IBM Z Decision Support
Version 1 Release 9

System Performance Feature Guide

SC28-3209-00

Note

Before using this information and the product it supports, read the information in "Notices" on page 147.

This edition applies to version 1, release 9, modification level 0 of IBM Z Decision Support (program number 5698-B06) and to all subsequent releases and modifications until otherwise indicated in new editions.

Last updated: July 2020

Contents

Figures

Tables

Preface

The System Performance Feature Guide describes how to use the System Performance Feature of IBM® Z Decision Support®.

The book describes some basic concepts of systems management and how to develop a performance and service-level strategy. It then discusses certain performance issues and shows how you can use IBM Z Decision Support to study those issues. The book presents a top-down approach to analyzing reports for performance management.

The following terms are used interchangeably throughout this book:

- MVS™, OS/390®, and z/OS
- OPC and IBM Tivoli Workload Scheduler for z/OS
- VM and z/VM®

Who should read this book

The System Performance Feature Guide is for performance analysts and system programmers who are responsible for meeting the service-level objectives established in your organization.

What this book contains

Use this book as a guideline for developing and executing your performance and service-level strategy.

The book has these chapters:

- Chapter 1, "Introducing the IBM Z Decision Support system performance," on page 1Introducing the IBM Z Decision Support System Performance Feature provides a brief description of IBM Z Decision Support, the System Performance Feature, and System Performance Feature components.
- Chapter 2, "Developing a performance and service-level strategy," on page 5 describes how to develop service-level objectives for your systems to ensure that users receive the service they need.
- Chapter 3, "Analyzing z/OS performance," on page 15 describes some performance issues that can affect the throughput and response time of z/OS users. It shows how to use IBM Z Decision Support reports to identify resource constraints.
- Chapter 4, "Analyzing DFSMS storage management data," on page 59 describes how to use IBM Z Decision Support reports to manage the direct access storage device (DASD) and tape storage available to your system.
- Chapter 5, "Analyzing Db2 performance," on page 75 describes some issues that can affect the performance of your Db2® subsystem. It shows some of the IBM Z Decision Support reports available for analyzing Db2 performance.
- Chapter 6, "Analyzing VM performance," on page 99describes some issues that can affect performance on a VM system. It shows some of the IBM Z Decision Support reports available for analyzing VM performance.
- Chapter 7, "Analyzing system operations and security," on page 117 describes how to monitor other aspects of your system, including security management, message analysis, operations management, problem management, and change management.
- Chapter 8, "Analyzing WebSphere MQ for z/OS performance," on page 125 describes the WebSphere® MQ for z/OS (MQSeries®) component, which provides reports on WebSphere MQ subsystem performance.
- Chapter 9, "Analyzing WebSphere Application Server (WAS) performance," on page 131 provides information on how to analyze WebSphere Application Server performance.
- Chapter 10, "Analyzing WebSphere Message Broker performance," on page 139 provides information on how to analyze WebSphere Message Broker.

- Appendix A, “Support information,” on page 143 describes how to obtain support for IBM software products.

A glossary and index follows the appendixes.

Publications

This section describes how to access the IBM Z Decision Support publications online.

For a list of publications and related documents, refer to “IBM Z Decision Support publications” on page 145.

Accessing publications online

Publications for this and all other IBM products, as they become available and whenever they are updated, can be viewed on the IBM Knowledge Center website where you can also download the associated PDF.

IBM Z Decision Support V1.9.0
https://www.ibm.com/support/knowledgecenter/SSH53X_1.9.0

IBM Knowledge Center
https://www.ibm.com/support/knowledgecenter

Accessibility

Accessibility features help users with a physical disability, such as restricted mobility or limited vision, to use software products successfully.

With this product, you can use assistive technologies to hear and navigate the interface. You can also use the keyboard instead of the mouse to operate all features of the graphical user interface.

For additional information, refer to the IBM Accessibility website:

https://www.ibm.com/accessibility

Support information

If you have a problem with your IBM software, you want to resolve it quickly. IBM provides the following ways for you to obtain the support you need.

- Searching knowledge bases: You can search across a large collection of known problems and workarounds, Technotes, and other information.
- Obtaining fixes: You can locate the latest fixes that are already available for your product.
- Contacting IBM Support: If you still cannot solve your problem, and you need to work with someone from IBM, you can use a variety of ways to contact IBM Support.

For more information about these ways of resolving problems, see Appendix A, “Support information,” on page 143.

Conventions used in this book

This guide uses several conventions for special terms and actions, operating system-dependent commands and paths, and margin graphics.

The following terms are used interchangeably throughout this book:

- MVS, OS/390®, and z/OS.
- VM and z/VM.

Typeface conventions

This guide uses the following typeface conventions.

Bold

- Lowercase commands and mixed case commands that are otherwise difficult to distinguish from surrounding text
- Interface controls (check boxes, push buttons, radio buttons, spin buttons, fields, folders, icons, list boxes, items inside list boxes, multicolumn lists, containers, menu choices, menu names, tabs, property sheets), labels (such as **Tip**, and **Operating system considerations**)
- Column headings in a table
- Keywords and parameters in text

Italic

- Citations (titles of books, diskettes, and CDs)
- Words defined in text
- Emphasis of words (words as words)
- Letters as letters
- New terms in text (except in a definition list)
- Variables and values you must provide

Monospace

- Examples and code examples
- File names, programming keywords, and other elements that are difficult to distinguish from surrounding text
- Message text and prompts addressed to the user
- Text that the user must type
- Values for arguments or command options

Changes in this edition

This edition is an update to the previous edition of the same book.

No technical changes have been added for this edition. However, chapters on EREP and INFOMAN have been removed as the products have been withdrawn from service.

Chapter 1. Introducing the IBM Z Decision Support system performance

IBM Z Decision Support helps you manage the performance service-level of your computer systems. The System Performance Feature helps you analyze the performance of your z/OS and Virtual Machine (VM) systems and their subsystems. This chapter introduces you to the type of information you can get from System Performance Feature reports.

Systems monitoring, tuning, and planning activities can be successful only if a complete picture of a system's behavior exists and is understood. IBM Z Decision Support helps you achieve this goal.

The System Performance Feature collects data recorded by numerous licensed programs, summarizes the data, and presents it in a variety of forms for use in systems management. The System Performance Feature has powerful functions that let you collect large volumes of data and store the data you need. For example, you can collect data from system management facilities (SMF), Resource Management Facility (RMF™), Data Facility Storage Management Subsystem (DFSMS), and other programs. Once the data is in the IBM Z Decision Support database, you can create and display reports that show the data from these sources. You can also consolidate data from different sources into one report.

Other optional IBM Z Decision Support features are:

Accounting
The Resource Accounting provides reports that show billing records for levels of system usage. You can specify the measurement units to be used, such as CPU seconds for resource billing or CICS transactions for function billing. Refer to the *Resource Accounting for z/OS* for more information.

AS/400
The AS/400 System Performance Feature provides reports that show AS/400 accounting, configuration, error, and performance data that has been collected from AS/400 nodes in your network. Refer to the *AS/400 System Performance Feature Guide and Reference* for more information.

Customer Information Control System (CICS)
The CICS Performance feature provides reports that can help you analyze the performance of your Customer Information Control System (CICS). For more information, refer to the *CICS Performance Feature Guide and Reference*.

Information Management System (IMS)
The IMS Performance feature provides performance information on your Information Management System (IMS). Refer to the *IMS Performance Feature Guide and Reference* for more information.

Networks
The Network Performance feature provides reports that show the response times, availability, configuration, throughput, and any problems in your network. Refer to the *Network Performance Feature Installation and Administration* for more information.

Distributed Systems Performance
The Distributed Systems Performance feature provides reports that show UNIX accounting, configuration, error, and performance data that has been collected from UNIX nodes in your network. Linux configuration and performance data on distributed systems is also reported. Refer to the *Distributed Systems Performance Feature Guide and Reference* for more information.

System Performance Feature components

The System Performance Feature includes the following components. Each component provides support for a specific operating system, subsystem, or licensed program.

Db2
The Db2 component collects data from the SMF log and stores data relevant to Db2 performance. You can create reports that show Db2 response times, transaction statistics, system statistics, buffer pool and EDM pool statistics, and Db2 exceptions.

DFSMS
The DFSMS component uses data from the DFSMS/MVS™ Data Collection Facility (DCOLLECT) to produce reports about active data set storage, volume usage, and Data Facility Storage Management Subsystem hierarchical storage manager (DFSMShsm™) backup and migration storage. You can use these reports to manage your storage subsystem on various levels and for capacity planning.

