Hit Ratio
```

5.3.18.3 VMXCDI Usage Considerations

The VMXCDI File contains one observation per device per storage director in each 3880-13 or 3880-23 controller and one observation per device in each 3990-3 controller that is monitored. This provides the ability to view activity for both storage directors in the controller independently or to combine them to view total activity for a volume.

5.3.18.4 VMXCDI Retrieval Examples

Compare activity for the storage directors controlling volume VMXARS.

```
DATA CACHE;
  SET &VMXX..VMXCDI01;
  FORMAT TIME TIME8. DATE DATE7.;
  IF VOLSER = 'VMXARS';
  TIME = TIMEPART(ENDTS);
  DATE = DATEPART(ENDTS);
RUN;
%LET BY = SYSID VOLSER DATE TIME CDISDID;
PROC SORT DATA=CACHE;
  BY &BY;
RUN;
DATA CACHE(KEEP=SYSID DATE TIME CDICAPAV CDICAPOF VOLSER
               READ1 WRITE1 READ2 WRITE2);
  SET CACHE;BY &BY;
  ARRAY READ(*)  8 READ1-READ2;
  ARRAY WRITE(*)  8 WRITE1-WRITE2;
  RETAIN READ1-READ2 WRITE1-WRITE2;
  IF FIRST.CDISDID THEN I=0;
  I+1;
  READ(I)  = SUM(CDISRNRQ,CDISRSRQ);
  WRITE(I) = SUM(CDIWNRQ,CDIWSQRQ);
  IF LAST.TIME;
    CDICAPAV = CDICAPAV/1048576;
    CDICAPOF = CDICAPOF/1048576;
RUN;
PROC TIMEPLOT;
  BY SYSID VOLSER DATE;
  ID TIME CDICAPAV CDICAPOF;
  PLOT READ1='1' READ2='2' WRITE1='A' WRITE2='B' /
  OVERLAY;
  LABEL DATE='Date' TIME='Time'
    CDICAPAV='Available Storage'
    CDICAPOF='Offline Storage'
    READ1='Reads SD1'
    READ2='Reads SD2'
    WRITE1='Writes SD1'
    WRITE2='Writes SD2';
RUN;
```

5.3.19 VMX Paging Cache Activity File (VMXCDP)

The VMX Paging Cache Activity File (VMXCDP) reports on DASD cache activity for 3880-X1 controllers at the scheduler interval.

This file is created from VM/ESA Monitor Data, domain 6, record 4 and domain 1, record 6.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.19.1 VMXCDP 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	VMXDVDEV	CDPSDID	YEAR	MONTH
	DAY	HOUR	ENDTS		
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	VMXDVDEV	CDPSDID	YEAR	MONTH
	ZONE				
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 5-39. VMXCDP Sort Sequence and Data Granularity

5.3.19.2 VMXCDP 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..M..E	CDPSDID	- Storage Director Identifier
X.....E	HOUR	- Hour of Day
X..M..E	MONTH	- Month of Year
X..M..E	SYSID	- System Identifier
X..M..E	VMXDVDEV	- Device Number
X.....E	WEEK	- Week of Year
X..M..E	YEAR	- Year of Century
X..M..E	ZONE	- Time Zone

```
Common Data Elements

 X..M..E  CPUMODEL - CPU Model Identification
 X..M..E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X..M..E  DEVCLASS - Device Class
 X..M..E  DEVTYPE  - Device Type
 X..M..E  DURATION - Recording Interval Time
 X..M..E  ENDTS    - End Time Stamp
 X..M..E  INTERVLS - Number of Recording Intervals
 X..M..E  STARTTS  - Start Time Stamp
 X..M..E  VMCINTTS - VM Interval Time Stamp
 X..M..E  VMCINTVL - VM Monitor Interval Length
 X..M..E  VMLEVEL  - VM System Release Level
 X..M..E  VMXDVSID - Host Subchannel ID Number
 X..M..E  VMXMOD   - CPU Processor Model Family
 X..M..E  VMXMODID - CPU Model Identifier
 X..M..E  VOLSER   - Volume Serial Number

Retained Data Elements

 X..M..E  CDPCDSTA - Channel Data Rate Status
 X..M..E  CDPCUTYP - Control Unit Type
 X..M..E  CDPDEVID - Device Identifier
 X..M..E  CDPOSSTA - Overall Subsystem Status
 X..M..E  CDPSDSTA - Storage Director Status
```

```
Accumulated Data Elements

 X..M..E  CDPBLKDS - Block Discards
 X..M..E  CDPBLKUO - DASD Block Update Operations
 X..M..E  CDPBLKUP - Data Blocks Written From Cache To DASD
 X..M..E  CDPBLKWR - Block Writes
 X..M..E  CDPCAPAV - Available Subsystem Storage Capacity
 X..M..E  CDPCAPCF - Configured Subsystem Storage Capacity
 X..M..E  CDPCAPOF - Offline Subsystem Storage Capacity
 X..M..E  CDPCAPUN - Unavailable Subsystem Storage Capacity
 X..M..E  CDPNSPRH - Non-Sequential Paging Read Hits
 X..M..E  CDPNSPRO - Non-Sequential Paging Read Operations
 X..M..E  CDPNSQPO - Non-Sequential Paging Operations
 X..M..E  CDPPMDXC - Paging Data Transfer Command Chains
 X..M..E  CDPPMDXN - Paging Transfers Without DASD Access
 X..M..E  CDPSQPRC - Sequential Paging Read Commands
 X..M..E  CDPSQPRH - Sequential Paging Read Hits
 X..M..E  CDPSQPRO - Sequential Paging Read Operations
 X..M..E  CDPSWPIN - Swap In Operations
 X..M..E  CDPSWPRC - Swap In Read Commands
 X..M..E  CDPSWPRH - Swap In Read Hits

Derived Data Elements

 X..M..E  CDPCHRAT - Cache Hit Ratio
 X..M..E  CDPNSRAT - Nonsequential Read Hit Ratio
 X..M..E  CDPPCSSE - Percent Subsystem Storage In Error
 X..M..E  CDPPCSSO - Percent Subsystem Storage Offline
 X..M..E  CDPPCSSU - Percent Subsystem Storage Unavailable
 X..M..E  CDPPSDBU - DASD Block Updates per Second
 X..M..E  CDPPSPAG - Paging Operations per Second
 X..M..E  CDPPSSWP - Swap-Ins per Second
 X..M..E  CDPRHRAT - Total Read Hit Ratio
 X..M..E  CDPSQRAT - Sequential Read Hit Ratio
 X..M..E  CDPSWRAT - Swap In Read Hit Ratio
```

5.3.19.3 VMXCDP Usage Considerations

The VMXCDP File contains one observation device per storage director in each 3880-11 or 3880-21 controller that is monitored. This provides the ability to view activity for both storage directors in the controller independently or to combine them to view total activity for a volume.

5.3.19.4 VMXCDP Data Retrieval Examples

Report the number of page operations per second for volume PAGE01.

```
%VMCBRK(FILE=VMXCDP01,START='11:00'T,END='11:15'T,
        PERIOD=15);
  IF VOLSER = 'PAGE01';
  PAGEIN = CDPNSQPO+CDPSQPRO;
  PAGERC = CDPNSPRO+CDPSQPRC;
  PAGERH = CDPNSPRH+CDPSQPRH;
RUN;
%LET BY = SYSID VOLSER DATE TIME;
PROC SORT DATA=VMXCDP;
  BY &BY;
RUN;
PROC SUMMARY;
  BY &BY;
  VAR PAGEIN PAGERC PAGERH;
  OUTPUT OUT=SUMMARY
  SUM=PAGEIN PAGERC PAGERH;
RUN;
PROC TIMEPLOT;
  BY SYSID VOLSER DATE;
  PLOT PAGEIN='+' PAGERC='*' PAGERH='>'/OVERLAY;
  ID TIME _FREQ_;
  LABEL PAGEIN='Page Ops'
        PAGERC='Page Reads'
        PAGERH='Read Hits'
        _FREQ_='Observation Count';
  TITLE "Activity for Paging Device";
RUN;
```

5.3.20 VMX Logical Partition Activity File (VMXLPR)

The VMX Logical Partition Activity File (VMXLPR) reports on the activity of the central electronic complex when VM/ESA is operating in a logical partition.

It is derived from data in VM/ESA Monitor domain 0, records 15, 16, and 17.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.20.1 VMXLPR 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	PRSMLPNO	YEAR	MONTH	DAY	
	HOUR	ENDTS				
DAYS	N/A					
WEEKS	N/A					
MONTHS	SYSID	PRSMLPNO	YEAR	MONTH	ZONE	
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 5-40. VMXLPR Sort Sequence and Data Granularity

5.3.20.2 VMXLPR 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	HOUR	- Hour of Day
X..M..E	MONTH	- Month of Year
X..M..E	PRSMLPNO	- Logical Partition Number
X..M..E	SYSID	- System Identifier
X.....E	WEEK	- Week of Year
X..M..E	YEAR	- Year of Century
X..M..E	ZONE	- Time Zone

```
Common Data Elements

 X..M..E  CPUMODEL - CPU Model Identification
 X..M..E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X..M..E  DURATION - Recording Interval Time
 X..M..E  ENDTS    - End Time Stamp
 X..M..E  INTERVLS - Number of Recording Intervals
 X..M..E  PRSMLPNM - Logical Partition Name
 X..M..E  PRSMNOPP - Number of Physical Processors
 X..M..E  PRSMTLPS - Total Defined Logical Partitions
 X..M..E  PRSMTPP  - Total Processors in Complex
 X..M..E  STARTTS  - Start Time Stamp
 X..M..E  VMCINTTS - VM Interval Time Stamp
 X..M..E  VMCINTVL - VM Monitor Interval Length
 X..M..E  VMLEVEL  - VM System Release Level
 X..M..E  VMXMOD   - CPU Processor Model Family
 X..M..E  VMXMODID - CPU Model Identifier

Retained Data Elements

 X..M..E  LPRCPTYP - Logical Processor CPU Type
 X..M..E  LPRL2ADJ - LPAR Capability Adjustment Factor (L-2)
 X..M..E  LPRL2CAP - Level-2 Configuration Capability
 X..M..E  LPRL2NAM - LPAR Name of the Level-2 Configuration
 X..M..E  LPRL2NUM - LPAR Number of the Level-2 Configuration
 X..M..E  LPRMONAV - Intervals ELMS is Available
 X..M..E  LPRTDCTM - TOD of Data Collection
```

```
Accumulated Data Elements

 X..M..E  LPRCAPP  - Intervals with Capping
 X..M..E  LPRCOLCT - Intervals Running in Owner LPAR
 X..M..E  LPRCPUCT - Number of Logical CPUs
 X..M..E  LPRDURA  - LPAR Duration
 X..M..E  LPREMSAV - Intvls Special Engine Enhanced LPAR Avai
 X..M..E  LPREPDTM - LPAR Effective Dispatch Time
 X..M..E  LPRGTS   - Global Time Slice
 X..M..E  LPRLPAR  - Intervals Running in LPAR
 X..M..E  LPRLPDTM - LPAR Dispatch Time
 X..M..E  LPRLPMTM - LPAR Management Time
 X..M..E  LPRL2CFG - No. L-2 Logical CPUs in Configure State
 X..M..E  LPRL2CNT - Total Level-2 Logical CPUs Configured
 X..M..E  LPRL2DED - No. L-2 Logical CPUs using L-1 Ded'd CPU
 X..M..E  LPRL2RVD - No. L-2 Logical CPUs in Reserved State
 X..M..E  LPRL2SBY - No. L-2 Logical CPUs in Standby State
 X..M..E  LPRL2SHR - No. L-2 Logical CPUs using L-1 Shr'd CPU
 X..M..E  LPRMGMTF - Intervals LPAR Mgt Time Facility Availab
 X..M..E  LPRPPWGT - LPAR Processing Weight
 X..M..E  LPRSFRUA - Intervals Enhanced LPAR Stats Unavail
 X..M..E  LPRSHRMD - Intervals in Shared Mode
 X..M..E  LPRSPNAV - Intvls Enhanced LPAR Stats + Secondary U
 X..M..E  LPRTDP   - Number of Dedicated CPUs
 X..M..E  LPRTVP   - Number of Shared CPUs
 X..M..E  LPRWAITF - Intervals with Wait Completion On

Minimum Data Elements

 X..M..E  LPRMNDTM - Minimum LPAR Dispatch Time
 X..M..E  LPRMNMTM - Minimum Pct LPAR Management Overhead
 X..M..E  LPRMNSSU - Minimum Real CPU Utilization

Maximum Data Elements

 X..M..E  LPRMXDTM - Maximum LPAR Dispatch Time
 X..M..E  LPRMXMTM - Maximum Pct LPAR Management Overhead
 X..M..E  LPRMXSSU - Maximum Real CPU Utilization

Derived Data Elements

 X..M..E  LPRAVDTM - Avg LPAR Dispatch Time per Interval
 X..M..E  LPRPCESU - LPAR Effective CPU Utilization
 X..M..E  LPRPCMTM - Percent LPAR Management Overhead
 X..M..E  LPRPCSSU - LPAR Real CPU Utilization
 X..M..E  LPRPCVPU - LPAR Logical CPU Utilization
```

5.3.20.3 VMXLPR Usage Considerations

The VMXLPR File contains one observation for each logical partition defined to the central electronic complex per monitor interval. This provides the ability to view activity for the VM/ESA system using data generated externally from the system itself and to view overall activity for the other logical partitions defined to the central electronic complex.

The file provides LPAR activity data, including LPAR dispatch time and partition management time. In addition, it contains system overhead spent managing the physical configuration, which is included in a special partition named PHYSICAL.

5.3.20.4 VMXLPR Data Retrieval Examples

Plot CPU consumption of each logical partition with total central electronic complex CPU consumption as a reference line.

```
PROC TIMEPLOT DATA=&VMXX..VMXLPR01;                                  00120001
  BY PRSMLPNM HOUR;                                                  00130001
  ID ENDTS LPRTDP LPRTVP;                                            00130001
  PLOT LPRPCVPU='>' LPRPCSSU='*'                                     00131001
  / OVERLAY HILOC POS=60 AXIS=0 TO 60;                               00140001
  TITLE "CPU Consumption by Logical Partition";
RUN;                                                                 00150001
```

5.3.21 VMX Logical Processor Activity File (VMXLCP)

The VMX Logical Processor Activity File (VMXLCP) reports on the activity of individual logical CPUs in the central electronic complex when VM/ESA is operating in a logical partition.

It is derived from data in VM/ESA Monitor domain 0, records 15, 16, and 17.

The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.3.21.1 VMXLCP 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	PRSMLPNO	VMXCPUAD	YEAR	MONTH
	DAY	HOUR	ENDTS		
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	PRSMLPNO	VMXCPUAD	YEAR	MONTH
	ZONE				
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 5-41. VMXLCP Sort Sequence and Data Granularity

5.3.21.2 VMXLCP 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	HOUR	- Hour of Day
X..M..E	MONTH	- Month of Year
X..M..E	PRSMLPNO	- Logical Partition Number
X..M..E	SYSID	- System Identifier
X..M..E	VMXCPUAD	- Processor Address
X.....E	WEEK	- Week of Year
X..M..E	YEAR	- Year of Century

```
 X..M..E  ZONE     - Time Zone

Common Data Elements

 X..M..E  CPUMODEL - CPU Model Identification
 X..M..E  CPUSER   - CPU/Processor Serial Number
 X.....E  DAY      - Day of Month
 X.....E  DAYNAME  - Name of Day of Week
 X..M..E  DURATION - Recording Interval Time
 X..M..E  ENDTS    - End Time Stamp
 X..M..E  INTERVLS - Number of Recording Intervals
 X..M..E  PRSMLPNM - Logical Partition Name
 X..M..E  PRSMNOPP - Number of Physical Processors
 X..M..E  PRSMTLPS - Total Defined Logical Partitions
 X..M..E  PRSMTPP  - Total Processors in Complex
 X..M..E  STARTTS  - Start Time Stamp
 X..M..E  VMCINTTS - VM Interval Time Stamp
 X..M..E  VMCINTVL - VM Monitor Interval Length
 X..M..E  VMLEVEL  - VM System Release Level
 X..M..E  VMXMOD   - CPU Processor Model Family
 X..M..E  VMXMODID - CPU Model Identifier

Retained Data Elements

 X..M..E  LCPCPTYP - Logical Processor CPU Type
 X..M..E  LCPTDCTM - TOD of Data Collection

Accumulated Data Elements

 X..M..E  LCPCAPP  - Intervals with Capping
 X..M..E  LCPCOLCT - Intervals Running in Owner LPAR
 X..M..E  LCPCPUCT - Logical CPUs in LPAR
 X..M..E  LCPDURA  - LPAR Duration
 X..M..E  LCPEFDTM - LCPU Effective Dispatch Time
 X..M..E  LCPEPDTM - LPAR Effective Dispatch Time
 X..M..E  LCPLCDTM - LCPU Dispatch Time
 X..M..E  LCPLCMTM - LCPU Management Time
 X..M..E  LCPLPAR  - Intervals Running in LPAR
 X..M..E  LCPLPDTM - LPAR Dispatch Time
 X..M..E  LCPLPMTM - LPAR Management Time
 X..M..E  LCPPPWGT - LPAR Processing Weight
 X..M..E  LCPSHRMD - Intervals Running Non-Dedicated
 X..M..E  LCPWAITF - Intervals with Wait Completion On
```

```
Derived Data Elements

 X..M..E  LCPPCSIL - Logical CPU Dispatch Share in LPAR
 X..M..E  LCPPCSSU - Logical CPU Shared System Utilization
 X..M..E  LCPPCVPU - Logical CPU Utilization in LPAR
```

5.3.21.3 VMXLCP Usage Considerations

The VMXLCP File contains one observation per logical CPU defined for each logical partition defined to the central electronic complex per monitor interval. This provides the ability to view activity for each processor using data generated externally from the VM/ESA system itself and to view activity of each of the logical CPUs defined to the central electronic complex.

The file provides logical processor activity data which includes logical processor dispatch time and the time spent managing the partition to which logical processor pertains. In addition, it provides the physical processor time spent managing the physical configuration, which is included in a special partition called PHYSICAL.

5.3.21.4 VMXLCP Data Retrieval Examples

Chart the CPU consumption (logical CPU dispatch time) for all logical partitions in the central electronic complex.

```
%VMCBRK(FILE=VMXLCP01,PERIOD=60);                                  00120001
RUN;                                                               00150001
PROC SORT;                                                         00130001
  BY SYSID DATE TIME PRSMLPNM;                                     00130001
RUN;                                                               00150001
PROC CHART;                                                        00131001
  BY SYSID DATE;                                                   00140001
  VBAR TIME / SUMVAR=LCPLCDTM SUBGROUP=PRSMLPNM DISCRETE;
  TITLE "Cumulative CPU Consumed by All Logical CPUs";
RUN;                                                               00150001
```

5.3.22 VMX Scheduler Parameters File (VMXSCX)

```
The VMX Scheduler Parameters File contains CP scheduler
tuning command settings, such as SET SRM IABIAS, SET SRM
DSPSLICE, SET SRM LDUBUF, SET SRM STORBUF.

This file is created from VM/ESA Monitor data, domain 1,
record 16 and domain 2, record 7.
The following sections describe the file's organization, list
the data elements maintained, and provide usage hints.

   1 - File Organization
   2 - Data Elements List
   3 - Usage Considerations
   4 - Retrieval Examples
```

5.3.22.1 VMXCPM 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	CHANNEL	VMXCSSID	YEAR	MONTH
	DAY	HOUR	ENDTS		
DAYS	SYSID	CHANNEL	VMXCSSID	YEAR	MONTH
	DAY	HOUR			
WEEKS	N/A				
MONTHS	SYSID	CHANNEL	VMXCSSID	YEAR	MONTH
	ZONE				
YEARS	N/A				
TABLES	N/A				

Generation Date: Mon, Feb 28, 2011

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 5-43. VMXSCX Sort Sequence and Data Granularity

5.3.22.2 VMXSCX 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 &amp;quot;XDWMYT&amp;quot; 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 X.....E  HOUR     - Hour of Day
 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  DAY      - Day of Month
 X.....   DAYNAME  - Name of Day of Week
 X.....E  ENDTS    - End Time Stamp
 X.....E  VMLEVEL  - VM System Release Level
 X.....E  VMXAGE   - Observation Age
 X.....E  VMXCPUAF - CPU Affinity setting
 X.....E  VMXSTCHG - Status Change

Retained Data Elements

 X.....E  SCXBIASD - Interactive Bias Duration
 X.....E  SCXBIASI - Interactive Bias Intensity
 X.....E  SCXLDGCP - System Loading Users Capacity
 X.....E  SCXLDGUS - Loading User Page Reads
 X.....E  SCXL1DSP - Openings in DLIST for E1 E2 E3 Users
 X.....E  SCXL2DSP - Openings in DLIST for E2 E3 Users
 X.....E  SCXL3DSP - Openings in DLIST for E3 Users
 X.....E  SCXP1LDG - Pct Loading Capacity for Q1 Q2 Q3 Users
 X.....E  SCXP1WSS - Pct Memory Available for E1 E2 E3 Users
 X.....E  SCXP2LDG - Pct Loading Capacity for Q2 and Q3 Users
 X.....E  SCXP2WSS - Pct Memory Available for E2 and E3 Users
 X.....E  SCXP3LDG - Pct Loading Capacity for Q3 Users
 X.....E  SCXP3WSS - Pct Memory Available for E3 Users
 X.....E  SCXTRUSR - Target User of SET CPUAFFINITY Command
 X.....E  SCXTSHOT - Hotshot Time Slice Size
 X.....E  SCXTSLIC - Minor Time Slice Size
 X.....E  SCXWSSMP - Maximum Working Set Percentage
 X.....E  SCXXPCTG - Percentage XSTORE Used in Calculations
```

5.3.22.3 VMXSCX Usage Considerations

The VMXSCX File is a spin file that contains one observation per set of monitor initialization data read (CP MONITOR START command issued) and one observation for each time a SET SRM command issued. Only one cycle exists in the detail time span. This file contains observations from several runs of the DAILY job. Each cycle is identified by the element VMXAGE. VMXAGE is 1 for the current cycle, 2 for the previous cycle, etc. The VMXAGE element can be used to match a particular VMXSCX observation with a cycle of the VMXSGP file.

5.3.22.4 VMXSCX Data Retrieval Examples

```
List a history of SET SRM changes for all systems in the
unit.

  PROC PRINT DATA=&VMXX..VMXSCX01 SPLIT = '*';
    BY SYSID;
    VAR ENDTS SCXTSLIC SCXTSHOT SCXBIASI SCXBIASD SCXLDGCP
        SCXP1LDG SCXP2LDG SCXP3LDG SCXP1WSS SCXP2WSS SCXP3WSS
        ;
    LABEL ENDTS = '*       Time*-------------------'
          SCXTSLIC = '   Minor*Time Slice*----------'
          SCXTSHOT  = '  Hotshot*Time Slice*----------'
          SCXBIASI = ' IABIAS*Intensity*---------'
          SCXBIASD = ' IABIAS*Duration*--------'
          SCXLDGCP = ' Loading*Capacity*--------'
          SCXP1LDG = ' LDUBUF* Q1-Q3*--------'
          SCXP2LDG = ' LDUBUF* Q2-Q3*--------'
          SCXP3LDG = ' LDUBUF*   Q3*--------'
          SCXP1WSS = 'STORBUF* E1-E3*--------'
          SCXP2WSS = 'STORBUF* E2-E3*--------'
          SCXP3WSS = 'STORBUF*   E3*--------';
    TITLE "History of SET SRM Changes";
  RUN;
```

5.3.23 VMX Channel Path Measurement File (VMXCPM)

```
 The VM Channel Path Measurement (VMXCPM) file quantifies
 utilization data and measurement characteristic data for all
 valid channel paths in the channel subsystem.

 This file is created from z/VM Monitor Data, domain 0, record
 18 or record 20 when the Channel-Path Measurement Facility
 (CPMF) with extended measurements is installed.

The following sections describe the file's organization, list
the data elements maintained, and provide usage hints.

   1 - File Organization
   2 - Data Elements List
   3 - Usage Considerations
   4 - Retrieval Examples
```

5.3.23.1 VMXCPM 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	CHANNEL	VMXCSSID	YEAR	MONTH
	DAY	HOUR	ENDTS		
DAYS	SYSID	CHANNEL	VMXCSSID	YEAR	MONTH
	DAY	HOUR			
WEEKS	N/A				
MONTHS	SYSID	CHANNEL	VMXCSSID	YEAR	MONTH
	ZONE				
YEARS	N/A				
TABLES	N/A				

Generation Date: Mon, Feb 28, 2011

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 5-43. VMXCPM Sort Sequence and Data Granularity

5.3.23.2 VMXCPM 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.

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.M..E	CHANNEL	- Physical Channel Address
XD....E	DAY	- Day of Month
XD....E	HOUR	- Hour of Day
XD.M..E	MONTH	- Month of Year
XD.M..E	SYSID	- System Identifier
XD.M..E	VMXCSSID	- Logical Channel Subsystem Identifier
XD....E	WEEK	- Week of Year
XD.M..E	YEAR	- Year of Century
XD.M..E	ZONE	- Time Zone

```
Common Data Elements

 XD....E  DAYNAME  - Name of Day of Week
 XD.M..E  DURATION - Recording Interval Time
 XD.M..E  ENDTS    - End Time Stamp
 XD.M..E  INTERVLS - Number of Recording Intervals
 XD.M..E  STARTTS  - Start Time Stamp
 X.....E  VMCINTTS - VM Interval Time Stamp
 XD.M..E  VMCINTVL - VM Monitor Interval Length
 XD.M..E  VMLEVEL  - VM System Release Level

Retained Data Elements

 XD.M..E  CPMCMG   - VM Channel Measurement Group
 XD.M..E  CPMCMMQ  - VM Channel Model Group Qualifier
 X.....E  CPMCVMSK - VM Channel Measurements Chars Validity M
 XD.M..E  CPMDUS   - VM FICON Data Unit Size
 XD.M..E  CPMHPDUS - VM HiperSockets Data Unit Size - LPAR
 XD.M..E  CPMHPMUS - VM HiperSockets Message Unit Size - LPAR
 XD.M..E  CPMHTDUS - VM HiperSockets Data Unit Size - CPC
 XD.M..E  CPMHTMUS - VM HiperSockets Message Unit Size - CPC
 XD.M..E  CPMTYPE  - VM Channel Path Type
 X.....E  CPMUVMSK - VN Channel Utilization Validity Mask

Accumulated Data Elements

 XD.M..E  CPMCBCCT - VM Total Bus Cycles Count
 XD.M..E  CPMCBSTM - VM Channel Path Busy Time
 XD.M..E  CPMCPINT - VM Channel Path Interval Time
 XD.M..E  CPMCPMF  - VM Channel Data Enabled
 XD.M..E  CPMCWUL  - VM FICON Channel Work Unit Count - LPAR
 XD.M..E  CPMCWUT  - VM FICON Channel Work Unit Count - CPC
 XD.M..E  CPMECPMF - VM Ext Channel Data Supported
 XD.M..E  CPMHDUSL - VM HiperSockets Data Units Sent - LPAR
 XD.M..E  CPMHDUST - VM HiperSockets Data Units Sent - CPC
 XD.M..E  CPMLBTTM - VM Channel Path Busy Time - LPAR
 XD.M..E  CPMLDUR  - VM FICON Read Data Unit Count - LPAR
 XD.M..E  CPMLDUW  - VM FICON Write Data Unit Count - LPAR
 XD.M..E  CPMMSGFL - VM HiperSockets Message Failures - LPAR
 XD.M..E  CPMMSNTL - VM HiperSockets Messages Sent - LPAR
 XD.M..E  CPMMSNTT - VM HiperSockets Messages Sent - CPC
 XD.M..E  CPMRECFL - VM HiperSockets Receive Failures - LPAR
 XD.M..E  CPMRECFT - VM HiperSockets Receive Failures - CPC
 XD.M..E  CPMSHRCT - VM Channel Path Shared
 XD.M..E  CPMTBTTM - VM Channel Path Busy Time - CPC
 XD.M..E  CPMTDUR  - VM FICON Read Data Unit Count - CPC
 XD.M..E  CPMTDUW  - VM FICON Write Data Unit Count - CPC
```

```
Maximum Data Elements

 XD.M..E  CPMMXIBC - VM Max FICON Internal Bus Cycles Per Sec
 XD.M..E  CPMMXUNP - VM Max FICON Channel Work Units Per Seco
 XD.M..E  CPMMXUNR - VM Max FICON Read Data Units Per Second
 XD.M..E  CPMMXUNW - VM Max FICON Write Data Units Per Second

Derived Data Elements

 XD.M..E  CPMHPMSZ - VM HiperSockets Avg Message Size - LPAR
 XD.M..E  CPMHTMSZ - VM HiperSockets Avg Message Size - CPC
 XD.M..E  CPMPCBSY - VM Percent Channel Busy - CPC
 XD.M..E  CPMPCBUS - VM Percent Bus Busy
 XD.M..E  CPMPCPBY - VM Percent Channel Busy - LPAR
 XD.M..E  CPMPSHPW - VM Hipersockets Write MB/Sec - LPAR
 XD.M..E  CPMPSHTW - VM Hipersockets Write MB/Sec - CPC
 XD.M..E  CPMPSPMS - VM HiperSockets Message Sent Rate - LPAR
 XD.M..E  CPMPSPR  - VM FICON Read MB/Sec - LPAR
 XD.M..E  CPMPSPUB - VM HiperSockets Receive Failed Rate - LP
 XD.M..E  CPMPSPUS - VM HiperSockets Message Failed Rate - LP
 XD.M..E  CPMPSPW  - VM FICON Write MB/Sec - LPAR
 XD.M..E  CPMPSTMS - VM HiperSockets Message Sent Rate - CPC
 XD.M..E  CPMPSTR  - VM FICON Read MB/Sec - CPC
 XD.M..E  CPMPSTUB - VM HiperSockets Receive Failed Rate - CP
 XD.M..E  CPMPSTW  - VM FICON Write MB/Sec - CPC
```

5.3.23.3 VMXCPM Usage Considerations

The VMXCPM File contains from one observation per valid channel up to a maximum of 256 per interval.

The data represented in the VMXCPM represents channel utilization that is attributable to the logical partition in which VM is running as well as data for the whole CPC.

5.3.23.4 VMXCPM Retrieval Examples

```
List of all active FICON channel paths during the monitor
interval by system.

DATA FICON ;
 SET &VMXX..VMXCPM01 ;
 IF CPMCMG='FICON' ;
  PROC PRINT ;
    BY SYSID;
     VAR ENDTS CPMTYPE CPMPCPBY  CPMPCBSY CPMPCBUS
         CPMPSPR CPMPSTR CPMPSPW CPMPSTW ;

   LABEL CHANNEL='ID'
         CPMTYPE='*Type'
         CPMPCPBY='Util*Part'
         CPMPCBSY='Util*Total'
         CPMPCBUS='Util*Bus'
         CPMPSPR='Read*MB/Sec*Part'
         CPMPSTR='Read*MB/Sec*Total'
         CPMPSPW='Write*MB/Sec*Part'
         CPMPSTW='Write*MB/Sec*Total';
    TITLE "FICON Activity by System" ;
  RUN;
```

5.3.24 VMX Shared Address Space File (VMXASC)

```
The VMX Shared Address Space File identifies page activity
for all address spaces running in a z/VM environment within
the measurement interval.  Resources identified include DASD
page activity, main storage page activity, stolen pages, and
locked pages.

This file is derived from the z/VM Domain 3 Record 14 monitor
record created by IBM's MONWRITE utility and Velocity
Software's zWRITE application.

The following sections describe the file's organization, list
the data elements maintained, and provide usage hints.

  1 - File Organization
  2 - Data Elements List
  3 - Usage Considerations
  4 - Retrieval Examples
```

5.3.24.1 - VMXASC 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 USER ASCNAME YEAR MONTH DAY HOUR ENDTS
DAYS	SYSID USER ASCNAME YEAR MONTH DAY HOUR
WEEKS	N/A
MONTHS	SYSID USER ASCNAME YEAR MONTH ZONE
YEARS	N/A
TABLES	N/A

Generation Date: Mon, May 6, 2013

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 5-44. VMXASC Sort Sequence and Data Granularity

5.4.24.2 VMXASC 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, May 6, 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.M..E	ASCNAME	- Address Space Name
XD....E	HOUR	- Hour of Day
XD.M..E	MONTH	- Month of Year
XD.M..E	SYSID	- System Identifier

```
 XD.M..E  USER     - User Identification
 XD....E  WEEK     - Week of Year
 XD.M..E  YEAR     - Year of Century
 XD.M..E  ZONE     - Time Zone

Common Data Elements

 XD.M..   CPUSER   - CPU/Processor Serial Number
 XD....E  DAY      - Day of Month
 XD.M..E  DURATION - Recording Interval Time
 XD.M..E  ENDTS    - End Time Stamp
 XD.M..E  INTERVLS - Number of Recording Intervals
 XD.M..E  STARTTS  - Start Time Stamp
 X.....E  VMCGMTTS - GMT TOD Time Stamp
 XD.M..E  VMCINTTS - VM Interval Time Stamp
 XD.M..E  VMCINTVL - VM Monitor Interval Length
 X.....E  VMCZONDF - TOD Clock Time Zone Differential
 XD.M..E  VMLEVEL  - VM System Release Level
 XD.M..E  VMXMOD   - CPU Processor Model Family
 XD.M..E  VMXMODID - CPU Model Identifier

Retained Data Elements

 XD.M..E  ASCCTXBK - Count of XSTOR Blocks Occupied
 XD.M..E  ASCDEFSZ - Defined Address Space Size
 XD.M..E  ASCNPSLT - Paging Slots Occupied
 XD.M..E  ASCPGLOK - Pages Locked <2G
 XD.M..E  ASCSSIZE - Address Space Size
 X.....E  ASCSTATE - Address Space Attributes
 XD.M..E  ASCTRSAF - Resident Pages <2G
 XD.M..E  ASCUSRPI - Users Permitted to Use This Address Spac
 X.....E  ASCVDNUM - VDISK Address
 X.....E  ASCVDUSR - VDISK User

Accumulated Data Elements

 XD.M..E  ASCCPLKA - Pages Locked >2G
 XD.M..E  ASCCTMIG - Pages Migrated to DASD from XSTOR
 XD.M..E  ASCCTPGR - Pages Read from DASD
 XD.M..E  ASCCTPGW - Pages Written to DASD
 XD.M..E  ASCCTPRG - Resident Pages >2G
 XD.M..E  ASCCTXRD - Pages Read from Expanded Storage
 XD.M..E  ASCCTXWT - Pages Written to Expanded Storage
 XD.M..E  ASCHLLC  - Pages Locked in Host Logical Storage
 XD.M..E  ASCHLRC  - Pages Resident in Host Logical Storage
 XD.M..E  ASCMVB2G - Times Page Moved from Frame >=2GB to <2G
 XD.M..E  ASCSTEAL - Pages Stolen from This Address Space
 XD.M..E  ASCVDISK - Count of Address Spaces Using VDISK
```

```
Derived Data Elements

 XD.M..E  ASCPGRRT - Pages Read from DASD per Second
 XD.M..E  ASCPGRXS - Pages Read from XSTOR per Second
 XD.M..E  ASCPGSRT - Pages Stolen per Second
 XD.M..E  ASCPGWRT - Pages Written to DASD per Second
 XD.M..E  ASCPGWXS - Pages Written to XSTOR per Second
 XD.M..E  ASCPSMGD - Pgs Migrated from DASD to XSTOR per Sec
```

5.4.24.3 VMXASC Usage Considerations

```
The VMX Shared Address Space File has one observation per
address space executing within the measurement interval.  It
includes information about pages written and read from DASD
as well as expanded storage. Additional information on
resident pages, pages stolen from each address space, and
locked pages is also provided.  The VMXASC is derived from
the z/VM Domain 3 Record 14 monitor record created by IBM's
MONWRITE utility and Velocity Software's zWRITE application.
```

5.4.24.4 VMXASC Retrieval Examples

```
VMXASC EXAMPLE

 The following example lists DASD and main storage page
 activity for address spaces executing on a z/VM system
 between the hours of 10:05 11:59 am.

DATA PROCESS;
 SET &VMXX..VMXASC01;
  IF HOUR=10 OR HOUR=11;
 PROC SORT ;
   BY  SYSID YEAR MONTH DAY HOUR USER ASCNAME;
 RUN;
 PROC PRINT  NOOBS;
   BY SYSID MONTH DAY HOUR ;
   VAR USER ASCNAME ASCCTPGR ASCCTPGW ASCCTXRD ASCCTXWT
       ASCCTMIG;
   TITLE1 "DASD and Main Storage Page Activity in Hour 10
           and 11";
 RUN;
```

5.4 LNX Information Area Files

This section identifies each file in the LNX 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 file maintained in the VMX Information Area is:

1 - Linux Process Activity File (LNXLXP)

5.4.1 Linux Process Activity File (LNXLXP)

The Linux Process Activity File identifies the resources consumed by each process that executed within a Linux guest within the measurement interval. Resources identified include CPU utilization, and paging.

This file is created from Velocity Software's zTCP Linux Process CPU Domain C monitor record (Record 9).
The following sections describe the file's organization, list the data elements maintained, and provide usage hints.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.4.1.1 LNXLXP 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	USER	VMCACT1	VMCACT2	VMCACT3
	LNXPNAME	YEAR	MONTH	DAY	HOUR
	ENDTS				
DAYS	SYSID	USER	VMCACT1	VMCACT2	VMCACT3
	LNXPNAME	YEAR	MONTH	DAY	HOUR
WEEKS	N/A				
MONTHS	SYSID	USER	VMCACT1	VMCACT2	VMCACT3
	LNXPNAME	YEAR	MONTH	ZONE	
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 5-45. LNXLXP Sort Sequence and Data Granularity

5.4.1.2 LNXLXP 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-    Data      Data Element
  Span *  Element    Description (LABEL)
 -------  --------   ----------------------------------------


Sequence/Summary Data Elements

 XD....E  DAY      - Day of Month
 XD....E  HOUR     - Hour of Day
 XD.M..E  LNXPNAME - Linux Process Name
 XD.M..E  MONTH    - Month of Year
 XD.M..E  SYSID    - System Identifier
 XD.M..E  USER     - User Identification
 XD.M..E  VMCACT1  - DIVISION
 XD.M..E  VMCACT2  - REGION
 XD.M..E  VMCACT3  - PROJECT
 XD....E  WEEK     - Week of Year
 XD.M..E  YEAR     - Year of Century
 XD.M..E  ZONE     - Time Zone

Common Data Elements

 XD....   DAYNAME  - Name of Day of Week
 XD.M..   DURATION - Recording Interval Time
 XD.M..E  ENDTS    - End Time Stamp
 XD.M..E  INTERVLS - Number of Recording Intervals
 XD.M..E  LNXAPLNM - Linux Application Name
 XD.M..E  LNXLGID  - Linux Logon Group Identifier
 XD.M..E  LNXLGPNM - Linux Logon Group Name
 XD.M..E  LNXLPSLV - ESALPS Release Level
 XD.M..E  LNXLUID  - Linux Logon User Identifier
 XD.M..E  LNXLUSNM - Linux Logon User Name
 XD.M..E  LNXMONID - Linux Monitor Identifier
 XD.M..E  LNXPRDID - Linux Product Identifier
 XD.M..E  LNXSTATS - Linux Service Machine Status
 XD.M..E  LNXSUPLV - ESALPS CA MICS Release Level
 XD.M..E  LNXTODTM - Linux TOD Clock Time
 XD.M..   STARTTS  - Start Time Stamp
 XD.M..E  VMCINTTS - VM Interval Time Stamp
 XD.M..E  VMCINTVL - VM Monitor Interval Length
```

```
Retained Data Elements

 X.....E  LXPMASK  - Linux Process Run Status Mask
 XD.M..E  LXPPARMS - Linux Process Parms
 XD.M..E  LXPPATH  - Linux Process Path Name
 X.....E  LXPPFLAG - Linux Process Execution Status Flag
 X.....E  LXPPID   - Linux Process Identifier
 X.....E  LXPPPID  - Linux Parent Process Identifier
 X.....E  LXPPRDUR - Linux Process Duration
 X.....E  LXPPRETS - Linux Process End Time Stamp
 XD.M..E  LXPPRGRP - Linux Process Group
 X.....E  LXPPROST - Linux Process Status
 XD.M..E  LXPPRRSS - Linux Process Resident Set Size
 X.....E  LXPPRSTS - Linux Process Start Time Stamp

Accumulated Data Elements

 XD.M..E  LXPCHSTM - Linux Child Process System CPU Time
 XD.M..E  LXPCHTTM - Linux Child Process Total CPU Time
 XD.M..E  LXPCHVTM - Linux Child Process User CPU Time
 XD.M..E  LXPCMIFT - Linux Child Process Minor Page Faults
 XD.M..E  LXPCMJFT - Linux Child Process Major Page Faults
 XD.M..E  LXPMAJFT - Linux Major Page Faults
 XD.M..E  LXPMINFT - Linux Minor Page Faults
 XD.M..E  LXPSTIME - Linux Process System CPU Time
 XD.M..E  LXPTTIME - Linux Process Total CPU Time
 XD.M..E  LXPUSRSZ - Linux User Virtual Memory Size
 XD.M..E  LXPVTIME - Linux Process User CPU Time

Minimum Data Elements

 XD.M..E  LXPMNJPF - Minimum Linux Major Page Faults
 XD.M..E  LXPMNNPF - Minimum Linux Minor Page Faults

Maximum Data Elements

 XD.M..E  LXPMXJPF - Maximum Linux Major Page Faults
 XD.M..E  LXPMXNPF - Maximum Linux Minor Page Faults

Derived Data Elements

 XD.M..E  LXPAVRSS - Linux Avg Resident Set Size
 XD.M..E  LXPPCTRS - Percent Resident to User Memory Size
 XD.M..E  LXPSMRAT - System Memory to Total Memory Ratio
 XD.M..E  LXPUMRAT - User Memory to Total Memory Ratio
```

5.4.1.3 LNXLXP Usage Considerations

The Linux Process Activity file has one observation per process that executed within the measurement interval. It includes information about CPU, and memory utilization, and is derived from the Domain C Record 9 monitor records created by Velocity Software's zVPS application.

5.4.1.4 LNXLXP Retrieval Examples

```
LNXLXP EXAMPLE

 The following example lists resources consumed by processes
 running within a Linux guest on z/VM system that ended
 between 10:10 and 10:59 am.  Processes are grouped by the
 users who executed these processes on the report.

