CA MICS® Resource Management

Analyzer Option for VSE/Power Guide

Release 12.9

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

The CA MICS Analyzer Option for VSE/POWER enables you to incorporate VSE/POWER data into the CA MICS Database to provide the necessary information for the management of your data center. This information supports operations, accounting and chargeback, capacity planning and management, and service level objectives.

The Analyzer Option for VSE/POWER is one of the data integration applications available from the CA MICS family of products. Data integration applications are special-purpose products that help you analyze and manage technology areas, such as VSE/POWER.

The Analyzer Option for VSE/POWER data integration application:

- Processes raw VSE/POWER account data and incorporates it into the CA MICS database
- Edits it to minimize error
- Interprets the data into more usable data elements
- Links the information to the organization via cost centers
- Prepares it to be analyzed and presented so that less experienced people can use it to make business decisions

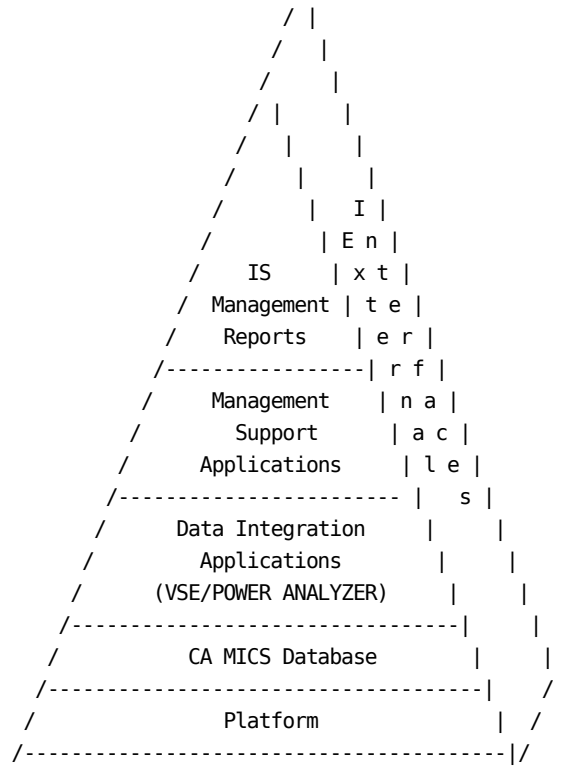


Figure 1-1. The CA MICS IS Management Support System

The Analyzer Option for VSE/POWER provides information for management support applications. Specifically, it provides information to CA MICS Accounting and Chargeback for use in such activities as cost management, billing, and chargeback.

The following sections provide some additional information about the Analyzer Option for VSE/POWER.

This section contains the following topics:

1.1 Primary Areas of Application

The CA MICS Analyzer Option for VSE/Power has valuable features that will help you do your job more efficiently and accurately. These features are described below:

- Consolidates data from multiple data centers

  The CA MICS Analyzer Option for VSE/Power integrates data from multiple data centers into an information database and then provides a common access method for reporting on batch activity.

- Provides graphic and tabular technical reports

  The reports provide information in a usable form for more effective and efficient analysis of specific system issues such as turnaround time and job tracking.

- Supports resource and priority accounting

  The accounting interface for the CA MICS Analyzer Option for VSE/Power enables IS to produce one invoice for users that use any combination of systems, including VSE, z/OS, VM, and VAX.

- Provides management indicators

  Managers can quickly identify VSE/POWER performance problems and trends that could lead to problems.

- Supports job groups and turnaround-time definitions

  The ability to define job groups and turnaround times allows IS to establish expected service levels for job classes defined by a user-coded exit.

- Provides a data dictionary

  Appendix B, Data Dictionary, describes the information found in the CA MICS Analyzer Option for VSE/Power information area files and how that information was derived.

- Supports the operational exception process

  The operational exception process allows you to define exceptions, capture data, and report on exceptional conditions.

The CA MICS Analyzer Option for VSE/Power provides the following benefits to your organization:

IMPROVE DAY-TO-DAY IS MANAGEMENT AND PLANNING

- Allows managers to track usage and plan corrective actions that ensure smooth operations.
- Enables managers to anticipate problems before the system is affected.

REDUCE RISK TO THE ENTERPRISE

- Enables managers to anticipate problems before the system is affected.
- Allows IS to define exceptional conditions and to assess the impact of these conditions.

CONTROL AND/OR HELP REDUCE COSTS

- Equitably charges users for the amount of resources used.

RAISE ENTERPRISE PRODUCTIVITY

- Reduces the task of coordinating information from multiple sources, so you can report or analyze information at the enterprise level instead of at the system level.
- Enables less knowledgeable VSE/POWER users to interpret the results of reports and to use information in the database.

1.2 Reporting and Inquiry Facilities

The CA MICS Analyzer Option for VSE/POWER provides 10 CA MICS Information Center Facility (MICF) inquiries, 3 management objective reports, and 11 exceptions created by the CA MICS exception process. These are described in detail in Chapters 3 and 4 of this guide.

MICF INQUIRIES

MICF is a panel-oriented productivity tool that enables you to gain access to information in the CA MICS Database. When you define input, selection criteria, and report options, MICF dynamically builds an inquiry program to fulfill your request. The inquiry program is executed either in batch or interactively and the results of the request are displayed at the terminal, printed, or cataloged for later replay according to your specifications.

The Analyzer Option for VSE/POWER includes standard MICF inquiries that you will find immediately useful. You can either use these inquiries directly from the MICF shared inquiry catalog or copy inquiries to your private inquiry catalog and tailor them to your specific needs.

MANAGEMENT OBJECTIVE REPORTS

The management objective reports provide a concise, graphical or tabular representation of an installation's processing objectives and how well they have been met. You define the objectives for the reports through a series of parameters. The reports are produced during normal CA MICS processing on a daily, weekly, and monthly basis. These reports can be activated or deactivated for a timespan (DAILY, WEEKLY, or MONTHLY) by coding the applicable REPORT statement in prefix.MICS.PARMS(EXECDEF).

EXCEPTION ANALYSIS REPORTS

See Chapter 4, VSE/POWER Analyzer Exceptions, for further information about the exceptions defined by the Analyzer Option for VSE/POWER.

REPORT INFORMATION

The Analyzer Option for VSE/POWER reports provide you with information for managing your VSE system. These reports can be grouped as follows:

SERVICE REPORTS--these reports provide information about the service that users of your VSE system are receiving.

- Job Initiation Objective reports the number of jobs for which on-time initiation objectives are met, missed, and exceeded.
- Job Turnaround Objective reports the number of jobs for which the on-time turnaround objectives are met, missed, or exceeded.
- Job Group Turnaround Received reports the number of jobs within a job group for which the on-time turnaround objectives are met, missed, or exceeded.

WORKLOAD REPORTS--these reports provide information about the way that your VSE system is being used.

- Daily Job/Class Activity reports the number of jobs executed by class.
- Special Forms Use by Shift reports the special forms being used for printing and the percent of use each form type gets.
- Partition Activity Audit reports job activity by partition

EXCEPTION REPORT--this report provides information to alert you that service may be degrading because some limit or limits have been reached.

- Top 20 Resource Consumers reports the top 20 jobs in the system by CPU time consumed.

GENERAL REPORTS--these reports provide general reporting information that does not fall into the other categories.

- Jobs Terminated with PFLUSH provides information about the number of jobs cancelled due to operator command.

- Jobs Terminated by RDREXIT provides information about the number of jobs cancelled because standards that are enforced by the installation reader exit routine were not met.

- Step Term Code Analysis provides a summary of job completions by termination code.

1.3 Product Considerations

Before installing the CA MICS Analyzer Option for VSE/Power, review the following items for applicability to your data center:

PRODUCT REQUIREMENTS

o Hardware Considerations

The DASD space requirements are as follows:

- Libraries: Approximately 16 MB of space is required in the CA MICS system libraries in addition to space already occupied.

- Database: Depends on installation specifications; DBMODEL facility is provided for database space planning.

o Supported Levels of VSE/Power

The CA MICS Analyzer Option for VSE/Power supports data from these levels of VSE/Power:

- Power 2.1.0 - Power 2.2.0 - Power 2.3.0
- Power 4.1 - Power 5.1 - Power 5.2
- Power 6.1 - Power 6.3 - Power 6.4
- Power 6.5 - Power 6.6 - Power 6.7
- Power 7.1 - Power 8.1 - Power 8.2
- Power 8.3 - Power 9.1

o Transport of Data From VSE to CA MICS

The VSE/Power account data is written to a sequential file in your VSE system. This data must be made accessible to CA MICS before it can be added to the CA MICS database. This can be done in any of the following ways:

- Make the DASD on which the data resides physically accessible to z/OS, and directly allocate the account data file in the CA MICS DAILY job.

- Copy the data to tape, and use the tape as input to the CA MICS DAILY job.

- If your VSE DASD is physically accessible from a VM system, use the VM Data Transfer Program to move the

data.

ORGANIZATIONAL CONSIDERATIONS

The CA MICS Analyzer Option for VSE/Power may present the first opportunity for your IS organization to work with personnel involved in departmental computing. To take advantage of this opportunity, we recommend that you investigate the following before installing the product:

- Reporting needs of the VSE/Power users
- Production standards (job statements, account codes, and so on)
- Perceived needs to expand the computing power of VSE

The CA MICS Analyzer Option for VSE/Power also provides input to the CA MICS Accounting and Chargeback Option. If you plan to activate this interface when you make the Analyzer Option for VSE/Power available to your users, consult the IMA administrator and the recipients of invoices produced by CA MICS Accounting and Chargeback to ensure that they are aware of the availability of the new information.

Chapter 2: USAGE GUIDELINES

The CA MICS Analyzer Option for VSE/POWER extends the power and flexibility of CA MICS to the VSE environment. Modeled after the Batch and Operations Analyzer product, VSE/POWER provides for consistent analysis of batch activity across all operating systems.

The following sections describe special issues associated with using the product and the interfaces provided to other CA MICS products.

This section contains the following topics:

2.1 Data Source Concepts

The IBM VSE/POWER product provides spooling and job queue management for the DOS/VSE operating system. In order to do this, POWER assumes control over a number of VSE partitions. It schedules batch jobs into these partitions and manages their output.

POWER Job Accounting is an optional facility that tracks activity within POWER-controlled partitions, as well as spool management activity performed by POWER. Job Accounting is triggered to record statistics by the occurrence of specific events. For example, the Execution Account Record is created whenever a job step (program) completes executing within a POWER-controlled partition. For a complete list of record types and their associated events, please refer to Section 6.2 of this guide.

Data is collected to the POWER Accounting File. Normally there is one file for each copy of VSE/POWER. However, the shared spool facility allows multiple POWER systems to share a single set of Account, Queue, and Data Files. When this feature is used, POWER adds fields to the front of each accounting record so that you can identify the system which created it.

Periodically, the Accounting File must be dumped to a separate disk or tape file. CA MICS uses this form of the data to produce the six files in the VSE/POWER Information Area. When the Accounting File is dumped and transmitted to z/OS for input into CA MICS, ensure that it is defined as a VB (Variable Blocked) file with the LRECL parameter greater than or equal to the longest record in the file. See Chapter 6 of this guide for data collection considerations. CA MICS file descriptions and the VSE/POWER records used to build them are found in Chapter 5.

2.2 Data Analysis

The CA MICS Analyzer Option for VSE/POWER includes facilities to help you better manage your batch environment. Where possible, they parallel those provided with the CA MICS Batch and Operations Analyzer. The sections below discuss these facilities in detail, paying particular attention to variations from this standard.

2.2.1 Job Group Service Analysis

The JOBGROUP variable is provided on the VSE/POWER Job Activity (PWRPJB) and Job Suspend (PWR_PJ) Files as a means of clustering jobs with similar resource or service requirements. User-defined parameters determine the type and quality of service that each group should receive. Since JOBGROUP is a summarization variable for these files at all timespans, it is always possible to perform meaningful analysis of service quality. The remainder of this section deals with differences in job group service analysis between the CA MICS Analyzer Option for VSE/POWER and the Batch and Operations Analyzer products.

The durations used to compute job turnaround time, and the event timestamps from which they are derived, are the result of the merging of records that represent different portions of a job's life. Limitations of VSE/POWER accounting data prevents the CA MICS product from tracking the same set of events provided in the Batch and Operation Analyzer files. Only the reader start, reader end, job start, and job end timestamps are available. As a result, when computing turnaround time, only reader time, input queue time, and job execution time are provided. No measures of the job output spool time are available. For more information on the merge processing, see Section 2.2.3 of this guide.

The accuracy of the event timestamps is a critical issue, particularly in a POWER shared spool environment. Since jobs can be read and executed on different machines, a synchronization error of just a few minutes in the system clocks can play havoc with service measurement. For example, ICCF is being used by programmers to submit compiler jobs to system VSE1. The jobs very quickly begin executing on system VSE2. These jobs constitute JOBGROUP 1 which has a turnaround requirement of five minutes (turnaround=input queue time). If the clock on VSE1 is ten minutes slow, then 100 percent of the jobs in JOBGROUP 1 will miss their service objective. If the clock is ten minutes fast, then 100 percent of the jobs will meet or exceed their service requirement; they will all show a turnaround time of zero minutes, since turnaround cannot be negative.

Another issue affecting timestamp accuracy occurs only for long-running jobs. Version 2.1.0 of VSE/POWER does not provide a means of accurately computing the start timestamp of a step that spans midnight. This can result in over stated values of the input reader time or job execution time. For more information, see Section 2.2.4 of this guide.

2.2.2 Input/Output Activity Analysis

The execution account record provides the sole measure of input/output activity for VSE/POWER. A table of start I/O counts occupies the last portion of the record and lists SIO counts by address for each device allocated to the job. The process of manipulating this data to form the elements in the Job, Program, and Device Activity Files requires a means to determine the device class and type associated with each address.

The PWRCONFG member of the unit PARMS library maps this relationship. By system identifier, it lists device address (or range) and the associated device type (i.e., 3380) and device class (i.e., DASD). This table is processed by the PWRPGEN job to produce a SAS format used in the DAY035 step of the DAILY database update job.

As the SIO table is read from the execution account record, the counts are accumulated to variables based on device class. These device class counters are available in the Job and Program Activity Files:

```
fffDSIO - DASD SIOs Issued
fffOSIO - Other SIOs Issued
fffSIO  - Total SIOs Issued
fffSSIO - Spool SIOs Issued
fffTSIO - Tape SIOs Issued
```

where fff is PJB or PPG.

The accuracy of these counts depends on the information provided in the PWRCONFG member. SIO counts are also maintained in the Device Activity File by address, device class, and type.

The values of device type and class provided in PWRCONFG are compared to list of valid values shown in Section 7.3.5. Any unrecognized values are flagged with a warning message. A valid but unrecognized device type will have little impact except possibly on locally developed reporting code. However, an unrecognized value of device class may affect the accuracy of the SIO counters.

The only device classes accumulated to fffDSIO are DASD and DASD/FBA (Fixed Block Architecture). Similarly, TAPE is accumulated to fffTSIO, and SPOOL to fffSSIO. Counts for other device classes are assigned to fffOSIO. Also added to fffOSIO are counts for devices not found in the configuration table. They are assigned device type and class values of *UNKNOWN.

2.2.3 Job Suspend Processing

An observation in the DETAIL level of the VSE/POWER Job Activity File (PWRPJB) contains the accumulated statistics of all steps making up a single job. The observation is constructed by merging data from the VSE/POWER Execution and Reader Account records. Jobs which are not completed at the time the data was prepared are placed in the Job Suspend File (PWR_PJ) until the balance of the job is found in subsequent database updates or a specified amount of time has passed.

Please note that the list and punch account records are not added to the Job Activity File. VSE/POWER does not provide a record indicating the completion of all output activity for a job.

Also missing from the VSE/POWER data is a record indicating job completion. CA MICS therefore must suspend the last executed job in each partition. When activity is next detected in that partition, the job is released from suspend. Jobs built only from reader record data will be suspended if reader processing terminated normally.

Users of the CA MICS Accounting and Chargeback Product should note that jobs held in suspend will not be charged until they are released. However, the steps for these jobs will be immediately placed into the VSE/POWER Program Activity File (PWRPPG) and charged normally.

Two variables are provided in the Job Activity and Job Suspend Files to allow you to determine the status of each job in the suspend process:

```
PJBMASK  - Record Construction Audit Mask
PJBSUSPN - Job Suspend Flag
```

The mask indicates whether reader data, step data, or both have been used in constructing the observation. The variable is character, two bytes long. The first byte contains an R if reader data was used, otherwise a period (.). Likewise, the second byte contains an S for step data or a period (.).

PJBSUSPN is numeric and contains the age of the suspended job. When first suspended, the value is set to one. Each DAILY update processed without further activity in that partition causes the value to increase by one. If the value becomes greater than the suspend limit specified in the PWROPS unit PARMS member, the job is released to the Job Activity File. When released, the value of PJBSUSPN is set to minus one to indicate that the job had been previously suspended.

If additional activity is detected for a suspended job, but the job is still the last to process in the partition, it will be resuspended with an age of one. As a result, long-running multistep jobs may remain suspended for longer than the suspend limit.

2.2.4 Analyzing Long-Running Jobs

Long-running jobs present an analysis challenge when the data source does not provide a means of checkpointing resource statistics. The situation exists for VSE/POWER Accounting data, since the finest data resolution available is the step end.

The lack of an interval measurement facility means that all of the resource statistics for a long-running job may be lost due to a system failure. This can seriously impact sites that perform chargeback. The only solution is to find some other means for charging these jobs. For example, CICS charges could be derived from internal measurement sources such as CMF.

The VSE/POWER Device Activity File (PWRPDA) provides a breakdown of device activity by hour. The first step in producing this file is to apportion the activity of a single job on a device across each hour that it spanned. This apportionment is based exclusively on the elapsed execution time. Clearly, a job such as CICS will have variations in load that will not be reflected in such an apportionment scheme. There is no easy or certain solution for this problem and it impacts any attempt to apportion the resources of long-running jobs.

Perhaps the most serious problem results from the lack of complete timestamp information in the VSE/POWER Account data records. The following fields are provided in the record to track the start and end of the measured event:

```
ACDATE - Date in the format defined for the system
ACSTRT - Start time
ACSTOP - Stop time
```

The ACDATE is the stop date. This means you can accurately build the ENDTS variable. If the job execution time is available, the STARTTS can be derived from the ENDTS.

However, this is not the case for VSE/POWER Version 2.1.0. For this data the start date must be estimated based on the relationship of the start time and end time. If start time is later than end time, then the start date must be at least one day earlier than the end date. This is a poor guess and can only detect some running jobs, but it is the only technique available.

2.3 Accounting and Chargeback Interface

The CA MICS Analyzer Option for VSE/POWER provides data elements that the CA MICS Accounting and Chargeback product can use to bill data center resource consumers. The elements that can be charged for are:

CHARGING ELEMENT	CHARGING ELEMENT DESCRIPTION
PJBCPUNI	POWER JOB CPU Instructions Executed
PJBOVHNI	POWER JOB CPU Overhead Instructions
PJBCPUTM	POWER JOB CPU Processor Time
PJBOVHTM	POWER JOB CPU Overhead Time
PJBTOHRS	POWER JOB Tape Occupancy Time
PJBVKHRS	POWER JOB Virtual KCore Hours
PJBKDSIO	POWER JOB DASD SIOs Issued
PJBKOSIO	POWER JOB OTHER SIOs Issued
PJBKTSIO	POWER JOB TAPE SIOs Issued
PJBKSSIO	POWER JOB SPOOL SIOs Issued
PJBKSIO	POWER JOB TOTAL SIOs Issued
PJBKSPPR	POWER JOB Lines Spooled
PJBSPPU	POWER JOB Cards Spooled
PJBSPPG	POWER JOB Pages Spooled
PJBSYSIN	POWER JOB INPUT Cards Read
PPGCPUNI	POWER STEP CPU Instructions Executed
PPGOVHNI	POWER STEP CPU Overhead Instructions
PPGCPUTM	POWER STEP CPU Processor Time
PPGOVHTM	POWER STEP CPU Overhead Time
PPGTOHRS	POWER STEP Tape Occupancy Time
PPGVKHRS	POWER STEP Virtual KCore Hours
PPGKDSIO	POWER STEP DASD SIOs Issued
PPGKOSIO	POWER STEP OTHER SIOs Issued
PPGKTSIO	POWER STEP TAPE SIOs Issued
PPGKSSIO	POWER STEP SPOOL SIOs Issued
PPGKSIO	POWER STEP TOTAL SIOs Issued
PPGKSPPR	POWER STEP Lines Spooled
PPGSPPU	POWER STEP Cards Spooled
PPGSPPG	POWER STEP Pages Spooled
POACNLR	POWER SPOOL Control Logical Records
POAEPGE	POWER SPOOL Extra Pages Printed
POAPGE	POWER SPOOL Pages Printed
POATKOBL	POWER SPOOL Tracks or Blocks Storage
POAENLR	POWER SPOOL Extra Logical Writer Record
POANLR	POWER SPOOL Logical Writer Records

When the CA MICS Accounting and Chargeback product produces invoices for data center users, those that have used VSE resources will be charged for their resources according to an algorithm that uses the data available from these elements. Typically, installations charge for time used (machine and operator), output (lines printed, lines punched, etc.), and machine time used to produce results to output (SIOs, etc.).

2.4 Capacity Management Interfaces

The CA MICS Analyzer Option for VSE/POWER does not have a formal interface to the CA MICS Capacity Planning or Performance Management components.

Elements that users of these products might find useful for ad hoc analyses include:

```
PDASIOCT - Device SIO Count
PJBCPUTM - Program Processor Time
PJBEXCTM - Job Execution Time
PJBGRPRC - Job Group Received
PJBINCLS - Job Input Class
PJBMXMUS - Maximum Step Memory Used
PJBMXTAP - Maximum Tape Devices Used
PJBOSIO  - Other SIOs Issued
PJBOVHTM - Overhead Time
PJBPCSEX - Pct Jobs Exceeding Target
PJBPCSMS - Pct Jobs Missing Target
PJBPCSMT - Pct Jobs Meeting Target
PJBSSIO  - Spool SIOs Issued
PJBTARTM - Job Turnaround Target Time
PJBTSIO  - Tape SIOs Issued
PJBTURTM - Job Turnaround Time
POAENLR  - Extra Logical Writer Records
POANLR   - Logical Writer Records
PPGALLTM - All Bound Time
PPGCPUTM - Program Processor Time
PPGDSIO  - DASD SIOs Issued
PPGEXCTM - Program Execution Time
PPGMEMUS - Kilobytes Memory Used
PPGOSIO  - Other SIOs Issued
PPGOVHTM - Overhead Time
PPGSIO   - Total SIOs Issued
PPGSPPG  - Pages Spooled
PPGSSIO  - Spool SIOs Issued
PPGTSIO  - Tape SIOs Issued
PRABUFRV - Buffers Received
PRABUFTR - Buffers Transmitted
PRAERROR - Errors Count
PRAINRSP - Invalid Responses
PRASESSN - Sessions
PRATIMOT - Timeouts
PRATRMER - Terminal Errors
```

The following elements are available only if your site has activated them:

PJBTOHRS - Tape Occupancy Hours (OFF)
PJBVKHRS - Virtual Kcore Hours (OFF)
PPGTOHRS - Tape Occupancy Hours (OFF)
PPGVKHRS - Virtual Kcore Hours (OFF)

Chapter 3: REPORTS

The CA MICS Analyzer Option for VSE/POWER produces reports using the batch and interactive reporting facilities of CA MICS. The types of reports produced are categorized as standard analysis, MICF inquiry, management objective, and exception reports.

CA MICS standard analysis reports provide a concise representation of an installation's workload, resource use, and response to the workload when data is not in a form suitable for a management objective or exception report.

MICF inquiries are precomposed printer reports and color graphics that are accessed via the CA MICS Information Center Facility (MICF). MICF inquiries help you produce meaningful reports from the data in the CA MICS database quickly and easily and provide you with the flexibility to code and save your own report formats.

Management objective reports provide a concise, graphic or tabular representation of your installation's processing objectives and how well they have been met.

Exception reports provide a concise, integrated, and itemized list of the problems impacting an installation's effectiveness in terms of availability, service, workload, standards, security, and performance. Exception reports are discussed in Chapter 4 of this guide.

While almost all of the reports can be used as distributed, most should be modified to meet your exact requirements. Section 3.4. of the CA MICS Analyzer Option for VSE/POWER Guide contains a description of macros available to customize the management objective reports. If you require more extensive modification, review Chapter 10, Modification.

This section contains the following topics:

3.1 Standard Analysis Reports

Instead of providing standard analysis reports, the CA MICS Analyzer Option for VSE/POWER provides MICF inquiries in the catalog group WORKLOAD. See Section 3.2 for information on MICF inquiries.

3.2 MICF Inquiries

The MICF inquiries provided with the CA MICS Analyzer Option for VSE/Power are listed in Figure 3-1.

VSE/POWER ANALYSIS PRODUCT

MICF DISTRIBUTED INQUIRIES

REPORT CLASS	REPORT NAME	REPORT IDENTIFIER			DATABASE FILE
		Color Graphics	List	Printer Graphics	
Except	VSE/POWER Top 20 Resource Consumers		PWRLD4		PWRPJB
General	VSE/POWER Jobs Terminated with PFLUSH		PWRLD1		PWRPJB, PWR_PJ
	VSE/POWER Jobs Terminated by RDREXIT		PWRLD2		PWRPJB, PWR_PJ
	VSE/POWER Step Term Code Analysis		PWRLD3		PWRPPG
Service	VSE/POWER Job Initiation Objective			PWRPD1	PWRPJB
	VSE/POWER Job Turnaround Objective			PWRPD2	PWRPJB
	VSE/POWER Job Group Turnaround Received			PWRPD4	PWRPJB
Workload	VSE/POWER Daily Job/Class Activity	PWRCD1			PWRPJB, PWR_PJ
	VSE/POWER Special Forms Use by Shift		PWRLD5		PWRPOA
	VSE/POWER Partition Activity Audit			PWRPD3	PWRPJB, PWR_PJ
Management Objectives	VSE/POWER Job Initiation Objective VSE/POWER Job Turnaround Objective VSE/POWER Job Group Turnaround Objective see Section 3.3 of this guide for details about these reports	PWRMD0, PWRMW0, PWRMM0			PWRPJB

Figure 3-1. MICF Distributed Inquiries, VSE/POWER Analyzer

MICF inquiry names follow the form cccptn, where:

 ccc is the three-character component identifier.

   p is the type of graphic output. Valid values are C
      (color graphics), L (printed report), or P (printer
      graphics).

   t is the report frequency. Valid values are X (detail),
      D (days), W (weeks), M (months), and Y (years).

   n is an alphanumeric character that differentiates this
      inquiry from others.

The inquiry named PWRLD4 is interpreted as:

```
                ---|||= the fourth list inquiry
                 | ||   being produced for the PWR
                 | ||   product at the DAYS timespan
                 | ||= may be run daily
                 | |= a list inquiry
                 |= a VSE/POWER product inquiry
```

The reports shown in Figure 3-1 are described in the
following sections:

```
    1 -  VSE/POWER Top 20 Resource Consumers
    2 -  VSE/POWER Jobs Terminated with PFLUSH
    3 -  VSE/POWER Jobs Terminated by RDREXIT
    4 -  VSE/POWER Step Term Code Analysis
    5 -  VSE/POWER Job Initiation Objective
    6 -  VSE/POWER Job Turnaround Objective
    7 -  VSE/POWER Job Group Turnaround Received
    8 -  VSE/POWER Daily Job/Class Activity
    9 -  VSE/POWER Special Forms Use by Shift
   10 -  VSE/POWER Partition Activity Audit
```

3.2.1 VSE/POWER Top 20 Resource Consumers

The VSE/POWER Top 20 Resource Consumers inquiry ranks the top 20 POWER jobs with the highest resource utilization on each VSE system by elapsed execution time, CPU time, and start I/O counts.

INTENDED USE:

The report can be used to identify POWER jobs that are utilizing the greatest amount of computer resources inefficiently and/or inappropriately. Operations managers can compare any previous days Top 20 Resource Consumers reports to see if the same jobs are listed on a daily basis. This can be an indication that some performance improvement may be needed to the jobs identified. Jobs listed on this report can also be the reason for bottlenecks, which prevent other jobs from getting the resources required to execute.

REPORT FORMAT:

This inquiry produces three tabular reports for each VSE system with rankings as follows: job duration (sum of the elapsed step execution times), CPU time (the amount of processor time used by the job), and SIO count (the number of start I/O requests issued by the job). Each report is grouped by system identifier and grouped by the columns described below:

Execution Time	The sum of the elapsed execution times for each step of the job
CPU Time	The amount of processor time consumed by the job
Total SIOs	The total number of start I/O requests issued by the job

Additional columns listed are:

Job Name	The job name identified on the job card
POWER Job	The job number assigned by IBMs VSE/POWER product
Start Time	The date and the time the job started executing
POWER Term Code	The job termination code. The values are based on the codes established by IBMs VSE/POWER product.

A sample of this report is shown in Figure 3-2.

```
INQUIRY ID:

    PWRLD4

DATA SOURCE:

    DETAIL.PWRPJB

DATA ELEMENTS USED:

    JOB      - Job Identification
    PWRJOBNO - POWER Job Number
    STARTTS  - Start Time Stamp
    PJBPTERM - POWER Termination Code
    PJBCPUTM - Program Processor Time
    PJBEXCTM - Job Execution Time
    PJBSIO   - Total SIOs Issued
```

```
                                  VSE/POWER Top 20 Resource Consumers - Job Duration
                                                  ABC Company

 INQUIRY:  PWRLD4                                                                                    RUN DATE:  DDMONYY

------------------------------------------------------ System Identifier=VSE1 ------------------------------------------------------

               Execution    Job          POWER              Start Time    POWER           CPU Time      Total SIOs
               Time         Name         Job Number                       Term Code

                 1:20:33    SK1016B4          7481    DDMONYY:23:43:22    10            0:19:52.48           42024
                 0:42:42    SK101662          7658    DDMONYY:01:38:13    10            0:00:49.02           17771
                 0:40:07    SK1016D3          7147    DDMONYY:20:40:18    10            0:00:00.90             533


------------------------------------------------------ System Identifier=VSE2 ------------------------------------------------------

               Execution    Job          POWER              Start Time    POWER           CPU Time      Total SIOs
               Time         Name         Job Number                       Term Code

                20:14:23    SK101697            35    DDMONYY:06:21:21    10            0:26:09.35          271226
                 5:29:28    POWER/VS             0    DDMONYY:00:22:26    10            1:47:39.88         1760414
                 5:19:21    SK101650          1956    DDMONYY:00:32:12    10            0:00:04.45             907
```

Figure 3-2. VSE/POWER Top 20 Resource Consumers Report

3.2.2 VSE/POWER Jobs Terminated with PFLUSH

The VSE/POWER Jobs Terminated with PFLUSH inquiry produces a tabular report of jobs that have been cancelled with the PFLUSH console command on each VSE system.

INTENDED USE:

The report can be used to audit operator activity and to see how many system resources were expended on jobs that were subsequently cancelled using the PFLUSH command. Jobs that are executing in a loop are usually cancelled if executing too long. Jobs cancelled due to a loop can be identified by noting the time the job started and observing a very high CPU time and system I/O request count.

REPORT FORMAT:

This inquiry produces a tabular report for each VSE system. The report is grouped by system identifier with the following columns:

JOB	The job name identified on the job card
PWRJOBNO	The job number assigned by IBM's VSE/POWER product
STARTTS	The date and start time when the job started executing
PJBCPUTM	The amount of processor time consumed by the job
PJBSIO	The total number of I/O requests issued by the job

A sample of this report is shown in Figure 3-4.

INQUIRY ID:

PWRLD1

DATA SOURCE:

DETAIL.PWRPJB

DATA ELEMENTS USED:

JOB - Job Identification
PWRJOBNO - POWER Job Number
STARTTS - Start Time Stamp
PJBCPUTM - Program Processor Time
PJBSIO - Total SIOs Issued

VSE/POWER Jobs Terminated with PFLUSH Command
ABC Company

INQUIRY: PWRLD1 RUN DATE: DDMONYY

System Identifier=VSE2

JOB	PWRJOBNO	STARTTS	PJBCPUTM	PJBSIO
SK2751H1	6698	DDMONYY:10:30:53.00	0:00:05.82	1675
SK102068	6742	DDMONYY:11:50:12.00	0:00:12.21	22819
SKB4R268	6850	DDMONYY:14:10:20.00	.	.

System Identifier=VSE3

JOB	PWRJOBNO	STARTTS	PJBCPUTM	PJBSIO
SK0419H1	7124	DDMONYY:20:04:53.00	0:00:00.91	673
SK2121H1	7164	DDMONYY:21:41:01.00	0:00:10.71	3354
SK2121H1	7193	DDMONYY:22:23:07.00	0:00:11.33	3841

Figure 3-4. VSE/POWER Jobs Terminated with PFLUSH

3.2.3 VSE/POWER Jobs Terminated by Exit Routines

The Jobs Terminated by Exit Routines inquiry produces a list, by system identifier, of jobs that were not processed because data center JCL coding standards were not met.

INTENDED USE:

The report can be used to investigate jobs that are failing frequently because they are not following JCL coding standards. These jobs usually represent wasted resources because they will need to be rerun. Operations manager can notify users submitting the jobs that the JCL in their jobs must be corrected to follow standards to prevent future JCL problems.

REPORT FORMAT:

This inquiry produces a tabular report for each VSE system. The report is grouped by system identifier with the following columns:

JOB	The job name identified on the job card
PWRJOBNO	The job number assigned by IBM's VSE/POWER product
RDRTS:	The date and the time when the job was submitted.
PWRJUSER	User information obtained from the user-assigned User Info programmer name field in the job card.

A sample report is shown in Figure 3-5.

INQUIRY ID:

PWRLD2

DATA SOURCE:

DETAIL.PWRPJB
DETAIL.PWR_PJ

DATA ELEMENTS USED:

JOB - Job Identification

PWRJOBNO - POWER Job Number
RDRTS - Reader Time Stamp
PWRJUSER - POWER JOB User Information

```
                                        Daily POWER Jobs Terminated by RDREXIT
                                                    ABC Company

INQUIRY:  PWRLD2                                                                                                RUN DATE:  DDMONYY

------------------------------------------------------ System Identifier=VSE1 ------------------------------------------------------

                              JOB             PWRJOBNO                  RDRTS    PWRJUSER

                              SK101678             3294    DDMONYY:03:11:55.00    16078KVTSTUPAL


------------------------------------------------------ System Identifier=VSE2 ------------------------------------------------------

                              JOB             PWRJOBNO                  RDRTS    PWRJUSER

                              SK7204H2              675    DDMONYY:00:21:57.00    01SALZ7204UTRL
                              SK101650                4    DDMONYY:03:32:53.00    16SALKICSSTATL


------------------------------------------------------ System Identifier=VSE3 ------------------------------------------------------

                              JOB             PWRJOBNO                  RDRTS    PWRJUSER

                              SK7204H2              257    DDMONYY:11:10:35.00    01SALZ720414KL
                              AUTONAME              391    DDMONYY:14:35:50.00
                              AUTONAME              418    DDMONYY:15:52:08.00
                              SK7204H2              602    DDMONYY:21:21:03.00    01SALZ7204UTRL
                              SK2121H1              716    DDMONYY:02:08:37.00    01SALG212103K
                              SK2121H1              717    DDMONYY:02:08:37.00    01SALG212103K
```

Figure 3-5. VSE/POWER Jobs Terminated by Exit Routines

3.2.4 VSE/POWER Step Term Code Analysis

The VSE/POWER Step Term Code Analysis inquiry is a tabular report of the frequency of VSE termination codes by system identifier.

INTENDED USE:

The report can be used to analyze the frequency of VSE termination codes for all jobs that executed on a VSE system. For example if the percentage of VSE termination code 10 is 94% for all termination codes represented, this indicates a high percentage of the jobs ran to normal completion. However, if the percentage of VSE termination code 24 (operator cancel) is above a pre-determined threshold for cancellations, the reason for the cancellations should be investigated.

REPORT FORMAT:

This inquiry produces a tabular report by system identifier (SYSID) and date:

PPGVTERM	The VSE step termination code
FREQUENCY	The number of occurrences of this termination code for the SYSID and date
PERCENT	The percent of total step terminations that this termination code represents
CUMULATIVE FREQUENCY	The sum of the frequency counts this termination code represents plus the frequency counts of other reported termination codes
CUMULATIVE PERCENT	The sum of the percent count this termination code represents plus the percent counts of other reported termination codes

A sample of this report is shown in Figure 3-6.

INQUIRY ID:

PWRLD3

DATA SOURCE:

DETAIL.PWRPPG

DATA ELEMENTS USED:

PPGVTERM - VSE Termination Code

VSE/POWER Step Term Code Analysis
ABC Company

INQUIRY: PWRLD3 RUN DATE: DDMONYY

System Identifier=VSE1 DATE=DDMONYY

VSE Termination Code

PPGVTERM	FREQUENCY	PERCENT	CUMULATIVE FREQUENCY	CUMULATIVE PERCENT
FF	5	1.1	5	1.1
00	3	0.7	8	1.8
10	414	94.7	422	96.6
23	11	2.5	433	99.1
24	4	0.9	437	100.0

System Identifier=VSE2 DATE=DDMONYY

VSE Termination Code

PPGVTERM	FREQUENCY	PERCENT	CUMULATIVE FREQUENCY	CUMULATIVE PERCENT
FF	18	4.8	18	4.8
00	2	0.5	20	5.4
10	318	85.5	338	90.9
23	27	7.3	365	98.1
24	6	1.6	371	99.7
25	1	0.3	372	100.0

Figure 3-6. VSE/POWER Step Term Code Analysis

3.2.5 VSE/POWER Job Initiation

The VSE/POWER Job Initiation Report quantifies the total POWER job activity on an hourly basis for the day and VSE system. The total activity is broken down by whether jobs met, exceeded, or missed their service objectives.

INTENDED USE:

This report can be used for evaluating the throughput and turnaround service for all POWER job activity and its relationship to the system load.

REPORT FORMAT:

This inquiry produces a chart with a vertical and horizontal axis. The horizontal axis represents the hour of the day and the vertical axis charts the number of batch jobs initiated that received the service as indicated: E for exceeded, H for hit, and M for missed. The management objective is shown by the dashed horizontal reference line printed across the chart. A chart is generated for each VSE system.

FREQUENCY	The number of jobs started during the hour
EXCEED	The proportion of jobs exceeding their service objectives
HIT	The proportion of jobs meeting their service objectives
MISS	The proportion of jobs missing their service objectives
HOUR	The hour of the day
REPTDATE	The date being reported on

A sample of this report is shown in Figure 3-7.