DFSMS/RMM
The Data Facility Storage Management System/Removable MediaManagement (DFSMS/RMM) component collects records from the extract data file for volume and data set records in Db2 tables.

EREP

The EREP component includes reports that complement the more detailed reports produced by the Environmental Record and Editing Printing (EREP) program. These reports provide management overview trend reports on hardware errors for DASD and virtual telecommunications access method (VTAM®) controlled devices, and initial program loads (IPLs) of the central processors.

HTTP Server for z/OS (ICSS)
The HTTP Server for z/OS (ICSS) component, previously known as Internet Connection Secure Server, provides reports on configuration and performance data retrieved from the HTTP Server for z/OS.

IXFP
The IBM Extended Facilities Product (IXFP) component provides reports on the data collected by the XSA/Reporter. XSA/Reporter is the part of IXFP that collects data from your RAMAC® Virtual Array (RVA) subsystem. IXFP reports contain data on channel statistics, device performance, drive module performance, and deleted data space release.

Linux on zSeries

The Linux on zSeries component (previously known as Linux for z/OS) provides reports on configuration and performance data that has been gathered from Linux on zSeries systems.

Lotus Domino for z/OS
The Lotus Domino for z/OS component (previously known as Domino) collects records from the SMF data set and stores extracted and summarized data in the IBM Z Decision Support database.

Message analysis/automation
The message analysis/automation component reports contain message statistics based on system log (SYSLOG) data from job entry subsystem 2 (JES2) or JES3 systems, and data from the NetView® log. You can use the reports to get statistics on message types and IDs, message rates, message suppression, and automation. The reports also indicate messages that are candidates to be suppressed, passed to NetView, or not passed to NetView.

RACF®
The Resource Access Control Facility (RACF) component reports are intended for the security administrator and the resource owners. You use the RACF component as a complement to the RACF Report Writer. This component also contains configuration data extracted by the RACF Database Unload utility. The RACF component supports both z/OS and z/VM data.

TCP/IP for z/OS
The TCP/IP for z/OS component (previously known as TCP/IP for MVS) collects records from the SMF data set and stores this data in the Db2® database.

Tivoli Information Management for z/OS (INFOMAN)
The Tivoli Information Management for z/OS (INFOMAN) component, (previously known as Tivoli Service Desk), contains reports for problem and change management. You can use these reports to analyze the number of changes and problems, and determine how many there have been, what caused them, and how long it took to resolve them.

Tivoli Storage Manager for z/OS (ADSM)
The Tivoli Storage Manager for z/OS (ADSM) component, previously known as Tivoli Storage Manager, collects data from the SMF log and provides reports on the archiving activity on your system.

Tivoli Workload Scheduler for z/OS (OPC)
The Tivoli Workload Scheduler for z/OS (OPC) component provides reports on Tivoli Workload Scheduler for z/OS (previously known as Operation Planning and Control). This component collects Tivoli Workload Scheduler for z/OS track log data and stores it in the IBM Z Decision Support

database. You can use the IBM Z Decision Support reporting dialog to create reports based on this data.

VM Accounting
The VM accounting component provides reports on your VM users. You can use these reports to analyze the resources being used and who is using them.

WebSphere
The WebSphere Activity and WebSphere Interval components collect data from the SMF log. You can create reports that show server statistics, web application response time and complete activity metrics with different levels of details (class, method).

MQSeries®
The WebSphere MQ for z/OS component, (previously known as MQSeries), collects records from the SMF data set and stores them in the Db2 database.

z/OS Availability
The z/OS Availability component (previously known as MVS Availability) provides availability statistics on the sysplex, the z/OS system, and the user-defined address space. This component includes overview and detailed reports to show specific availability problems.

z/OS Interval Job/Step Accounting
The z/OS Interval Job/Step Accounting component (previously known as MVS Interval Job/Step Accounting) manages SMF30 subtype records in such a way that accounting information is always available.

z/OS Performance Management (MVSPM)
The z/OS Performance Management (MVSPM) component, (previously known as MVS Performance Management), includes reports showing the use of major system resources at both a system and workload level. You can use the z/OS Performance Management component to analyze and manage the processor, processor storage, and I/O subsystem resources. You can also investigate exceptions and deviations from service-level objectives. This component includes overview reports for assessing overall system performance and detailed reports that help you investigate particular performance problems.

z/OS System (MVS)
The z/OS System (MVS) component, (previously known as MVS), includes reports that help you monitor system management areas such as utilization and capacity. You can use the reports to see trends for the most important measurements and report on system management areas.

This component includes exceptions reports and detail, overview, and trend reports for analyzing response times and resource usage. It also provides job, started-task, Time Sharing Option (TSO) session, program, device, and printer statistics. Initial program load (IPL) details are also supplied.

z/VM Performance
The z/VM Performance component (previously known as VMPRF) provides reports on virtual machine processor usage and response times, system performance characteristics, and system configuration. You can use this data for response-time analysis or for a breakdown of user or user-group activity.

Chapter 2. Developing a performance and service-level strategy

This chapter describes the steps involved in performance management - the activity of allocating data processing resources to applications according to service-level objectives.

As a performance analyst, your responsibility is to measure, monitor, and manage system resources to meet the business needs of an organization. Users like to know that someone regularly reviews their system resource, and optimizes their performance.

Consistency is key to performance management. Airlines do not occasionally check the maintenance of airplanes and banks do not sporadically balance accounts. To maintain your service levels, you must check them regularly.

Figure 1 on page 5 shows that performance management is a cyclic process; the steps are iterative in nature. After alleviating a bottleneck, remeasure and reevaluate performance to verify that you did not cause a different constraint and to ensure that the change worked.

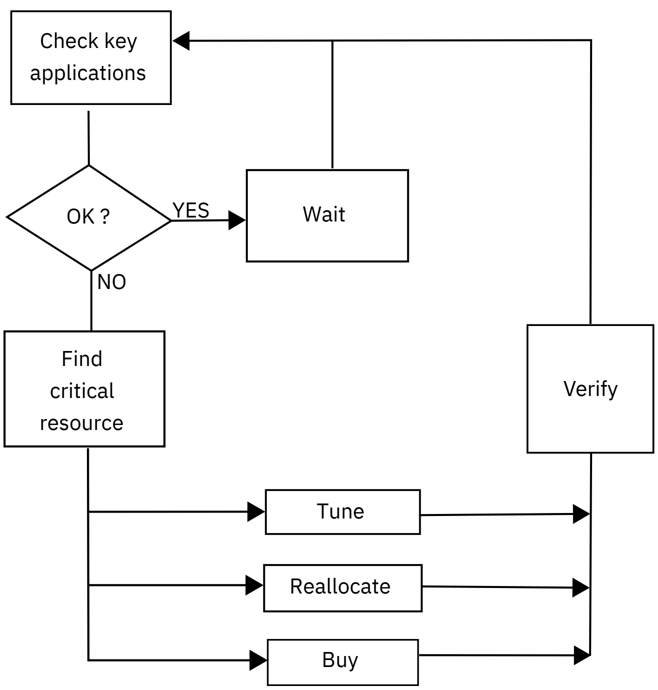

Figure 1. Performance management cycle

The process of measuring, evaluating, and verifying is necessary every time potential solutions are identified and implemented. The possible results are:

- The problem is solved, and no new problems are identified. Analysis is complete.
- The immediate problem being addressed is solved, but another problem becomes apparent. This result is not unusual because one impasse often disguises another. In this event, analysis begins again.
- The problem remains and further investigation is required. Analysis continues.

To manage performance:

1. Define service-level objectives.
2. Gather performance data.
3. Create useful reports from the performance data.
4. Analyze performance reports to see if your objectives have been met.

5. Identify potential bottlenecks in the system.
6. Analyze detailed performance data about suspected constraints.
7. Decide where you can obtain the resources needed.
8. Verify the removal of performance constraints.

Steps 2 though 8 represent the daily performance management process. Repeat Step 1 as the needs of your organization change.

Defining service-level objectives

To monitor system performance, you must first have a standard by which to measure current performance. Most organizations have key applications that require better service than other applications. Your organization must establish priorities for each application and then ensure that each application receives an adequate level of service.