DATA PROCESS;
 SET DETAIL.LNXLXP01;
  IF  USER='LINUX1';
  IF HOUR=10;
  IF LXPTTIME GT 0;
 PROC SORT ;
   BY  USER LNXLUSNM LNXPNAME ENDTS;
 RUN;
 PROC PRINT  NOOBS;
   BY LNXLUSNM;
   VAR LNXPNAME LXPVTIME LXPSTIME LXPTTIME;
   TITLE1"Processes that Executed in HOUR 10';
   TITLE2"By User';
 RUN;
```

Chapter 6: DATA SOURCES

The CA MICS Analyzer Option for VM/CMS processes CP account data and monitor data from VM systems.

This section contains the following topics:

6.1 Data Collector Options and Operation (see page 417)
6.2 Processing Considerations (see page 425)
6.3 Record Descriptions (see page 432)
6.4 Operating System Details (see page 447)

6.1 Data Collector Options and Operation

Data collector options are described in the following sections:

1 - CP Account Data Options
2 - VM Monitor Data Options
3 - Velocity Monitor Data Options
4 - CA VM:Account Operation

6.1.1 VM Account Data Options

In VM, the user ID for the accounting virtual machine is defined as part of the SYSTEM_USERIDS statement (ACCOUNT1 or ACCOUNT2) in the system configuration file or in the SYSACNT macro in the HCPSYS ASSEMBLE file. The accounting records are created and kept in storage by the CP accounting system service; and sent to the accounting virtual machine's disk when requested. The accounting virtual machine uses the IUCV facility to connect to the CP accounting system service and transfers the record from storage to disk. There can be more than one accounting virtual machine defined in VM/ESA Release 2.0 or later systems.

The JOURNALING statement in the system configuration file, or the SYSJRL macro in the HCPSYS ASSEMBLE file determines the following:

- whether to include the journaling facility,
- whether to enable the system being initialized to set and query the journaling facility,
- and what to do if someone tries to log on to the system or link to a disk without a valid password.

They determine whether or not Account journal records (types 04, 05, and 06) will be generated. For journal records to be created, either specify JOURNAL=YES on the SYSJRL macro in the HCPSYS ASSEMBLE file or FACILITY ON on the JOURNALING statement in the system configuration file.

Example: (in CONFIG SYS)

```
System_Userids,
   Account DISKACNT AUTOLOG,
   ...
Journaling,
   Facility      on,
   Set_and_Query on,
   Logon,
     Account after 3 attempts,
     Lockout after 9 attempts for 10,
   Link,
     Account after 5 attempts,
     Message after 4 attempts,
     Disable after 6 attempts
```

Account data is queued for the accounting virtual machine's

retrieval even if the virtual machine is not logged on and CP is started with the COLD option. The RECORDING ACCOUNT command can be used to start or stop the creation of account records.

By default, CP lets 20 account records accumulate in storage before it notifies the accounting virtual machine. The limit can be changed by issuing the command:

```
RECORDING ACCOUNT LIMIT xx
```

where xx is the new limit.

Once the account records have been retrieved from storage and placed on the accounting virtual machine's A-disk, there are two VM Data Transfer Programs from which to choose:

1. The accounting virtual machine can be set up to signal the VM Data Transfer Program, so VMT can pick up the data from the A-disk.

2. It may punch the account records to the virtual reader of the VMT.

For information on setting up VMT for account data transfer, see section 7.1 Transfer Parameters in the VM Data Transfer Program Guide.

6.1.2 VM Monitor Data Options

VM monitor files are written into virtual storage by CP and retrieved by an application program that resides in a virtual machine. The virtual storage is defined by a shared segment that is accessed by CP and the application, simultaneously. The application provided by IBM is the CMS program MONWRITE. The MONWRITE program will access the shared segment and write the data either to disk or tape. MONWRITE output is in a special format that contains control information that is a direct image of the data in the DCSS. The MONWRITE output file must be processed by the VM Data Transfer Program to extract the data from the control information and create standard variable length records for processing by the CA MICS Analyzer Option for VM/CMS.

Use the CP MONITOR command to set the options for Monitor data collection.

Suggested parameters:

1. Enable sample domains STORAGE, PROCESSOR, USER, and I/O. Domains SYSTEM and MONITOR are automatically enabled. In the I/O Domain, enable only CPVOL devices for data collection unless some other volumes also require tracking. This will minimize data volume. It is not recommended that sample data collection restrictions be implemented for the USER domain as this will affect active users and storage utilization measurements in the VMXSGP file.

2. Enable event domains STORAGE, I/O, and PROCESSOR. This provides configuration change data at very low overhead. The amount of data is minimal. If you want to track interactions (IUCV and VMCF activity) for specific users, enable the USER domain for these users only. Due to the potential data volume, it is not recommended that event collection be enabled for all users.

   Enable the SCHEDULER domain to collect system timer pop data and scheduler parameter change data. To do this, define or determine a userid that is never logged on or that is logged on infrequently, and enable SCHEDULER data collection for that user. The userid of a virtual machine is mandatory for SCHEDULER domain collection.

3. If data is produced that your site does not need to process, the data can be dropped in the VM Data Transfer Program, in _USRSEL in the DAILY update, or in the _USRSfff macro in the DAILY update. Care should be taken when selecting which monitor records to dropped. Certain Domain 1 configuration records should not be dropped. See Section 6.2.5 for a list of configuration records required by the CA MICS Analyzer Option for VM/CMS. For information on coding these macros, see chapter 10.

4. Set the monitoring interval to one minute and the high frequency sampling interval to one second. If the data volume is so high that problems occur in the DAILY update due to DASD limitations, the monitoring interval can be lengthened.

   Note: Lengthening the monitoring interval is not recommended because data distortion occurs. Data will be missing from domain 4 (USER) data and consequently from the VMX User Activity Sample Data File (VMXVXU). A number of elements in the VMX System Global Data File (VMXSGP) are also derived from domain 4 data. These elements, and any computed elements based on them, will be affected by lengthening the interval. If the monitoring interval is lengthened, it is not necessary to change the high frequency sampling interval. The high frequency sampling interval does not affect data volume.

   Examples:

   ```
   CP MON EVENT  EN I/O ALL
   CP MON EVENT  EN PROCESSOR
   CP MON EVENT  EN STORAGE
   CP MON EVENT  EN SCHEDULER USERID $ALLOC$
   CP MON SAMPLE EN I/O CPVOL
   CP MON SAMPLE EN I/O VOL CMS001 CMS002
   CP MON SAMPLE EN USER ALL
   CP MON SAMPLE EN PROCESSOR
   CP MON SAMPLE EN STORAGE
   CP MON INTERVAL 1 MINUTE
   CP MON RATE     1 SECOND
   MONWRITE MONDCSS *MONITOR DISK <file id>
   #CP MON START
   ```

   Notes:

   The MONWRITE command MUST be started before the MONITOR START command can be issued. The name of the file created by MONWRITE can be chosen for your site.

   Each monitor file processed by the VM and CMS Analyzer MUST begin with configuration data. The configuration area time limit must be examined to ensure that MONWRITE is able to process configuration records within the time limit. In order to cut a monitor file, MONWRITE must be stopped by a CP EXTERNAL command or MONWSTOP immediate command. However, monitoring is stopped only by the MONITOR STOP command.

The MONITOR domain (1) and the SYSTEM domain (0) cannot be disabled from sample data collection. Domain 1 contains the configuration data that is required by the CA MICS Analyzer Option for VM/CMS.

Examples:

Monitoring is stopped when MONWRITE terminates:

```
#CP EXTERNAL     (or MONWSTOP)
            -->  terminate MONWRITE and monitor
MONWRITE MONDCSS *MONITOR DISK <new file id>
            -->  start MONWRITE again
#CP MON START
            -->  start monitor
```

In shared mode, monitoring does not stop unless a MONITOR STOP command is issued. Thus, there is no need to issue another MONITOR START command for continuing monitoring:

```
#CP EXTERNAL     (or MONWSTOP)
            -->  terminate MONWRITE
MONWRITE MONDCSS *MONITOR DISK <new file id>
            -->  start MONWRITE again
```

5. The CONFIG size (CP MONITOR SAMPLE CONFIG SIZE and (CP MONITOR EVENT CONFIG SIZE) must be large enough to store configuration information for all hardware, devices, and virtual machines connected to the VM system. If CONFIG SIZE is not large enough, configuration records could be dropped during monitor data collection. We recommend that the CONFIG size be 30% larger than the space required to hold all configuration records.

   See the IBM z/VM documentation for more information on CONFIG SIZE.

6.1.3 Velocity Monitor Data Options

Velocity monitor data is placed in the MONDCSS saved segment after it is prepared by CP. The Velocity Software application program, ESAWRITE, which resides on another virtual machine, retrieves the monitor data from the saved segment after it receives an IUCV interrupt from the monitor. The IUCV interrupt tells ESAWRITE the type and the location of the data. After ESAWRITE has accessed the data and processed it, ESAWRITE issues an IUCV reply to tell CP it is done with processing the data. ESAWRITE will write the data to either disk or tape. ESAWRITE will reconnect to *MONITOR every hour so a current set of configuration data can be extracted.

The output file from ESAWRITE is written as VB 1248 record length records, which is a format accepted by the CA MICS Analyzer Option for VM/CMS. The VM File Transfer Program (FTP) can be used to transmit the ESAWRITE output to z/OS for processing by the CA MICS Analyzer Option for VM/CMS.

The CA MICS Analyzer Option for VM/CMS requires certain Domain 1 configuration records to process to maintain database file integrity and report any exceptions if they occur. If a subset of monitor records is being extracted from the ESAWRITE output monitor file, review Section 6.2.6 to ensure you are selecting the required Domain 1 configuration records.

To set the options for monitor data collection, refer to Velocity Software documentation.

6.1.4 CA VM:Account Operation

CA VM:Account is a resource accounting, reporting, and capacity management system for the VM system. CA VM:Account processes the standard CP accounting records (80-byte records) and supplements the accounting records by adding additional information, thus creating 240-byte records.

CA VM:Account runs an EXEC that receives the CP raw accounting records by the userid that is defined in the ACCOUNT1 or the ACCOUNT2 parameter of the SYSTEM_USERIDS statement in the SYSTEM CONFIG file for the VM system. The EXEC blocks the data into groups of 82 records, and sends the blocks to the CA VM:Account service virtual machine for processing. This EXEC passes accounting records between CP and CA VM:Account.

OVERVIEW

CA VM:Account receives the raw CP 80 byte accounting record from the retriever machine. The retriever machine is a service machine that collects CP accounting records from CP and passes them in spool files to CA VM:Account.

When the raw records are received, a copy of the records is created called the RAWDATA $$yymmdd. The RAWDATA file contains all the data received by CA VM:Account that were not rejected due to an invalid userid, or a rate not specified for a particular project, package, or customer.

The RAWDATA records are processed by calculating charges, based on rates that have been set in rate tables and master description files. After charges have been calculated, the RAWDATA records are now known as costed records and are written to the Current Period-to-End file known as PERIODTE CURRENT.

The PERIODTE CURRENT file contains the costed records from the beginning of the accounting period:

- The first 80 bytes of a PERIODTE CURRENT record contain the same information as the RAWDATA record it came from, although some fields may be in a different order or in a different format.

- The last 160 bytes of a PERIODTE CURRENT record contain customer, package, project, and charge information.

6.2 Processing Considerations

There are several considerations for processing VM Monitor and Account data. Some of these considerations are discussed in the following sections:

1 - Supplying Data to CA MICS
2 - Data Volume - VM Monitor
3 - Dropping Input Records
4 - Monitor Suspend Considerations and Handling
5 - Consolidated Monitor Input
6 - Configuration Record Requirements

6.2.1 Supplying Data to CA MICS

Because VM and MVS are separate operating systems and CA MICS is run in MVS, the VM data must be moved from VM to MVS. This task can be accomplished by using the VM Data Transfer Program. When VM Monitor data is being processed by the VM and CMS Analyzer the VM Data Transfer Program must be used to preprocess the data. For more information, see the VM Data Transfer Program Guide.

6.2.2 Data Volume - VM Monitor

The VM Monitor is a high-volume data source. While the number of records is dependent on system size, the absolute volume is determined by the Monitor domains that are enabled. Sample data (domains 0, 3, 4, 5, and 6) is produced once per interval (usually once per minute). Event data is produced when a discrete event occurs.

Domain 0 and domain 3 records that represent system-wide measurements are combined into a single observation in the CA MICS Database. For a 24-hour period with a one-minute interval, this results in 1440 observations. The domain 0 and domain 3 records that represent CPU specific measurements and the domain 5 records are combined into a single observation per CPU in the CA MICS Database. For a 24-hour period with a one-minute interval, this results in 1440 observations per CPU.

Domain 3 records that represent shared system information are kept in the CA MICS Database at the rate of one per saved system per monitor interval. Domain 3 records that provide CP-owned volume data are kept at the rate of one observation per CP volume per interval. Block paging data is kept at the rate of one observation per interval. Channel path contention data is kept at the rate of one observation per CHPID per interval, which can be as much as 256 observations per interval.

Domain 4 sample records are produced at the rate of two per logged-on virtual machine per interval. If an average of 100 users log on over a 24-hour period, this results in 288,000 observations in the CA MICS Database. This class of records builds the VMX User Activity Sample Data File (VMXVXU) and VMX User Transaction Sample Data File (VMXVXT), which are the largest interval files in the VMX Information Area.

Domain 6 records are produced at the rate of one per subchannel per interval, and an additional one per cached DASD per interval. The number of observations depends on how many devices are monitored in the I/O domain.

Event records (domains 2, 4, and 7), when enabled, produce two to four times as much volume as interval records for the same period. This data is normally collected for short periods when detailed system analysis is required. Other event records, which report configuration changes, produce very small amounts of data.

6.2.3 Dropping Input Records

If your VM data contains records that produce CA MICS files that are not used by your site, these records can be omitted from processing in the USRSEL exit. Dropping these records in USRSEL, rather than in the file exit, reduces the CPU requirements in the DAY080 step of the CA MICS DAILY update. For information on using the USRSEL exit, see section 10.2.1.

Alternatively, records can be dropped using the VM Data Transfer Program record selection mechanism. This helps to reduce data transfer time in addition to saving DAY080 processing time.

When coding USRSEL, use care when selecting which monitor records to drop. The CA MICS Analyzer Option for VM/CMS requires certain Domain 1 configuration records to maintain data integrity. See Section 6.2.6 for a list of configuration records and other monitor domains that are dependent on domain 1 monitor records. Additional information on which database files are populated by these records is also provided.

6.2.4 Monitor Suspend Considerations and Handling

The VM Monitor may suspend when the event portion of the monitor saved segment is full, or the IUCV path to the receiving virtual machine running the application that retrieves the data from the save segment is quiesced. If suspend occurs, the monitor input data could have incomplete or missing sample configuration records required by the DAY080 processing step. Your monitor input data can contain domain 1 record 12 Event Start of Record Suspend records and domain 1 record 3 Suspension records at the beginning of your monitor input file. A usual monitor input file should start with a domain 1 record 1 Event Profile.

The monitor suspend records domain 1, records 12 and 3 can be present anywhere in the input data. There can be periods (intervals or hours) of suspend records followed by other domain records (that is, domains 2 - 8 and 10, if running Velocity Software zVPS, domain C), then periods of configuration records followed by other domain records, followed by more suspend records.

Monitor suspend records will be detected by the CA MICS Analyzer Option for VM/CMS by displaying a MICSLOG warning message VMC00314W in the DAY080 MICSLOG. In addition, the DAY080 processing step will most likely abend with a "VMC00067E Configuration Record for Processor nn is Missing" because the domain 1 record 5 is missing in the input data.

Note: For more information about Messages, see Appendix A

To avoid interruption to the daily processing of the CA MICS Analyzer Option for VM/CMS, user exit USRSEL can be coded to drop monitor suspend records and skip to the first set of domain 1, record 1, 4, 5, and 9 monitor records in the input data.

Edit prefix.MICS.USER.SOURCE(#BASEXIT) and enter the following code after the following statement, %INCLUDE SOURCE(#BASEXIT);

```
  MACRO _USRSEL

   IF ROUTINE = 'DYVMCFMT' THEN DO;
     IF MRHDRDM EQ 1 THEN DO;
      IF MRHDRRC EQ 12 THEN SUSPEND = 1;
      IF MRHDRRC EQ 1 THEN SUSPEND = 0;
     END;
      IF SUSPEND EQ 1 THEN DO;
```

```
        SKIP_REC = 1;

    /* Optional Display info in the SASLOG to track
        back to problem system   */

        IF DDREC = 1 THEN PUT '**PROBLEM SYSTEM DATASET
            NAME: ' DSNAME=;
      END;
   END;

  %
```

Note: This code sets a flag called SUSPEND. We recommend that you use SUSPEND in your coding. Flag SUSPEND is provided as a retained variable that determines when variable SKIP_REC should be set to drop records from your input data and skip to the next set of domain 1 records beginning with record 1.

After the DAY080 processing step completes, be sure to remove the code added to prefix.MICS.USER.SOURCE(#BASEXIT).

6.2.5 Consolidated Monitor Input

If your site has more than one VM system providing monitor data for CA MICS, you have the option of using separate DD names for each system, or a single DD name for all systems. If you are using tape to supply data to the VM and CMS Analyzer, using consolidated input can reduce the number of tape drives required by the DAY080 step of the daily update.

Refer to Section 7.2.1 Unit Level Parameters, for more information.

6.2.6 Configuration Record Requirements

The CA MICS Analyzer Option for VM/CMS requires Domain 1 configuration records to maintain data integrity and provide static information for the VM system, such as device and processor configuration, base measurement data (memory, paging, monitor, and user). Certain monitor records are used to report any exceptions that may occur on the VM system or the monitor, such as monitor suspend events, changes in monitoring, and CPU capability changes.

If the source of your monitor data is from Velocity Software, you need to ensure the configuration records required for CA MICS are contained in the extract file from ESAWRITE to CA MICS. Due to high data volume, your VM systems programmer may decide not to collect device configuration and I/O activity information. If that is the case, the Domain 6 and Domain 1, Record 6 configuration records are not collected.

The VM Transfer Program allows monitor records to be selected for transmission to z/OS. You must also take into consideration the Domain 1 configuration records that are selected to be sent.

The list below should be used to determine the Domain 1 configuration records required by the CA MICS Analyzer Option for VM/CMS. Also note that certain non Domain 1 records are dependent on configuration records to provide complete VM system statistics.

The table below can also be used to decide which VM and CMS database files can be deactivated or activated when configuration records are dropped.

The Required column has three options:

YES - Should not be dropped from the input data

NO - Can be dropped from the input data

COND - Should be dropped if related domain records are dropped

Domain/ Record	Description	Required YES/NO/COND	Database File/s
01/01	Event Profile	YES	VMXSCF
01/02	Event Alteration	YES	VMCVXC
01/03	Suspend End	YES	VMCVXC
01/04	System Configuration	YES	VMXSCF
01/05	Processor COnfiguration	YES	VMXPCF
01/06	Device Configuration	COND If Domain 6 records extracted	VMXDCF VMXXDV
01/07	Memory Configuration	YES	VMXSCF
01/08	Paging Configuration	NO	VMXACF
01/09	Sample Profile	YES	VMXSCF
01/10	Monitor Alteration	YES	VMCVXC
01/11	Interval End	YES	VMXSCF
01/12	Start of Suspend	YES	VMCVXC
01/13	End of Frame	NO	None
01/14	Domain Detail	NO	None
01/15	User Configuration	COND If Domain 4 records active	VMXVXF VMXVXU VMX_VU VMXVXT VMX_VT
01/16	Scheduler Settings	NO	VMXSCX
01/17	Expanded Storage	YES	VMXSCF
01/18	CPU Capability Change	YES	VMCVXC
01/19	QDIO Device Configuration	NO	None
01/20	HyperPAV Pool Definition	NO	None

6.3 Record Descriptions

The CA MICS Analyzer Option for VM/CMS processes data from CP account and VM Monitor. These data sources are largely independent. Only in the cases of VM Monitor domain 4 and CP account type 01 data is there an overlap, in which both sources use the same fields for reporting. The information provided by each record type, class, or domain is discussed in the following sections.

1 - CP Account Record Descriptions
2 - VM Monitor Record Descriptions

6.3.1 VM Account Record Descriptions

The records produced by VM Account fall into two categories: resource usage and journaling. The resource usage records provide information about CPU use, I/O counts, and dedicated and temporary device use. The journal records provide information about security related events. The following sections cover the creation of each record type.

1 - Resource Records
2 - Journal Records

6.3.1.1 Resource Usage Records

Account records for resource consumption that are produced by the CP are created when a virtual machine:

a) logs off the system,

b) changes its account number dynamically,

c) is the subject of a CP ACNT CLOSE command.

Records produced by the subject of a CP ACNT CLOSE command. Records produced by RSCS are created when a spool file is successfully transmitted to its destination (or to the next intermediate node on its route). VCNA records are created when a virtual machine relinquishes control of a VCNA controlled terminal either by logoff or disconnect.

Network Data Transmissions Records (type 0C) are produced for any virtual machine user when their user directory option specifies NETAccounting or NETRouter.

These records are discussed in the following sections:

1 - Virtual Machine Resource Usage Records
2 - Device Allocation Records
3 - RSCS Records
4 - VCNA Records
5 - Network Data Transmissions Records

6.3.1.1.1 Virtual Machine Resource Usage Records

Resource Usage records (types 01 and C1) identify virtual (problem state) and total CPU use, logon duration (connect time as different from terminal use time), I/O counts to non-spooled devices, and use of spooled devices.

The I/O counts are for DASD, TAPE, and any other device that is controlled directly by the virtual machine, but does not contain counts for I/O performed through the *BLOCKIO system service, or for console I/O (the console is a spooled device).

The records read from a virtual card reader reflect the number of records in the spool file when it is opened for input, not the number of records read from the file.

6.3.1.1.2 Device Allocation Records

Device allocation records (types 02, C2, 03, and C3) are created when a virtual device is detached from the virtual machine. They can also be created using the information from section 6.3.1.1.

Device allocation records are created for any device that is allocated for exclusive control of the virtual machine. Allocations can occur from use of the CP DEFINE command for temporary disk, the operator CP ATTACH command, the DIAL command, or the DEDICATE command in the VM directory.

There are two device records: one for temporary disk (type 03/C3) and one for dedicated devices (type 02/C2). Each record provides device connect (allocation) time and device type information. In addition, the dedicated device record carries the real address of the terminal used for the virtual machine console (terminal), and the temporary disk record carries the amount of space allocated.

6.3.1.1.3 RSCS Records

RSCS records are not produced by the CP but are passed to the CP by RSCS using the account interface diagnose instruction X'4C'. These are generically known as user records or C0 records. They are identified in the CA MICS VM and CMS Analyzer by installation code in the %VMC0EXIT, so that a distinction is possible between RSCS records and records produced by other facilities.

The information provided by RSCS is the origin and destination of the spooled file (whether the file is transmitted or received), the spool class, original and local spool numbers, record count, and device type of the virtual device used to create the file.

6.3.1.1.4 VCNA Records

VCNA records (type 07) are created when a user logs off or disconnects from VM and has been using a VCNA controlled terminal as a console device. These records are created by VCNA and passed to the CP. VCNA records are not subject to the CP ACNT CLOSE command, therefore, if an installation views the resource use of a virtual machine on a periodic basis, VCNA data will not be available with time stamps that coincide with other Account records.

VCNA records provide counts of request units sent to and received from VTAM, characters sent to and received from the CP, the number of console lines written (which is different from full screen writes), and the number of copy requests using the CP copy screen image facility.

6.3.1.1.5 Network Data Transmissions Records

Type 0C (network data transmissions) records are produced for any virtual machine user with NETAccounting or NETRouter specified as an option in their user directory. Type 0C records are created when accounting is invoked using the CP ACNT command or when a field in the accounting record is about to overflow; and the user has sent or received data from a Network device.

These accounting records provide information about the Linux use of system resources, such as the use of virtual network devices. Virtual network devices include coupled Virtual Channel to Channel Adaptors, IUCV connections, and Virtual Network Interface Cards (NIC), sometimes known as Guest LANs.

The content of the type 0C records depends on the type of Network device being used for the data transmission; and there are three different Network device formats depending on the value of byte 78 in each record.

Column 78	Network Device
x'00'	Virtual NIC
x'01'	Virtual Channel to Channel Adaptor
x'02'	ICUV or APPC/VM

Accounting records produced when a data transfer occurred through a Virtual NIC have four additional sub-formats determined by the value in byte 77, called the data descriptor.

Column 77	Data Descriptor
x'00'	Bytes sent to or received from Routers
x'01'	Bytes sent to or received from non-Routers
x'02'	Bytes sent to or received via Broadcast
x'03'	Bytes sent to or received via Multicast

6.3.1.2 Journal Records

Account records for journaling are used to report security-related events in VM. The collection of these records is controlled by the SYSJRL macro in DMKSYS. Each of the journal records is discussed in the following sections:

1 - Invalid Link Attempt Records
2 - Successful Link Records
3 - Invalid Logon Attempt Records
4 - User Disconnect Records

6.3.1.2.1 Invalid Link Attempt Records

Type 06 (invalid link attempt) records are created when the number of invalid link attempts by a user exceeds a designated threshold. An invalid link attempt occurs when a user tries to link to another user's minidisk with an invalid password.

Type 06 records provide the following: the user ID issuing the command, time and date, terminal address or LUNAME, invalid password, and the user ID that owns the minidisk.

6.3.1.2.2 Successful Link Records

Type 05 (successful link attempt) records are created when a user links to the protected minidisk of another virtual machine. A minidisk is protected when it has a password other than 'ALL'.

Type 05 records provide the following: the user ID issuing the command, time and date, terminal address or LUNAME, the userid owning the minidisk, and the minidisk address of the owning virtual machine.

6.3.1.2.3 Invalid Logon Attempt Records

Type 04 (invalid logon attempt) records are created when the number of invalid logon attempts by a user exceeds a designated threshold. An invalid logon attempt occurs when a user tries to log on to another user's minidisk with an invalid password.

Type 04 records provide the following: the user ID issuing the command, time and date, terminal address or LUNAME, invalid password, and the user ID to be logged on.

6.3.1.2.4 User Disconnect Records

Type 08 (user disconnect) records are produced when a virtual machine is disconnected from the system (i.e., stays logged on, but without a terminal for a console). Unlike the other journal records, this record is not controlled by the SYSJRL macro in DMKSYS. Type 08 records provide the date and time of disconnect.

6.3.1.3 Other Account Records

Any product that runs in VM can add records to the CP account file. This is done by the standard interface DIAGNOSE X'4C'. Records created using this interface are called "user" records and contain the userid of the virtual machine creating the record and a record type code of C0. The remaining 70 bytes in the record are formatted by the application within the virtual machine. Several vendor products, including IBM products, use this interface: RSCS (whose data is processed by CA MICS), TSAF (Transparent Services Access Facility), SQL/DS, and others.

Some sites have code (local or vendor created) that adds new account record types for extending the information provided by VM. These records and the user (C0) records can be introduced into the CA MICS VM database through the use of exits available within the CA MICS Analyzer Option for VM/CMS. For more information, see Chapter 10 Modification.

6.3.2 VM Monitor Record Descriptions

The VM Monitor generates data that is organized into domains, each of which contains one or more records. The domains store sample data and event data.

Sample data uses single as well as high frequency samples. Single sample data is collected at a site-specified interval that can range from six seconds to one hour. High frequency sample data is collected at a site-specified interval that can range from .01 to 30 seconds. It is reported at the end of each interval.

Event data results from a monitor call instruction. It captures events, such as transactions, as they occur in the system.

The VM domains are structured as follows:

```
     SYSTEM    - provides system-wide utilization data

     MONITOR   - provides initial system configuration
                 information and monitor configuration
                 information and changes

     SCHEDULER - provides highly detailed data concerning the
                 scheduling of a virtual machine

     STORAGE   - provides data concerning the use of
                 CP-managed storage

     USER      - provides detailed virtual machine resource
                 utilization information

     PROCESSOR - provides CPU configuration change and
                 dispatcher data

     I/O       - provides detailed device utilization data,
                 and I/O subsystem configuration alteration
                 data

     SEEK      - provides further detail for DASD activity

     Virtual   - provides activity for virtual networks
     Network

     APPLDATA  - provides monitor data generated by
                 application programs for performance tuning
```

```
     Linux     - provides monitor data generated by Linux
                 guests running on a VM system.
```

Most domains contain both sample and event data, but a few contain only one type. The Seek and Schedule Domains contain only event data and the System Domain contains only sample data.

The System and Monitor Domains are automatically started when the monitor is started; and may not be halted. All other domains either rely on information in these two domains or provide supplementary data for them. Figure 6-1 illustrates the relationships.

The System Domain provides high-level, system-wide information. All other domains provide detailed information that can be used for performance problem analysis, problem determination, or to support the System Domain.

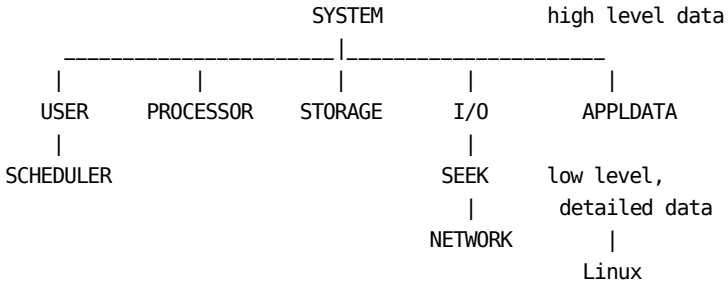


Figure 6-1. VM Monitor Domain Relationships

The following sections contain additional information:

1 - System Domain Records
2 - Monitor Domain Records
3 - Scheduler Domain Records
4 - Storage Domain Records
5 - User Domain Records
6 - Processor Domain
7 - I/O Domain Records
8 - Seek Domain
9 - Virtual Network Records
10 - APPLDATA Domain
11 - Velocity zVPS Linux Domain Records

6.3.2.1 System Domain Records

The System Domain (domain 0) provides an overview of system performance and resource consumption and utilization. The data in this domain is a high level view of all other domain data. The records in the domain provide information at the global system level and the processor level. This breakdown of information allows analysis of system activity on a complex-wide basis for general performance monitoring. It also provides information at the CPU level to determine how well VM is balancing and managing work based on scheduler and dispatcher parameters.

- The System Data Record contains significant activity counts on a per processor basis, including soft abends, standard diagnose instructions, instructions simulated, external interrupts, SIGPs, machine checks, real I/O instructions and interrupts, and system page and spool I/O operations. These represent system-wide utilization data. Utilization rates can be determined when combined with the Monitor Sample Profile record information.

- The Processor Data Record complements the System Data Record by providing CPU information on a per processor basis. The CPU information includes problem and supervisor state time, wait state time, SIE intercepts passed to CP and SIE instructions executed, page faults, fast path instruction simulations, and VMDBKs that were dispatched with hard affinity. This data is useful in determining the percents busy and waiting for each CPU.

- The Real Storage and Expanded Storage Records also store data on a per processor basis. These records include numbers of virtual machines in wait for a page read, the sizes of the various areas that comprise real storage as allocated by CP, the number of pages that CP free storage has extended into the DPA, pages in use by CP for page areas, and the number of page frames moved to and retrieved from expanded storage.

- The Communication Activity Record provides IUCV and VMCF counts for each processor. Communication Activity data includes the number of times data was transferred successfully and unsuccessfully between IUCV or VMCF partners. All CP IUCV services are categorized, along with categories for user IUCV and VMCF.

- The Processor Scheduler Data Record provides information about the number of virtual machines added to the dispatch list by queue type.

- Auxiliary Storage (CP-owned volumes) is reported on a system-wide basis. This information includes the sizes of the various system areas used for paging and spooling and the portion of each of these areas that is in use. The number of spool files that have been created or purged is reported, as well as the number of times an area type was more than 90% occupied.

- The Shared Segment Record reports the number of active discontiguous shared segments, the named saved segments and the number of active shared page frames that are provided for the entire system.

- The User Data Record provides system-wide information for virtual machine processor utilization. This record provides logged-on and dormant user counts, counts of trivial transactions completed for UP, MP, and QUICKDISP virtual machines; and CPU consumed by virtual machines in each of those states.

- Channel Path Contention data is also a system-wide measure. This information provides data on the number of channel paths that were concurrently busy and the number of times a particular channel path was busy.

- The Global Scheduler Data Record is used to provide: counts of virtual machines in various scheduler lists for the system, dispatch queue stay times and factors, and the total of all absolute and relative shares for virtual machines in the dispatch list.

- The User Wait States Record provides global system data summarizing the numbers of virtual machines in a wait state. The wait states are identified as I/O, PAGE, CPU, SVM (service virtual machine), console function, simulation wait, and asynchronous I/O wait. These are instantaneous counts of users in a wait state.

- The Logical Partition Utilization records provide data for the Central Electronic Complex (CEC) when VM operates in a logical partition. This includes LPAR control parameters, logical CPU dispatch time, logical CPU effective dispatch time, and LPAR management time.

6.3.2.2 Monitor Domain Records

The Monitor Domain (domain 1) provides two types of information: monitor configuration and changes to the monitor, and system configuration data.

Information about the monitor includes:

- Sample interval
- High frequency sampling rate
- Domains enabled
- Changes to the monitor while active

The system configuration data includes:

- Storage configuration and map
- Expanded storage configuration and assignment
- CP VOLUME configuration and map
- I/O subsystem configuration by subchannel
- Users logged on and running mode
- Scheduler parameter settings

6.3.2.3 Scheduler Domain Records

The Scheduler Domain (domain 2) provides detailed information that allows the tracing of a virtual machine through the scheduler, based on system interaction and scheduler list transition. The data includes terminal I/O, as well as eligible list to dispatch list transitions. Modification of scheduling algorithm parameters are reported, including SET SRM changes, SET SHARE changes, and SET QUICKDISP changes. Data is also reported based on the system time pop frequency of information produced on the INDICATE LOAD command.

6.3.2.4 Storage Domain Records

The Storage Domain (domain 3) provides data pertaining to memory use and allocation, expanded storage use, and paging.

The data includes:

- CP free storage subpool utilization
- Available list replenishment thresholds and replenishment scans
- Shared segment detail
- CP-owned volume I/O counts and configuration changes
- Shared segment use
- Expanded storage use
- Block paging statistics
- Shared address space usage
- Mapped minidisk volume I/O counts
- configuration changes, shared segment use, expanded storage

6.3.2.5 User Domain Records

The User Domain (domain 4) provides large amounts of data for a virtual machine and the VMDBKs associated with the virtual machine. The types of data included in this domain are logon/logoff information, interval end information, and transaction end information. The transaction data is based on the state model which reflects a transaction as viewed by the scheduler and monitor, rather than the event model, which required virtual machine or application action to signal transaction start and end.

6.3.2.6 Processor Domain

The Processor Domain (domain 5) provides changes in the CPU configuration within the processor complex and dispatcher information, such as VMDBK steals and dispatcher long paths taken. This domain also reports CPU configuration changes that result from the VARY PROCESSOR command.

6.3.2.7 I/O Domain Records

The I/O Domain (domain 6) provides information that is similar in structure to the Processor Domain. This data includes I/O subsystem changes, such as terminal enable/disable or volume attached to system. This domain provides detailed cache controller data, I/O counts, device behavior in SIE assist; and device connect, disconnect, and function pending times.

6.3.2.8 Seek Domain

The Seek Domain (domain 7) provides additional detail for DASD devices. This data can be used for determining the activity of a particular volume at the cylinder and head level. The information is used for analyzing the I/O utilization of a minidisk.

6.3.2.9 Virtual Network Records

The Virtual Network (domain 8) provides additional detail for Virtual networks. This data can be used for determining the activity and status (LinKUp or LinkDown) of a virtual NIC connection to a virtual network (Guest LAN) or VSWITCH).

6.3.2.10 APPLDATA Domain

The APPLDATA Domain (domain 10) provides monitor data for application programs. This data includes the application product identification, release level of the application, userid of the virtual machine, the offset to the area where the application data resides, and the application data. The usage of this data is application dependent.

6.3.2.11 Velocity zVPS Linux Domain Records

The Linux Domain (Domain C) provides data pertaining to internal Linux activity such as processes and applications executing within Linux guest.

6.3.3 CA VM:Account Record Descriptions

The first 80 bytes of the CA VM:Account record in the PERIODTE CURRENT file contain raw data in a standard format. For example, columns 1 through 8 always contain a userid, regardless of where the userid was located on the original (raw data format) record.

Records are identified by the accounting data type in columns 79-80 of a PERIODTE-format accounting record. The original record type code are represented in columns 81-82. The record type was originally located in columns 79-80 of the RAWDATA record.

Data types and record types in PERIODTE format records can have one of two record types. The record has the first record type (0n) if the operating system creates it. The record has the second record type (Cn) if a service virtual machine issues a DIAGNOSE X'4C' to create it. See table below.

Name	Datatype	Record Type
SESSION	01	01 or C1
DEDEV	02	02 or C2
TDISK	03	03 or C3
LINKS	05	05 or C5
VSCS	07	07 or C7
LGAI	0C	0C

Because the above mentioned CA VM:Account records contain the same information as the CP accounting records, see Section 6.3.1 for details about these records.

6.4 Operating System Details

This section provides details about the various components of VM systems. It describes the behavior of VM functions and the differences between similar functions in various levels of VM.

Subjects discussed in the following sections include:

1 - Instruction Simulation
2 - Shadow Table Maintenance
3 - PR/SM and LPAR

6.4.1 Instruction Simulation

To allow an operating system to run in a virtual machine, VM must allow all System 370 facilities and resources to be used by the operating system. However, in order to run more than one virtual machine at a time, VM must share the real facilities and resources among all the virtual machines in the system. This includes real memory, I/O devices, and CPU time. Virtual machines occasionally interfere with one another in their use of system facilities and CP must control the interferences using instruction simulation.

Certain instructions in a 370 system are known as privileged instructions. These instructions must be executed while the system is in supervisor state. VM owns the real hardware. In order to protect the virtual machines from each other and to protect itself from virtual machines, CP (the VM supervisor or control program) allows only a virtual machine to run in problem state.

CP is the only entity that runs in supervisor state in a VM system. This is known as real supervisor state. Virtual machines run in real problem state (with some exceptions noted later). This presents a problem for the operating system in the virtual machine: it expects to be able to execute privileged instructions, but CP has prevented it from doing so by running the virtual machine in problem state. CP solves this problem by intercepting the hardware interrupt generated by the attempt to execute a privileged instruction while the system is in real problem state. CP then determines whether the operating system in the virtual machine believes that it is running in supervisor state (this is termed virtual supervisor state) and, if so, enters instruction simulation to provide the function of the instruction to the virtual machine.

In the area of I/O, CP must intercept the I/O instructions for simulation. For start commands (SIO, SIOF, SSCH, etc.), CP must also translate the channel program so that the storage addresses provided by the virtual machine are converted to real addresses. In addition, CP performs cylinder relocation for DASDs that are shared, so that the virtual machine will read or write the correct area on the disk. Then, CP translates the address of the device provided by the virtual machine to the address of the real device before starting the I/O operation.

The process of instruction simulation can be very time-consuming and CPU-intensive. To improve performance, a number of assists are available for VM systems:

- 370-mode systems consist of VMA (virtual machine assist), ECPS (extended control program support), and PMA (preferred machine assist). These facilities allow the execution of some privileged instructions by an operating system in a virtual machine even though the real system is in problem state. PMA provides additional performance improvements by allowing a single virtual machine to execute in real supervisor state. This eliminates CP intervention in I/O to devices owned by the virtual machine, as well as eliminating most of the instruction simulation for the virtual machine.

- For XA mode systems, the performance assists are a part of SIE (the start interpretive execution instruction). SIE is a composite of the 370 VM assists, with additions applicable to VM/XA.

- SIE Assist is a feature of SIE. This feature allows the virtual machine operating system to initiate I/O operations to a device that has been dedicated to the exclusive use of the virtual machine without CP having to perform any translation or relocation.

The development of VM and of hardware assists for VM has reduced the overhead in CP to support a virtual machine to the point that, in many cases, it is almost negligible.

6.4.2 Shadow Table Maintenance

Shadow table maintenance is the mechanism that VM utilizes to manage the storage needs of virtual machines that are executing a guest operating system with one or more virtual address spaces (e.g., MVS and SVS). The following diagram shows how this mechanism works.

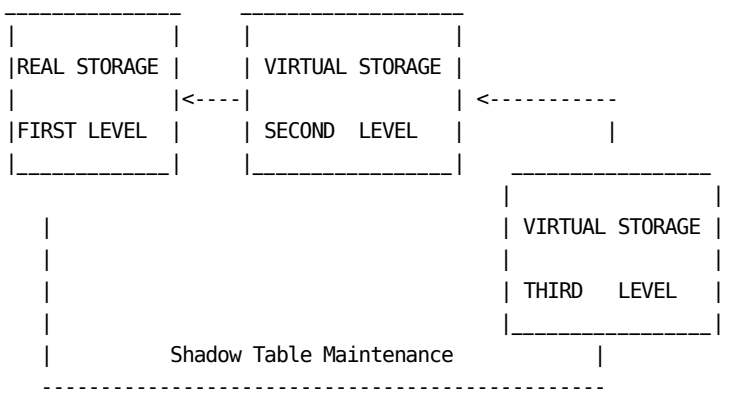


Figure 6-2. Shadow Table Maintenance

A guest operating system that is running in a virtual machine interprets the virtual storage of VM as real storage. The guest operating system therefore creates its own virtual address spaces, or third-level storage. Considerable overhead would be incurred by a virtual machine if each address had to be translated twice, (e.g., from third-level to second-level virtual storage, and then from second-level to first-level real storage). Shadow tables eliminate some of this overhead by enabling VM to selectively map third-level virtual storage directly to first level real storage.

Shadow table entries are first marked invalid. The shadow table can be invalidated in many ways (e.g., a new control register one load or PTLB instruction call by the virtual machine). The shadow table entries are filled in as the virtual machine runs and VM translates third-level storage to first-level storage.

The VM command that controls multiple shadow table maintenance is as follows:

```
   SET STMULTI n USEG xxx CSEG yyy

   The 'n' parameter controls the maximum number of shadow
   table origin blocks that VM can allocate before shadow
   table stealing and invalidation occurs.  The default value
   is 3; the limit is 6.  This limit can be increased by
   changing field DMKSYSMS in module DMKSYS.  For the MVS
   guest operating system, 'n' should be set to a value that
   is equal to 2 plus the average number of active
   initiators.