INQUIRY ID:

PWRPD1

DATA SOURCE:

DETAIL.PWRPJB

DATA ELEMENTS USED:

HOUR - Hour of Day
JOBGROUP - Job Group
PJBSRVMS - Jobs Missing Target
PJBSRVEX - Jobs Exceeding Target
PJBSRVMT - Jobs Meeting Target

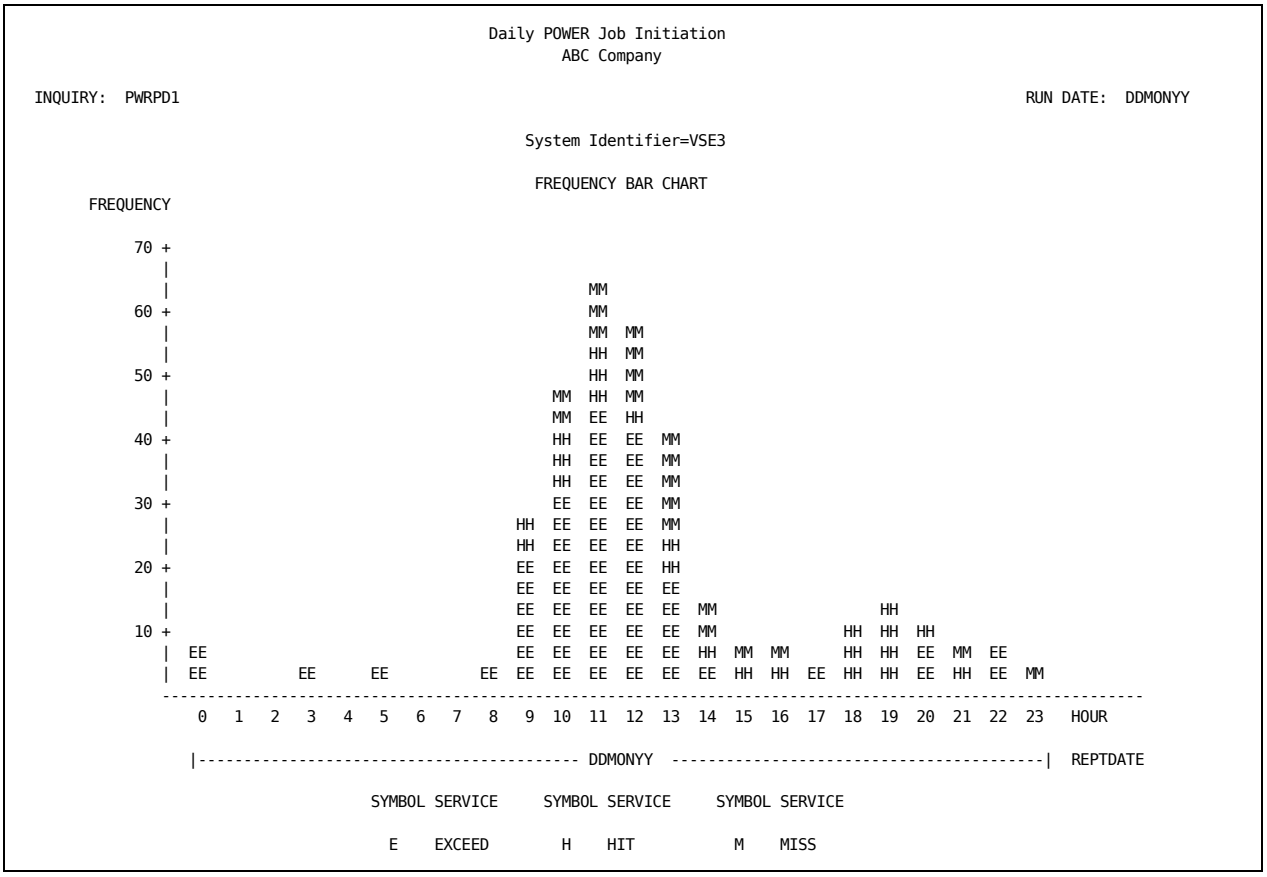


Figure 3-7. VSE/POWER Job Initiation Objective

3.2.6 VSE/POWER Job Turnaround Objective

The VSE/POWER Job Turnaround Objective Report shows the turnaround service received for all VSE/POWER jobs that started on an hourly basis for the identified day and computing system.

INTENDED USE

This report can be used for evaluating the turnaround service for all POWER job activity and its relationship to the system load. This report is useful to operation managers, capacity planners, and performance analysts; it helps them evaluate the quality of batch service provided. While similar to the Job Initiation Report, it provides a clearer assessment of quality of service.

REPORT FORMAT

This inquiry produces a chart with a vertical and horizontal axis. The horizontal axis represents the hours of the day, and the vertical axis charts the percentage of batch jobs initiated in the hour that received the requested turnaround time. The management objective for jobs that received the requested turnaround is indicated by the dashed horizontal reference line printed across the chart.

PERCENT	The percent of jobs which began executing during the hour which met or exceeded their service objectives
HOUR	The hour of the day that observations are recorded for
REPTDATE	The date being reported on

A sample report is presented in Figure 3-8.

INQUIRY ID:

PWRPD2

DATA SOURCE:

DETAIL.PWRPJB

DATA ELEMENTS USED:

HOUR - Hour of Day
JOBGROUP - Job Group

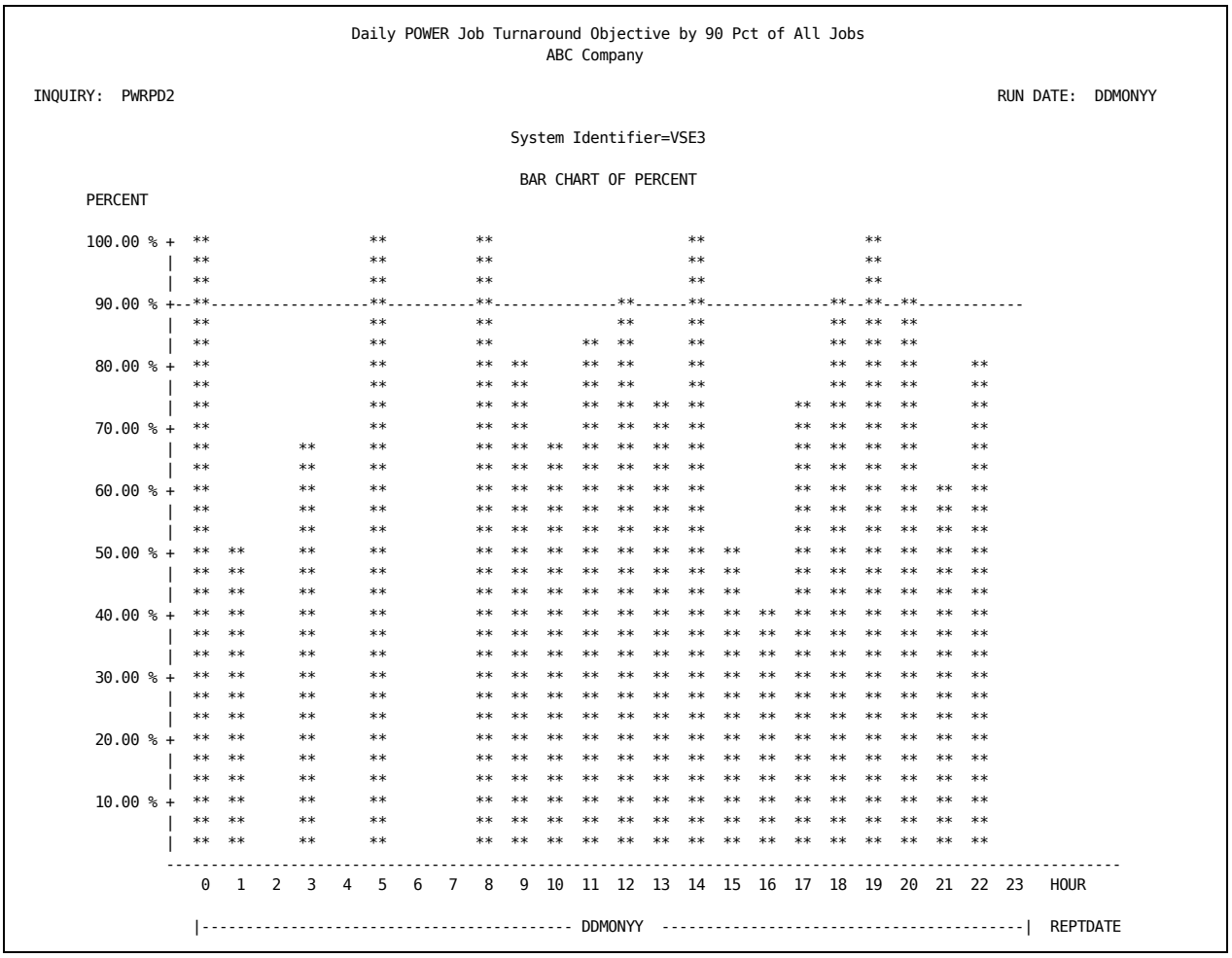


Figure 3-8. VSE/POWER Job Turnaround Objective

3.2.7 VSE/POWER Job Group Turnaround Received

The VSE/POWER Job Group Turnaround Received Report displays the batch service received for the specified job group on an hourly basis for the identified day and computing system.

INTENDED USE

This report can be used for evaluating the turnaround service for all POWER job activity within specified groups and its relationship to the system load. This report is useful to operation managers, capacity planners, and performance analysts; it helps them evaluate the quality of batch service provided. While similar to the Job Initiation Report, it provides a clearer assessment of quality of service.

REPORT FORMAT

This inquiry produces a chart with a vertical and horizontal axis. The horizontal axis represents the hours of the day, and the vertical axis charts the percentage of batch jobs for the reported job group that were initiated during the hour and received the requested turnaround time. The management objective to be tracked is, for each hour, the percentage of batch jobs within this job group for which the requested turnaround was met or exceeded.

The reference line represents the turnaround objective. Its value is set within the inquiry, but may be changed by the user.

The sample report presents information by SYSID, job group, and date. The fields reported are:

PERCENT:	The percent of jobs which began executing during that hour which met or exceeded their service objective
HOUR:	The hour of the day that observations are recorded
REPTDATE	The date being reported

A sample report is shown in Figure 3-9.

INQUIRY ID:

PWRPD4

DATA SOURCE:

DETAIL.PWRPJB

DATA ELEMENTS USED:

SYSID - System Identifier
HOUR - Hour of Day
JOBGROUP - Job Group

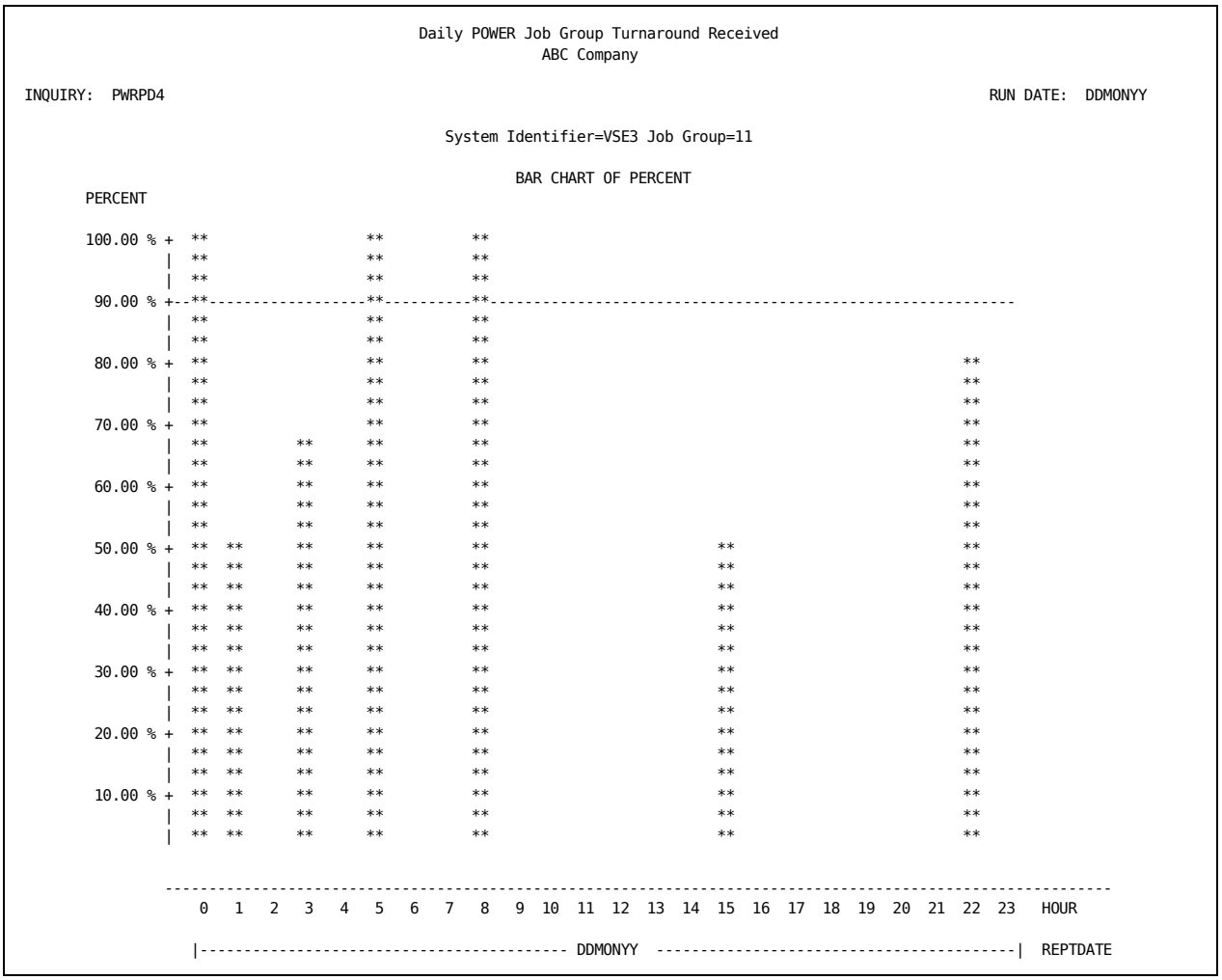


Figure 3-9. VSE/POWER Job Group Turnaround Received

3.2.8 VSE/POWER Daily Job/Class Activity

The VSE/POWER Daily Job/Class Activity Report quantifies the total job activity on an hourly basis in each job class for the identified day and computing system.

INTENDED USE

This report can be used for evaluating how the job class structure is being used. It can tell you which jobs are being run in the job class that allows the longest time to run and whether most jobs in the same job class are being run at a particular time. Answers to these questions can help the operations and performance staff evaluate overall job activity.

REPORT FORMAT

The report's output is a stacked vertical bar chart.

The sample report presents information by SYSID, job class, hour, and date. The fields reported are:

HOUR	The hour of the day that observations are recorded
JOBCLASS	Job class in which the job ran

This is a color graphic report, so it cannot be shown in this guide.

INQUIRY ID:

PWRCD1

DATA SOURCE:

DETAIL.PWRPJB

DATA ELEMENTS USED:

SYSID - System Identifier
HOUR - Hour of Day
JOBCLASS - Job Input Class

3.2.9 VSE/POWER Special Forms Use by Shift

The VSE/POWER Special Forms Use by Shift inquiry produces a frequency distribution, by form number, of the number of times each form number was used during a user-defined period of time (shift) for the preceding day.

INTENDED USE:

The report provides an analysis of the amount of usage of the different available forms. Examination of this report can lead to the elimination of unused forms or to an increase in the availability of additional forms. This report can also be used to identify high usage forms that might, for example, be converted to "form flash" applications in order to save on printing costs and operator interventions.

REPORT FORMAT:

This inquiry produces a tabular report by system identifier (SYSID) and date: The fields reported are:

ZONE	The zone during which the form was used
FREQUENCY	The number of times the form was used during the zone, weighted by the number of logical lines printed
ROW/COL PERCENT	The percent of total forms for this zone the form represents

A sample report is shown in Figure 3-10.

INQUIRY ID:

PWRLD5

DATA SOURCE:

DETAIL.PWRPOA

DATA ELEMENTS USED:

SYSID - System Identifier
ZONE - Time Zone
POANLR - Logical Writer Records

FORMNUM - Form Number

VSE/POWER Special Forms Use by Shift
ABC Company

INQUIRY: PWRLD5 RUN DATE: DDMONYY

System Identifier=VSE3

TABLE OF FORMNUM BY ZONE

FORMNUM(Form Number)
ZONE(Time Zone)

FREQUENCY PERCENT ROW PCT COL PCT	1	2	TOTAL
RB11	24863 4.42 100.00 11.58	0 0.00 0.00 0.00	24863 4.42
ST11	95602 17.00 37.01 44.53	162720 28.94 62.99 46.81	258322 45.94
ST12	7900 1.40 15.65 3.68	42583 7.57 84.35 12.25	50483 8.98
ST13	5301 0.94 19.09 2.47	22469 4.00 80.91 6.46	27770 4.94
7X2	79671 14.17 44.18 37.11	100666 17.90 55.82 28.96	180337 32.07
TOTAL	214712 38.18	347651 61.82	562363 100.00

FREQUENCY WEIGHTED BY POANLR (NUMBER OF LOGICAL RECORDS)

Figure 3-10. VSE/POWER Special Forms Use by Shift

3.2.10 VSE/POWER Partition Activity Audit

The VSE/POWER Partition Activity Audit Report provides a graphical view of batch activity by partition.

INTENDED USE:

The report can be used by the systems programming staff to review the effectiveness of the class structure and class-partition assignments. The operations manager might use it to track production activity for each partition.

REPORT FORMAT:

The report provides a tabular and graphical view of jobs for each VSE system. The report is grouped by system identifier, partition and date with the following columns and notations:

JOB NAME	The job name identified on the job card.
JOB NUM	The job number assigned by IBM's VSE/POWER product.
CLASS	The job input class.
STARTTM	The time when the job began executing.
ENDTM	The time when the job completed.
S	The job start time.
E	The job end time.
*	The start time and end time are the same.
+	The end time of the last available step of an incomplete (suspended) job.
< or >	A job's execution spans midnight and is continued from (if <) or on (if >) another plot.

A sample report is shown in Figure 3-11.

INQUIRY ID:

PWRPD3

DATA SOURCE:

DETAIL.PWRPJB
DETAIL.PWR_PJ

DATA ELEMENTS USED:

JOB - Job Identification
PWRJOBNO - POWER Job Number
STARTTS - Start Time Stamp
VSEPRTID - VSE Partition Identifier
ENDTS - End Time Stamp
SYSID - System Identifier
JOBCLASS - Job Input Class

```
                                VSE/POWER PARTITION ACTIVITY AUDIT
                                        ABC Company

INQUIRY:  PWRPD3                                                                RUN DATE:  05JANYY

                                SYSTEM =VSE2 PARTITION =BG DATE=04JANYY


          JOB NAME      JOB     CLASS     STARTTM     ENDTM        MIN                                  MAX
                        NUM                                        0:00                                24:00
                                                                   *-------------------------------------------*
          SK1016B4     6629     R          3:50        4:31       |        SE      |              |               |
          SK24B226     6658     X          5:59        8:30       |            S---|E             |               |
          SK071222     6661     X          8:34        8:35       |                |*             |               |
          SK104558     6664     X          8:40        8:40       |                |*             |               |
          SK071322     6666     X          8:50        8:58       |                | *            |               |
          SK101650     6669     X          9:06        9:18       |                | SE           |               |
          SK071222     6670     X          9:19        9:20       |                |  *           |               |
          SK245122     6672     X          9:25        9:37       |                |  *           |               |
          SK798053     6680     X          9:46        9:46       |                |   *          |               |
          SK071222     6684     X          9:48        9:49       |                |   *          |               |
          SK101697     6686     X          9:56        9:57       |                |   *          |               |
          SK1016B4     6693     X         10:28       10:28       |                |    *         |               |
          SK209028     6696     X         10:30       10:30       |                |    *         |               |
          SK1016B4     6697     R         10:30       10:30       |                |    *         |               |
          SK2751H1     6698     R         10:30       10:34       |                |    *         |               |
          SK798053     6702     X         10:42       10:44       |                |    *         |               |
          SK681103     6714     X         10:56       10:57       |                |     *        |               |
          SK101697     6715     X         11:06       11:07       |                |     *        |               |
          SK101697     6717     X         11:09       11:10       |                |     *        |               |
          SK101697     6718     X         11:15       11:15       |                |      *       |               |
          SK2467H2     6722     R         11:25       12:12       |                |      SE      |               |
          SK101697     6761     X         12:13       12:13       |                |       *      |               |
          SK102068     6767     X         12:28       12:28       |                |        *     |               |
          SK101697     6796     X         13:13       13:20       |                |         SE   |               |
          SK1023C0     6804     X         13:20       13:21       |                |          *   |               |
          SK257143     6822     X         13:49       13:50       |                |           *  |               |
          SK071226     6860     X         14:14       14:17       |                |           SE |               |
          SK071322     6861     X         14:17       14:25       |                |            * |               |
          SK071226     6864     X         14:25       14:36       |                |            * |               |
          SK2090H2     6880     X         14:36       14:43       |                |            * |               |
          SK2124H1     6882     R         14:43       15:05       |                |            SE|               |
          SK2124H1     6882     R         15:15       15:38       |                |              *|              |
          SK2751GH     6940     R         15:41       15:51       |                |              SE              |
          SK2460H2     6956     R         15:53       16:37       |                |               SE             |
          SK245126     7018     X         16:41       16:44       |                |               |*             |

                          LEGEND:  S=START, E=END, *=BOTH, +=JOB INCOMPLETE, < OR > =JOB CONTINUED
```

Figure 3-11. VSE/POWER Partition Activity Audit

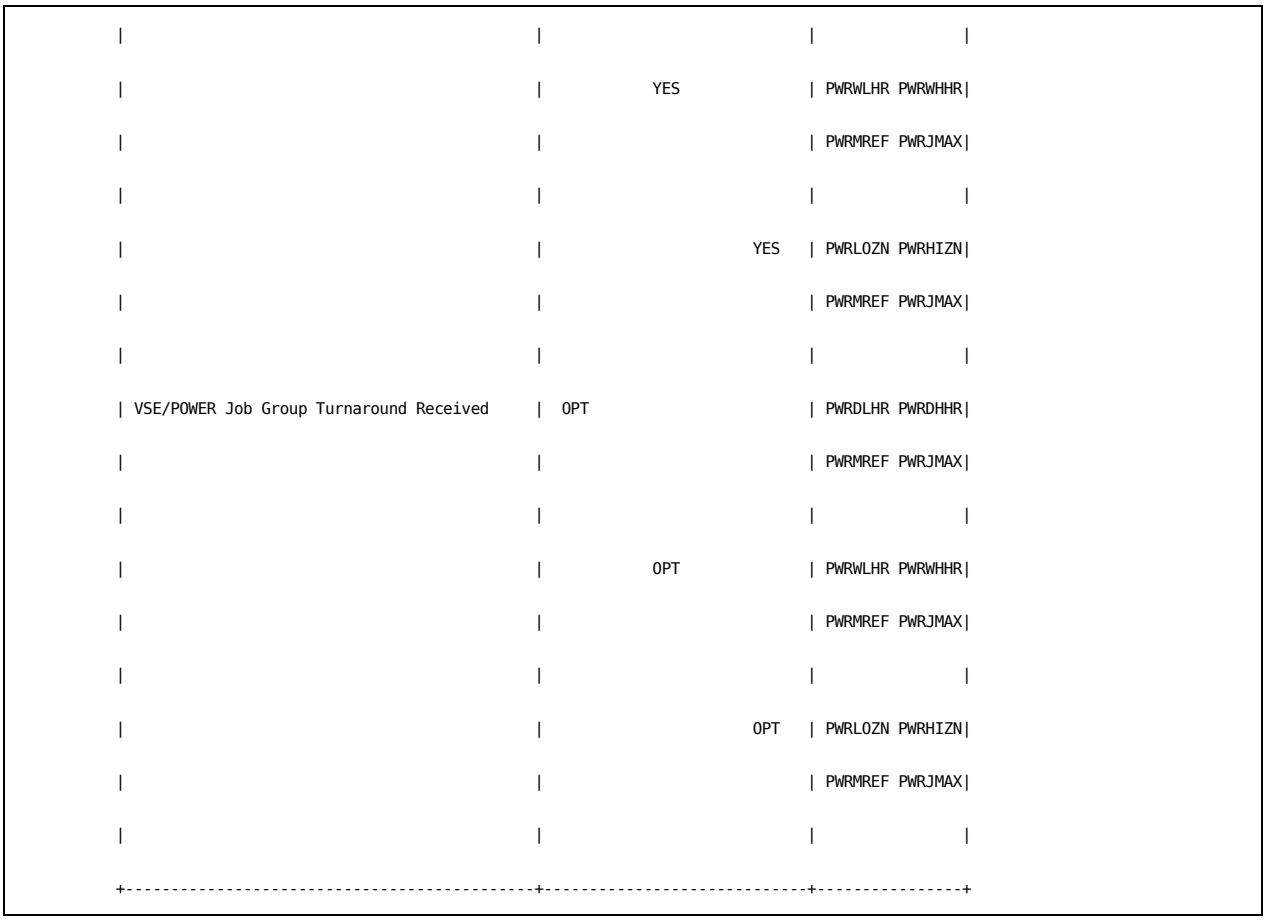
3.3 Management Objective Reports

Management objective reports are produced daily, weekly, and monthly as a standard output of the CA MICS operational jobs. The reports available from the CA MICS Analyzer Option for VSE/POWER are listed in Figure 3-13.

ANALYZER OPTION FOR VSE/POWER

MANAGEMENT OBJECTIVE REPORTS

Report Name	Time Produced			Macros Used *
	Daily	Weekly	Monthly	
VSE/POWER Job Initiation Objective	YES			PWRDLHR PWRDHHR PWRPTR PWRJSAX
		YES		PWRWLHR PWRWHHR PWRPTR PWRJSAX
			YES	PWRLOZN PWRHIZN PWRMPTR PWRMJSX
VSE/POWER Job Turnaround Objective	YES			PWRDLHR PWRDHHR PWRMREF PWRJMAX



```
|                                          |                           |               |
|                                          |           YES             | PWRWLHR PWRWHHR|
|                                          |                           | PWRMREF PWRJMAX|
|                                          |                           |               |
|                                          |                     YES   | PWRLOZN PWRHIZN|
|                                          |                           | PWRMREF PWRJMAX|
|                                          |                           |               |
| VSE/POWER Job Group Turnaround Received  |  OPT                      | PWRDLHR PWRDHHR|
|                                          |                           | PWRMREF PWRJMAX|
|                                          |                           |               |
|                                          |           OPT             | PWRWLHR PWRWHHR|
|                                          |                           | PWRMREF PWRJMAX|
|                                          |                           |               |
|                                          |                     OPT   | PWRLOZN PWRHIZN|
|                                          |                           | PWRMREF PWRJMAX|
|                                          |                           |               |
+------------------------------------------+---------------------------+---------------+
```

* - Macro descriptions are provided in Section 3.4 of this guide.
OPT - This report must be must be user tailored and activated as described in Section 3.4.

Figure 3-13. Management Objective Reports, Analyzer Option for VSE/POWER

The daily report tracks activity by hour within the day. The weekly report tracks activity by hour within day for a full week. The monthly report tracks activity by zone within month.

The following fields are used, depending on the report version:

HOUR: The hour of the day.

REPTDATE: The date reported.

ZONE: The time zone (shift).

YRMONTH: The year and month reported.

In each case, the report is produced as a vertical bar chart, with response on the vertical axis. Depending on the hour range you select, the report may be produced as a horizontal bar chart instead (see Section 3.4 for more information).

The following sections describe the management objective reports:

1 - VSE/POWER Job Initiation Objective
2 - VSE/POWER Job Turnaround Objective
3 - VSE/POWER Job Group Turnaround Received

3.3.1 VSE/POWER Job Initiation Objective

The VSE/POWER Job Initiation Objective Report shows the number of jobs started by time interval. The total number of jobs is broken down by whether each job met, missed, or exceeded its service objective.

Figure 3-14 provides a sample of the report produced by the CA MICS DAILY operational job.

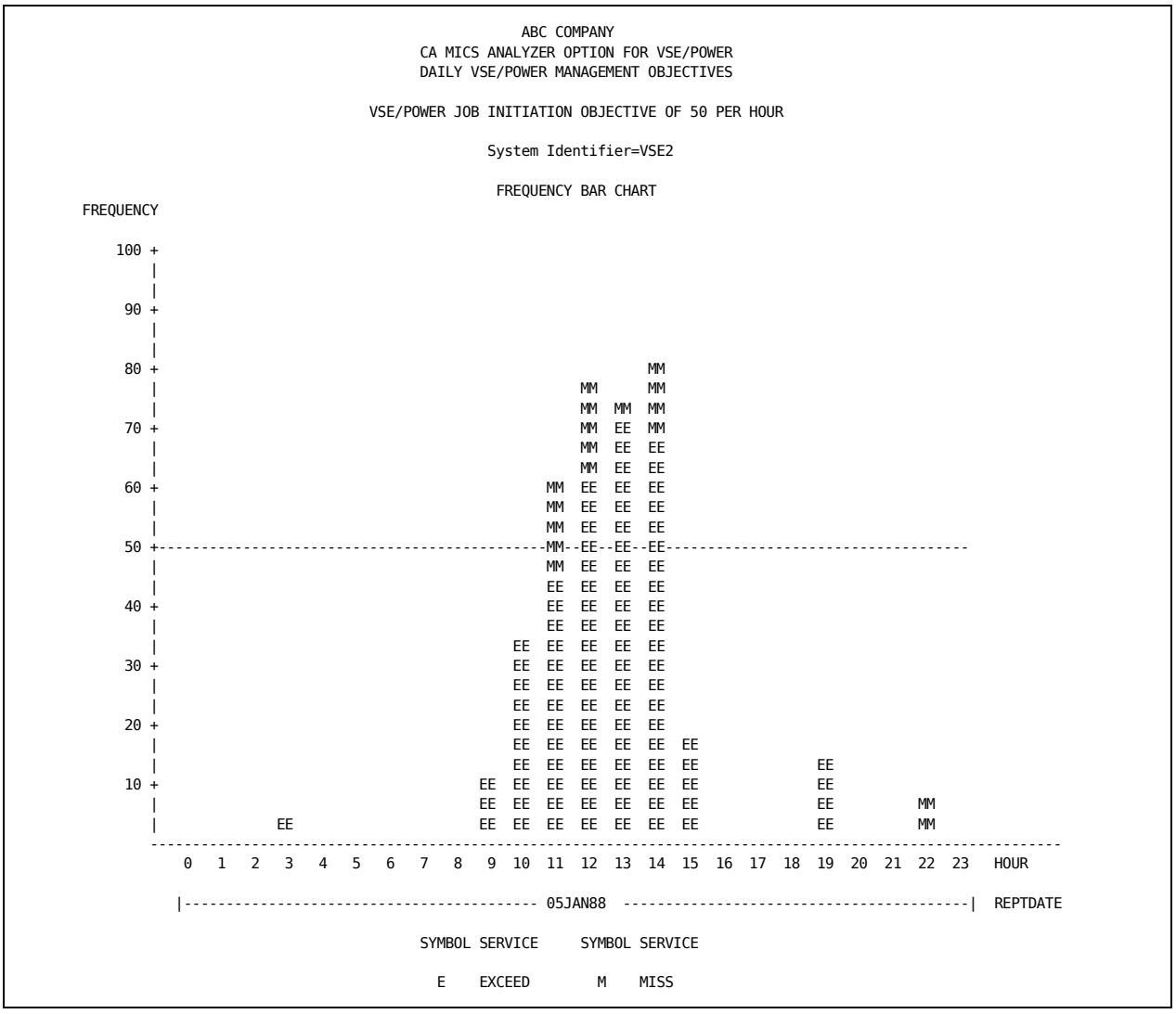


Figure 3-14. VSE/POWER Job Initiation Objective

The report is produced by system identifier with the following variables reported:

FREQUENCY: The number of jobs started during the hour.

EXCEED: The proportion of jobs exceeding their service objectives.

HIT: The proportion of jobs meeting their service objectives.

MISS: The proportion of jobs missing their service objectives.

HOUR: The hour of the day.

REPTDATE: The date being reported on.

The report objective is indicated by the reference line and is also shown in the report title. Section 3.4 provides the macro names that allow you to change the objective value.

The weekly report shows seven days worth of information and the monthly report shows the information by time zones for each of the last 12 months.

This report is useful for operations managers and capacity planners who need to evaluate the quality of service provided and its relationship to the system load.

3.3.2 VSE/POWER Job Turnaround Objective

The VSE/POWER Job Turnaround Objective Report presents a chart of jobs that were initiated and completed within the established target. A sample report is presented in Figure 3-15.

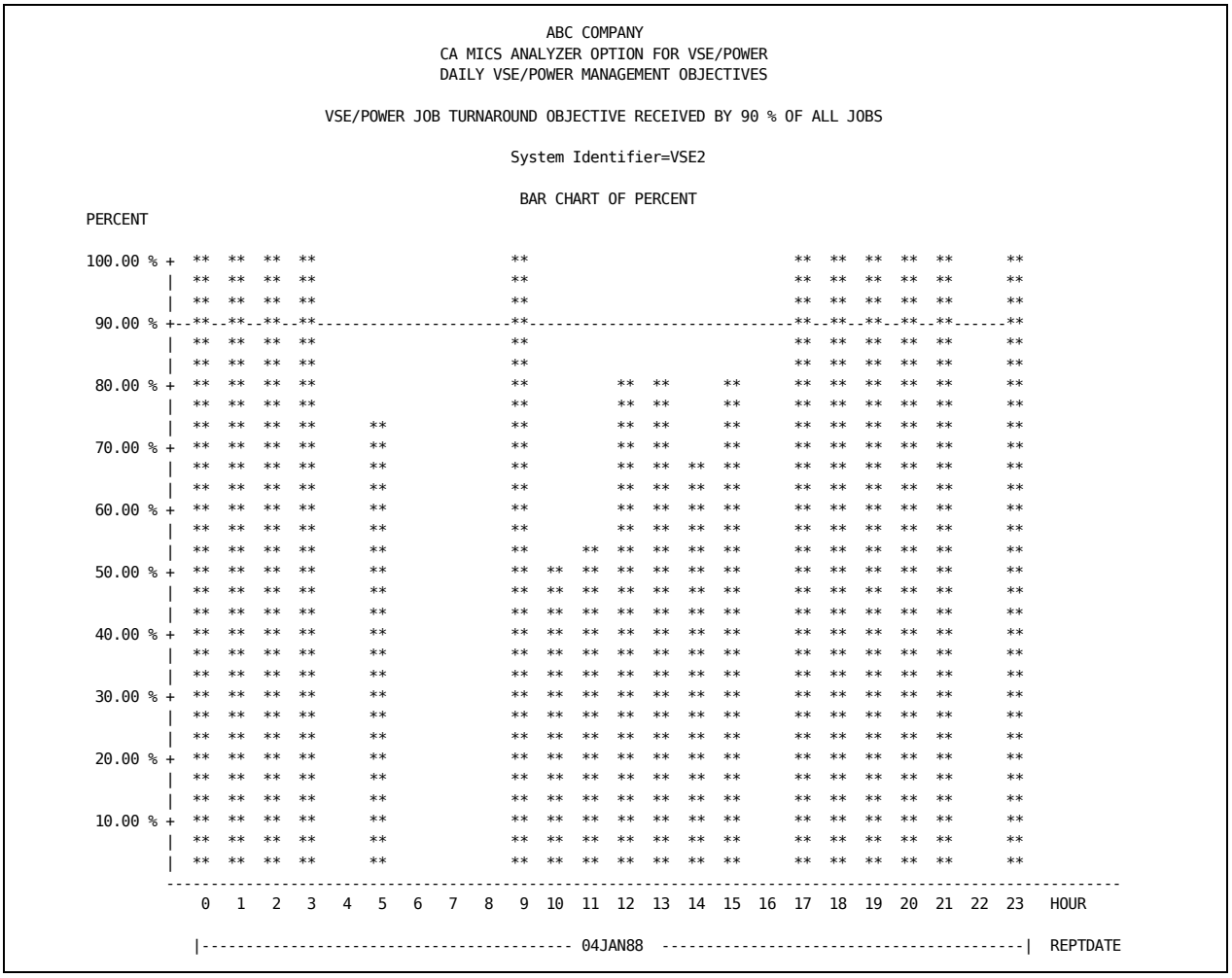


Figure 3-15. VSE/POWER Job Turnaround Objective

The report is produced by system identifier with the following variable reported:

PERCENT: The percent of jobs which began executing during that time period which met or exceeded their service objectives.

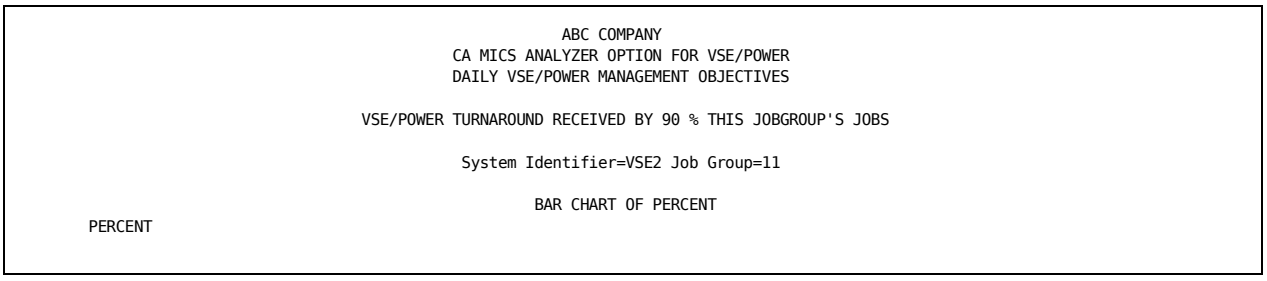
The report objective is indicated by the reference line and is also shown in the report title. Section 3.4 provides the macro names that allow you to change the objective value.

The weekly report shows seven days worth of information and the monthly report shows the information by time zones for each of the last 12 months.

Useful to operation managers, capacity planners, and performance analysts, this report helps evaluate the quality of batch service provided. While similar to the Job Initiation Objective Report, it provides a clearer assessment of service quality.

3.3.3 VSE/POWER Job Group Turnaround Received

The VSE/POWER Job Group Turnaround Received Report presents a chart of jobs that were initiated and completed within the established target for a selected job group or job groups. A sample daily report is presented in Figure 3-16.



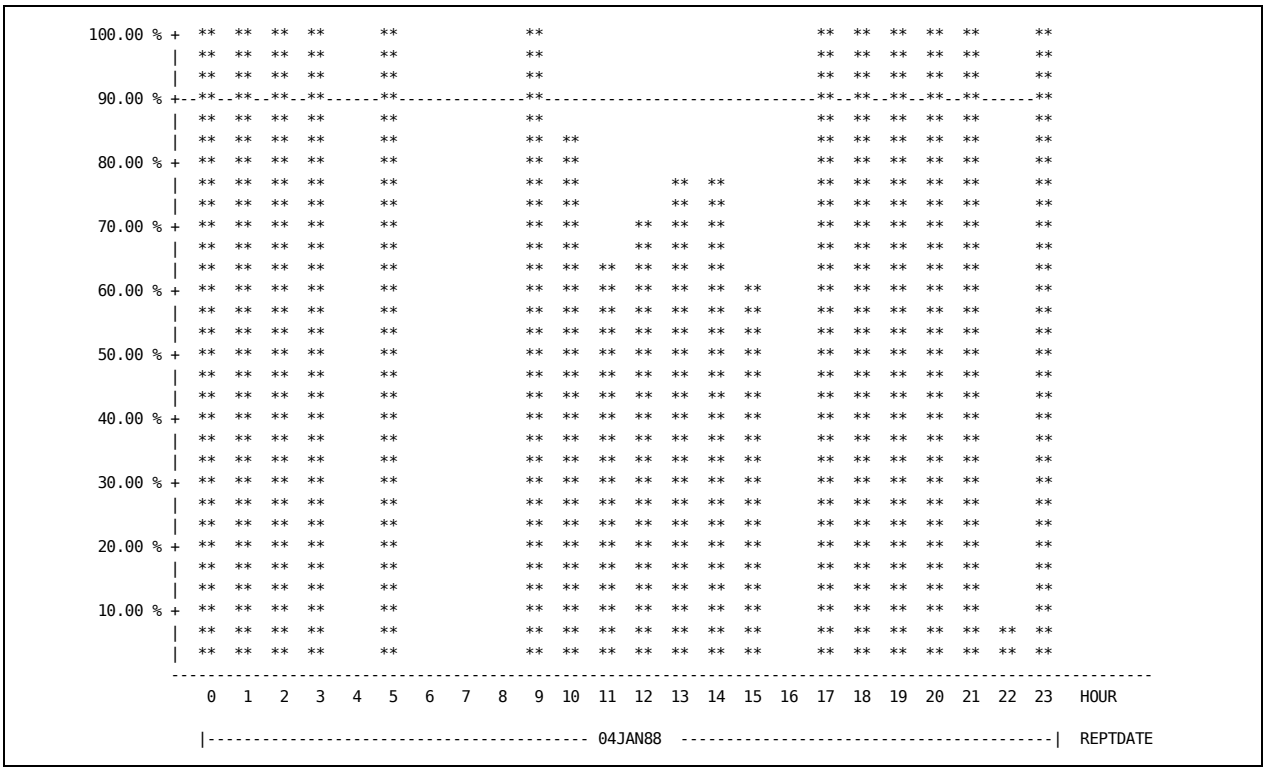

100.00 %
90.00 %
80.00 %
70.00 %
60.00 %
50.00 %
40.00 %
30.00 %
20.00 %
10.00 %
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 HOUR
04JAN88
REPTDATE

Figure 3-16. VSE/POWER Job Group Turnaround Received
The report is produced by system identifier with the following variables reported:

PERCENT: The percent of jobs which began executing during that time period which met or exceeded their service objectives.