The business decisions resulting from prioritizing applications or workloads constitute service-level objectives. Some organizations outline the level of service an IT department is to provide in a formal service-level agreement (SLA). Not all organizations develop formal SLAs with their users. If this is the case, at least establish informal service-level objectives. Otherwise, you cannot determine what is acceptable and unacceptable performance, and therefore needs to be investigated and corrected.

How you define good performance for your system depends on your particular data processing needs and their priority. Service-level objectives should be realistic, in line with your budget, understandable, and measurable.

The definition of service-level objectives is iterative. Expect to update your service-level objectives with every change in:

- Workload
- Resource consumption of the work
- Turnaround and response time requirements
- Needs or priorities of the business
- Product capabilities, which can lead to revised performance expectations

Detailed objectives make such changes noticeable and help identify solutions to performance problems that arise because of the changing nature of the workload. The definition of service-level objectives is not a trivial task. It is necessary for disciplined performance analysis and for planning new applications and additional work.

To define service-level objectives:

1. Specify the terms in which to state objectives.
2. Document the current workload—the amount and categories of work. For example:

   TSO
      Trivial and non-trivial transactions

   IMS
      Transaction types

   CICS
      Transaction types

   Conversational Monitor System (CMS)
      Transaction types

   Batch
      Job classes (MVS and VM)

   Group the categories according to the needs of the business (for example, accounting, sales, order entry, and so on). You categorize your workload to establish priorities and different objectives. This grouping lets you more effectively report on the management and contribution of departments and resources in support of the business. Reclassify workload based on the organization's needs.

3. Measure and document resources used by each workload category during acceptable response times.
4. Set service-level objectives for each workload category using these acceptable response times and activity loads.

The following sections describe each of the previous steps in detail.

Specifying terms for objectives

Service-level objectives must state what is expected of the system in specific terms for each workload category. Common objectives can include:

- Average response time
- Activity load (the total number of transactions completed in a day, hour, or minute)
- System and subsystem availability, including mean time to failure and the number and duration of downtimes
- Windows provided for batch jobs

Objectives such as these define the system's workload and determine the requirements of data processing resources—processor, processor storage, and I/O.

The first step in defining service-level objectives is to choose the terms in which you will specify objectives. You monitor system performance to ensure that you provide users with an adequate level of service. The key indicators of service levels are:

- Response time
  - Average response time for interactive work (TSO, IMS, CICS, CMS, and so on)
  - Turnaround time for batch work
- Activity load
  - Batch throughput
  - Interactive transaction rate
  - Number of concurrent interactive users

Your goal is to give the users the best possible response times. However, saying you will give the user a 1-second response time is not enough. You have to quantify under what conditions you can deliver that response time. You must define each workload by response time and by the activity load.

For example, you might define your performance objective for the CICS subsystem as a 2-second response time for up to 200 transactions per minute. If the activity load exceeds 200 transactions per minute, you can no longer guarantee the 2-second response time.

Documenting the workload

Before setting objectives, you must understand and document the current workload on the system.

It is **very important** to fully document your workload. Some of the most significant performance gains to be achieved are accomplished by workload management. The more details you know about the workload, the better you can manage it.

You should document your workload by the amount and categories of work:

- Priority of the work
- Current activity rates and response time levels
- Different periods during which objectives and priorities vary for the same work
- Resource consumption of the work
- Types of users requiring different objectives
- Ability to track and report on work according to your organization's needs, such as by departments, groups, projects, and so on

Start with the workload categories that are currently defined for your system (for example, TSO and CMS transaction types, batch windows, and so on). Usually these categories are not fully defined. Table 1 on page 8 lists some factors to consider when defining each workload category, including resources consumed by each workload. When initially defining the categories, the resource consumption will probably reflect expected resource usage; the measurement of actual resource usage on z/OS is described in more detail in Measuring resource consumption.

Table 1. Suggested factors to include in documentation of workload

TSO	**By transaction type**	**For total TSO**
• Minimum/maximum/average number of active users logged on per hour/shift/day		X
• Average processor time per transaction	X	X
• Processor storage used	X	X
• Average I/O time per transaction	X	X
• Average service units required to complete	X	X
• TSO region size		X
• TSO command name and/or user ID	X	
Batch	**By batch window**	**For total batch**
• Number of jobs per unit of time (hour, shift, and so on)	X	X
• Arrival rate	X	X
• Average elapsed time	X	
• Processor time per hour/shift/day	X	X
• Processor storage used	X	X
• Number of EXCPs	X	
• Average service units required to complete	X	
• Virtual storage requested	X	
CICS	**By transaction type**	**For total CICS**
• Minimum/maximum/average number of active terminals per hour/shift/day		X
• Average/maximum number of transactions per second	X	X
• Maximum number of transactions per hour	X	X
• Average/maximum number of transactions per schedule	X	
• Average/maximum number of file control calls per transaction	X	X
• Average/maximum number of TP calls per transaction	X	X
• Average/maximum processor time per transaction	X	X
• Processor storage used	X	X
• Transient data and temporary storage calls per transaction	X	X
IMS or CICS with DB	**By transaction type**	**For total IMS**

Table 1. Suggested factors to include in documentation of workload (continued)		
• Minimum/maximum/average number of active terminals per hour/shift/day		X
• Average/maximum number of transactions per second	X	X
• Maximum number of transactions per hour	X	X
• Average/maximum number of transactions per schedule	X	
• Control blocks (PSBs and DMBs) loaded per schedule (IMS only)	X	
• Average/maximum number of DB calls per transaction	X	X
• Average/maximum number of DC calls per transaction	X	X
• Average/maximum processor time per transaction	X	X
• Processor storage used	X	X
• Average/elapsed time per transaction	X	X
• Largest control block storage required		X

After you define the categories, review them for definition problems. The different categories distinguish work by resource needs, objectives that must be met, priorities, and so on. For example, all jobs submitted from similar development groups in different locations should receive the same turnaround time. However, because of distribution of the completed work to different locations and possible time differences in actually returning output to the submitters, you might want to further separate this work—to give priority, for example, to jobs that must be distributed to locations in different time zones, where delays in turnaround time can have a significant effect on the users.

You should determine the factors listed in the above table for each level of each workload category. For example, determine the factors for batch, TSO, CICS, and IMS or other subsystems. Within each subsystem, determine the factors for further breakdowns of that workload type:

- TSO, CICS, and IMS divided into transaction types
- Batch work divided into batch windows
- VM users divided into accounting groups
- All categories divided by peak hours and off shifts

z/OS lets you associate transactions with a set of performance characteristics through performance groups. You can group your transactions into categories using performance group numbers. You assign performance groups through the installation control specification. For complete information, refer to the *z/OS Initialization and Tuning Guide*.

VM lets you group transactions and users through user classes. For z/VM data, user classes might be CMS users, service machines, and guests.

By assigning each distinct workload type to a separate performance group, you can use IBM Z Decision Support reports to obtain data on the processor, processor storage, and I/O activity for each category. Performance groups are the basic entity for z/OS workload measurements.

Measuring resource consumption

After you categorize your workload, measure the resources actually used by different workload categories. Essentially, you want to identify the amounts of processor, processor storage, and I/O resources required for each workload type. Table 1 on page 8 suggests resource requirements that might be measured.

Note: The System Performance Feature provides reports that give details on TSO, batch, and VM workloads. To measure resources used by CICS or IMS, you must have the IBM Z Decision Support CICS Performance feature or the IBM Z Decision Support IMS Performance feature installed and in use on your system.

Track the resource measurements for an extended period so that they encompass all variations in the workload. To identify exceptional conditions, track job-related and transaction-related data, both as an average and as a distribution. Such exceptions help you judge the effectiveness of your workload categories and the possible need for installation controls on the exceptional work. For example:

- Batch jobs, whose resource consumption places them in the top 10% of their class for resource use, might require reclassification.
- If the resource data varies widely for a particular job or user class (that is, if there is no distinct pattern) that job or user class might require redefinition or a more tolerant performance objective.

The resource data you collect further defines the workload categories. Understanding the resources consumed by each workload category, at each level, helps you judge the reasonableness of your objectives. Also from this data, you can set resource limits for each category, for example one processor minute for each job in job class X. After you understand the resource limits for each class, consider using installation controls and procedures to track and enforce these limits. On z/OS, you can enforce resource limits (such as elapsed time, number of I/Os, amount of processor storage, and resource allocation) using SMF exits. For complete information on the available SMF exits and how to code them, refer to the z/OS library.