   USEG xxx defines the size of the pre-allocated page tables
   for each STO block.  If the value specified is too small,
   stealing occurs.  This can also degrade the performance of
   a guest operating system under VM.

   CSEG yyy defines the number of full segments at the high
   end of storage that are shared among all address spaces.
   For MVS or SVS, yyy should be the number of 64K segments
   in the PLPA.


Shadow Table Bypass

VM allows bypassing of shadow table support for the MVS
nucleus.  The area of storage to be bypassed must start at
location 0; and the virtual (third-level) and real
(second-level) storage addresses must be equal for that
virtual machine.

The command that establishes the bypass support is:

   SET STBYPASS nnnnnK

where "nnnnn" is storage size.
```

6.4.3 PR/SM and LPAR

Processor Resource/Systems Manager (PR/SM) is a hardware facility that enables logical partitioning of a physical processor complex, provides additional byte-multiplexer channel capability, and supports the VM multiple preferred guest function.

In Logically Partitioned (LPAR) mode, the resources of a processor complex can be distributed among multiple logical partitions, where a logical partition is a virtual processor complex that can run a system control program.

The performance of a logical partition is controlled by:

- The number of logical processors online. A logical processor is the name used to describe a central processor when in LPAR mode. A central processor is part of a processor complex.
- Whether processors are dedicated to or shared by logical partitions.
- The processing weight assigned to each logical partition and whether the capped function is turned on or off.
- Whether wait completion is on or off.
- The processor dispatch time interval.

The CA MICS Analyzer Option for VM/CMS provides measurements and status indicators describing PR/SM LPAR operations when VM is running in at least one logical partition. The VMXLCP (VMX Logical Processor Activity) file provides information about each logical processor's activity in a logical partition, while the VMXLPR (VMX Logical Partition Activity) file provides measurements for each logical partition in the processor complex.

PROCESSORS

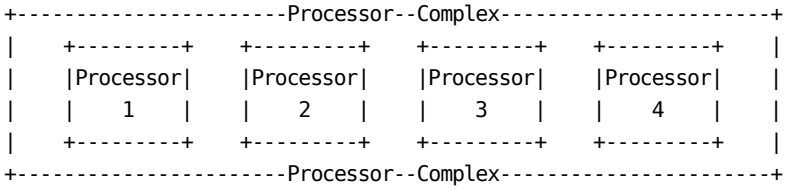
Processors can be dedicated to a logical partition or shared among multiple logical partitions.

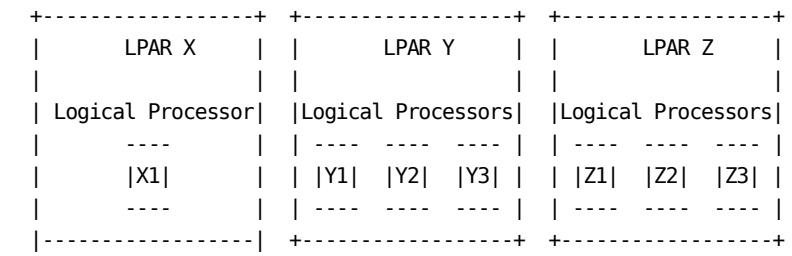
Processors that are dedicated to logical partitions are always available for use, but the capacity that is not used by that partition cannot be used by other partitions.

A shared logical processor can be dispatched on any sharable

physical processor. Therefore, there is no correspondence between the shared logical processors and the physical processors on which the logical processors are dispatched.

For example, the following diagram shows a site that has one processor complex containing four physical processors and three logical partitions. One LPAR defines one logical processor and the other two define three logical processors apiece.





If Processor 1 is dedicated to logical processor X1, the only work Processor 1 is permitted to do is work from logical processor X1. The work from logical processors Y1 through Y3 and Z1 through Z3 must be performed on Processors 2, 3, and 4.

If Processor 1 is shared by the logical processors, all of the shared logical processors are candidates to be dispatched on Processor 1. Processor 1's priorities are established by the processing weights and implementation of the capping and Wait Completion functions discussed later in this section.

The following data elements identify the configuration of a logical partition:

PRSMLPNM	Logical Partition Name
PRSMLPNO	Logical Partition Number
PRSMTPP	Physical Processor Count
PRSMNOPP	Total Sharable Physical Processors
PRSMTLPS	Number of Defined Logical Partitions
LPRCPUCT	Number of Logical CPUs
LPRSHRMD	Intervals with Shared Processors

```
  LPRCOLCT   Intervals in Measurement Collector LPAR
  LCPCPUCT   Number of Logical CPUs
  LCPSHRMD   Intervals with Shared Processor
  LCPCOLCT   Intervals in Measurement Collector LPAR
```

PROCESSING WEIGHTS

A logical partition with dedicated processors is not affected by processing weights.

A LPAR with shared processors uses processing weights to specify the portion of the shared processor resources allocated to a logical partition when there is contention for the shared processor resources.

Processing weights can range from 1 to 999. The relative share of processing resources available to a shared partition is calculated by dividing the processing weight of the partition by the total processing weight for all active, sharing logical partitions. For example, there are three shared logical partitions defined in the complex:

LPAR Name	Processing Weight	Number of Online Processors
X	30	1
Y	70	3
Z	50	3

The dispatcher allocates shared processing resources to each of the logical partitions as follows:

LPAR Name		Processing Share
X	30/150 =	20.00%
Y	70/150 =	46.67%
Z	50/150 =	33.33%

The share of processing resources for each online logical processor is calculated by dividing the relative share for the logical partition by the number of online logical processors.

LPAR Name		LCPU Share
X	20.00/1 =	20.00%
Y	46.67/3 =	15.56%

```
      Z        33.33/3 =   11.11%

A logical partition can receive more than its processing
share when there is excess processor capacity, and can
receive less than its processing share when its workload
demand is lower than the capacity specified by its weight.

The PR/SM capping function enables you to limit CPU resource
usage so a logical partition will not receive more than its
processing share.  Capping is done on a per logical processor
basis.

The following data elements report processing weight and
capping status for a logical partition:

  LPRPPWGT   LPAR Processing Weight
  LPRCAPP    Intervals with Capping
  LCPPPWGT   LPAR Processing Weight
  LCPCAPP    Intervals with Capping

WAIT COMPLETION

The Wait Completion setting does not affect logical
partitions with dedicated processors.

The Wait Completion setting controls whether a shared
processor is dispatched on an event-driven basis (Wait
Completion=Off) or a time-driven basis (Wait Completion=On).
With event-driven dispatching, the physical processor is
reassigned to another logical processor when the logical
processor enters a wait state or when it reaches the end of
its dispatch time slice.  With time-driven dispatching,
shared processor resources are distributed on the basis of
time intervals.  Logical processors are given an entire
dispatch interval even if they have no work to do.

The operator can limit the amount of continuous physical
processor running time that a logical processor can be
dispatched.  Values range from 001 through 100 milliseconds.
By default, processor running time is dynamically calculated
based on the number of physically shared processors and the
number of active logical processors.

The following elements indicate the Wait Completion status of
a logical partition or a logical processor:

  LPRWAITF   Intervals with Wait Completion On
  LPRGTS     Global Time Slice
  LCPWAITF   Intervals with Wait Completion On
```

```
DISPATCH TIME AND MANAGEMENT TIME

LPAR dispatch time is accumulated when a physical processor
is assigned to a logical processor within a logical
partition, whether or not the logical processor is actually
busy.  LPAR dispatch time includes wait time if Wait
Completion is enabled.  (See the WAIT COMPLETION discussion
earlier in this section.)

There are two types of LPAR management time:  time spent
managing the physical configuration and time spent managing
the partition.  To differentiate between them, the CA MICS
Analyzer Option for VM/CMS includes data from an extra
logical partition with the name "PHYSICAL" in its VMXLPR and
VMXLCP files.

Time spent managing the physical configuration is contained
in the logical partition named PHYSICAL.  Time spent
managing the physical configuration is not attributable to
any of logical partitions.

Time spent managing the partition is contained in logical
partitions other than the partition named PHYSICAL.

The following data elements can be used to assist in
analyzing CPU busy and LPAR management overhead:

LPRLPDTM   LPAR Dispatch Time
LPREPDTM   LPAR Effective Dispatch Time
LPRLPMTM   LPAR Management Time
LPRAVDTM   Average LPAR Dispatch per Interval
LPRPCSSU   LPAR Real CPU Utilization
LPRPCESU   LPAR Effective CPU Utilization
LPRPCMTM   Percent LPAR Management Overhead
LPRDURA    LPAR Duration
LPRMXSSU   Maximum Real CPU Utilization
LPRMNSSU   Minimum Real CPU Utilization
LPRMXDTM   Maximum LPAR Dispatch Time
LPRMNDTM   Minimum LPAR Dispatch Time
LPRMXMTM   Maximum Pct LPAR Management Overhead
LPRMNDTM   Minimum Pct LPAR Management Overhead

LCPLCDTM   LCPU Dispatch Time
LCPEFDTM   LCPU Effective Dispatch Time
LCPLCMTM   LCPU Management Time
LCPLPDTM   LPAR Dispatch Time
LCPEPDTM   LPAR Effective Dispatch Time
LCPLPMTM   LPAR Management Time
```

```
LCPPCSSU   LCPU Shared System Utilization
LCPPCSIL   LCPU Dispatch Share in LPAR
LCPDURA    LPAR Duration

CPU UTILIZATION

To determine the CPU time consumed by a logical partition
that runs a VM system, you need to know whether the logical
processors share physical processors or have dedicated
physical processors.

Note:  For a logical partition that runs z/OS, see $PRSMCPU
in the Data Dictionary appendix.

Effective real CPU busy time measures physical processor
resources that are actually consumed by the workloads within
a logical partition.

Physical Processor Dedicated to a Logical Processor:

  LPRPCVPU (LPAR Logical CPU Utilization) is 100% busy.
  LPREPDTM (LPAR Effective Dispatch Time) is the time that
  processor resources were allocated to the partition, but
  not what was actually used by VM.  Use the VMXSGP file's
  SGPTOTTM element (Total CPU Time Used) for effective real
  CPU time consumed by VM SCP.  SGPWATTM (Total System Wait
  Time) is VM's idle time, but this time cannot be used by
  any of other logical partitions.

Physical Processor Shared by Logical Processors with Wait
Completion On:

  LPREPDTM (LPAR Effective Dispatch Time) is the time that
  processor resources were assigned to the partition
  (excluding management time), but not what was actually used
  by VM.  With Wait Completion enabled, the physical
  processor is not reassigned to another logical processor
  until the dispatch time slice ends.  Use the VMXSGP file's
  SGPTOTTM element (Total CPU Time Used) for effective real
  CPU time used by the logical partition.  The difference
  between LPREPDTM and SGPTOTTM in a measurement interval is
  the time that VM has no work to do, but a physical
  processor is still allocated to a VM partition.

Physical Processor Shared by Logical Processors with Wait
Completion Off:

  Use either the VMXLPR file's LPREPDTM element (LPAR
  Effective Dispatch Time) or the VMXSGP file's SGPTOTTM
```

element (Total CPU Time Used) for effective real CPU busy time used by the VM system. This time does not include the LPAR management time. Even if VM has no work to do, a physical processor is still allocated to the VM partition so no time is wasted.

To compute the processor complex's idle time, which is the capacity that is neither used by the workload within a logical partition nor used by the LPAR management functions, do the following:

1. Add the value of effective real CPU busy time from all the partitions except PHYSICAL at the measurement interval.

2. Add the value of LPAR management time (LPRLPMTM) from all the partitions including PHYSICAL at the measurement interval.

3. Multiply the LPAR duration (LPRDURA) from a logical partition that was active during the measurement interval by the quotient of the total number of physical processors (PRSMTPP) divided by the number of active LPAR intervals (LPRLPAR) during the measurement interval.

   The result is the total physical processor time available in the processor complex.

4. Idle time is total physical processor time available minus effective real CPU busy time from all partitions except "PHYSICAL" minus total LPAR management time (that is, items 3, 1, and 2 listed above.)

   Idle time includes the time that a physical processor is assigned to a logical processor, but the physical processor is not actually busy because the logical processor is dedicated. Or, if the physical processor is shared, Wait Completion is set to On.

6.4.4 Virtual Machine CPU Time vs. Linux CPU Time

The CA MICS Analyzer Option for VM/CMS provides CPU utilization information at the VM guest level and for all processes that executed on a Linux system.

A Linux operating system running under z/VM is a guest or virtual machine. For each Linux guest, CPU utilization information is provided in z/VM account records and monitor data. The CA MICS Analyzer Option for VM/CMS stores this information in the VM User Resource Accounting (VMCCMU) and VMX User Activity Sample Data (VMXVXU) files.

The VMCCMU file is derived from the CP accounting '01' record and contains the following CPU time data elements:

```
 CMUOTIME - Supervisor CPU Time
 CMUTTIME - Total CPU Time
 CMUVTIME - Virtual CPU Time
```

The VMXVXU file is derived from VM Domain 4 Record 3 monitor records and contains the following CPU time data elements:

```
 VXUTTIME - Total CPU Time Used
 VXUVTIME - Virtual CPU Time Used
```

Internal Linux CPU utilization for all Linux processes that executed on a Linux system is recorded in Domain C Record 9 monitor records generated by Velocity Software's zVPS product. The CA MICS Analyzer Option for VM/CMS stores this information in the Linux Process Activity (LNXLXP) file.

The LNXLXP file contains the following CPU time data elements:

```
 LXPSTIME - Linux Process System CPU Time
 LXPTTIME - Linux Process Total CPU Time
 LXPVTIME - Linux Process User CPU Time

 LXPCHSTM - Linux Child Process System CPU Time
            (LXPCHSTM is added to LXPSTIME)

 LXPCHTTM - Linux Child Process Total CPU Time
            (LXPCHTTM is added to LXPTTIME)

 LXPCHVTM - Linux Child Process User CPU Time
            (LXPCHVTM is added to LXPVTIME)
```

z/VM CPU Time and Internal Linux CPU Time Relationship

```
-----------------------------------------------------

Internally, Linux does not record realistic CPU times.  For
each process, the CPU times recorded internally by Linux are
reasonably correct proportionately, but not in absolute
terms, meaning that the Linux operating system is not aware
it is sharing CPU resources with other Linux guests.  To deal
with this issue, the total CPU time of the Linux guest as
recorded by the z/VM monitor is pro-rated proportionately
across all of the processes on an interval basis. The
pro-ration is based on the CPU time Linux 'thinks' each
process used.

If you were to total LXPTTIME for all the processes in the
LNXLXP file for z/VM Linux guest for a given interval, it
'should' equal VXUTTIME in the VMXVXU file and CMUTTIME in
the VMCCMU file for the same Linux guest over the same
interval.  The intent of Velocity Software is to achieve a
100% agreement of these CPU times, but there are boundary
issues with processes crossing over interval boundaries and
other issues.  Over time their software updates continuously
get closer to achieve the 100% agreement.

The CPU time data element values in the LNXLXP file should
never be compared with the corresponding CPU time data
element values in the VMCCMU and VMXVXU files.  LXPVTIME is
roughly analogous to TCB (Task Control Block) time for a z/OS
system, and LXPSTIME is roughly analogous to z/OS SRB (System
Resource Block) time. TCB time is the CPU time used while
executing application code and SRB time is the CPU time used
executing system code called by the application code.  The
LNXLXP CPU times reflect Linux internal (system) versus
process (user) allocations and have nothing to do with the
z/VM system and user times in the VMCCMU and VMXVXU files.
```

Chapter 7: PARAMETERS

This chapter provides information about the installation of the CA MICS Analyzer Option for VM/CMS and should be used by the system administrator as a detailed reference in conjunction with the PIOM.

This chapter shows you how to complete the required definitions for installing the CA MICS Analyzer Option for VM/CMS as part of the CA MICS installation process. Installing CA MICS requires both of the following:

- Gaining the necessary understanding of your site and its needs
- Translating that understanding into CA MICS parameters

This chapter will lead you through the parameter specifications required for installing the Analyzer. You must do the following:

- Arrive at a number of policy decisions
- Fill out several worksheets
- Translate the worksheet entries into the corresponding CA MICS parameter library member entries

The focus of this chapter is on the considerations that are unique to the CA MICS Analyzer Option for VM/CMS. Chapters 2 and 3 of the PIOM document the mechanics of the CA MICS installation process and include checklists that describe each installation step.

If you have a question at any time during your review of the material presented here, contact CA Technical Support at http://ca.com/support.

This section contains the following topics:

7.1 Complex Level Parameters (see page 462)
7.2 Unit Level Parameters (see page 476)

7.1 Complex Level Parameters

This section presents a discussion and explanation of parameters that are defined at the CA MICS complex level and affect the CA MICS Analyzer Option for VM/CMS. The following topics are discussed:

1 - Account Codes
2 - Component Generation Parameters (VMCGENIN)

7.1.1 Account Codes

This section explains how to define the CA MICS Analyzer Option for VM/CMS account code structure. These definitions are valid for every database unit in the database complex.

In CA MICS, data is stored by account number in the VM User Resource Accounting File (VMCCMU), VM Device Accounting File (VMCDAC), VM RSCS Accounting File (VMCVRA), and VM Terminal Network Usage File (VMCNTU).

The parameters in sharedprefix.MICS.PARMS(VMCACCT) define:

- The number of accounting fields that will be carried in the file
- The length of each field
- The descriptive names that are associated with each field

Note that once you have defined these fields, you must provide a routine that assigns them values (the VMCACRT member).

Because the accounting fields are part of the sort sequence, there will be at least one record for each unique combination of values (and perhaps more than one because there are other fields in the keys). The tradeoff you must make is between keeping a fine level of granularity via the accounting fields and therefore having a large database, or keeping a coarser level of granularity but perhaps not meeting your reporting or analysis requirements.

Once you have determined what the accounting structure is, you must decide how you want it expressed in the database. To do so, you must determine the number of accounting fields (CA MICS supports from 1 to 9) and what will be placed in them.

The following sections discuss:

1 - Account Code Definition (VMCACCT)
2 - Account Code Derivation Exit (VMCACRT)

7.1.1.1 Account Code Definition (VMCACCT)

The CA MICS Analyzer Option for VM/CMS accounting fields have the names of the form VMCACTn, where n is the sequential number of the accounting field. If three fields are defined, they will be VMCACT1, VMCACT2, and VMCACT3 (in your accounting structure, these might identify division, region, and project, respectively). The sequential number is called the accounting field "level" number. There is a maximum of nine levels. A sample account code structure is provided with the distribution materials.

Figure 7-1 is a worksheet for collecting the data. Once you have completed this form, code the contents of sharedprefix.MICS.PARMS(VMCACCT) as follows:

- Blank statements are permitted. Comments are coded by beginning the statement with an asterisk (*).

- Account levels are provided in ascending order, starting with 1.

- Up to nine levels are permitted, with no gaps between the numbers permitted.

- The format of a statement is free-form, but positional. All parameters are required. The format of each statement is as follows:

  n m 'descriptive title assigned to VMCACTn'

  where:

  'n' is account level.

  The level of importance of each element is specified, with Level 1 being the most important (major field) and the highest level number being the least important (minor field); 1 to 9 levels can be specified.

  'm' is field length.

  The length of the specified element, which may range from 1-30.

  'descriptive title' is the description of the account level.

  The title that describes the account code field is

limited to forty characters.

o An account level can be deactivated in the DAYS, WEEKS, MONTHS, and YEARS timespans. Setting account code levels by timespan is accomplished by specifying the timespan mask on the account code statements in VMCACCT, as follows:

n T(......) m 'descriptive title assigned to VMCACTn'

where:

Each "." position in the timespan parameters represents a timespan, in this order: DETAIL, DAYS, WEEKS, MONTHS, YEARS, and TABLES. For each timespan, a "Y" indicates that the account level should be active, or an "N" indicates that the account level should be inactive.

The following example shows the VMCACT3 element being retained only at the DETAIL and DAYS timespans.

```
1            8  'DIVISION'
2            8  'REGION'
3  T(YYNNNY) 8  'PROJECT'
```

See the PIOM, Section 2.3.1.7, for more information on the timespan mask.

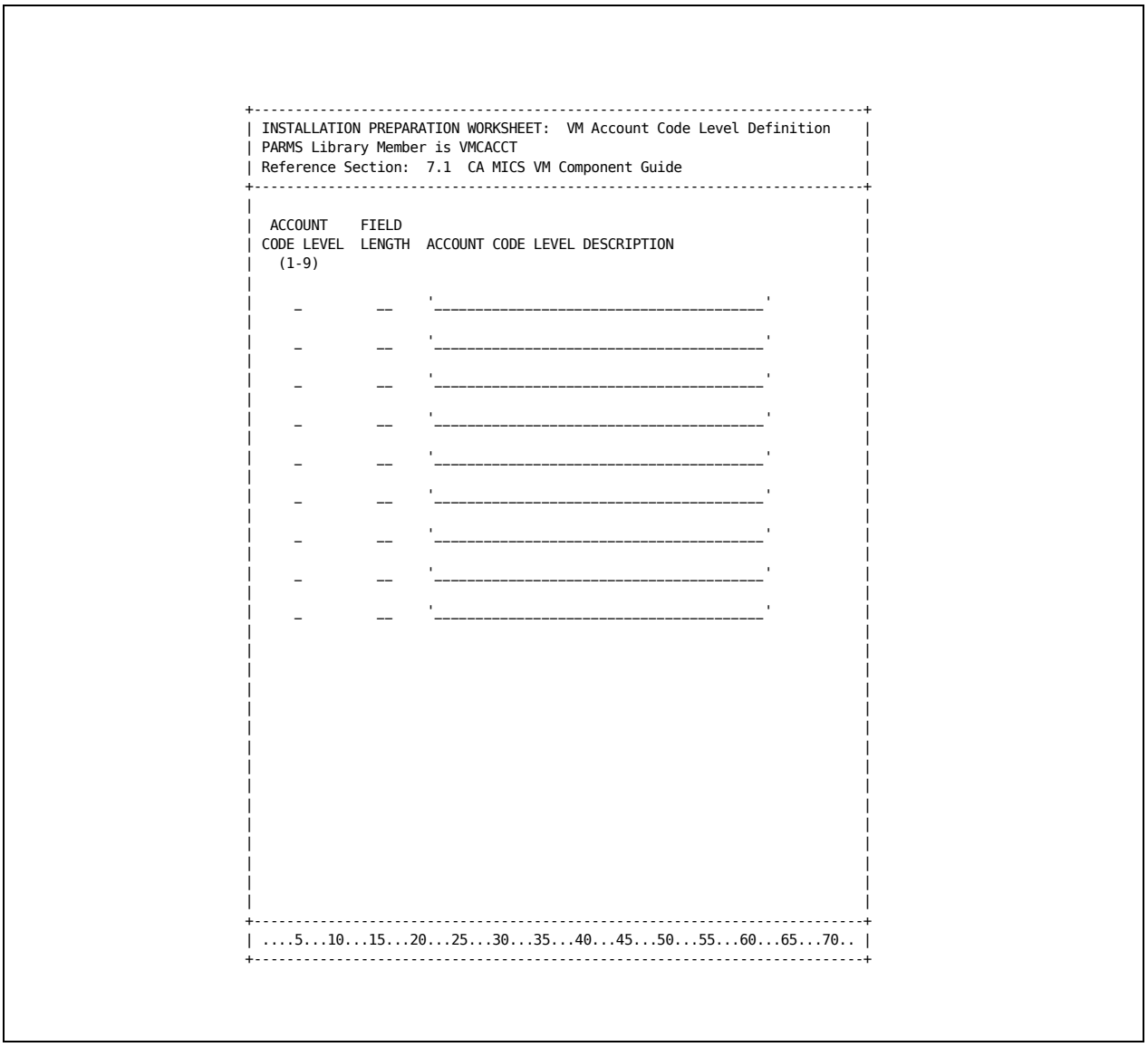


Figure 7-1. CP Account Code Level Definition Worksheet

7.1.1.2 Account Code Derivation Exit (VMCACRT)

This section explains how to code an account code exit for the CA MICS Analyzer for VM/CMS. These definitions will be used for every database unit that contains the Analyzer in the database complex.

An Account Code Exit Routine is a user-written routine that is invoked for each input record processed. This SAS routine builds the account code data elements defined in the previous section. For example, if the VMCACCT member in sharedprefix.MICS.PARMS defines four account code levels, this user routine has the responsibility of building the data value of those four fields for each record processed.

The user account code routine is written in SAS code. The testing and accuracy of the process is the responsibility of the user. However, the CA MICS Analyzer Option for VM/CMS provides a sample Account Code Exit Routine to help verify the accuracy of the user modifications to the routine. The worksheet for structuring the Account Code Exit Routine is shown in Figure 7-2.

Each of the data elements contained in the User Resource Accounting File (VMCCMU), Device Accounting File (VMCDAC), RSCS Accounting File (VMCVRA), or Terminal Network Usage File (VMCNTU) is available for use by the Account Code Exit Routine. The account code exit can determine which file it has access to by examining the element WRK_NAME. WRK_NAME is set to the name of the file accessed, that is, VMCCMU, VMCDAC, VMCVRA, or VMCNTU. Refer to the appropriate section in Chapter 5 for a list of the elements that are available for use.

Data elements frequently used to determine account code values are:

USER - VM Userid
VMCACNT - CP Account Code

In the VMCNTU file, USER is either a VM userid or a terminal identification.

```
The object of the account code derivation process is to build
the data elements VMCACT1 to VMCACT9 (as many as have been
defined in VMCACCT) from the data made available in the VMC
detail data.  Follow the exit coding guidelines given in
Section 2.3.1.5, Notes on Coding CA MICS Parameters, in
Chapter 2 of the PIOM.

This example assumes that you have defined three account code
levels in the VMCACCT parms member that are derived as
described below:

VMCACT1 - Corporate Division
VMCACT2 - Region within the Division
VMCACT3 - Project
```

The VMCACCT member for the account definition would be:

```
1 8 'DIVISION'
2 8 'REGION'
3 8 'PROJECT'
```

```
+-------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET:  VM Account Code Exit               |
|                                                                         |
| PARMS Library Member is VMCACRT                                         |
| Reference Sections: 7.1 CA MICS VM Component Guide                      |
+-------------------------------------------------------------------------+
|                                                                         |
| * VALIDATE FOR VALID ACCOUNT CODES, WHERE POSSIBLE ;                    |
|                                                                         |
| IF { account data is valid } THEN DO;                                   |
|                                                                         |
| * BUILD ACCOUNT CODE FIELDS AS IN THE WORKSHEET 7-1                     |
|                                                                         |
|     VMCACT1=field source 1 ;                                            |
|     ...                                                                 |
|     ...                                                                 |
|     ...                                                                 |
|     VMCACTn=field source n ;                                            |
|     _____________________________________________________________       |
|     _____________________________________________________________       |
|     _____________________________________________________________       |
|     _____________________________________________________________       |
|     _____________________________________________________________       |
|                                                                         |
| END;                                                                    |
| ELSE DO;                                                                |
|                                                                         |
| * ROUTINE TO BUILD INSTALLATION OVERHEAD ACCOUNT CODES ;                |
|                                                                         |
|     VMCACT1='overhead category' ;                                       |
|     ...                                                                 |
|     ...                                                                 |
|     ...                                                                 |
|     VMCACTn='overhead category' ;                                       |
|     _____________________________________________________________       |
|     _____________________________________________________________       |
|     _____________________________________________________________       |
|     _____________________________________________________________       |
|                                                                         |
| END;                                                                    |
|                                                                         |
+-------------------------------------------------------------------------+
| ....5...10...15...20...25...30...35...40...45...50...55...60...65...70.. |
+-------------------------------------------------------------------------+
```

Figure 7-2. Account Code Routine Definition Worksheet

The SAS code defined for the user account code exit would be as follows:

```
* ABC'S  ACCOUNT CODE DERIVATION EXIT
*
*  THE VM USERID (USER) IS THE SOURCE OF INFORMATION
*  FOR THE ACCOUNT CODES
*
*   IF THE FIRST TWO CHARACTERS OF USER ARE NOT EQUAL TO
*       'DE', 'PR' OR 'TS', OR THE REGION IS BLANK, THE
*       ACCOUNT CODES ARE SET TO THE INSTALLATION
*       OVERHEAD ACCOUNT CODES.
*;

    VMCACT1=SUBSTR(USER,1,2) ;
    VMCACT2=SUBSTR(USER,3,2) ;

    IF (VMCACT1 NE 'DE' AND
        VMCACT1 NE 'PR' AND
        VMCACT1 NE 'TS') OR
        VMCACT2 EQ ' '   OR
        VMCACT2 EQ '00'X
      THEN DO;
       VMCACT1='**';
       VMCACT2='*';
       VMCACT3='   ';
    END;
    ELSE VMCACT3 = SUBSTR(USER,4,5);
```

7.1.1.3 Account Code Derivation for Linux Exit (LNXACRT)

This section explains how to code an account code exit for the files in the Linux (LNX) information area.

This exit routine is a user-written routine. It is similar to the VMCACRT routine described in the previous section and is invoked for each Linux process activity input record processed. This SAS routine builds the account code data elements defined in the previous section. For example, if the VMCACCT member in sharedprefix.MICS.PARMS defines four account code levels, this user routine has the responsibility of building the data value of those four fields for each record processed.

The user account code routine is written in SAS code. The testing and accuracy of the process is the responsibility of the user. However, a sample Account Code Exit Routine is provided to help verify the accuracy of the user modifications to the routine. The worksheet for structuring the CP Account Code Exit Routine is shown in Figure 7-2.

Each of the data elements contained in the Linux Process Activity File (LNXLXP) is available for use by the Account Code Exit LNXACRT Routine. See the appropriate section in Chapter 5 for a list of the elements that are available for use.

Data elements frequently used to determine account code values are:

LNXLUSNM - Linux Logon User Name
LNXLGPNM - Linux Logon Group Name

7.1.2 Component Generation Parameters (VMCGENIN)

A generation definition statement member is provided for the CA MICS Analyzer Option for VM/CMS in the sharedprefix.MICS.GENLIB member VMCGENIN. VMCGENIN identifies the information areas, files, and data elements.

This section provides instructions for modifying VMCGENIN to meet the requirements of your site. The following topics are covered:

1 - GENLIB OPTION Statement
2 - Activating DSUM Support for Analyzer Files

7.1.2.1 GENLIB OPTION Statement

A Generation Definition Statement member is provided in the sharedprefix.MICS.GENLIB library under the name VMCGENIN. Review this file to check the OPTION statement definition for VMC. You can change the default definition to meet your needs, as described below.

Note: The VMCGENIN member contains uppercase and lowercase characters. The keywords for the OPTION statement should be entered in uppercase.

OPTION statement keywords for VMC:

VMACT/NOVMACT - Include CA VM:Account data elements
VMACH/NOVMACH - Include CA VM:Account charge fields
ECPMF/NOECPMF - Include 0/20 Monitor record fields

VMCGENIN OPTIONS - IMPACT ON THE DATABASE

The options in VMCGENIN are used to cause the inclusion of elements in the database for the different levels of VM. This is accomplished by using cluster codes. Each element defined in VMCGENIN has a cluster code associated with it. These are:

Code	OPTION Keyword	Description
00		Not system level dependent
19	VMACT	VM:Account fields
10	VMACH	VM:Account charge fields only
20	ECPMF	0/20 Monitor Record Channel Path Extended Measurements fields

Note: When VM is upgraded from one level to the next, the records created by the Monitor are not reformatted. Instead they are extended to provide space for the new information. This provides upward compatibility in the Monitor data.

The same rule applies to VM/ESA and z/VM. z/VM is a successor to VM/ESA. OPTION ZVM3 will keep data elements for all z/VM releases; however, data elements added by the newer level of VM will not be found in the database until a VMCCGEN is run. By running a VMCCGEN, the new VM level is added to the CA MICS complex.

Activating CA VM:Account Data Elements

These options are provided to activate CA VM:Account data elements:

- VMACT/NOVMACT activates and deactivates CA VM:Account non-charge data elements.
- VMACH/NOVMACH activates and deactivates CA VM:Account charge data elements.
- ECPMF/NOECPMF activates and deactivates Channel Path Measurement data elements from the 0/20 monitor record.

For example, if the VMCGENIN OPTION specification is VMACT NOVMACH, CA VM:Account data elements with cluster code 19 will be kept in the database and CA VM:Account data elements with cluster code 10 will not be kept.

7.1.2.2 Activating DSUM Support for Analyzer Files

General support for DSUM (exception daily summarization) is provided in the CA MICS Analyzer Option for VM/CMS. This support is provided to allow a site to reduce the size of the online databases by eliminating the DETAIL timespan of a file and retaining the DAYS timespan in its place. This is typically used when the detail information is not required for reporting purposes. To activate this support, the following steps must be taken:

1 - The FILE statement for the product file for which this support is required must specify exception processing for the summarization from the DETAIL file to the DAYS file. For more information, see Section 4.2.4.2.1 of the System Modification Guide.

2 - Sharedprefix.MICS.SOURCE must contain a DSUM member for the file. The name for this member is #fffDSUM, where fff is the three character identifier for the file. See sharedprefix.MICS.SOURCE(#VMUDSUM) for an example of a DSUM routine.

3 - Change the FILE statement in VMCGENIN by changing the DAYS timespan indicator to E. If the DETAIL timespan is being deactivated (by specifying N for the DETAIL timespan indicator), then the audit archive must also be deactivated. Follow the checklists in Sections 6.2.2.1 and 6.2.3.2 of the System Modification Guide for deleting files and audit archives from a unit.

4 - Run sharedprefix.MICS.CNTL(VMCCGEN) to regenerate the CA MICS Analyzer Option for VM/CMS.

5 - Regenerate the VM unit level support macros by running prefix.MICS.CNTL(VMCPGEN).

```
6 - If a file is enabled or disabled for a timespan,
    prefix.MICS.CNTL(CYCLEGEN) must be run in each unit that
    has VMC installed.

7 - Regenerate operation job DAILY. Job WEEKLY must be
    regenerated if audit archive was deactivated for the
    file.

DSUM support can be deactivated by changing the FILE
statement to its previous state, then running VMCCGEN and
VMCPGEN for each unit in which the CA MICS Analyzer Option
for VM/CMS is installed.

The steps described above must be followed to complete the
activation of this support.
```

7.2 Unit Level Parameters

This section shows you how to specify the operational parameters which control the processing of the CA MICS VM and CMS Analyzer. Operational parameters are stored in the prefix.MICS.PARMS library associated with the database unit you will be running. These parameters are incorporated into the CA MICS system by the VMCPGEN process. Note that you must run the VMCPGEN process for these options to take effect.

Figure 7-3 identifies the SAS code generated during the VMCPGEN process.

MICS.PARMS Input Members	USER.SOURCE Output Members and MACRO Names	USER.LOAD Output Formats	MICS.PARMS Output Members
VMCOPS INPUTVMC	#VMCMSTR %VMCLUPA %VMCLUPX %VMCINCL	VMCSYS	WORKVMC

Figure 7-3. VMCPGEN Code Generator Cross-Reference List

The sections which follow describe the parameters which must be completed in prefix.MICS.PARMS(VMCOPS) prior to VMCPGEN execution.

1 - Component Options (VMCOPS)
2 - Component-Specific JCL Information (INPUTVMC)
3 - Database Space Modeling (DBMODEL)
4 - CA MICS JCL Parameters (JCLDEF)
5 - CA MICS System Identification (SYSID)

7.2.1 Component Options (VMCOPS)

The CA MICS Analyzer Option for VM/CMS options member (VMCOPS) is read by the parameter generation routine, VMCPGEN. VMCPGEN builds the master macros and stores them in prefix.MICS.USER.SOURCE(#VMCMSTR)

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 VMCOPS member defines the characteristics of input data, identifies the JCL reference to the input data, specifies the number of work data sets to be used, and designates specific action that the daily format routine should take when there is no match of CPU model or serial numbers. The following list contains the statements that can be coded in the VMCOPS member:

```
Required Statements
-------------------

OPTIONS

The following statements are required, depending on what
data is being processed.

MON      - IBM MONWRITE and Velocity monitor data
ACT      - CP Account data
VMACCT   - CA VM:Account account data

Optional Statements
-------------------

ALTCPU
COMMON
SPIN
VERIFY
WORK
MULTWORK|NOMULT
RESTART
INCRUPDATE

A worksheet for recording the parameter values for the VMCOPS
member is supplied in Figure 7-4.
```

7.2.1.1 OPTIONS Statement

```
The OPTIONS statement is required and used to identify the
 primary CPU for an orgsysid.

 The OPTIONS statement has the following format:

 OPTIONS orgsysid CPU-model CPU-serial-number

 where

 orgsysid  - is the original system identifier used in the
 --------    SYSID member of prefix.MICS.PARMS to identify
             this VM system.

 CPU-model - is the processor model that is used to run
 ---------   this VM system. It is specified as a number:
             one to four digits in length.

 CPU-serial-number - is the processor serial number of the
 -----------------   CPU that is used to run this VM system.
                     It is specified as a hexadecimal number:
                     one to six digits in length.

  The CPU model CPU and serial number can be determined doing
  one of the following:

  1. Ask your local data center's system programmer who is
     responsible for maintaining the systems for the CPU model
     and serial number of the system or systems to be added.