The report objective is indicated by the reference line and is also shown in the report title. Section 3.4 provides the macro names that allow you to change the objective value.

The weekly report shows seven days worth of information and the monthly report shows the information by time zones for each of the last 12 months.

Since this report references user-defined job groups, it must be modified to select job groups of interest. The report is not produced by any of the operational jobs prior to selecting job groups. For information on how to select job groups, see Chapter 10 of this guide.

Useful to operations managers, capacity planners, and performance analysts, this report helps evaluate the quality of batch service provided. Similar to the Job Turnaround Objective Report, this report allows analysts to focus their attention on job groups of particular interest.

3.4 How to Produce Reports

The CA MICS Analyzer Option for VSE/POWER provides standard reporting capabilities. The content of the reports can be tailored to match your requirements. See Chapter 4 of this guide for information on tailoring exception reports.

SAS macros control the appearance of management objective reports. To adjust the objectives for management objective reports, use the macros contained in either the complex or unit level #PWRMOBJ member of the SOURCE data set.

Changes made to macros at the complex level affect all units. To limit changes to a particular unit, add the macro to the unit level member following the %INCLUDE SOURCE(#PWRMOBJ) statement. The complex level member, as distributed, contains default objective s.

The macros provided with the Analyzer Option for VSE/POWER are in the form of SAS macro language symbolic variable assignments. The following macros are available:

PWRPTR: The objective (reference line) for the daily and weekly Job Initiation Reports.

PWRJSAX: The axis limit for the daily and weekly Job Initiation Reports.

PWRMPTR: The objective (reference line) for the monthly Job Initiation Report.

PWRMJSX: The axis limit for the monthly Job Initiation Report.

PWRMREF: The objective (reference line) for all Job Turnaround Reports.

PWRJMAX: The axis limit for all Job Turnaround Reports.

PWRDLHR: The low hour limit for all daily reports.

PWRDHHR: The high hour limit for all daily reports.

PWRWLHR: The low hour limit for all weekly reports.

PWRWHHR: The high hour limit for all weekly reports.

PWRLOZN: The low zone limit for all monthly reports.

PWRHIZN: The high hour limit for all monthly reports.

More extensive changes to management objective reports require modifications to the report source code. These changes are discussed in Chapter 10 of this guide.

The reporting strategies available through the CA MICS I/S Management Support System are discussed in the following sections:

1 - Batch Operations
2 - CA MICS Information Center Facility II
3 - Interactive Reporting (MSAS)

3.4.1 Batch Operations

Management objective and exception reports are produced by the standard operational reporting jobs DAILYRPT, WEEKRPT, and MONTHRPT. These jobs are submitted for execution by the corresponding database update job or may be submitted manually.

Control over which report sets are produced is provided through the dynamic execution options (EXECDEF) member in the unit level PARMS library. Section 2.3.5 of the PIOM describes the EXECDEF member.

3.4.2 CA MICS Information Center Facility II

The CA MICS Information Center Facility (MICF) is a menu-based system that operates under IBM's Interactive Systems Productivity Facility (ISPF). Those familiar with ISPF commands will find that MICF menus behave like ISPF menus. The major features of MICF include the following:

- Structured inquiry composition
- Logical inquiry modification
- Direct inquiry composition
- Execution-time specifications
- Output replay
- Facilities for expert users
- Distributed inquiries (refer to Figure 3-1)
- User-written reports

For more information on using MICF to run the distributed inquiries or to create your own inquiries, the CA MICS MICF User Guide.

3.4.3 Interactive Reporting (MSAS)

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

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

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

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

To use MSAS, see Chapter 3 of this guide.

Chapter 4: EXCEPTIONS

The CA MICS Analyzer Option for VSE/POWER supports the standard CA MICS exception process.

CA MICS exception processing enables you to focus on problems impacting its effectiveness in terms of availability, service, workload, standards, security, and performance for the different areas of responsibility (e.g., TSO, IMS, VSE/POWER). A more detailed description of the exception process is presented in Chapter 2 of the CA MICS Standard Reports Guide.

The standard exceptions for this product are shown in Figure 4-1. Each standard exception test is shipped with default values. However, to make effective use of the exception process, you must evaluate your needs and modify these sample values accordingly (setting values is discussed briefly in Section 4.2 of this guide and in more detail in Section 2.4 of the CA MICS Standard Reports Guide).

Number	Severity	Management Area	Exception Description
14001	Critical	Service	Job Turnaround Time Exceeded Objective
14002	Impacting	Performance	Job Elapsed Time Exceeded Limit
14003	Impacting	Performance	Job Input Queue Time Exceeded Limit
14053	Warning	Workload	Program SIO/CPU-Second Ratio Exceeds Limit
14056	Warning	Standards	Unauthorized Program User
14057	Warning	Workload	Program Resource Consumption Exceeded Limit
14101	Warning	Performance	Degraded Printer Performance
14102	Warning	Standards	Lines Printed Exceeded Limit
14151	Impacting	Performance	RJE Excessive Errors
14152	Impacting	Performance	RJE Excessive Timeouts
14153	Impacting	Performance	RJE Excessive Invalid Transmissions

Figure 4-1. VSE/POWER Analysis Product Exceptions

This section contains the following topics:

4.1 Exception Process Overview

The CA MICS exception process consists of the exceptions themselves, a set of standard reports, CA MICS Information Center Facility (MICF) inquiries, an exception test routine, and an exception value analysis routine. Each of these is described below. A more detailed description of the process is presented in Chapter 2 of the Standard Reports Guide.

EXCEPTIONS

An exception is the occurrence of an event that merits visibility and attention. It may be an occurrence that is a distinct problem (for example, CICS ABENDED at 2:00 p.m.), one that may be a problem and requires further research (for example, a TSO user overloaded the system from 1:00 to 1:30 p.m.), or one that represents a standard, security, or audit violation (for example, user XYZ is not authorized to use PDZAP and was detected using it seven times yesterday).

Because the volume of exception occurrences can be quite large, CA MICS provides the means to categorize, aggregate, consolidate, and prioritize them to meet your needs. Each exception has the following:

- An Exception Number for unique definition.
- A Severity Level to signify degree of importance.
- A Management Area to identify area of responsibility.

STANDARD REPORTS

The standard exception reports provide a concise, integrated method for problem reporting. The reports that can be produced as part of the CA MICS DAILY job are:

- Exception Management Overview Report
- Severity Level Exception Summary Report
- Management Area Exception Summary Report

You control which reports are produced via the REPORT EXCEPTIONS statement in prefix.MICS.PARMS(EXECDEF). Refer to Section 2.3.5 of the PIOM for more information.

Two additional standard reports can be produced as required to provide the necessary background detail to effectively analyze reported exceptions:

o Full Exception Detail Report
o Short Exception Detail Report

MICF INQUIRIES

The catalog group EXCEPT that is shipped with CA MICS contains a number of standard MICF inquiries that can be used to report exception conditions. In the following list of reports, if the value of x is C, color graphic reports are produced using SAS/GRAPH. If the value of x is P, printer graphic reports are produced without using SAS/GRAPH.

o BASxM1 - Monthly Exception Summary Report
o BASxM2 - Monthly Management Area Exception Summary Report
o BASxM3 - Monthly Information Area Exception Summary Report
o BASxM4 - Monthly Exception Management Overview Report
o BASxM5 - Monthly Information Area Exception Overview Report
o BASxW1 - Weekly Exception Summary Report
o BASxW2 - Weekly Management Area Exception Summary Report
o BASxW3 - Weekly Information Area Exception Summary Report

The following inquiries that produce printed reports are also available:

o BASLD2 - Daily Severity Level Exception Summary Report
o BASLD3 - Daily Management Area Exception Summary Report
o BASLD4 - Daily Short Exception Detail Report
o BASLD5 - Daily Full Exception Detail Report
o BASLD6 - Daily Exception Ranking Report
o BASLM6 - Monthly Exception Ranking Report

These standard inquiries have execution-time parameter selections that permit you to report on a subset of the exceptions. For example, inquiry BASLD5 allows selection on SYSID, Information Area, Management Area, Severity Level, and other criteria.

EXCEPTION TEST ROUTINE

Each component has an exception test routine that is invoked in the DAY200 step of the CA MICS DAILY job. You control which routines are invoked using the CREATE EXCEPTIONFILES statement in prefix.MICS.PARMS(EXECDEF). Refer to the PIOM, Section 2.3.5, for more information on EXECDEF.

An exception test routine, written in the SAS language,

defines the exception, and tests to determine whether or not the exception condition is present in the data being processed. The distributed exception test routine for each component is contained in sharedprefix.MICS.SOURCE(DYcccEXC), where ccc is the component identifier. When the DAY200 step invokes the test routines, it does so by %INCLUDEing the DYcccEXC member from prefix.MICS.USER.SOURCE. As distributed, that member then %INCLUDEs the member from sharedprefix.MICS.SOURCE. See Section 4.2 of this guide, Setting Exception Values, for more information on DYcccEXC.

EXCEPTION VALUE ANALYSIS ROUTINE

Each component has an exception value analysis (EVA) routine that can be used to help determine values for the exception conditions. The EVA routine extracts information from the CA MICS database. Descriptive statistics for the values of variables used in exception tests are printed in the Exception Value Analysis Report. The EVA routine for each component is stored in sharedprefix.MICS.SOURCE(cccEVA). The job control to execute the EVA process is contained in prefix.MICS.CNTL(cccEVA).

4.2 Setting Exception Values

Exception test routines contain exception tests which completely define the tests made to determine the exception condition and the definitions which define and classify the exception for reporting and analysis. The following is a sample exception test:

```
*
**  14002
**  POWER JOB ELAPSED TIME EXCEEDED LIMIT
*;
  IF JOBGROUP=1 AND PJBEXCTM > '00:01:00'T
  OR JOBGROUP=2 AND PJBEXCTM > '00:30:00'T
  OR JOBGROUP=3 AND PJBEXCTM > '00:59:99'T
    THEN DO;
    EXCCODE='14002';
    SEVERITY='I';
    MGMTAREA='PERFORMANCE';
    EXCDESC1='JOB ELAPSED TIME EXCEEDED LIMIT';
    EXCDESC2='JOBGROUP=' || PUT(JOBGROUP,3.)
           || ' ELAPSED TIME(H:M:S)='
           || PUT(PJBEXCTM,TIME.);
    LINK HIT;
  END;
```

This exception test is processed for each observation that is in the latest cycle of the DETAIL VSE/POWER Job Activity File (PWRPJB). The test is positive only when the value of the JOBGROUP element is one, two, or three and the PJBEXCTM element exceeds the associated time limit. For jobs in group one, the duration must be greater than one minute. For jobs in group two, the duration must be greater than 30 minutes, and for jobs in group three, the duration must be greater than or equal to one hour.

When the test is positive, the exception is categorized by providing the appropriate values for EXCCODE, SEVERITY, and MGMTAREA. EXCDESC1 provides a constant title for the exception. EXCDESC2 provides variable information for the conditions that caused the test to be positive. The LINK HIT statement invokes a routine that causes the exception condition eventually to be written to the Exception Activity File (ADMEXC) for later processing by the standard reports or MICF inquiries.

For most organizations, the exception limits used here would either cause too many exceptions or too few exceptions to be reported, which defeats one of the purposes of exception

processing: to report on "out of the ordinary" conditions that warrant attention. Therefore, each organization must determine and set its own exception values.

Furthermore, the values for some exception conditions should be determined uniquely for different environments within a single organization. For example, a system paging rate that would be excessive during the nightly batch processing may be normal during the daytime hours, which have heavy interactive usage. Also, different processors can support different paging rates. Numbers used as exception values should not be defined without some analysis of installation history, performance, and user requirements.

The CA MICS Database provides a most useful information base for analyzing historical performance. The exception value analysis routine uses the CA MICS Database to produce a report that provides a statistical analysis of the values of the variables used in the standard exception tests distributed with the products. Using the results of this analysis, along with your installation's internal political, security, or standards policies; industry publications such as CA TSO Performance Statistics; and possibly your own analysis of data element behavior, you can determine the modifications that should be made to each exception test to provide meaningful exceptions in your environments.

You can also modify the exception test routine to add your own tests for additional exception conditions not detected by the standard tests delivered with the product.

To modify the exception test routine, follow the steps outlined in Section 2.4 of the CA MICS Standard Reports Guide.

4.3 Detailed Exception Descriptions

This section provides an in-depth description of each standard exception available in the exception report process for the CA MICS Analyzer Option for VSE/POWER. The exceptions are organized by number and appear sequentially starting with exception 14001.

The description format provides the title, number, and a statement on the purpose, rationale, and definition for each exception to give you some insight as to the meaning and use of the information. Lastly, we list SAS code with a short explanation on modifying the exception threshold values.

14001: Job Turnaround Time Exceeded Objective

```
+-----------------+
| E X C 1 4 0 0 1 |  Job Turnaround Time Exceeded Objective
+-----------------+

FILE         VSE/POWER Job Activity File (PWRPJB)

PURPOSE      Identifies jobs that missed the turnaround
             target.

RATIONALE    Identifying jobs that miss the turnaround
             objectives is important when determining the
             effectiveness of data center operations and for
             determining which jobs failed to meet
             performance objectives.

DEFINITION   This exception is detected when a VSE/POWER
             job's turnaround time exceeds the
             installation-defined time limit.

EXCEPTION    The SAS statements identifying the exception
STATEMENTS   situation and describing the condition are
             shown below:
*
**  14001
**  POWER JOB TURNAROUND TIME EXCEEDED OBJECTIVE
*;
  IF PJBSRVMS = ONE THEN DO;
    EXCCODE='14001';
    SEVERITY='C';
    MGMTAREA='SERVICE';
    EXCDESC1='POWER JOB TURNAROUND TIME EXCEEDED OBJECTIVE';
    EXCDESC2='JOBGROUP=' || PUT(JOBGROUP,3.)
           || ' TURN TIME(H:M:S)='
           || PUT(PJBTURTM,TIME.)
           || ' TARGET='
           || PUT(PJBTARTM,TIME.);
    LINK HIT;
  END;

THRESHOLD       Not applicable.
MODIFICATION
```

14002: Job Elapsed Time Exceeded Limit

```
+----------------+
| E X C 1 4 0 0 2 |  Job Elapsed Time Exceeded Limit
+----------------+

FILE         VSE/POWER Job Activity File (PWRPJB)

PURPOSE      Identifies long running VSE/POWER jobs.

RATIONALE    Identifying  long  running  VSE/POWER  jobs  is
             important when determining the effectiveness  of
             data center operations and for determining which
             jobs failed to meet performance objectives.

DEFINITION   This exception  is  detected  when  a  VSE/POWER
             job's   elapsed   time   (i.e.,   execution  time)
             exceeds the installation-defined time limit  for
             a specified job group.

EXCEPTION    The  SAS  statements  identifying  the exception
STATEMENTS   situation  and  describing  the  condition   are
             shown below:
*
**  14002
**  POWER JOB ELAPSED TIME EXCEEDED LIMIT
*;
  IF JOBGROUP=1 AND PJBEXCTM > '00:01:00'T
  OR JOBGROUP=2 AND PJBEXCTM > '00:30:00'T
  OR JOBGROUP=3 AND PJBEXCTM > '00:59:99'T
    THEN DO;
    EXCCODE='14002';
    SEVERITY='I';
    MGMTAREA='PERFORMANCE';
    EXCDESC1='JOB ELAPSED TIME EXCEEDED LIMIT';
    EXCDESC2='JOBGROUP=' || PUT(JOBGROUP,3.)
           || ' ELAPSED TIME(H:M:S)='
           || PUT(PJBEXCTM,TIME.);
    LINK HIT;
  END;

THRESHOLD       Modify the job groups and their corresponding
MODIFICATION    execution times as appropriate for your site.

                jobgroup          The     numeric       value
                                  identifying   the  selected
                                  job category.  For group 3,
                                  it appears as:
```

	JOBGROUP = 3
execution-time	The amount of execution time is described as hours, minutes, and seconds. A threshold of 30 minutes appears as: PJBEXCTM > '00:30:00'T

14003: Job Input Queue Time Exceeded Limit

```
+----------------+
| E X C 1 4 0 0 3 |  Job Input Queue Time Exceeded Limit
+----------------+

FILE         VSE/POWER Job Activity File (PWRPJB)

PURPOSE      Identifies jobs waiting a long time in the input
             queue for initiation.

RATIONALE    The amount of time that a job waits in the input
             queue  may  be  key  to the analysis of poor job
             turnaround.  Reasons  for  input  queue  delays
             include heavy workload, poor partition settings,
             or ineffective job classes and priorities.

DEFINITION   This  exception  is  detected  when a  VSE/POWER
             job's input queue time exceeds  the installation
             defined time limit for a specified job group.

EXCEPTION    The  SAS  statements  identifying  the exception
STATEMENTS   situation  and  describing  the  condition  are
             shown below:

*
**  14003
**  POWER JOB INPUT QUEUE TIME EXCEEDED LIMIT
*;
  IF JOBGROUP=1 AND PJBINQTM > '00:00:15'T
  OR JOBGROUP=2 AND PJBINQTM > '00:05:00'T
  OR JOBGROUP=3 AND PJBINQTM > '00:15:00'T
    THEN DO;
    EXCCODE='14003';
    SEVERITY='I';
    MGMTAREA='PERFORMANCE';
    EXCDESC1='JOB INPUT QUEUE TIME EXCEEDED LIMIT';
    EXCDESC2='JOBGROUP=' || PUT(JOBGROUP,3.)
           || ' INPUT QUEUE TIME(H:M:S)='
           || PUT(PJBINQTM,TIME.);
    LINK HIT;
  END;

THRESHOLD       Modify  the values used  for  job  group  and
MODIFICATION    input queue  time  as  appropriate  for  your
                site.

                jobgroup            The     numeric     value
                                    identifying the  selected
```

	job category. For group 3, it appears as: JOBGROUP=3
input-queue-time	The amount of input queue time is described as hours, minutes, and seconds. A threshold of 30 minutes appears as: PJBINQTM > '00:30:00'T

14053: Program SIO/CPU-Second Ratio Exceeds Limit

```
+-----------------+
| E X C 1 4 0 5 3 |  Program  SIO/CPU-Second  Ratio   Exceeds
+-----------------+  Limit
```

FILE — VSE/POWER Program Activity File (PWRPPG)

PURPOSE — Identifies programs which incurred a high ratio of SIO/CPU-Second.

RATIONALE — Programs which have a high SIO to CPU-Second ratio are I/O bound and are good candidates for improvement through data set reblocking.

DEFINITION — This exception is detected when a program's SIO per CPU-Second ratio exceeds the installation defined values. In addition, the step's CPU time and total SIOs must exceed the installation defined values.

EXCEPTION STATEMENTS — The SAS statements identifying the exception situation and describing the condition are shown below:

```
*
** 14053
** PROGRAM SIO/CPU-SECOND RATIO EXCEEDS LIMIT
*;
  RATIO=ZERO;
  IF PPGCPUTM > ZERO THEN RATIO=PPGSIO/PPGCPUTM;
  IF RATIO > 10 AND PPGCPUTM > '00:09:00'T  AND PPGSIO > 1000
    THEN DO;
    MGMTAREA='WORKLOAD';
    EXCCODE='14053';
    SEVERITY='W';
    EXCDESC1='PROGRAM SIO/CPU-SECOND RATIO EXCEEDS LIMIT';
    EXCDESC2='SIO/CPU-SEC=' || PUT(RATIO,5.)
           || ' SIO=' || PUT(PPGSIO,6.)
           || ' CPU TIME=' || PUT(PPGCPUTM,TIME.);
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION — Modify the values used for sio/cpu-sec, cputime, and sios according to the following conventions:

sio/cpu-sec — the value for a ratio of sio/cpu-sec of 10 appears as:

```
                    RATIO > 10

cputime             the amount of CPU time is
                    described as hours, minutes,
                    and seconds.  A threshold of 5
                    minutes appears as:

                    PPGCPUTM > '00:05:00'T

sios                the number of SIOs is
                    specified.  A threshold of
                    1000 SIOs appears as:

                    PPGSIO > 1000
```

14056: Unauthorized Program User

```
+----------------+
| E X C 1 4 0 5 6 |  Unauthorized Program User
+----------------+
```

FILE — VSE/POWER Program Activity File (PWRPPG)

PURPOSE — Identifies unauthorized users who have executed sensitive programs.

RATIONALE — The use of these programs can result in loss of system integrity or security. By identifying unauthorized users you can limit your exposure. The exception as distributed is only an example and must be tailored for your site.

DEFINITION — This exception is detected when an unauthorized user executes the PDZAP, OBJMAINT, or CORGZ programs.

EXCEPTION STATEMENTS — The SAS statements identifying the exception situation and describing the condition are shown below:

```
*
**  14056
**  UNAUTHORIZED PROGRAM USE (VSE/POWER)
*;
  IF (
      PROGRAM='PDZAP' OR PROGRAM='OBJMAINT'
      OR PROGRAM='CORGZ'
     ) AND
     NOT(JOB=:'SYS')
     THEN DO;
     EXCCODE='14056';
     SEVERITY='W';
     MGMTAREA='STANDARDS';
     EXCDESC1='UNAUTHORIZED PROGRAM USE (VSE/POWER)';
     EXCDESC2='PROGRAM USED: ' || PROGRAM;
     LINK HIT;
  END;
```

THRESHOLD MODIFICATION — Tailor the list of PDZAP, OBJMAINT, or CORGZ program names contained in the definition string and specify those jobs which should be excluded from causing an exception according to the following conventions:

program-name — The name of the program, specified as a one to eight character alphanumeric field, must be enclosed in quotes as a standard SAS character literal. Additional program names must be connected to the list with an OR.

job-name — The name of the job to be used for selection, specified as a one to eight character alphanumeric field, must be enclosed in quotes as a standard SAS character literal. Additional job names must be enclosed in the parentheses and be connected to the list with an OR. For all job names that do not begin with SYS or MAINT, the specification is:

```
AND NOT(JOB=:'SYS' OR JOB=:'MAINT')
```

By placing a colon prior to the literal, the job selection is satisfied by all job names having their first n digits match the string. For example, the test JOB=:'SYS' selects are job names starting with SYS.

14057: Program Resource Consumption Exceeded Limit

```
+----------------+
| E X C 1 4 0 5 7 |  Program  Resource  Consumption  Exceeded
+----------------+  Limit
```

FILE — VSE/POWER Program Activity File (PWRPPG)

PURPOSE — Identifies programs which utilized a large quantity of resources.

RATIONALE — Heavy resource consuming programs may have a degrading effect on overall system performance. Identifying and controlling these jobs may be crucial to maintaining consistent performance.

DEFINITION — This exception is detected when a program's resource consumption exceeds the installation defined values.

EXCEPTION STATEMENTS — The SAS statements identifying the exception situation and describing the condition are shown below:

```
*
** 14057
** PROGRAM RESOURCE CONSUMPTION EXCEEDED LIMIT (VSE/POWER)
*;

  IF PPGCPUTM > '00:09:00'T
    OR PPGSIO > 50000
    THEN DO;
    EXCCODE='14057';
    SEVERITY='W';
    MGMTAREA='WORKLOAD';
    EXCDESC1= 'PROGRAM RESOURCE CONSUMPTION EXCEEDED LIMIT
    (VSE/POWER)';
    EXCDESC2=' SIO=' || PUT(PPGSIO,6.)
           || ' CPU TIME=' || PUT(PPGCPUTM,TIME.);
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION — Modify the values used for cpu-time and sios according to the following conventions:

cpu-time — The amount of CPU time is described as hours, minutes, and seconds. A threshold of 10 minutes and 30 seconds

	appears as: PPGCPUTM > '00:10:30'T
sios	The number of SIOs is specified. A threshold of 50000 SIOs appears as: PGMSIOS > 50000

14101: Degraded Printer Performance

```
+----------------+
| E X C 1 4 1 0 1 |  Degraded Printer Performance
+----------------+

FILE         VSE/POWER Output Activity File (PWRPOA)

PURPOSE      Identifies printer activity at a rate lower than
             the expected performance level.

RATIONALE    Printer rates may fall considerably below the
             expected performance level due to the printing
             of special or invalid characters, printer
             problems such as out of paper, and the use of
             special print trains. Applications with a
             printing rate below expected performance should
             be examined for possible procedural
             improvements.

DEFINITION   This exception is detected when an output file
             is printed at a rate below the installation
             defined expected performance level.

EXCEPTION    The SAS statements identifying the exception
STATEMENTS   situation and describing the condition are
             shown below:

*
** 14101
** DEGRADED PRINTER PERFORMANCE (VSE/POWER)
*;
  RATE=ZERO;
  IF POAACTTM > ZERO THEN RATE=(POANLR / (POAACTTM/60));
  IF RATE AND POAACTTM > ONE AND RATE < 9
    THEN DO;
    MGMTAREA='PERFORMANCE';
    EXCCODE='14101';
    SEVERITY='W';
    MGMTAREA='PERFORMANCE';
    EXCDESC1='DEGRADED PRINTER PERFORMANCE (VSE/POWER)';
    EXCDESC2='LINES=' || PUT(POANLR,6.)
           || ' PRINTER TIME='   || PUT(POAACTTM,TIME.)
           || ' LINES/MINUTE=' || PUT(RATE,6.);
    LINK HIT;
  END;

THRESHOLD       Modify the values used for printer-time and
MODIFICATION    print-rate according to the following
```

```
conventions:

printer-time    The  amount  of printer active
                time for a value of 5  minutes
                appears as:

                POAACTTM > '00:05:00'T

print-rate      The  print rate for a value of
                400 lines per  minute  appears
                as:

                RATE < 400
```

14102: Lines Printed Exceeded Limit

```
+----------------+
| E X C 1 4 1 0 2 |  Lines Printed Exceeded Limit
+----------------+

FILE         VSE/POWER Output Activity File (PWRPOA)

PURPOSE      Identifies  print volumes exceeding installation
             standards.

RATIONALE    Occurrences of large  print  volumes  should  be
             investigated  to determine whether the output is
             actually  needed  and  useful.   Consideration
             should be given to eliminating printing which is
             not needed, using microfiche where feasible,  or
             increasing  the  number  of  lines  per  inch to
             reduce paper usage.

DEFINITION   This exception is detected when a  print  volume
             exceeds the installation defined standard.

EXCEPTION    The  SAS  statements  identifying  the exception
STATEMENTS   situation  and  describing  the  condition   are
             shown below:

*
** 14102
** LINES PRINTED EXCEEDED LIMIT (VSE/POWER)
*;
  IF POANLR > 190000
    THEN DO;
    EXCCODE='14102';
    SEVERITY='W';
    MGMTAREA='STANDARDS';
    EXCDESC1='LINES PRINTED EXCEEDED LIMIT';
    EXCDESC2='LINES PRINTED=' || PUT(POANLR,7.);
    LINK HIT;
  END;

THRESHOLD       Modify   the  value  used  for  lines-printed
MODIFICATION    according to the following conventions:

                lines-printed    The   numeric   value   for  a
                                 lines-printed value of 50000
                                 appears as:

                                 POANLR > 50000
```

14102: Lines Printed Exceeded Limit

```
+-----------------+
| E X C 1 4 1 5 1 |  RJE Excessive Errors
+-----------------+
```

FILE — VSE/POWER RJE Activity File (PWRPRA)

PURPOSE — Identifies RJE error counts exceeding installation standards.

RATIONALE — Excessive RJE errors are an indication of poor line quality. It is advisable to have the quality of the line service evaluated with a line monitor.

DEFINITION — This exception is detected when the RJE transmission error count exceeds the defined standard.

EXCEPTION STATEMENTS — The SAS statements identifying the exception situation and describing the condition are described below:

```
*
** 14151
** RJE EXCESSIVE ERRORS
*;

  IF PRAERROR > 5
    THEN DO;
    MGMTAREA='PERFORMANCE';
    EXCCODE='14151';
    SEVERITY='I';
    EXCDESC1='RJE EXCESSIVE ERRORS';
    EXCDESC2='NUMBER OF RJE ERRORS = '   || PUT(PRAERROR,5.);
    LINK HIT;
  END;
```

THRESHOLD MODIFICATION — Modify the value for RJE errors detected according to the following conventions:

rje-errors — The numeric value for a error count of 5000 appears as:

PRAERROR > 5000

14102: Lines Printed Exceeded Limit

```
+-----------------+
| E X C 1 4 1 5 2 |  RJE Excessive Timeouts
+-----------------+

FILE         VSE/POWER RJE Activity File (PWRPRA)

PURPOSE      Identifies    RJE   timeout   counts   exceeding
             installation standards.

RATIONALE    Excessive RJE timeouts are an indication of  low
             line   utilization.    The   lines   may   not   be
             required, or possibly can be consolidated.

DEFINITION   This  exception is detected when the RJE timeout
             count exceeds the defined standard.

EXCEPTION    The  SAS  statements  identifying  the exception
STATEMENTS   situation  and  describing  the  condition   are
             described below:
*
** 14152
** RJE EXCESSIVE TIMEOUTS
*;
  IF PRATIMOT > 1000
    THEN DO;
    MGMTAREA='PERFORMANCE';
    EXCCODE='14152';
    SEVERITY='I';
    EXCDESC1='RJE EXCESSIVE TIMEOUTS';
    EXCDESC2='NUMBER OF TIMEOUTS = ' || PUT(PRATIMOT,5.);
    LINK HIT;
  END;

THRESHOLD       Modify  the  value for RJE timeouts  detected
MODIFICATION    according to the following conventions:

                rje-timeouts      The  numeric  value  for  a
                                  timeout   count   of   100
                                  appears as:

                                  PRATIMOT > 100
```

14153: RJE Excessive Invalid Transmissions

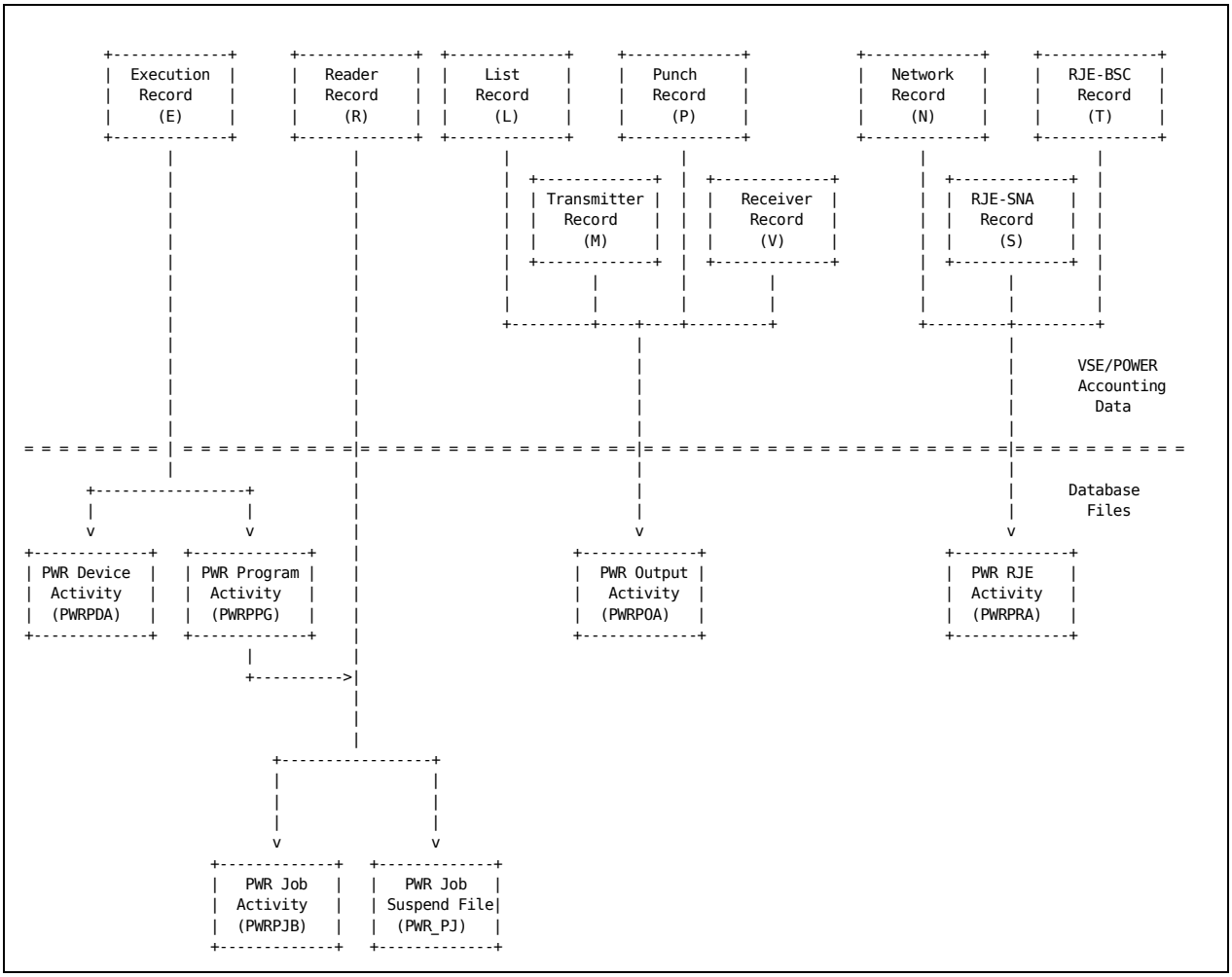
```
+----------------+
| E X C 1 4 1 5 3 |  RJE Excessive Invalid Transmissions
+----------------+

FILE         VSE/POWER RJE Activity File (PWRPRA)

PURPOSE      Identifies  RJE  invalid  transmission  counts
             exceeding installation standards.

RATIONALE    Excessive  RJE  invalid  transmissions  are  an
             indication  of  poor  line  quality.   It  is
             advisable  to  have  the  quality  of  the  line
             service evaluated with a line monitor.

DEFINITION   This exception is detected when the RJE  invalid
             transmission count exceeds the defined standard.

EXCEPTION    The  SAS  statements  identifying  the exception
STATEMENTS   situation  and  describing  the  condition  are
             described below:
*
** 14153
** RJE EXCESSIVE INVALID TRANSMISSIONS
*;
  IF PRAINRSP > 1
    THEN DO;
    MGMTAREA='PERFORMANCE';
    EXCCODE='14153';
    SEVERITY='I';
    EXCDESC1='RJE EXCESSIVE INVALID TRANSMISSIONS';
    EXCDESC2='NUMBER OF INVALID TRANSMISSIONS = '
             || PUT(PRAINRSP,5.);
    LINK HIT;
  END;

THRESHOLD       Modify    the    value    for    RJE    invalid
MODIFICATION    transmissions   detected   according   to   the
                following conventions:

                invalid-trans     The numeric  value  for  an
                                  error  count  of  3 appears
                                  as:

                                  PRAINRSP > 3
```

Chapter 5: FILES

The CA MICS Analyzer Option for VSE/POWER builds the VSE/POWER Information Area in the CA MICS Database. The information area consists of the following files, holding data produced by IBM's VSE/POWER product:

- VSE/POWER Job Activity (PWRPJB) File
- VSE/POWER Job Suspend (PWR_PJ) File
- VSE/POWER Program Activity (PWRPPG) File
- VSE/POWER Output Activity (PWRPOA) File
- VSE/POWER Remote Activity (PWRPRA) File
- VSE/POWER Device Activity (PWRPDA) File

Figure 5-1 illustrates the data flow from input sources through product processing to producing files. Chapter 6 of this guide describes each record type.



FIGC Figure 5-1. Analyzer Option for VSE/POWER Data Flow

This section contains the following topics:

5.1 Data Element Naming Conventions

The data elements in the CA MICS Database files follow naming conventions that depend on whether they are standard or common data elements. Standard data elements use the first three characters of their name to identify the file in which they are defined. The exception to this rule is that elements in the Job Suspend File (PWR_PJ), which is designed to parallel the Job Activity File (PWRPJB), use the PJB prefix. The following chart lists 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
VSE/POWER Job Activity	PWRPJB	PJB
VSE/POWER Job Suspend	PWR_PJ	PJB
VSE/POWER Program Activity	PWRPPG	PPG
VSE/POWER Output Activity	PWRPOA	POA
VSE/POWER Remote Activity	PWRPRA	PRA
VSE/POWER Device Activity	PWRPDA	PDA

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. You will find common data elements listed under Sequence/Summary Data Elements and Common Data Elements in Section 5.2.3.2, Data Elements List.

5.2 VSE/POWER Information Area Files

This section describes each file built by the CA MICS Analyzer Option for VSE/POWER. The descriptions contain the following:

- File description
- File organization
- Data elements list

Data elements are defined in Appendix B, Data Dictionary.

The table below gives an overview of the timespans at which each file is supported. This table has been generated to accurately reflect the CA MICS system at your installation.

VSE/POWER Information Area	Date Generated: Tue, May 12, 2009

XDWMYT	File	File Name
X.....	PWR_PJ	VSE/POWER JOB SUSPEND FILE
X.....	PWR_PP	VSE/POWER PROGRAM SUSPEND FILE
X.....	PWR_PR	VSE/POWER READER SUSPEND FILE
.DWM..	PWRPDA	VSE/POWER DEVICE ACTIVITY FILE
X..MY.	PWRPJB	VSE/POWER JOB ACTIVITY FILE
X..M..	PWRPOA	VSE/POWER OUTPUT ACTIVITY FILE
X..M..	PWRPPG	VSE/POWER PROGRAM ACTIVITY FILE
X..M..	PWRPRA	VSE/POWER REMOTE ACTIVITY FILE

Figure 5-2. VSE/POWER File Timespan Granularity Chart

The files are described in the following sections:

1 - VSE/POWER Job Activity (PWRPJB) File
2 - VSE/POWER Job Suspend (PWR_PJ) File
3 - VSE/POWER Program Activity (PWRPPG) File
4 - VSE/POWER Output Activity (PWRPOA) File
5 - VSE/POWER Remote Activity (PWRPRA) File
6 - VSE/POWER Device Activity (PWRPDA) File

5.2.1 VSE/POWER Job Activity (PWRPJB) File

```
FILENAME=PWRPJBxx

The records in the VSE/POWER Job Activity File quantify the
resources consumed by a single POWER job, including all VSE
jobs within it.  Additional information includes termination
reasons, turnaround specifications, and identification fields
such as Job Priority, Job Class, etc.  The records in this
file are built from the Reader Account Record and the
Execution Account Record of VSE/POWER Accounting.

The following sections describe the file's organization and
list the data elements maintained.

    1 - File Organization
    2 - Data Elements List
    3 - Usage Considerations
    4 - Retrieval Examples
```

5.2.1.1 File Organization

The table below identifies data elements by which this file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in that timespan. At the DETAIL level, data is sequenced but not summarized.