To provide the user an incentive to classify work correctly on your z/OS system, you might want to produce exception reports that list each transaction that exceeds the resource limits for its class.

Setting objectives

After measuring resource use of the workload types, document the acceptable response times and activity loads for each category. Use these objectives as a baseline for discussing additional workload growth, resource management, and resource acquisition.

Examine the objectives and consider the user requirements and the priority of the work. If necessary, revise the objectives to meet the needs of your current system.

When setting response time objectives, consider any time the user sees that is not reflected in the measurement of the objective. For example, network delays are not reflected in RMF response-time measurements. If TSO trivial transactions require a 1-second response time, you could:

- Use 1 second as the objective for the time measured by RMF and then have a separate objective for the network delay (for example, 0.5 seconds).
- Use the total response time (1.5 seconds) as the objective.

The following figure shows a sample service-level objective for a z/OS system.

```
                          Average            Average
Workload               response time      activity load
type                     (seconds)    (transactions per second)
--------------------------------------------------------------

TSO trivial                 0.9              50.0

TSO nontrivial              2.5              15.0

Batch PRIME               280.0               2.0

Batch NIGHT               700.0               0.5
```

Figure 2. Sample service-level objective for z/OS

The following figure shows a sample service-level objective for a VM system.

```
Workload          Average            Average
type            response time      activity load
                  (seconds)     (transactions per second)
-----------------------------------------------------------------

Trivial             0.5                1.0

Nontrivial          1.5                0.5

QuickDsp            1.0                0.3
```

Figure 3. Sample service-level objective for VM

Gathering performance data

Most systems and subsystems provide data on how well they perform. To get the measurements you need for performance analysis, make sure the appropriate records are being written to the logs. For detailed information on the records you need for the System Performance Feature, refer to the *System Performance Feature Reference Volume I*.

A basic set of measurements provides enough data to focus on specific potential problem areas in the system. For example, you could use performance tracking worksheets to compare your performance measurements against your service-level objectives and focus attention on any service-level objectives that are not met.

Keep several samples available (for the times when performance problems are occurring and for the times when service-level objectives are being met) and document the workload at the time of each sample. Sample measurements help you judge whether the reported value of a specific measurement indicates a possible problem area in the system.

z/OS Workload Management

z/OS Workload Management provides a solution for managing workload distribution, workload balancing, and distributing resources to competing workloads. MVS Workload Management is the combined cooperation of various subsystems (such as APPC, CICS, IMS, JES, and TSO/E) with the z/OS workload manager (WLM) component.

The many monitoring and reporting products show different views of how well z/OS is managing work, or how well individual subsystems are managing work.

There are many performance and tuning externals in z/OS and in various subsystems, as well as throughout monitoring and reporting products. z/OS and the subsystems each have their own terminology for similar concepts, each have their own controls, and the controls are not coordinated.

Workload Management provides:

- A way to define z/OS externals and tune z/OS without having to specify low-level parameters. The focus is on setting performance goals for work, and letting the workload manager handle processing to meet the goals.
- New z/OS performance management externals in a service policy. These externals are expressed in terms commonly used in SLAs.
- Automatic work and resource management support that dynamically adapts as needed. It lets you meet your service goals for work while making efficient use of system resources.

Displaying and creating reports

IBM Z Decision Support helps you collect and report on performance data. The System Performance Feature includes many predefined reports that present the performance data in a clear and understandable format. By analyzing these reports, you can identify performance problems in your system and determine what needs to be modified in your system to solve those problems. Refer to the *System Performance Feature Reference Volume I* for a description of each report provided with the System Performance feature.

The System Performance Feature reports vary from general overview reports to very detailed ones. If these reports do not meet your needs, you can create your own reports using the IBM Z Decision Support reporting dialog. Refer to the *Guide to Reporting* for complete information on using the dialogs to create your own reports.

Analyzing reports

Each report that you display provides specific information about a resource in your system. You can display and analyze reports as you need them. As you become more familiar with the IBM Z Decision Support reports, you will probably choose a standard set of reports to run on a daily basis. These reports should give you an overview of your system performance and immediately show any problems.

This book presents a top-down approach to analyzing the reports for performance management. You should start with reports that present an overview of the system and let you see the big picture. If something in the big picture indicates that there may be a problem somewhere in the system, you must then display more detailed reports to identify the potential problem.

Identifying potential bottlenecks

Your main goal is to ensure that your system meets its service-level objectives. Because service is typically measured by response time, you should analyze the response times of the various applications and groups in your system. If you are not meeting your service-level objectives, examine IBM Z Decision Support reports to determine which system resource (processor, processor storage, or I/O) is causing the delay.

Analyze IBM Z Decision Support reports that show you the status of your processor, processor storage, and I/O subsystem. The other chapters in this book suggest reports to use to identify potential bottlenecks in your system. They explain the data presented and what to look at to find delays. From this data, you can identify the resource that delays the users. Use IBM Z Decision Support to display reports that show how that particular resource is used and by whom.

Analyzing detailed data

The rest of this book shows how you can use some of the predefined reports for the various components to identify potential problems in your system and to track down the cause and possible solution. The chapters describe some of the reports, what they show you, and when you should use them. For example, if you conclude that users are being delayed in the I/O subsystem, you can display reports that identify the DASD volumes with the highest response times.

Deciding where to get more resources

When you identify a resource shortage that causes delays in one of your key applications, examine your options for getting more of that resource.

You can get more resources for a particular workload by:

- Tuning the system to use existing resources more efficiently
- Reallocating them from a less important application
- Buying them

Evaluate the cost of getting more resources and estimate the effect that the extra resources will have on applications. Examine all of your options and present them to management. Consider how each option affects the business objectives.

Before considering any solution, ensure that it addresses any bottleneck identified in your system. There are no all-embracing solutions among performance suggestions. Each solution must be considered for your organization, its specific bottlenecks, and the effect of performance on your business objectives.

Verifying removal of performance constraints

After making changes, verify that the application is running as it should and that your changes have had the effect that you expected. Ensure that each change improved performance and that applications are now receiving the proper level of service.

Proper running of an application is not necessarily verification. The application might run better because of a change in user demand or changes other than the ones you made. Ensure that the changes you made did solve the problem, then quantify the improvement.

When all applications are running within their service-level objectives, wait for further violations of the service-level objectives.

Continuing the process

As stated before, performance management is an iterative process. When you analyze your performance, you find that either:

- You have met your service-level objectives.

  Wait and monitor performance according to plan.
- You have not met your service-level objectives.

  Investigate the cause of the delays and try to correct the problem.

Chapter 3. Analyzing z/OS performance

This chapter discusses some aspects of z/OS performance, and the service-level reporting and analysis that you can perform using z/OS-generated data. The chapter shows you how to use IBM Z Decision Support reports to analyze your system.

Once your organization has established service-level objectives, you need to see that those objectives are met consistently and to the best of the system's capabilities.

These major resources can affect user response times:

- Processor
- Processor storage
- I/O
- Network

If you are not meeting service-level objectives, investigate the use of these resources to identify areas of contention.

Note: This book covers the processor, processor storage, and I/O resources. To investigate network resources, use the IBM Z Decision Support Network Performance feature.

The discussion of each resource does not cover all the aspects of a computer system and environment that might influence that resource's measurement and analysis. What specifically affects a transaction's response time always depends on the circumstances at the time the event occurs or when the measurement sample is taken.

This chapter explains where the data comes from and how it is measured and recorded by the system. It then describes how to use the IBM Z Decision Support reports to analyze system performance.

By analyzing reports and data for each resource, you can identify whether a performance or service problem truly exists. If so, you can identify more closely the source of the problem and what you can adjust to overcome it.

This chapter presents a top-down approach to monitoring the performance of your z/OS system and its major resources. When viewing performance data, you should take a total systems perspective. Each unit of work produced by a system requires more than just one resource. The first step is to look at daily reports to see if you are meeting service-level objectives. If not, you must look more closely at the major resources of the computer system.

The three main host resources needed to complete a unit of work such as a transaction or job are:

- Processors
- Processor storage
- I/O

Table 2. Resource decisions

When making decisions about...	Examine...
Processors	Processor storage and DASD indicators
DASD	Processor and processor storage contention indicators
Processor storage	Paging and processor indicators

Before installing new hardware or software products, evaluate all three major system resources to determine what the contention indicators are showing for available capacity. Ask the question "If I buy more, what will I receive in return?" One of the major factors when trying to determine the benefits of change is seeing the total picture first.