  2. Browse the monitor data from the system to be processed
     in CA MICS after it is created on your z/OS system.

    a. Set HEX on and execute a FIND: f '01000005'X 1 (this is
       Domain 1 Record 5 Processor Configuration record which
       contains the CPU model and CPU serial number).

    b. Set COLS on, see example below:

       ----+----1----+----2----+----3----+----4----+
       0000C1C2CEBE000000290BA00001000000000000
       1005A2D40540000000083F500004000000003000
                        mmsss

       position 19 mm=model, position 21 sss=serial number

       In the example above the CPU model is 2098 and the
       CPU serial number is 03BFA5
```

The VMCOPS OPTIONS statement would be coded as follows:

```
OPTIONS    SYSA  2098  03BFA5

ORGSYSID          : SYSA
CPU-model         : 2098
CPU-serial-number : 03BFA5
```

3. Run the following SAS program which inputs raw monitor data and processes only the Domain 1 Record 5 record:

```
DATA RAW;
FORMAT VMXCPUAD HEX2. INPUTTS DATETIME19.2;
INFILE INVMC LENGTH=RECLEN RECFM=VB;
    INPUT @001  MRHDRDM   PIB1.
               +1
                MRHDRRC   PIB2.
                INPUTTS   TODSTAMP8.
               +4    @;

     IF MRHDRDM = 1 AND MRHDRRC =5;
           INPUT @017
              +2
              WRK_MDL    PK2.
              WRK_SER    PIB3.  @;
 PROC FREQ;
   TABLES WRK_MDL*WRK_SER/ NOCOL NOROW NOPERCENT;
   FORMAT WRK_SER HEX6. ;
RUN;
```

OUTPUT:

```
        The SAS System

      The FREQ Procedure

  Table of WRK_MDL by WRK_SER

   WRK_MDL     WRK_SER

   Frequency|  013B11|  Total
   ---------+--------+
       2817 |     98 |     98
   ---------+--------+
   Total          98       98
```

In the example above the CPU model is 2817 and the CPU serial number is 013B11

The VMCOPS OPTIONS statement would be coded as follows:

```
OPTIONS    SYSA  2817  013B11

ORGSYSID          : SYSA
CPU-model         : 2817
CPU-serial-number : 013B11
```

Note: If a new system (ORGSYSID) is added to the CA MICS Analyzer Option for VM/CMS in a unit by adding an OPTIONS statement and a data source statement (for example, MON and ACT), make sure the new system (ORGSYSID) is added to prefix.MICS.PARMS(SYSID).

7.2.1.2 ALTCPU Statement

```
The ALTCPU statement is optional and can be used to assign
 alternate CPUs to this ORGSYSID using the CPU-serial-number.
 The ALTCPU statement can be used to transition to a new CPU
 system by assigning the new CPU serial number to this
 ORGSYSID during a system upgrade.

 The ALTCPU statement has the following format:

 ALTCPU orgsysid CPU-model CPU-serial-number

 where

 orgsysid  - is the original system identifier specified
 --------    on a previous OPTIONS statement.

 CPU-model - is the processor model that is used to run
 ---------   this VM system.  It is specified as a number:
             one to four digits in length.

 CPU-serial-number - is the processor serial number of the
 -----------------   CPU that is used to run this VM system.
                     It is specified as a hexadecimal
                     number: one to six digits in length.

 The ALTCPU statement must be accompanied with the OPTIONS
 statement and only applies to monitor data.  Using
 the ALTCPU statment for account data has no effect on
 assigning ORGSYSID in account data.

 Refer to the previous Section 7.2.1.1 to determine the
 CPU model and CPU serial number.

  After the CPU model and CPU serial number are determined,
  the ALTCPU statement is coded as follows:

     In the example, CPU model is 2098 and the CPU serial
     number is 02AAB1 the ALTCPU statement is coded as
     follows:

         OPTIONS    SYSA  2098  03BFA5
         ALTCPU     SYSA  2098  02AAB1

         ORGSYSID          : SYSA
         CPU-model         : 2098
         CPU-serial-number : 02AAB1

  SYSTEM UPGRADES
```

```
--------------

Use the ALTCPU statement to transition to a new system
during a system upgrade when there is a requirement to
process monitor data prior to a system upgrade and after a
system upgrade into the same DAY080 update step.  The steps
below can help with this transition. You can do one of the
following (In the example below, there is an upgrade to the
zEC12, IBM zEnterprise EC12).

1. Follow the steps in the previous section, 7.2.1.1 to
   determine the CPU model number and the CPU serial number
   of the upgraded system from the monitor created after
   the upgrade.

2. In our example, it was determined the new CPU model is
   2728 and the CPU serial number is 04DFC2 for the
   upgraded system.

     The VMCOPS OPTIONS and ALTCPU statements would be coded
     as follows:

       OPTIONS    SYSA  2098  03BFA5
       ALTCPU     SYSA  2728  04DFC2

     * CPU model and CPU serial number prior to upgrade

       ORGSYSID          : SYSA
       CPU-model         : 2098
       CPU-serial-number : 03BFA5

       CPU model and CPU serial number after upgrade in
       the ALTCPU statement

       ORGSYSID          : SYSA
       CPU-model         : 2728
       CPU-serial-number : 04DFC2

3. After the transition has passed and all the data from the
   old system has been processed, the VMCOPS OPTIONS
   statement can be updated:

   Change the OPTIONS statement to include the new
   CPU model and CPU serial number, then delete
   the ALTCPU statement.

       OPTIONS    SYSA  2728  04DFC2

Remember after any change to VMCOPS you must run VMCPGEN.
```

7.2.1.3 VMACCT Statement

```
The VMACCT statement is required only if CA VM:Account data
is being processed. It is used to identify the JCL reference
to be used when reading CA VM:Account data for this system.

The VMACCT statement has the following format:

VMACCT orgsysid ddname

where

orgsysid  - is the original system identifier specified
--------    on a previous OPTIONS statement.

ddname    - is the ddname used to read CA VM:Account data for
------      this system.  If ddname is specified as *, a
            COMMON statement must be used to provide the
            DDNAME.
```

7.2.1.4 MON Statement

```
The MON statement is required only if VM Monitor data is
being processed.  These statements identify the JCL reference
to use when reading VM Monitor data for this system.

The MON statement has the following format:

MON    orgsysid ddname

where

orgsysid  - is the original system identifier specified
--------    on a previous OPTIONS statement.

ddname    - is the ddname used to read VM Monitor data for
------      this system.  If ddname is specified as *,
            a COMMON statement must be used to provide
            the DDNAME.
```

7.2.1.5 ACT Statement

```
The ACT statement is required only if CP account data is
being processed. These statements are used to identify the
JCL reference to be used when reading CP account data for
this system.

The ACT statement has the following format:

ACT    orgsysid ddname

where

orgsysid  - is the original system identifier specified
--------    on a previous OPTIONS statement.

ddname    - is the ddname used to read CP account data for
------      this system.
```

7.2.1.6 COMMON Statement

```
The COMMON statement is optional and is used to identify the
JCL reference to be used when reading monitor data for one or
more VM systems from the same input ddname.

The COMMON statement has the following format:

COMMON datatype ddname

where

datatype  - is MON.

ddname    - is the ddname used to read the data type for
------      the system.

Note:  Do not use the COMMON statement if ddnames are
specified on the MON statement.  Only use the COMMON
statement if ddname is specified as *.
```

7.2.1.7 SPIN Statement

```
The SPIN statement is optional and specifies the retention
period for the product's spin files.

The SPIN statement has the following format:

SPIN filename DAYS nn CYCLES nn

where

filename  - is VMXSCF, VMXPCF, VMXACF, VMXXSF, VMXVXF,
--------    VMXDCF, or VMXSCX.

nn  - is the number of days or cycles of the retention
--    period. The DAY080 step uses this value rather than
      the suspendlimit in the EXECDEF member of
      prefix.MICS.PARMS to determine the retention period.
      The default value for both DAYS and CYCLES is 01.

A SPIN file is not aged using the standard CA MICS aging
macro.  It only has a 01 cycle.  To be deleted from the
database, an observation of a SPIN file must have been in the
file for the number of days specified and the number of
cycles (DAILY runs) specified.
```

7.2.1.8 VERIFY Statement

The VERIFY statement is optional and is used to determine what action to take when read from an input ddname specified on a MONITOR statement, or when a MON statement comes from a system not named on the associated OPTIONS or ALTCPU statement for the orgsysid.

The VERIFY statement has the following format:

VERIFY action

where

action - is ABORT or NOABORT. If the VERIFY statement is
------- omitted, the default is NOABORT. When ABORT is coded and a mismatch is detected, a U997 abend is used to halt processing. If the data is from a ddname specified on a COMMON statement, this option has no effect. When there is no match for all the systems with the ddname specified as * on a MON statement (depending on the data type on the COMMON statement), a U310 abend halts processing. However, USRSEL exit can be used to correct the orgsysid.

The following sample VMCOPS defines four VM systems whose monitor data and/or account data are processed by the CA MICS Analyzer Option for VM/CMS. VERIFY ABORT is specified to halt DAY080 processing if a mismatch on the CPU models and serial numbers occurs. Six working data sets (WORK) are specified to be used in DAY080 step. The retention period for the VMXSCF file and the VMXDCF file is defined to be three days or three DAILY runs.

VM01 is a 3090 system with two CPUs whose serial numbers are 123456 and 223456, respectively. VM Monitor data is read from the ddname MXA01, and CP account data is read from the ddname AXA01 for system VM01.

Both VM02 and VM03 are ES9000 systems and the input of VM Monitor data from both systems is from a single COMMON ddname: MSACOMN. VM03 also supplies CP account data using the ddname ASA04.

Note that these ddnames defined on VMCOPS (MXA01, AXA01, ASA03, and MSACOMN) must be defined on INPUTVMC.

```
   VERIFY  ABORT
```

```
WORK    6
COMMON  MON MSACOMN
*
SPIN VMXSCF DAYS 03 CYCLES 03
SPIN VMXDCF DAYS 03 CYCLES 03
*
OPTIONS VM01 3090 123456
ALTCPU  VM01 3090 223456
MON     VM01 MXA01
ACT     VM01 AXA01
*
OPTIONS VM02 9021 099999  /* ES9000 */
ALTCPU  VM02 9021 199999
MON     VM02 *
*
OPTIONS VM04 9121 288888  /* ES9000 */
MON     VM03 *
ACT     VM03 ASA03
```

7.2.1.9 WORK, MULTWORK, and NOMULT 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 three (3).
               The maximum is nine (9).

        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:

- Run cccPGEN
- 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 the following if neither MULTWORK nor NOMULT
parameters is specified:

MULTWORK _VT _VU ASC BPG CDI CDT CHC CMU CPV CT1 NDT NTU
         QCT SGP SPP STP VXT VXU XDV XLF XSS LXP

NOMULT   ACF DCF LCP LPR PCF SCF SCX STA VXF XDS XSF XST
         XTW


The implementation of internal restart support in this
component requires the creation of a number of temporary
files not defined in sharedprefix.MICS.GENLIB(VMCGENIN).

Files eligible for multiple WORK support are listed below:

Note:  Files marked with * are not enabled to use multiple
WORK files for performance reasons.

  _VT    VMX User Transaction Event Data File
  _VU    VMX User Activity Event Data File
* ACF    VMX CPVOL Configuration Data File
  ASC    VMX Shared Address Space Work File
  BPG    VMX Block Paging Data File
  CDI    VMX DASD Cache Activity File
  CDT    WRK Intermediate Cache Activity File
  CHC    VMX Channel Path Contention File
  CMU    VMC User Resource Accounting File
  CPV    VMX CP Owned Volume Data File
  CT1    WRK Intermediate Cache Activity File
* DCF    VMX Device Configuration File
* LCP    VMX Logical Processor Activity File
* LPR    VMX Logical Partition Activity File
  NTU    VMC Network Terminal Usage File
  NDT    VMC Network Data Transmission File
```

```
* PCF    VMX Processor Configuration File
  QCT    WRK Intermediate Performance Data File
* SCF    VMX System Configuration File
* SCX    VMX Scheduler Parameters File
  SGP    VMX System Global Data File
  SPP    VMX System Processor Data File
* STA    WRK Intermediate Status Work File
  STP    VMX System Timer Pop File
* VXF    VMX User Data Configuration File
  VXT    VMX User Transaction Sample Data File
  VXU    VMX User Activity Sample Data File
* XDS    VMX Device SEEKs Data File
  XDV    VMX Device Activity File
  XLF    WRK Intermediate User Activity File
* XSF    VMX Shared Storage Configuration
  XSS    VMX Shared Storage Management File
* XST    WRK Intermediate Shared Storage File
* XTW    WRK Intermediate Shared Storage File

The following section discusses changing the WORK option:

    1 - Change the Number of Work Files
```

7.2.1.9.1 Change the Number of Work Files

To change the number of work files used in the CA MICS VM and CMS Analyzer processing in Step DAY080, follow the checklist provided below for each unit.

```
   ************************************************
   *                                              *
   *           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.
```

7.2.1.10 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.

- RESTARTCKPT data_set_allocation_parameters
- RESTARTWORK data_set_allocation_parameters
- DYNAMWAIT minutes

Processing Phases:

This product employs two database update processing phases followed by the two common roll-up phases.

Phase	Description
FORMAT	Read the raw input data, convert to SAS format, and output to intermediate work files.
DBUPDATE	Sort intermediate work file contents, eliminate duplicate input data, prepare for DETAIL cycle creation, merge data across optional multiple work files, enhance data content, and create the new DETAIL cycle.
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.



The following section discusses enabling this option:

    1 - Enable Internal Step Restart
```

7.2.1.10.1 Enable Internal Step Restart

To enable the internal step restart in the CA MICS VM and CMS Analyzer, 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.
```

7.2.1.11 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(JCLGEND) or by specifying        *
*         DAILY in prefix.MICS.PARMS(JCLGENU) and           *
*         executing prefix.MICS.CNTL(JCLGENU).              *
*                                                          *
*         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.

- 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.

- 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.

- 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.

- 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.

- 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.

- 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):

  - Incremental update checkpoint file,
    ```
    //IUCKPT   DD DSN=prefix.MICS.ccc.IUCKPT,.....
    ```
  - Incremental update DETAIL
    ```
    //IUDETAIL DD DSN=prefix.MICS.ccc.IUDETAIL,.....
    ```
  - 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.

```
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.

- 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.

- 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.

- 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.

- 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.
```

```
INCRSPLIT
---------

This statement is optional and defaults to this:

INCRSPLIT IGNORE

Specify the following if you want the incremental update job
for this product to get input measurement data from the
output of the SPLITSMF job.  The optional
data_set_allocation_parameters are used by the SPLITSMF job
when creating the measurement data file for this product.

INCRSPLIT  USE  data_set_allocation_parameters

Note:  INCRSPLIT is ignored when you specify INCRUPDATE NO.

This option would be used when multiple products in a
single unit database are enabled to incremental update.  The
SPLITSMF job performs the same function for incremental
update jobs as the DAILY job DAYSMF step performs for the
DAYnnn database update steps.

o  The SPLITSMF job dynamically allocates, catalogs, and
   populates prefix.MICS.ccc.IUSPLTDS data sets for each
   product in the unit database for which you specified both
   the INCRUPDATE YES and INCRSPLIT USE parameters.  These
   data sets are then deleted after processing by the
   appropriate INCRccc job.

o  Specify data set allocation parameters, separated by
   blanks, according to SAS LIBNAME statement syntax.  If you
   need multiple lines, repeat the INCRSPLIT keyword on each
   continuation line.

o  INCRSPLIT 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.


Specify the following or accept the default if you want the
incremental update jobs for this product to get their input
measurement data from the data sets specified in the INPUTccc
(or INPUTSMF) member of prefix.MICS.PARMS:
```

```
INCRSPLIT  IGNORE

When you specify INCRSPLIT IGNORE, this product will NOT
participate in SPLITSMF job processing.


Example 1:

INCRSPLIT USE  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):

INCRSPLIT  USE  STORCLAS=MICSTEMP UNIT=SYSDA
INCRSPLIT       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.



The following section discusses enabling this option:

    1 - Implement Incremental Update
```

7.2.1.11.1 Implement Incremental Update

```
To implement incremental update in the CA MICS VM and CMS
Analyzer, follow the checklist provided below:


   ************************************************
   *                                              *
   *           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).

- 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.
- 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.

7.2.1.12 DETAIL Tape Processing 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:

- 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.

- The data set name (DSN) for each DETAIL tape in a component step is:

  tapeprefix.tmics.DETAIL.iiifff(+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)

- 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).

- 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.

- 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.

- 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.

- 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

7.2.1.12.1 VMC DETAIL Tape Supported Files

The CA MICS VM/CMS Analyzer supports DETAIL tape processing for the following file:

o LNXLXP - Linux Process Activity File

7.2.1.12.2 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

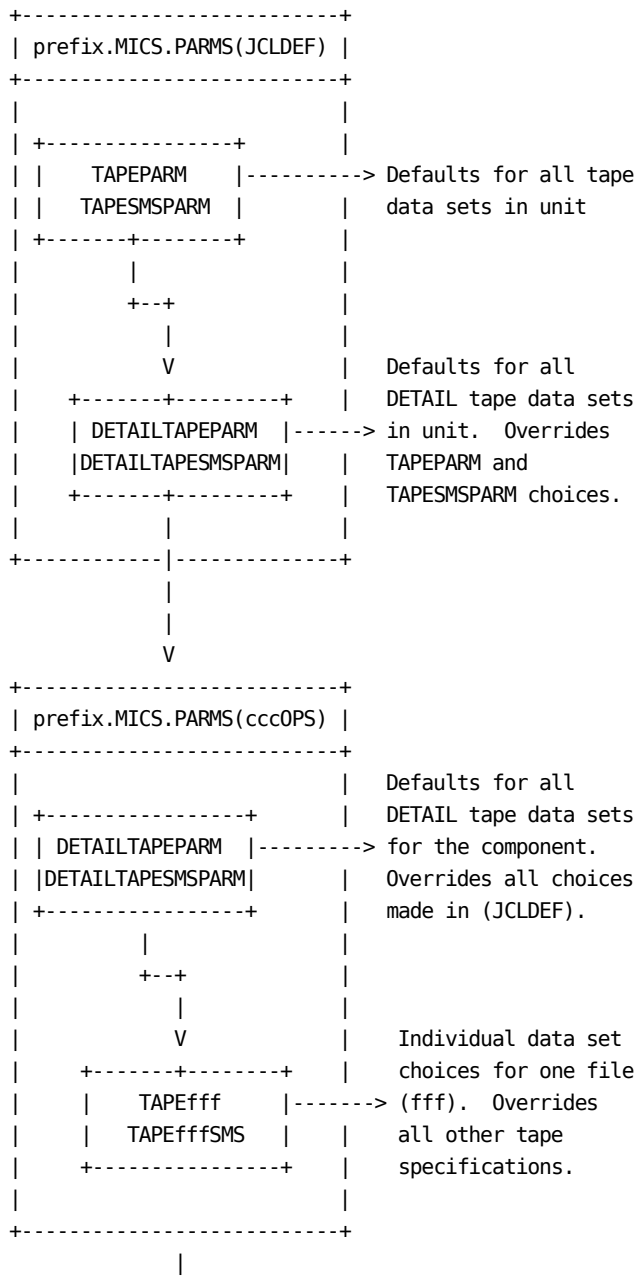
  Used for all DETAIL tape data sets created by a particular component. Overrides all choices that were made in JCLDEF. Specified in prefix.MICS.PARMS(cccOPS), where ccc represents the component.

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



```
                    V

// DD statements and GDG generation
```

7.2.1.12.3 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.

7.2.1.12.4 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:

- Each parameter is described in detail below.

- 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).

```
DETAILTAPEPARM GDG=nnn UNIT=unitname VOLCOUNT=n
DETAILTAPEPARM EXPDT=expdt / RETPD=nnn / NOEXPDT
DETAILTAPEPARM COMP/NOCOMP
DETAILTAPEPARM STACK/NOSTACK AFF/NOAFF
DETAILTAPEPARM MODEL='modeldsn' / MODEL=NOMODEL
DETAILTAPEPARM DISK / TAPE  SPACE=spacedef
DETAILTAPEPARM VOLSER=volser
```

The following is a sample DETAILTAPEPARM statement:

```
DETAILTAPEPARM GDG=3 VOLCOUNT=7
```

In this example, the default parameter values are used for most of the DETAILTAPEPARM parameters.

The following describes each of the DETAILTAPEPARM parameters:

- GDG=nnn

  Defaults to 7.

  The number of entries to be created in the GDG for each DETAIL tape data set.

  Overrides the TAPEPARM statement GDG=nnn.

- UNIT=unitname

  Defaults to the value specified on the TAPEPARM statement or to UNIT=3490 if UNIT=unitname is not specified on the TAPEPARM 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 TAPEPARM statement UNIT=unitname Value, the old-style TAPEUNIT statement, or both.

- 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.

- 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.

- 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.

- 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.

- 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.

- 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.

- 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.

- 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.

- 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.

- 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.
```

7.2.1.12.5 DETAILTAPESMSPARM and TAPEfffSMS Statements

In prefix.MICS.PARMS(cccOPS), the optional DETAILTAPESMSPARM and TAPEfffSMS 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.

TAPEfffSMS

The TAPEfffSMS statement is only permitted in prefix.MICS.PARMS(cccOPS). It is not supported in prefix.MICS.PARMS(JCLDEF). TAPEfffSMS 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 TAPEfffSMS 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.

7.2.1.12.6 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.

7.2.1.12.7 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:

cccGDGGN DAILY

If Incremental Update is active, add this:

INCRccc

___ 4. Submit prefix.MICS.CNTL(JCLGENU) to regenerate the JCL for DAILY and cccGDGGN.

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(cccGDGGN) 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 ... cccfffnn ;
   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;
```

7.2.2.12.8 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.

7.2.1.13 Sample VMCOPS

Following is a sample of required and optional statements that may be coded in prefix.MICS.PARMS(VMCOPS).

```
                      +-------------------------------------------------------------------------+
                      | INSTALLATION PREPARATION WORKSHEET: Options Definitions                  |
                      |                                                                         |
                      | PARMS Library Member is VMCOPS                                          |
                      +-------------------------------------------------------------------------+
                      |                                                                         |
                      | VMC PROCESSING OPTIONS:                                                 |
                      |                                                                         |
                      |                                                                         |
System oriented options |   OPTIONS    _______(orgsysid)  ________(CPU model)  _______(CPU serial)  |
                      |                                                                         |
                      |   ALTCPU     _______(orgsysid)  ________(CPU model)  _______(CPU serial)  |
                      |                                                                         |
                      |   MON        _______(orgsysid)  ________(ddname)                        |
                      |                                                                         |
                      |   ACT        _______(orgsysid)  ________(ddname)                        |
                      |                                                                         |
                      |   VMACCT     _______(orgsysid)  ________(ddname)                        |
                      |                                                                         |
                      |   COMMON     _______(datatype)  ________(ddname)                        |
                      |                                                                         |
                      |                                                                         |
General unit options  |   COMMON     _______(datatype)  ________(ddname)                        |
                      |                                                                         |
                      |   SPIN       _______(filename)  DAYS ________   CYCLES _______          |
                      |                                                                         |
                      |   SPIN       _______(filename)  DAYS ________   CYCLES _______          |
                      |                                                                         |
                      |   VERIFY     _______(ABORT|NOABORT)                                     |
                      |                                                                         |
                      |   WORK       _______ n data_set_allocation_parameters                   |
                      |   MULTWORK|NOMULT  fff ... fff                                         |
                      |   RESTART    _______(YES|NO)                                           |
                      |                                                                         |
                      |   INCRUPDATE _______(YES|NO)                                           |
                      |   INCRDB     _______(PERM/TAPE/DYNAM)                                   |
                      |   INCRDETAIL _______ data_set_allocation_parameters                     |
                      |   INCRDAYS   _______ data_set_allocation_parameters                     |
                      |   INCRCKPT   _______ data_set_allocation_parameters                     |
                      |   INCRSPLIT  _______(YES|NO)                                           |
                      |                                                                         |
                      |   TAPELXP            <specification overrides>                          |
                      |   TAPELXPSMS         <SMS parameter overrides>                          |
                      |                                                                         |
                      |   DETAILTAPEPARM     <unit specification overrides>                     |
                      |   DETAILTAPESMSPARM  <unit SMS specification overrides>                 |
                      |                                                                         |
                      +-------------------------------------------------------------------------+
                      | ....5...10...15...20...25...30...35...40...45...50...55...60...65...70..  |
                      +-------------------------------------------------------------------------+
```

7.2.1.14 VMCOPS and INPUTVMC Examples

Here are examples of VMCOPS and INPUTVMC statements.

```
 Example one
 -----------

  This example identifies multiple CPUs for multiple
  ORGSYSIDs for both account and monitor data data which will
  be processed into separate DDs in the DAY080 update step
  JCL identified in the ACT, VMACCT and MON statements.

  VMCOPS:

   OPTIONS    SYSA  2098  01BFA5
     ACT      SYSA  ACTSYSA
     VMACCT   SYSA  VMASYSA
     MON      SYSA  MONSYSA

   OPTIONS    SYSB  2817  02BFA5
     ACT      SYSB  ACTSYSB
     MON      SYSB  MONSYSB

   OPTIONS    SYSC  2827  04CD13
     ACT      SYSC  ACTSYSC
     MON      SYSC  MONSYSC

  INPUTVMC:

  //ACTSYSA  DD DISP=SHR,DSN=HLQ.ACCOUNT.DATA.SET.SYSA
  //VMASYSA  DD DISP=SHR,DSN=HLQ.VMACCNT.DATA.SET.SYSA
  //ACTSYSB  DD DISP=SHR,DSN=HLQ.ACCOUNT.DATA.SET.SYSB
  //ACTSYSC  DD DISP=SHR,DSN=HLQ.ACCOUNT.DATA.SET.SYSC
  //MONSYSA  DD DISP=SHR,DSN=HLQ.MONITOR.DATA.SET.SYSA
  //MONSYSB  DD DISP=SHR,DSN=HLQ.MONITOR.DATA.SET.SYSB
  //MONSYSC  DD DISP=SHR,DSN=HLQ.MONITOR.DATA.SET.SYSC

 Example two
 -----------

  This example identifies multiple CPUs for two ORGSYSIDs
  for monitor data using the ALTCPU statement. Data is
  read into separate DDs in the DAY080 update step JCL
  identified in the MON statement.

  VMCOPS:

     OPTIONS   SYSA  2098    01B415
     ALTCPU    SYSA  2098    02B415
     ALTCPU    SYSA  2098    03B415
```

```
   ALTCPU    SYSA  2098    04B415
   ALTCPU    SYSA  2818    00B6B7
   ALTCPU    SYSA  2818    02B5B7
     MON     SYSA  MONSYSA

   OPTIONS   SYSB  2098    06D325
   ALTCPU    SYSB  2098    05D325
   ALTCPU    SYSB  2098    04D325
   ALTCPU    SYSB  2098    02D325
   ALTCPU    SYSB  2817    01D4C7
   ALTCPU    SYSB  2817    04D6C7
     MON     SYSB  MONSYSB

 INPUTVMC:

 //MONSYSA  DD DISP=SHR,DSN=HLQ.MONITOR.DATA.SET.SYSA
 //MONSYSB  DD DISP=SHR,DSN=HLQ.MONITOR.DATA.SET.SYSB

Example three
-----------

 This example identifies multiple CPUs for two ORGSYSIDs
 for monitor data using the ALTCPU statement. Data is read
 into one DD statement in the DAY080 update step JCL
 identified in the COMMON statement.

 VMCOPS:

   COMMON MON MONCOMM
   OPTIONS   SYSA  2098    01B415
   ALTCPU    SYSA  2098    02B415
   ALTCPU    SYSA  2098    03B415
   ALTCPU    SYSA  2098    04B415
   ALTCPU    SYSA  2818    00B6B7
   ALTCPU    SYSA  2818    02B5B7
     MON     SYSA  *

   OPTIONS   SYSB  2098    06D325
   ALTCPU    SYSB  2098    05D325
   ALTCPU    SYSB  2098    04D325
   ALTCPU    SYSB  2098    02D325
   ALTCPU    SYSB  2817    01D4C7
   ALTCPU    SYSB  2817    04D6C7
     MON     SYSB  *

 INPUTVMC:

 //MONCOMM  DD DISP=SHR,DSN=HLQ.MONITOR.DATA.SET.COMM
```

7.2.2 Component-Specific JCL Information (INPUTVMC)

This section shows you how to specify the database unit level JCL parameters required to install the CA MICS Analyzer Option for VM/CMS. This product processes the following types of input:

- VM Monitor data is stored by an application program (for example, MONWRITE) as either a CMS disk file or tape file
- Velocity monitor data is stored as a CMS disk file
- CP account data is stored as a CMS disk file
- CA VM:Account data is stored as a CMS disk file

These files must be transported to the z/OS system that runs CA MICS. The VM Data Transfer Program is available to assist you in moving the data from your VM system to your z/OS system.

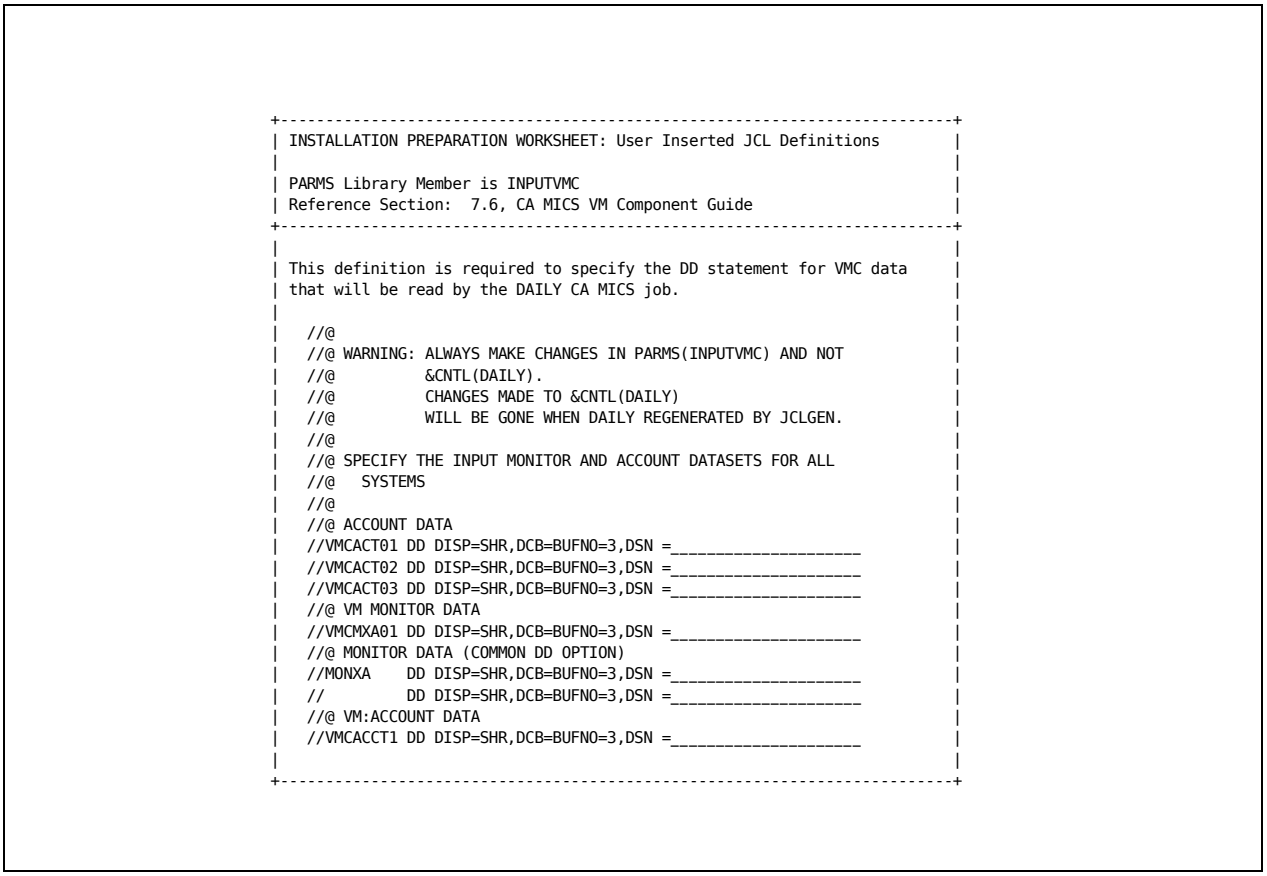
The input DD statements, which define the VM data files area, are supplied in the INPUTVMC member of the prefix.MICS.PARMS library and referenced by the MON, ACT, VMACCT, and COMMON statements of VMCOPS. A worksheet for preparing the INPUTVMC member is provided in Figure 7-5.

It is likely that you will have data from multiple systems as input to the VM tracking update process. To provide for account data from multiple systems, supply one DD statement per system using the ddnames that are specified in VMCOPS. To provide for monitor data from multiple systems, supply one DD statement per system using the ddnames that are specified in VMCOPS or provide the single ddname that is specified on the COMMON statement in VMCOPS, and concatenate the data sets.

For example, in this JCL, the unit processes CP account data from SYS1 and SYS2; and VM monitor data from SYS1, SYS2, and SYS3. The monitor data from SYS2 and SYS3 is brought in using a single ddname, which is specified in VMCOPS. In addition, VM monitor data is brought in from a fourth system, SYS4. The DD statements read as follows:

```
//VMCACT01 DD DISP=SHR,DSN=VM.SYS1.ACCOUNT.DATA
//VMCACT02 DD DISP=SHR,DSN=VM.SYS2.ACCOUNT.DATA
//VMCMON01 DD DISP=SHR,DSN=VM.SYS1.MONITOR.DATA
//VMCMONXX DD DISP=SHR,DSN=VM.SYS2.MONITOR.DATA
// DD DISP=SHR,DSN=VM.SYS3.MONITOR.DATA
//VMCMXAXX DD DISP=SHR,DSN=VM.SYS4.MONITOR.DATA
```

```
     Note:  Due to the text substitution process used by
            JCLGEN to produce //*, /*, and & in the
            JCL produced by JCLGEN, the following symbols
            must be used:

            //@     produces    //*
            /@      produces    /*
            ?       produces    &


DCB PARAMETERS

DCB record description parameters are required only when data
is provided to the DAY080 step by unlabeled tape.

VM monitor data has LRECL=4092 and RECFM=VB.  If the tape is
created by the VM monitor, specify BLKSIZE=4096.
Otherwise, use the blocksize used by the creating
program.  If blocksize cannot be determined, you can use a
BLKSIZE=32756.

VM monitor data that is decomposed by the CA MICS VM Data Transfer
Program has LRECL=8188 and RECFM=VB.

Velocity monitor data that is created by zWRITE usually has a
LRECL=1248 and RECFM=VB.

CP account data and CA VM:Account data is not typically
brought into CA MICS by tape. If an unlabeled tape is used,
the LRECL for CP account is 80 and the LRECL for CA
VM:Account data is 240.  Use the blocking
information that is specified by the creating program for each data
source.

PROVIDING DATA TO CA MICS

Getting the data to be processed each day into the CP account
and VM monitor data sets you specify is an operational
responsibility of the CA MICS System Administrator.  The
preferred methods are by use of the VM Data Transfer Program
or some similar facility, or by standard label tape.
Guidelines are provided for transporting the VM data files to
z/OS in the chapter named Processing.

Use the CA MICS VM Data Transfer Program to
decompose the data from MONWRITE format to CA MICS readable
format if you want to process VM Monitor data in CA MICS.
```



```
+-------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET: User Inserted JCL Definitions        |
|                                                                         |
| PARMS Library Member is INPUTVMC                                        |
| Reference Section:  7.6, CA MICS VM Component Guide                     |
+-------------------------------------------------------------------------+
|                                                                         |
| This definition is required to specify the DD statement for VMC data    |
| that will be read by the DAILY CA MICS job.                             |
|                                                                         |
|   //@                                                                   |
|   //@ WARNING: ALWAYS MAKE CHANGES IN PARMS(INPUTVMC) AND NOT           |
|   //@          &CNTL(DAILY).                                            |
|   //@          CHANGES MADE TO &CNTL(DAILY)                             |
|   //@          WILL BE GONE WHEN DAILY REGENERATED BY JCLGEN.           |
|   //@                                                                   |
|   //@ SPECIFY THE INPUT MONITOR AND ACCOUNT DATASETS FOR ALL            |
|   //@   SYSTEMS                                                         |
|   //@                                                                   |
|   //@ ACCOUNT DATA                                                      |
|   //VMCACT01 DD DISP=SHR,DCB=BUFNO=3,DSN =_____________________         |
|   //VMCACT02 DD DISP=SHR,DCB=BUFNO=3,DSN =_____________________         |
|   //VMCACT03 DD DISP=SHR,DCB=BUFNO=3,DSN =_____________________         |
|   //@ VM MONITOR DATA                                                   |
|   //VMCMXA01 DD DISP=SHR,DCB=BUFNO=3,DSN =_____________________         |
|   //@ MONITOR DATA (COMMON DD OPTION)                                   |
|   //MONXA    DD DISP=SHR,DCB=BUFNO=3,DSN =_____________________         |
|   //         DD DISP=SHR,DCB=BUFNO=3,DSN =_____________________         |
|   //@ VM:ACCOUNT DATA                                                   |
|   //VMCACCT1 DD DISP=SHR,DCB=BUFNO=3,DSN =_____________________         |
|                                                                         |
+-------------------------------------------------------------------------+
```

Figure 7-5. INPUTVMC JCL Definition Worksheet

7.2.3 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)

7.2.3.1 Data Retention Specifications (FILE Statements)

This section describes cycle (data retention) definitions that are required, discusses how to determine the data cycles required, and provides instructions for completing the FILE statements in the PARMS member DBMODEL.

DATABASE DATA RETENTION WORKSHEET DEFINITION

Figure 7-6 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 more information, see Section 2.3.3.2.1.5 in the PIOM.

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:

- 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.

- 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.

VM COMPONENT DATA RETENTION DEFINITION

The CA MICS Analyzer Option for VM/CMS contains the VMC and VMX Information Areas. The FILE statements shown below specify the default online and archive history retention limits for each of these files in t@fi-archese information areas.

A complete description of the FILE statement is provided in section 2.3.4.1, Preparing the Modeling Input, in the PIOM.

```
+-------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET: Database Data Retention Definitions  |
|                                                                         |
| PARMS Library Member is DBMODEL                                         |
|                                                                         |
| Reference Sections: 7.2.3.1                                             |
+------+-------------------------------------------------+-----------------+
|      |           Online Database Retention             |  Archive Cut-Off  |
| File |DETAIL    DAYS   WEEKS   MONTHS  YEARS  TABLES | WEEKS    MONTHS   |
| Name |                                                 |                 |
+------+-------------------------------------------------+-----------------+
|VMCVXC| __(01)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMCCMU| __(10)  00(00)  00(00)  __(13) __(2)   00(00) |___ (001) ___ (001)|
|VMCDAC| __(10)  00(00)  00(00)  __(13) __(2)   00(00) |___ (001) ___ (001)|
|VMCVRA| __(10)  00(00)  00(00)  __(13) __(2)   00(00) |___ (001) ___ (001)|
|VMCNTU| __(10)  00(00)  00(00)  __(13) __(2)   00(00) |___ (001) ___ (001)|
|VMCNDT| __(10)  00(00)  00(00)  __(13) __(2)   00(00) |___ (001) ___ (001)|
+------+-------------------------------------------------+-----------------+
|VMXSCF| __(01)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXPCF| __(01)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXSGP| __(10)  __(10)  __(02)  __(01) __(0)   00(00) |___ (001) ___ (001)|
|VMXSPP| __(03)  __(01)  __(01)  __(01) __(0)   00(00) |___ (001) ___ (001)|
|VMXDCF| __(01)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXXDV| __(01)  __(01)  00(00)  __(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMXXDS| __(01)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXACF| __(01)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXCPV| __(03)  __(01)  __(01)  __(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMXVXF| __(01)  00(00)  00(00)  __(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXVXU| __(01)  __(01)  00(00)  __(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMX_VU| __(01)  __(01)  00(00)  __(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMXVXT| __(01)  __(01)  00(00)  __(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMX_VT| __(01)  __(01)  00(00)  __(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMXCHC| __(03)  00(00)  __(01)  __(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMXBPG| __(01)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXXSS| __(01)  00(00)  00(00)  __(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMXXSF| __(01)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXSTP| __(03)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXCDI| __(03)  00(01)  00(00)  00(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMXCDP| __(03)  00(01)  00(00)  00(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMXLPR| __(03)  00(01)  00(00)  00(01)  0(0)   00(00) |000 (000) 000 (000)|
|VMXLCP| __(03)  00(00)  00(00)  00(00)  0(0)   00(00) |000 (000) 000 (000)|
|VMXSCX| __(01)  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 7-6. Database Data Retention Worksheet

7.2.4 CA MICS JCL Parameters (JCLDEF)

There are a few required parameters, and many optional ones, which you specify in prefix.MICS.PARMS(JCLDEF) to guide the JCLGEN process as it generates CLISTs and jobs tailored for your processing environment. This member contains one statement for each option defined.