Because CA MICS allows you to define the timespans in which a file is supported, this table has been generated as part of the installation process and accurately reflects the CA MICS system at your installation.

Timespan	Level of Data Granularity				
DETAIL	SYSID	PWRACT1	PWRACT2	PWRACT3	JOBGROUP
	YEAR	MONTH	ZONE	DAY	HOUR
	ENDTS				
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	PWRACT1	PWRACT2	JOBGROUP	YEAR
	MONTH	ZONE			
YEARS	SYSID	PWRACT1	PWRACT2	JOBGROUP	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 5-3. PWRPJB Time-Span Granularity Chart

5.2.1.2 Data Elements List

```
The list below identifies data elements contained in this
file.  The format for data element entries is

    time_span  data_element - description

where:

time_span     = the timespans in which the data element is
                supported.  A period (.) indicates that the
                data element is not supported.  Supported
                timespans are indicated by the letters
                XDWMYTE, meaning:

                  X - DETAIL
                  D - DAYS
                  W - WEEKS
                  M - MONTHS
                  Y - YEARS
                  T - TABLES AREA
                  E - ESSENTIAL

data_element  = the data element name.

description   = the long name of the data element.

Because CA MICS allows you to define the timespans in which a
file is supported, this table has been generated as part of
the installation process and accurately reflects 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..MY.E  JOBGROUP - Job Group
 X..M..E  MONTH    - Month of Year
```

```
 X..MY.E  PWRACT1  - DIVISION
 X..MY.E  PWRACT2  - PROJECT
 X.....E  PWRACT3  - USER
 X..MY.E  SYSID    - System Identifier
 X.....E  WEEK     - Week of Year
 X..MY.E  YEAR     - Year of Century
 X..MY.E  ZONE     - Time Zone

Common Data Elements

 X.....E  DAYNAME  - Name of Day of Week
 X..MY.E  ENDTS    - End Time Stamp
 X.....E  JOB      - Job Identification
 X.....E  JOBCLASS - Job Input Class
 X..MY.E  MICSVER  - CA MICS Version Number
 X..MY.E  ORGSYSID - Originating System Identification
 X.....E  PWRJOBNO - POWER Job Number
 X.....E  PWRJUSER - POWER JOB User Information
 X.....E  PWRNETAC - NJE Network Account Number
 X.....E  RDRTS    - Reader Time Stamp
 X..MY.E  STARTTS  - Start Time Stamp
 X.....E  VSEPRTID - VSE Partition Identifier
 X..MY.E  VSESYSID - VSE System Identifier
 X..MY.E  VSEVER   - VSE Version

Retained Data Elements

 X.....E  PJBEPRTY - Job Execution Priority
 X.....   PJBEXCNN - Execution Node Name
 X.....E  PJBGRPRC - Job Group Received
 X.....E  PJBINCLS - Job Input Class
 X.....   PJBINCPU - Input Processor System ID
 X.....   PJBINDEV - Input Device Name
 X.....   PJBINVID - Input Processor VSE System ID
 X.....   PJBJPRTY - Job Input Priority
 X.....E  PJBMASK  - Record Construction Audit Mask
 X.....   PJBORGID - Origin User Identification
 X.....   PJBORGJN - Origin Job Number
 X.....   PJBORGNN - Origin Node Name
 X.....E  PJBPTERM - POWER Termination Code
 X.....E  PJBRENTS - Job Reader End Time Stamp
 X.....   PJBSRMT  - Source Remote
 X.....E  PJBSUSPN - Job Suspend Flag
 X..MY.E  PJBTARTM - Job Turnaround Target Time
 X.....E  PJBVTERM - VSE Termination Code

Accumulated Data Elements

 X..MY.E  PJBALLTM - All Bound Time
```

```
 X..MY.E  PJBBADCT - Jobs with Bad SIO Counts
 X..MY.E  PJBCOST  - Job Cost
 X..MY.E  PJBCOUNT - POWER Jobs Executed
 X..MY.   PJBCPUNI - Program Instructions
 X..MY.E  PJBCPUTM - Program Processor Time
 X..MY.E  PJBDSIO  - DASD SIOs Issued
 X..MY.E  PJBEXCTM - Job Execution Time
 X..MY.E  PJBEXITC - Jobs Cancelled by RDREXIT
 X..MY.E  PJBFLUSH - Jobs Flushed
 X..MY.E  PJBINQTM - Input Queue Time
 X..MY.E  PJBIOERR - Jobs Terminated with I/O Error
 X..MY.E  PJBOSIO  - Other SIOs Issued
 X..MY.   PJBOVHNI - Overhead Instructions
 X..MY.E  PJBOVHTM - Overhead Time
 X..MY.E  PJBRDRTM - Reader Time
 X..MY.E  PJBSIO   - Total SIOs Issued
 X..MY.E  PJBSPPG  - Pages Spooled
 X..MY.   PJBSPPR  - Lines Spooled
 X..MY.   PJBSPPU  - Cards Spooled
 X..MY.E  PJBSRVEX - Jobs Exceeding Target
 X..MY.E  PJBSRVMS - Jobs Missing Target
 X..MY.E  PJBSRVMT - Jobs Meeting Target
 X..MY.E  PJBSSIO  - Spool SIOs Issued
 X..MY.E  PJBSTEPS - Phases Executed
 X..MY.   PJBSYSIN - Input Cards Read
 X..MY.   PJBTKOBL - Tracks or Blocks for Input Storage
 X..MY.E  PJBTSIO  - Tape SIOs Issued
 X..MY.E  PJBTURTM - Job Turnaround Time
 X..MY.E  PJBVJOBS - VSE Jobs Executed

Maximum Data Elements

 X..MY.E  PJBMXCTM - Maximum Job Program Processor Time
 X..MY.E  PJBMXDAS - Maximum DASD Devices Used
 X..MY.E  PJBMXMUS - Maximum Step Memory Used
 X..MY.E  PJBMXSIO - Maximum Job SIOs Issued
 X..MY.E  PJBMXTAP - Maximum Tape Devices Used

Derived Data Elements

 X..MY.E  PJBPCSEX - Pct Jobs Exceeding Target
 X..MY.E  PJBPCSMS - Pct Jobs Missing Target
 X..MY.E  PJBPCSMT - Pct Jobs Meeting Target
```

5.2.1.3 Usage Considerations

This section identifies the special considerations or techniques related to using the PWRPJB File.

1. You must exercise care in using the special date and time data elements contained in each CA MICS file. As the file's granularity increases in higher timespans, certain fields lose significance and should not be used.

   - HOUR should not be used in the MONTHS timespan or above.
   - DAY and DAYNAME should not be used in the WEEKS timespan or above.
   - WEEK should not be used in the MONTHS timespan or above.

2. The following data elements have meaning only when using the PWRPJB File in the DETAIL timespan because they lose significance once summarization has been performed. These data elements appear in the PWRPJB File in the DETAIL timespan only.

   JOB - Job Identification
   JOBCLASS - Job Input Class
   PJBEPRTY - Job Execution Priority
   PJBEXCNN - Execution Node Name
   PJBGRPRC - Job Group Received
   PJBINCLS - Job Input Class
   PJBINCPU - Input Processor System ID
   PJBINDEV - Input Device Name
   PJBINVID - Input Processor VSE System ID
   PJBJPRTY - Job Input Priority
   PJBMASK - Record Construction Audit Mask
   PJBORGID - Origin User Identification
   PJBORGJN - Origin Job Number
   PJBORGNN - Origin Node Name
   PJBPTERM - POWER Termination Code
   PJBRENTS - Job Reader End Time Stamp
   PJBSRMT - Source Remote
   PJBSUSPN - Job Suspend Flag
   PJBVTERM - VSE Termination Code
   PWRACTx - POWER Account Codes
   PWRJOBNO - POWER Job Number
   PWRJUSER - POWER Job User Information
   RDRTS - Reader Time Stamp
   VSEPRTID - VSE Partition Identifier

3. The following data elements are user-defined. The value

in each is dependent on user-selected options or user-supplied SAS code:

```
PJBCOST  - Job Cost
PJBCPUNI - Program Instructions
PJBDSIO  - DASD SIOs Issued
PJBGRPRC - Job Group Received
PJBOSIO  - Other SIOs Issued
PJBOVHNI - Overhead Instructions
PJBPCSEX - Pct Jobs Exceeding Target
PJBPCSMS - Pct Jobs Missing Target
PJBPCSMT - Pct Jobs Meeting Target
PJBSSIO  - Spool SIOs Issued
PJBTARTM - Job Turnaround Target Time
PJBTSIO  - Tape SIOs Issued
PJBTURTM - Job Turnaround Time
```

For more details, refer to the Data Dictionary in Appendix B of this guide.

4. The following elements are normally unavailable. They are turned off for all timespans in order to conserve storage.

   ```
   PJBTOHRS - Tape Occupancy Hours
   PJBVKHRS - Virtual Kcore Hours
   ```

5. The observations in the PWRPJB File are created by merging information from VSE/POWER Account Data record types R and E. This provides a complete picture of the input queue and execution stages of a job's processing. Output processing has not been added because of the lack of a unique job identifier.

   The best identifier available for VSE/POWER jobs consists of the elements PWRJOBNO and JOB. The SYSID element can also be added if there is only one POWER system, or if the systems do not share work via shared spool or NJE facilities. Even with these restrictions, it is possible to combine observations incorrectly for unrelated jobs.

5.2.1.4 Retrieval Examples

This section presents a retrieval example for the VSE/POWER Job Activity File.

In the example, several CA MICS macros are used to specify the files used and the summarization done.

Example: Print a listing of the 50 jobs that were the largest consumers of CPU resources.

```
DATA JOBS ;
  SET &pPWRX..PWRPJB01 ;
RUN ;

PROC SORT ;
  BY DESCENDING PJBCPUTM ;
RUN ;

PROC PRINT DATA=JOBS(OBS=50) UNIFORM LABEL ;
TITLE 'TOP 50 CPU CONSUMING JOBS' ;
  ID SYSID JOB PWRJOBNO ;
  VAR PJBCPUTM PJBEXCTM PJBDSIO PJBTSIO ;
RUN ;
```

5.2.2 VSE/POWER Job Suspend (PWR_PJ) File

FILENAME=PWR_PJxx

Since all of the information from the single execution of a POWER job may not be available in a single update of CA MICS, it is necessary to maintain the incomplete job information for further completion. The VSE/POWER Job Suspend File contains the consolidated record for a POWER job if the last step record has not been received. This file parallels the VSE/POWER Job Activity File (PWRPJB) and contains the same variables.

The following sections describe the file's organization and list the data elements maintained.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.2.2.1 File Organization

The table below identifies data elements by which this file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in that timespan. At the DETAIL level, data is sequenced but not summarized.

Because CA MICS allows you to define the timespans in which a file is supported, this table has been generated as part of the installation process and accurately reflects the CA MICS system at your installation.

Timespan	Level of Data Granularity				
DETAIL	SYSID YEAR ENDTS	PWRACT1 MONTH	PWRACT2 ZONE	PWRACT3 DAY	JOBGROUP 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 5-4. PWR_PJ Time-Span Granularity Chart

5.2.2.2 Data Elements List

```
The list below identifies data elements contained in this
file.  The format for data element entries is

    time_span  data_element - description

where:

time_span     = the timespans in which the data element is
                supported.  A period (.) indicates that the
                data element is not supported.  Supported
                timespans are indicated by the letters
                XDWMYTE, meaning:

                  X - DETAIL
                  D - DAYS
                  W - WEEKS
                  M - MONTHS
                  Y - YEARS
                  T - TABLES AREA
                  E - ESSENTIAL

data_element  = the data element name.

description   = the long name of the data element.

Because CA MICS allows you to define the timespans in which a
file is supported, this table has been generated as part of
the installation process and accurately reflects 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  JOBGROUP - Job Group
```

```
 X.....E  MONTH    - Month of Year
 X.....E  PWRACT1  - DIVISION
 X.....E  PWRACT2  - PROJECT
 X.....E  PWRACT3  - USER
 X.....E  SYSID    - System Identifier
 X.....E  WEEK     - Week of Year
 X.....E  YEAR     - Year of Century
 X.....E  ZONE     - Time Zone

Common Data Elements

 X.....E  DAYNAME  - Name of Day of Week
 X.....E  ENDTS    - End Time Stamp
 X.....E  JOB      - Job Identification
 X.....E  JOBCLASS - Job Input Class
 X.....E  MICSVER  - CA MICS Version Number
 X.....E  ORGSYSID - Originating System Identification
 X.....E  PWRJOBNO - POWER Job Number
 X.....E  PWRJUSER - POWER JOB User Information
 X.....E  PWRNETAC - NJE Network Account Number
 X.....E  RDRTS    - Reader Time Stamp
 X.....E  STARTTS  - Start Time Stamp
 X.....E  VSEPRTID - VSE Partition Identifier
 X.....E  VSESYSID - VSE System Identifier
 X.....E  VSEVER   - VSE Version

Retained Data Elements

 X.....E  PJBEPRTY - Job Execution Priority
 X.....E  PJBEXCNN - Execution Node Name
 X.....E  PJBGRPRC - Job Group Received
 X.....E  PJBINCLS - Job Input Class
 X.....E  PJBINCPU - Input Processor System ID
 X.....E  PJBINCRS - Suspended by INCRccc Flag
 X.....E  PJBINDEV - Input Device Name
 X.....E  PJBINVID - Input Processor VSE System ID
 X.....E  PJBJPRTY - Job Input Priority
 X.....E  PJBMASK  - Record Construction Audit Mask
 X.....E  PJBORGID - Origin User Identification
 X.....E  PJBORGJN - Origin Job Number
 X.....E  PJBORGNN - Origin Node Name
 X.....E  PJBPTERM - POWER Termination Code
 X.....E  PJBRENTS - Job Reader End Time Stamp
 X.....E  PJBSRMT  - Source Remote
 X.....E  PJBSUSPN - Job Suspend Flag
 X.....E  PJBTARTM - Job Turnaround Target Time
 X.....E  PJBVTERM - VSE Termination Code

Accumulated Data Elements
```

```
 X.....E  PJBALLTM - All Bound Time
 X.....E  PJBBADCT - Jobs with Bad SIO Counts
 X.....E  PJBCOST  - Job Cost
 X.....E  PJBCOUNT - POWER Jobs Executed
 X.....E  PJBCPUNI - Program Instructions
 X.....E  PJBCPUTM - Program Processor Time
 X.....E  PJBDSIO  - DASD SIOs Issued
 X.....E  PJBEXCTM - Job Execution Time
 X.....E  PJBEXITC - Jobs Cancelled by RDREXIT
 X.....E  PJBFLUSH - Jobs Flushed
 X.....E  PJBINQTM - Input Queue Time
 X.....E  PJBIOERR - Jobs Terminated with I/O Error
 X.....E  PJBOSIO  - Other SIOs Issued
 X.....E  PJBOVHNI - Overhead Instructions
 X.....E  PJBOVHTM - Overhead Time
 X.....E  PJBRDRTM - Reader Time
 X.....E  PJBSIO   - Total SIOs Issued
 X.....E  PJBSPPG  - Pages Spooled
 X.....E  PJBSPPR  - Lines Spooled
 X.....E  PJBSPPU  - Cards Spooled
 X.....E  PJBSRVEX - Jobs Exceeding Target
 X.....E  PJBSRVMS - Jobs Missing Target
 X.....E  PJBSRVMT - Jobs Meeting Target
 X.....E  PJBSSIO  - Spool SIOs Issued
 X.....E  PJBSTEPS - Phases Executed
 X.....E  PJBSYSIN - Input Cards Read
 X.....E  PJBTKOBL - Tracks or Blocks for Input Storage
 X.....E  PJBTSIO  - Tape SIOs Issued
 X.....E  PJBTURTM - Job Turnaround Time
 X.....E  PJBVJOBS - VSE Jobs Executed

Maximum Data Elements

 X.....E  PJBMXCTM - Maximum Job Program Processor Time
 X.....E  PJBMXDAS - Maximum DASD Devices Used
 X.....E  PJBMXMUS - Maximum Step Memory Used
 X.....E  PJBMXSIO - Maximum Job SIOs Issued
 X.....E  PJBMXTAP - Maximum Tape Devices Used

Derived Data Elements

 X.....E  PJBPCSEX - Pct Jobs Exceeding Target
 X.....E  PJBPCSMS - Pct Jobs Missing Target
 X.....E  PJBPCSMT - Pct Jobs Meeting Target
```

5.2.2.3 Usage Considerations

This section identifies the special considerations or techniques related to using the PWR_PJ File.

1. This file is identical to the PWRPJB File. Any considerations given in Section 5.2.1.3 also apply to this file.

2. The following elements may be useful when combining this file with the PWRPJB File:

   o PJBSUSPN - Job Suspend Flag. Indicates the suspend status:

     -1 = Previously suspended (PWRPJB only)
      0 = Never suspended (PWRPJB only)
    > 0 = Suspend age - number of updates since last job activity (PWR_PJ only)

5.2.2.4 Retrieval Examples

This section presents a retrieval example for the VSE/POWER Job Suspend File.

In the example, several CA MICS macros are used to specify the files used and the summarization done.

Example: List all jobs read in more than two days ago which have not started executing yet.

```
DATA JOBS ;
  SET &PPWRX..PWR_PJ01 ;
  IF PJBMASK EQ 'R.' AND PJBSUSPN GE 2 ;
RUN ;

PROC PRINT DATA=JOBS(OBS=50) UNIFORM LABEL N ;
  LABEL PJBSUSPN='SUSPEND AGE' ;
  BY SYSID ;
  ID SYSID JOB PWRJOBNO ;
  VAR RDRTS PJBSUSPN JOBCLASS PJBJPRTY ;
RUN ;
```

5.2.3 VSE/POWER Program Activity (PWRPPG) File

```
FILENAME=PWRPPGxx

The VSE/POWER Program Activity File quantifies the resource
consumption for the execution of a program in a POWER Job.
Additional information includes termination reasons,
turnaround specifications, and identification fields such as
Job Priority, Jobs Class, etc.  The records in this file are
built from the Execution Account Record of VSE/POWER
Accounting.

The following sections describe the file's organization and
list the data elements maintained.

    1 - File Organization
    2 - Data Elements List
    3 - Usage Considerations
    4 - Retrieval Examples
```

5.2.3.1 File Organization

The table below identifies data elements by which this file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in that timespan. At the DETAIL level, data is sequenced but not summarized.

Because CA MICS allows you to define the timespans in which a file is supported, this table has been generated as part of the installation process and accurately reflects the CA MICS system at your installation.

Timespan	Level of Data Granularity				
DETAIL	SYSID	PWRACT1	PWRACT2	PWRACT3	PROGRAM
	YEAR	MONTH	ZONE	DAY	HOUR
	ENDTS				
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	PWRACT1	PWRACT2	PROGRAM	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 5-5. PWRPPG Time-Span Granularity Chart

5.2.3.2 Data Elements List

```
The list below identifies data elements contained in this
file.  The format for data element entries is

    time_span  data_element - description

where:

time_span     = the timespans in which the data element is
                supported.  A period (.) indicates that the
                data element is not supported.  Supported
                timespans are indicated by the letters
                XDWMYTE, meaning:

                  X - DETAIL
                  D - DAYS
                  W - WEEKS
                  M - MONTHS
                  Y - YEARS
                  T - TABLES AREA
                  E - ESSENTIAL

data_element  = the data element name.

description   = the long name of the data element.

Because CA MICS allows you to define the timespans in which a
file is supported, this table has been generated as part of
the installation process and accurately reflects 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..M..E  MONTH    - Month of Year
 X..M..E  PROGRAM  - Program Name
```

```
 X..M..E  PWRACT1  - DIVISION
 X..M..E  PWRACT2  - PROJECT
 X.....E  PWRACT3  - USER
 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.....E  DAYNAME  - Name of Day of Week
 X..M..E  DURATION - Recording Interval Time
 X..M..E  ENDTS    - End Time Stamp
 X.....E  JOB      - Job Identification
 X.....E  JOBCLASS - Job Input Class
 X..M..E  MICSVER  - CA MICS Version Number
 X..M..E  ORGSYSID - Originating System Identification
 X.....E  PWRJOBNO - POWER Job Number
 X.....E  PWRJUSER - POWER JOB User Information
 X.....E  PWRNETAC - NJE Network Account Number
 X..M..E  STARTTS  - Start Time Stamp
 X.....E  VSEJOB   - VSE Job Name
 X.....E  VSEJUSER - VSE JOB User Information
 X.....E  VSEPRTID - VSE Partition Identifier
 X..M..E  VSESYSID - VSE System Identifier
 X..M..E  VSEVER   - VSE Version

Retained Data Elements

 X.....E  PPGEPRTY - Job Execution Priority
 X.....   PPGEXCNN - Execution Node Name
 X.....E  PPGMEMUS - Kilobytes Memory Used
 X.....   PPGORGID - Origin User
 X.....   PPGORGJN - Origin Job Number
 X.....   PPGORGNN - Origin Node Name
 X.....E  PPGPTERM - POWER Termination Code
 X.....E  PPGRTYPE - Step Record Type
 X.....   PPGSRMT  - Source Remote
 X.....E  PPGVTERM - VSE Termination Code

Accumulated Data Elements

 X..M..E  PPGALLTM - All Bound Time
 X..M..E  PPGBADCT - SIO Count May Be Wrong Flag
 X..M..E  PPGCOST  - Program Cost
 X..M..E  PPGCOUNT - Program Executions
 X..M..   PPGCPUNI - Program Instructions
 X..M..E  PPGCPUTM - Program Processor Time
 X..M..E  PPGDSIO  - DASD SIOs Issued
```

```
 X..M..E  PPGEXCTM - Program Execution Time
 X..M..E  PPGFLUSH - Programs Flushed
 X..M..E  PPGIOERR - Programs Terminated with I/O Error
 X..M..E  PPGOSIO  - Other SIOs Issued
 X..M..   PPGOVHNI - Overhead Instructions
 X..M..E  PPGOVHTM - Overhead Time
 X..M..E  PPGSIO   - Total SIOs Issued
 X..M..E  PPGSPPG  - Pages Spooled
 X..M..   PPGSPPR  - Lines Spooled
 X..M..   PPGSPPU  - Cards Spooled
 X..M..E  PPGSSIO  - Spool SIOs Issued
 X..M..E  PPGTSIO  - Tape SIOs Issued

Maximum Data Elements

 X..M..E  PPGMXCTM - Maximum Program Processor Time
 X..M..E  PPGMXDAS - Maximum DASD Devices Used
 X..M..E  PPGMXMUS - Maximum Memory Used
 X..M..E  PPGMXSIO - Maximum SIOs Issued
 X..M..E  PPGMXTAP - Maximum Tape Devices Used
```

5.2.3.3 Usage Considerations

This section identifies the special considerations or techniques related to using the PWRPPG File.

1. You must exercise care in using the special date and time data elements contained in each CA MICS file. As the file's granularity increases in higher time-spans, certain fields lose significance and should not be used.

   - HOUR should not be used in the MONTHS time-span or above.
   - DAY and DAYNAME should not be used in the WEEKS time-span or above.
   - WEEK should not be used in the MONTHS time-span or above.

2. The following data elements have meaning only when using the PWRPPG File in the DETAIL time-span, because they lose significance once summarization has been performed. These data elements appear in the PWRPPG File in the DETAIL time-span only.

   JOB - Job Identifier
   JOBCLASS - Job Input Class
   PPGEPRTY - Job Execution Priority
   PPGMEMUS - Kilobytes Memory Used
   PPGPTERM - POWER Termination Code
   PPGRTYPE - Step Record Type
   PPGVTERM - VSE Termination Code
   PWRJOBNO - POWER Job Number
   PWRJUSER - POWER Job User Information
   VSEJOB - VSE Job Name
   VSEJUSER - VSE Job User Information
   VSEPRTID - VSE Partition Identifier
   PPGDRMT - Destination Remote
   PPGEXCNN - Execution Node Name
   PPGORGID - Origin User
   PPGORGJN - Origin Job Number
   PPGORGNN - Origin Node Name
   PPGSRMT - Source Remote
   PPGUFLDL - User Field Length

3. The following data elements are user-defined. The value in each is dependent on user-selected options or user-supplied SAS code:

   PPGCOST - Program Cost
   PPGDSIO - DASD SIOs Issued

PPGOSIO - Other SIOs Issued
PPGSIO   - Total SIOs Issued
PPGSSIO - Spool SIOs Issued
PPGTSIO - Tape SIOs Issued
PWRACTX - POWER Account Codes
PPGCPUNI - Program Instructions
PPGOVHNI - Overhead Instructions

For more details, refer to the Data Dictionary in Appendix B of this guide.

4. The following elements are normally unavailable. They are turned off for all time-spans in order to conserve storage.

PPGTOHRS - Tape Occupancy Hours
PPGUFLD1 - User Field 1
PPGUFLD2 - User Field 2
PPGUFLD3 - User Field 3
PPGUFLD4 - User Field 4
PPGUFLD5 - User Field 5
PPGUFLDL - User Field Length
PPGVKHRS - Virtual Kcore Hours

5.2.3.4 Retrieval Examples

This section presents retrieval examples for the VSE/POWER Program Activity File.

In the examples, several CA MICS macros are used to specify the files used and the summarization done.

1. List the frequency of use for all programs by system ID for the last three months.

```
PROC FREQ DATA=%MFILE(DB=P,TS=MONTHS,F=PPG01-03) ;
  BY SYSID ;
  WEIGHT PPGCOUNT ;
  TABLE PROGRAM ;
RUN ;
```

2. Report the use of program BACKUP by department (PWRACT1) over the last two months, plus the current month to date. Summarize the resources consumed.

```
DATA SAS ;
  SET %MFILE(DB=P,TS=MONTHS,F=PPG00-02) ;
  IF PROGRAM EQ 'BACKUP' ;
RUN ;

%LET BY = PROGRAM PWRACT1 ;
%LET BREAK = PWRACT1 ;
PROC SORT ;
  BY &BY ;
RUN ;
DATA ;
 SET ;
 %PPGSUM ;
RUN ;

PROC PRINT UNIFORM LABEL ;
  BY PROGRAM ;
  ID PROGRAM PWRACT1 ;
  VAR PPGCOUNT PPGTSIO PPGMXTAP PPGCPUTM ;
RUN ;
```

5.2.4 VSE/POWER Output Activity (PWRPOA) File

```
FILENAME=PWRPOAxx

The VSE/POWER Output Activity File quantifies the print and
punch activity of the VSE/POWER system.  Observations in this
file are built from the List Account Record, the Punch
Account Record, and the Transmitter/Receiver Record of
VSE/POWER Accounting.

The following sections describe the file's organization and
list the data elements maintained.

    1 - File Organization
    2 - Data Elements List
    3 - Usage Considerations
    4 - Retrieval Examples
```

5.2.4.1 File Organization

The table below identifies data elements by which this file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in that timespan. At the DETAIL level, data is sequenced but not summarized.

Because CA MICS allows you to define the timespans in which a file is supported, this table has been generated as part of the installation process and accurately reflects the CA MICS system at your installation.

Timespan	Level of Data Granularity				
DETAIL	SYSID	PWRACT1	PWRACT2	PWRACT3	SYSOUT
	FORMNUM	YEAR	MONTH	ZONE	DAY
	HOUR	ENDTS			
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID	PWRACT1	PWRACT2	SYSOUT	FORMNUM
	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 5-6. PWRPOA Time-Span Granularity Chart

5.2.4.2 Data Elements List

```
The list below identifies data elements contained in this
file.  The format for data element entries is

    time_span  data_element - description

where:

time_span     = the timespans in which the data element is
                supported.  A period (.) indicates that the
                data element is not supported.  Supported
                timespans are indicated by the letters
                XDWMYTE, meaning:

                  X - DETAIL
                  D - DAYS
                  W - WEEKS
                  M - MONTHS
                  Y - YEARS
                  T - TABLES AREA
                  E - ESSENTIAL

data_element  = the data element name.

description   = the long name of the data element.

Because CA MICS allows you to define the timespans in which a
file is supported, this table has been generated as part of
the installation process and accurately reflects 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..M..E  FORMNUM  - Form Number
 X.....E  HOUR     - Hour of Day
 X..M..E  MONTH    - Month of Year
```

```
 X..M..E  PWRACT1  - DIVISION
 X..M..E  PWRACT2  - PROJECT
 X.....E  PWRACT3  - USER
 X..M..E  SYSID    - System Identifier
 X..M..E  SYSOUT   - SYSOUT Class
 X.....E  WEEK     - Week of Year
 X..M..E  YEAR     - Year of Century
 X..M..E  ZONE     - Time Zone

Common Data Elements

 X.....E  DAYNAME  - Name of Day of Week
 X.....E  DEVADDR  - Device Address
 X..M..E  ENDTS    - End Time Stamp
 X.....E  JOB      - Job Identification
 X..M..E  MICSVER  - CA MICS Version Number
 X..M..E  ORGSYSID - Originating System Identification
 X.....E  PWRJOBNO - POWER Job Number
 X.....E  PWRJUSER - POWER JOB User Information
 X.....E  PWRNETAC - NJE Network Account Number
 X..M..E  STARTTS  - Start Time Stamp
 X..M..E  VSESYSID - VSE System Identifier
 X..M..E  VSEVER   - VSE Version

Retained Data Elements

 X.....   POAACTNN - Action Node Name
 X.....   POAADJNN - Adjacent Node Name
 X.....E  POACPSPR - Copies Printed
 X.....   POACPYMD - 3800 Copy Modification Module Name
 X.....   POADRMT  - Destination Remote
 X.....   POADSTID - Destination User Identification
 X.....   POADSTNN - Destination Node Name
 X.....   POAFLASH - 3800 Forms Overlay Name
 X.....   POALSTSG - Last Segment Indicator
 X.....E  POAOPRTY - Output Priority
 X.....   POAORGID - Origin User Identification
 X.....   POAORGJN - Original Job Number
 X.....   POAORGNN - Origin Node Name
 X.....E  POAPTERM - POWER Termination Code
 X.....E  POAQTYPE - Queue Type
 X.....   POASEGCT - Segment Count
 X.....   POASRMT  - Source Remote

Accumulated Data Elements

 X..M..E  POAACTTM - Active Time
 X..M..E  POACNLR  - Control Logical Records
 X..M..E  POACOST  - Output Cost
```

```
X..M..E  POACOUNT - Activity Count
X..M..E  POAENLR  - Extra Logical Writer Records
X..M..E  POAEPGE  - Extra Pages Printed
X..M..E  POANLR   - Logical Writer Records
X..M..E  POANXDEL - Records Deleted by NETEXIT/XMTEXIT
X..M..E  POANXINS - Records Inserted by NETEXIT/XMTEXIT
X..M..E  POAPGE   - Pages Printed
X..M..   POATKOBL - Tracks or Blocks for Input Storage
X..M..   POAXRDEL - Records Deleted by OUTEXIT
X..M..   POAXRINS - Records Inserted by OUTEXIT
```

5.2.4.3 Usage Considerations

This section identifies the special considerations or techniques related to using the PWRPOA File.

1. You must exercise care in using the special date and time data elements contained in each CA MICS file. As the file's granularity increases in higher time-spans, certain fields lose significance and should not be used.

   - HOUR should not be used in the MONTHS time-span or above.
   - DAY and DAYNAME should not be used in the WEEKS time-span or above.
   - WEEK should not be used in the MONTHS time-span or above.

2. The following data elements have meaning only when using the PWRPOA File in the DETAIL time-span, because they lose significance once summarization has been performed. These data elements appear in the PWRPOA File in the DETAIL time-span only.

   DEVADDR - Device Address
   JOB   - Job Identifier
   POAACTNN - Action Node Name
   POAADJNN - Adjacent Node Name
   POACPSPR - Copies Printed
   POACPYMD - 3800 Copy Modification Module Name
   POADRMT - Destination Remote
   POADSTID - Destination User Identification
   POADSTNN - Destination Node Name
   POAFLASH - 3800 Forms Overlay Name
   POALSTSG - Last Segment Indicator
   POAOPRTY - Output Priority
   POAORGID - Origin User Identification
   POAORGJN - Original Job Number
   POAORGNN - Origin Node Name
   POAPTERM - POWER Termination Code
   POAQTYPE - Queue Type
   POASEGCT - Segment Count
   POASRMT - Source Remote
   PWRJOBNO - POWER Job Number
   PWRJUSER - POWER Job User Information

3. The following data elements are user-defined. The value in each is dependent on user-selected options or user-supplied SAS code:

PWRACTx - POWER Account Codes
POACOST - Output Cost

For more details, refer to the Data Dictionary in Appendix B of this guide.

4. Observations in the PWRPOA File are derived from two different types of VSE/POWER Account records. Spool I/O activity records track movement of spool files to and from local unit record devices, including Reader (type R), printer (type L), and punch (type P). When CA MICS reads these records, it assigns the record type to the element POAQTYPE.

   The other class of data includes the remote spool transmission records: transmitter (type M) and receiver (type V). These records are produced when a spool file is sent to or received from a remote VSE/POWER system. Each record also has associated with it the queue type as listed above (R, L, or P).

   To help identify these records, the FORMNUM element is assigned a special value:

   FORMNUM='*r*q'

   where:

   r = VSE/POWER record type: M for transmitter, V for receiver.
   q = Spool queue type (same as POAQTYPE)

   For example, a punch file being transmitted to another system would result in the following:

   POAQTYPE = 'P' FORMNUM = '*M*P'

   Spool I/O records contain the FORMNUM obtained from the record.

5.2.4.4 Retrieval Examples

This section presents retrieval examples for the VSE/POWER Output Activity File.

In the examples, several CA MICS macros are used to specify the files used and the summarization done.

1. Produce a summary of the remote spool traffic for the last three days. Summarize it by queue, type, origin, and destination.

```
DATA SPOOL ;
  SET %MFILE(DB=P,TS=DETAIL,F=POA01-03) ;
  IF FORMNUM EQ: '*' ;
  TYPE=SUBSTR(FORMNUM,2,1) ;
RUN ;

%LET BY = SYSID POAQTYPE TYPE POAORGNN POAACTNN
 POADSTNN ;
%LET BREAK = POADSTNN ;
PROC SORT ;
  BY &BY ;
DATA SPOOL ;
  SET ;
  %POASUM ;
RUN ;

PROC FORMAT ;
  VALUE $TYPE 'M'='SEND'
              'V'='RECEIVED' ;
  VALUE $QUEUE 'P'='PUNCH'
               'L'='LIST'
               'R'='READER' ;
RUN ;

PROC PRINT UNIFORM LABEL ;
  FORMAT TYPE $TYPE. POAQTYPE $QUEUE. ;
  BY SYSID POAQTYPE ;
  ID &BY ;
  VAR POANLR ;
RUN ;
```

2. Report printer activity yesterday by system, device address, and form number.

```
DATA SPOOL ;
  SET %MFILE(DB=P,TS=DETAIL,F=POA01) ;
  IF FORMNUM NE: '*' AND POAQTYPE EQ 'L' ;
```

```
RUN ;

%LET BY = SYSID DEVADDR FORMNUM ;
%LET BREAK = FORMNUM ;
PROC SORT ;
  BY &BY ;
DATA SPOOL ;
  SET ;
  %POASUM ;
RUN ;

PROC PRINT UNIFORM LABEL ;
  BY SYSID DEVADDR ;
  ID &BY ;
  VAR POANLR POAENLR ;
  SUM POANLR POAENLR ;
  SUMBY DEVADDR ;
RUN ;
```

5.2.5 VSE/POWER Remote Activity (PWRPRA) File

```
FILENAME=PWRPRAxx

The VSE/POWER Remote Activity File maintains information on
the BSC and SNA RJE Sessions.  For SNA RJE Sessions, the
session information is taken from the SNA Account Record of
VSE/POWER Accounting.  The session duration and
identification is maintained.  For BSC RJE Sessions, the
session information is taken from the RJE Account Record of
VSE/POWER Accounting.  Session duration and identification
are maintained as well as activity and errors.  The Network
Record is used here also.


The following sections describe the file's organization and
list the data elements maintained.

    1 - File Organization
    2 - Data Elements List
    3 - Usage Considerations
    4 - Retrieval Examples
```

5.2.5.1 File Organization

The table below identifies data elements by which this file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in that timespan. At the DETAIL level, data is sequenced but not summarized.

Because CA MICS allows you to define the timespans in which a file is supported, this table has been generated as part of the installation process and accurately reflects the CA MICS system at your installation.

Timespan	Level of Data Granularity				
DETAIL	SYSID ZONE	PRARMTID DAY	PRARMTNM HOUR	YEAR ENDTS	MONTH
DAYS	N/A				
WEEKS	N/A				
MONTHS	SYSID ZONE	PRARMTID	PRARMTNM	YEAR	MONTH
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 5-7. PWRPRA Time-Span Granularity Chart

5.2.5.2 Data Elements List

```
The list below identifies data elements contained in this
file.  The format for data element entries is

    time_span  data_element - description

where:

time_span     = the timespans in which the data element is
                supported.  A period (.) indicates that the
                data element is not supported.  Supported
                timespans are indicated by the letters
                XDWMYTE, meaning:

                  X - DETAIL
                  D - DAYS
                  W - WEEKS
                  M - MONTHS
                  Y - YEARS
                  T - TABLES AREA
                  E - ESSENTIAL

data_element  = the data element name.

description   = the long name of the data element.

Because CA MICS allows you to define the timespans in which a
file is supported, this table has been generated as part of
the installation process and accurately reflects 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..M..E  MONTH    - Month of Year
 X..M..E  PRARMTID - Remote Identifier
```

```
 X..M..E  PRARMTNM - Remote Name
 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.....E  DAYNAME  - Name of Day of Week
 X..M..E  DURATION - Recording Interval Time
 X..M..E  ENDTS    - End Time Stamp
 X..M..E  MICSVER  - CA MICS Version Number
 X..M..E  ORGSYSID - Originating System Identification
 X..M..E  STARTTS  - Start Time Stamp
 X..M..E  VSESYSID - VSE System Identifier
 X..M..E  VSEVER   - VSE Version

Retained Data Elements

 X.....E  PRAPTERM - POWER Termination Reason
 X..M..E  PRATYPE  - Remote Type
 X.....   PRAUSER  - SIGNON User Information

Accumulated Data Elements

 X..M..E  PRABUFRV - Buffers Received
 X..M..E  PRABUFTR - Buffers Transmitted
 X..M..E  PRAERROR - Errors Count
 X..M..E  PRAINRSP - Invalid Responses
 X..M..E  PRASESSN - Sessions
 X..M..E  PRATIMOT - Timeouts
 X..M..E  PRATRMER - Terminal Errors
```

5.2.5.3 Usage Considerations

This section identifies the special considerations or techniques related to using the PWRPRA File.

1. You must exercise care in using the special date and time data elements contained in each CA MICS file. As the file's granularity increases in higher timespans, certain fields lose significance and should not be used.

   - HOUR should not be used in the MONTHS timespan or above.
   - DAY and DAYNAME should not be used in the WEEKS timespan or above.
   - WEEK should not be used in the MONTHS timespan or above.

2. The following data elements have meaning only when using the PWRPRA File in the DETAIL timespan, because they lose significance once summarization has been performed. These data elements appear in the PWRPRA File in the DETAIL timespan only.

   PRAPTERM - POWER Termination Reason
   PRAUSER - SIGNON User Information

3. The following elements are normally unavailable. They are turned off for all timespans in order to conserve storage.

   PRALPSWD - Line Password
   PRANPSWD - Node Password

5.2.5.4 Retrieval Examples

This section presents a retrieval example for the VSE/POWER Remote Activity File.