Using RMF and SMF data

Most system-wide resource information is captured by the Resource Management Facility (RMF). RMF stores the data it collects in SMF records.The z/OS System (MVS) and z/OS Performance Management (MVSPM) components collect RMF data from the SMF log. The reports shown in this chapter include RMF and SMF data.

When using RMF data, you must consider how RMF measures the data. Within a time interval, RMF measures by exact count or by sampling. RMF makes an exact count measurement of a system indicator by computing the difference between its value at the beginning of an interval and its value at the end of the interval.

RMF makes a sampling measurement of a system indicator by recording its value at each cycle within the interval. A cycle is a subdivision of an interval. For example, each minute in an interval can be divided into sixty cycles that are 1 second long. If RMF is set up to take a sample every second during a 15-minute interval, RMF collects 900 samples during that interval. In most cases, this number of samples is sufficient.

You control the length of the interval and the cycle for the session. The default cycle is one second. Setting the cycle lower than one second causes more system overhead. As the cycle gets smaller, each sample is more likely to find the system performing the same functions as in the previous sample. So, the sample adds little additional information. The use of a smaller cycle value (while holding interval constant) should not be detrimental to the accuracy of the data, but any increase in accuracy might be of questionable benefit when compared with the system overhead that is introduced.

At the end of an interval, RMF gathers the data collected at each cycle and prepares to report the results. RMF stores the contents of each report in an SMF record. Each SMF record contains information similar to the contents of the corresponding formatted report. For each system activity that you select, RMF collects data and formats an SMF record to hold the data.

For more information on RMF and SMF, refer to the following books:

- *z/OS Resource Measurement Facility: User’s Guide*
- *z/OS System Management Facilities*

Analyzing overall performance

To analyze the performance of your system, look at the overall performance for a given day. You should analyze your system at least once a day to ensure that you are meeting your service-level objectives. IBM Z Decision Support provides reports that give an overview of the activity of your computer system. You can create the reports for any period (for example, once a day, twice a day, or as often as you need).

Most service-level objectives are based on response times and activity loads. You must ensure that your key applications are meeting those service-level objectives. The MVSPM Workload Response Time Components Overview report shows the average number of transactions processed during each shift for each workload type and the average response time for the transactions.

```
               MVSPM Workload Response Time Components Overview
                         System: 'MVS1'   Date: '1999-11-19'
                             Period: PERIOD_NAME


                      Trans-      Max      Average   CPU    Out   Paging
 Period     Workload  actions   response  response  time   ready   time
  name        type    (/min)     (sec)      (sec)   (sec)  (sec)  (sec)
 --------   --------  -------  ---------  --------  -----  -----  ------
 MORNING    BATCH         0.8      114.9      47.1    6.0    0.4    0.00
            TSOC          3.4       22.6       2.3    0.3    0.0    0.00
            TSOS          3.6        0.1       0.0    0.0    0.0    0.00

 NIGHT      BATCH         0.8      630.4     238.6   59.0    0.5    0.00
            TSOC         24.8      111.9       2.3    0.3    0.0    0.00
            TSOS         55.1        0.6       0.0    0.0    0.0    0.00

 PRIME      BATCH         1.4      465.8     131.4   31.6    2.3    0.04
            TSOC         59.9      363.0       2.7    0.6    0.1    0.00
            TSOS        148.5        0.7       0.1    0.0    0.0    0.00



               IBM Z Decision Support Report: MVSPM04
```

Figure 4. MVSPM Workload Response Time Components Overview report

Note: This report includes only workloads that do transaction recording.

For Db2 application response times, see Analyzing Db2 response time. For CICS response times, refer to the *CICS Performance Feature Guide and Reference*. For IMS response times, refer to the *IMS Performance Feature Guide and Reference*.

Use this report to verify that you are meeting your service-level objectives. First, look at the average response time to see if the values are acceptable. Then check that the workloads are not exceeding the transaction rate that they agreed to. Adding more transactions to the system can affect response time.

If a workload is not receiving the appropriate service level, you must determine what resource is delaying the work. Look at overall system performance to determine which system resource is constrained.

Analyzing performance at the system level

The MVSPM CPU and Processor Storage Activity Overview report summarizes the use of the processor and processor storage by hour for the specified day.

MVSPM CPU and Processor Storage Activity Overview
System: 'MVS1' Date:'1999-11-19'
Period: PERIOD_NAME

Period name	Time	CPU busy (%)	1 or more inready wait (%)	Average inready users	Average inwait users	Average outready users	Avg page delayed users	ES->CS pages (/sec)	Page migrate (/sec)	Avg TSO users	Avg lready users	Avg lwait users	Logical swaps (%)	Average waiting enqueue
MORNING	00:00	20.4	0.3	0.8	65.7	0.0	0.0	0.7	0.0	9.6	0.6	43.8	0.0	0.0
	01:00	4.6	0.2	0.1	65.0	0.0	0.0	0.0	0.0	9.0	0.3	44.6	0.0	0.0
	02:00	4.3	0.1	0.1	65.0	0.0	0.0	0.0	0.0	9.0	0.3	44.6	0.0	0.0
	03:00	8.0	0.2	0.3	65.9	0.0	0.0	0.1	0.0	9.0	0.3	43.7	0.0	0.0
	04:00	9.9	0.2	0.4	66.4	0.0	0.0	1.0	0.0	9.0	0.4	44.8	0.0	0.0
	05:00	6.3	0.2	0.2	65.7	0.0	0.0	1.8	0.0	9.4	0.5	46.0	0.0	0.0
	06:00	8.1	0.3	0.3	66.1	0.0	0.0	1.4	0.0	10.0	1.0	45.7	0.0	0.0
	07:00	8.6	0.3	0.3	66.5	0.0	0.0	6.0	0.0	15.7	0.6	47.7	0.0	0.0
PRIME	08:00	32.2	0.9	1.3	68.0	0.1	0.0	20.6	0.0	39.4	1.3	38.9	0.0	0.0
	09:00	35.5	2.8	1.5	72.0	0.4	0.0	125.3	5.3	78.8	1.3	33.4	1.3	0.1
	10:00	60.6	20.5	3.3	73.7	0.9	0.0	262.5	5.2	102.8	1.2	21.2	1.0	0.1
	11:00	50.2	12.8	2.7	76.0	0.8	0.0	239.1	3.7	102.7	1.2	19.8	0.6	0.0
	12:00	70.3	28.8	4.0	76.4	1.2	0.0	313.2	8.8	110.0	1.1	17.6	1.3	0.0
	13:00	40.7	8.9	2.2	74.9	1.5	0.0	294.3	7.8	117.0	1.5	23.4	1.6	0.1
	14:00	59.7	34.3	4.0	58.6	0.7	0.0	197.3	0.0	62.1	0.7	18.5	0.0	0.0
	15:00	41.0	11.7	2.3	60.2	0.4	0.0	181.2	2.0	92.1	0.7	25.3	0.1	0.3
	16:00	41.3	14.6	2.5	57.8	0.3	0.0	119.0	0.0	87.8	0.7	29.2	0.0	0.1
NIGHT	17:00	48.2	9.3	2.5	57.9	0.4	0.0	152.7	0.0	67.2	0.6	16.1	0.0	0.0
	18:00	27.6	1.1	1.2	56.7	0.0	0.0	22.6	0.0	50.3	0.7	32.5	0.0	0.0
	19:00	33.3	1.5	1.4	56.2	0.1	0.0	38.1	0.0	42.3	0.7	32.3	0.0	0.0
	20:00	24.8	0.7	1.1	57.2	0.0	0.0	23.7	0.0	35.3	0.8	36.6	0.0	0.0
	21:00	20.1	0.4	0.8	55.4	0.1	0.0	20.7	0.1	26.3	0.6	38.9	0.3	0.0
	22:00	13.7	0.2	0.5	53.2	0.0	0.0	2.1	0.0	18.2	0.5	46.1	0.0	0.0
	23:00	11.4	0.1	0.4	52.2	0.0	0.0	0.6	0.0	11.1	0.3	39.2	0.0	0.0

IBM Z Decision Support Report: MVSPM02

Figure 5. MVSPM CPU and Processor Storage Activity Overview report

These columns indicate how well the resources of your system are performing:

CPU busy (%)
Shows how busy the system was during the day. CPU busy indicates the capacity level of your system. It is not a contention indicator.