The SMFRECORDING statement used by the JCLGEN process to build the Daily CA MICS job must not specify VMC, because no VM data is available from this source.

7.2.5 CA MICS System Identification (SYSID)

Each VM system that providing data for CA MICS must
be defined in prefix.MICS.PARMS(SYSID). Some coding
for SYSID differs between a VM system and a z/OS system.
The following section is designed to address these differences.
For details on creating prefix.MICS.PARMS(SYSID), see
section 2.3 in the PIOM.

The SMF SYSID parameter that is specified in SYSID is the original
sysid (orgsysid) referenced by VMCOPS for the VM system.
This orgsysid is an arbitrary assignment by your site as a name for
the VM system.

The CPU MODEL specified in SYSID is compared with the CPU
Model that is specified for the system in VMCOPS. If a significant
portion of the model is not consistent between the two
members, a warning message is issued by VMCPGEN.

The RMF FACTOR is not used by the CA MICS Analyzer Option for
VM/CMS. Enter the RMF FACTOR as 1.

Enter the LOGICAL SYSID, MIPS RATE, and SYSTEM NAME
as described in the PIOM, Sections 2.3.1.9 and
2.3.2.2.

The System Description Worksheet is shown in Figure 7-7 for
reference purposes. For installation purposes, see Figure
2-9 in the PIOM.

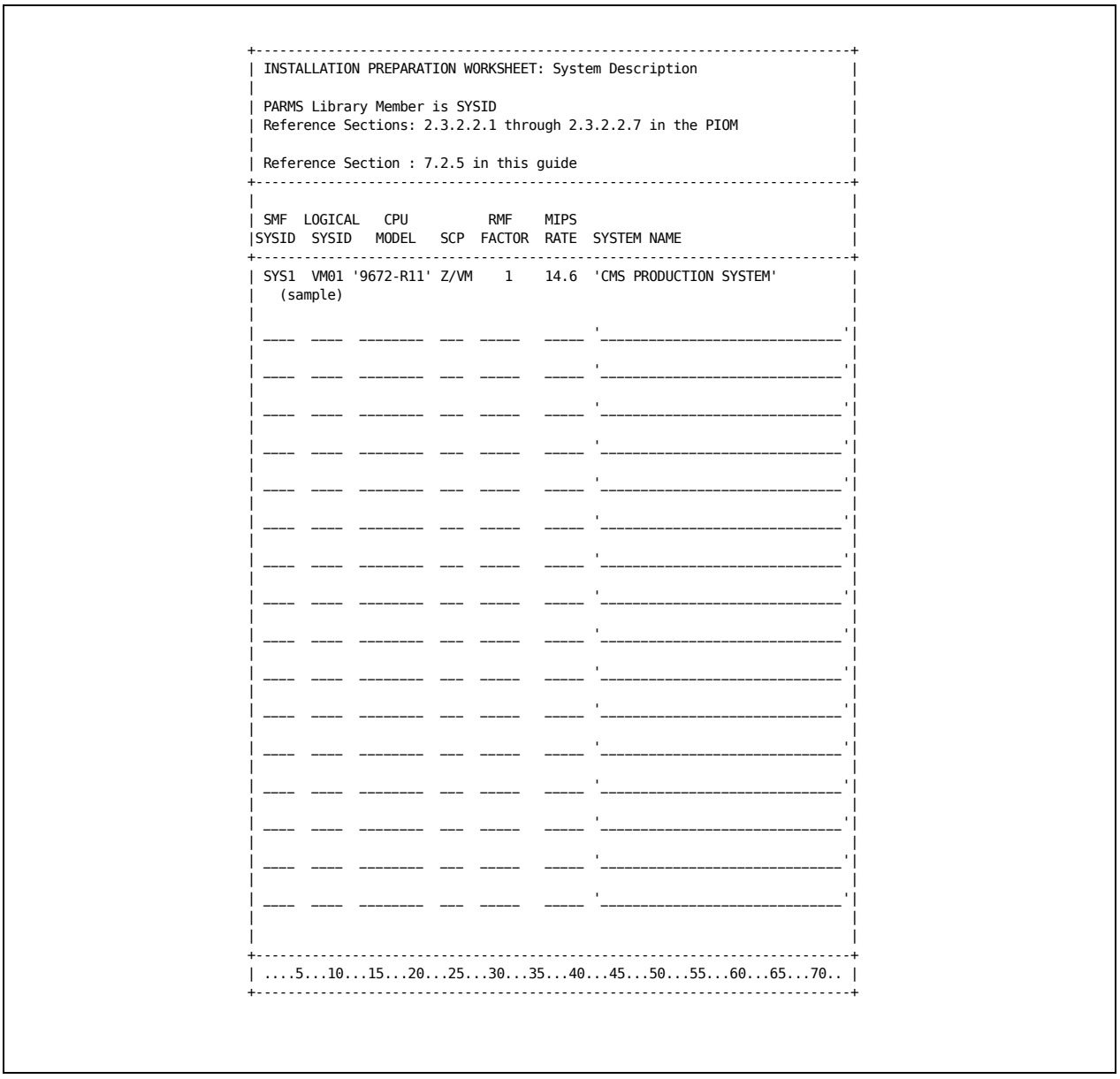


Figure 7-7. System Description Worksheet

Chapter 8: INSTALLATION

Install the CA MICS Analyzer Option for VM/CMS by following the instructions in the PIOM.

SYSID, SITE, ZONE, PRFGP, and other parameter files that are used by more than one CA MICS product are defined in Chapter 2 of the PIOM. Section 3.8 of the PIOM provides step-by-step checklists that must be followed to ensure successful installation. Chapter 7 provides instructions for correctly defining parameters specific to this Analyzer.

Chapter 9: PROCESSING

This chapter describes the daily processing flow of the CA MICS Analyzer Option for VM/CMS, the exit points available in the DAILY update, and considerations for running the VM DAILY update.

This section contains the following topics:

9.1 Processing Considerations (see page 567)
9.2 Analyzer Daily Update Flow (see page 568)
9.3 DAY080 Physical Structure (see page 590)
9.4 Daily Update Physical Structure (see page 591)
9.5 Account Physical Structure (see page 592)
9.6 VM Monitor Processing Physical Structure (see page 594)

9.1 Processing Considerations

The DAY080 step (of the Daily Update job) processes VM data. The input data may reside in any sequential file, as defined by INPUTVMC.

CP Account Data

CP account data consists of 80-byte fixed-length records. This data must be read from the VM spool. Then, it must be transported to your z/OS system, which will be running CA MICS. The VM Data Transfer Program will perform this function for you.

VM Monitor Data

In VM, the monitor data is written directly to disk or tape by IBM's MONWRITE program or another application designed to interface with the *MONITOR system service. VM Monitor data can be transferred to z/OS using the CA MICS VM Data Transfer Program. VM Monitor data must be extracted from the MONWRITE format by the VM Data Transfer Program.

You may want to exclude portions of the raw data from processing, particularly if event data is being collected. Data for an entire domain, or particular record type within a domain, can be excluded by using the VM Data Transfer Program or by using the USRSEL exit provided in CA MICS. For example, you may choose to exclude data from the domain 2 (Scheduler) record, types 1 through 6 and 8 through 10; the domain 4 (User) record, types 8 through 10; and the domain 7 (Seek) record.

Be careful when excluding records, because certain types of data are needed to properly process input. For example, Domain 1 (Monitor) data cannot be excluded, because it is necessary for obtaining fundamental configuration data.

Chapter 6 includes a more detailed discussion of monitor collection and data inclusion/exclusion considerations for VM.

U300 ABEND

The U300 abend in CA MICS can be issued from two locations within the CA MICS Analyzer Option for VM/CMS Daily update:

1. VMCACTFI - at the end-of-input, for all systems providing CP account data

2. VMCMXAFI - at the end-of-input, for all systems providing VM Monitor data

If any of the input processes (CP account or VM monitor) encounters a situation in which no data or insufficient data has been read and kept, a U300 abend occurs. BOTH input processes must read and keep enough data to satisfy the test. For more information, see the Abnormal Terminations documentation in chapter 4 in the PIOM.

9.2 Analyzer Daily Update Flow

This section describes the daily processing flow for the CA MICS Analyzer Option for VM/CMS and its relation to the locations of the VMC User Exit Routines. For specific inquires, see the source code (program modules DYVMCFMx, VMCACTxx, and VMCMXAxx) or contact CA Technical Support at http://ca.com/support.

The update process is divided into seven distinct processing phases:

- Input Raw Account Phase
- Account Post-Processing Phase
  - develop derived elements
  - interface with Installation Accounting
- Input Raw VM Monitor Phase
- VM Monitor Post-Processing Phase
  - develop interval data values
  - develop derived elements
- Consolidation Phase
  - joining of exceptions and log files
- Database Timespan Update Phase
- File Aging Phase

Note: Throughout this chapter, standard VM logic is represented within lined boxes and product exits are represented within starred boxes.

The following sections describe the general logic flow in each phase and the location of standard user exit routines:

1 - Input Raw Account Data
2 - Account Post-Processing
3 - Input Raw VM Monitor Data
4 - VM Monitor Post-Processing
5 - Consolidation
6 - Database Timespan Update
7 - File Aging

9.2.1 Input Raw Account Data

CP account data is input from data sets defined by INPUTVMC. These data sets contain 80 byte fixed-length records.

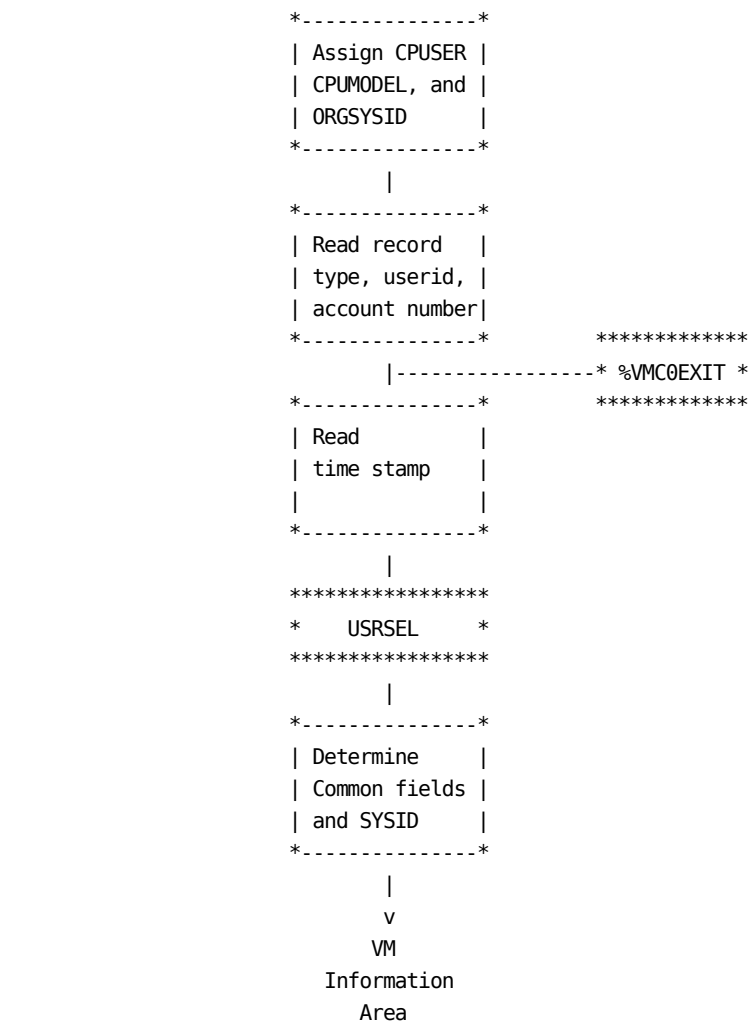
ORGSYSID

There is no information within the data itself that allows the determination of the system of origin, so the CA MICS ORGSYSID is assigned to the data by association with the input DDNAME. This information is specified in VMCOPS, and is provided to the DAY080 step in the %VMCLUPA macro generated by VMCPGEN.

TIMESTAMPS

CP account data provides time and date information for the time at which the account record was created. The standard CP account records that provide some measurement of elapsed time also provide a duration that is used by the CA MICS Analyzer Option for VM/CMS to determine the start time for the record. This method is not used by VCNA and VSCS to create the type 07 record, which has both start and end times in the data. In this case, duration is calculated by the CA MICS Analyzer Option for VM/CMS.

Refer to Figure 9-1 for the logic flow for the Input Raw Account Data. The first part of this figure illustrates the common handling of each record. Part 2 illustrates processing specific to each record type.



This %VMC0EXIT exit is only taken for record type C0 or unrecognized record types. It is used to identify C0 records produced by RSCS, and in concert with other VMC Account exits.

Figure 9-1. Input Raw Account Data (Part 1 of 2)

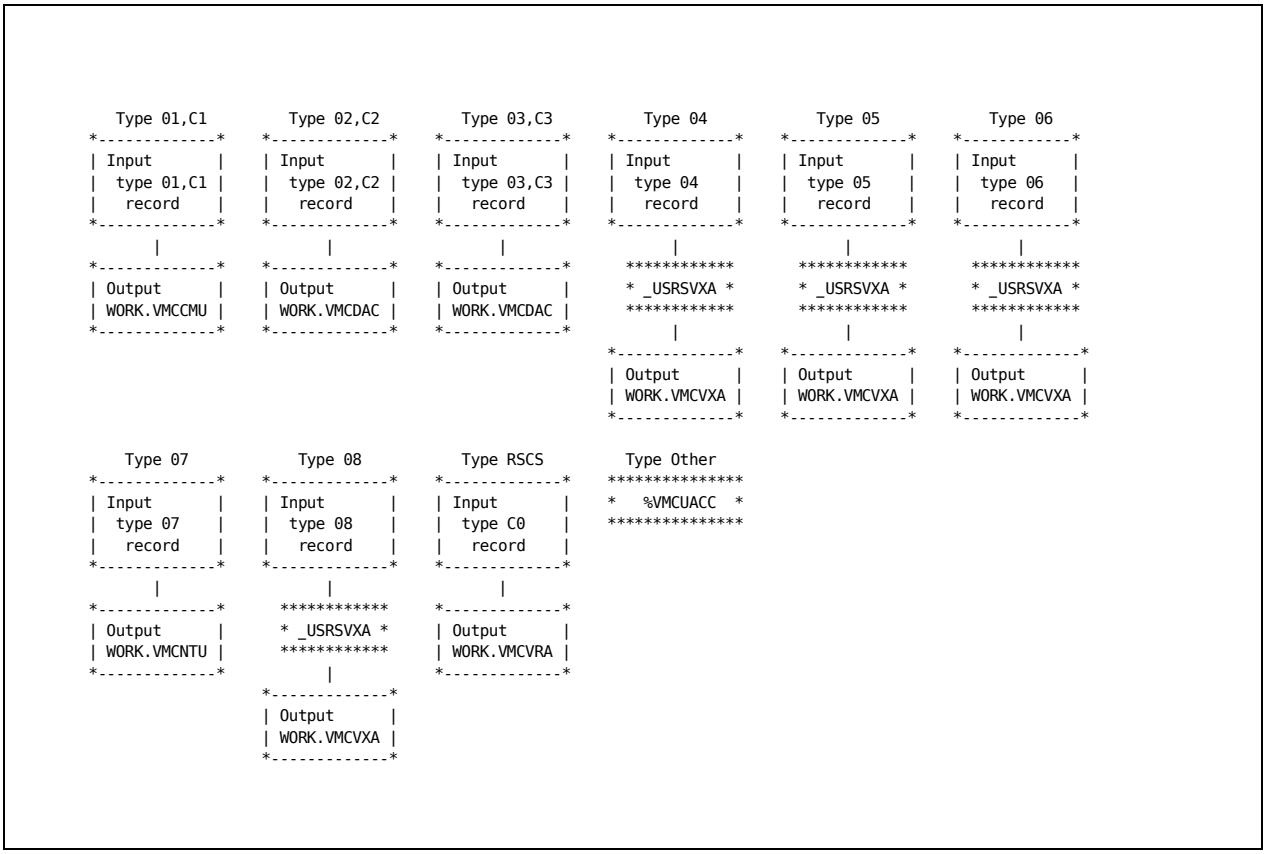


Figure 9-1. Input Raw Account Data (Part 2 of 2)

9.2.2 Account Post-Processing

Post-processing for CP account data is used to:

- Perform calculations on the file elements
- Call VMCACRT and CA MICS Accounting and Chargeback
- Sequence the files
- Place the files into the DETAIL database

Refer to Figure 9-2 for an illustration of processing for each file in the Account Post-Processing section of DAY080.

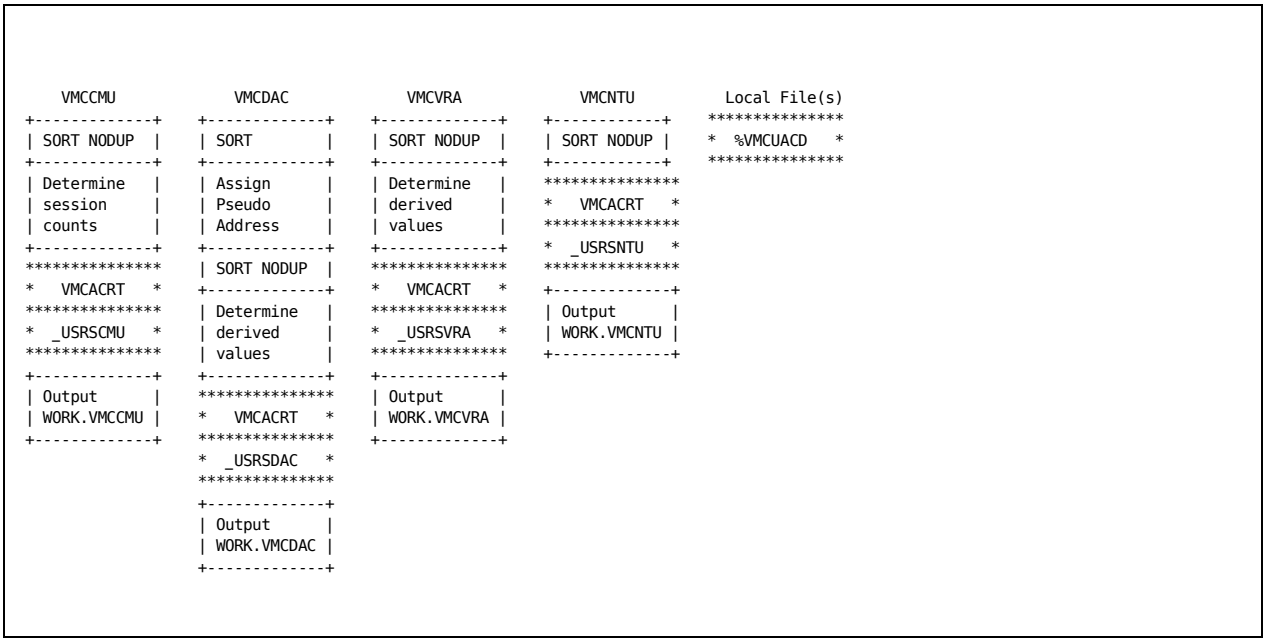


Figure 9-2. Account Post-Processing

VMCCMU: The elements CMUNRSES and CMUACSES are determined.

VMCDAC: Due to the manner in which the CP creates account data, it is possible to have duplicate data. For example, a user attaches two 5-cylinder 3380 TDISKS, executes a CP ACNT CLOSE DMKMID function, and executes an account number change. This function creates two 03 account records. If another CP ACNT function is performed, identical 03 records are created.

When SAS PROC SORT NODUP is run, it drops duplicate data. The CA MICS Analyzer Option for VM/CMS does not perform a PROC SORT NODUP on the VMCDAC file.

A device number is assigned to each observation so Number of Allocations (DACNRALC) and Number of Accounting Allocations (DACACALC) is reported correctly for each device that is online. Data elements Device Type (DEVTYPE) and Device Class (DEVCLASS) are determined by formats VMDEVTYP and VMDEVCLS in format routine VMCFMT by checking feature codes for some devices.

VMCVRA: The elements DEVTYPE and DEVCLASS are determined by means of the VMDEVTYP and VMDEVCLS formats in VMCFMT.

9.2.3 Input Raw VM Monitor Data

VM Monitor data comes from two sources: IBM's MONWRITE utility and Velocity Software's ESAWRITE utility. VM Monitor data is input from data sets defined by INPUTVMC. These data sets contain variable length records (up to 8188-bytes in length).

CONFIGURATION RECORDS

VM Monitor files contain a set of configuration records (Domain 1) generated when VM Monitor is started. These records provide static data for the system, such as processor and device configuration data, base measurement data, and the time the monitor was started. This information is critical for building the files in the VMX information area and maintaining data integrity for all VM Monitor base files. If a Monitor input file is read and the input data does not begin with the following Domain 1 configuration records, the DAY080 processing step will abend U997 with either VMC00063E, VMC00067E, VMC00069E, or VMC00300E warning messages displaying in the MICSLOG.

Domain/ Record	Description
01/01	Event Profile
01/04	System Configuration
01/05	Processor COnfiguration
01/09	Monitor Sample

If may be possible that other Domain 1 records such as the 1/14 Domain Detail monitor records may be present at the beginning of the input data. As long as the above mentioned records are in the first set or group of Domain 1 records, processing of the data should continue.

See Section 6.2.6, Configuration Record Requirements, for more information on configuration records. If the monitor input data begins with a 01/12 Start of Suspend record, see Section 6.2.4, Monitor Suspend Considerations and Handling.

See Appendix A for explanation of the above mentioned MICSLOG messages and reasons why these configuration records may be missing.

ORGSYSID, SYSID, CPUMODEL

The CA MICS ORGSYSID is assigned to the data based on the input DDNAME, by association with the CPU serial number specified in VMCOPS and read from the Domain 1, record 5 Record (Processor Configuration). The parameter generation process, VMCPGEN, creates the CA MICS variable GENSYSID that is assigned to ORGSYSID. ORGSYSID is passed to BASE MACRO %SYSID to create the CA MICS data elements CPUMODEL and SYSID. See Section 7.2.1 for more information about VMCOPS.

```
TIMESTAMPS

For files that have one observation per input record, the end
timestamp is taken directly from the input data (the time of
day clock stamp contained in each monitor record header).
For other files, the end timestamp is taken from the record
with the highest timestamp that contributes to the
observation.

The Interval Timestamp (VMCINTTS) is taken from the first
Domain 1/11 record of the previous sample interval.

HOST SUBCHANNEL ID

The source data for the VMXDVSID data element is defined as
four bytes in most input records.  It is input as '+2
VMXDVSID PIB2.', because the first two bytes always contain
X'0001'.  Additionally, leaving '0001' as is would make the
element difficult to use.  When an I/O Instruction (SSCH,
RSCH, and so on)  is issued, '0001' is a requirement of the
hardware.

VMXCHC FILE

This file contains up to 256 observations per interval.  A
way to reduce this number has been provided.  An array
(CHPID) is maintained that contains the count of devices
attached to each channel.  Any CHPID without a device is
represented by a zero or missing value.  This array can be
tested in the USRSCHC user exit:

    SKIP_REC = (CHPID(CHCHPID+1) > 0);

This can also be done automatically by setting the MACRO
variable CHP_SEL to YES in USER.SOURCE(#VMCEXIT), in which
case we will bypass all output logic for the file when an
observation for an orphan CHPID is detected.  See Figure 9-5
for the logic flow in the Input VM Monitor Data in DAY080.
Part 1 of Figure 9-3 illustrates processing performed for
Domain 1 monitor configuration records at initialization.
Parts 2 through 4 of Figure 9-3 illustrate processing
peculiar to each Domain/Record type.
```

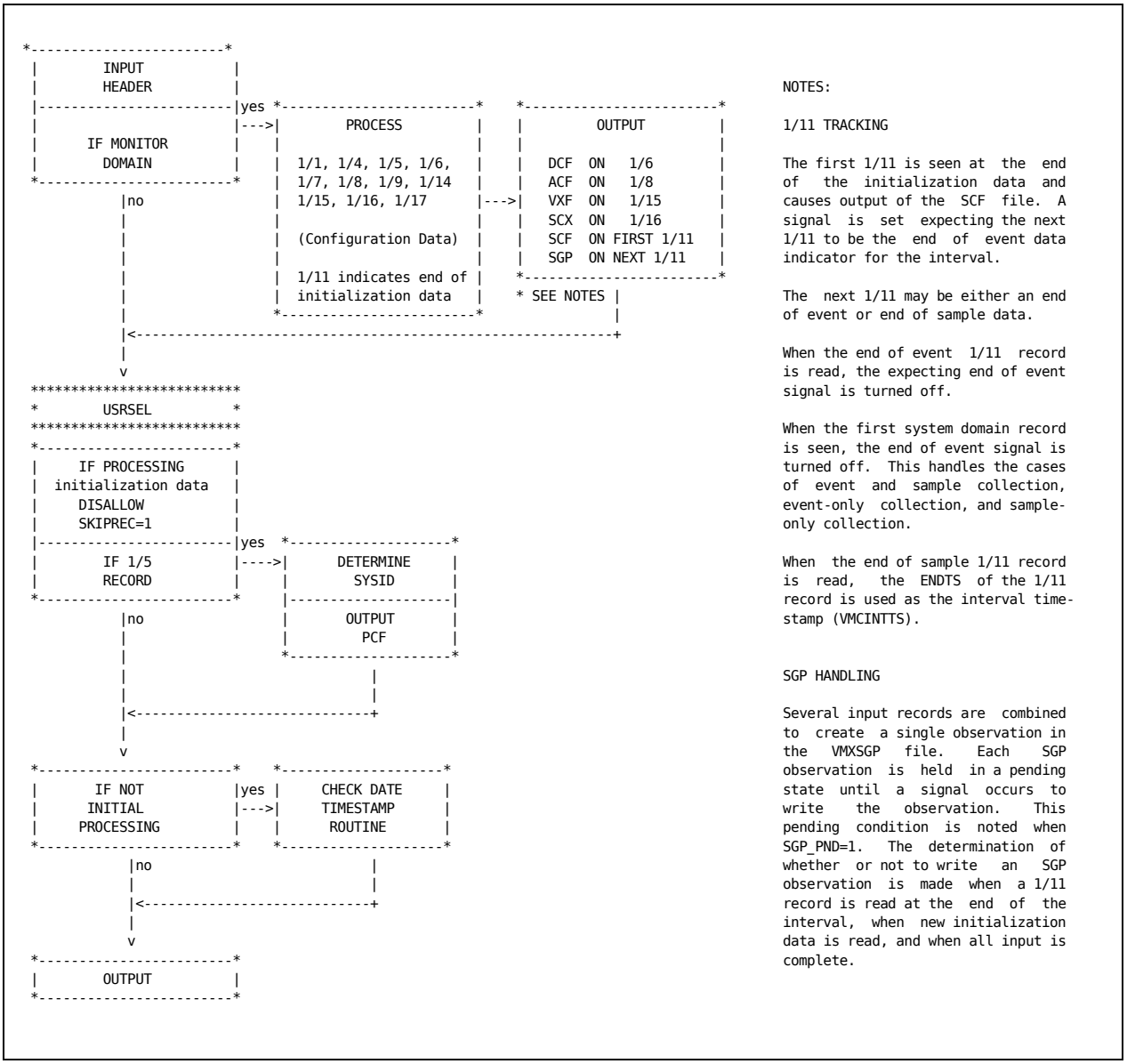


Figure 9-3a. Input VM Monitor Data (Part 1 of 4)

SYSTEM DOMAIN PROCESSING

INPUT RECORD	USER EXIT	OUTPUT FILE
0/1	USRSSPP	VMXSP001
0/2	USRSSPP	VMXSP002
0/4	USRSSPP	VMXSP004
0/5	USRSSPP	VMXSP005
0/11	USRSSPP	VMXSP011
0/13	USRSSPP	VMXSP013
0/9	USRSCHC	VMXCHC09
0/18	USRSCHC	VMXCHC18
* 1 observation per CHPID		
0/15, 0/16, 0/17	USRSLCP	VMXLCP00
* 0/15 is held for 0/16 and 0/17		
* 1 observation per logical CPU		
0/15, 0/16, 0/17	USRSLPR	VMXLPR00
* 0/15 is held for 0/16 and 0/17		
* 1 observation per logical partition		
0/3, 0/6, 0/7 0/8, 0/9, 0/10 0/12, 0/14, 0/19	USRSSGP	VMXSGP00
* held until 1/11		
* 1 observation per sample interval		

MONITOR DOMAIN PROCESSING

INPUT RECORD	USER EXIT	OUTPUT FILE
1/6	USRSDCF	VMXDCF00
1/8	USRSACF	VMXACF00
1/15	USRSVXF	VMXVXF00
1/16	USRSSCX	VMXSCX00
1/1, 1/4, 1/5 1/7, 1/9, 1/17	USRSSCF	VMXSCF00
* held until 1/11		
* 1 observation per monitor start		
1/12, 1/3	USRSVXC	VMCVXX00
* 1/12 held for 1/3		
1/2, 1/10	USRSVXC	VMCVXX00
1/11	USRSVXC	VMCVXX00
* if cols 13-14 contains EP or SP		

Figure 9-3b. Input VM Monitor Data (Part 2 of 4)

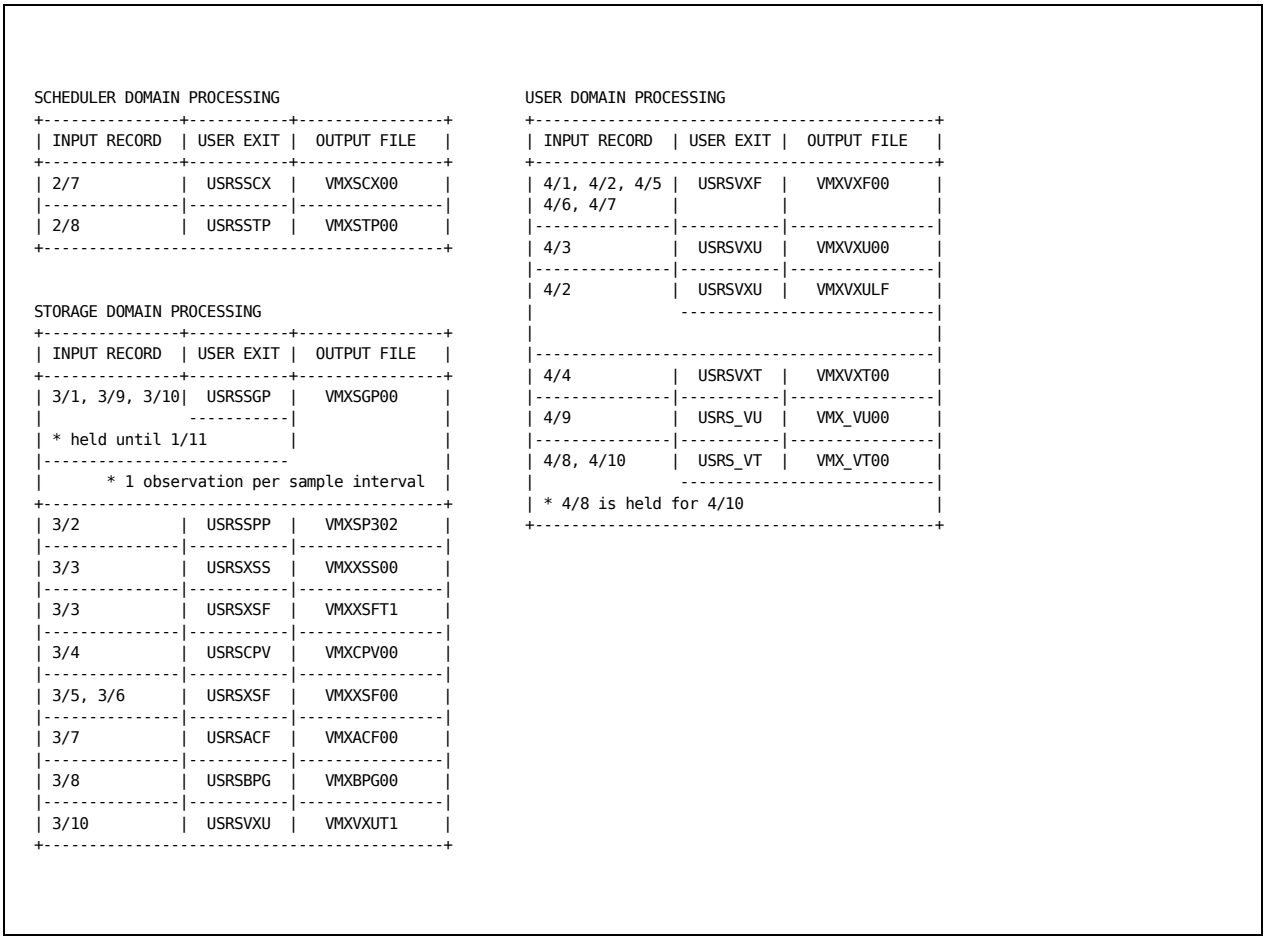
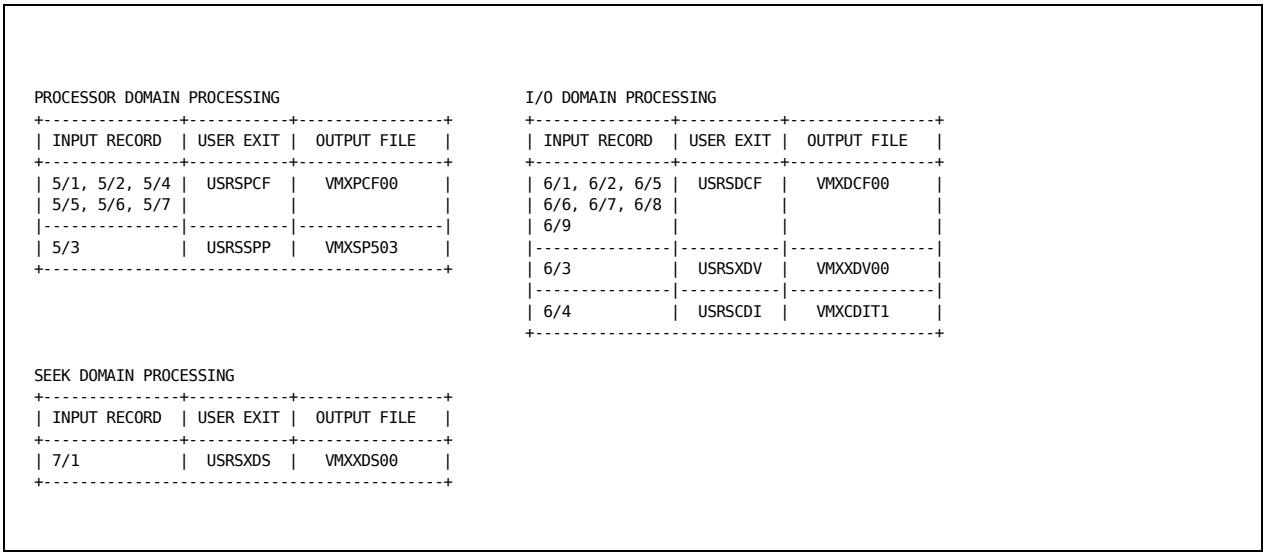


Figure 9-3c. Input VM Monitor Data (Part 3 of 4)



PROCESSOR DOMAIN PROCESSING

INPUT RECORD	USER EXIT	OUTPUT FILE
5/1, 5/2, 5/4	USRSPCF	VMXPCF00
5/5, 5/6, 5/7		
5/3	USRSSPP	VMXSP503

I/O DOMAIN PROCESSING

INPUT RECORD	USER EXIT	OUTPUT FILE
6/1, 6/2, 6/5	USRSDCF	VMXDCF00
6/6, 6/7, 6/8		
6/9		
6/3	USRSXDV	VMXXDV00
6/4	USRSCDI	VMXCDIT1

SEEK DOMAIN PROCESSING

INPUT RECORD	USER EXIT	OUTPUT FILE
7/1	USRSXDS	VMXXDS00

Figure 9-3d. Input VM Monitor Data (Part 4 of 4)

9.2.4 VM Monitor Post-Processing

INTERVAL VALUE DETERMINATION (DELTA PROCESSING)

Post-processing for VM Monitor data calculates interval values from cumulative counters provided by the CP.

The interval value is determined by subtracting cumulative counter fields from two consecutive monitor records.

CONFIGURATION RECORDS DURING POST-PROCESSING

Processor Configuration Record (Domain 1, Record 4)

The System Configuration (VMXSCF) file contains information from the System Configuration record. When the configuration record for a system is missing in the VMXSCF file, DAY080 abends with a U997 and MICSLOG message VMC00069E is displayed.

It may be possible that a monitor suspend event occurred that could cause the domain 1 record 4 to be missing. See Section 6.2.4 for more information on how to detect if the abend is a result of monitor suspend and what to do.

Processor Configuration Record (Domain 1, Record 5)

The Processor Configuration (VMXPCF) file contains information from the Processor Configuration record. When the configuration record for a processor is missing in the VMXPCF file, DAY080 abends with code U997 and MICSLOG message VMC00067E is displayed.

It should be investigated by your sites' VM system programmer why the domain 1 record 5 Processor Configuration records for the new VARIED ON processor are not being collected by the MONITOR.

It may be possible that a monitor suspend event occurred that could cause the domain 1 record 5 to be missing. See Section 6.2.4 for more information on how to detect if the abend is a result of monitor suspend and what to do.

If you determine that the missing domain 1 record 5 configuration record is not missing because of a monitor suspend event, then for a limited time only, the U997 abend can be prevented. You can code a SYSPARM='NODATA NOXACFG' on the EXEC card of the DAY080 step. The use of the NOXACFG parameter should be restricted. If used for an indefinite period of time, it can cause unpredictable results in your database. We highly recommend that NOXACFG be removed after the first successful daily run.

Do not code SYSPARM='NODATA NOXACFG' to override the U997 abend without first contacting CA Technical Support at http://ca.com/support. Have a copy of the MICSLOG and SASLOG ready before contacting them.

```
Monitor Sample (Domain 1, Record 9)
-----------------------------------

The Monitor Sample record contains the length of the interval
that is used to assign a value to Interval Length
(VMCINTVL).  VMCINTVL is used during the post-processing
phase to determine when delta processing should be done.  If
the Domain 1 Record 9 is missing in the input data, data
element VMCINTVL results in missing values because the
Monitor Sample record is missing in the input data.  This can
cause observations to be dropped and invalid delta values in
the CA MICS database during delta processing.

The DAY080 update-step abend with a U997 and MICSLOG message
VMC00063E is displayed if the Domain 1 record 9 is detected
missing.

It may be possible that a monitor suspend event occurred
that could cause the domain 1 record 9 monitor record to be
missing.  See Section 6.2.4 for more information on how to
detect if the abend is a result of monitor suspend and what
to do.

If you use Velocity Software's ESAMON and ESAWRITE monitoring
and data collection software, ensure that the CA MICS extract
file from ESAWRITE is selecting the domain 1 record 9 monitor
record.

If you determine that the missing domain 1 record 9 sample
configuration record is not missing because of a monitor
suspend event, or is not selected for the CA MICS extract as
a last resort, the U997 abend can be overridden. You can code
a SYSPARM='NODATA NOD1R9' on the EXEC card of the DAY080
step.  The use of the NOD1R9 parameter should be restricted.
Be aware when you use this override that it can cause
observations to be dropped in some database files and invalid
data element values during delta processing.  We highly
recommend that NOD1R9 be removed after the first successful
daily run.
```

Do not code SYSPARM='NODATA NOD1R9' to override the U997 abend without first contacting CA Technical Support at http://ca.com/support. Have a copy of the MICSLOG and SASLOG ready before contacting them.

Figure 9-4 illustrates VM Monitor post-processing by file.