In the example, several CA MICS macros are used to specify the files used and the summarization done.

Example: Summarize remote activity statistics by system and remote system identifier.

```
%LET BY = SYSID PRARMTID ;
%LET BREAK = PRARMTID ;
PROC SORT DATA=%MFILE(DB=P,TS=MONTHS,F=PRA)
 OUT=SESSIONS ;
  BY &BY ;
RUN ;
DATA ;
  SET ;
  %PRASUM ;
RUN ;

PROC PRINT LABEL UNIFORM ;
  BY SYSID ;
  ID SYSID PRARMTID PRATYPE ;
  VAR PRASESSN DURATION PRABUFRV PRABUFTR ;
RUN ;
```

5.2.6 VSE/POWER Device Activity (PWRPDA) File

FILENAME=PWRPDAxx

The VSE/POWER Device Activity File quantifies the SIO activity to devices by partitions that are under the control of VSE/POWER. It is built from the SIO segments in the Execution Account Record of VSE/POWER Accounting. Device Class and Device Type are derived from user-supplied configuration parameters.

The following sections describe the file's organization and list the data elements maintained.

1 - File Organization
2 - Data Elements List
3 - Usage Considerations
4 - Retrieval Examples

5.2.6.1 File Organization

The table below identifies data elements by which this file is sequenced and summarized in each timespan. N/A indicates that the file is not supported in that timespan. At the DETAIL level, data is sequenced but not summarized.

Because CA MICS allows you to define the timespans in which a file is supported, this table has been generated as part of the installation process and accurately reflects the CA MICS system at your installation.

Timespan	Level of Data Granularity					
DETAIL	N/A					
DAYS	SYSID	DEVCLASS	DEVTYPE	DEVADDR	YEAR	
	MONTH	DAY	HOUR			
WEEKS	SYSID	DEVCLASS	DEVTYPE	DEVADDR	YEAR	
	WEEK	ZONE	HOUR			
MONTHS	SYSID	DEVCLASS	DEVTYPE	DEVADDR	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. PWRPDA Time-Span Granularity Chart

5.2.6.2 Data Elements List

```
The list below identifies data elements contained in this
file.  The format for data element entries is

    time_span  data_element - description

where:

time_span     = the timespans in which the data element is
                supported.  A period (.) indicates that the
                data element is not supported.  Supported
                timespans are indicated by the letters
                XDWMYTE, meaning:

                  X - DETAIL
                  D - DAYS
                  W - WEEKS
                  M - MONTHS
                  Y - YEARS
                  T - TABLES AREA
                  E - ESSENTIAL

data_element  = the data element name.

description   = the long name of the data element.

Because CA MICS allows you to define the timespans in which a
file is supported, this table has been generated as part of
the installation process and accurately reflects 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

 .D....E  DAY      - Day of Month
 .DWM..E  DEVADDR  - Device Address
 .DWM..E  DEVCLASS - Device Class
```

```
 .DWM..E  DEVTYPE  - Device Type
 .DW...E  HOUR     - Hour of Day
 ......E  JOB      - Job Identification
 .D.M..E  MONTH    - Month of Year
 ......E  PWRJOBNO - POWER Job Number
 .DWM..E  SYSID    - System Identifier
 .DW...E  WEEK     - Week of Year
 .DWM..E  YEAR     - Year of Century
 .DWM..E  ZONE     - Time Zone

Common Data Elements

 .D....E  DAYNAME  - Name of Day of Week
 .DWM..E  ENDTS    - End Time Stamp
 .DWM..E  MICSVER  - CA MICS Version Number
 .DWM..E  ORGSYSID - Originating System Identification
 .DWM..E  STARTTS  - Start Time Stamp
 .DWM..E  VSESYSID - VSE System Identifier
 .DWM..E  VSEVER   - VSE Version

Accumulated Data Elements

 .DWM..E  PDASIOCT - Device SIO Count
```

5.2.6.3 Usage Considerations

This section identifies the special considerations or techniques related to using the PWRPDA File.

1. You must exercise care in using the special date and time data elements contained in each CA MICS file. As the file's granularity increases in higher timespans, certain fields lose significance and should not be used.

   - HOUR should not be used in the MONTHS timespan or above.
   - DAY and DAYNAME should not be used in the WEEKS timespan or above.
   - WEEK should not be used in the MONTHS timespan or above.

2. The following data elements are user-defined. The value in each is dependent on user-selected options or user-supplied SAS code:

   DEVCLASS - Device Class
   DEVTYPE - Device Type

   For more details, refer to the Data Dictionary in Appendix B of this guide.

5.2.6.4 Retrieval Examples

This section presents a retrieval example for the VSE/POWER Device Activity File.

In the example, several CA MICS macros are used to specify the files used and the summarization done.

Example: Produce a vertical bar chart of I/O activity for tape drives by hour and sub-grouped by device address.

```
DATA DEVICES ;
  SET &PPWRD..PWRPDA01 ;
  IF DEVCLASS EQ 'TAPE' ;
  DATE = DATEPART(ENDTS) ;
  FORMAT DATE DATE7. ;
RUN ;

PROC SORT ;
  BY SYSID DATE ;
RUN ;

PROC CHART ;
  BY SYSID DATE ;
  VBAR HOUR/DISCRETE SUMVAR=PDASIOCT SUBGROUP=DEVADDR ;
RUN ;
```

Chapter 6: DATA SOURCES

The CA MICS Analyzer Option for VSE/POWER uses IBM's VSE/POWER accounting data to produce files in the VSE/POWER Information Area of the CA MICS Database. IBM's PACCOUNT command allows you to save the data in the VSE/POWER account file; you are responsible for getting this saved data to CA MICS. See Section 7.3.6 of this guide for information on defining the input source DD statements.

The Analyzer Option for VSE/POWER uses the following record types to produce files:

Execution Account Record, type E
Reader Account Record, type R
List Account Record, type L
Punch Account Record, type P
Transmitter/Receiver Account Record, types M/V, respectively
Network Account Record, type N
RJE-BSC Line Account Record, type T
RJE-SNA Session Account Record, type S

The System Up Account Record, type U, is used to produce an operational status report as shown in Section 9.2.5 of this guide.

The following records are not used by the Analyzer Option for VSE/POWER:

Spool-Access Connect Account Record, type C
Spool-Access Operation Account Record, type X

Details about records used by this CA MICS product are discussed later in this chapter.

This section contains the following topics:

6.1 Data Collector Considerations

The data produced by the Job Accounting facility of VSE/POWER is controlled through the PACCNT macro. Individual records can be turned off or the entire facility can be disabled. Contact your VSE systems programmer to determine what collection options are active.

POWER data is collected continuously to an Account disk file. This must be dumped to an external file using the PACCOUNT operator command, which permits the file to be dumped to a disk or tape file, or to the punch spool. CA MICS can accept data produced using either the disk or tape forms of the command. The punch form of the command produces segmented card image records that would have to be reassembled to be used by CA MICS. CA MICS Product Support can advise you about the issues related to using this technique. You must make sure that the reassembled POWER data file is built as a VB (Variable Blocked) file with the LRECL attribute greater than or equal to the longest record in the file.

When preparing data for CA MICS, please do not combine data from different systems into a single file. CA MICS expects account data from each VSE/POWER system or shared spool environment to be provided through a separate DD card. Since only data from shared spool environments contains a system identifier, it would be impossible for CA MICS to determine the source of a given record if data from different systems were combined. Also, the CA MICS job summarization logic assumes that all systems associated with a given DD card share the same spool file set. If this is not true, it may attempt to combine records from unrelated jobs into a single job observation.

Another issue that may impact shared spool environments is system clock synchronization. Differences in clock settings can invalidate certain job service elements, including PJBTURTM, PJBSRVMT, PJBSRVMS, and PJBSRVEX. You should keep the system clocks in as close agreement as is practical.

6.2 Record Descriptions

The VSE/POWER accounting records shown in Figure 6-1 are used by the CA MICS Analyzer Option for VSE/POWER. The figure shows when each record is written and what each record contains.

Figure 5-1 in Chapter 5 of this guide illustrates the relationship between these records and the files in the VSE/POWER Information Area of the CA MICS Database. Chapter 3 discusses how the files are used to generate the reports produced by the Analyzer Option for VSE/POWER.

RECORD NAME TYPE	WRITTEN	RECORD CONTENTS
Execution Type E	At the end of each VSE job step	Job identification CPU time I/O counts
Reader Type R	For every job submitted for spooling	Job identification Number of input lines spooled
List Type L	For each list queue entry processed by a list task	Job identification Number of lines and pages printed Where printed
Punch Type P	For each punch queue entry processed by a punch task	Job identification Number of cards punched Where punched
Transmitter/ Receiver Type M / Type V	For every job or output transmission via a connection or during a session	Job identification Routing information Number of records
Network Type N	For each session terminated on an existing communication path	Activities during a session
RJE-BSC Line Type T	For a user session sign-off or when a line	Transmission statistics

	stop occurs	
RJE-SNA Session Type S	When an SNA user session ends	Transmission statistics
System Up Type U	When start-up completes for the VSE/POWER system	Status of the VSE/POWER partition

Figure 6-1. VSE/POWER Record Descriptions

Chapter 7: PARAMETERS

This chapter describes how to complete the required parameter definitions needed to install the CA MICS Analyzer Option for VSE/Power. The system administrator should use it as a detailed reference in conjunction with the PIOM.

Installing CA MICS requires the following:

- Gaining the necessary understanding of your data center and its needs.
- Translating that understanding into CA MICS parameters.

In particular, 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.

This chapter focuses on considerations that are unique to the CA MICS Analyzer Option for VSE/Power. 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 Technical Support at http://ca.com/support.

This section contains the following topics:

7.1 Environmental Considerations

Before coding product parameters, you need to know about the VSE environments at your installation. Before you specify parameters for the product:

- Examine existing job groups, account codes, and job control information used by the VSE system. This information helps you code appropriate values for account codes and PWRGROUP parameters.
- Review the default options for the parameters to determine their applicability to your site.
- Review reporting requirements to determine whether or not you need to code MICF inquiries to satisfy the needs of the VSE group.
- Review current CA MICS parameter specifications:
  - Review database unit specifications to determine which unit or units should include VSE/POWER data.
  - Review the SYSID parameter to determine whether or not it correlates to the VSE data. Your site's SYSID definition is located in the prefix.MICS.PARMS data set associated with each database unit.
  - Review the CA MICS ZONE parameter to ensure that it reflects the variations in VSE data. This parameter is located in the prefix.MICS.PARMS data set associated with each database unit.
  - Examine existing accounting standards for other CA MICS components, if applicable, to ensure that the CA MICS Accounting and Chargeback product aggregates information consistently.

7.2 Complex Level Parameters

Complex level parameters describe the CA MICS Analyzer Option for VSE/POWER to the CA MICS system. These parameters, listed below and described in the subsections noted, are:

1 - Account Code Specification (PWRACCT)
2 - Account Code Exit Routine (PWRACRT)
3 - Generation (PWRGENIN)

7.2.1 Account Code Specification (PWRACCT)

Account codes tell CA MICS how to associate jobs with the individual or group responsible for running the jobs. Account codes are part of the file keys of the files that support them, as discussed below. The PIOM discusses account codes in greater detail (see Section 2.3.1.7.1).

The account code specification parameter PWRACCT defines how many account fields will be maintained in the CA MICS Analyzer Option for VSE/Power files that support account codes (CA MICS supports from one to nine account fields). In addition to the number of fields, the PWRACCT parameter describes the length of each field and its contents.

The following Analyzer Option for VSE/Power files support account codes:

- VSE/POWER Job Activity File (PWRPJB)
- VSE/POWER Job Suspend File (PWR_PJ)
- VSE/POWER Program Activity File (PWRPPG)
- VSE/POWER Output Activity File (PWRPOA)

The parameters specified in the PWRACCT member of the sharedprefix.MICS.PARMS data set define the account field structure. To assign values to the account fields, you need to code an account code exit routine, PWRACRT, which is described later in this chapter.

PREPARATION

Before you define the account code structure, consult your site's VSE system programmer to determine:

- What naming standards are used to associate jobs to those responsible for running them.
- How codes are specified.
- How codes are verified.

Consider the following in determining the number of account fields required to meet your data center's reporting and analysis needs:

- Account fields are part of the file keys for the files that support them. As such, at least one record is generated for each combination of values. (There may be more than one record generated, because other fields also

make up the file keys.) A greater number of account fields means that the database will be larger than if fewer account fields are defined, but reporting and analysis can be performed in greater detail.

- If you anticipate needing to expand the account code structure in the future, establish an extra account code now to eliminate the need to retrofit the database later to add the new account code.

COLLECTING DATA TO SPECIFY ACCOUNT CODES

A worksheet for organizing the information needed to code PWRACCT is shown in Figure 7-1.

```
+-------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET:  VSE/POWER Account Code Specification|
| PARMS Library Member is PWRACCT                                         |
| Reference Section:  7.2.1, CA MICS Analyzer Option for VSE/POWER Guide  |
+-------------------------------------------------------------------------+
|                                                                         |
|  ACCOUNT  TIMESPAN   FIELD                                              |
| CODE LEVEL  MASK     LENGTH  DESCRIPTION                                |
|   (1-9)                                                                 |
|                                                                         |
|     _     T(_____)   __    '_____________________________________'     |
|                                                                         |
|     _     T(_____)   __    '_____________________________________'     |
|                                                                         |
|     _     T(_____)   __    '_____________________________________'     |
|                                                                         |
|     _     T(_____)   __    '_____________________________________'     |
|                                                                         |
|     _     T(_____)   __    '_____________________________________'     |
|                                                                         |
|     _     T(_____)   __    '_____________________________________'     |
|                                                                         |
|     _     T(_____)   __    '_____________________________________'     |
|                                                                         |
|     _     T(_____)   __    '_____________________________________'     |
|                                                                         |
|     _     T(_____)   __    '_____________________________________'     |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
+-------------------------------------------------------------------------+
| ....5...10...15...20...25...30...35...40...45...50...55...60...65...70.. |
+-------------------------------------------------------------------------+
```

Figure 7-1. VSE/POWER Account Code Specification Worksheet

The following is a description of the fields in Figure 7-1:

ACCOUNT CODE LEVEL: The level of importance of the element. Level 1 is the most important field and level 9 is the least important.

TIMESPAN MASK: Account code levels can be deactivated in the DAYS, WEEKS, MONTHS, and YEARS timespans. For example, the DETAIL timespan may have account code levels 1, 2, and 3, while the MONTHS timespan has only level 1. Setting account code levels by timespan is accomplished by specifying this additional parameter on the account code statements, as described below.

FIELD LENGTH: The length of the specified element. Valid lengths range from 1 to 30.

DESCRIPTION: The title that describes the account code level. Valid descriptions range from 1 to 40 characters.

Once prepared, code the contents of the worksheet in the data set sharedprefix.MICS.PARMS(PWRACCT). Observe the following conventions when coding:

- Blank statements are permitted. Comments are coded by beginning the statement with an asterisk (*).
- Account levels are provided in ascending order, starting with 1. You can specify up to nine account levels, with no gaps between the numbers.
- The statement is free-form but positional.

A sample PWRACCT member is:

```
* SAMPLE ACCOUNT CODE SPECIFICATION
     1 T(YYYYYY)  3 'DIVISION'
     2 T(YYYYYY)  3 'PROJECT'
     3 T(YNNNNN) 16 'USER'
```

The primary field in this account code structure is the division identifier. This identifier is limited to three positions. The second field identifies the project, and is also limited to three positions. The final field identifies the user that submitted the job, and can contain up to 16 positions. PWRACCT3 is only kept in the DETAIL timespan.

SPECIFYING THE TIMESPAN MASK

Timespans can be specified on the account code statement using the timespan mask parameter T(......) as follows:

level T(......) length 'descriptive title'

This example shows the placement and syntax of the parameter on the account code statement. Each position (.) in the timespan parameter 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 and an N indicates that the account level should be inactive.

This parameter is optional. If it is not coded, the assumed value is T(YYYYYY).

The following rules apply if you code this parameter:

- DETAIL must always be Y. Account codes may not be deactivated in this timespan.

- If DAYS is N, then WEEKS, MONTHS, and YEARS must also be N. If the account code level is inactive in the DAYS timespan, it cannot be active in higher timespans.

- If MONTHS is N, then YEARS must be N. If the account code level is inactive in the MONTHS timespan, it cannot be active in the YEARS timespan.

7.2.2 Account Code Exit Routine (PWRACRT)

An account code exit routine tells CA MICS how to assign values to the account fields defined by the account code specification.

The routine is invoked to build the elements PWRACTn, where n is the account code level. For example, if PWRACCT defines three account code levels, PWRACRT builds the elements PWRACT1, PWRACT2, and PWRACT3.

The routine sets the values of all the PWRACTn variables in the following files:

- VSE/POWER Job Activity File (PWRPJB)
- VSE/POWER Job Suspend File (PWR_PJ)
- VSE/POWER Program Activity File (PWRPPG)
- VSE/POWER Output Activity File (PWRPOA)

You code one routine, but it is invoked three times. The SAS variable PWRACTEX can be used to determine the file for which the routine is being invoked as follows:

PWRACTEX	FILE
PJB	VSE/POWER Job Activity File
PJB	VSE/POWER Job Suspend File
PPG	VSE/POWER Program Activity File
POA	VSE/POWER Output Activity File

You can use any of the data elements available in the files for which the routine is invoked except the fffCOST variable which has not yet been calculated (see Chapter 5, Files, for lists of data elements). The variable VSEDDNAM is also available; its value is the DDname of the records used to build the observation.

When processing data for the VSE/POWER Program Activity File, you can use the job elements that come from IBM's VSE/POWER Accounting Reader record. The element PJBMASK will have a value of R if these elements are valid. These job elements are:

PJBDRMT	Destination Remote
PJBINCPU	Input Processor System ID
PJBINDEV	Input Device Name
PJBORGID	Origin User Identification
PJBORGNN	Origin Node Name
PJBJPRTY	Job Input Priority

```
    PJBPTERM   POWER Termination Code
    PJBSRMT    Source Remote
    PJBSYSIN   Input Cards Read
    PJBTKOBL   Input Storage Tracks
```

Code the account code exit routine in SAS and verify that it is correct. A sample PWRACRT member is provided in sharedprefix.MICS.PARMS(PWRACRT). See Section 2.3.1.5, Notes on Coding CA MICS Parameters, in the CA MICS Planning, Installation, Operation, and Maintenance Guide for coding help. Figure 7-2 provides a worksheet for coding your routine.

EXAMPLE

The account code specification illustrated in Section 7.2.1 defines account code levels by division, project, and user, respectively. The account code exit routine to support these account codes might be:

```
* SAMPLE ACCOUNT CODE EXIT ROUTINE
*
    PWRACT1 = SUBSTR(PWRJUSER,ONE,3);
    PWRACT2 = SUBSTR(JOB,ONE,3);
    IF PWRACTEX = 'PPG' THEN PWRACT3 = VSEJUSER;
                        ELSE PWRACT3 = BLANKS;
```

For each record, the value for division (PWRACT1) is the first three characters of the PWRJUSER variable. The value for project (PWRACT2) is the first three characters of the JOB variable. The value of user (PWRACT3) is set to the value of VSEJUSER if the record goes to the VSE/POWER Program Activity File (PWRPPG). If the record goes to either the Job Activity (PWRPJB) or Output Activity (PWRPOA) File, the value for user is set to blanks.

```
+-------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET:  VSE/POWER Account Code Exit Routine |
|                                                                         |
| PARMS Library Member is PWRACRT                                         |
| Reference Section:  7.2.2, CA MICS Analyzer Option for VSE/POWER 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                     |
```

```
|                                                                          |
|     PWRACT1=field source 1 ;                                             |
|     PWRACTn=field source n ;                                             |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|                                                                          |
| END;                                                                     |
| ELSE DO;                                                                 |
|                                                                          |
| * ROUTINE TO BUILD INSTALLATION OVERHEAD ACCOUNT CODES ;                  |
|                                                                          |
|     PWRACT1='overhead category' ;                                        |
|     PWRACTn='overhead category' ;                                        |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|     _____________________________________________________________________ |
|                                                                          |
| END;                                                                     |
|                                                                          |
+--------------------------------------------------------------------------+
| ....5...10...15...20...25...30...35...40...45...50...55...60...65...70.. |
+--------------------------------------------------------------------------+
```

Figure 7-2. VSE/POWER Account Code Exit Routine Worksheet

7.2.3 Generation (PWRGENIN)

The CA MICS Analyzer Option for VSE/POWER does not need an OPTION statement in sharedprefix.MICS.GENLIB(PWRGENIN). If, after running the product for six months, you need to perform special tailoring, see Chapter 6 of the CA MICS System Modification Guide.

7.3 Unit Level Parameters

Unit level parameters describe the structure and sources of IBM's VSE/POWER Accounting data.

This section contains information on the following unit level parameters for the CA MICS Analyzer Option for VSE/POWER:

1 - VSE/POWER Options (PWROPS)
2 - VSE/POWER Job Groups (PWRGROUP)
3 - VSE/POWER Job Group Routine (PWRGPRTE)
4 - VSE/POWER Job Turnaround Time Routine (PWRTNRTE)
5 - VSE/POWER Configuration (PWRCONFG)
6 - Input Source Definition (INPUTPWR)
7 - Generating PWR System Code (PWRPGEN)
8 - Database Space Modeling (DBMODEL)

7.3.1 VSE/POWER Options (PWROPS)

This section shows you how to specify the operational statements that control the CA MICS Analyzer Option for VSE/POWER processing.

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 PWROPS member describes the data to be processed by the DAY035 DAILY update step and allows you to customize the process. A worksheet for recording the statement values for PWROPS is shown in Figure 7-3.

The PWROPS member contains the following statements, described in detail below. A comment line is defined by an asterisk (*) as the first non-blank character.

```
Required Statements
-------------------
OPTIONS

Optional Statements
-------------------
SUSPENDLIMIT
```

7.3.1.1 OPTIONS Statement

The OPTIONS statement describes the attributes of IBM's VSE/Power accounting data that will be processed by the CA MICS Analyzer Option for VSE/Power. You must provide one OPTIONS statement for each VSE/Power system.

Consider consulting with your VSE systems programmer before implementing the PWROPS definition to ensure that the definition reflects your VSE system.

The OPTIONS statement has the following format:

```
OPTIONS orgsysid ddname vsesysid datefmt sysid_sw version
```

where:

orgsysid = The original system identifier as described in prefix.MICS.PARMS(SYSID) for this unit.

ddname = The ddname pointing to the data set containing the VSE/Power accounting data.

vsesysid = The single-character VSE/Power system identification contained in the SYSID record header, or X if there is no header.

If your VSE/Power system was generated with shared spool support, VSE/Power uses VSESYSID in all account records and in its responses to PDISPLAY requests. The value of VSESYSID was set in IBM's SYSID parameter of the POWER macro used to install VSE/Power. It may have been overridden by a SET statement in IBM's VSE/Power autostart function.

datefmt = The system date format as defined to VSE/Power; it must be either MM/DD/YY or DD/MM/YY. The DATE parameter of the STDOPT statement controls the format of the system date. The STDOPT statement is usually executed as part of a VSE system's Automatic System Initialization (ASI).

sysid_sw = Specify SYSID if the data contains a system identification header; if not, specify NOSYSID.

VSE/Power systems generated with shared spool support contain a system identification header.

To determine whether or not your system was generated with shared spool support, examine the SHARED parameter of the POWER macro used to install VSE/Power. A value of NO indicates that the system was generated without shared spool support. A value of (Q) indicates that the system was generated with shared spool support. A value of (Q,A) indicates that the system was generated with shared spool support and a shared accounting file.

version = The version and release of IBM's VSE/Power that produced the data. Valid values are 2.1.0, 2.2.0, 2.3.0, 4.1, 5.1, 5.2, 6.1, 6.3, 6.4, 6.5, 6.6, 6.7, 7.1, 8.1, 8.2, 8.3, and 9.1.

CA MICS requires that all data produced by VSE/Power systems that share the same queues be processed through a single DD statement. Each VSE/Power system has a unique VSE system identification. If the systems were generated with shared accounting file support (a value of (Q,A) on the SHARED parameter of IBM's POWER macro), your VSE/Power system will build only one POWER accounting file. However, if each system has its own POWER accounting file, the input from all systems sharing the same queues must be concatenated through a single DD statement for input to CA MICS.

Example:

```
OPTIONS  VSE1  INPUTVSE  1  MM/DD/YY  SYSID   7.1
OPTIONS  VSE2  INPUTVSE  2  MM/DD/YY  SYSID   7.1
OPTIONS  VSE3  INPUTVSE  3  MM/DD/YY  SYSID   7.1
OPTIONS  VSE4  INPUTVSE  4  MM/DD/YY  SYSID   7.1
OPTIONS  VSEA  INPUTA    X  MM/DD/YY  NOSYSID 5.2
```

Accounting data for the VSE/Power shared spool complex is read from ddname INPUTVSE. It is associated with ORGSYSID VSE1, VSE2, VSE3, or VSE4 depending upon the VSESYSID value of 1, 2, 3, or 4, respectively. All of the data from these systems is processed as if it were produced by a VSE/Power system at Version 7 Release 1. The system date format in the records is MM/DD/YY.

All of the records processed from ddname INPUTA will be associated with ORGSYSID VSEA. These records contain no SYSID header, have a system date format of MM/DD/YY, and will be processed as though they were produced by a VSE/Power system at Version 5.2.

The two DD statements in prefix.MICS.PARMS(INPUTPWR) will be for INPUTVSE and INPUTA. The PWRPGEN process will check the PWROPS and INPUTPWR members for a one-to-one correspondence of ddnames. See Section 7.3.6 of this guide for more information on INPUTPWR.

7.3.1.2 SUSPENDLIMIT Statement

The SUSPENDLIMIT statement specifies the maximum number of days that an incomplete job should remain in the VSE/POWER Job Suspend File (PWR_PJ). Refer to Section 2.2.3 of this guide for a description of the suspend process.

The SUSPENDLIMIT statement has the following format:

```
SUSPENDLIMIT days
```

where: days is the limit in days (DAILY updates) in the range of 0 to 99.

7.3.1.3 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 zero (0).
               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 as follows if neither MULTWORK nor NOMULT
parameters are specified:

MULTWORK _PJ _PP _PR PDA PJB POA PPG PRA DAT PGT


The following files are eligible for multiple WORK support:

PWR VSE/POWER Information Area Files

  _PJ    VSE/POWER Job Suspend File
  _PP    VSE/POWER Program Suspend File
  _PR    VSE/POWER Reader Suspend File
  PDA    VSE/POWER Device Activity File
  PJB    VSE/POWER Job Activity File
  POA    VSE/POWER Output Activity File
  PPG    VSE/POWER Program Activity File
  PRA    VSE/POWER Remote Activity File
  DAT    VSE/POWER Device Activity File (internal only)
  PGT    VSE/POWER Program Activity File (internal only)

The following section discusses changing the WORK option:

    1 - Change the Number of Work Files
```

7.3.1.3.1 Change the Number of Work Files

```
To change the number of work files used in the CA MICS
Analyzer Option for VSE/POWER processing in step DAY035,
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.

7.3.1.4 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 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

- 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.

- 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.

- 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.

- 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
```

7.3.1.4.1 Enable Internal Step Restart

```
To enable the internal step restart in the CA MICS Analyzer
Option for VSE/POWER, follow the checklist provided below:

   ************************************************
   *                                              *
   *          ENABLE INTERNAL STEP RESTART        *
   *                                              *
   ************************************************

___ 1.  Edit prefix.MICS.PARMS(cccOPS), where (ccc) is the
        component identifier, and specify:

        RESTART YES

        For additional information on related topic, review
        the documentation for this product on WORK,
        RESTARTWORK, and RESTARTCKPT parameters to override
        default data set allocation parameters.

___ 2.  Submit the job in prefix.MICS.CNTL(cccPGEN).

___ 3.  Edit prefix.MICS.PARMS(JCLGENU) so that it contains a
        single line that reads:

        DAILY

        or, if incremental update is enabled for this product
        in this unit database, specify:

        DAILY INCRccc

        where ccc is the product ID.

___ 4.  Submit the job in prefix.MICS.CNTL(JCLGENU).  Ensure
        that there are no error messages in MICSLOG or
        SYSTSPRT, that the MICSLOG contains the normal
        termination message, BAS10999I, and that the job
        completes with a condition code of zero.

___ 5.  The following operational job(s) have changed:

        DAILY

        INCRccc (if incremental update is enabled)

        If your site has implemented the operational CA MICS
        processes in a scheduling product, the JCL may have
```

```
to be refreshed in that product.  See the scheduling
product's administrator for the exact processes
involved in updating that product's representation of
the CA MICS jobs.
```

7.3.1.5 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

7.3.1.5.1 Implement Incremental Update

```
To implement incremental update in the CA MICS Analyzer
Option for VSE/POWER, 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.

7.3.1.6 Sample PWROPS

Following is a sample of required and optional statements
that may be coded in prefix.MICS.PARMS(PWROPS).

```
+-------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET: VSE/POWER Analyzer Options           |
|                                                                         |
| PARMS Library Member is PWROPS                                          |
| Reference Section:  7.3.1, CA MICS Analyzer Options for VSE/POWER Guide  |
+-------------------------------------------------------------------------+
|                                                                         |
| VSE/POWER ANALYZER PROCESSING OPTIONS:                                  |
|                                                                         |
| ORGSYSID   ________________                                             |
|                                                                         |
|   DDNAME   ________________                                             |
|                                                                         |
|   VSESYSID  __________ (X IF NOSYSID IS SPECIFIED FOR SYSID_SW)         |
|                                                                         |
|   DATEFMT   _____ (MM-DD-YY)  _____ (DD-MM-YY)                          |
|                                                                         |
|   SYSID_SW  _____ (SYSID)     _____ (NOSYSID)                           |
|                                                                         |
|   VERSION   _____ (2.1.0)     _____ (2.2.0)     _____ (2.3.0)           |
|                                                                         |
|                                                                         |
| ORGSYSID   ________________                                             |
|                                                                         |
|   DDNAME   ________________                                             |
|                                                                         |
|   VSESYSID  __________ (X IF NOSYSID IS SPECIFIED FOR SYSID_SW)         |
|                                                                         |
|   DATEFMT   _____ (MM-DD-YY)  _____ (DD-MM-YY)                          |
|                                                                         |
|   SYSID_SW  _____ (SYSID)     _____ (NOSYSID)                           |
|                                                                         |
|   VERSION   _____ (2.1.0)     _____ (2.2.0)     _____ (2.3.0)           |
|                                                                         |
|                                                                         |
+-------------------------------------------------------------------------+
| ....5...10...15...20...25...30...35...40...45...50...55...60...65...70.. |
+-------------------------------------------------------------------------+
```

Figure 7-3. VSE/POWER Options Definition Worksheet

7.3.2 VSE/POWER Job Groups (PWRGROUP)

The PWRGROUP parameter tells CA MICS how to calculate actual turnaround times and establishes job groups for your VSE/POWER data.

Users generally hold the data center responsible for those parts of a job's life that the data center has control over (such as the amount of time it takes for an operator to mount a tape). Because turnaround time definitions, though not universally definable, tend to reflect users' perceptions, they contain only job-related resource use times.

A job group provides a means of classifying batch job activity in a record by its service requirements, job resource requirements, or other factors.

The statement format presented here is followed by more detail later in this section.

STATEMENT FORMAT

The statement format for the PWRGROUP parameter is

```
TURNTIME  a  b  c
GROUP     d  e  f  g
DEADLINE  h  i  j  f  g
OPEN      k  f  g
```

where:

TURNTIME = the keyword for the turnaround service definition for standard and deadline jobs.

a = a time descriptor. Valid values are STANDARD (standard job -- GROUP statement below) and DEADLINE (a deadline job -- DEADLINE statement below).

b = the origin of the job. Valid values are LOCAL (submitted locally) and REMOTE (submitted from a remote site).

c = the list of elements used to determine actual job service. The list entries are numeric and must be separated by at least one blank. See Figure 7-4 for a list of valid values.

```
GROUP =      the keyword for standard job group (one with
             a conventional service requirement like 10
             minute turnaround).

d =          an identifier.  Valid values for standard job
             groups are integers from 1 to 149.

e =          the target turnaround time in minutes.  Valid
             values are integers that are less than six
             digits long.  This target must be greater than
             or equal to the target of the previous GROUP, if
             there is one.

f =          the service factor.  It allows the CA MICS
             Accounting and Chargeback product to perform
             priority accounting.

g =          a descriptive title for the job group.  Valid
             titles are up to 30 characters long and must
             be enclosed in single quotes.

DEADLINE =   the keyword for a deadline job group (one
             that must be completed within a stated time
             frame).

h =          an identifier.  Valid values are integers from
             150 through 179.

i =          a group made up of four sets of times.  The
             first two digits of i represent the starting
             hour of the deadline group.  The second two
             digits of i represent the starting minutes of
             the deadline group.  The third two digits of i
             represent the ending hour of the deadline group.
             The fourth two digits of i represent the
             ending minutes of the deadline group.  Valid
             hours values are integers from 00 through 23.
             Valid minutes values are integers from 00
             through 59.

j =          the number of days spanned for the deadline.
             Valid values are integers from 1 to 7, where 1
             denotes that the deadline falls in the same day.

OPEN =       the keyword for an open job group (one that
             does not lend itself to turnaround time
             studies).

k =          an identifier.  Valid values are integers from
```

180 through 195.

A sample PWRGROUP statement is

```
TURNTIME STANDARD LOCAL 1 2 3
GROUP      1 10 1 '10 MIN BATCH SERVICE'
DEADLINE 150 08 00 20 00 1 1 'IN BY 8 OUT BY 8 SERVICE'
OPEN     180 1 'CICS'
```

Actual turnaround times are calculated as the sum of elements 1, 2, and 3 (see Figure 7-4). The standard job group 1 is described as providing 10 minute batch service. Its turnaround time objective is 10 minutes and its service factor of 1 means that, for accounting purposes, this job is charged at its normal rate--neither discounted nor surcharged.

The deadline job group 150 provides for jobs that are in by 8:00 a.m. (08 00) and out by 8:00 p.m. (20 00). Jobs can span one day (jobs submitted by 8 a.m. Monday are due to complete by 8 p.m. Monday). The service factor of 1 again means that, for accounting purposes, the jobs are neither discounted nor surcharged.

Jobs from the CICS system fall into the open job group 180. They are charged at their normal rate (neither discounted nor surcharged).

More information on job group service evaluation is provided in Section 2.2.1 of this guide.

The worksheets in Figures 7-4 and 7-5 are provided to help you determine the values of these parameters.

```
+-------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET: Turnaround Time Definition           |
|                                                                         |
| PARMS Library Member is PWRGROUP                                        |
| Reference Section:  7.3.2, CA MICS Analyzer Option for VSE/POWER Guide  |
+-------------------------------------------------------------------------+
|                                                                         |
| The following comments give the meanings of the numbers to be coded     |
| for Standard Job Groups.  The data element, the name in parentheses,    |
| is available in the PWRPJB file.                                        |
|                                                                         |
|      1 - Reader Time          (PJBRDRTM)                                |
|      2 - Input Queue Time     (PJBINQTM)                                |
|      3 - Job Execution Time   (PJBEXCTM)                                |
|                                                                         |
```

```
| The turnaround time for a STANDARD job group is defined as the sum of    |
| the elements listed below.                                               |
|                                                                          |
| TURNTIME STANDARD LOCAL  _ _ _                                           |
| TURNTIME STANDARD REMOTE _ _ _                                           |
|__________________________________________________________________________|
|                                                                          |
| The following comments give  the meanings of the allowable starting      |
| time stamp numbers for DEADLINE Job Groups:                              |
|                                                                          |
|      1 - Reader Time Stamp (RDRTS)                                       |
|      2 - Job Reader End Time Stamp (PJBRENTS)                            |
|      3 - Start Time Stamp (STARTTS)                                      |
|                                                                          |
| Allowable ending time stamp numbers and their meanings are:              |
|                                                                          |
|      3 - Start Time Stamp (STARTTS)                                      |
|      4 - End Time Stamp (ENDTS)                                          |
|                                                                          |
| You cannot specify "3" as both the start and end time stamp              |
| designator.                                                              |
|                                                                          |
| TURNTIME DEADLINE LOCAL  _ _                                             |
| TURNTIME DEADLINE REMOTE _ _                                             |
|                                                                          |
+--------------------------------------------------------------------------+
| ....5...10...15...20...25...30...35...40...45...50...55...60...65...70.. |
+--------------------------------------------------------------------------+
```

Figure 7-4. VSE/POWER Job Group Worksheet, Turnaround Time Definition

```
+--------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET:  VSE/POWER Job Groups                 |
|                                                                          |
| PARMS Library Member is PWRGROUP                                         |
| Reference Section:  7.3.2, CA MICS Analyzer Option for VSE/POWER Guide   |
+--------------------------------------------------------------------------+
|  STANDARD Job Groups (# is 1-149)                                        |
|                                                                          |
|             Turn Time                                                    |
|              Target   Service   Descriptive Title                        |
|         #    (mins)   Factor                                             |
|                                                                          |
|  GROUP ___   ______   _____    '_____________________________________'   |
|                                                                          |
|  GROUP ___   ______   _____    '_____________________________________'   |
|                                                                          |
```

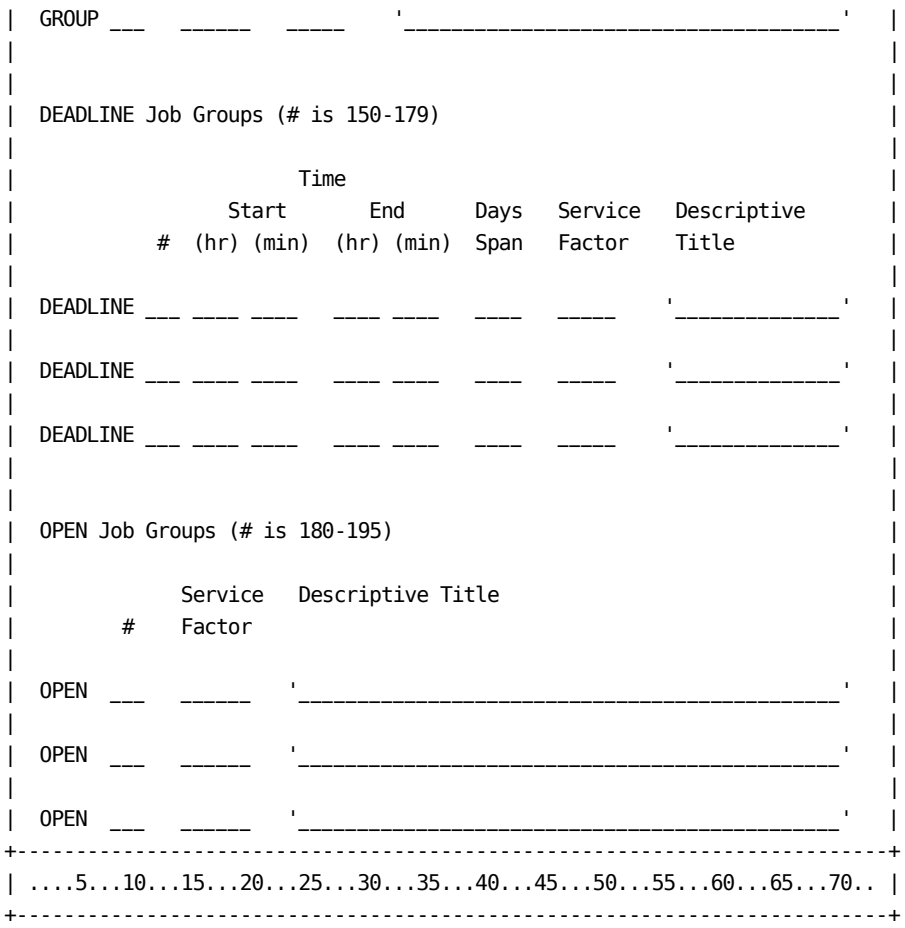


Figure 7-5. VSE/POWER Job Group Worksheet, VSE/POWER Job Groups Definition

7.3.3 VSE/POWER Job Group Routine (PWRGPRTE)

The VSE/POWER Job Group Routine assigns a value to the data element JOBGROUP for each observation in the VSE/POWER Job Activity (PWRPJB) and VSE/POWER Job Suspend (PWR_PJ) Files. The routine does NOT have to consider the computation of job turnaround time or the determination of the JOBGROUP service received, as these tasks are handled automatically by CA MICS through the parameters in prefix.MICS.PARMS(PWRGROUP).