Average inready users
Shows the average number of address spaces ready and waiting to be dispatched. Compare this number to the number of processors on which work can be dispatched.

Average inwait users
Indicates the average number of address spaces waiting on another resource before it can continue processing.

Average outready users
Indicates the number of address spaces swapped out and waiting to be swapped in. These address spaces are ready to perform tasks, but are kept swapped out because of SRM values. A high number of out and ready users and a low percentage of 1 or more inready wait users indicate that the multiprogramming level (MPL) adjustment is incorrect or contention for another system resource is high. The MPL is the number of address spaces resident in central storage at any one time.

Avg page delayed users
Indicates the average number of address spaces delayed by paging. This value shows contention for processor storage.

ES ” CS pages (/sec)
Indicates storage movement from expanded to central storage. This value indicates central storage contention.

Page migrate (/sec)
Shows the storage movement from expanded to auxiliary storage. This value indicates expanded storage contention.

From this report, you can determine which resources you must analyze in more detail.

Analyzing performance at the workload level

Besides knowing how the system is doing, you must know which workloads are using the system and possibly causing or experiencing any contention. The MVSPM Workload Resource Utilization Overview report shows which workloads were using which resources during each shift.

```
                  MVSPM  Workload Resource Utilization Overview
                           System: 'MVS1'  Date: '2000-04-13'
                              Period: PERIOD_NAME


                            CPU   System  Page-    Cent   Exp    I/O     In     Out
Period     Workload  usage  usage   ins     stor   stor   rate    stor   ready
 name        type     (%)    (%)    (/sec)  (MB)   (MB)  (/sec)  users  users
--------   --------  -----  ------  ------  -----  -----  ------  -----  -----
MORNING    BATCH       6.9     1.7     0.0   17.8    0.0   13.33    1.4    0.0
           CICS        0.9     0.2     0.0   29.4    2.3    6.84   15.8    0.4
           STC        10.9     2.7     0.0  279.7   28.3   88.78   43.8    0.1
           SYSTEM      1.2     0.3     0.0    7.4    1.5    0.05    5.0    0.0
           TSOC        1.6     0.4     0.0    0.3    0.0    2.42    0.1    0.0
           TSOS        0.1     0.0     0.0    0.0    0.0    0.12    0.0    0.0
                     -----  ------  ------  -----  -----  ------  -----  -----
                 *    21.5     5.4     0.0  334.7   31.8  111.53   66.1    0.5

NIGHT      BATCH      57.2    14.3     0.0   29.0    0.0  179.81    3.4    0.0
           CICS        0.6     0.2     0.0   17.9    2.3    4.88    5.4    0.5
           STC         6.8     1.7     0.0  249.5   34.9    9.58   41.8    0.1
           SYSTEM      2.5     0.6     0.0    7.3    2.1    0.06    5.0    0.0
           TSOC       14.8     3.7     0.0    3.0    0.0   31.80    0.9    0.0
           TSOS        1.4     0.4     0.0    0.3    0.0    2.58    0.1    0.0
                     -----  ------  ------  -----  -----  ------  -----  -----
                 *    83.2    20.8     0.0  307.1   39.4  228.72   56.6    0.7

PRIME      BATCH      64.5    16.1     0.0   29.1    0.0  183.54    4.5    0.1
           CICS        2.7     0.7     0.2   24.8    2.3   21.58   13.8    1.1
           STC        18.7     4.7     1.2  213.8   50.7   47.94   42.0    0.4
           SYSTEM      4.2     1.1     0.0    6.8    2.1    0.14    4.8    0.0
           TSOC       58.6    14.6     0.2    8.6    0.0   57.58    2.1    0.1
           TSOS        3.9     1.0     0.1    0.7    0.0    7.04    0.3    0.0
                     -----  ------  ------  -----  -----  ------  -----  -----
                 *   152.6    38.2     1.8  283.7   54.1  317.82   67.6    1.7


               IBM Z Decision Support Report: MVSPM03
```

Figure 6. MVSPM Workload Resource Utilization Overview report

This report indicates which applications are using the most resources. You can use this report to identify the applications used most by your organization. You might find that an application uses too many resources and should have a lower priority. You can also identify batch work that should be run off prime shift.

The MVSPM System Resources by Workload Type report (see Figure 7 on page 20) presents the overall picture for the major system resources: processor, processor storage, and I/O. It is a graphical representation of the MVSPM Workload Resource Utilization Overview report. Use this report to see the leverage for resources by workload type.

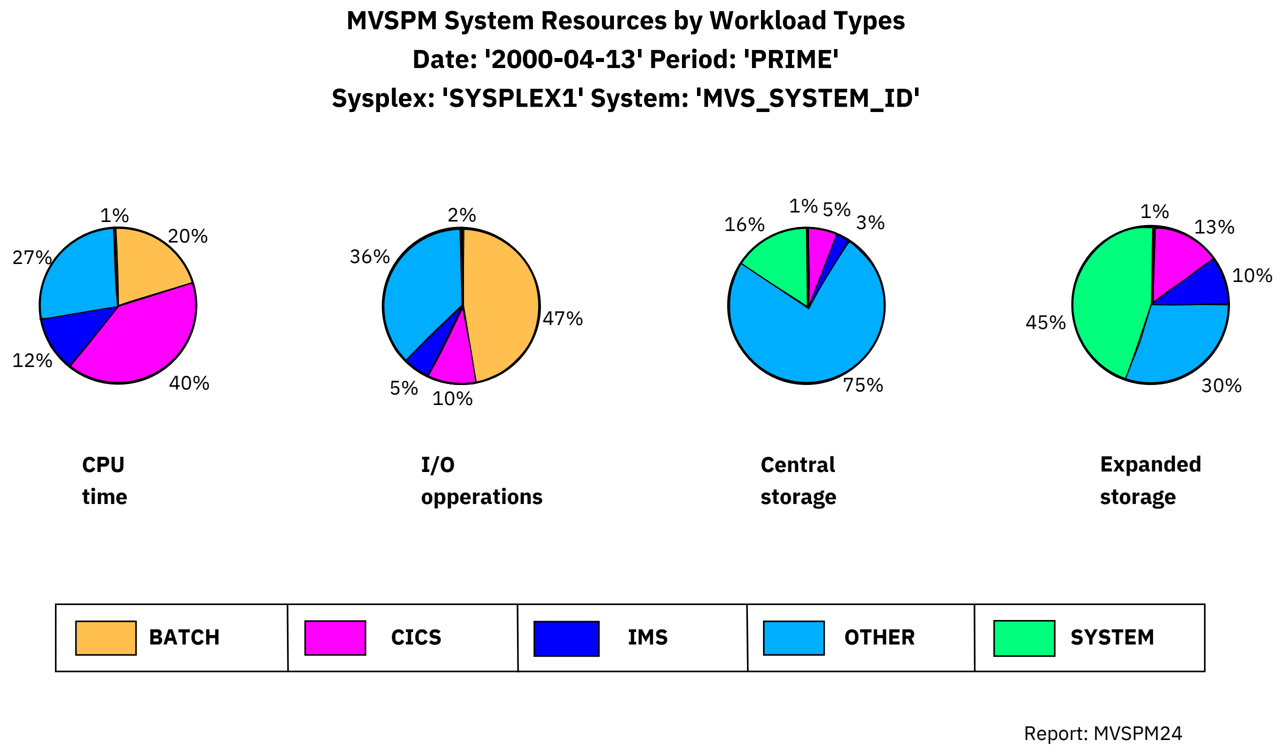

Figure 7. MVSPM System Resources by Workload Type report

CPU time represents the processor busy caused by each workload type. I/O operations represent the number of I/Os for each workload based on I/O service units. Central storage represents the percent of allocated central storage for each workload type. Expanded storage represents the percent of allocated expanded storage for each workload type. If your system is experiencing storage contention, use this report to easily identify the workloads with the most processor storage. This report shows relative shares (that is, the percentage of the resource used, and not the total amount) of each resource.

Workload Manager

With workload management, you define performance goals and assign a business importance to each goal. You define the goals for work in business terms, and the system decides how much resource, such as CPU and storage, should be given to meet the goal.