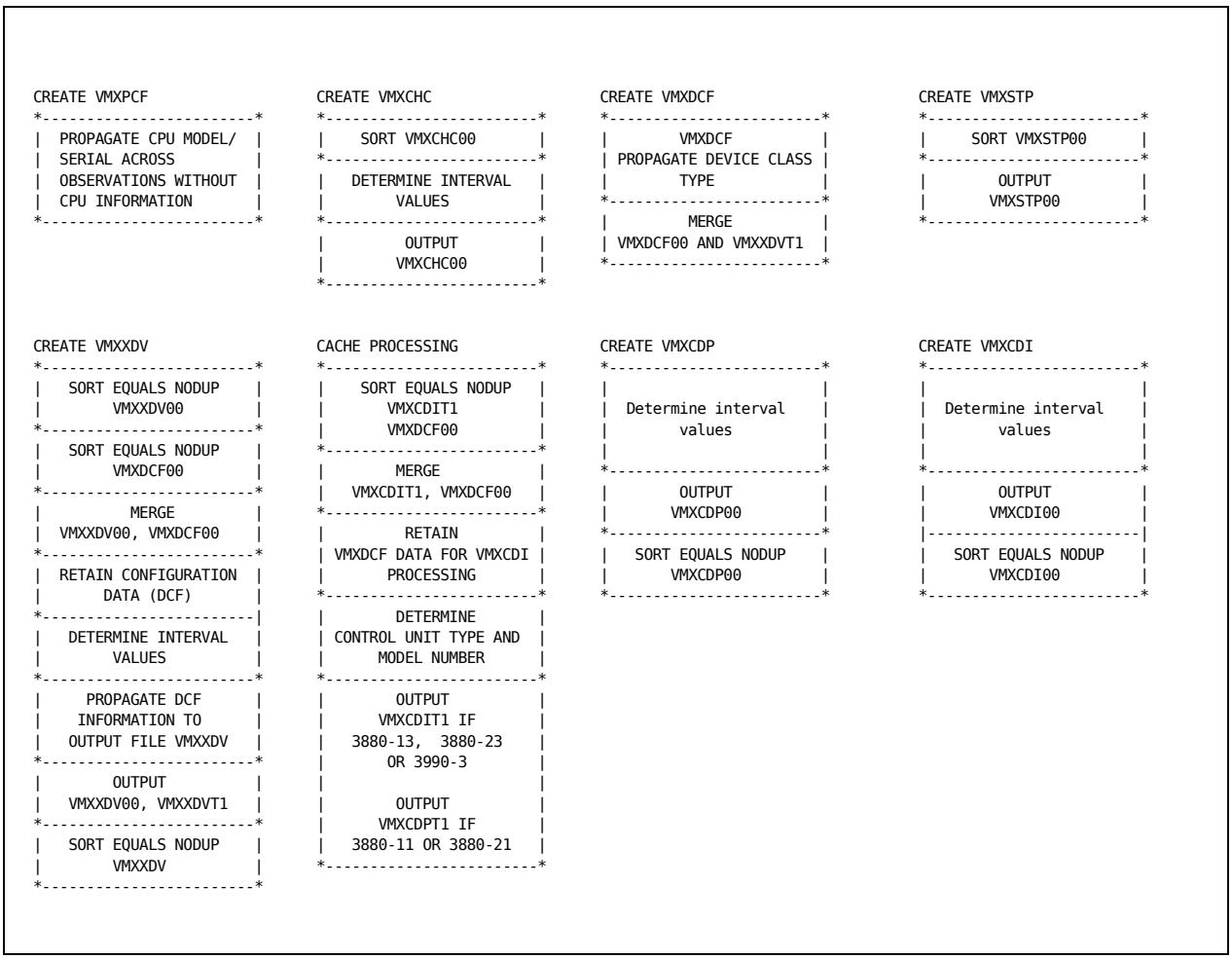


Figure 9-4a. VM Monitor Post Processing (Part 1 of 6)

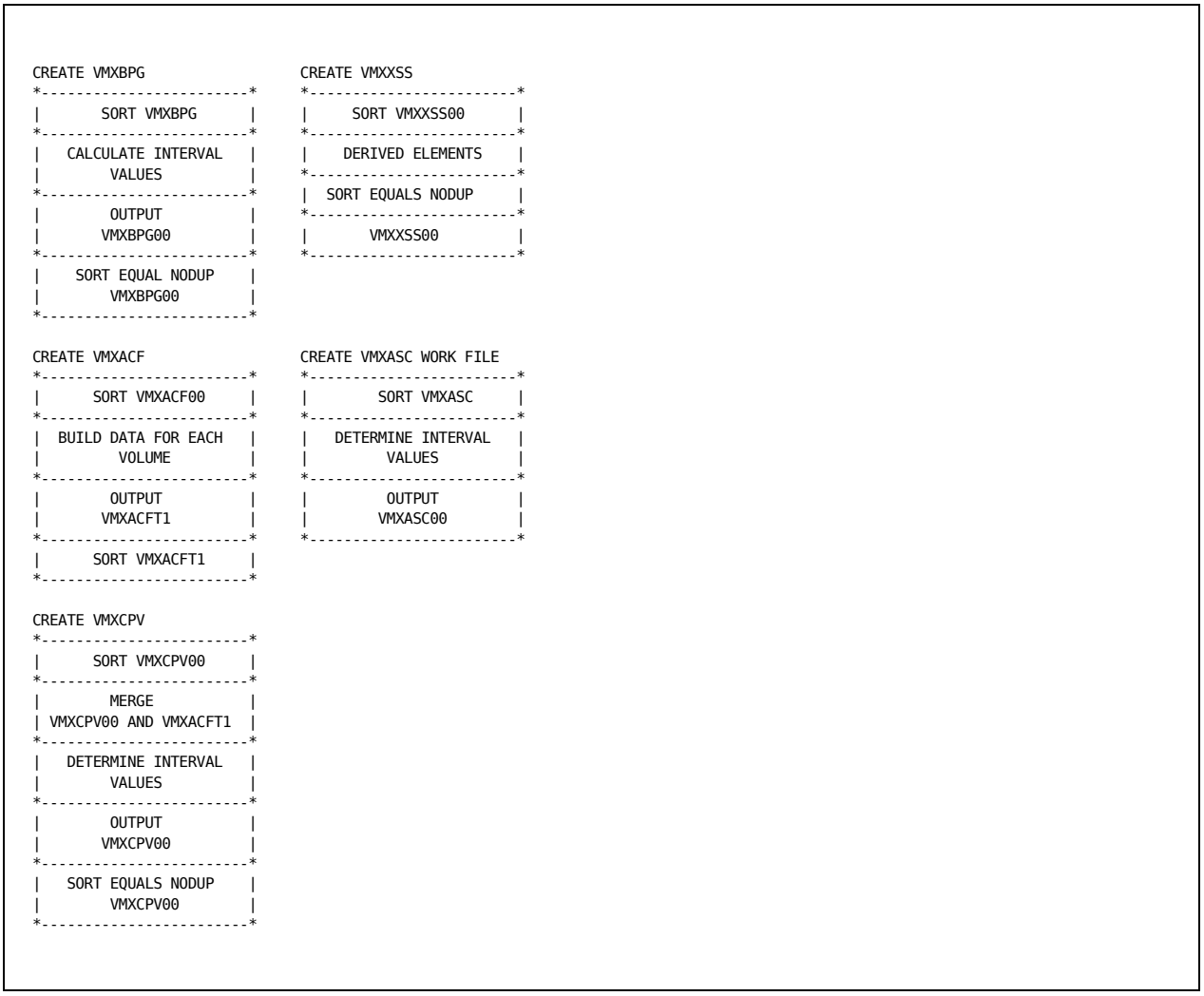


Figure 9-4b. VM Monitor Post Processing (Part 2 of 6)

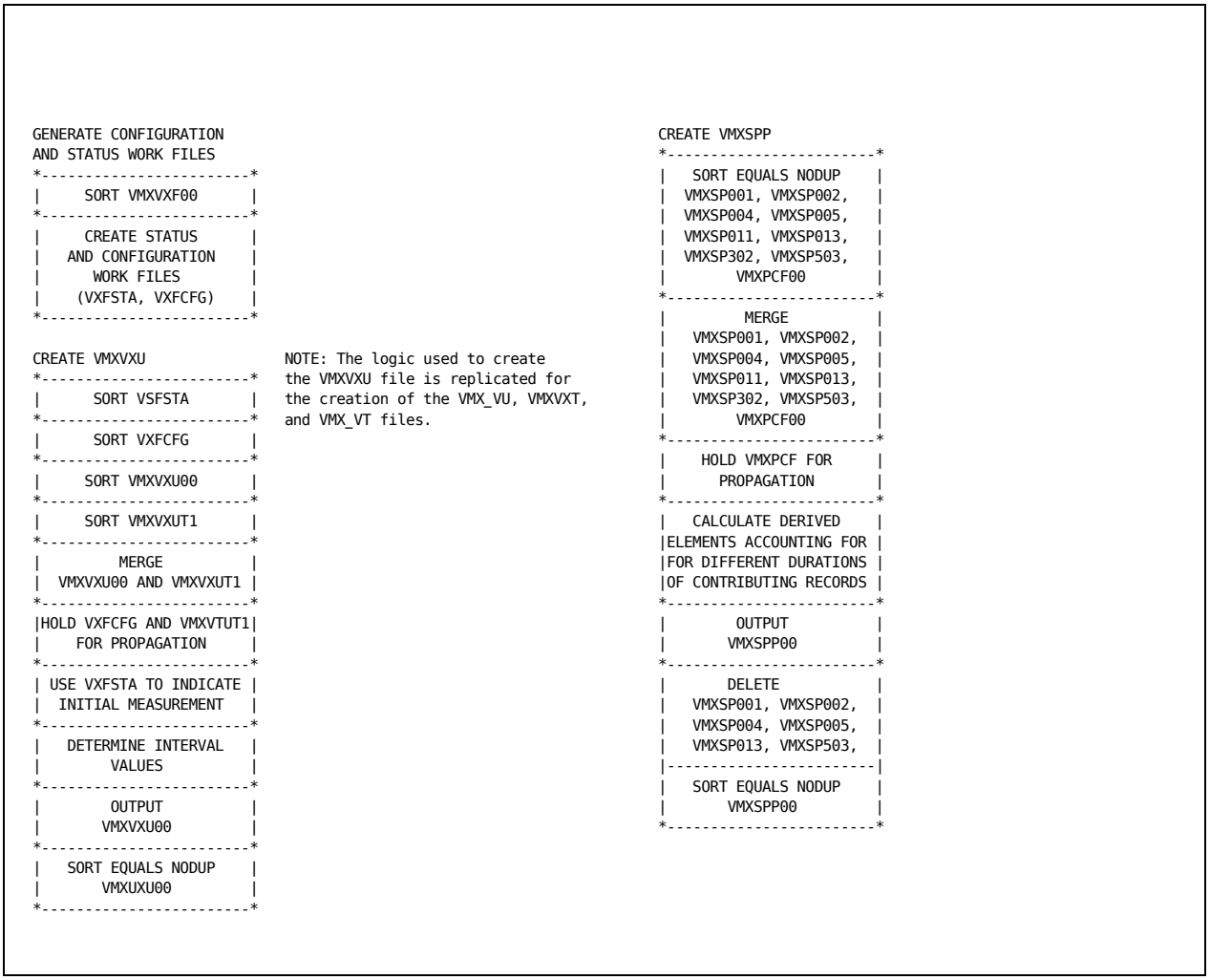


Figure 9-4c. VM Monitor Post Processing (Part 3 of 6)

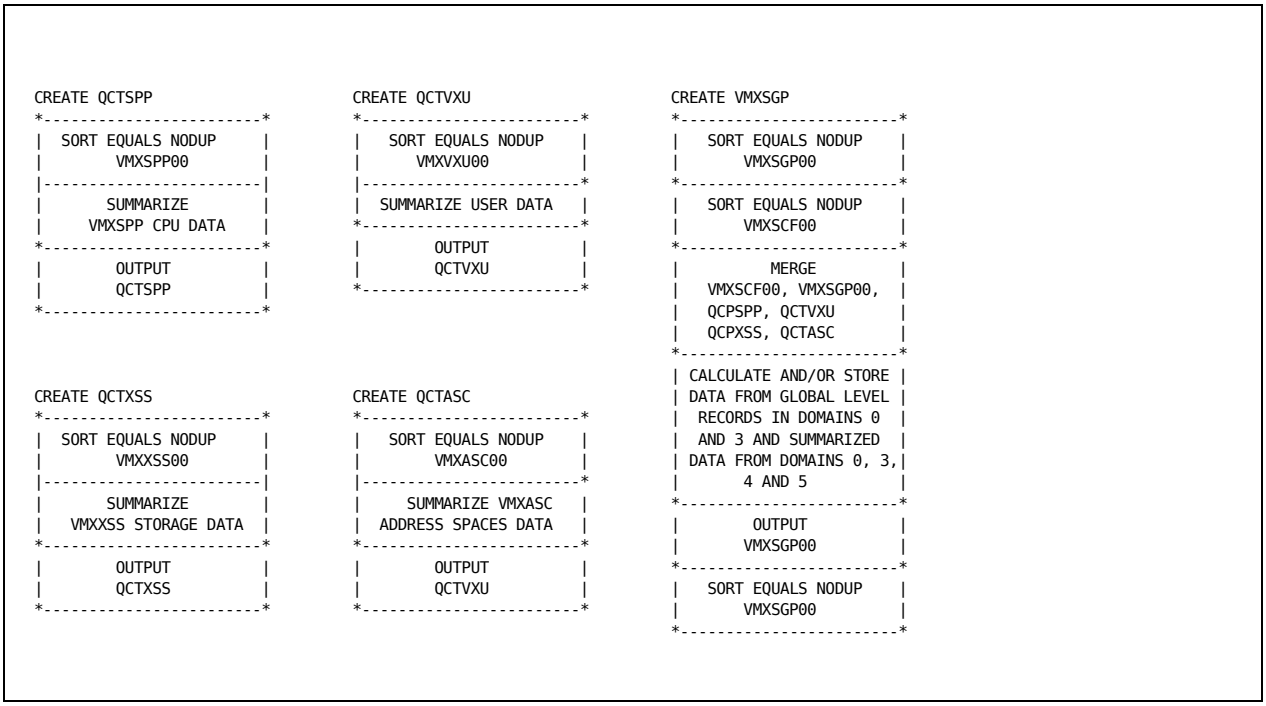


Figure 9-4d. VM Monitor Post Processing (Part 4 of 6)

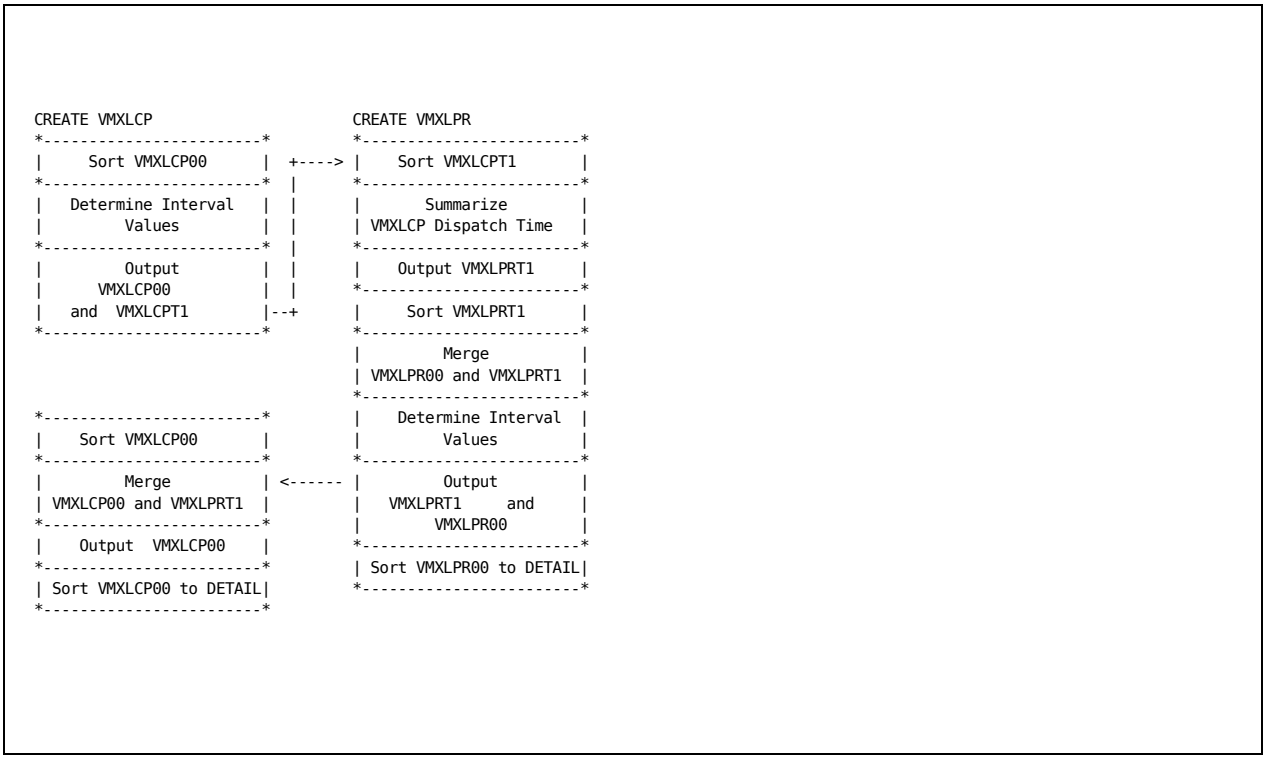


Figure 9-4e. VM Monitor Post Processing (Part 5 of 6)

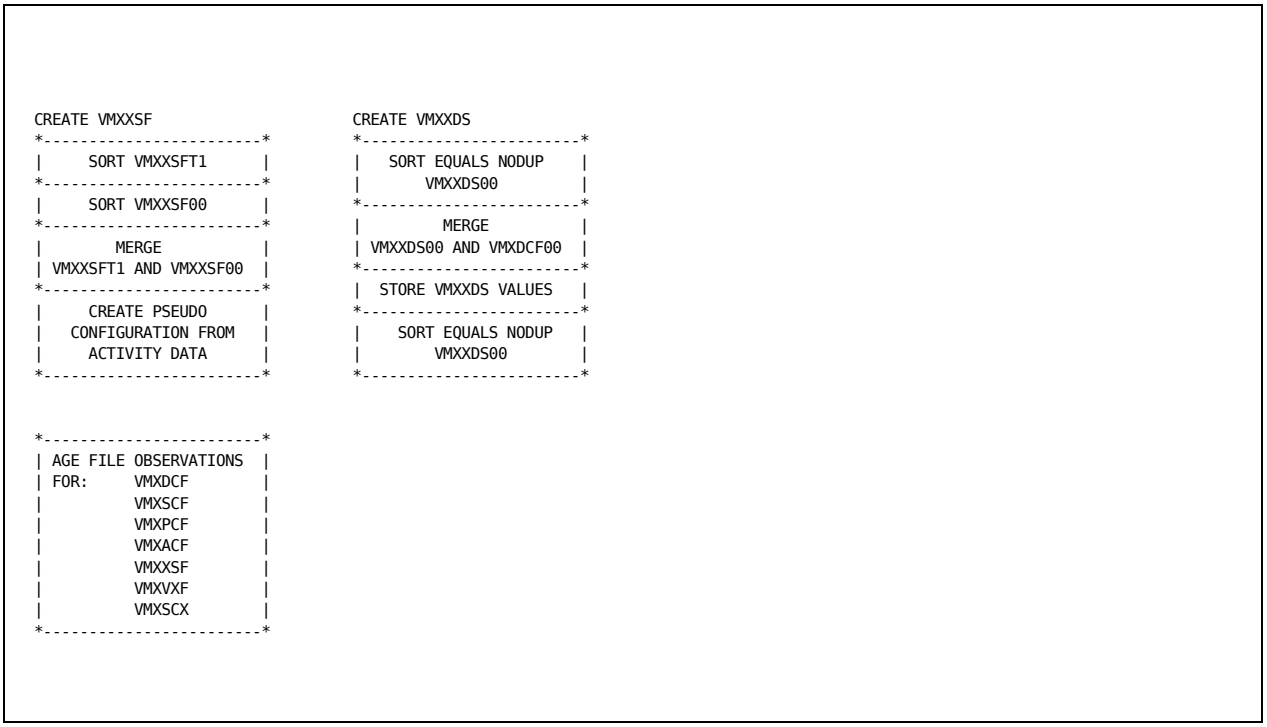


Figure 9-4f. VM Monitor Post Processing (Part 6 of 6)

9.2.5 Consolidation

An Input History Log (ADMIHL) and a VM Exceptions File (VMCVXC) are created during both phases: Account input and VM Monitor input. The IHL files and VXC files are consolidated for use in reporting and integrity checking.

Figure 9-5 shows this file consolidation logic.

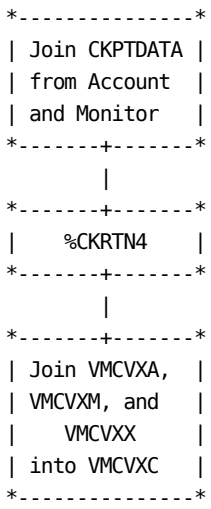


Figure 9-5. Consolidation

9.2.6 Database Time-Span Update

Using the newly created DETAIL or DAYS files, the DAYS, WEEKS, and MONTHS time-spans are updated to retain the latest day, week-to-date, and month-to-date data.

There are no exits provided in this phase.

9.2.7 File Aging

The CA MICS Analyzer Option for VM/CMS files are aged to delete the oldest cycles and rename the work files to current cycles.

There are no exits provided in this phase.

9.3 DAY080 Physical Structure

This section provides an overview of the physical structure of the daily update for the CA MICS Analyzer Option for VM/CMS, DAY080.

The bulk of the code required for the daily update is brought together by member DAY080 in INCLLIB (see Figure 9-6). This provides all the requisite code for the CA MICS architectural structure, together with the site-generated code that defines the product support for files and input processing.

```
Module:         Calls:              Description:                            Created by:

INCLLIB(DAY080)

          USOURCE($BASMSTR)         BAS master control macros               BASPGEN
           SOURCE($BASSFD,          VMC component SFD flags                 VMCCGEN
                  $VMCSFD,          BAS component SFD flags                 BASCGEN
                  #VMCEXIT)         VMC unit level exit overrides           Installation tailored
          USOURCE(#BASEXIT,         BAS unit level exit overrides           Installation tailored
                  $VMCCYCS,         VMC unit cycle definition               CYCLEGEN
                  #VMCMSTR,         VMC complex level master processing code Delivered code
                  $VMCDYST,         VMC %cccDYST PARAMETERS                 VMCPGEN
                  $VMCXCKP,         VMC %cccXCKPT PARAMETERS                VMCPGEN
                  #VMCEXIT,         VMC unit level exit overrides           Installation tailored
                  #USRMAC)          Installation-defined macros             Installation tailored
           INCLLIB(BASACT)          Installation Accounting Component code  Delivered code
           SOURCE(BASDAILY)         VMC DAILY Update Processing             VMCCGEN
```

Figure 9-6. DAY080 Structure

9.4 Daily Update Physical Structure

$VMCXDEF, which resides in sharedprefix.MICS.SOURCE, contains the information that controls the processing flow of the VM and CMS Analyzer Daily Update that is required at the unit level (see Figure 9-9). This information is used by BASDAILY to invoke the appropriate modules for processing, which is determined based upon parameters contained in VMCOPS.

The Account processing logic modules, VMCACTFI and VMCACTFU, are invoked when the VM Account data is to be processed in the unit. The VMCACTST module is invoked when no Account data is to be processed.

Following the VMCACTxx processing, the Input History Log file is again renamed and saved so that it is not overwritten by the VM Monitor processing routines. The VM Monitor Data logic modules, VMCMXAFI and VMCMXAFU, are invoked when the VM Monitor Data is to be processed within the unit; otherwise, VMCMXAST is called after which the Input History Log file from the VM Account processing step is renamed and saved.

Following the data reduction steps, the saved Input History Log files are combined and %CKRTN4 is invoked which provides installation access to the history file information. The VM Exception files created in the Account and VM Monitor data reduction steps are then combined into the VMCVXC File for later processing by the exception modules in DAY200.

```
SOURCE(VMCACTFM)

  call %INCACT (PHASE=FORMAT1)     VM Account Process Flow Executor          Resides in SOURCE(#VMCMSTR)

    if ACTSW = YES then
      call SOURCE(VMCACTFI)          VM Account Raw Data Input Logic
     else
      call SOURCE(VMCACTST)          VM Account Dummy Process Stub
    endif

SOURCE(VMCACTUP)

  call %INCACT (PHASE=DBUPDAT1)    VM Account Process Flow Executor          Resides in SOURCE(#VMCMSTR)

    call SOURCE(VMCACTFU)            VM Account Database Update Logic

SOURCE(VMCMXAFM)

  call %INCMXA (PHASE=FORMAT2)     VM Monitor Process Flow Executor          Resides in SOURCE(#VMCMSTR)

    if MXASW = YES then
      call SOURCE(VMCMXAFI)          VM Raw Data Input Logic
     else
      call SOURCE(VMCMXAST)          VM Dummy Process Stub
    endif

SOURCE(VMCMXAUP)

  call %INCMXA (PHASE=DBUPDAT2)    VM Monitor Process Flow Executor          Resides in SOURCE(#VMCMSTR)

    call SOURCE(VMCMXAFU)            VM Database Update Logic

SOURCE(DYVMCFMU)

  call %CKRTN4                     Input history log examination via
                                   exit _USRIHL in #BASEXIT
```

Figure 9-7. Daily Update Structure

9.5 Account Physical Structure

The VM Account Data processing contains two logical sections, the input raw data module (VMCACTFI) and the post-processing module (VMCACTFU).

The input raw data section consists of two subsections. The first subsection sets up the files that will be created, together with element lengths and formats, as well as constant value (literal) and work element definitions. This is followed by %CKRTN1, which performs checkpoint validation; %VMCLUPA, which calls the second subsection for data input; and %CKRTN3 for end-of-input processing.

The second input raw data subsection begins at label ACTPROC and continues to the end of the SAS data step. This subsection is called by %VMCLUPA for each system defined in VMCOPS which provides VM Account data to CA MICS. The data input processing is confined to a single loop which terminates at end-of-file for each separate system.

The data input loop identifies the current input record, and if the record is either a 'C0' (user format) or an unrecognized type, %VMC0EXIT is called to allow installation record identification as an RSCS record, or to cause %CKRTN2 processing to be executed (date range validation). The record is then passed to the appropriate routine for data input. Unrecognized records are passed to the %VMCUACC exit for input. (See Chapter 10 for a complete description of VM and CMS Analyzer exits.)

The post-processing section contains one subsection per file built from VM Account data. These subsections provide any extra file-specific processing, invoke exit VMCACRT, and provide the standard interface with the CA MICS Installation Accounting Product.

Additional processing for the VMCCMU File determines new session start.

For the VMCDAC File, new device allocations are identified and device typing is performed. See Section 9.2.2 for special notes about VMCDAC processing.

VMCVRA post-processing assigns a value to the element VRASBSYS for the Installation Accounting Product and assigns device type based on the virtual unit record device identified in the input data.

VMCNTU post-processing performs no extra function, other than interfacing to CA MICS Accounting and Chargeback.

9.6 VM Monitor Processing Physical Structure

VM Monitor Data processing contains two logical sections: the input raw data module (VMCMXAFI) and the post-processing module (VMCMXAFU).

The input raw data section consists of two subsections. The first subsection sets up the files that will be created, together with element lengths and formats, as well as constant values (literals) and work element definitions. This is followed by %CKRTN1, which performs checkpoint validation; %VMCLUPX, which calls the second subsection for data input; and %CKRTN3 for end-of-input processing.

The second input raw data subsection begins at label MXAPROC and continues to the end of the SAS DATA step. This subsection is called by %VMCLUPX for each system defined in VMCOPS that provides VM Monitor data to CA MICS. The data input processing is confined to a single loop that terminates at end-of-file for each input DD statement.

The data input loop identifies the current input record. If the current input record is an Event Profile Record (Domain 1/01) or an Interval End Record (Domain 1/11), pending observations for the VMXSGP file are output. If the Interval End Record (Domain 1/11) is the first record encountered, then an observation is output to the VMXSCF file and it is used to establish the interval start timestamp, VMCINTTS.

If the first data records input are not from Domain 1, CA MICS ABENDs with a 998 as the initial configuration data. Additionally, if a Domain 1/5 record (processor configuration) is encountered that has information from the wrong system and VMCOPS specified VERIFY ABORT, an ABEND 998 is taken (CA MICS SYSID checking may also ABEND with code 310).

Following initial monitor domain processing the USRSEL exit is called. For Domain 1/05 records, %SYSID is invoked and an observation is written to the VMXPCF file. Finally, %CHKRTN2 is called for all records.

The next function reads the remaining data from the input records and outputs to the appropriate file. For the VMXVXU file, the Domain 4/03 record information is held pending a corresponding Domain 4/04 record. For the VMX_VT file, the Domain 4/08 record information is held pending a corresponding Domain 4/10 record. For the VMXSGP file, output occurs prior to the next Domain 0/01 Record (System Data - Processor). This is required because each VMXSGP observation is built from multiple input records that are adjacent in the input data stream. For example, information contained in certain records from Domain 0 (03, 06, 07, 08, 09, 10, 12 and 14) and Domain 3 (10) is held for producing VMXSGP observations. For all other files, output is synchronous with input. However, many of the files created at this point are work files that will be merged in the post-processing section.

The post-processing section contains one subsection per file built from VM Monitor data. These subsections provide any extra file specific processing. For example, both the VMXPCF and VMXDCF files have information propagated through all observations since each observation may have been created from different input source records.

Post-processing for files containing interval measurements consists of determining the difference between the values of rolling counters in chronologically adjacent and ascending observations. In addition to a change in the fundamental unit of measure for a file, a break is recognized when the duration between two adjacent records is greater than the interval length multiplied by a macro variable, &VMXINTV. &VMXINTV is defined at #VMCMSTR at sharedprefix.MICS.SOURCE with default value of 1.5. For the VMXVXU and VMXVXT files, information is obtained from sample data.

Creation of the VMXSPP file involves merging together the various workfiles (i.e., VMXSPdnn, where d is a domain number and nn is the record type within that domain) and the VMXPCF processor configuration information used to propagate processor specific information through each newly created observation.

Once the VMXVXU and VMXSPP files have been created, information is summarized across both files to arrive at system-wide measures needed for completing the VMXSGP file. These summarized files, along with the VMXSCF file and the VMXSGP file, are merged, interval calculations are performed, and derived elements are computed, completing the observations for the VMXSGP file.

Once the remaining files have been processed, observations are aged in the configuration files VMXDCF, VMXSCF, VMXPCF, VMXACF, VMXXSF, VMXVXF, and VMXSCX.

Chapter 10: MODIFICATION

The CA MICS Analyzer Option for VM/CMS is tailored primarily through the use of options and parameters, and sometimes through the use of exits. This chapter provides checklists for changing parameters and descriptions of the exits available during the DAILY update processing.

Before implementing an exit, we recommend that you determine whether the required modification can be accomplished by means of parameter modification or file tailoring. If an exit must be used, be sure to see Chapter 9 for details concerning where the exit is called during the DAILY processing.

This section contains the following topics:

10.1 Parameter Modification (see page 597)
10.2 Standard User Exits (see page 599)

10.1 Parameter Modification

This section provides checklists for use in tailoring the VM and CMS Analyzer. The following topics are addressed:

1 - Add or Delete a Data Source for a System
2 - Add or Change Serial Numbers for a System

10.1.1 Add or Delete a Data Source for a System

A data source is a type of data that will be processed for a given system. The data sources include MON and ACT. A data source is added for a system when the system is initially added to the unit; when the system is migrated from VM to z/VM; or when the system was initially set up for only one data source and it is necessary to add another (for example, initially set up for account data and later added monitor data).

When modifying both VMCOPS and INPUTVMC, the changes to prefix.MICS.PARMS must be made to both members before any jobs are run.

___ 1. Modify prefix.MICS.PARMS(VMCOPS)

___ 2. Modify prefix.MICS.PARMS(INPUTVMC)

___ 3. Submit prefix.MICS.CNTL(VMCPGEN)

___ 4. If VMCPGEN completes successfully (check MICSLOG for message VMC00098 indicating successful completion), continue with step 5; otherwise, correct the errors and continue with step 3.

___ 5. Update prefix.MICS.PARMS(JCLGENU) so that it contains the single line:

DAILY

___ 6. Submit prefix.MICS.CNTL(JCLGENU) to implement the changes and create a new DAILY job stream.

For information about the data source statements, see section 7.2.1.4 and 7.2.1.5.

10.1.2 Add or Change Serial Numbers for a System

It is necessary to change the CPU serial and model numbers for an installed system when you upgrade from one processor type to another. This process involves either the modification of existing OPTIONS or ALTCPU statements or the addition of ALTCPU statements in VMCOPS. This requires a VMCPGEN execution.

See section 7.2.1.1 and 7.2.1.2 for information about the OPTIONS and ALTCPU statements.

10.2 Standard User Exits

At times, additional system customization is needed to tailor CA MICS to your installation. 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:

- is likely to occur, based on our experience
- presents only a slight possibility of compromising the CA MICS data base or operational integrity, and
- 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, the CA MICS System Administrator has four methods by which CA MICS can be augmented, changed, and enhanced by the user.

- CA MICS standard option and parameter definitions

  The diverse installation and definition options provide most users with adequate provisions for customizing CA MICS.

- Modification through user exits

  Often there may be installation-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.

- Extension through user-written components

  Although most users do not initially consider this a way of modifying a standard CA MICS component, 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 SMF Component.

- System code modification

  Finally, if all else fails, the user may be able to satisfy the installation's requirements by implementing a source code change.

The purpose of this section is to introduce and explain the user exits which are provided with this component. The user must carefully explore whether or not it is necessary to develop a user exit routine. To this end, the user should first ensure that the requirement cannot be satisfied through one of the standard options or definitions. This process can be facilitated by discussing the requirement with the CA MICS Product Support Group.

```
The design, coding, testing, and implementation of CA MICS
user exit routines should be done carefully since errors may
result and corrupt the data.  The user should be meticulous
in the definition and validation of the exit routines to
ensure that system integrity and performance is not adversely
affected.

 GENERAL CONSIDERATIONS

 The user exits for the CA MICS Analyzer Option for VM/CMS are
 identified below.

        o  General Exits

           USRSEL   - File Processing Selection
           USRIHL   - Checkpoint File Inspection

        o  Analyzer Input Exits

         for account data:
           VMCUACA  - VMCACTFI Data Statement File Definition
           VMCUACB  - VMCACTFI Format, Lengths, and Labels
           VMC0EXIT - VMCACTFI C0 (USER) and Custom Record
                      Inspection
           VMCUACC  - VMCACTFI Custom Record Input


         for VM monitor data
           VMCUMXA  - VMCMXAFI Data Statement File Definition
           VMCUMXB  - VMCMXAFI Format, Lengths, and Labels
           VMCUMXC  - VMCMXAFI Custom Record Input
           VMCAPDAT - VMCMXAFI Domain 10 (APPLDATA) Record
                      Input (VM only)

        o  Analyzer Output Exits

         VMCACTFI - CP Account Data Processing

           USRSCMU  -  Resource Accounting File Exit
           USRSDAC  -  Device Accounting File Exit
           USRSVRA  -  RSCS Accounting File Exit
           USRSNTU  -  Network Accounting File Exit
           USRSNDT  -  Network Data Transmissions File Exit

         VMCMXAFI - VM Monitor Data Processing

           USRSSGP  -  System Global Data File Exit
           USRSSCF  -  System Configuration File Exit
           USRSSPP  -  System Processor Data File Exit
```

```
   USRSPCF  -  Processor Configuration File Exit
   USRSDCF  -  Device Configuration File Exit
   USRSXDV  -  Device Activity File Exit
   USRSXDS  -  Device Seeks Data File Exit
   USRSACF  -  CPVOL Configuration File Exit
   USRSCPV  -  CP Owned Volume Data File Exit
   USRSVXF  -  User Data Configuration File Exit
   USRSVXU  -  User Activity Sample Data File Exit
   USRS_VU  -  User Activity Event Data File Exit
   USRSVXT  -  User Transaction Sample Data File Exit
   USRS_VT  -  User Transaction Event Data File Exit
   USRSCHC  -  Channel Path Contention File Exit
   USRSBPG  -  Block Paging Data File Exit
   USRSXSF  -  Shared Storage Configuration File Exit
   USRSXSS  -  Shared Storage Management File Exit
   USRSSTP  -  System Timer Pop File Exit
   USRSCDI  -  DASD Cache Activity File Exit
   USRSCDP  -  Paging Cache Activity File Exit
   USRSLPR  -  Logical Partition Activity File Exit
   USRSLCP  -  Logical Processor Activity File Exit
   USRSSCX  -  Scheduler Parameters File Exit

o  Analyzer Parameter-Related Exits

   VMCACRT  -  Account Code Assignment Exit

o  Analyzer Special Exits

   VMCUACD  -  VMCACTFU Custom Record Post-Processing
   VMCUMXD  -  VMCMXAFU Custom Record Post-Processing
   VMCUPRC  -  General Exit following Post-Processing


o  Linux Only Exits

   USRSLXP  -  Linux Process Activity File Exit
   USRXLXP  -  DETAIL LNXLXP Disk Data Selection Exit
   USRDLXP  -  DAYS LNXLXP Data Selection Exit
   LNXACRT  -  Account Code Assignment Exit for Linux
   USRTLXP  -  DETAIL Tape Data Selection Exit

Note:  The USRXLXP and USRXLXP exits do not follow the same
conventions as the other raw exits, where subsetting records
affect not only the DETAIL, but also the higher timespan
files as well.  The USRXLXP and USRTLXP exits affect only the
DETAIL timespan files.

ADDITIONAL FIELDS AVAILABLE TO EXITS
```

Area data elements, which are available for use by the user exits, vary according to the processing phase and the particular user exit. Special data elements that are used in the Analyzer update process but are not stored in the CA MICS database (and therefore have no corresponding data dictionary description) are described below. The individual user exit descriptions list which of these elements are available for the exit routine.

Utility Elements

Several elements that are used throughout the Analyzer are retained. These elements are used to set flags or indicators, or to clear fields. They are:

```
ONE     Numeric, Value 1
YES     Numeric, Value 1
ZERO    Numeric, Value 0
NO      Numeric, Value 0
BLANKS  Character, Value Blanks
```

SKIP_REC - Skip (delete) Record Indicator

During the Input Raw Data Phases, this indicator can be set to ONE (1) within the USRSEL exit to cause the current input record to be dropped before it is processed further by the data input routine.

This indicator can also be set to ONE (1) by the file output exits (USRSfff) to inhibit output of the current observation to the database.

ACTSFLAG - Input Routine Indicator

During the Input Raw Data Phases, this indicator is set to show which input phase is being executed.

```
ACTSFLAG=1; /* Input Account Data Phase */
ACTSFLAG=2; /* Input VM, z/VM Monitor Data */
```

This flag is used within the USRSEL exit to determine execution based on input data type.

Utility Elements Available in VMCACTFI (Account Data)

VRACHK - USER Account (C0) Record is from RSCS

This element is set in user exit VMC0EXIT to indicate that the current C0 account record was produced by RSCS. VRACHK must be set to ONE (1) to indicate that the record contains RSCS data; otherwise, it must be set to ZERO (0).

VRALCLNN - RSCS Local Node Name

This element is set in user exit VMC0EXIT to indicate the local node name (which is different from userid) of the RSCS virtual machine that produced the current C0 Account record. This element may also be set in user exit USRSVRA. This element is only significant when VRACHK is set to ONE.

TS_READ - Read Standard CP Account Format Date/Time Field

This indicator is set in VMC0EXIT when an Account record not ordinarily handled by VMC, such as a non-RSCS C0 record, or other site-defined records must be processed. When TS_READ is set to ONE (1), the date/timestamp in columns 17-28 of the record will be read and Check Routine 2 processing invoked. If the date/timestamp is not in columns 17-28 of the record or it is not in standard CP account format (MMDDYYHHMMSS), TS_READ must be set to ZERO (0).

TS_CHK2 - Do Check Routine 2 Processing

This indicator is set in VMC0EXIT when an Account record not ordinarily handled by VMC, such as a non-RSCS C0 record, or other site-defined records must be processed. Setting TS_CHK2 to ONE (1) causes Check Routine 2 to be invoked to date-range validate the input record. When TS_CHK2 is set to ONE (1), the code in VMC0EXIT must set the element ENDTS to the value represented by the timestamp in the record being processed.

Caution:

Any record that is not date-range validated may cause duplicate data to be placed in the database.

Any record that does not go through Check Routine 2 processing will not be dropped if SKIP_REC is set to ONE in USRSEL.

The following sections identify the user exits available for

```
processing, provide a general overview of the processing
logic, and describe each of the user exits in detail.

The complete list of sections follows:

      1 - General Exits
      2 - Analyzer Input Exits
      3 - Analyzer Output Exits
      4 - Analyzer Parameter-Related Exits
```

10.2.1 General Exits

General exits are invoked for all products and a full description of how to use these exits appears in the CA MICS System Modification Guide. They are presented here to identify the special user exits available for the CA MICS Analyzer Option for VM/CMS.

Each 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 Installation Accounting Component. It also shows which data elements are available and any special considerations that should be taken into account.

```
+---------------+
|  U S R S E L  | File Processing Selection Exit
+---------------+
```

DESCRIPTION: This exit allows access to each record processed by any product. It can be used to select or exclude specific records.