This routine is coded in SAS and is placed in prefix.MICS.PARMS(PWRGPRTE). The routine must assign the correct JOBGROUP to every job passed to it; testing the routine is your responsibility.

Your VSE/POWER job group routine has each of the data elements contained in the VSE/POWER Job Activity (PWRPJB) File available to it, as well as the data element VSEDDNAM. VSEDDNAM contains the DDname used to read the VSE/POWER accounting records that are used to build the observation. The data elements most often used to determine JOBGROUP classification are:

```
JOB      - Job Identification (jobname)
JOBCLASS - Job Input Class
PWRJUSER - POWER Job User Information
VSEPRTID - VSE Partition Identifier
```

IMPORTANT NOTE: Your code must not modify the values contained in the fields listed above under ANY circumstances. The only field that should be modified is the data element JOBGROUP, which must be set to the correct value for the job which was executed. No CA MICS user exit may issue a RETURN statement or exit in any fashion other than "falling off at the bottom".

EXAMPLE

An installation commits to turnaround times based on the job class specified for the service requested. The relevant portion of this shop's specification in prefix.MICS.PARMS(PWRGROUP) is:

```
GROUP 1     10 1 '10 MIN BATCH SERVICE'         (class A)
GROUP 3     30 1 '30 MIN BATCH SERVICE'         (class B)
GROUP 5     60 1 '60 MIN BATCH SERVICE'         (class C)
GROUP 7    240 1 '4 HOUR BATCH SERVICE'         (class D)
GROUP 9   1440 1 '24 HOUR BATCH SERVICE'        (class E)
GROUP 11 10080 1 '7 DAY BATCH SERVICE'          (class F)
```

```
    GROUP 13 10080 1 'OTHER'
```

Assuming that the above job groups correspond to job classes A through F, the following SAS code would serve as this site's PWRGPRTE exit:

```
    IF JOBCLASS='A' THEN JOBGROUP=1 ;
    ELSE IF JOBCLASS='B' THEN JOBGROUP=3 ;
    ELSE IF JOBCLASS='C' THEN JOBGROUP=5 ;
    ELSE IF JOBCLASS='D' THEN JOBGROUP=7 ;
    ELSE IF JOBCLASS='E' THEN JOBGROUP=9 ;
    ELSE IF JOBCLASS='F' THEN JOBGROUP=11;
    ELSE JOBGROUP=13;
```

Jobs not matching the standard six groups that were defined are assigned to job group 13. This "defensive programming" should always be done.

7.3.4 VSE/POWER Job Turnaround Time Routine (PWRTNRTE)

The turnaround time routine PWRTNRTE provides flexibility for those sites whose turnaround time calculations cannot be performed by the TURNTIME statement of PWRGROUP.

This routine can use any of the data elements available in the VSE/POWER Job Activity File (PWRPJB); however, you can only modify the contents of the elements PJBTURTM, SUBMITTS, and PURGETS.

For jobs in a standard job group, the routine is entered after the element PJBTURTM is calculated. This exit may modify PJBTURTM.

For jobs in a DEADLINE job group, the routine is entered after the calculation of the elements SUBMITTS and PURGETS, the time the job is considered to be available to the system and the time the job is out of the system, respectively. This exit may modify either or both of these elements. PJBTURTM will be missing for jobs in a DEADLINE job group because the element does not have meaning for a DEADLINE job group.

For jobs in an OPEN job group, the routine is entered with PJBTURTM, SUBMITTS, and PURGETS all having missing values because these elements do not have meaning for an OPEN job group.

Code your routine in SAS, following the guidelines for coding exits documented in Section 2.3.1.5 of the CA MICS Planning, Installation, Operation, and Maintenance Guide. Save your routine in prefix.MICS.PARMS(PWRTNRTE).

7.3.5 VSE/POWER Configuration (PWRCONFG)

The VSE/POWER configuration member PWRCONFG contains a list, by VSESYSID, of device addresses with their device types and device classes.

The information is used to accumulate a table of start I/O counts (SIOs) by device class which is used to build the VSE/POWER Program Activity (PWRPPG) and VSE/POWER Job Activity (PWRPJB) Files. In addition, the value of the device class is used as a summarization key to the VSE/POWER Device Activity (PWRPDA) File.

A VSE/POWER configuration sample and a worksheet to help you build this member is shown in Figure 7-6. The fields are described after the figure.

```
+-------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET:  VSE/POWER Configuration Worksheet |
|                                                                         |
| PARMS Library Member is PWRCONFG                                        |
| Reference Section:  7.3.5, CA MICS Analyzer Option for VSE/POWER Guide |
+-------------------------------------------------------------------------+
| SAMPLE:                                                                 |
|                                                                         |
| *                                                                       |
| * ORGSYSID   DEVICE        DEVICE   DEVICE                              |
| *            ADDRESS(ES)   CLASS    TYPE                              |
| *                                                                       |
|   VSE1       00C           SPOOL    2501                              |
|   VSE1       01F           SPOOL    3211                              |
|   VSE1       140-14F       DASD     3350                              |
|   VSE1       150-15F       DASD/FBA 3370                              |
|   VSE1       380-38F       TAPE     3420                              |
|   VSE2       00C           SPOOL    2501                              |
|   VSE2       01F           SPOOL    3211                              |
|   VSE2       140-14F       DASD     3350                              |
|   VSE2       150-15F       DASD/FBA 3370                              |
|   VSE2       380-38F       TAPE     3420                              |
|   VSE3       00C           SPOOL    2501                              |
|   VSE3       01F           SPOOL    3211                              |
|   VSE3       140-14F       DASD     3350                              |
|   VSE3       150-15F       DASD/FBA 3370                              |
|   VSE3       380-38F       TAPE     3420                              |
|   VSE4       00C           SPOOL    2501                              |
|   VSE4       01F           SPOOL    3211                              |
|   VSE4       140-14F       DASD     3350                              |
|   VSE4       150-15F       DASD/FBA 3370                              |
```

```
|   VSE4       380-38F       TAPE     3420                              |
|                                                                        |
| WORKSHEET:                                                             |
|                                                                        |
|              DEVICE        DEVICE   DEVICE                            |
|   ORGSYSID   ADDRESS(ES)   CLASS    TYPE                              |
|                                                                        |
|   _____      _____         _____    _____                             |
|   _____      _____         _____    _____                             |
|   _____      _____         _____    _____                             |
|                                                                        |
+------------------------------------------------------------------------+
| ....5...10...15...20...25...30...35...40...45...50...55...70..         |
+------------------------------------------------------------------------+
```

Figure 7-6. VSE/POWER Configuration Sample and Worksheet

The following is a description of the fields in Figure 7-6:

ORGSYSID: Original system identifier as described in prefix.MICS.PARMS(SYSID) for this unit.

DEVICE ADDRESS(ES): Value representing the location of the device.

DEVICE CLASS: Recognized values are COMMGEAR, DASD, DASD/FBA, GRAPHICS, SPECIAL, SPOOL, TAPE, and UNIT-REC. Other values may be used (see comments below).

DEVICE TYPE: The device model (e.g., 3380).

DASD and DASD/FBA (Fixed Block Architecture) devices:

2305-1	2305-2	2311	2314	2314/19	2319
3310	3330	3330V	3330-11	3340	3344
3350	3370	3375	3380	3380D	3380E
3390	3390-1	3390-2	3390-3	3390-9	

TAPE devices:

2400	2401	2402	2403	2415	2420
3400	3410	3411	3420	3430	3480
3490	3490E	8809			

SPOOL and UNIT-REC devices:

1052	1219	1255	1259	1275-PC	1275-SC
1287	1288	1403	1404	1403/4	1419-PC

```
  1442       1443       2150       2495       2501       2520
  2540       2540-R     2540-P     2671       3102       3203
  3210       3211       3213       3215       3230       3262
  3268       3505       3525       3694       3800       3800-3
  3820       3838       3851       3881       3886       3890
  3895       3896       4245       4248       4250       7436

GRAPHICS devices:

  1053       2250       2260       3036       3158       3180
  3250       3270       3277       3278       3279       3284
  3286       3287       328X       7350

COMMGEAR devices:

  ICA        270X       2701       2702       2703       3705
  3725       37X5       7171       7770       SERIES1

SPECIAL devices:

  CTC        3088       7443
```

CA MICS performs separate validations on the device type and device class values for each device address. If the device type value is not recognized or if the device class and device type combination is not valid for the address, a warning message is issued. Processing continues, but the SIO counts for the device will be accumulated into the OTHER category. Chapter 2, Usage Guidelines, discusses the validation and SIO counts in greater detail.

7.3.6 Input Source Definition (INPUTPWR)

The member INPUTPWR describes the location of the input data source used by the CA MICS Analyzer Option for VSE/POWER. The member consists of the prototype JCL that is incorporated into the DAILY job via JCLGEN.

You must code one DD statement for each ddname referenced in the prefix.MICS.PARMS(PWROPS) member (see Section 7.3.1). The PWRPGEN job will cross-check the DD statements in the INPUTPWR member and the ddnames referenced in the PWROPS member. An ABEND will occur if there is not a one-to-one correspondence of ddnames between these two members.

For example, assume the following statements were coded in prefix.MICS.PARMS(PWROPS):

```
OPTIONS  VSE1  INPUTVSE  1  DD/MM/YY  SYSID   7.1
OPTIONS  VSE2  INPUTVSE  2  DD/MM/YY  SYSID   7.1
OPTIONS  VSE3  INPUTVSE  3  DD/MM/YY  SYSID   7.1
OPTIONS  VSE4  INPUTVSE  4  DD/MM/YY  SYSID   7.1
OPTIONS  VSEA  INPUTA    X  MM/DD/YY  NOSYSID 5.2
```

A corresponding INPUTPWR member may be:

```
//INPUTVSE  DD  DISP=SHR,DSN=SHARED.SPOOL.VSE.INPUT
//INPUTA    DD  DISP=SHR,DSN=TEST.SYSTEM.INPUT
```

Getting the input data to be processed each day into the data set you specify here is your responsibility. See Section 2.1 of this guide for more information on the collection and preparation of VSE/POWER input data.

7.3.7 Generating PWR System Code (PWRPGEN)

The cccPGEN process activates the specifications you coded in the members of the prefix.MICS.PARMS library using system code generator routines.

Before submitting the PWRPGEN job, you must code the input members, listed below, in the prefix.MICS.PARMS library:

- PWR Product Options Definition (PWROPS)
- PWR Job Group Definition (PWRGROUP, PWRGPRTE, PWRTNRTE)

The chart below identifiers the members input by the PWRPGEN job and the generated output members, output macros (a member may contain the definition of more than one SAS macro), and SAS formats that are generated. You must code the input members prior to submitting the PWRPGEN job. The output libraries are at the database unit level (prefix) unless noted otherwise.

System Code Generator	prefix.MICS.PARMS Input Member	prefix.MICS.USER. SOURCE Output Members and SAS Macro Names	prefix.MICS. MUOLIB Output SAS Formats
PWRPGEN	PWROPS	$PWRMSTR PWRLOOP PWRTURN PWRGRP	
	PWRGROUP		$PWRCNFG PWRTURN $PWRGRP SF_PG

7.3.8 Database Space Modeling (DBMODEL)

This section describes the information required to define the Analyzer to the Database Space Modeling Facility. Specifically, the user must provide values for the cycle (data retention) definitions. The special considerations for defining the FILE input statements to the Database Space Modeling facility are discussed.

Section 2.3.4 of the PIOM, Database Space Modeling Facility, explains how these values are used in estimating the DASD requirements of the database, and how the user can input these values to the modeling facility.

This section covers the following topics:

1 - Data Retention Specifications (FILE Statements)

2 - DBMODEL Input Statements

7.3.8.1 Data Retention Specifications (FILE Statements)

Data retention specifications tell the CA MICS database how many cycles of data, for each file in each supported timespan, to save both online and in archive mode. Figure 7-7 is a worksheet that helps you collect this information.

The suggested retention period for each file is shown in parentheses for each timespan (DETAIL, DAYS, WEEKS, MONTHS, YEARS, and the TABLES area). If 00 shows as the retention value for the timespan, the file is not supported in the timespan. To add support, you must perform database tailoring as described in Chapter 6 of the CA MICS System Modification Guide.

```
+-------------------------------------------------------------------------+
| INSTALLATION PREPARATION WORKSHEET: Data Retention Specifications       |
|                                                                         |
| PARMS Library Member is DBMODEL                                         |
|                                                                         |
| Reference Section:  7.3.8.1, CA MICS Analyzer Option for VSE/POWER Guide |
+------+-----------------------------------------+-------------------+
|      |          Online Database Retention      |  Archive Cut-Off  |
| File |DETAIL    DAYS   WEEKS   MONTHS  YEARS  TABLES | WEEKS    MONTHS   |
| Name | __(NA)  __(NA)  __(NA)  __(NA)   _(NA)  __(NA)|___ (NA)  ___ (NA) |
+------+-----------------------------------------+-------------------+
|PWRPJB| __(10)  __(00)  __(00)  __(06)   _(01)  __(00)|___ (000) ___ (024)|
|PWR_PJ| 01(01)  00(00)  00(00)  00(00)   0(00)  00(00)|000 (000) 000 (000)|
|PWRPPG| __(10)  __(00)  __(00)  __(02)   _(00)  __(00)|___ (000) ___ (006)|
|PWRPOA| __(10)  __(00)  __(00)  __(06)   _(01)  __(00)|___ (000) ___ (024)|
|PWRPRA| __(10)  __(00)  __(00)  __(06)   _(00)  __(00)|___ (000) ___ (024)|
|PWRPDA| 00(00)  __(10)  __(09)  __(06)   _(00)  __(00)|___ (027) ___ (012)|
+------+-----------------------------------------+-------------------+
| ....5...10...15...20...25...30...35...40...45...50...55...60...65...70.. |
+-------------------------------------------------------------------------+
```

Figure 7-7. Data Retention Specifications Worksheet

The statement format is

```
  FILE ccc cccfff xx dd ww mm yy tt hw hm
```

where:

ccc = the product identifier. The VSE/POWER product's identifier is PWR.

cccfff = the file name. The Analyzer Option for VSE/POWER has six files, PWRfff. Valid values

for fff are documented in Chapter 5, Files.

xx = the number of days of data being retained in the DETAIL timespan of the database for the specified file. Valid values are integers.

dd = the number of days of data being retained in the DAYS timespan of the database for the specified file. Valid values are integers.

ww = the number of weeks of data being retained in the WEEKS timespan of the database for the specified file. Valid values are integers.

mm = the number of months of data being retained in the MONTHS timespan of the database for the specified file. Valid values are integers.

yy = the number of years of data being retained in the YEARS timespan of the database for the specified file. Valid values are integers.

tt = the number of levels of data being retained in the TABLES data area of the database for the specified file. Valid values are integers. All database units must have the same value for any given file.

hw = the number of weeks of data being retained on the archive weekly history file. Valid values are integers.

hm = the number of months of data being retained on the archive monthly history file. Valid values are integers.

For more information about FILE statements, see Section 2.3.4.1.2 of the CA MICS Planning, Installation, Operation, and Maintenance Guide.

7.3.8.2 DBMODEL Input Statements

The DBMODEL member of prefix.MICS.PARMS provides the input to the Database Space Modeling Facility.

Update the DBMODEL member using the information collected on the worksheet in Figure 7-7.

To actually perform the space modeling, submit the jobs as described in the CA MICS Planning, Installation, Operation, and Maintenance Guide, Section 2.3.4.2.

Chapter 8: INSTALLATION

The CA MICS Analyzer Option for VSE/Power is installed using the checklists in Section 3.8 of the PIOM and the information collected on the worksheets in Chapter 7, Parameters, of this product guide.

Chapter 9: PROCESSING

The processing information in this chapter provides an overview of how data from IBM's VSE/POWER accounting files becomes part of the CA MICS Database.

This section contains the following topics:

9.1 Processing Overview

The CA MICS Analyzer Option for VSE/POWER makes use of the standard database update and summarization facilities of CA MICS to maintain its data in the online and offline databases. VSE/POWER data is supplied to CA MICS in the DAY035 step of the DAILY database update job.

The daily update step:

- Reads and formats the raw data
- Maintains data integrity by eliminating duplicate data
- Summarizes device activity by hour
- Summarizes step activity by job
- Builds the accounting journal files if the CA MICS Accounting and Chargeback product is used
- Adds new cycles to the DETAIL and DAYS timespans
- Updates the week-to-date and month-to-date cycles.

A critical aspect of summarizing step activity is suspend processing. Suspend processing manages information on jobs which were still executing at the time that the VSE/POWER data was collected for CA MICS. For more information on suspend processing, see Section 2.2.3 of this guide. A more detailed description of DAY035 processing is presented in the next section.

Weekly processing is performed by the WEEK035 and WEEK300 steps. The WEEK035 step splits the week-to-date files into a new week-to-date and a new weekly 01 cycle.

The WEEK300 step creates the offline database files. As distributed, the Analyzer Option for VSE/POWER produces the following files:

```
  tapeprefix.MICS.AUDIT.PWRPJB - Job Activity Audit                    00065000
  tapeprefix.MICS.AUDIT.PWRPOA - Output Activity Audit                 00071000
  tapeprefix.MICS.AUDIT.PWRPPG - Program Activity Audit                00077000
  tapeprefix.MICS.HISTW.PWRPDA - Device Activity History               00092000
```

Monthly processing is performed in steps MONTH035 and MONTH300. The MONTH035 step splits the month-to-date files into a new month-to-date and a new cycle 01. It also updates the year-to-date cycle.

The monthly offline database files are created by the MONTH300 step. As distributed, the Analyzer Option for VSE/POWER produces the following files:

```
  tapeprefix.MICS.HISTM.PWRPDA - Device Activity History              00104000
  tapeprefix.MICS.HISTM.PWRPJB - Job Activity History                 00109000
  tapeprefix.MICS.HISTM.PWRPOA - Output Activity History              00114000
  tapeprefix.MICS.HISTM.PWRPPG - Program Activity History             00119000
  tapeprefix.MICS.HISTM.PWRPRA - RJE Activity History                 00124000
```

Finally, the optional yearly processing is performed by the YEAR035 step, which splits the year-to-date files into a new year-to-date and a new cycle 01.

9.2 Daily Update Processing Flow

The DAY035 step of the DAILY job processes information in the following phases using the routines shown:

- Phase 1 - Input Raw Data
  DYPWRFMT to format the POWER data.

- Phase 2 - Information Area Processing
  DYPWRFMT to eliminate duplicate data, summarize device and step activity, and build the journal files used by the CA MICS Accounting and Chargeback product.

- Phase 3 - Data Base Timespan Processing
  DYPWRSUM to create DAYS timespan files from DETAIL and to update week-to-date and month-to-date information where applicable.

- Phase 4 - File Aging
  DYPWRAGE to age the DETAIL and DAYS cycles, deleting the oldest cycle. This routine also replaces the week-to-date and month-to-date cycles.

The first four of the sections which follow describe the general logic flow in each of the phases, as well as the location of standard user routines and user exits. In the figures contained in these sections, exits that are user-modifiable are shown in boxes whose corners are marked by asterisks (*). The final section describes the operational reports produced by step DAY035.

1 - Input Raw Data Phase
2 - VSE/POWER Information Area Processing Phase
3 - Data Base Timespan Update Phase
4 - File Aging Phase
5 - Operational Reports

9.2.1 Input Raw Data Phase

In the Input Raw Data Phase, records are read from VSE/POWER Accounting files using the INPUTPWR DD statements and the DYPWRFMT routine. The data is processed and formatted to create SAS observations.

The following user exits are invoked in this phase:

- General Exits

  _USRSEL - input record selection

- Output Exits

  USRDPDA - PWRPDA element derivation
  USRDPPG - PWRPPG element derivation
  USRDPOA - PWRPOA element derivation
  USRSPRA - PWRPRA record selection
  USRDPJB - PWR_PR element derivation

- Parameter-Related Exits

  PWRACRT - account code derivation (PWRPOA)

This phase is shown in Figures 9-1 and 9-2.

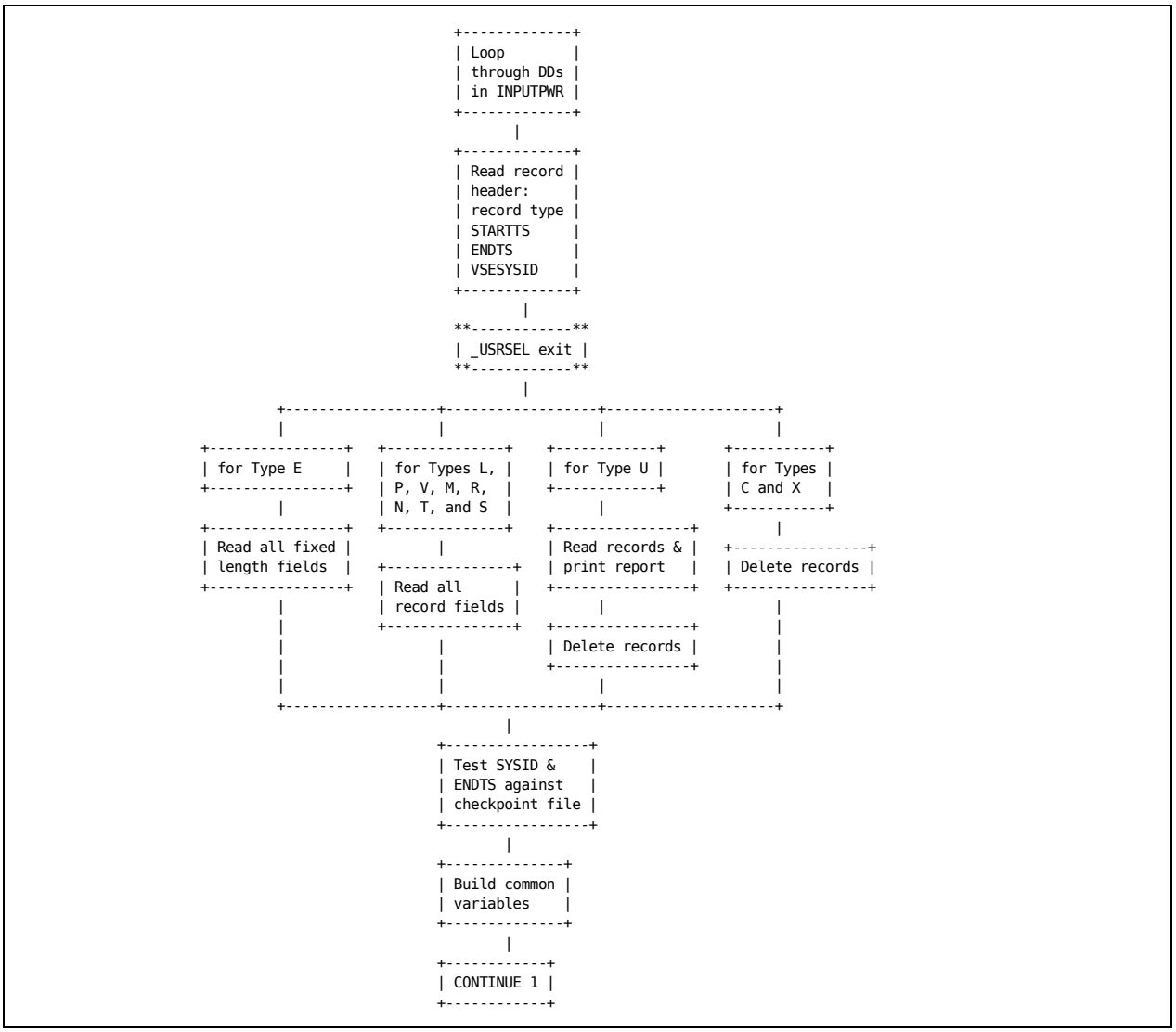


Figure 9-1. Input Raw Data Phase (Part 1 of 2)

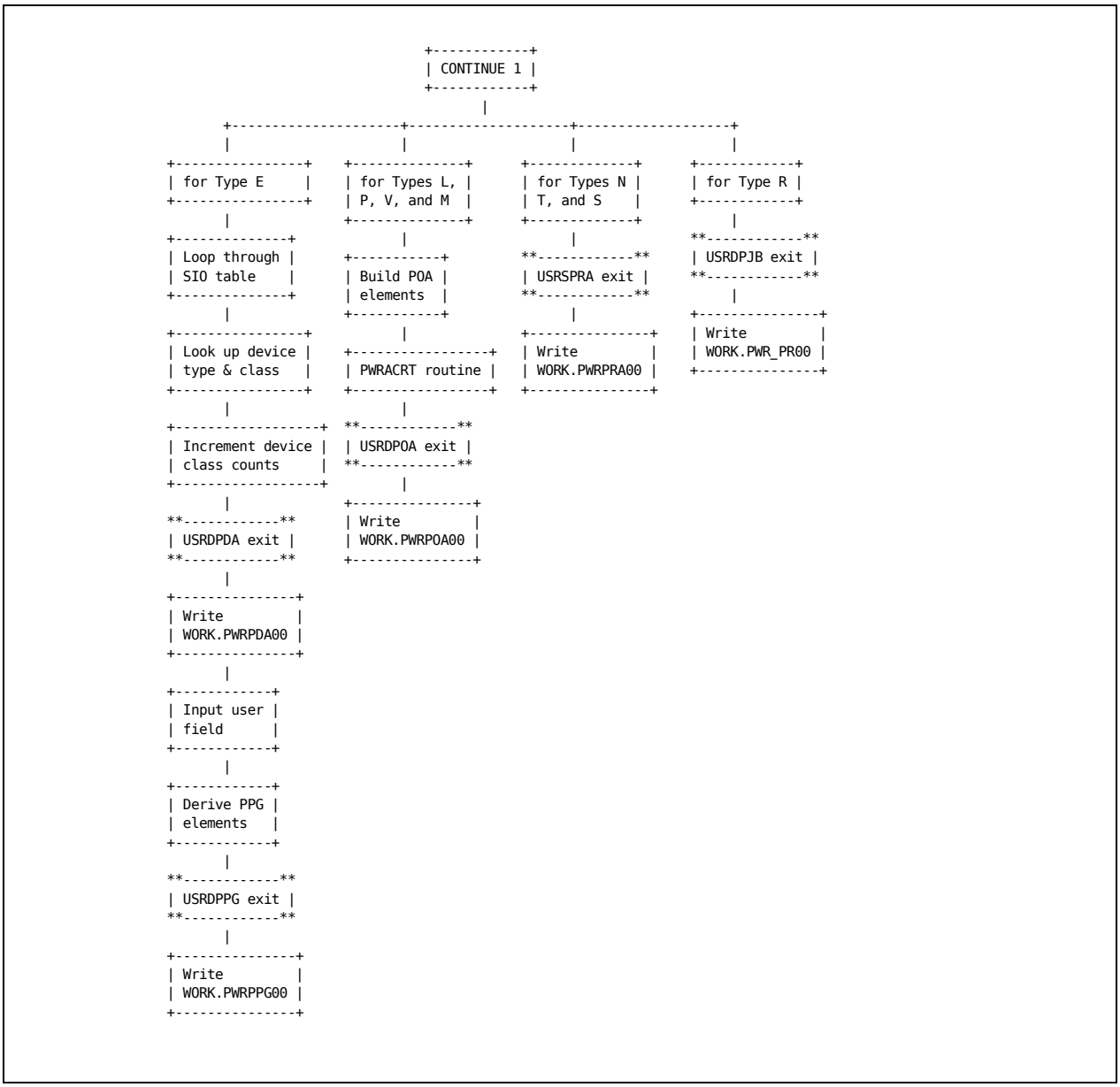


Figure 9-2. Input Raw Data Phase (Part 2 of 2)

9.2.2 VSE/POWER Information Area Processing Phase

The VSE/POWER Information Area Processing Phase takes the work files from Phase I and manipulates them to produce the new cycles for the DETAIL timespan. This is also part of the DYPWRFMT module.

The following user exits are invoked in this phase:

- Output Exits

  USRSPOA - PWRPOA record selection
  USRSPDA - PWRPDA record selection
  USRSPPG - PWRPPG record selection
  USRSPJB - PWRPJB record selection

- Parameter-Related Exits

  PWRACRT - account code derivation (PWRPPG)
  PWRGPRTE - job group derivation
  PWRTNRTE - job turnaround derivation
  PWRACRT - account code derivation (PWRPJB)

The interface to the CA MICS Accounting and Chargeback product contains the following exits that are not normally modified:

_USRUJPO - PWRPOA accounting interface
_USRUJPP - PWRPPG accounting interface
_USRUJPJ - PWRPJB accounting interface

Figures 9-3 and 9-4 illustrate this phase.

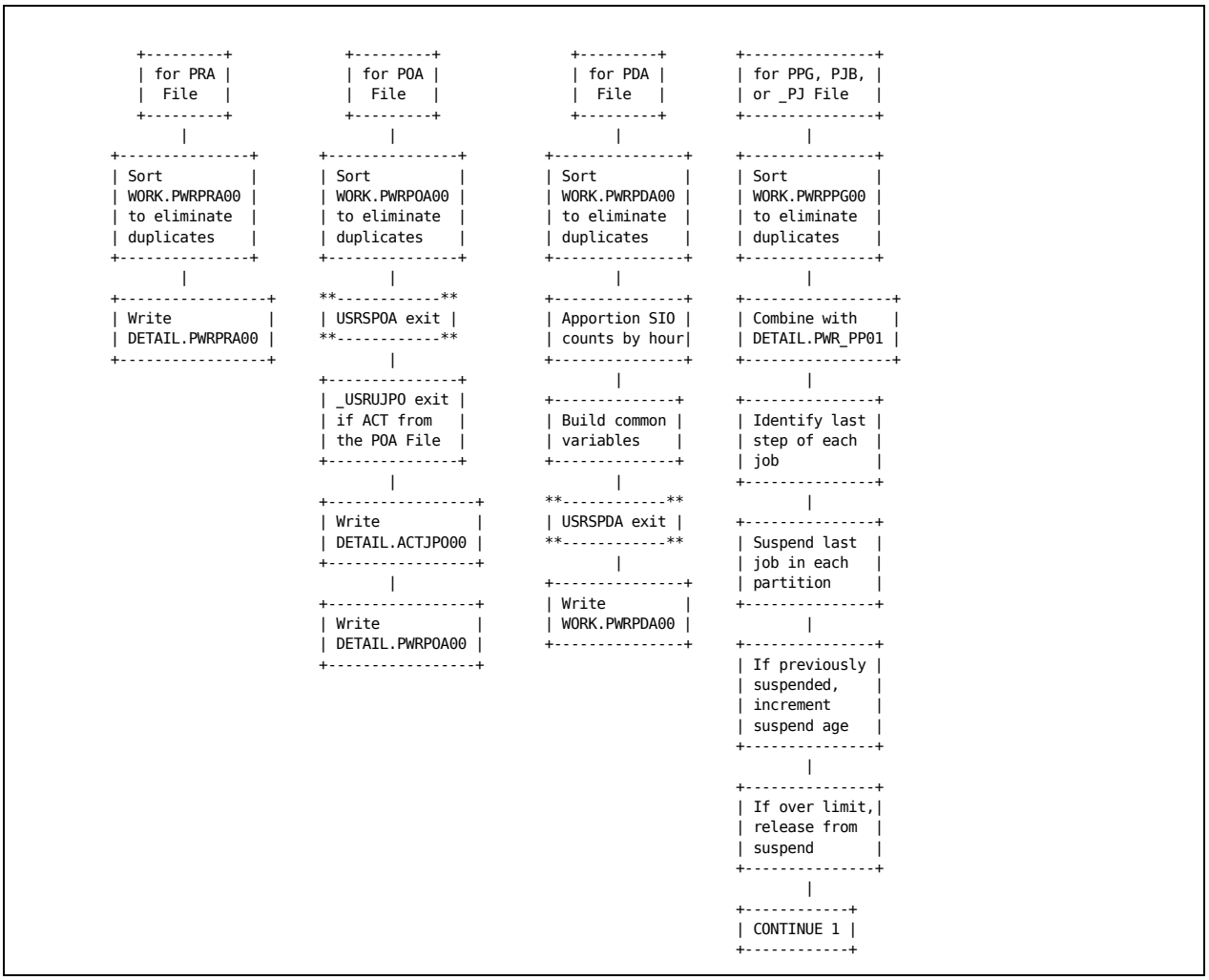


Figure 9-3. VSE/POWER Information Area Processing Phase (Part 1 of 2)

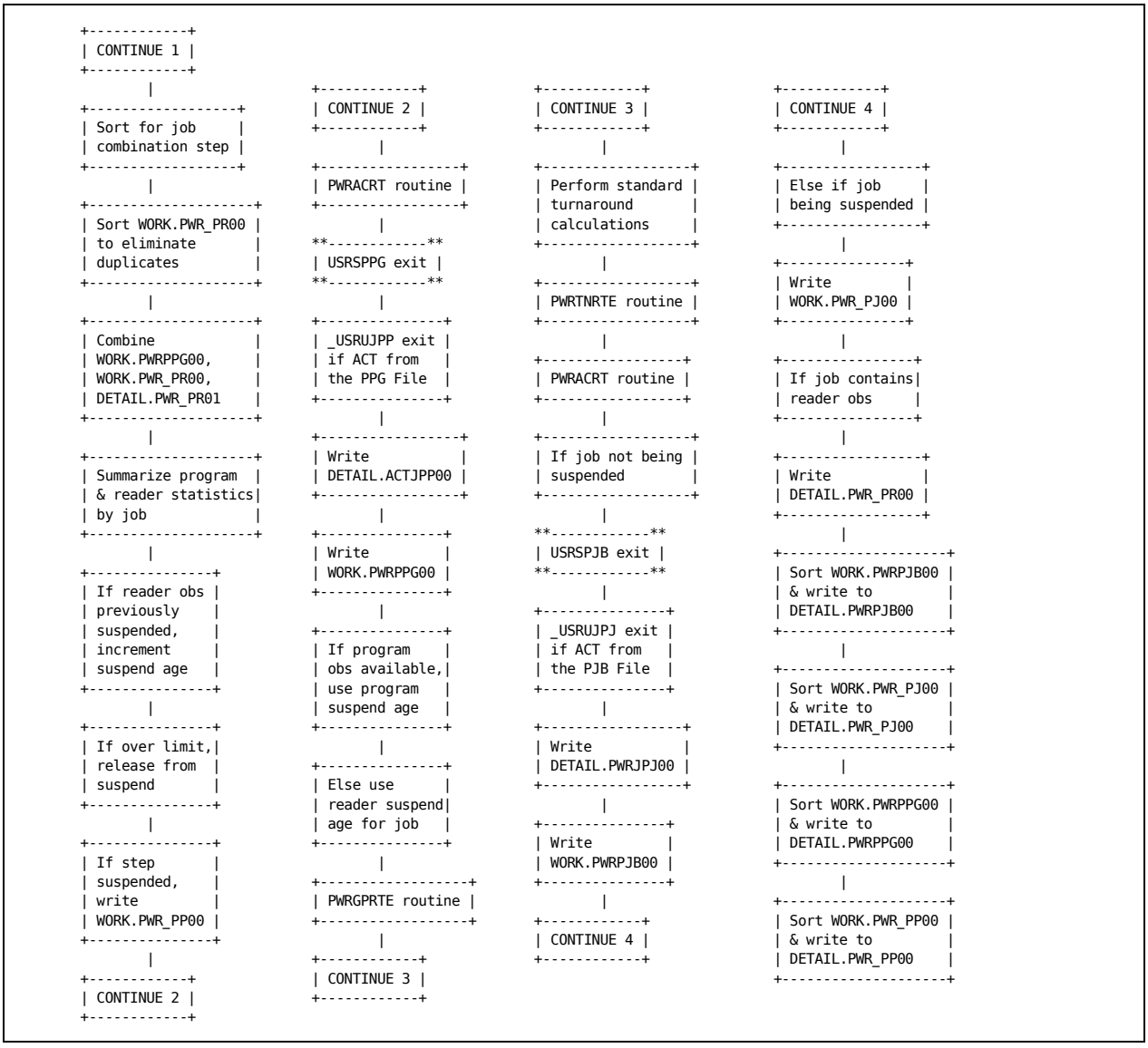


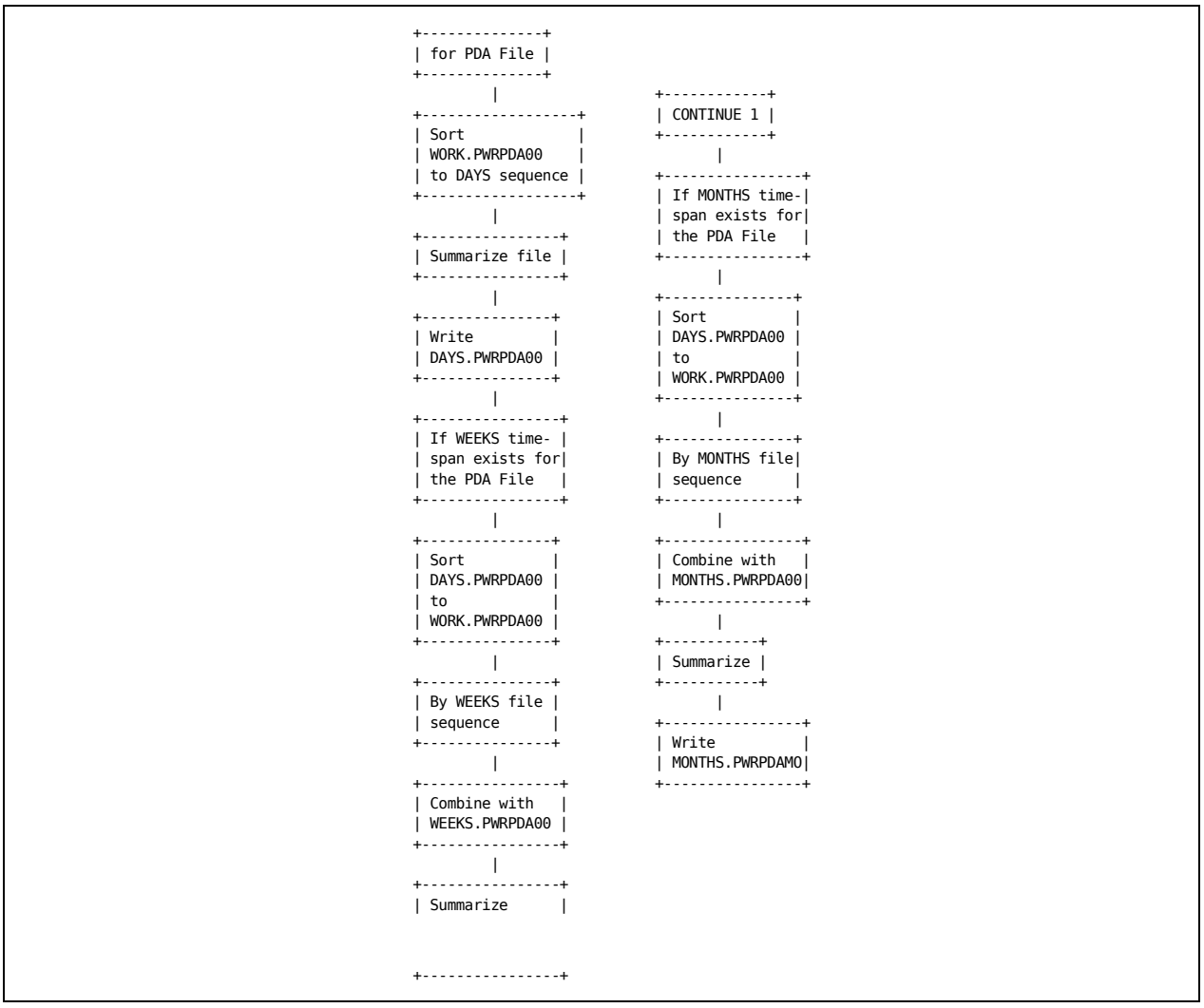
Figure 9-4. VSE/POWER Information Area Processing Phase (Part 2 of 2)

9.2.3 Database Timespan Update Phase

Processing in the Database Timespan Update Phase is controlled by the DYPWRSUM module. However, its first action is to include the #PDADSUM module to build the new DAYS cycle from the PWRPDA working file. This extra step is required because the file does not exist in the DETAIL timespan. The remaining processing updates the week-to-date and month-to-date cycles of each file as required.