An organization must know what it is expected to accomplish in the form of performance goals, as well as how important it is to the business that the performance goals be achieved. With workload management, you define performance goals for work, and the system matches resources to the work to meet those goals. The system constantly monitors and adapts processing to meet the goals. Reporting reflects whether the system is meeting its goals. Figure 8 on page 21 is an example of reporting the actual values against the goals specified in the workload management service definitions:

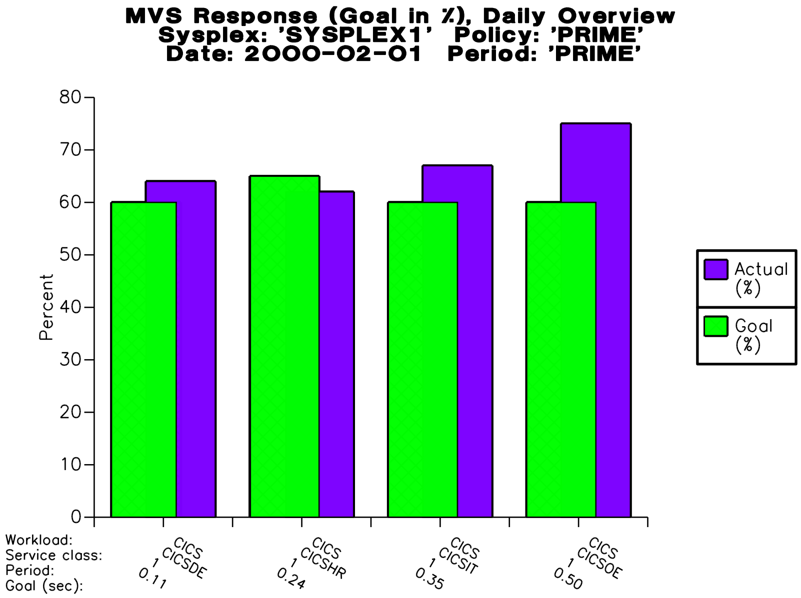

Figure 8. MVS Response (Goal in %) Daily Overview report

For each system, workload management handles the system resources. Workload management coordinates and shares performance information across the sysplex. How well workload management manages one system is based on how well the other systems are meeting goals. If there is contention for resources, workload management takes action based on the importance of the work and how well the goals are being met.

Analyzing processor resources

To identify and reduce nonproductive processor time, divide the processor time into categories. Measure how much time is used in each category and identify those categories that might be reduced or eliminated for a positive effect on performance without an adverse effect on system functions. To do this, you must first understand how the operating system measures and reports processor usage.

Measuring processor time

Processor time is a measure of the time that a job or task controls the processor. When a job receives control of the processor, the operating system stores the current time-of-day (TOD) clock value. When the job loses control of the processor, z/OS subtracts the current TOD clock value from the start value, and records the results as the duration of the job.

The primary sources of z/OS processor usage are:

SMF type 70 record

This record provides total system-wide processor usage for a recording interval. An SMF type 70 record includes the number of seconds in the interval that the processor was in wait state (that is, waiting to execute instructions). When creating this record, RMF does not use sampling. RMF uses the hardware processor TOD clock as its source for the processor wait time for the measured interval.

The processor active time is the duration of the interval minus the wait time. The ratio of processor active time to interval duration, expressed as a percentage, is the common measure of percent processor busy.

Many processor complexes consist of several individual processors that share system resources and work. The operating system that manages the work that is using these processors has access to all processors for both work and measurement purposes. RMF supports all processors that are physically or logically made available to the operating system. RMF stores the wait times for each processor in the same SMF type 70 record at the end of each measurement interval.

If a processor complex includes four individual processors all physically available to the z/OS system, then four measures of processor wait time will be found in each record at the end of each interval.

A processor can be logically made available through the Processor Resource/Systems Manager (PR/SM™).

For an LPAR partition with wait completion enabled, wait time is subtracted from logical processor dispatch time. For an LPAR partition with wait completion not enabled, wait time is not used. Logical processor dispatch time is used.

SMF type 72 record

This record provides processor usage data for a performance group for the recording interval. You can subtract the sum of the performance group processor times from the total processor active time derived from an SMF type 70 record to measure unassigned processor time. This unassigned time is the processor active time that z/OS does not directly associate with any performance group.

Processor time includes:

TCB time

The time a job executes under a task control block (TCB).

SRB time

The time a job executes under a service control block (SRB).

RCT time

The time used for a region control task (RCT).

IIT time

The time used for processing I/O interrupts (IIT).

HST time

The time used for hiperspace service (HST).

z/OS records these processor times separately in SMF type 72 records.

z/OS records the SRB and TCB times as service units. z/OS converts each of the times into service units by multiplying the time by an SRM processor-dependent constant. For example, z/OS converts the time a job executes under a TCB into TCB service units by multiplying the TCB time by an SRM constant that is processor-dependent. RMF adds the value to the TCB service unit field of the SMF type 72 record of the performance group that the job belongs to. z/OS converts the SRB time in the same way.

z/OS records RCT time in 0.001 024 seconds, and IIT and HST in 0.000 001 seconds.

Starting from z/OS version 1.6, the SMF type 72 record also contains:

IFA Time

The CPU time spent on the Integrated Facility for Application (IFA) coprocessor. IFA Time, in hundredths of a second, is present as a separate field, but it is also added to the TCB time if the IFA eligible work runs on the standard CPs.

Starting from z/OS version 1.8, the SMF type 72 record also contains:

IIP Time

The CPU time spent on the Integrated Information Processor (IIP) processor. IIP Time, in hundredths of a second, is present as a separate field, but it is also added to the TCB time if the IIP eligible work runs on the standard CPs.

Report performance groups are also recorded in SMF type 72 records. Report performance groups report statistics on transactions. Depending on the number of ways statistics are collected, there can be more than one report performance group for a transaction. Exclude report performance groups when summaries are produced to avoid counting workload more than once.

The SMF type 72 record contains subtypes 1 and 2 when z/OS is running in compatibility mode, and types 3 and 4 when z/OS is running in goal mode. Subtypes 1 and 2 contain information about performance groups, and subtypes 3 and 4 contain information about service classes.

SMF type 30 records

This record provides information about processor usage at the job level. A type 30 record contains operation information, such as the job and step start and end times, step processor time, step termination status, device allocation start time, and problem program start time, for a job or step.

In this record, z/OS records processor time as TCB, SRB, RCT, IIP (processing I/O interrupts, same as IIT in SMF type 72 record), HPT (same as HST in SMF type 72 record), and IFA & IIP (CPU time spent on Integrated Information Processors), in hundredths of a second. This record also provides initiator processor time.

Each job or step is associated with a performance group number. You cannot compare data in SMF type 30 records to SMF type 72 records, because they do not show processor usage for the same time period. SMF type 30, subtypes 4 and 5, records show total CPU used for the step and job.

The following figure illustrates the division of processor times and the overlap of the different records.

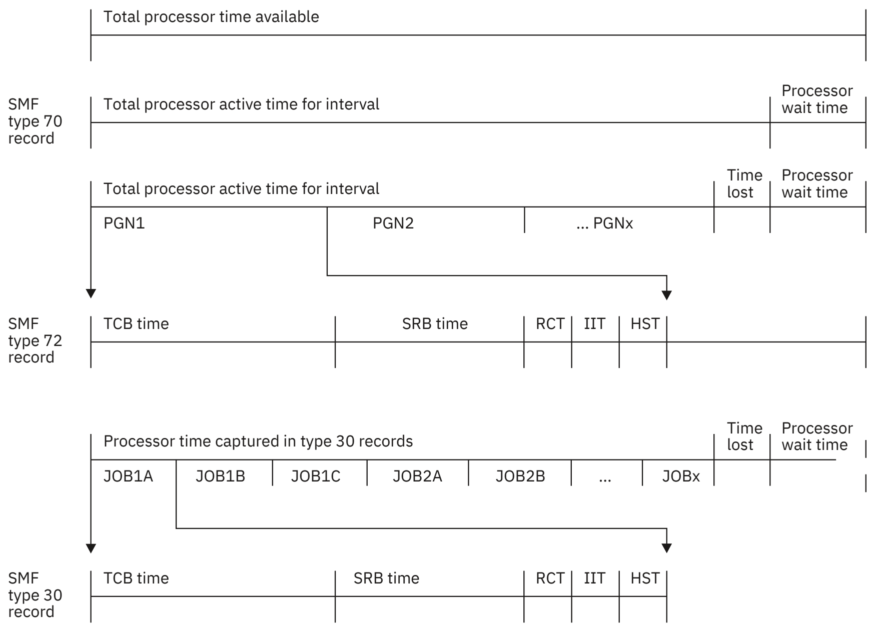

Figure 9. Processor times

Understanding capture ratio

The above figure shows that not all of the processor time captured in SMF type 70 records is captured in SMF type 72 records. For example, the processor time consumed in the termination of the job or step and the time used to write the job or step record to SMF is not included in the SMF type 72 record, but is included in the SMF type 70 record.