INVOCATION: This exit gains control after a DETAIL input record has been read and before the record formats have been defined.

ACCOUNTING INTERFACE: No interface to CA MICS Accounting and Chargeback is provided at this point.

USES: This exit may be used to delete records before processing or to alter the data.

ELEMENTS AVAILABLE:

```
  ROUTINE -   The name of the routine that invoked this exit.
              To use this routine in the CA MICS Analyzer
              Option for VM/CMS, test for ROUTINE =
              'DYVMCFMT'.
  MRHDRDM  -  VM record Monitor domain
  MRHDRRC  -  VM monitor record code
  VMCRCODE -  CP account record type
  ENDTS    -  End timestamp for the record being processed.
  ORGSYSID -  The original SYSID for the system that
              produced the data being processed.
  SUSPEND  -  FLAG to skip records if Monitor suspend
```

CODING RESTRICTIONS:

This exit should not contain code that executes a LINK or

```
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:  Because this exit is used by all products,
the source code is located in #BASEXIT of the appropriate
source libraries.

SAMPLE USER EXIT:  Drop all VM Monitor Device activity
records.

       IF ROUTINE = 'DYVMCFMT' THEN DO;
         /* DROP all Domain 6 records                */
         IF MRHDRDM EQ 6 THEN SKIP_REC = YES;
         ELSE SKIP_REC = NO;
       END;

+---------------+
|  U S R I H L  | Examine Input History Log
+---------------+

DESCRIPTION:  This exit permits the inspection of the DETAIL
timespan checkpoint file immediately following processing of
all input records in the format routine.

INVOCATION:  This exit is given control by %CKRTN4, which
reads the checkpoint file as an integrity check to verify
that data has been read and that file data input has been
terminated correctly.

ACCOUNTING INTERFACE:  No interface to CA MICS Accounting and
Chargeback is provided at this point.

USES:  This exit may be used to verify that no duplicate VM
Monitor data has been read.  Due to the relative granularity
of the timestamps in the prefix.MICS.CHECKPT.DATA file and
the VM Monitor data when the same data is input to the
CA MICS Analyzer Option for VM/CMS in separate cycles, it is
possible that not all input records are date range deleted,
as they should be, in the second cycle.  This exit can check
for a reasonable number of records kept when date range
deletion has occurred.  This problem does not arise with VM
Account data.

ELEMENTS AVAILABLE:  All elements in the ADMIHL file,
including the following:

  ROUTINE  -  The name of the routine that invoked this exit.
              To use this routine in the CA MICS Analyzer
              Option for VM/CMS, test for ROUTINE =
```

```
              'DYVMCFMT'.
  COMPT -     The name of the data input step that provided
              the observation in the file CKPTDATA.  The
              values for this in the CA MICS Analyzer Option
              for VM/CMS are:
                'VMC ACT' for Account data input
                'VMC MXA' for VM Monitor data input
                'VMC VCNA' for Account record type 07
  KEPT -      The number of input records kept from the input
              for this run.
  DATETM_D -  The number of records deleted based on
              date/time criteria.
  RDRSEL_D  - The number of records deleted due to the flag
              SKIP_REC being set to 1 in the _USRSEL exit.
  SHORT_D -   The number of records deleted due to a
              detection of incorrect record length.
  OPTION_D  - The number of records deleted due to the
              detection of an unrecognized input record.
  ORGSYSID -  Original SYSID for the system that produced
              the data being processed.
  SYSID    -  The logical SYSID for the system that produced
              the data being processed.

CODING RESTRICTIONS:

  This exit should not contain code that executes a LINK or
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:  Because this exit is used by all products,
the source code is located in #BASEXIT of the appropriate
source libraries.

SAMPLE USER EXIT:

  IF ROUTINE EQ 'DYVMCFMT' AND COMPT EQ :'VMC MON' THEN DO;
    IF DATETM_D GT 0 AND KEPT LT 100 THEN DO;
      LINE = "--- TOO FEW MONITOR RECORD KEPT - ABENDING";
      CALL MICSLOG('VMC99');
      ABORT ABEND 998;
    END;
  END;
```

10.2.2 Analyzer Input Exits

This section provides a description of the CA MICS VM and CMS Analyzer standard user exits which are invoked during the data input phases of the daily update processing flow.

Each 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 Installation Accounting Component. It also shows which data elements are available and any special considerations that should be taken into account.

NOTE: The examples for the following exits assume that a user file is being added to the CA MICS VM and CMS Analyzer and that it has been defined in VMCGENIN.

Input record format when record is generated by CP diagnose instruction X'4C', Generate USER Account Record:

Field Name	Columns	Description
USER	1-8	Name of virtual machine generating the record
ACCOUNT	9-16	VM account number
DATE	17-22	Date: MMDDYY
TIME	23-28	Time: HHMMSS
DURATION	29-32	Length of time represented by the record
NAME	33-40	The name of the item being counted
CHGUSER	41-48	Name of the virtual machine to be charged for using the resource
VMCRCODE	79-80	'C0' VM USER Account record ID

Input record format when record is not generated by the CP diagnose instruction X'4C', Generate USER Account Record:

Field Name	Columns	Description
USER	1-8	Name of virtual machine using the resource
ACCOUNT	9-16	VM account number
DATE	17-22	Date: MMDDYY
TIME	23-28	Time: HHMMSS
DURATION	29-32	Length of time represented by the record
NAME	33-40	The name of the item being counted
VMCRCODE	79-80	'XX' installation specified ID

```
+---------------+
| V M C U A C A | VMCACT Data Statement File Definition
+---------------+

DESCRIPTION:  The VMCUACA exit is used to add new SAS data
files to the DATA statement of the VMCACT routine.

INVOCATION:  This exit is invoked during compilation of the
DATA statement in VMCACT.

ACCOUNTING INTERFACE:  There is no interface to CA MICS
Accounting and Chargeback.

USES:  This exit may be used to add one or more installation-
defined files to the Input Raw Account Data Phase.

ELEMENTS AVAILABLE:  Not Applicable

CODING RESTRICTIONS:  This macro should contain only code
that is valid on a SAS DATA statement.  It should not contain
a semicolon.  If a semicolon is present, a SAS 180 error will
occur.

SPECIAL NOTES:  Not Applicable

SAMPLE USER EXIT:  Define an intermediate file for
supplemental Account data:

  %MACRO VMCUACA;
   %MAINWRK(CCC=VMC,WRK=UAC,KEEP=%UACKEEP(TS=DETAIL)
  %MEND VMCUACA;


+---------------+
| V M C U A C B | VMCACT Formats, Lengths, and Labels
+---------------+

DESCRIPTION:  The VMCUACB exit is used to provide SAS
formats, lengths, and optional labels for any new SAS data
files added to the DATA statement of the VMCACT routine using
the %VMCUACA macro.

INVOCATION:  This exit is invoked during compilation of the
DATA statement in VMCACT.

ACCOUNTING INTERFACE:  There is no interface to CA MICS
Accounting and Chargeback.

USES:  This exit may be used to provide SAS FORMAT, LENGTH
```

```
and LABEL statements for installation-defined files in the
Input Raw Account Data phase.

ELEMENTS AVAILABLE:  Not Applicable

CODING RESTRICTIONS:  This macro should not contain any
executable code.

SPECIAL NOTES:  Not Applicable

SAMPLE USER EXIT:  Provide formats and lengths for an
intermediate file containing supplemental Account data:

  %MACRO VMCUACB;
   %UACFMT(TS=DETAIL);
   %UACLEN(TS=DETAIL);
  %MEND VMCUACB;


+-----------------+
| V M C 0 E X I T | VMCACT Record Inspection
+-----------------+

DESCRIPTION:  The VMC0EXIT exit is used for two purposes:

1)  Inspect C0 Account records to determine whether or not
    RSCS data is present.

2)  Set processing flags for other custom or nonstandard
    Account records.

INVOCATION:  This exit is given control when a record other
than:

   0C, 01, C1, 02, C2, 03, C3, 04, 05, 06, 07, 08, C8

is detected.  The data elements VMCRCODE, USER, and VMCACNT
have been read.  TS_CHK2 and TS_READ are set to ZERO when the
exit is called.

ACCOUNTING INTERFACE:  There is no interface to CA MICS
Accounting and Chargeback.

USES:  This exit is used to indicate the presence of RSCS
data and to cause date/time stamp read, and/or Check Routine
2 processing for custom or nonstandard Account records.

ELEMENTS AVAILABLE:
```

```
  ORGSYSID -  Original SYSID, set by VMCPGEN
  VMCRCODE -  Input record type code
  USER     -  USERID from columns 1-8 of the record
  VMCACNT  -  The account code from columns 9-16 of the
              record
  VRACHK   -  RSCS data indicator, exit sets to ONE if
              VMCRCODE = 'C0' and the record contains RSCS
              data.
  VRALCLNN -  RSCS local node name, exit sets this element
              when VRACHK = ONE.
  TS_READ  -  Exit sets TS_READ = ONE when the record
              contains a VM Account format date/time stamp,
              but the record contains data that is not
              normally handled by VMC.
  TS_CHK2  -  Exit sets TS_CHK2 = ONE when the record does
              not contain a VM Account format date/time
              stamp, but the exit routine has set ENDTS and
              wants a record that contains data not normally
              handled by VMC to be date-range validated by
              Check Routine 2.

CODING RESTRICTIONS:

  This exit should not contain code that executes a LINK or
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:  Not Applicable

SAMPLE USER EXIT:  Detect RSCS and intermediate files
containing supplemental Account data:

  %MACRO VMC0EXIT;
   IF VMCRCODE EQ 'C0' THEN DO;
     IF USER EQ :'NETWORK' THEN DO;  /* RSCS ?          */
       VRACHK = YES;                 /* FLAG AS RSCS    */
       VRALCLNN = 'VMSYS';           /* LOCAL NODE NAME */
     END;
     ELSE IF USER EQ 'COUNTER' THEN  /* C0 FORM OF OUR  */
       TS_READ = YES;                /* XX RECORD       */
   END;
   ELSE IF VMCRCODE EQ 'XX' THEN DO; /* OUR RECORD ?    */
     TS_READ = YES;                  /* OUR RECORD HAS  */
   END;                              /* VM DATE/TIME    */
  %MEND  VMC0EXIT;
```

```
+---------------+
| V M C U A C C | VMCACT User and Custom Record Input
+---------------+

DESCRIPTION:  The VMCUACC exit is used to read data elements
from a record not normally handled by VMC.

INVOCATION:  This exit is called during the execution of the
unrecognized record routine, during the Input Raw Account
Data phase in VMCACT.

ACCOUNTING INTERFACE:   There is no interface to CA MICS
Accounting and Chargeback.

USES:  This is used to read installation-defined account
records and to output observations to installation-defined
files.

ELEMENTS AVAILABLE:

  SYSID    -  System Identifier
  VMCRCODE -  Input record type code
  USER     -  USERID from columns 1-8 of the record
  VMCACNT  -  The account code from columns 9-16 of the
              record
  SKIP_REC -  Drop input record

CODING RESTRICTIONS:

  This exit should not contain code that executes a LINK or
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:  Always code a hanging '@' on the INPUT
statement(s) to hold the current record.  Failure to do this
will cause the following record to be lost when the current
record is flushed by VMC.

Set SKIP_REC to zero (0) to indicate that the record has been
read; otherwise, set SKIP_REC to one (1).  This is used to
increment the proper record counters.

SAMPLE USER EXIT:  INPUT data element from an installation-
defined account record.

  %MACRO VMCUACC;

    IF VMCRCODE = 'C0' AND USER = 'COUNTER' THEN DO;
      /* C0 RECORD FORMAT                                */
```

```
      INPUT @29 DURATION PIB4.    /* LENGTH OF TIME      */
                UACSTUFF $CHAR8.  /* WHAT WE'RE COUNTING */
                USER     $CHAR8.  /* CHARGE-TO USERID    */
                @;                /* HOLD THAT RECORD !  */
      SKIP_REC = NO;              /* NOTE: DATA READ     */
    END;
    ELSE IF VMCRCODE = 'XX' THEN DO;
      /* RECORD FORMAT WHEN CP ACCOUNT IS NOT USED       */
      INPUT @29 DURATION PIB4.    /* LENGTH OF TIME      */
                UACSTUFF $CHAR8.  /* WHAT WE'RE COUNTING */
                @;                /* HOLD THAT RECORD !  */
      SKIP_REC = NO;              /* NOTE: DATA READ     */
    END;
    ELSE SKIP_REC = YES;          /* NOTE: UNKNOWN RECD. */

    IF NOT SKIP_REC THEN DO;      /* DO MORE WORK...     */
      STARTTS = ENDTS - DURATION; /* START TIME          */
      %WRKOUT(CCC=VMC,WRK=UAC);   /* WRITE THE OBS.      */
    END;
  %MEND  VMCUACC;

+-----------------+
| V M C A P D A T |  Application Monitor Record Input
+-----------------+

DESCRIPTION:  The VMCAPDAT exit is used to read data elements
from an application record in the VMCMXA routine.

INVOCATION:  In VMCMXA, this exit is called when a domain 10
VM Monitor record (Appldata) is encountered.  The macro
parameter &INFOAREA identifies the area where this exit is
invoked:

  'VMX'

ACCOUNTING INTERFACE:  There is no interface to CA MICS
Accounting and Chargeback.

USES:  This is used to read monitor data generated by one or
more application programs in VM systems.

ELEMENTS AVAILABLE:

  SYSID    -  System Identifier
  ENDTS    -  The time the record was created by VM
  APL_OFF  -  Offset to the application data area
  APL_LEN  -  Length of the application data in bytes
  USER     -  Userid generating the data
  APL_PROD -  The product ID and the release level of the
```

```
              application
  APL_STAT -  Status flag
  MRHDRDM  -  Monitor record domain
  MRHDRRC  -  Monitor record type

CODING RESTRICTIONS:

  This exit should not contain code that executes a LINK or
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:  Always code a hanging '@' on the INPUT
statement(s) to hold the current record.  Failure to do this
will cause the following record to be lost when the current
record is flushed by VMC.

To add new SAS data files to the DATA statement, code the
VMCUMXA exit for VM Monitor application data.

SAMPLE USER EXIT:  INPUT data elements from a specific
application generated monitor record in a VM system.

  %MACRO VMCAPDAT(INFOAREA);

    IF &INFOAREA = 'VMX'          /* VM APPLICATION  */
      AND APL_PROD =: '9999999'   /* PRODUCT ID 9999999  */
    THEN DO;                      /* READ THE DATA       */
      INPUT @APL_OFF              /* POSITION CURSOR     */
               APLEFLD1  $CHAR8.  /* INPUT FIELD 1       */
               APLEFLD2  PIB4.    /* INPUT FIELD 2       */
                @;                /* HOLD THAT RECORD !  */
      %WRKOUT(CCC=VMX,WRK=APL);   /* WRITE THE OBS.      */
    END;

  %MEND VMCAPDAT;

+---------------+
| V M C U M X A | VMCMXA Data Statement File Definition
+---------------+

DESCRIPTION:  The VMCUMXA exit is used to add new SAS data
files to the DATA statement of the VMCMXA routine.

INVOCATION:  This exit is invoked during compilation of the
DATA statement in VMCMXA.

ACCOUNTING INTERFACE:  There is no interface to CA MICS
Accounting and Chargeback.
```

```
USES:  This exit may be used to add one or more installation-
defined files to the Input VM Monitor Data Phase.

ELEMENTS AVAILABLE:  Not Applicable

CODING RESTRICTIONS:  This macro should contain only code
that is valid on a SAS DATA statement.  It should not contain
a semicolon.  If a semicolon is present, a SAS 180 error will
occur.

SPECIAL NOTES:  Not Applicable

SAMPLE USER EXIT:  Define an intermediate file for
supplemental Monitor data:

  %MACRO VMCUMXA;
   %MAINWRK(CCC=VMC,WRK=UMX,KEEP=(SYSID USER ENDTS USRFLD1
                                  USRFLD2)
  %MEND VMCUMXA;


+---------------+
| V M C U M X B | VMCMXA Formats, Lengths, and Labels
+---------------+

DESCRIPTION:  The VMCUMXB exit is used to provide SAS
formats, lengths, and optional labels for any new SAS data
files added to the DATA statement of the VMCMXA routine using
the %VMCUMXA macro.

INVOCATION:  This exit is invoked during compilation of the
DATA statement in VMCMXA.

ACCOUNTING INTERFACE:  There is no interface to CA MICS
Accounting and Chargeback.

USES:  This exit may be used to provide SAS FORMAT, LENGTH
and LABEL statements for installation-defined files in the
Input Raw Monitor Data phase.

ELEMENTS AVAILABLE:  Not Applicable

CODING RESTRICTIONS:  This macro should not contain any
executable code.

SPECIAL NOTES:  Not Applicable

SAMPLE USER EXIT:  Provide formats and lengths for an
intermediate file containing supplemental Monitor data:
```

```
  %MACRO VMCUMXB;
   FORMAT USRFLD1 TIME8. USRFLD2 $8.;
   LENGTH USRFLD1 4 USRFLD2 $8;
   LABEL  USRFLD1='User Field 1'
          USRFLD2='User Field 2';
  %MEND  VMCUMXB;


+---------------+
| V M C U M X C | VMCMXA User and Custom Record Input
+---------------+

DESCRIPTION:  The VMCUMXC exit is used to read data elements
from a record not defined to VMCMXA.

INVOCATION:  This exit is called during the execution of the
unrecognized record routine, during the Input VM Monitor Data
phase in VMCMXA.

ACCOUNTING INTERFACE:  There is no interface to CA MICS
Accounting and Chargeback.

USES:  This is used to read installation-defined account
records and to output observations to installation-defined
files.

ELEMENTS AVAILABLE:

  SYSID    -  System Identifier
  MRHDRDM  -  Monitor record domain
  MRHDRRC  -  Monitor record type
  ENDTS    -  The time the record was created by VM
  SKIP_REC -  Drop input record

CODING RESTRICTIONS:

  This exit should not contain code that executes a LINK or
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:  Always code a hanging '@' on the INPUT
statement(s) to hold the current record.  Failure to do this
will cause the following record to be lost when the current
record is flushed by VMC.

Set SKIP_REC to zero (0) to indicate that the record has been
read; otherwise, set SKIP_REC to one (1).  This is used to
increment the proper record counters.
```

```
SAMPLE USER EXIT:  INPUT data element from an installation-
defined monitor record.

  %MACRO VMCUMXC;

    IF MRHDRDM = 10 AND MRHDRRC = 1 THEN DO;
      INPUT @9  USRFLD1  PIB4.    /* LENGTH OF TIME      */
                USRFLD2  $CHAR8.  /* WHAT WE'RE COUNTING */
                USER     $CHAR8.  /* USERID              */
                @;                /* HOLD THAT RECORD !  */
      SKIP_REC = NO;              /* NOTE: DATA READ     */
      %WRKOUT(CCC=VMX,WRK=UMX);   /* WRITE THE OBS.      */
    END;
    ELSE SKIP_REC = YES;          /* NOTE: UNKNOWN RECD. */

  %MEND  VMCUMXC;
```

10.2.3 Analyzer Parameter-Related Exits

Each 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 Installation Accounting Component. It also shows which data elements are available and any special considerations that should be taken into account.

```
+---------------+
| V M C A C R T | Account Code Assignment Exit
+---------------+

DESCRIPTION:  The VMCACRT exit is used to build the elements
that are in the VM Account Code Structure.

INVOCATION:  This exit gains control during the Account Data
Post-Processing Phase in VMCACTFU when the Accounting Journal
File exits are called.

ACCOUNTING INTERFACE:  This exit builds the VMCACCTx elements
for CA MICS Accounting and Chargeback.

USES:  This exit is used to set the VMCACCTx elements based
on the contents of USER, VMCACNT, or other elements.
```

```
ELEMENTS AVAILABLE:

  VMCRCODE -  VM Account record type.
  ORGSYSID -  The original SYSID for the system which
              produced the data being processed.
  VMCACNT  -  VM Account code.
  WRK_NAME -  Name of the file that is being processed by the
              accounting journal routine.

CODING RESTRICTIONS:

  This exit should not contain code that executes a LINK or
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:  N/A

SAMPLE USER EXIT:  See section 7.1.1.2 for a sample VMCACRT
routine.

+---------------+
| L N X A C R T | Account Code Assignment Exit for Linux
+---------------+

DESCRIPTION:  The LNXACRT exit is used to build the elements
that are in the VM Account Code Structure in the Linux (LNX)
information area.

INVOCATION:  This exit gains control during the Account Data
Post-Processing Phase in VMCMXAFU.

ACCOUNTING INTERFACE:  This exit builds the VMCACCTx elements
for CA MICS Accounting and Chargeback.

USES:  This exit is used to set the VMCACCTx elements based
on the contents of Linux identification data elements.

ELEMENTS AVAILABLE:

  USER     -  Virtual Machine Identifier
  LNXLGID  -  Logon group identifier on the Linux system
  LNXLUID  -  Logon user identifier on the Linux system
  LNXLUSNM -  Logon user name on the Linux system
  LNXLGPNM -  Logon group name on the Linux system
  LNXPNAME -  Linux process name
```

```
CODING RESTRICTIONS:

  This exit should not contain code that executes a LINK or
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:  N/A

SAMPLE USER EXIT:  See section 7.1.1.3 for a sample LNXACRT
routine.
```

10.2.4 Analyzer Output Exits

This section provides a description of the CA MICS VM and CMS Analyzer standard user exits that are invoked during the post-processing phases of the daily update processing flow.

Each 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 Installation Accounting Component. It also shows which data elements are available and any special considerations that should be taken into account.

```
        USRSVXM        VMCVXC -  VM Exception Activity
                                 Intermediate file
```

INVOCATION: Each exit is given control during the Input Raw Monitor Data Phase, immediately prior to the observation being written to the work file. At this point, no delta values have been calculated for any element that originates with a period-to-date counter. PRFMIDLE is an example of this type of element.

ACCOUNTING INTERFACE: Not Applicable

USES: These exits may be used to drop the current observation from the database.

ELEMENTS AVAILABLE: All elements in the file to which the observation will be written.

CODING RESTRICTIONS:

This exit should not contain code that executes a LINK or RETURN statement. A GOTO statement may only reference a label defined within the exit.

SPECIAL NOTES:

```
+---------------+
| U S R S f f f | VMC File Output Exits for Account Data
+---------------+
```

DESCRIPTION: These exits are used to edit the current observation before it is written to the DETAIL file, and to invoke the Accounting Journal file code in CA MICS Accounting and Chargeback.

```
exit: USRSCMU  file: VMCCMU -  User Resource Accounting file
      USRSDAC        VMCDAC -  Device Accounting file
      USRSVRA        VMCVRA -  RSCS  Accounting file
      USRSNTU        VMCNTU -  Network Accounting file
      USRSNDT        VMCNDT -  Network Data Transmissions
                               File

INVOCATION:  These exits are invoked during the Account Data
Post-Processing Phase immediately before the observation is
passed to the Accounting Journal Interface routine, before
being written to the DETAIL file.

ACCOUNTING INTERFACE:  These exits call the Accounting
Journal Interface.

USES:  This exit may be used to drop the current observation
from the database and to set elements that are used in the
CA MICS Accounting and Chargeback structure.

ELEMENTS AVAILABLE:  All elements in the file to which the
current observation will be written.

  WRK_FILE  -  The name of the file to which the observation
               will be written.
               Values:  VMCCMU, VMCDAC, VMCVRA, VMCNTU,
                        VMCNDT

CODING RESTRICTIONS:  Not Applicable

SPECIAL NOTES:  Each of these exits invokes two other exits
by default.  These are the USRUfff exit for the file and the
USRUJff journal interface exit for the file.

SAMPLE USER EXIT:

  MACRO _USRSCMU
       _USRUCMU   /* USER EXIT            */
       _USRUJCM   /* JOURNAL INTERFACE    */
  SKIP_REC = YES; /* DROP ALL DETAIL DATA */
  %

+---------------+
| U S R U f f f | VMC File Modification Exits for Account
+---------------+ Data

DESCRIPTION:  These exits are used to allow installation-
specific processing prior to execution of CA MICS Accounting
and Chargeback interface code.  This processing may include
```

```
creation of additional variables added to the file by the
installation.

exit: USRUCMU  file: VMCCMU -  User Resource Accounting file
      USRUDAC        VMCDAC -  Device Accounting file
      USRUVRA        VMCVRA -  RSCS Accounting file
      USRUNTU        VMCNTU -  Network Accounting file
      USRUNDT        VMCNDT -  Network Data Transmissions
                               File

INVOCATION:  These exits are invoked during the Account Data
Post-Processing Phase by the corresponding _USRSfff exit for
the file immediately before the observation is passed to the
Accounting Journal Interface routine, before being written to
the DETAIL file.

ACCOUNTING INTERFACE:  These exits may modify or create
elements used by the Accounting Journal Interface.

USES:  These exits may be used to determine the contents of
elements, which cannot be determined directly from the disk
itself.

ELEMENTS AVAILABLE:  All elements in the file to which the
current observation will be written.

  WRK_FILE  -  The name of the file to which the observation
               will be written.
               Values:  VMCCMU, VMCDAC, VMCVRA, VMCNTU,
                        VMCNDT

CODING RESTRICTIONS:

  This exit should not contain code that executes a LINK or
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:  Not Applicable

SAMPLE USER EXIT:

 MACRO _USRUCMU
  /* ------ ASSIGN CMUXMODE FOR DVMS AND GUESTS ------- */
  SELECT (USER);
   WHEN ('MVS') CMUXMODE = 'GUEST';
   WHEN ('RSCS') CMUXMODE = 'GUEST';
   OTHERWISE; /* - USE DEFAULT VALUES - */
  END;
  %
```

```
+---------------+
| U S R S f f f | VMC File Output Exits for VM Monitor Data
+---------------+

DESCRIPTION:  These exits are used to edit the current
observation before it is written to the work file.  The name
of each exit corresponds to the name of the file for which
the observation is to be output:

exit: USRSSGP  file: VMXSGP -  System Global Performance
      USRSSCF        VMXSCF -  System Configuration
      USRSSPP        VMXSPP -  System Processor Performance
      USRSPCF        VMXPCF -  Processor Configuration
      USRSDCF        VMXDCF -  Device Configuration
      USRSXDV        VMXXDV -  Device Activity
      USRSXDS        VMXXDS -  Devices SEEKs data
      USRSACF        VMXACF -  CPVOL Configuration
      USRSCPV        VMXCPV -  CP Owned Volume Data
      USRSVXF        VMXVXF -  User Data Configuration
      USRSVXT        VMXVXT -  User Interaction Sample Data
      USRS_VT        VMX_VT -  User Interaction Event Data
      USRSVXU        VMXVXU -  User Activity Sample Data
      USRS_VU        VMX_VU -  User Activity Event Data
      USRSCHC        VMXCHC -  Channel Path Contention Data
      USRSBPG        VMXBPG -  Block Paging Data
      USRSXSF        VMXXSF -  Shared System Configuration
      USRSXSS        VMXXSS -  Shared System Management
      USRSCDI        VMXCDI -  DASD Cache Activity Data
      USRSCDP        VMXCDP -  Paging Cache Activity Data
      USRSLPR        VMXLPR -  LPAR Activity Data
      USRSLCP        VMXLCP -  LPAR CPU Activity Data
      USRSSCX        VMXSCX -  Scheduler Parameters
      USRSASC        VMXASC -  Shared Address Spaces Work
                               file
      USRSVXX        VMCVXC -  VM Exception Activity
                               Intermediate file

      USRSLXP        LNXLXP    Linux Process Activity
                               Sample Data

      USRTLXP                  Linux Process Activity
                               for Detail Tape

      USRXLXP                  Linux Process Activity when
                               DETAIL on disk

      USRDLXP                  Linux Process Activity in
```

```
                                   DAYS

      USRSLXS        LNXLXS    Linux System Activity
                               Sample Data

INVOCATION:  Each exit is given control during the Input VM,
z/VM Monitor Data Phase, immediately prior to the observation
being written to the work file.  At this point, no delta
values have been calculated for any element that originates
with a period-to-date counter.  SGPCPUTM is an example of
this type of element.

ACTIVATION of the USRSLXP exit is invoked using the indirect
exit convention described in section 4.3.1.2 of the System
Modification Guide.

Note: For more information about indirect exit coding, see the section 4.3.1.2 Indirect
Exit Points in the System Modification Guide.

ACCOUNTING INTERFACE:  Not Applicable

USES:  These exits may be used to drop the current
observation from the database.

ELEMENTS AVAILABLE:  All elements in the file to which the
observation will be written.
$SCOL
CODING RESTRICTIONS:

  This exit should not contain code that executes a LINK or
  RETURN statement.  A GOTO statement may only reference a
  label defined within the exit.

SPECIAL NOTES:

  USRSVXU:  Elements in the VMXSGP file that are summed from
  elements in the VMXVXU file will be understated if VMXVXU
  observations are dropped.  These elements include active
  users and estimated working set size.

  USRSCHC:  The VMXCHC file can contain up to 256
  observations per monitor interval, that is one per CHPID
  available to the VM system.  Many of the 256 CHPIDs will
  have no devices attached to them.  An array is constructed
  in VMCMXAFI during input that contains the number of
  devices attached to each channel path.  This array can be
  tested in USRSCHC to determine whether or not an
  observation should be written to the work file when no
  devices attached to the CHPID are represented by the
```

```
  observation.

  Example:

  SKIP_REC = CHPID(CHCHPID+1) < 1;

  An alternative method for bypassing these observations has
  been provided.  The SAS macro variable &CHP_SEL can be set
  to cause all observations that represent CHPIDs without
  attached devices to be dropped automatically.  This
  variable can be set in #VMCEXIT outside any macro.  The
  default setting is to keep these observations.

  Example:

  automatically drop observations:

  %LET CHP_SEL = YES; ...or... %LET CHP_SEL = 1;

  keep all observations (the default):

  %LET CHC_SEL = NO; ...or... %LET CHP_SEL = 0;

SAMPLE USER EXITS:

  MACRO _USRSSGP
  IF ZONE NE 1 THEN SKIP_REC = YES; /* KEEP PRIME SHIFT */
  %

  MACRO _USRSCHC
  SKIP_REC = CHPID(CHCHPID+1) < 1; /* drop unused CHPIDs */
  %

+---------------+
| U S R D L X P |  DAYS Linux Process Activity File Exit
+---------------+

DESCRIPTION:  This exit allows access to the data elements
used to build records in the DAYS Linux Process Activity
(LNXLXP) File.  This exit follows the same conventions as all
other input exits; therefore, any record subsetting affects
not only the DAYS timespan but all higher timespans as well.

INVOCATION:  This exit gains control immediately prior to the
output of records to the DAYS Linux Process Activity
(LNXLXP).

ACCOUNTING INTERFACE:  No interface to CA MICS Accounting and
Chargeback.
```

```
USES:  This exit is used to add or change data elements or to
delete observations from the DAYS Linux Process Activity
file.  To exclude an observation, set SKIP_REC to one.

ELEMENTS AVAILABLE:  All elements in the DAYS Linux Process
Activity File.

ACTIVATION:  The USRDLXP exit is invoked using the indirect
exit convention described in section 4.3.1.2 of the System
Modification Guide.
```

Note: For more information about indirect exit coding, see the section 4.3.1.2 Indirect Exit Points in the System Modification Guide.

```
+---------------+
| U S R X L X P |  DETAIL Only File Exits
+---------------+

DESCRIPTION:  USRXfff exit is available for the Linux Process
Activity (LNXLXP) File.

- The USRXLXP exit is invoked during the initial output of
  Linux Process Activity File.

User code in the USRXLXP exit only modifies observations that
are output to the DETAIL timespans for the LNXLXP file.
These modifications are not propagated to the DAYS and higher
timespans.

For example, using the USRXLXP exit to restrict output of
Linux process activity records, where the Linux USER name
begins with "LX1" to the DETAIL LNXLXP01 Linux Process
Activity file has no impact on the observations summarized in
the DAYS and higher timespan LNXLXP files.

INVOCATION:  Exit gains control in the SAS data step that
creates the DETAIL timespan LNXLXP00 file in
sharedprefix.MICS.SOURCE(VMCLINXO).

ACCOUNTING INTERFACE:  No interface to CA MICS Accounting and
Chargeback.

USES:  The USRXfff exits allow subsetting and data element
modification that is restricted to the DETAIL timespan LNXLXP
file.

If you choose to create a DETAIL timespan file for the LNXLXP
file on tape rather than on DASD, use the USRTfff exit.
```

```
ELEMENTS AVAILABLE:  All elements in the Linux Process
Activity File.

ACTIVATION:  The USRDLXP exit is invoked using the indirect
exit convention described in section 4.3.1.2 of the System
Modification Guide.

Note: For more information about indirect exit coding, see the section 4.3.1.2 Indirect
Exit Points in the System Modification Guide.

SAMPLE USER EXIT - USRXLXP:

DESCRIPTION:  This sample exit drops Linux processes with the
process name of 'ABC'.

/*                                                  */
/* Drop all Linux process with the process name of  */
/* 'ABC'.                                           */
/*                                                  */
/*                                                  */
IF LNXPNAME NE: 'ABC' THEN SKIP_REC=1 ;


+---------------+
| U S R T f f f |  DETAIL Tape Data Selection Exit
+---------------+

DESCRIPTION:  The USRTfff exits provide access to DETAIL file
records prior to their output to DETAIL tape data sets.  See
Chapter 7 of this guide for information about DETAIL tape
processing and a list of eligible component files.

INVOCATION:  The USRTfff exit is invoked prior to output to
DETAIL tape files.

ACCOUNTING INTERFACE:  No interface is provided.

USES:  Use this exit to limit the amount of data that is
written to DETAIL tape files, or to modify the contents of
data elements written to DETAIL tape files.  Note that
USRTfff exit code only affects the DETAIL tape file content
for a particular file (fff). It has no impact on the normal
unit database content for the same file (fff).

ELEMENTS AVAILABLE:  All elements in the DETAIL timespan of
the file.

ACTIVATION:  The USRTfff exits are invoked using the indirect
```

exit convention described in section 4.3.1.2 of the System Modification Guide.

Note: For more information about indirect exit coding, see the section 4.3.1.2 Indirect Exit Points in the System Modification Guide.

All of the USRTfff indirect exits are activated in sharedprefix.MICS.SOURCE($cccEXIT). In $cccEXIT, you will find the following statement for each file (fff) eligible for DETAIL tape:

```
%LET USRTfff = ;
```

The exit is activated by providing a member name to the right of the equal sign:

```
%LET USRTfff = exitname ;
```

If you code an exitname, you MUST create a member in each unit's prefix.MICS.USER.SOURCE library with that name. Provide valid SAS code in the member. This code will be invoked prior to the output of records to the DETAIL tape file.

Example:

sharedprefix.MICS.SOURCE(#VMCEXIT):

```
%LET USRTLXP = LXPTEXIT ;
```

prefix.MICS.USER.SOURCE(LXPTEXIT):

```
IF LNXPNAME NE 'PRO1' THEN SKIP_REC=1 ;
```

In this example, LXPTEXIT was chosen as the name for the DETAIL tape exit for the LNXLXP file. Member LXPTEXIT was created in the unit level prefix.MICS.USER.SOURCE library for each unit where DETAIL tape was activated for the LNXLXP file.

The exit was coded to limit the records written to the DETAIL tape file to those from a particular queue manager, QMN1.

Note: The exit member name must be identical in each unit's prefix.MICS.USER.SOURCE library, but the contents of each member can be different (or identical) for each unit.

CODING RESTRICTIONS: Ensure that you have activated DETAIL

tape processing for a file, and executed VMCPGEN prior to activating and coding USRTfff exits.

10.2.5 Special Exits

This section provides an in-depth description of the CA MICS Analyzer Option for VM/CMS standard user exits that are not classified as input or output in the processing flow.

Each 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 Installation Accounting Component. It also shows which data elements are available and any special considerations that should be taken into account.

```
+---------------+
| V M C A C R T | Account Routine
+---------------+
```

This exit is discussed in detail in Section 7.1.1.2.

```
+---------------+
| L N X A C R T | Linux Account Routine
+---------------+
```

This exit is discussed in detail in Section 7.1.1.3.

```
+---------------+
| V M C U A C P | VMCACT User File Pre-Processing
+---------------+
```

DESCRIPTION: This exit is used to process data read within the VMCUACC exit or to manipulate files prior to standard post processing.

INVOCATION: This exit is invoked during the Account Data Post-Processing Phase in VMCACT before post-processing has been performed for the standard VMC Account-based files.

ACCOUNTING INTERFACE: No interface is provided.

USES: This exit may be used to SORT and process any file created in the VMCUACC exit, or to add elements to a file created by the account input phase.

```
ELEMENTS AVAILABLE:  All elements in the files processed.

CODING RESTRICTIONS:  The code in this exit must be self-
contained SAS DATA and/or PROC steps.

SPECIAL NOTES:  If the data input using the VMCUACC exit is
intended to be added to a standard CA MICS VMC file, this
exit can be used to combine the custom work file with the
existing standard file in the DETAIL data set.
```

```
SAMPLE USER EXIT:  Sort the work file, set counters for
summarization, and write the file to DETAIL:

  %MACRO VMCUACP;
  %LET BY = SYSID USER ENDTS;
  %WRKSORT(CCC=VMC,WRK=CMU);
  %WRKSORT(CCC=VMC,WRK=UAC);
  /* THE VMCUAC DATA WILL HAVE ENDTS LESS THAN OR EQUAL */
  /* TO THE VMCCMU ENDTS. THE VMCCMU WILL BE UPDATED TO */
  /* CONTAIN A COUNT OF EVENTS ACCUMULATED FROM THE     */
  /* VMCUAC FILE.                                       */
  DATA
  %MAINWRK(CCC=VMC,WRK=CMU);
   MERGE %WRKMERG(CCC=VMC,WRK=CMU,IN=CMU)
         %WRKMERG(CCC=VMC,WRK=UAC,IN=UAC) ;
    BY &BY;
    RETAIN PND 0;
    IF CMU THEN DO;
      /* OUTPUT IF COUNTS PENDING OR NO COUNTS AT ALL   */
      IF PND OR LAST.USER THEN
          %WRKOUT(CCC=VMC,WRK=CMU);
      CMUCOUNT=0;
      PND = 0;
    END;
    ELSE DO;
      PND = 1;
      CMUCOUNT+1;
    END;
    IF LAST.USER THEN DO;
      /* OUTPUT IF COUNTS PENDING                       */
      IF PND THEN
          %WRKOUT(CCC=VMC,WRK=CMU);
      CMUCOUNT=0;
      PND = 0;
    END;
  RUN;
  %MEND  VMCUACP;

Note:  This example is based on the examples for exits
VMCUACA, VMCUACB, VMCUACC, and VMC0EXIT.

+---------------+
| V M C U A C D | VMCACT User File Post-Processing
+---------------+

DESCRIPTION:  This exit is used to process data read within
the VMCUACC exit.
```

```
INVOCATION:  This exit is invoked during the Account Data
Post-Processing Phase in VMCACT after post-processing has
been performed for all standard VMC Account-based files.

ACCOUNTING INTERFACE:  No interface is provided.

USES:  This exit may be used to SORT and process any file
created in the VMCUACC exit and write the file to the DETAIL
data set.

ELEMENTS AVAILABLE:  All elements in the files processed.

CODING RESTRICTIONS:  The code in this exit must be self-
contained SAS DATA and/or PROC steps.

SPECIAL NOTES:  If the data input using the VMCUACC exit is
intended to be added to a standard CA MICS VMC file, this
exit can be used to combine the custom work file with the
existing standard file in the DETAIL data set.

SAMPLE USER EXIT:  Sort the work file, set counters for
summarization, and write the file to DETAIL:

  %MACRO VMCUACD;
   %LET BY = %UACSEQ(TS=DETAIL);
   %WRKSORT(CCC=VMC,WRK=UAC);

    DATA  %WRKXFREF(OP=SASFILE,WRK=UAC,CCC=VMC)
          (%UACFILE(OP=FILEOPTS,TS=DETAIL) LABEL=&UACSFDF);
     SET %WRKMERG(CCC=VMC,WRK=UAC);
     BY &BY;
     %UACLBL;                            /* ELEMENT LABELS */
     RETAIN ZERO 0 ONE 1;
     IF FIRST.UACSTUFF THEN UACCOUNT = ONE; /* ONE SET ON */
     ELSE UACCOUNT = ZERO;                  /* CHANGE     */
   RUN;
  %MEND  VMCUACD;

Note:  This example is based on the examples for exits
VMCUACA, VMCUACB, VMCUACC, and VMC0EXIT.


+---------------+
| V M C U M X D | VMCMXA User File Post-Processing
+---------------+

DESCRIPTION:  This exit is used to process data read within
the VMCUMXC exit.
```

```
INVOCATION:  This exit is invoked during the z/VM Monitor
Data Post-Processing Phase in VMCMXA after post-processing
has been performed for all standard monitor-based files.

ACCOUNTING INTERFACE:  No interface is provided.

USES:  This exit may be used to SORT and process any file
created in the VMCUMXC exit and write the file to the DETAIL
data set.

ELEMENTS AVAILABLE:  All elements in the files processed.

CODING RESTRICTIONS:  The code in this exit must be self-
contained SAS DATA and/or PROC steps.

SPECIAL NOTES:  If the data input using the VMCUMXC exit is
intended to be added to a standard CA MICS VMC file, this
exit can be used to combine the custom work file with the
existing standard file in the DETAIL data set.

SAMPLE USER EXIT:  Sort the work file, set counters for
summarization, and write the file to DETAIL:

  %MACRO VMCUMXD;
   %LET BY = SYSID USER ENDTS;
   PROC SORT NODUP NOEQUALS DATA=WORK.USERFILE; BY &BY; RUN;

   DATA DETAIL.USERFILE(KEEP=SYSID USER ENDTS USERFLD1
     USERFLD2  LABEL=Sample User Monitor File);
     SET WORK.USERFILE; BY &BY;
     RETAIN ZERO 0 ONE 1;
     .....custom processing.....
     OUTPUT DETAIL.USERFILE;
   RUN;
  %MEND  VMCUMXD;

Note:  This example is based on the examples for exits
VMCUMXA, VMCUMXB, and VMCUMXC.

+---------------+
| V M C U P R C | Post-Processing Exit
+---------------+

DESCRIPTION:  This exit is used to perform any site-specific
function that requires access to the VMC files after data
input and post-processing has completed.  The Input History
Log and VMC Exception Files have been created from the
intermediate file before this exit is called.
```

INVOCATION: This exit is invoked after all other processing has been completed and before the summary and aging routines are entered.