There are no user exits invoked in this phase.

Figures 9-5 and 9-6 depict the database timespan update phase.



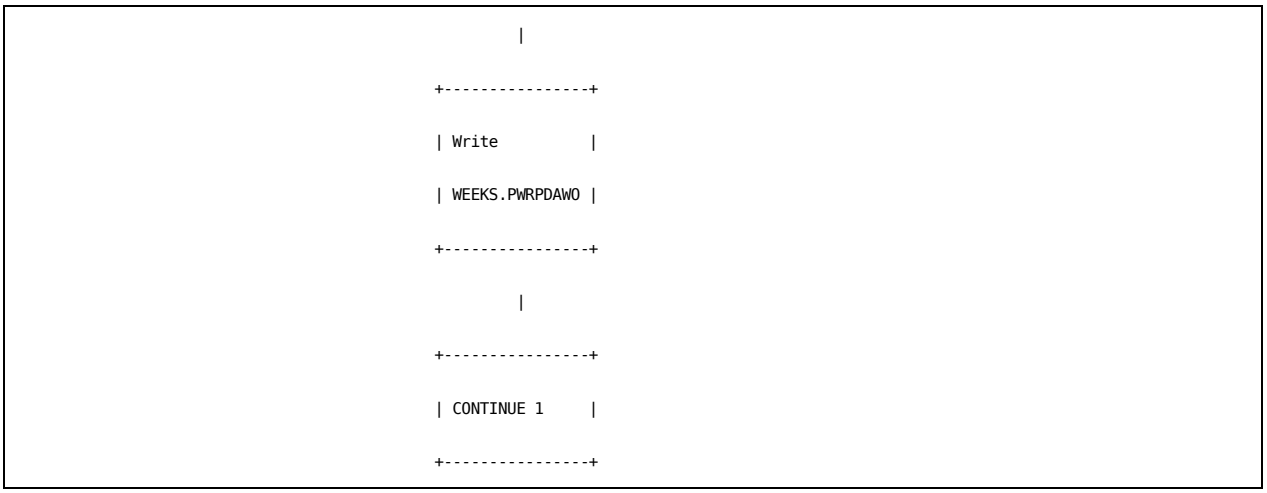
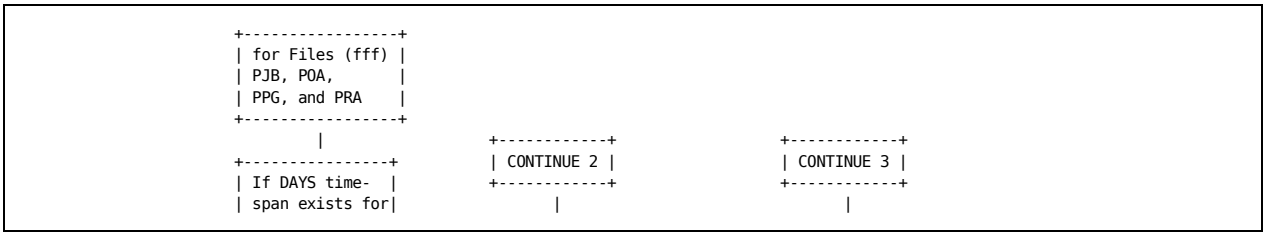


Figure 9-5. Database Timespan Update Phase (Part 1 of 2)



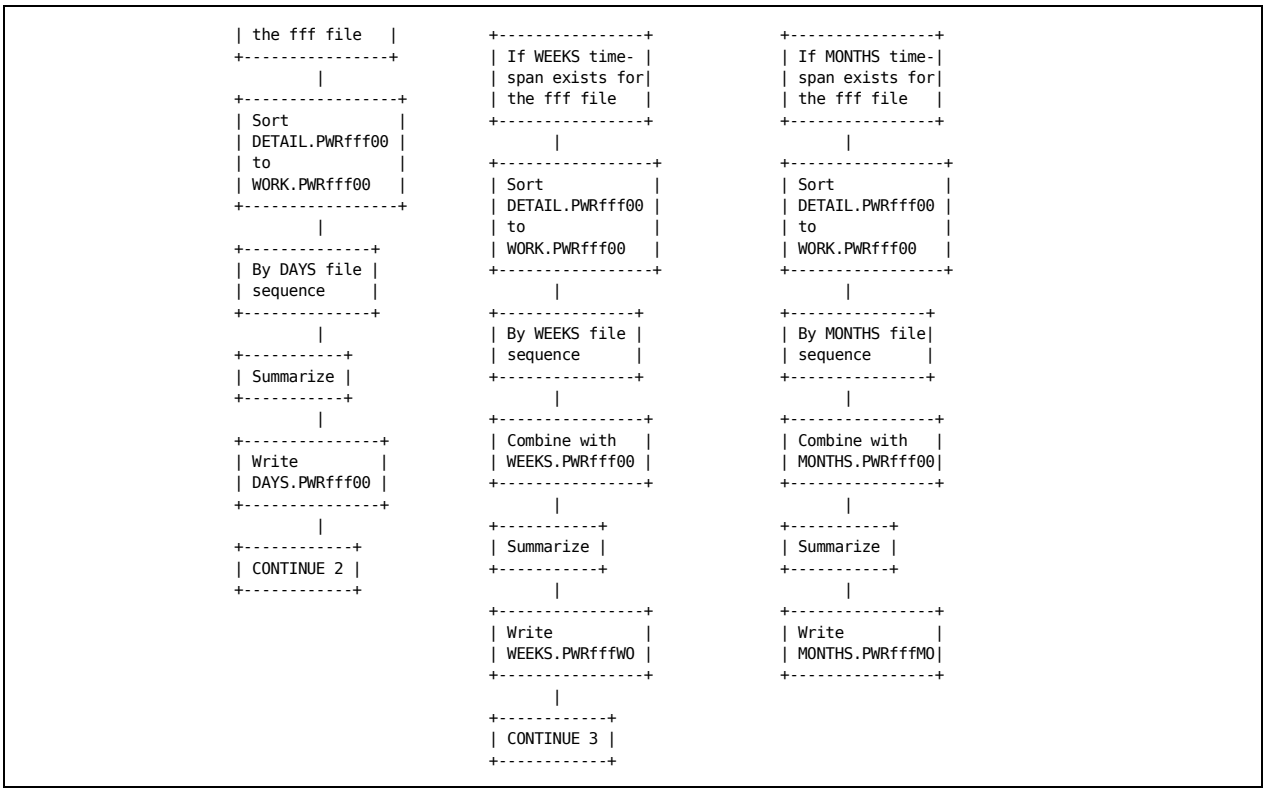


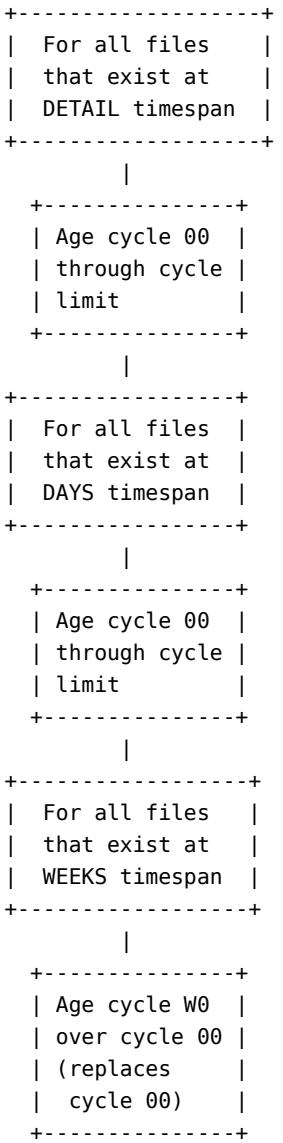
Figure 9-6. Database Timespan Update Phase (Part 2 of 2)

9.2.4 File Aging Phase

The File Aging Phase deletes the oldest cycles of the DETAIL and DAYS cycles, replacing them with the next oldest and so forth until the new interim cycle (00) becomes the current cycle (01). This phase also replaces the existing week-to-date and month-to-date cycles (00) with the updated cycles (W0 and M0, respectively).

There are no user exits invoked in this phase.

This process is illustrated in Figure 9-7.



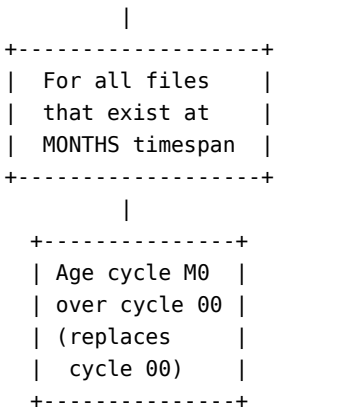


Figure 9-7. File Aging Phase

9.2.5 Operational Reports

The System-Up Report is produced by step DAY035 during the Input Raw Data Phase (see Section 9.2.1 for a description of the processing flow). POWER Account Data type S records, which are not otherwise used by CA MICS, are read to create this report. The type S records are produced at the startup completion of VSE/POWER. They record the environmental conditions in effect for that copy of POWER.

The System-Up Report is a tabular listing of the elements provided in a each record. Figure 9-8 contains an example of this report.

```
                                CA MICS VSE POWER ACCOUNTING COMPONENT - DAY035 FORMAT ROUTINE

VSE/POWER SYSTEM-UP FOR ORGSYSID VSEA VSESYSID 8 COMPLETED AT 04JAN88:08:57:38.00
  VERSION / MODIFICATION FROM RECORD....03.0                POWER PARTITION.....................F6
  POWER LEVEL IDENTIFIER FROM RECORD....G85                 PARTITION SIZE.....................712K
  VSE VERSION FROM OPTIONS STATEMENT....2.1.0               NUMBER OF DATA FILE EXTENTS..........2
  FEATURE FLAGS.....................00000000              TRACKS / BLOCKS IN DATA FILE.........60
  GETVIS SIZE.........................100K                 TRACKS / BLOCKS IN QUEUE FILE.........5
  REAL STORAGE SIZE...................128K                 TRACKS / BLOCKS IN ACCOUNT FILE.......30

VSE/POWER SYSTEM-UP FOR ORGSYSID VSE2 VSESYSID 2 COMPLETED AT 04JAN88:05:55:40.00
  VERSION / MODIFICATION FROM RECORD....03.0                POWER PARTITION.....................F1
  POWER LEVEL IDENTIFIER FROM RECORD....G85                 PARTITION SIZE.....................512K
  VSE VERSION FROM OPTIONS STATEMENT....2.1.0               NUMBER OF DATA FILE EXTENTS..........1
  FEATURE FLAGS.....................00000000              TRACKS / BLOCKS IN DATA FILE.........28855
  GETVIS SIZE.........................48K                  TRACKS / BLOCKS IN QUEUE FILE.........3072
  REAL STORAGE SIZE...................128K                 TRACKS / BLOCKS IN ACCOUNT FILE.......6200

VSE/POWER SYSTEM-UP FOR ORGSYSID RMTA VSESYSID   COMPLETED AT 05JAN88:15:04:40.00
  VERSION / MODIFICATION FROM RECORD....V2R2                POWER PARTITION.....................F1
  POWER LEVEL IDENTIFIER FROM RECORD....A45                 PARTITION SIZE.....................768K
  VSE VERSION FROM OPTIONS STATEMENT....2.2.0               NUMBER OF DATA FILE EXTENTS..........1
  FEATURE FLAGS.....................00000000              TRACKS / BLOCKS IN DATA FILE.........30
  GETVIS SIZE.........................256K                 TRACKS / BLOCKS IN QUEUE FILE.........30
  REAL STORAGE SIZE...................64K                  TRACKS / BLOCKS IN ACCOUNT FILE.......30

VSE/POWER SYSTEM-UP FOR ORGSYSID RMTA VSESYSID   COMPLETED AT 05JAN88:15:41:04.00
  VERSION / MODIFICATION FROM RECORD....V2R3                POWER PARTITION.....................F1
  POWER LEVEL IDENTIFIER FROM RECORD....A92                 PARTITION SIZE.....................768K
  VSE VERSION FROM OPTIONS STATEMENT....2.2.0               NUMBER OF DATA FILE EXTENTS..........1
  FEATURE FLAGS.....................00000000              TRACKS / BLOCKS IN DATA FILE.........30
  GETVIS SIZE.........................256K                 TRACKS / BLOCKS IN QUEUE FILE.........30
  REAL STORAGE SIZE...................64K                  TRACKS / BLOCKS IN ACCOUNT FILE.......30
```

Figure 9-8. Systems-up Report

The following elements are reported:

ORGSYSID:	The system identification for VSE/POWER.
Completed Time Stamp:	The date and time when startup was complete.
Record Version:	The version and modification level from the record. This field can contain either the Account Data record version in the form xx.y or the VSE/POWER product version in the form VxRz, where x is the version, y is the modification level, and z is the release. The format used depends upon the VSE/POWER version.
POWER Level:	VSE/POWER product level.
PWROPS Version:	The POWER version associated with this ORGSYSID in the OPTIONS statement of the PWROPS member.
Feature Flags:	Binary flags indicating which product features are active.
GETVIS Size:	The maximum storage available for GETVIS allocations in the VSE/POWER partition.
Real Size:	Storage The maximum processor storage available to VSE/POWER for fixing pages.
POWER Partition:	The partition used by VSE/POWER.
Partition Size:	The total partition size for VSE/POWER.
Data File Extents:	The number of physical disk extents allocated to the POWER Data File.
Data File Size:	The size of the Data File in tracks or DBLKs.
Queue File Size:	The size of the Queue File in tracks or DBLKs.

Account
File Size: The size of the Account File in tracks or DBLKs.

This report is useful to the VSE systems programming staff as an audit trail for changes to the VSE/POWER environment. It can also be used by the CA MICS Administrator to track the VSE/POWER version which is being used.

In a non-shared spool environment the type S record provides the only possible validation of the VSE/POWER version by CA MICS. In this case, a discrepancy between the record version and the PWROPS version causes CA MICS to issue message PWR1127E and ABEND with a User 998. The last entry in the System-Up Report lists the record which caused the ABEND.

The sample report in Figure 9-8 includes an example of this problem. The last two entries are for ORGSYSID RMTA. When POWER was initialized at 15:04:40, it was executing VSE/POWER Version 2, Release 2, which agrees with the version provided in the PWROPS member (2.2.0). However, POWER was restarted at 15:41:04 with Version 2, Release 3. In order to avoid corrupting the database, CA MICS immediately ABENDed.

Chapter 10: MODIFICATION

At times, additional system customizing is needed to tailor CA MICS to your data center. There are many user exit points provided within CA MICS. Each of these points corresponds to a position in the CA MICS logic where user modification:

- Is likely to occur, based on our experience
- Presents only a slight possibility of compromising CA MICS Database or operational integrity
- 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, we consider that the CA MICS System Administrator has four levels or 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 data center-dependent requirements that are not adequately addressed through the standard options and parameters. In these cases one or more of the user exits provided with CA MICS may be used to insert user-written routines that satisfy the requirements.

o Extension through User-Written Components

  Although most users do not, at first, consider this a way of modifying the CA MICS product environment, the development of user-written products may be the most advantageous way of addressing the need to modify a standard CA MICS product. For example, many times it may be better to write a user product to handle additional SMF data than to attempt to modify the logic of the SMF Component.

o System Code Modification

  Finally, if all else fails, you may be able to satisfy your data center's requirements by implementing a source code change.

This chapter explains the user exits that are provided with the CA MICS Analyzer Option for VSE/POWER product. You must carefully explore whether it is necessary to develop a user exit routine. First ensure that the requirement cannot be satisfied through one of the standard options or definitions by discussing your requirement with CA Technical Support.

BEFORE ATTEMPTING TO ACTIVATE AN EXIT, you should read and fully understand the information contained in Section 4.3, User Exit Facilities, of the CA MICS System Modification Guide. The following material is intended to supplement, not replace, that discussion.

The design, coding, testing, and implementation of CA MICS user exit routines should be approached with caution as errors may result, causing the corruption of the data. You should be extremely careful in the definition and validation of the exit routines to ensure that the system integrity and performance has not been adversely affected.

The following sections identify the user exits available for VSE/POWER product processing, provide a general overview of product processing logic, and describe in detail each of the user exits.

1 - Standard User Exits
2 - File and Data Element Customization

10.1 Standard User Exits

There are two types of user exits that can be used to modify CA MICS logic: the general exits, which apply to all the installed products, and the product exits, which apply only to the product for which they are provided.

The general exits are briefly described in this guide. They are described in more detail in Section 4.3, User Exit Facilities, in the CA MICS System Modification Guide.

The product exits are classified according to their purpose and include product input, output, parameter-related, and accounting exits.

Input exits are invoked as the raw input records are read. Output exits are invoked just before CA MICS writes an observation to the CA MICS database.

The CA MICS Analyzer Option for VSE/POWER product provides the general input exit, _USRSEL, which you may use to select or exclude data meeting special installation-defined criteria, to add data elements, or to modify record fields as they are read.

VSE/POWER includes one input exit routine for each file, except PWRPRA. The exit is invoked in the input format processing phase, and may be used to modify elements before they are passed to the information area processing phase. Since no additional processing occurs for PWRPRA in the IA phase, the file output exit is used instead.

One output exit routine is provided for each product file and may be used to selectively block the writing of specified records, alter data elements in the records, or produce additional records for the target file or a user-defined file.

The product parameter-related exits are also user exits, but are defined as part of the standard CA MICS installation process. They are normally associated with other parameters or options. For example, the Account Code Derivation Routine is a user exit routine for completing values for account codes that were defined in a separate but related member of MICS.PARMS, PWRACRT, the Account Code Definition parameter member. These exits are identified in this chapter but are described in Chapter 7, Defining VSE/POWER Parameters.

The user exits for the Analyzer Option for VSE/POWER are

```
identified below and discussed in the sections noted:

    1 - General Exits

        _USRSEL  - Input record selection

    2 - Input Exits

        USRDPDA - PWRPDA element derivation
        USRDPPG - PWRPPG element derivation
        USRDPOA - PWRPOA element derivation
        USRDPJB - PWR_PR element derivation

    3 - Parameter-Related Exits

        PWRACRT  - VSE/POWER account code derivation
        PWRGPRTE - Job group derivation
        PWRTNRTE - Job turnaround time derivation

    4 - Output Exits

        USRSPDA - PWRPDA record selection
        USRSPPG - PWRPPG record selection
        USRSPOA - PWRPOA record selection
        USRSPRA - PWRPRA record selection
        USRSPJB - PWRPJB record selection
```

10.1.1 General Exits

This section provides a description of the general exits that are shared by VSE/POWER with the other installed CA MICS products.

Each exit description includes the name and title, a description of its purpose, when it is invoked, and whether or not it has an interface to the CA MICS Accounting and Chargeback product. It also shows which data elements are available, the special considerations to note, and a sample user exit.

Input Record Selection Exit (_USRSEL)

```
+-------------+
| U S R S E L |  Input Record Selection Exit
+-------------+
```

DESCRIPTION: The exit provides the ability to select records that may be processed by the VSE/POWER input format routine. It is similarly invoked to provide data selection for each CA MICS component.

INVOCATION: The exit gains control during the input format phase of the POWER daily update process step (DAY035). It is invoked after the account data SYSID header and other common fields have been read, but before the front-end record processing. See Chapter 9 of this guide for more information.

ACCOUNTING INTERFACE: No interface is provided.

USES: This exit may be used to exclude records from processing based on their source, type, or origin.

ELEMENTS AVAILABLE: This exit is referenced by many different CA MICS products. To help identify the caller, the variable ROUTINE is initialized with the module name. For POWER, it will contain DYPWRFMT.

The element SKIP_REC is initialized to zero just prior to the exit. If the exit code sets this to one, that record will be bypassed.

The following POWER elements are also available:

VSEDDNAM - Input DD from which the record was read
PWRRTYPE - Record type
VSESYSID - VSE/POWER system identifier

```
  STARTTS  - Event start timestamp
  ENDTS    - Event end timestamp
  VSEDATE  - Event date from the ACDATE field
  STARTTM  - Event start time of day from the ACSTRT field
  ENDTM    - Event end time of day from the ACSTOP field

CODING RESTRICTIONS:  See the System Modification Guide,
Section 4.3.2.1.

SPECIAL NOTES:

SAMPLE USER EXIT:

  DESCRIPTION:  DOS Job Accounting data can have limited
  compatibility with VSE/POWER Accounting data (see Chapter 2
  of this guide for more information).  For this example
  only, the execution account record is compatible.  All
  other records from the DOS account data file will be
  skipped.

  USER EXIT EXAMPLE:

    MACRO _USRSEL
      IF ROUTINE EQ 'DYPWRFMT' THEN DO ;
        IF VSEDDNAM EQ 'DOSACCT'
         AND PWRRTYPE NE 'E'
         THEN SKIP_REC=1 ;
      END ;
    %
```

10.1.2 Input Exits

This section presents a description of the input exits that are provided in the input format processing routine of the CA MICS Analyzer Option for VSE/POWER product. The exits are listed in alphabetical order.

Each exit description includes the name and title, a description of its purpose, when it is invoked, and whether or not it has an interface to the CA MICS Accounting and Chargeback component. It also shows which data elements are available, the special considerations to note, and a sample user exit.

PWRPDA Element Derivation Exit (USRDPDA)

```
+---------------+
| U S R D P D A |  PWRPDA Element Derivation Exit (USRDPDA)
+---------------+
```

DESCRIPTION: The exit provides control over the observations added to the working version of the PWRPDA File during input format processing.

INVOCATION: The exit is invoked in module DYPWRFMT during the processing of the SIO table in the Execution Account Record. See Chapter 9 of this guide for more information.

ACCOUNTING INTERFACE: No interface is provided.

USES: File elements can be modified before they undergo any summarization processing in the Information Area phase of DYPWRFMT. The exit can also be used to prevent selected observations from being added.

ELEMENTS AVAILABLE: Most elements available in the DETAIL timespan of the PWRPPG File and all elements of the PWRPDA File.

CODING RESTRICTIONS: See the CA MICS System Modification Guide, Section 4.3.2.1.

SPECIAL NOTES: This is an indirect macro language exit (see Section 4.3.2.1 of the CA MICS System Modification Guide). The member assignment statement is provided in prefix.MICS.USER.SOURCE($PWREXIT).

SAMPLE USER EXIT:

DESCRIPTION: In this exit, the user wants to save space by

```
  retaining only DASD and tape activity in the PWRPDA File.

  USER EXIT EXAMPLE:

    Member assignment:

      %LET USRDPDA = MYEXIT1 ;

    Exit code in prefix.MICS.USER.SOURCE(MYEXIT1):

      IF DEVCLASS NE:  'DASD' AND DEVCLASS NE 'TAPE'
       THEN SKIP_REC = 1 ;


PWR_PR Element Derivation Exit (USRD_PR)
+---------------+
| U S R D P J B |  PWR_PR Element Derivation Exit (USRD_PR)
+---------------+

DESCRIPTION:  The exit provides control over the observations
added to the working version of the PWR_PR File during input
format processing.  The PWR_PR File contains data from the
Reader Account records that will later be combined with the
Program Activity File to construct the Job Activity File
(PWRPJB).

INVOCATION:  The exit is invoked in module DYPWRFMT prior to
adding the observation to the PWR_PR File.  See Chapter 9 of
this guide for more information.

ACCOUNTING INTERFACE:  No interface is provided.

USES:  File elements can be modified before they undergo any
summarization processing in the Information Area phase of
DYPWRFMT.  The exit can also be used to prevent selected
observations from being added.

ELEMENTS AVAILABLE:  Most elements available in the DETAIL
timespan of the PWR_PR File.

CODING RESTRICTIONS: See the CA MICS System Modification
Guide, Section 4.3.2.1.

SPECIAL NOTES:  This is an indirect macro language exit (see
Section 4.3.2.1 of the CA MICS System Modification Guide).
The member assignment statement is provided in
prefix.MICS.USER.SOURCE($PWREXIT).

SAMPLE USER EXIT:
```

```
  DESCRIPTION:  In this example the user does not want to
  retain jobs being routed to a particular remote system.

  USER EXIT EXAMPLE:

    Member assignment:

      %LET USRDPJB = MYEXIT1 ;

    Exit code in prefix.MICS.USER.SOURCE(MYEXIT1):

      IF PJBDRMT EQ 5
       THEN SKIP_REC = 1 ;

PWRPOA Element Derivation Exit (USRDPOA)
+---------------+
| U S R D P O A |  PWRPOA Element Derivation Exit (USRDPOA)
+---------------+

DESCRIPTION:  The exit provides control over the observations
added to the working version of the PWRPOA File during input
format processing.

INVOCATION:  The exit is invoked in module DYPWRFMT prior to
adding the observation to the PWRPOA File.  See Chapter 9 of
this guide for more information.

ACCOUNTING INTERFACE:  No interface is provided.

USES:  File elements can be modified before they undergo any
summarization processing in the Information Area phase of
DYPWRFMT.  The exit can also be used to prevent selected
observations from being added.

ELEMENTS AVAILABLE:  Most elements available in the DETAIL
timespan of the PWRPOA File.

CODING RESTRICTIONS:  See the CA MICS System Modification
Guide, Section 4.3.2.1.

SPECIAL NOTES:  This is an indirect macro language exit (see
Section 4.3.2.1 of the CA MICS System Modification Guide).
The member assignment statement is provided in
prefix.MICS.USER.SOURCE($PWREXIT).

SAMPLE USER EXIT:

  DESCRIPTION:  In this example the user knows that output to
```

```
  form number F is routed to a device that automatically
  produces two copies of the output.

  USER EXIT EXAMPLE:

    Member assignment:

      %LET USRDPOA = MYEXIT1 ;

    Exit code in prefix.MICS.USER.SOURCE(MYEXIT1):

      IF FORMNUM EQ 'F' THEN DO ;
        POANLR = POANLR*2 ;
        POAENLR = POAENLR*2 ;
      END ;
```

PWRPPG Element Derivation Exit (USRDPPG)

```
+---------------+
| U S R D P P G |  PWRPPG Element Derivation Exit (USRDPPG)
+---------------+
```

DESCRIPTION: The exit provides control over the observations added to the working version of the PWRPPG File during input format processing.

INVOCATION: The exit is invoked in module DYPWRFMT prior to adding the observation to the PWRPPG File. See Chapter 9 of this guide for more information.

ACCOUNTING INTERFACE: No interface is provided.

USES: File elements can be modified before they undergo any summarization processing in the Information Area phase of DYPWRFMT. The exit can also be used to prevent selected observations from being added.

ELEMENTS AVAILABLE: Most elements available in the DETAIL timespan of the PWRPPG File.

CODING RESTRICTIONS: See the CA MICS System Modification Guide, Section 4.3.2.1.

SPECIAL NOTES: This is an indirect macro language exit (see Section 4.3.2.1 of the CA MICS System Modification Guide). The member assignment statement is provided in prefix.MICS.USER.SOURCE($PWREXIT).

SAMPLE USER EXIT:

```
DESCRIPTION:  In this case the user has added an additional
measurement field to the user accounting portion of the
Execution Account record.  It will be added to the PWRPPG
File, and, through step summarization, to the PWRPJB File.

USER EXIT EXAMPLE:

  Member assignment:

    %LET USRDPPG = MYEXIT1 ;

  Exit code in prefix.MICS.USER.SOURCE(MYEXIT1):

    IF PPGUFLDL GE 8 THEN
     PPGMEASR = INPUT(SUBSTR(PPGUFLD1,5,4),PIB4.) ;
    ELSE PPGMEASR = .  ;
```

10.1.3 Parameter-Related Exits

```
The following exits are defined during system generation.
They are invoked by direct %INCLUDE of the member containing
the code.

Chapter 7 of this guide describes each member.

VSE/POWER Account Code Derivation (PWRACRT)
+---------------+
| P W R A C R T |  VSE/POWER Account Code Derivation
+---------------+

DESCRIPTION:  Derives the value of the account code elements
described in prefix.MICS.PARMS(PWRACCT).

INVOCATION:  The exit is referenced once for each of the
following files:  PWRPPG, PWRPOA, and PWRPJB.

ACCOUNTING INTERFACE:  No interface is provided.

USES:  This exit sets the value of the PWRACTx variables.

ELEMENTS AVAILABLE:  The PWRACTEX variable can be tested to
determine the exit for which the file was invoked.  It will
contain the three-character file identifier.  All detail
timespan elements of the file are available.

CODING RESTRICTIONS:  Refer to Section 7.2.2 of this guide.

SPECIAL NOTES:       The exit code is contained in
prefix.MICS.PARMS(PWRACRT).

SAMPLE USER EXIT:  See Figure 7-2 in Section 7.2.2. for a
sample exit.


VSE/POWER Job Group Derivation (PWRGPRTE)
+-----------------+
| P W R G P R T E |    VSE/POWER Job Group Derivation
+-----------------+

DESCRIPTION:  Derives the value of the JOBGROUP variable of
the PWRPJB File.

INVOCATION:  The exit is invoked during the step
summarization processing, just before the PWRPJB observation
is added to the file.  See Chapter 9 of this guide for more
information.
```

```
ACCOUNTING INTERFACE:  No interface is provided.

USES:  This exit sets the value of the JOBGROUP variable.

ELEMENTS AVAILABLE:  See Chapter 7 of this guide for more
information.

CODING RESTRICTIONS:  Refer to Section 7.3.3 of this guide.

SPECIAL NOTES:      The exit code is contained in
prefix.MICS.PARMS(PWRGPRTE).

SAMPLE USER EXIT:  See Section 7.3.3 for a sample exit.


VSE/POWER Job Turnaround Time Derivation (PWRTNRTE)
+-----------------+
| P W R T N R T E |  VSE/POWER Job Turnaround Time Derivation
+-----------------+

DESCRIPTION:  Overrides the standard turnaround and deadline
event calculations.

INVOCATION:  The exit is referenced during step summarization
prior to adding the new observation to the PWRPJB File.  See
Chapter 9 of this guide for more information.

ACCOUNTING INTERFACE:  No interface is provided.

USES:  The exit is used to override the default calculations
of job turnaround time and the deadline event timestamps
prior to assessment of service quality.

ELEMENTS AVAILABLE:  All elements of the PWRPJB File DETAIL
timespan are available.  See Section 7.3.4 of this guide for
more information.

CODING RESTRICTIONS:  Refer to Section 7.3.4 of this guide.

SPECIAL NOTES:      The exit code is contained in
prefix.MICS.PARMS(PWRTNRTE).

SAMPLE USER EXIT:  See Chapter 7 for a sample exit.
```

10.1.4 Output Exits

Output exits are invoked just prior to adding the observation to the DETAIL timespan (or DAYS for PWRPDA) of the file. They can be used to modify the value of elements or prevent selected observations from being added.

The following exit description applies to these files:

PWRPJB - Job Activity File
PWRPPG - Program Activity File
PWRPOA - Output Activity File
PWRPRA - Remote Activity File
PWRPDA - Device Activity File

POWER File Selection Exit (USRSfff)

```
+---------------+
| U S R S f f f |  POWER File Selection Exit (USRSfff)
+---------------+
```

DESCRIPTION: The exit provides the ability to modify or select observations immediately prior to output.

INVOCATION: These exits are invoked in the information area processing phase of the DYPWRFMT routine immediately before output of the file. See Chapter 9 of this guide for more information.

ACCOUNTING INTERFACE: No interface is provided.

USES: Allows elements to be modified and observations to be selectively excluded from processing.

ELEMENTS AVAILABLE: All elements in the file are available.

CODING RESTRICTIONS: Refer to the CA MICS System Modification Guide, Section 4.3.2.1.

SPECIAL NOTES: This is an indirect macro language exit (see Section 4.3.2.1 of the CA MICS System Modification Guide). The member assignment statement is provided in prefix.MICS.USER.SOURCE($PWREXIT).

SAMPLE USER EXIT:

DESCRIPTION: In this example the user wants to delete SYSID VSE1 only from the PWRPPG File.

```
USER EXIT EXAMPLE:

  Member assignment:

    %LET USRSPPG = MYEXIT1 ;

  Exit code in prefix.MICS.USER.SOURCE(MYEXIT1):

    IF SYSID EQ 'VSE1' THEN SKIP_REC ;
```

10.2 File and Data Element Customization

See Section 4.2 of the CA MICS System Modification Guide for information on file and data element customization.

Appendix A: MESSAGES

Messages

The MICSLOG program converts message codes into text and places the message identifier and text into the MICSLOG data set. MICSLOG messages are printed after job output, enabling you to relate the messages to job steps.

The format of a MICSLOG message is

```
    hh:mm:ss.hs cccnnnnns ttttttttttttttttttttttttttttttttttt
```

where:

hh:mm:ss.hs = the time that the message was issued.

cccnnnnns = the message identifier. The message identifier consists of three parts:

- ccc = three-letter component identifier
- nnnnn = the message number
- s = the severity identifier, where I is an informational message, W indicates a warning, and E indicates an error.

ttt...ttt = the message text. If the text spans more than one line, the message identifier is repeated.

Both the SAS log and the MICSLOG contain information about jobs that were processed. The MICSLOG program writes messages to both the MICSLOG data set and to the SAS log because CA MICS runs under SAS. When reviewing the output of a job, begin by looking at the MICSLOG data set because it contains fewer lines of information, making it easier to see whether or not a job or job step completed as expected. If the MICSLOG data set indicates a problem, consult the SAS log to help debug the problem if the information in the MICSLOG data set does not lead you to the source of the problem.

The remainder of this appendix shows the MICSLOG messages produced for the CA MICS Analyzer Option for VSE/POWER product. In addition to the message identifier and text, we present the reason for the message, suggest actions to take to resolve the problem that most likely caused the message to be identified, and refer you to sources for additional information to help you investigate the problem.

SHORT RECORD ISSUES
(Messages PWR01100 - PWR01124)

CA MICS may find short records in the following circumstances:
 If the proportion of short records is large (more than 5 percent), the problem can be caused by an error in the data specifications in prefix.MICS.PARMS(PWROPS). This type of error will also result in 'data invalid' messages appearing on the SAS log. Review the OPTIONS parameter and your assumptions about the data.

Short records may be due to an error in the monitor itself. Carefully review the occurrences of the error for any pattern. Do all the errors occur in a single record type? Do they occur only in related records (i.e., Transmitter and Receiver records only)? If possible, review the structure of the records echoed on the SAS log. Are there any fields missing? If any of these are true, then the problem is most likely in the monitor. Have your DOS systems programmer search the INFO/VM-VSE database for reported problems. Likely keywords for the search include the field name in error or the record type combined with one or more of the following: INCORROUT, VSE, POWER, and ACCOUNTING.

If the error occurs only in a small number of records, it may be due to a non-repeatable problem such as a system failure or disk error. However, if the condition continues to occur over multiple DAILY updates, you should investigate it further. Only you can say what level of data errors constitute a significant problem for your environment.

```
+-----------------+
| P W R 0 0 0 0 1 |
+-----------------+
```

TEXT: VSE/POWER PARAMETER GENERATION STARTED
 TYPE: Information

```
   REASON:  Acknowledges that the parameter generation
            process has started.

   ACTION:  None necessary.

   REFERENCES:  PWRPGEN

+-----------------+
| P W R 0 0 0 0 2 |
+-----------------+

   TEXT:  LENGTH OF ORGSYSID MUST BE 4 OR LESS. ORGSYSID
            FOUND = %TOKEN1

   TYPE:  Error
   REASON:  The ORGSYSID coded on the OPTIONS statement
            exceeds four characters.

   ACTION:  Correct the ORGSYSID on the OPTIONS statement of
            prefix.MICS.PARMS(PWROPS)

   REFERENCES:  PWRPGEN

+-----------------+
| P W R 0 0 0 0 3 |
+-----------------+

   TEXT:  DEVICE ADDRESS %ADD1C IS NOT A VALID ADDRESS

   TYPE:  Error

   REASON:  The address contains invalid hex characters.
    ACTION:  Correct the address.

   REFERENCES:  PWRPGEN

+-----------------+
| P W R 0 0 0 0 4 |
+-----------------+

   TEXT:  DEVICE ADDRESS 2 MUST BE GT ADDRESS 1. ADDRESS
            RANGE FOUND = %TOKEN2

   TYPE:  Error

   REASON:  Address values in a range specification must be
            in ascending sequence.  For example, 140-14F is
            valid, but 14F-140 is not.
    ACTION:  Correct the value coded for device address in
```

```
            prefix.MICS.PARMS(PWRCONFG).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 0 5 |
+-----------------+

   TEXT:  DEVICE ADDRESS RANGE DOES NOT CONTAIN VALID
          HEXADECIMAL ADDRESS.  ADDRESS RANGE FOUND = %TOKEN2

   TYPE:  Error

   REASON:  One or both of the addresses specified contain
            invalid hexadecimal characters.
    ACTION:  Correct the value coded for device addresses in
            prefix.MICS.PARMS(PWRCONFG).

   REFERENCES:  PWRPGEN

+-----------------+
| P W R 0 0 0 0 6 |
+-----------------+

   TEXT:  DEVICE ADDRESS NOT SPECIFIED

   TYPE:  Error

   REASON:  The device address in prefix.MICS.PARMS(PWRCONFG)
            is required.

   ACTION:  Code one or more device addresses in the PWRCONFG
            member of prefix.MICS.PARMS.

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 0 7 |
+-----------------+

   TEXT:  FORMAT OF DEVICE ADDRESS NOT RECOGNIZED. DEVICE
            ADDRESS FOUND = %TOKEN2

   TYPE:  Error

   REASON:  The device address specification must be in the
            form xxx or xxx-xxx.
```

ACTION: Correct the value for device address coded in prefix.MICS.PARMS(PWRCONFG).

REFERENCES: PWRPGEN

+----------------+
| P W R 0 0 0 0 8 |
+----------------+

TEXT: ADDRESS RANGES MUST HAVE SAME FIRST 2 CHARACTERS. DEVICE ADDRESS FOUND = %TOKEN2

TYPE: Error

REASON: The address range form can be used to describe a maximum of 16 contiguous addresses.

ACTION: Correct the value for device address coded in prefix.MICS.PARMS(PWRCONFG).

REFERENCES: PWRPGEN

+----------------+
| P W R 0 0 0 0 9 |
+----------------+

TEXT: DEVICE CLASS NOT SPECIFIED

TYPE: Error

REASON: The device class in prefix.MICS.PARMS(PWRCONFG) is required.

ACTION: Code one or more device classes in the PWRCONFG member of prefix.MICS.PARMS.