ACCOUNTING INTERFACE: No interface is provided.

USES: This exit may be used to examine files from any of the input data processing routines for site-specific use.

ELEMENTS AVAILABLE: All elements in the files processed.

CODING RESTRICTIONS: The code in this exit must be self-contained SAS DATA and/or PROC steps.

SPECIAL NOTES: Not Applicable

SAMPLE USER EXIT: For a complex that has a VM guest of VM, determine what part of the VM CPU consumption the VM user IMAGUEST is between 9:30 a.m. and 11:30 a.m.

```
 %MACRO VMCUMXD;
  DATA USER(KEEP=TTIME USER SYSID);
    SET &XVMU..VMXVXU00; BY SYSID USER;
    IF SYSID = 'VMSP';
    IF USER  = 'IMAGUEST'; /* GUEST OF SP SYSTEM   */
    IF '09:30'T < ENDTS <= '11:30'T;
    TTIME + VMUTTIME;      /* COLLECT ALL CPU TIME */
    IF LAST.USER;
    OUTPUT;
    STOP;
  RUN;
  DATA SYS(KEEP=SYSTIME SYSID);
    SET &XSGP..VMXSGP00; BY SYSID;
    IF SYSID = 'VMXA';
    IF '09:30'T < ENDTS <= '11:30'T;
    SYSTIME + SGPTOTTM;   /* COLLECT ALL CPU TIME */
  RUN;
  DATA _NULL_;
    MERGE SYS USER(RENAME=(SYSID=GUEST));
    PCT = 100 * USER/SYS;
    FILE PRINT;
    PUT "USERID " USER "ON SYSTEM " GUEST " CONSUMED"
        PCT 4.1 "PERCENT OF THE CPU OF SYSTEM " SYSID;
  RUN;
 %MEND  VMCUPRC;
```

Appendix A: MESSAGES

Messages

This appendix lists the messages generated by the CA MICS Analyzer Option for VM/CMS. The messages are printed in both the MICSLOG and SAS log data sets. When reviewing the output of a job, first look at the MICSLOG data set because it will contain fewer lines of information, thus making it easier to see whether or not a job or job step completed as expected. If the MICSLOG message does not provide enough information, see this appendix for more information about the message or the SAS log to help debug the problem.

The format of a MICSLOG message is

```
    hh:mm:ss.hs cccnnnnns ttttttttttttttttttttttttttttttttttt
```

hh:mm:ss.hs	is the time that the message was issued.
cccnnnnns	is the message identifier, which consists of three parts: ccc — three letter component identifier nnnnn — the message number s — the severity identifier, I is an informational message, W indicates a warning, and E indicates an error
ttt...ttt	is the message text. If the text spans more than one line, the message identifier is repeated.

The remainder of this appendix shows the MICSLOG messages produced for the CA MICS Analyzer Option for VM/CMS. In addition to the message identifier and text, this section presents the reason for the message, suggests actions to take to resolve the problem, and refers you to additional sources of information to help you understand the problem and its resolution.

```
+----------+
| VMC00001 |
+----------+

   TEXT:        INPUT:  DSN=dataset.name VOL=SER=volser

   TYPE:        Information

   REASON:      A new input file was started.

   ACTION:      None required.

   REFERENCES:  VMCPGEN


+----------+
| VMC00002 |
+----------+

   TEXT:        INPUT:  SPECIFIED SYSTEM ID:  xxxxxxxx IS
                MORE THAN 4 CHARACTERS IN LENGTH.

   TYPE:        Error

   REASON:      The ORGSYSID specified in SYSID statement is
                too long.

   ACTION:      Correct the ORGSYSID definition in SYSID.

                NOTE:  This error will occur only if the
                       SYSID member has not been validated by
                       BASPGEN.

   REFERENCES:  VMCPGEN


+----------+
| VMC00003 |
+----------+

   TEXT:        SPECIFIED CPU MODEL NUMBER:  xxxxxx IS MORE
                THAN FOUR CHARACTERS IN LENGTH.

   TYPE:        Error

   REASON:      The processor model on an OPTIONS or ALTCPU
                statement is too long.
```

```
   ACTION:      Correct the processor model in VMCOPS.

   REFERENCES:  VMCPGEN


+----------+
| VMC00004 |
+----------+

   TEXT:        SPECIFIED CPU MODEL NUMBER:  xxxxxx IS NOT
                NUMERIC.

   TYPE:        Error

   REASON:      The processor model on an OPTIONS or ALTCPU
                statement contains a non-numeric character.

   ACTION:      Correct the processor model in VMCOPS.

   REFERENCES:  VMCPGEN


+----------+
| VMC00005 |
+----------+

   TEXT:        SCP FOR orgsysid IS NOT A VM SYSTEM.

   TYPE:        Warning

   REASON:      The SCP definition for an ORGSYSID specified
                in SYSID is not identified as a VM system.
                This is determined by comparing the first two
                characters of the SCP definition with the
                characters 'VM'.

   ACTION:      None required.  This is a consistency check
                only.

   REFERENCES:  VMCPGEN


+----------+
| VMC00006 |
+----------+

   TEXT:         SYSID AND VMCOPS CPU MODEL FOR orgsysid DO
                 NOT MATCH.
```

```
   TYPE:         Warning

   REASON:       The CPU model specified in SYSID and the CPU
                 model specified in VMCOPS for the ORGSYSID
                 are not the same.

   ACTION:       None required.  This is a consistency check
                 only.  If your site uses the MIPS values
                 specified in SYSID, this message may
                 indicate an out-of-date value for MIPS.

   REFERENCES:   VMCPGEN


+----------+
| VMC00007 |
+----------+

   TEXT:         MORE THAN ONE OPTIONS STATEMENT FOR ORGSYSID
                 xxxx FOUND IN VMCOPS.

   TYPE:         Error

   REASON:       Only one OPTIONS statement per ORGSYSID is
                 allowed.

   ACTION:       Delete the extra OPTIONS statement or change
                 it to ALTCPU.

   REFERENCES:   VMCPGEN


+----------+
| VMC00009 |
+----------+

   TEXT:         AN OBSOLETE KEYWORD HAS BEEN SPECIFIED.  IT
                 WILL BE IGNORED.  TO PERFORM SCHEDULER DATA
                 SELECTION, CODE MACRO USRSSCH IN
                 prefix.USER.MICS.SOURCE(#VMCEXIT).

   TYPE:         Warning

   REASON:       More than three parameters were found on an
                 OPTIONS statement.

   ACTION:       None required.  Delete the extra parameters
                 to prevent the message.  If your site is
                 using SCHEDULE class data, review the USRSSCH
```

```
                macro.

   REFERENCES:  VMCPGEN


+----------+
| VMC00010 |
+----------+

   TEXT:        ORIGINAL SYSID:   xxxx HAS BEEN PREVIOUSLY
                DEFINED.

   TYPE:        Error

   REASON:      An ORGSYSID was detected more than once in
                SYSID.

   ACTION:      Correct SYSID.

                NOTE:  This error should not occur if SYSID
                       has been validated by BASPGEN.

   REFERENCES:  VMCPGEN


+----------+
| VMC00011 |
+----------+

   TEXT:        THE MAXIMUM SYSID LIMIT HAS BEEN REACHED.
                CONTACT CA MICS PRODUCT SUPPORT GROUP.

   TYPE:        Error

   REASON:      More than 50 ORGSYSIDs have been defined in
                SYSID.

   ACTION:      Delete extraneous systems from SYSID.  If
                your site needs 50 or more ORGSYSIDs, contact
                Technical Support at http://ca.com/support
                for assistance.

   REFERENCES:  VMCPGEN


+----------+
| VMC00012 |
+----------+
```

```
   TEXT:        SPECIFIED DDNAME:  xxxxxxxx IS MORE THAN
                EIGHT CHARACTERS IN LENGTH.

   TYPE:        Error

   REASON:      A DDNAME specification on a MONITOR, ACCOUNT,
                or COMMON statement is too long.

   ACTION:      Correct the DDNAME specification.

   REFERENCES:  VMCPGEN


+----------+
| VMC00013 |
+----------+

   TEXT:        THIS ORIGINAL SYSID:  xxxx AND DATA SOURCE
                xxxxxxxx HAVE ALREADY BEEN DEFINED.

   TYPE:        Error

   REASON:      An ACCOUNT or MONITOR statement for the
                ORGSYSID has been found more than once in
                VMCOPS.

   ACTION:      Delete the extraneous statement.

   REFERENCES:  VMCPGEN


+----------+
| VMC00014 |
+----------+

   TEXT:        SPECIFIED ORIGINAL SYSID:  xxxx HAS NOT BEEN
                DEFINED ON A PREVIOUS OPTIONS STATEMENT.

   TYPE:        Error

   REASON:      The OPTIONS statement for an ORGSYSID must
                precede the ACCOUNT and/or MONITOR statement
                for the ORGSYSID.

   ACTION:      Add or move the OPTIONS statement into the
                correct position in VMCOPS.

   REFERENCES:  VMCPGEN
```

```
+----------+
| VMC00015 |
+----------+

   TEXT:        SUBPARAMETER:  xxxxxxxx IS NOT VALID ON A
                xxxxxxxx STATEMENT.

   TYPE:        Error

   REASON:      An invalid parameter was found on a statement
                in VMCOPS.

   ACTION:      Correct the specified statement and
                parameter.

   REFERENCES:  VMCPGEN


+----------+
| VMC00016 |
+----------+

   TEXT:        KEYWORD:  xxxxxxxx IS NOT RECOGNIZED.

   TYPE:        Error

   REASON:      An invalid statement was found in VMCOPS.

   ACTION:      Correct or delete the specified statement.

   REFERENCES:  VMCPGEN


+----------+
| VMC00017 |
+----------+

   TEXT:        END OF MEMBER xxxxxxxx INPUT.

   TYPE:        Information

   REASON:      End-of-file was reached on an input file.

   ACTION:      None required.

   REFERENCES:  VMCPGEN
```

```
+----------+
| VMC00018 |
+----------+

   TEXT:        REQUIRED STATEMENT MISSING: xxxxxxxx

   TYPE:        Error

   REASON:      A parameter statement required in VMCOPS was
                not found.

   ACTION:      Add the required statement.

   REFERENCES:  VMCPGEN


+----------+
| VMC00019 |
+----------+

   TEXT:        ERROR(S) ENCOUNTERED IN VM UNIT PARAMETER
                DEFINITIONS, MACRO GENERATION ABORTED.
                CORRECT THE ERRORS AND RE-EXECUTE VMCPGEN.

   TYPE:        Error

   REASON:      Errors were found during VMCOPS parameter
                analysis.

   ACTION:      Correct the previously flagged errors and
                resubmit VMCPGEN.

   REFERENCES:  VMCPGEN


+----------+
| VMC00020 |
+----------+

   TEXT:        DUPLICATE DDNAMES NOT ALLOWED: ddname

   TYPE:        Error

   REASON:      A DDNAME entered on the current statement was
                used on a previous statement,

   ACTION:      Change the DDNAME specification.

   REFERENCES:  VMCPGEN
```

```
+----------+
| VMC00021 |
+----------+

   TEXT:        COMMON INPUT DD NOT ALLOWED FOR: datatype

   TYPE:        Error

   REASON:      An asterisk ('*') was coded as the DDNAME on
                the parameter statement.

   ACTION:      Change the DDNAME to an explicit
                specification.

   REFERENCES:  VMCPGEN


+----------+
| VMC00022 |
+----------+

   TEXT:        COMMON INPUT DD FOR datatype HAS ALREADY BEEN
                DEFINED.

   TYPE:        Error

   REASON:      A COMMON statement for the named data source
                has already been read.

   ACTION:      Delete the extraneous statement.

   REFERENCES:  VMCPGEN


+----------+
| VMC00023 |
+----------+

   TEXT:        MEMBER member IS EMPTY OR CONTAINS NO DD
                STATEMENTS.

   TYPE:        Error

   REASON:      An input member was found to contain no
                parameters.  In the case of INPUTVMC, no DD
                statements were found.
```

```
   ACTION:      Create or correct the member indicated.

   REFERENCES:  VMCPGEN


+----------+
| VMC00024 |
+----------+

   TEXT:        DDNAME ddname NOT FOUND IN INPUTVMC.

   TYPE:        Error

   REASON:      A DDNAME specified on an ACCOUNT, MONITOR, or
                COMMON statement in VMCOPS was not found on a
                DD statement in INPUTVMC.

   ACTION:      Correct or create INPUTVMC.

   REFERENCES:  VMCPGEN


+----------+
| VMC00025 |
+----------+

   TEXT:        INPUTVMC DDNAME ddname NOT REFERENCED IN
                VMCOPS.

   TYPE:        Warning

   REASON:      A DDNAME specified in INPUTVMC was not
                referred to in VMCOPS.

   ACTION:      None required.  This is a consistency check
                only.

   REFERENCES:  VMCPGEN


+----------+
| VMC00026 |
+----------+

   TEXT:        WORK FILE SPECIFICATION: nnn IS NOT IN THE
                RANGE: 0<nnn<=max

   TYPE:        Error
```

```
   REASON:      The number of work data sets specified on the
                WORK statement was greater than the allowable
                maximum.

   ACTION:      Correct the WORK statement.

   REFERENCES:  VMCPGEN


+----------+
| VMC00027 |
+----------+

   TEXT:        EXCEPTION SUMMARIZATION SUPPORT IS NOT
                AVAILABLE FOR FILE iiifff

   TYPE:        Error

   REASON:      The DSUM member for the specified file is not
                defined in the sharedprefix.MICS.SOURCE.

   ACTION:      Create the DSUM member #fffDSUM

   REFERENCES:  VMCPGEN


+----------+
| VMC00028 |
+----------+

   TEXT:        INVALID DATA IN SPIN STATEMENT: fff

   TYPE:        Error

   REASON:      The specified file is not recognized as a
                SPIN file.

   ACTION:      Correct the filename in the SPIN statement.

   REFERENCES:  VMCPGEN


+----------+
| VMC00029 |
+----------+

   TEXT:        INPUTVMC DDNAME:  ddname IS MORE THAN EIGHT
                CHARACTERS IN LENGTH
```

```
   TYPE:        Error

   REASON:      A DDNAME specified in INPUTVMC is too long.

   ACTION:      Correct the DDNAME in INPUTVMC.

   REFERENCES:  VMCPGEN


+----------+
| VMC00030 |
+----------+

   TEXT:        LESS THAN n PARAMETERS SPECIFIED AT xxxxxxxx
                STATEMENT

   TYPE:        Error

   REASON:      The specified statement in VMCOPS requires at
                least n parameters.

   ACTION:      Correct the statement in VMCOPS.

   REFERENCES:  VMCPGEN


+----------+
| VMC00031 |
+----------+

   TEXT:        ORGSYSID orgsysid IS NOT DEFINED IN
                PARMS(SYSID)

   TYPE:        Error

   REASON:      The specified ORGSYSID is not found in
                PARMS(SYSID).

   ACTION:      Create the ORGSYSID indicated in PARMS(SYSID)
                or correct VMCOPS.

   REFERENCES:  VMCPGEN


+----------+
| VMC00032 |
+----------+

   TEXT:        SPECIFIED CPU SERIAL NUMBER: xxxxxx IS MORE
```

```
                THAN SIX CHARACTERS IN LENGTH

   TYPE:        Error

   REASON:      The specified CPU serial number on an OPTIONS
                or ALTCPU statement is too long.

   ACTION:      Change the CPU serial number in
                prefix.MICS.PARMS(VMCOPS).

   REFERENCES:  VMCPGEN


+----------+
| VMC00033 |
+----------+

   TEXT:        SPECIFIED CPU SERIAL NUMBER:  xxxxxx IS NOT
                HEXADECIMAL.

   TYPE:        Error

   REASON:      The CPU serial number on an OPTIONS or ALTCPU
                statement contains a non-hexadecimal
                character.

   ACTION:      Correct the CPU serial number in VMCOPS.

   REFERENCES:  VMCPGEN


+----------+
| VMC00050 |
+----------+

   TEXT:        INTERNAL ERROR:  RETURN TO TOP OF DATA STEP
                OCCURRED DURING datatype INPUT.  REPORT THIS
                ERROR TO CA MICS PRODUCT SUPPORT GROUP.

   TYPE:        Error

   REASON:      A condition that caused  _N_ to exceed the
                value 1 occurred.

   ACTION:      Contact Technical Support at
                http://ca.com/support with the dump
                information provided in the DAY080 listing.

   REFERENCES:  DAY080
```

```
+----------+
| VMC00051 |
+----------+

   TEXT:        VM datatype RECORDS READ FROM DD: ddname

   TYPE:        Information

   REASON:      An end-of-file condition was encountered.

   ACTION:      None required.

   REFERENCES:  DAY080


+----------+
| VMC00052 |
+----------+

   TEXT:        record type      : count

   TYPE:        Information

   REASON:      This message accompanies VMC00051.  It
                contains input counts by record type.

   ACTION:      None required.

   REFERENCES:  DAY080


+----------+
| VMC00053 |
+----------+

   TEXT:        orgsysid:  UNKNOWN RECORD TYPE ENCOUNTERED:
                datatype:  record type

   TYPE:        Information

   REASON:      An input record not recognized by VMC was
                read.

   ACTION:      None required.  A summary message is produced
                at the end of data step.

   REFERENCES:  DAY080
```

```
+----------+
| VMC00054 |
+----------+

   TEXT:        orgsysid: DATA SET: ddname = dataset.name

   TYPE:        Information

   REASON:      A change in input data set occurred.

   ACTION:      None required.

   REFERENCES:  DAY080


+----------+
| VMC00056 |
+----------+

   TEXT:        orgsysid:  SHORT RECORD FROM ddname dsname :
                           RECORD NUMBER nnnn, TYPE xxxx,
                           LENGTH nn

   TYPE:        Warning

   REASON:      An input record was shorter that expected.

   ACTION:      Verify that the input data is correct.  This
                may indicate a problem in VM.

   REFERENCES:  DAY080


+----------+
| VMC00058 |
+----------+

   TEXT:        CPU MODEL SPECIFIED IN PARMS(VMCOPS) DOES NOT
                MATCH CPU MODEL IN MONITOR HEADER.  OPTION
                VERIFY=ABORT SPECIFIED IN VMCOPS.

   TYPE:        Error

   REASON:      The VM Monitor header record contains a CPU
                identifier different from that specified in
                VMCOPS for this ORGSYSID.
```

ACTION: Verify that the correct data is provided in the DDNAME indicated in message VMC00054. Correct VMCOPS and submit VMCPGEN if necessary. ABEND U997 accompanies this message.

REFERENCES: DAY080

+----------+
| VMC00059 |
+----------+

TEXT: CPU MODEL SPECIFIED IN PARMS(VMCOPS) DOES NOT MATCH CPU MODEL IN MONITOR HEADER. OPTION VERIFY=NOABORT SPECIFIED IN VMCOPS.

TYPE: Error

REASON: The VM Monitor header record contains a CPU identifier different from that specified in VMCOPS for this ORGSYSID.

ACTION: Verify that the correct data is provided in the DDNAME indicated in message VMC00054. Correct VMCOPS and submit VMCPGEN if necessary. Processing continues.

REFERENCES: DAY080

+----------+
| VMC00060 |
+----------+

TEXT: CURRENT DATE: date/time RUN_DT FROM CHECKPOINT ROUTINE: date/time

TYPE: Information

REASON: To provide consistency verification status.

ACTION: None required.

REFERENCES: DAY080

+----------+
| VMC00063 |

```
+----------+

   TEXT:        orgsysid:  DOMAIN 1 RECORD 9 CONFIGURATION
                RECORD MISSING!

   TYPE:        Error

   REASON:      Domain 1 Record 9 configuration record is not
                found in the input monitor data because of
                one of the following:

                o The size of configuration area in the
                  monitor saved segment is not large enough
                  to store configuration records.


                o Record is dropped from the ESAWRITE extract
                  to CA MICS if using Velocity Software'S
                  ESAWRITE utility.


                o Record is dropped during record selection
                  in the VMT Transfer program.


                o A monitor suspend event occurred.  MICSLOG
                  will display warning message VMC00314W.


                o You are using user exit %USRSEL to drop
                  monitor records, and this monitor record is
                  being dropped.


   ACTION:      Take the following action depending on the
                reason:

                o Notify the VM systems programmer to check
                  size of the configuration area in the
                  monitor saved segment. To ensure the
                  configuration area is large enough to store
                  all configuration records, it is
                  recommended to allocate the configuration
                  area 30% larger than required.


                o Review the record selection criteria for
                  the ESAWRITE extract file to CA MICS if you
                  are using Velocity Software to ensure the
```

Domain 1 Record 9 is being written and not excluded.

o See Section 7.2.3 of the VMT Transfer Program guide, review record selection, and check to see if 01/09 is excluded.

o See explanation of message VMC00314W.

o Check code in %USRSEL to ensure you are not dropping these records.

Contact Technical Support at http://ca.com/support with the information provided in the MICSLOG.

We recommend that you do not take any of the following actions to override the U997 abend before the problem is diagnosed.

1. Remove the input data set from the offending VM system in the DAY080 step JCL and rerun. Substitute DD DUMMY on the DDNAME of the data set you want to remove. See MICSLOG VMC00054I that displays the DDNAME in the DAY080 MICSLOG. Be sure you remove the DD DUMMY from the DAY080 step JCL after DAY080 completes.

2. Add the SYSPARM 'NODATA NOD1R9' override to the EXEC DAY080 SYSPARM= JCL card and rerun.

   CAUTION! Use of this override should be restricted. Be aware that use of this override will have full database impact with missing observations and invalid data values if you allow the data to process. MICSLOG message VMC00064I will be displayed when the rerun is complete. Be sure you remove the 'NODATA NOD1R9' from the SYSPARM parameter after DAY080 completes.

```
   REFERENCES:  DAY080


+----------+
| VMC00064 |
+----------+

   TEXT:        PROCESSING CONTINUES DUE TO SYSPARM='NODATA
                NOD1R9' IS SPECIFIED

   TYPE:        Information

   REASON:      U997 abend is overridden when the monitor
                sample record (Domain 1 Record 9) is missing
                in the monitor input data.  This message
                follows VMC00063 if NOD1R9 is specified in
                the SYSPARM parameter.

   ACTION:      Use of NOD1R9 should be restricted.  We
                recommend removal of NOD1R9 from SYSPARM
                immediately after a DAILY run.  If left for
                an indefinite period of time, dropped
                observations from some database files and
                invalid data element values could go
                undetected.

   REFERENCES:  DAY080


+----------+
| VMC00065 |
+----------+

   TEXT:        DD STATEMENT: ddname IS EMPTY.

   TYPE:        Warning

   REASON:      The input data set referenced by the
                indicated DDNAME was found to contain no
                data.

   ACTION:      Verify that this condition is expected.

   REFERENCES:  DAY080


+----------+
| VMC00066 |
+----------+
```

TEXT: orgsysid: LONG RECORD FROM: ddname dsname : RECORD NUMBER xxxx, TYPE x/xx, LENGTH xxx

TYPE: Warning

REASON: The length of the input record indicated is longer than expected.

ACTION: This may indicate new fields added to the record. Contact Technical Support at http://ca.com/support with the information provided in the listing.

REFERENCES: DAY080

```
+----------+
| VMC00067 |
+----------+
```

TEXT: sysid: CONFIGURATION RECORD FOR PROCESSOR xx IS MISSING!

TYPE: Error

REASON: The configuration record (Domain 1 Record 5) for the processor specified is missing in the input monitor data because of one of the following:

- A processor was VARIED online during monitoring. MONITOR EVENT PROCESSOR was set to DISABLE in the monitor profile for event data collection.
- The size of configuration area in the monitor saved segment is not large enough to store configuration records.
- Record is dropped from the ESAWRITE extract to CA MICS if using Velocity Software'S ESAWRITE utility.
- Record is dropped during record selection

in the VMT Transfer program.

- A monitor suspend event occurred. MICSLOG will display warning message VMC00314W.

- You are using user exit %USRSEL to drop monitor records, and this monitor record is being dropped.

ACTION: Take the following action depending on the reason:

- Notify the VM systems programmer to check if the monitor event profile for event data collection of the offending VM system is disabled. "MONITOR EVENT DISABLE PROCESSOR"

- Notify the VM systems programmer to check size of the configuration area in the monitor saved segment. To ensure the configuration area is large enough to store all configuration records, it is recommended to allocate the configuration area 30% larger than required.

- Review the record selection criteria for the ESAWRITE extract file to CA MICS if you are using Velocity Software to ensure the Domain 1 Record 5 is being written and not excluded.

- See Section 7.2.3 of the VMT Transfer Program guide, review record selection, and check to see if 01/05 is excluded.

- See explanation of message VMC00314W.

- Check code in %USRSEL to ensure you are not dropping these records.

Contact Technical Support at
http://ca.com/support with the information
provided in the listing and the record dump
in the SASLOG.

We recommend that you not code NOXACFG in the
SYSPARM to override the U997 abend before the
problem is diagnosed.

REFERENCES: DAY080

```
+----------+
| VMC00068 |
+----------+
```

TEXT: PROCESSING CONTINUES DUE TO SYSPARM=NOXACFG
SPECIFIED

TYPE: Information

REASON: U997 abend is overridden when the
configuration record of a processor is
missing in the VM Monitor data. This message
follows VMC00067 if NOXACFG is specified in
the SYSPARM parameter.

ACTION: Use of NOXACFG should be restricted to those
times when it is absolutely essential. We
recommend removal of NOXACFG from SYSPARM
immediately after a DAILY run. If left for
an indefinite period of time, it is
conceivable that some problems could go
undetected.

REFERENCES: DAY080

```
+----------+
| VMC00069 |
+----------+
```

TEXT: sysid: SYSTEM CONFIGURATION RECORD IS
MISSING!

TYPE: Error

REASON: The system configuration record (Domain 1, Record 4) for the SYSID indicated is missing in the VM Monitor data.

REASON: Domain 1 Record 4 configuration is not found in the input monitor data because of one of the following:

- The size of configuration area in the monitor saved segment is not large enough to store configuration records.
- Record is dropped from the ESAWRITE extract to CA MICS if using Velocity Software'S ESAWRITE utility.
- Record is dropped during record selection in the VMT Transfer program.
- A monitor suspend event occurred. MICSLOG will display warning message VMC00314W.
- You are using user exit %USRSEL to drop monitor records, and this monitor record is being dropped.

ACTION: Take the following action depending on the reason:

- Notify the VM systems programmer to check size of the configuration area in the monitor saved segment. To ensure the configuration area is large enough to store all configuration records, it is recommended to allocate the configuration area 30% larger than required.
- Review the record selection criteria for the ESAWRITE extract file to CA MICS if you are using Velocity Software to ensure the Domain 1 Record 4 is being written and not excluded.

o See Section 7.2.3 of the VMT Transfer Program guide, review record selection, and check to see if 01/04 is excluded.

o See explanation of message VMC00314W.

o Check code in %USRSEL to ensure you are not dropping these records.

Contact Technical Support at http://ca.com/support with the information provided in the listing and the record dump in the SASLOG.

REFERENCES: DAY080

+----------+
| VMC00070 |
+----------+

TEXT: CPU MODEL IN MONITOR HEADER DOES NOT MATCH ANY CPU MODELS IN PARMS(VMCOPS) FOR SYSTEMS FROM COMMON DD: ddname

TYPE: Error

REASON: The VM Monitor header record read from the DDNAME indicated contains a CPU identifier different from any systems that are specified in VMCOPS for this COMMON DDNAME.

ACTION: Verify that correct data is provided in the DDNAME indicated. Correct VMCOPS if necessary. Abend 310 accompanies this message.

REFERENCES: DAY080

+----------+
| VMC00071 |
+----------+

TEXT: INVALID NETWORK DEVICE TYPE ENCOUNTERED IN

```
                TYPE 0C RECORD.  NETWORK DEVICE TYPE HEX
                VALUE =%NETTYPE.  THIS MESSAGE WILL ONLY BE
                WRITTEN A MAXIMUM OF FIVE TIMES.

   TYPE:        Warning

   REASON:      The CP accounting record Type 0C encountered
                an invalid device type.


   ACTION:      Value must be either x'00', x'01', or x'02'.
                Advise your VM systems programmer of the
                invalid network types in the type 0C account
                record.

                Records will be not be processed.

   REFERENCES:  DAY080


+----------+
| VMC00098 |
+----------+

   TEXT:        >>> process HAS COMPLETED.

   TYPE:        Information

   REASON:      A process in a VMC job has completed.

   ACTION:      None required.

   REFERENCES:  DAY080


+----------+
| VMC00099 |
+----------+

   TEXT:        free form text

   TYPE:        Information

   REASON:      An input record from a parameter file is
                being echoed to the log file.  This message
                may be used to provide diagnostic
                information.

   ACTION:      None required.
```

```
   REFERENCES:  May be used by any program.


+----------+
| VMC00200 |
+----------+

   TEXT:        VM EXCEPTIONS REPORTED FROM dbfile: count

   TYPE:        Information

   REASON:      Status on number of exceptions found in the
                named VMC database file.

   ACTION:      None required.

   REFERENCES:  DYVMCEXC


+----------+
| VMC00250 |
+----------+

   TEXT:        FILE xxx IS NOT TURNED ON IN tttttt TIMESPAN
                FOR MBO REPORTING

                NO REPORTS FROM THE FILE xxx WILL BE PRODUCED

   TYPE:        Warning

   REASON:      To produce MBO reports from the file xxx
                requires that the file be turned on in the
                timespan requested for reporting.

   ACTION:      Turn on the file in VMCGENIN if MBO reports
                from the specified file are needed. Then run
                VMCCGEN. Otherwise, ignore the message.

   REFERENCES:  DYVMCMBO, MNVMCMBO, WKVMCMBO


+----------+
| VMC00300 |
+----------+

   TEXT:        MONITOR INITIALIZATION DATA DOES NOT OCCUR AT
                THE BEGINNING OF INPUT FOR SYSTEM sysid.
                THIS DATA IS REQUIRED FOR CORRECT PROCESSING
```

OF VM MONITOR FILES. FIRST INPUT RECORD: record type

TYPE: Error

REASON: Domain 1 Record 1 Event Profile configuration record is not found at the beginning of the input monitor data because of one of the following:

- The input monitor may have not been processed by the VM Data Transfer Program. Record format is invalid for CA MICS to process.
- The size of configuration area in the monitor saved segment is not large enough to store configuration records.
- Record is dropped from the ESAWRITE extract to CA MICS if using Velocity Software'S ESAWRITE utility.
- Record is dropped during record selection in the VMT Transfer program.
- A monitor suspend event occurred. MICSLOG will display warning message VMC00314W.
- You are using user exit %USRSEL to drop monitor records, and this monitor record is being dropped.

ACTION: Take the following action depending on the reason:

- Check the format of the monitor file from the offending VM (ISPF 3.4 then I) system. If the format is FB and not V or VB the data is invalid. Run the VMT Transfer program and send the file again.
- Notify the VM systems programmer to check size of the configuration area in the monitor saved segment. To ensure the configuration area is large enough to store all configuration records, it is recommended to allocate the configuration

```
                  area 30% larger than required.

                o Review the record selection criteria for
                  the ESAWRITE extract file to CA MICS if you
                  are using Velocity Software to ensure the
                  Domain 1 Record 1 is being written and not
                  excluded.

                o See Section 7.2.3 of the VMT Transfer
                  Program guide, review record selection, and
                  check to see if 01/04 is excluded.

                o See explanation of message VMC00314W.

                o Check code in %USRSEL to ensure you are not
                  dropping these records.

   REFERENCES:  VMCMXA


+----------+
| VMC00301 |
+----------+

   TEXT:        VM MONITOR INITIALIZATION
                AT:  time
                SYSTEM         :  orgsysid
                SYSTEM LEVEL: VvvRrMm   SERVICE LEVEL:  nnnn

   TYPE:        Information

   REASON:      Monitor configuration data has been read.

   ACTION:      None required.

   REFERENCES:  VMCMXA


+----------+
| VMC00302 |
+----------+

   TEXT:        CPU            : serial# model#

   TYPE:        Information

   REASON:      A processor configuration record has been
                read.
```

```
   ACTION:      None required.

   REFERENCES:  VMCMXA


+----------+
| VMC00303 |
+----------+

   TEXT:        DOMAINS ENABLED:
                EVENT  : domain list
                SAMPLE : domain list
                H.F.   : domain list

   TYPE:        Information

   REASON:      Monitor configuration data has been read.

   ACTION:      None required.

   REFERENCES:  VMCMXA


+----------+
| VMC00304 |
+----------+

   TEXT:        CURRENT INPUT DATA SET DOES NOT START WITH
                INITIALIZATION DATA

                FIRST INPUT RECORD:  record type

                ORGSYSID: orgsysid IS ASSUMED

   TYPE:        Warning

   REASON:      The input data set containing VM Monitor data
                does not have monitor initialization (Domain
                1 Record 1) information in the first record.
                The first data set for the specified system
                does contain initialization data, but the
                current data set does not.  The data set of
                missing initialization data is assumed for
                the specified system. This can happen
                because:

                o The input monitor may have not been
                  processed by the VM Data Transfer Program.
                  Record format is invalid for CA MICS to
```

process.

- The size of configuration area in the monitor saved segment is not large enough to store configuration records.
- Record is dropped from the ESAWRITE extract to CA MICS if using Velocity Software'S ESAWRITE utility.
- Record is dropped during record selection in the VMT Transfer program.
- A monitor suspend event occurred. MICSLOG will display warning message VMC00314W.
- You are using user exit %USRSEL to drop monitor records, and this monitor record is being dropped.

ACTION: Take the following action depending on the reason:

- Check the format of the monitor file from the offending VM (ISPF 3.4 then I) system. If the format is FB and not V or VB the data is invalid. Run the VMT Transfer program and send the file again.
- Notify the VM systems programmer to check size of the configuration area in the monitor saved segment. To ensure the configuration area is large enough to store all configuration records, it is recommended to allocate the configuration area 30% larger than required.
- Review the record selection criteria for the ESAWRITE extract file to CA MICS if you are using Velocity Software to ensure the Domain 1 Record 1 is being written and not excluded.
- See Section 7.2.3 of the VMT Transfer Program guide, review record selection, and check to see if 01/04 is excluded.
- See explanation of message VMC00314W.

```
                o Check code in %USRSEL to ensure you are not
                  dropping these records.

   REFERENCES:  VMCMXA


+----------+
| VMC00305 |
+----------+

   TEXT:        UNRECOGNIZED VM SYSTEM LEVEL:  PRODUCT BIT
                MAP = xxxxxxxxxxxxxxxx.  CONTACT MICS
                PRODUCT SUPPORT GROUP.

   TYPE:        Warning

   REASON:      The program product bit map in the monitor
                configuration record indicates that the VM
                system's release level is not recognized as
                belonging to any known VM system.  This may
                be caused by a transmission error that
                occurred on the network or by processing
                data from an unsupported VM system.

   ACTION:      Contact Technical Support at
                http://ca.com/support with the message text.

   REFERENCES:  VMCMXA


+----------+
| VMC00306 |
+----------+

   TEXT:        UNRECOGNIZED CPU MODEL VERSION CODE:  xxxx
                for CPU MODEL nnnn

   TYPE:        Warning

   REASON:      The CPU model version code in the monitor
                configuration record is not recognized as
                belonging to the specific CPU model in the
                message text.  This may be caused by
                processing data from a new CPU model version
                or from a new CPU model that is not currently
                supported.

   ACTION:      Contact Technical Support at
```

```
                http://ca.com/support with the message text.

   REFERENCES:  VMCMXA


+----------+
| VMC00310 |
+----------+

   TEXT:        UNRECOGNIZED CONTROLLER:  type AT ADDRESS:
                nnnn

   TYPE:        Information

   REASON:      A domain 6, type 4 record was read for a
                cache controller that is not recognized by
                the CA MICS Analyzer Option for VM/CMS.

   ACTION       None required.  Contact Technical Support at
                http://ca.com/support for assistance.

   REFERENCES:  VMCMXA, step CACHE


+----------+
| VMC00311 |
+----------+

   TEXT:        UNRECOGNIZED CP AREA TYPE: type

   TYPE:        Information

   REASON:      A domain 1, type 8 or domain 3, type 7 record
                was read that referenced a CP volume area
                that is not recognized by CA MICS.

   ACTION       None required.  Contact Technical Support at
                http://ca.com/support for assistance.

   REFERENCES:  VMCMXA, step ACF1.


+----------+
| VMC00312 |
+----------+

   TEXT:        RECORD TYPE 4/03 IS NOT FOLLOWED BY RECORD
                TYPE 4/04 FOR USERID: user
```

```
   TYPE:        Warning

   REASON:      A domain 4, type 3 record for the specified
                user is not followed by a domain 4, type 4
                record for the same userid in the VM Monitor
                data.  This may be caused by the data being
                truncated due to lack of monitor sample
                segment space to store all the sample data
                for a given interval.

   ACTION:      Check the exception file (VMCVXC) for
                occurrences of EXC08064.  If there are any,
                advise your VM systems programmer to increase
                the monitor saved segment space.

   REFERENCES:  VMCMXA


+----------+
| VMC00313 |
+----------+

   TEXT:        orgsysid:  sample/event DATA TRUNCATION
                OCCURRED AT time DUE TO INSUFFICIENT MONITOR
                SEGMENT SPACE

   TYPE:        Warning

   REASON:      This exception is detected when the target
                message type of the IUCV parameter list in
                the VM Monitor control record contains an
                indicator of insufficient space in the
                monitor saved segment.

   ACTION:      Advise the VM systems programmer of the need
                to increase the monitor saved segment space.

   REFERENCES:  VMCMXA


+----------+
| VMC00314 |
+----------+

   TEXT:        orgsysid:  MONITOR EVENT DATA SUSPENSION
                STARTED FROM date/time TO date/time

                RECORD LOST:  count
```

TYPE: Warning

REASON: A VM Monitor domain 1 type 03 record was read. Receipt of this message indicates the loss of event data due to monitor suspension. Monitoring is suspended either because the event portion of the monitor saved segment is full or the IUCV path to the receiving virtual machine has been quiesced.

ACTION: Notify the VM systems programmer of this message. This may indicate a requirement to increase the event portion of monitor saved segment size. Another reason is MONWRITE or if using Velocity Software, ESAWRITE may have been in a race condition and missed the IUCV interrupt when the MONITOR signaled data was ready to write.

If the DAY080 update step abends with U997 error message VMC00067E refer to Section 6.2.4 to avoid interruption to your DAILY processing.

REFERENCES: VMCMXA

+----------+
| VMC00400 |
+----------+

TEXT: UNKNOWN VELOCITY SOFTWARE ESALPS RELEASE %CASUPREL ENCOUNTERED IN DOMAIN C RECORDS. COUNT RECORDS REJECTED.
ORGSYSID: %ORGSYSID
HIGHEST ESALPS RELEASE KNOWN TO CA MICS: %CASUPLVL
CONTACT CA MICS SUPPORT

TYPE: Warning

REASON: Domain C records from an unsupported CA release of Velocity Software ESALPS product have been encountered and skipped.

ACTION: Contact Technical Support at http://ca.com/support and inform them of the release value that was encountered in the

Domain C records.

REFERENCES: None