REFERENCES: PWRPGEN

+----------------+
| P W R 0 0 0 1 0 |
+----------------+

TEXT: DEVICE CLASS %DEVCLASS IS UNKNOWN

TYPE: Warning

REASON: The CA MICS Analyzer Option for VSE/POWER product

accumulates SIO counts based on a list of recognized device classes. The PWR00012 message is issued when you code an unrecognized value for device class.

ACTION: Ensure that the device class coded in prefix.MICS.PARMS(PWRCONFG) meets your reporting requirements for SIO information.

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 1 1 |
+-----------------+
```

TEXT: LENGTH OF DEVICE CLASS MUST BE 8 OR LESS. DEVICE CLASS FOUND = %TOKEN3

TYPE: Error

REASON: The value of device class is limited to eight or fewer characteristics.

ACTION: Correct the value of device class coded in prefix.MICS.PARMS(PWRCONFG).

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 1 2 |
+-----------------+
```

TEXT: DEVICE TYPE NOT SPECIFIED

TYPE: Error

REASON: The device type in prefix.MICS.PARMS(PWRCONFG) is required.

ACTION: Code one or more device types in the PWRCONFG member of prefix.MICS.PARMS.

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 1 3 |
+-----------------+
```

TEXT: DEVICE TYPE %DEVTYPE IS UNKNOWN

TYPE: Warning

REASON: The CA MICS Analyzer Option for VSE/POWER product is distributed with a list of device types that it uses to validate the VSE/POWER configuration at your installation. The validation process controls how SIO counts are accumulated in the CA MICS database. The PWR00013 message is issued when you code a value for device type that is not in the list.

ACTION: Ensure that the value for device type coded in prefix.MICS.PARMS(PWRCONFG) enables you to meet your reporting requirements.

REFERENCES: PWRPGEN

| P W R 0 0 0 1 4 |

TEXT: DEVICE TYPE %DEVTYPE IS NOT KNOWN IN CLASS %DEVCLASS

TYPE: Warning

REASON: The CA MICS Analyzer Option for VSE/POWER product is distributed with a set of device types and classes that it uses to control how SIO counts are accumulated. The PWR00014 message is issued when a device type and device class pair is not in the list.

ACTION: Ensure that your reporting needs for SIO count information can be met based on the device type and device class pairs coded in prefix.MICS.PARMS(PWRCONFG).

REFERENCES: PWRPGEN

| P W R 0 0 0 1 5 |

TEXT: LENGTH OF DEVICE TYPE MUST BE 8 OR LESS. DEVICE TYPE FOUND = %TOKEN4

```
   TYPE:  Error

   REASON:  The value of device type is limited to eight or
            fewer characteristics.

   ACTION:  Correct the value of device type coded in
            prefix.MICS.PARMS(PWRCONFG).

   REFERENCES:  PWRPGEN
+-----------------+
| P W R 0 0 0 1 6 |
+-----------------+

   TEXT:  NO CONFIGURATION HAS BEEN SPECIFIED. PJB ELEMENTS
            MAY BE INVALID

   TYPE:  Warning

   REASON:  No configuration information was provided. All
            SIO counts will be assigned to a device class and
            type of *UNKNOWN.  Also, device class SIO counts
            will be accumulated to element fffOSIO ('other').

   ACTION:  Define your system configuration in
            prefix.MICS.PARMS(PWRCONFG) if necessary to meet
            your measurement needs.
    REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 1 7 |
+-----------------+

   TEXT:  FATAL ERRORS ENCOUNTERED IN PWRCONFG MEMBER

   TYPE:  Error

   REASON:  The parameter specifications and/or validations
            performed for information coded in PWRCONFG did
            not complete satisfactorily.

   ACTION:  Review the error messages printed in MICSLOG
            prior to this message.  These earlier messages
            point you to the areas that you need to research
            to resolve the fatal errors.

   REFERENCES:  PWRPGEN
```

```
+-----------------+
| P W R 0 0 0 1 8 |
+-----------------+
```

TEXT: LENGTH OF ORGSYSID MUST BE 4 OR LESS. ORGSYSID FOUND = %TOKEN2

TYPE: Error

REASON: Valid values for the original system identifier field of prefix.MICS.PARMS(PWROPS) are one to four characters long.

ACTION: Correct the value coded for ORGSYSID in prefix.MICS.PARMS(PWROPS).

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 1 9 |
+-----------------+
```

TEXT: LENGTH OF DDNAME MUST BE 8 OR LESS. DDNAME FOUND = %TOKEN3

TYPE: Error

REASON: Valid values for the DDNAME field of the PWROPS member of prefix.MICS.PARMS are one to eight characters long.

ACTION: Correct the value of the DDNAME field of prefix.MICS.PARMS(PWROPS).

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 2 0 |
+-----------------+
```

TEXT: VSE SYSID MUST BE A NUMERIC VALUE OR X. VALUE FOUND = %VSESYSID

TYPE: Error

REASON: Valid values for the field VSESYSID in the PWROPS member of prefix.MICS.PARMS are the single digits one through nine or the character X.

```
   ACTION:  Correct the value coded for VSESYSID in
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 2 1 |
+-----------------+

   TEXT:  LENGTH OF VSE SYSID MUST BE 1. VSE SYSID FOUND =
            %TOKEN4

   TYPE:  Error

   REASON:  Valid values for the field VSESYSID in the PWROPS
            member of prefix.MICS.PARMS are the single digits
            one through nine or the character X.

   ACTION:  Correct the value coded for VSESYSID in
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 2 2 |
+-----------------+

   TEXT:  DATE FORMAT VALUE NOT RECOGNIZED; IT MUST BE
            DD/MM/YY OR MM/DD/YY.
          DATE FORMAT FOUND = %TOKEN5

   TYPE:  Error

   REASON:  Valid values for the DATEFMT field of the PWROPS
            member of prefix.MICS.PARMS are DD/MM/YY and
            MM/DD/YY.

   ACTION:  Correct the value coded for the DATEFMT field in
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 2 3 |
+-----------------+
```

```
   TEXT:  LENGTH OF DATE FORMAT MUST BE 8. DATE FORMAT
            FOUND = %TOKEN5
    TYPE:  Error

   REASON:  Valid values for the DATEFMT field of the PWROPS
            member of prefix.MICS.PARMS are DD/MM/YY and
            MM/DD/YY.

   ACTION:  Correct the value coded for the DATEFMT field in
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 2 4 |
+-----------------+

   TEXT:  INVALID SYSID OPTION SPECIFIED. OPTION FOUND =
            %TOKEN6

   TYPE:  Error

   REASON:  Valid values for the SYSID_SW field in the PWROPS
            member of prefix.MICS.PARMS are SYSID and
            NOSYSID.

   ACTION:  Correct the value of the SYSID_SW field in
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 2 5 |
+-----------------+

   TEXT:  INVALID VSE VERSION SPECIFIED:  %TOKEN7

   TYPE:  Error

   REASON:  Valid values for the VERSION field of the PWROPS
            member of prefix.MICS.PARMS are 2.1.0 and above.

   ACTION:  Correct the value coded for VERSION in
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN
```

```
+-----------------+
| P W R 0 0 0 2 6 |
+-----------------+

   TEXT:  LENGTH OF VSE VERSION MUST BE 8 OR LESS. VSE
            VERSION FOUND = %TOKEN7

   TYPE:  Error

   REASON:  Valid values for the VERSION field of the PWROPS
            member of prefix.MICS.PARMS are 2.1.0 and above.

   ACTION:  Correct the value coded for VERSION in
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 2 7 |
+-----------------+

   TEXT:  OPTION SYSID SPECIFIED AND NUMERIC VALUE FOR VSE
            SYSID WAS NOT SPECIFIED

   TYPE:  Error

   REASON:  Stating that SYSID is the valid value for the
            field SYSID_SW means that there must be SYSID
            headers provided in the data.  Specifying X as
            the value for the field VSESYSID means that there
            are not headers provided in the data.

   ACTION:  Review the specifications for the fields SYSID_SW
            and VSESYSID in prefix.MICS.PARMS(PWROPS).
            Ensure that the values coded logically support
            one another (SYSID_SW=SYSID means VSESYSID=single
            digit one through nine; SYSID_SW=NOSYSID means
            VSESYSID=X).

   REFERENCES:  PWRPGEN

+-----------------+
| P W R 0 0 0 2 8 |
+-----------------+

   TEXT:  OPTION NOSYSID SPECIFIED AND A NUMERIC VALUE FOR
```

```
            VSE SYSID HAS BEEN SPECIFIED.
          VSE SYSID WILL NOT BE USED FOR RECORD
            SELECTION.  VSE SYSID HAS BEEN SET TO X.

   TYPE:  Warning

   REASON:  Stating that X is the valid value for the field
            VSESYSID means that SYSID headers will not be
            processed from the data, even if they are present
            (as indicated by coding SYSID for the SYSID_SW
            field).
   ACTION:  Review the specifications for the fields SYSID_SW
            and VSESYSID in prefix.MICS.PARMS(PWROPS).
            Ensure that the value coded for VSESYSID reflects
            the kind of data collection you intend to perform
            (collect by SYSID or not).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 2 9 |
+-----------------+

   TEXT:  ORGSYSID %ORGSYSID IS DUPLICATELY DEFINED

   TYPE:  Error

   REASON:  Each ORGSYSID value must be uniquely defined in
            an OPTIONS statement.

   ACTION:  Review the OPTIONS statement in
            prefix.MICS.PARMS(PWROPS) for the indicated value
            of ORGSYSID.

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 3 0 |
+-----------------+

   TEXT:  DDNAME %DDNAME AND VSE SYSID %VSESYSID DUPLICATELY
            DEFINED

   TYPE:  Error
   REASON:  The combination of DDNAME and VSESYSID uniquely
            identify the data from a single VSE/POWER system.
            There should be only one OPTIONS statement for
```

```
            each system.

   ACTION:  Review the OPTIONS statements in
            prefix.MICS.PARMS(PWROPS) for the indicated
            system.

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 3 1 |
+-----------------+

   TEXT:  LIMIT OF DEFINED ORGSYSIDS HAS BEEN REACHED.
            CONTACT CA MICS PRODUCT SUPPORT GROUP
   TYPE:  Error

   REASON:  A maximum of 99 OPTIONS statements (VSE/POWER
            systems) are supported.

   ACTION:  Contact the CA MICS Product Support Group for
            assistance.

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 3 2 |
+-----------------+

   TEXT:  UNIDENTIFIED KEYWORD ENCOUNTERED. KEYWORD FOUND =
            %TOKEN1
   TYPE:  Error

   REASON:  The only keyword statement supported is
            OPTIONS.

   ACTION:  Correct the value of the first field in
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 3 3 |
+-----------------+

   TEXT:  NO VALID OPTIONS STATEMENTS HAVE BEEN FOUND
   TYPE:  Error
```

```
   REASON:  You must define at least one VSE/POWER system to
            be processed.

   ACTION:  Add one or more OPTIONS statements to
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 3 4 |
+-----------------+

   TEXT:  DDNAME %DDNAME SPECIFIED IN INPUTPWR NOT FOUND IN
            PWROPS
   TYPE:  Error

   REASON:  CA MICS generates input format code based on the
            DDname values in PWROPS.  All DDnames must be
            defined to be processed.

   ACTION:  Add the missing DDname to
            prefix.MICS.PARMS(PWROPS).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 3 5 |
+-----------------+

   TEXT:  DDNAME %DDNAME SPECIFIED IN PWROPS DOES NOT HAVE A
            CORRESPONDING INPUT STATEMENT IN INPUTPWR


   TYPE:  Error

   REASON:  CA MICS generates input format code based on the
            DDname values in PWROPS.  All DDnames must be
            defined to be processed.

   ACTION:  Add the missing DDname to
            prefix.MICS.PARMS(INPUTPWR).

   REFERENCES:  PWRPGEN


+-----------------+
```

```
| P W R 0 0 0 3 6 |
+-----------------+

   TEXT:  FATAL ERRORS ENCOUNTERED IN PWROPS/INPUTPWR

   TYPE:  Error

   REASON:  The CA MICS Analyzer Option for VSE/POWER product
            validates the DDnames in
            prefix.MICS.PARMS(INPUTPWR) against the values in
            the DDname field of prefix.MICS.PARMS(PWROPS).
            There must be an exact match between the values
            in both members because INPUTPWR tells the CA
            MICS product the location of the data set that
            contains the data for the system specified by the
            PWROPS statement.

   ACTION:  Review the specifications for the DDname field in
            prefix.MICS.PARMS(PWROPS) and the DDnames in
            prefix.MICS.PARMS(INPUTPWR).  Correct the members
            so that these specifications match.
    REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 3 7 |
+-----------------+

   TEXT:  KEYWORD NOT RECOGNIZED. KEYWORD FOUND = %TOKEN3

   TYPE:  Error

   REASON:  Valid values for the time descriptor associated
            with prefix.MICS.PARMS(PWRGROUP) field turntime
            are LOCAL and REMOTE.

   ACTION:  Correct the time descriptor.

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 3 8 |
+-----------------+

   TEXT:  VALID TURNAROUND TIME DATA ELEMENTS NOT SPECIFIED

   TYPE:  Error
```

REASON: At least one valid (numeric 1-3) turnaround time component must be specified.

ACTION: Correct the TURNTIME statement in prefix.MICS.PARMS(PWRGROUP).

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 3 9 |
+-----------------+
```

TEXT: TURNAROUND TIME ELEMENTS NOT NUMERIC

TYPE: Error

REASON: The turnaround time element identifiers must be numeric in the range 1-3.

ACTION: Correct the TURNTIME statement in prefix.MICS.PARMS(PWRGROUP).

REFERENCES: PWRPGEN

```
 +-----------------+
| P W R 0 0 0 4 0 |
+-----------------+
```

TEXT: STANDARD TURNAROUND VALUES MUST BE 1, 2, OR 3

TYPE: Error

REASON: The turnaround time element identifiers must be numeric in the range 1-3.

ACTION: Correct the TURNTIME statement in prefix.MICS.PARMS(PWRGROUP).

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 4 1 |
+-----------------+
```

TEXT: STANDARD TURNAROUND VALUES HAVE BEEN REPEATED

TYPE: Error

REASON: Each turnaround element can be used only once.

ACTION: Correct the TURNTIME statement in prefix.MICS.PARMS(PWRGROUP).

REFERENCES: PWRPGEN

+----------------+
| P W R 0 0 0 4 2 |
+----------------+

TEXT: DUPLICATE STANDARD LOCAL DEFINITIONS

TYPE: Error

REASON: The turnaround time definition for standard job groups with local jobs has been defined on more than one TURNTIME statement.

ACTION: Correct the TURNTIME statement in prefix.MICS.PARMS(PWRGROUP).

REFERENCES: PWRPGEN

+----------------+
| P W R 0 0 0 4 3 |
+----------------+

TEXT: DUPLICATE STANDARD REMOTE DEFINITIONS

TYPE: Error

REASON: The turnaround time definition for standard job groups with remote jobs has been defined on more than one TURNTIME statement.

ACTION: Correct the TURNTIME statement in prefix.MICS.PARMS(PWRGROUP).

REFERENCES: PWRPGEN

+----------------+
| P W R 0 0 0 4 4 |
+----------------+

TEXT: NO ENDING DEADLINE TIME DATA ELEMENT SPECIFIED

TYPE: Error

```
   REASON:  You must specify both a starting and an ending
            time for DEADLINE jobs.

   ACTION:  Correct the TURNTIME statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 4 5 |
+-----------------+
   TEXT:  STARTING VALUE FOR DEADLINE MUST BE 1, 2, OR 3

   TYPE:  Error

   REASON:  An invalid value was specified for the deadline
            start time.  It must be numeric in the range 1-3.

   ACTION:  Correct the TURNTIME statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 4 6 |
+-----------------+

   TEXT:  ENDING VALUE FOR DEADLINE MUST BE 3 OR 4
   TYPE:  Error

   REASON:  The end time element for deadline jobs must be
            either 3 or 4.

   ACTION:  Correct the TURNTIME statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 4 7 |
+-----------------+

   TEXT:  DEADLINE 3 3 NOT VALID
   TYPE:  Error
```

```
   REASON:  The starting and ending times for deadline jobs
            must be different.

   ACTION:  Correct the TURNTIME statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 4 8 |
+-----------------+

   TEXT:  DEADLINE VALUES MUST BE ASCENDING

   TYPE:  Error
   REASON:  The deadline end time value must be later than
            the start time.

   ACTION:  Correct the TURNTIME statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 4 9 |
+-----------------+

   TEXT:  ONLY STARTING AND ENDING VALUES MAY BE SPECIFIED
            FOR DEADLINE TURNAROUND

   TYPE:  Error
   REASON:  Only starting and ending values are specified for
            deadline turnaround.

   ACTION:  Correct the TURNTIME statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 0 |
+-----------------+

   TEXT:  DUPLICATE DEADLINE LOCAL DEFINITIONS.

   TYPE:  Error
```

```
   REASON:  The turnaround time definition for deadline
            job groups with local jobs has been defined on
            more than one TURNTIME statement.

   ACTION:  Correct the TURNTIME statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 1 |
+-----------------+

   TEXT:  DUPLICATE DEADLINE REMOTE DEFINITIONS.

   TYPE:  Error
    REASON:  The turnaround time definition for deadline
            job groups with remote jobs has been defined on
            more than one TURNTIME statement.

   ACTION:  Correct the TURNTIME statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 2 |
+-----------------+

   TEXT:  KEYWORD NOT RECOGNIZED IN TURNTIME STMT. KEYWORD
            FOUND = %TOKEN2

   TYPE:  Error
   REASON:  The first keyword on the TURNTIME statement must
            be either STANDARD or DEADLINE.

   ACTION:  Correct the TURNTIME statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 3 |
+-----------------+

   TEXT:  TARGET TIMES MUST NOT BE LESS THAN THE TARGET TIME
```

```
            SPECIFIED FOR THE PREVIOUS JOB GROUP
          TARGET FOUND = %TARGET.  PREVIOUS FOUND =
            %LASTTAR

   TYPE:  Error

   REASON:  The GROUP statements must be in ascending
            sequence by the turnaround target time specified
            on each statement in order to assure that there
            is no overlap.

   ACTION:  Correct the GROUP statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 4 |
+-----------------+
   TEXT:  TARGET TIME IS EQUAL TO PREVIOUS TARGET TIME

   TYPE:  Warning

   REASON:  The turnaround service target assigned to each
            job group in the GROUP statement must not
            overlap.

   ACTION:  Correct the GROUP statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 5 |
+-----------------+
   TEXT:  PURGE TIME MUST BE GT SUBMIT TIME WITH DAYS
            SPANNED OF 1.

   TYPE:  Error

   REASON:  If a deadline job's processing is expected to
            complete in a single calendar day, then the
            target purge time of day must be later than the
            submit time of day.

   ACTION:  Correct the DEADLINE statement in
            prefix.MICS.PARMS(PWRGROUP).
```

```
   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 6 |
+-----------------+

   TEXT:  KEYWORD NOT RECOGNIZED. KEYWORD = %TOKEN1

   TYPE:  Error

   REASON:  The only valid keywords for the PWRGROUP member
            are TURNTIME, GROUP, DEADLINE, and OPEN.

   ACTION:  Correct the DEADLINE statement in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 7 |
+-----------------+
   TEXT:  %VALUEN IS NOT NUMERIC. VALUE FOUND = %VALUEC

   TYPE:  Error

   REASON:  The value shown is from the last line processed.
            A numeric value was expected.

   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 8 |
+-----------------+
   TEXT:  %VALUEN MUST BE IN THE RANGE %VALUEL - %VALUEH.
            VALUE FOUND = %VALUE

   TYPE:  Error

   REASON:  The value shown is from the last line processed.
            The value is outside of the expected range.

   ACTION:  Correct the statement in error in
```

```
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 5 9 |
+-----------------+
   TEXT:  GROUP VALUE IS NOT NUMERIC. VALUE FOUND = %TOKEN2

   TYPE:  Error

   REASON:  The value shown is from the last line processed.
            The job group identifier must be numeric.

   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN

+-----------------+
| P W R 0 0 0 6 0 |
+-----------------+

   TEXT:  GROUP VALUE IS NOT INTEGER. VALUE FOUND = %TOKEN2

   TYPE:  Error

   REASON:  The value shown is from the last line processed.
            The job group identifier must be an integer.

   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 6 1 |
+-----------------+

   TEXT:  STANDARD GROUP VALUE NOT IN RANGE 1-149. VALUE
            FOUND = %GROUP

   TYPE:  Error

   REASON:  The value shown is from the last line processed.
            The job group identifier for standard groups must
            be in the range 1-149.
```

```
   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 6 2 |
+-----------------+

   TEXT:  DEADLINE GROUP VALUE NOT IN RANGE 150-179. VALUE
            FOUND = %GROUP

   TYPE:  Error

   REASON:  The value shown is from the last line processed.
            The job group identifier for deadline groups must
            be in the range 150-179.

   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 6 3 |
+-----------------+
   TEXT:  OPEN GROUP VALUE NOT IN RANGE 180-195. VALUE
            FOUND = %GROUP

   TYPE:  Error

   REASON:  The value shown is from the last line processed.
            The job group identifier for open groups must be
            in the range 180-195.

   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 6 4 |
+-----------------+

   TEXT:  JOB GROUP NAME IS NOT ENCLOSED IN QUOTES
```

```
   TYPE:  Error

   REASON:  The value shown is from the last line processed.
            The job group name must be enclosed in single
            quotation marks to delimit it from the rest of
            the statement.

   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 6 5 |
+-----------------+

   TEXT:  JOB GROUP PREVIOUSLY DEFINED. NAME = %DUPDDN

   TYPE:  Error

   REASON:  The job group name has been assigned to more than
            one job group identifier.  Job group names must
            be unique.

   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 6 6 |
   TEXT:  ALL 4 TURNTIME COMBINATIONS MUST BE DEFINED

   TYPE:  Error

   REASON:  All of the following turnaround time definition
            statements must be provided:

              TURNTIME STANDARD LOCAL ...
              TURNTIME STANDARD REMOTE ...
              TURNTIME DEADLINE LOCAL ...
              TURNTIME DEADLINE REMOTE ...

   ACTION:  Add the missing statement(s) to
            prefix.MICS.PARMS(PWRGROUP).
```

```
   REFERENCES:  PWRPGEN

+-----------------+
| P W R 0 0 0 6 7 |
+-----------------+

   TEXT:  NO JOB GROUPS WERE DEFINED IN THE PWRGROUP MEMBER

   TYPE:  Warning

   REASON:  At least one job group must be defined.

   ACTION:  Add the missing statement(s) to
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN
 +-----------------+
| P W R 0 0 0 6 8 |
+-----------------+

   TEXT:  FATAL ERRORS ENCOUNTERED IN PWRGROUP

   TYPE:  Error

   REASON:  The parameter specifications, validations, or both
            performed for information coded in PWRGROUP did
            not complete satisfactorily.

   ACTION:  Review the error messages printed in MICSLOG
            prior to this message.  These earlier messages
            point you to the areas that you need to research
            to resolve the fatal errors.

   REFERENCES:  PWRPGEN

+-----------------+
| P W R 0 0 0 6 9 |
+-----------------+

   TEXT:  CONFIGURATION INFORMATION NOT SPECIFIED FOR SYSTEM
            %ORGSYSID.  SOME PJB ELEMENTS MAY BE INVALID

   TYPE:  Warning

   REASON:  No configuration information was provided for the
            ORGSYSID listed.  All SIO counts will be assigned
            to a device class and type of *UNKNOWN.  Also,
            device class SIO counts will be accumulated to
            element fffOSIO (other).
```

```
   ACTION:  Define the system configuration in
            prefix.MICS.PARMS(PWRCONFG) if required to meet
            your measurement needs.

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 7 0 |
+-----------------+

   TEXT:  OPTIONS NOT SPECIFIED FOR SYSTEM %ORGSYSID
            SPECIFIED IN PWRCONFG

   TYPE:  Error

   REASON:  There must be a one-to-one correspondence between
            the systems identified in the PWRCONFG member
            and the systems listed on the OPTIONS statement
            of prefix.MICS.PARMS(PWROPS).

   ACTION:  Review the system identifiers listed in the
            PWRCONFG and PWROPS members of prefix.MICS.PARMS.
            Revise the system identifiers and ensure that
            there is a one-to-one correspondence between the
            members.

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 7 1 |
+-----------------+

   TEXT:  VSE/POWER PARAMETER GENERATION COMPLETED
            SUCCESSFULLY
   TYPE:  Information

   REASON:  All processing has completed successfully.

   ACTION:  None necessary.

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 7 5 |
+-----------------+
```

TEXT: PROCESSING DSN=%DSNAME

TYPE: Information

REASON: Notice that processing has started on the parameter member.

ACTION: None necessary.

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 7 6 |
+-----------------+
```

TEXT: %LINEIN

TYPE: Information

REASON: For each line of the member processed, this message echoes its content to the MICSLOG.

ACTION: None necessary.

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 7 7 |
+-----------------+
```

TEXT: BUILDING DEVICE TYPE/DEVICE CLASS FORMAT

TYPE: Information

REASON: Acknowledges that the first step of generating code for the product is building the format for device types and classes.

ACTION: None necessary.

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 7 8 |
+-----------------+
```

TEXT: BUILDING PREFIX.MICS.USER.SOURCE($PWRMSTR).

TYPE: Information

REASON: Acknowledges that the $PWRMSTR code is being built.

ACTION: None necessary.

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 7 9 |
+-----------------+
```

TEXT: CONFIRMING SYSID MATCHES IN PWROPS AND PWRCONFG.

TYPE: Information

REASON: Acknowledges that the SYSID matching process has started.

ACTION: None necessary.

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 8 0 |
+-----------------+
```

TEXT: JOBGROUP SERVICE FACTOR IS REQUIRED BEFORE THE TITLE.

TYPE: Error

REASON: The job group service factor is missing from the previous statement. It must be listed prior to the job group title in the statement.

ACTION: Correct the statement in error in prefix.MICS.PARMS(PWRGROUP).

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 8 1 |
+-----------------+
```

TEXT: JOBGROUP SERVICE FACTOR IS NOT NUMERIC. VALUE FOUND = %VALUEC.

```
   TYPE:  Error

   REASON:  The job group service factor must be numeric. If
            priority accounting is not being used, set this
            value to one.

   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).

   REFERENCES:  PWRPGEN
+-----------------+
| P W R 0 0 0 8 2 |
+-----------------+

   TEXT:  JOBGROUP SERVICE FACTOR IS NOT IN RANGE .1 TO
            99.9.  VALUE FOUND = %PGPFCTR

   TYPE:  Error

   REASON:  The job group service factor must be in the range
            of .1 to 99.9.  If priority accounting is not in
            use, set this value to one.

   ACTION:  Correct the statement in error in
            prefix.MICS.PARMS(PWRGROUP).
    REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 8 3 |
+-----------------+

   TEXT:  PROCESSING COMPLETED FOR MEMBER %PGENNAME

   TYPE:  Information

   REASON:  Acknowledges that the contents of the member of
            prefix.MICS.PARMS has been processed.

   ACTION:  None necessary.

   REFERENCES:  PWRPGEN

+-----------------+
| P W R 0 0 0 8 4 |
+-----------------+

   TEXT:  DEVICE ADDRESS RANGE MUST BE SPECIFIED WITHOUT
```

```
            BLANKS

   TYPE:  Error

   REASON:  Device range definitions must be in the form
            xxx-xxx.  They may not contain embedded blanks.

   ACTION:  Correct the value coded for device address in
            prefix.MICS.PARMS(PWRCONFG).

   REFERENCES:  PWRPGEN
+-----------------+
| P W R 0 0 0 8 5 |
+-----------------+

   TEXT:  A FATAL ERROR HAS BEEN DETECTED; JOB ABENDING

   TYPE:  Error

   REASON:  Before generating the $PWRMSTR code, errors
            detected in the validation and processing of the
            members of the prefix.MICS.PARMS data set must be
            resolved.  This ABEND ensures that code is not
            generated if errors have been detected.

   ACTION:  Review the error messages printed in MICSLOG
            prior to this message.  These earlier messages
            point you to the areas that you need to research
            to resolve the fatal errors.

   REFERENCES:  PWRPGEN


+-----------------+
| P W R 0 0 0 8 6 |
+-----------------+

   TEXT:  DDNAME %DDNAME IS SPECIFIED IN PWROPS FOR MULTIPLE
            ORGSYSIDS
          AT LEAST ONE SPECIFIES THE NOSYSID OPTION

   TYPE:  Error

   REASON:  This error can result from one of two conditions.
            First, the OPTIONS statements for the DDname may
            contain a mixture of SYSID and NOSYSID keywords.
            This is invalid since all data read from an input
            DD must be in the same format.
```

The other possibility is that all of the data being read is in NOSYSID form. This is invalid because it is impossible to tell the source of the data without the VSESYSID contained in the SYSID record header.

ACTION: Review the data attributes and correct the OPTIONS statements in prefix.MICS.PARMS(PWROPS).

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 8 7 |
+-----------------+
```

TEXT: INVALID SUSPEND LIMIT: %TOKEN2
VALUE MUST BE AN INTEGER IN THE RANGE 0-99

TYPE: Error

REASON: The suspend limit value is out of range. It must be in days and have a value between 0 and 99 inclusive.

ACTION: Review the data attributes and correct the SUSPENDLIMIT statement.

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 0 0 8 8 |
+-----------------+
```

TEXT: SUSPENDLIMIT KEYWORD NOT FOUND; THE DEFAULT VALUE OF FIVE WILL BE USED

TYPE: Information

REASON: Notification that the default value will be used.

ACTION: None required.

REFERENCES: PWRPGEN

```
+-----------------+
| P W R 0 1 0 0 0 |
+-----------------+
```

```
 TEXT:  FIRST INPUT TIMESTAMP THIS RUN = %WRKTS
       FROM ORGSYSID %ORGSYSID DDNAME %VSEDDNAM

   TYPE:  Information

   REASON:  Acknowledges that CA MICS has started to process
            data.

   ACTION:  None necessary.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 0 0 1 |
+-----------------+

   TEXT:  INPUT STATISTICS FOR FILE---------- %VSEDDNAM
          -C- SPOOL-ACCESS CONNECT RECORDS--- %MSGCC
            -S- RJE/SNA ACCOUNT RECORDS-------- %MSGCS
          -E- EXECUTION ACCOUNT RECORDS------ %MSGCE
            -T- RJE/BSC ACCOUNT RECORDS-------- %MSGCT
          -L- LIST ACCOUNT RECORDS----------- %MSGCL
            -U- SYSTEM-UP ACCOUNT RECORDS------ %MSGCU
          -M- TRANSMITTER ACCOUNT RECORDS---- %MSGCM
            -V- RECEIVER ACCOUNT RECORDS------- %MSGCV
          -N- NETWORK ACCOUNT RECORDS-------- %MSGCN
            -X- SPOOL-ACCESS OPERATION RECORDS- %MSGCX
          -P- PUNCH ACCOUNT RECORDS---------- %MSGCP
            -?- UNKNOWN RECORD TYPES----------- %MSGC0
          -R- READER ACCOUNT RECORDS--------- %MSGCR
            --- TOTAL RECORDS IN FILE---------- %MSGCTOT

   TYPE:  Information

   REASON:  Acknowledges that the number of records shown for
            each record type listed were read in.

   ACTION:  None necessary.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 0 0 2 |
+-----------------+

   TEXT:  VSE SYSID %VSESYSID NOT DEFINED IN
            PREFIX.MICS.PARMS(PWROPS).  ABEND FOLLOWS.
```

TYPE: Error

REASON: An unknown VSE system identification has been found in the data. All VSE/POWER systems must be defined in PWROPS.

ACTION: Define the new systems in the PWROPS, PWRCONFG, and SYSID members of prefix.MICS.PARMS.

REFERENCES: DYPWRWFM

```
+-----------------+
| P W R 0 1 0 0 3 |
+-----------------+
```

TEXT: PWRGPRTE DID NOT ASSIGN VALID VALUE TO JOBGROUP. ABEND FOLLOWS.

TYPE: Error

REASON: The PWRGPRTE routine assigned a job group value that was not defined in the PWRGROUP parameter member. To prevent entering invalid data in the database, processing is ABENDed.

ACTION: Review the job group information in the members PWRGROUP and PWRGPRTE of prefix.MICS.PARMS and revise the parameters to match one another.

REFERENCES: DYPWRWFM

```
+-----------------+
| P W R 0 1 1 0 0 |
+-----------------+
```

TEXT: SHORT RECORD OF UNKNOWN TYPE FOUND IN FILE %VSEDDNAM
RECORD COUNT FROM ALL FILES IS %INSMF
NEXT COLUMN TO BE READ IS %COL
LENGTH OF RECORD IS %RECLEN

TYPE: Warning

REASON: This message is printed when a short record is encountered and it is not possible to read the record type.

```
   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 0 2 |
+-----------------+
   TEXT:  SHORT EXECUTION ACCOUNT RECORD FOUND IN FILE
            %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning

   REASON:  This message is printed when a short record is
            encountered.

   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM
 +-----------------+
| P W R 0 1 1 0 4 |
+-----------------+

   TEXT:  SHORT EXECUTION ACCOUNT RECORD FOUND IN FILE
            %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning

   REASON:  This message is printed when a short record is
            encountered.

   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.
   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 0 6 |
+-----------------+
```

```
   TEXT:  SHORT EXECUTION ACCOUNT RECORD FOUND IN FILE
            %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning

   REASON:  This message is printed when a short record is
            encountered.
    ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 0 8 |
+-----------------+

   TEXT:  SHORT EXECUTION ACCOUNT RECORD FOUND IN FILE
            %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning
   REASON:  This message is printed when a short record is
            encountered.

   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 1 0 |
+-----------------+

   TEXT:  SHORT LIST ACCOUNT RECORD FOUND IN FILE %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN
   TYPE:  Warning

   REASON:  This message is printed when a short record is
            encountered.
```

```
   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 1 2 |
+-----------------+

   TEXT:  SHORT PUNCH ACCOUNT RECORD FOUND IN FILE %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning

   REASON:  This message is printed when a short record is
            encountered.

   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 1 4 |
+-----------------+

   TEXT:  SHORT TRANSMITTER OR RECEIVER ACCOUNT RECORD FOUND
            IN FILE %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning

   REASON:  This message is printed when a short record is
            encountered.

   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 1 6 |
```

```
+-----------------+

   TEXT:  SHORT READER ACCOUNT RECORD FOUND IN FILE
            %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning

   REASON:  This message is printed when a short record is
            encountered.

   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM
 +-----------------+
| P W R 0 1 1 1 8 |
+-----------------+

   TEXT:  SHORT NETWORK ACCOUNT RECORD FOUND IN FILE
            %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning

   REASON:  This message is printed when a short record is
            encountered.

   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 2 0 |
+-----------------+

   TEXT:  SHORT RJE - BSC ACCOUNT RECORD FOUND IN FILE
            %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning
```

```
   REASON:  This message is printed when a short record is
            encountered.
   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 2 2 |
+-----------------+

   TEXT:  SHORT RJE - SNA ACCOUNT RECORD FOUND IN FILE
            %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning
   REASON:  This message is printed when a short record is
            encountered.

   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 2 4 |
+-----------------+

   TEXT:  SHORT SYSTEM-UP ACCOUNT RECORD FOUND IN FILE
            %VSEDDNAM
          RECORD COUNT FROM ALL FILES IS %INSMF
          NEXT COLUMN TO BE READ IS %COL
          LENGTH OF RECORD IS %RECLEN

   TYPE:  Warning

   REASON:  This message is printed when a short record is
            encountered.

   ACTION:  Using the diagnostic information printed in the
            message text, review the records indicated.

   REFERENCES:  DYPWRWFM
```

```
+-----------------+
| P W R 0 1 1 2 5 |
+-----------------+

   TEXT:  VSE/POWER VERSION SPECIFICATION IS IN ERROR FOR
            ORGSYSID %ORGSYSID
          PLEASE VERIFY VERSION FOR ALL SYSTEMS LISTED IN
            PREFIX.MICS.PARMS(PWROPS)

   TYPE:  Warning

   REASON:  The VSE/POWER version provided in the PWROPS
            member does not match the Account record version
            found in the System Identifier header.  Note,
            this applies only to sharing spool environments,
            as only they produce data containing the
            header.

            Processing will continue.  However, the VSVER
            element will contain incorrect values for one or
            more observations.

   ACTION:  Review the contents of prefix.MICS.PARMS(PWROPS)
            and revise the VERSION and ORGSYSID fields of the
            parameter as applicable.

   REFERENCES:  DYPWRWFM


+-----------------+
| P W R 0 1 1 2 6 |
+-----------------+

   TEXT:  POWER RECORD VERSION %PWRRVER IS NOT SUPPORTED

   TYPE:  Error

   REASON:  The CA MICS Analyzer Option for VSE/POWER product
            supports only specified releases of IBM's
            VSE/POWER.
    ACTION:  Review the CA MICS product's requirements (listed
            in the product guide) and the release of IBM's
            product that is available at your installation.
            Update the version field in the PWROPS member of
            prefix.MICS.PARMS.

   REFERENCES:  DYPWRWFM
```

```
+-----------------+
| P W R 0 1 1 2 7 |
+-----------------+

   TEXT:  VSE/POWER VERSION SPECIFICATION IS IN ERROR FOR
            ORGSYSID %ORGSYSID
          PLEASE VERIFY VERSION FOR ALL SYSTEMS LISTED IN
            PREFIX.MICS.PARMS(PWROPS)
          UNABLE TO CONTINUE PROCESSING

   TYPE:  Error

   REASON:  The VSE/POWER version provided in the PWROPS
            member does not match that found in the System-Up
            Account Data record.  This message may be
            accompanied by INVALID DATA errors in the SAS
            log.  The daily update step will ABEND to avoid
            corrupting the database.

   ACTION:  Examine the System-Up report for the record or
            product version found.  Review the contents of
            prefix.MICS.PARMS(PWROPS) and revise the VERSION
            and ORGSYSID fields of the parameter.

   REFERENCES:  DYPWRWFM
 +-----------------+
| P W R 0 1 1 2 8 |
+-----------------+

   TEXT:  THE FOLLOWING VSESYSID(S) ARE NOT DEFINED IN
            PWROPS FOR DD NAME %VSEDDNAM - %SKPIDS
          %SKPCOUNT RECORDS HAVE BEEN SKIPPED AND WILL NOT
            BE INCLUDED IN SUBSEQUENT RECORD COUNTS

   TYPE:  Warning

   REASON:  CA MICS will process only data belonging to
            VSE/POWER systems which have been defined in
            prefix.MICS.PARMS(PWROPS).  Data for other
            systems will be skipped.

   ACTION:  If you wish to add the referenced system(s) to
            the CA MICS database, you must define them in
            prefix.MICS.PARMS members PWROPS, PWRCONFG, and
            SYSID.  You can use the FORCE option to reprocess
            the data.

   REFERENCES:  DYPWRWFM
```

```
+-----------------+
| P W R 0 1 1 2 9 |
+-----------------+
```

TEXT: VSE/POWER VERSION %PWRVER IS NOT SUPPORTED,
CONTACT CA MICS PRODUCT SUPPORT

TYPE: Error

REASON: CA MICS processed data that contains an unsupported VSE/POWER release found in the System-Up Account data record.

ACTION: Check the DAY035 SASLOG for dump of the record with invalid release. The System-Up Account data record can be found by checking for a "U" in the second position of the record. Note the value in position 33 and contact CA MICS Product Support.

REFERENCES: DYPWRWFM

Appendix B: DATA DICTIONARY

The Data Dictionary is only available at your site, where it has been customized to your configuration and your product change level.
To see the Data Dictionary at your site, follow the instructions under Document Browse in the Document Access guide.