The capture ratio is the ratio of accounted-for time in the SMF type 72 record to the total processor time measured in SMF type 70 record. You compute the capture ratio by comparing the data in the SMF type 72 record and the SMF type 70 record.

Several things can make the capture ratio vary. For example, a storage-constrained system will show a lower capture ratio. A small RMF cycle time (below 1 second), large dispatching queues, and implementation of SLIP PER trap activity cause variations in the capture ratio. What can cause the capture ratio to vary depends on the release of z/OS being used. Refer to *System Management Facilities (SMF)* for the release of z/OS that you are using.

Analyzing processor usage

Processor busy alone does not sufficiently indicate processor capacity requirements. You should investigate high processor busy to determine which workloads are causing it. Having a processor at 100% busy indicates that processor capacity is at a maximum, and adding workload will not provide more answers. Lower priority work waits longer for processor cycles. This may or may not be acceptable, depending on your service-level and response-time requirements.

Processor performance issues occur when a lack of processor cycles causes service-level objectives to be missed. This can occur when:

- High priority work dominates the processor
- An application runs at the speed of a processor
- A lot of work arrives at the same time (workload arrival patterns)

You must analyze low processor busy to determine if another system resource is causing response-time delays. Processor contention is demonstrated by showing processor busy indicators and latent demand indicators.

The average processor busy for the shift is the average busy of all the processors in the system. In a multiprocessor system, if each processor was 70% busy for the interval, then the average system busy is 70%.

In an LPAR environment, the processor busy represents the logical busy per processor based on the logical dispatch time.

IBM Z Decision Support provides a number of reports that show CPU usage. The MVSPM Average CPU Busy, Hourly Trend report presents the hourly usage trend for your system. This report includes all systems for which data is collected.

```
                       Sysplex: 'SVPLEX5 '  Date: 2006-05-13
                                    Period: All
                                Processor type: CP

                         MVS_system
                         <- S5C -->      < TOTAL ->
Time
-----                  --------------  ---------------
13:00                        9.7            9.7
14:00                       74.5           74.5
                     ==============  ===============
                            42.1           42.1


   IBM Z Decision Support: MVSPM07
```

Figure 10. MVSPM Average CPU Busy, Hourly Trend report

This report presents the overall picture of your processor usage for the day you specified and for the processor type (CP, IFA or IIP) you specified when creating the report. Processor busy patterns are an important part of the performance road map. Processor contention problems occur at peak periods. You can easily identify which times and which system to investigate for more detail. You can investigate the busy systems to look for delays caused by insufficient processor capacity. You should also look at the systems that seem to be underused. Under utilization is not always caused by a lack of work. Sometimes the processor cycles cannot be used because of a resource contention, such as storage constraints and delays in transferring data. At times, you must look at the impact of other systems, particularly in the area of DASD sharing or system consolidation.

The arrival time of work on different systems is an important factor in processor load balancing. When planning to use LPAR as a vehicle for processor load balancing, this factor helps determine if processor balancing will be beneficial.

Analyzing latent demand

Other indicators can help you understand if there are workload delays. The MVSPM CPU Busy and Other Indicators, Hourly Trend report shows some of the indicators.

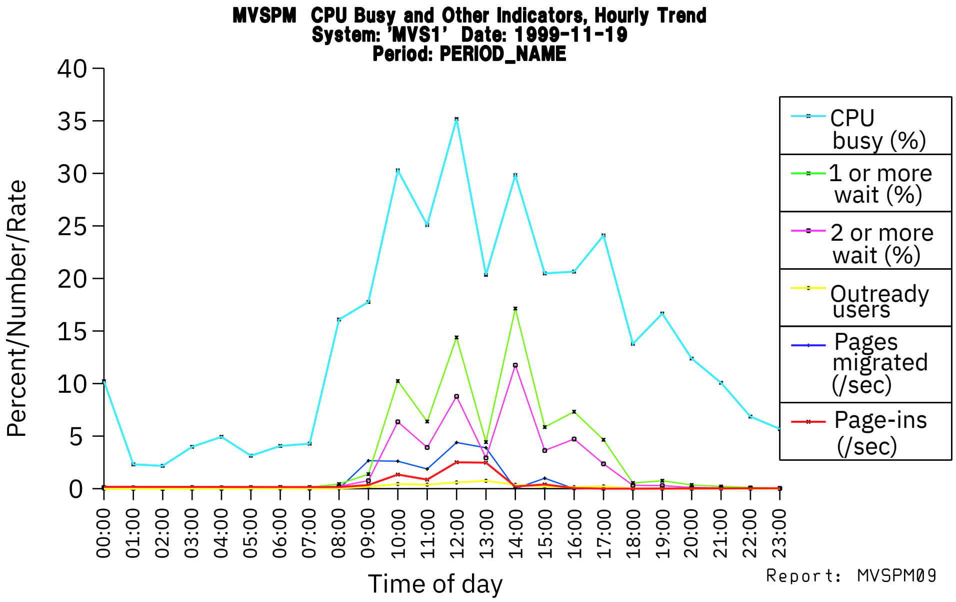

Figure 11. MVSPM CPU Busy and Other Indicators, Hourly Trend report

These indicators appear in the report:

1 or more wait (%)
Indicates the percentage of time that all processors were dispatched and there was at least one address space waiting to be dispatched.

2 or more wait (%)
Indicates that at least 2 or more address spaces are waiting to be dispatched.

Outready users
Indicates the number of address spaces swapped out and waiting to be swapped in. These address spaces are ready to perform tasks, but are kept swapped out because of System Resource Manager (SRM) values.

Pages migrated (/sec)
Indicates the rate at which pages from expanded storage are being migrated to auxiliary storage. This value indicates contention for expanded storage.

Page-ins (/sec)
Indicates the total non-virtual input output (VIO) page rate. This value shows the contention for central storage.

This report is a cornerstone when looking for latent demand. It identifies whether work is out and waiting or whether work is being delayed due to storage contention.

Latent demand is caused by having more in and ready users than processor cycles. The number of users waiting and the number of users out and ready indicate the level of latent demand. High numbers indicate processor contention. A high number of out and ready users and a low percentage of 1 or more wait users indicate that the MPL adjustment is incorrect or contention for another system resource is high.

The impact of latent demand is determined by which workloads are waiting for a processor and whether those workloads are meeting their service-level objectives. Latent demand for processor cycles implies that the applications have the necessary processor storage and I/O to execute, but are being delayed due to a lack of processor cycles. In some batch environments, latent demand for processor cycles may be good. This latent demand can be a sign of unconstrained I/O and processor storage resources.

The best solution depends on the workloads causing the latent demand for the processor. Swappable versus nonswappable workloads require different management techniques. For batch and TSO workloads, use the installation performance specification (IPS) to manage throughput by creating multiple performance group numbers and periods for low- and high-priority work. In an online environment, where there is no low-priority work that can be delayed, you may have to change your service-level objectives, or manage the resources (for example, change priorities of workload or move some work to a different processor) to meet the service-level objectives or consider installing additional processor capacity.

Analyzing processor busy by workload type provides insight into which workloads are waiting for processor cycles. You can have latent demand even when the average processor busy is not at 100%. For example, 70% processor busy means that 70% of the time the processor is 100% busy. Latent demand is a queue length of additional work that is ready to be dispatched on the processor. It may result from an overcommittment of processor resources or a skewed arrival pattern of processor activity.

To understand the times when latent demand is low, you must analyze the other resource contention indicators to ensure that processor storage or I/O bottlenecks are not preventing work from being dispatched. You may be able to off-load peak hour processor requirements if your prime shift demands and service requirements allow it.

This report includes two fields (page-ins and pages migrated) that show processor storage contention. Page-ins is the demand page-in rate from auxiliary storage and pages migrated is the page rate from expanded storage to auxiliary storage. Additional reports can help determine if the contention is for central or expanded storage. For more information on analyzing your processor storage resources, see Analyzing processor storage resources.

Analyzing who is using the processor

When you determine that there is contention for the processor, you must determine which workloads are using it. The MVSPM CPU Busy by Workload Types, Hourly Trend report shows the percent of the system used by each workload type.

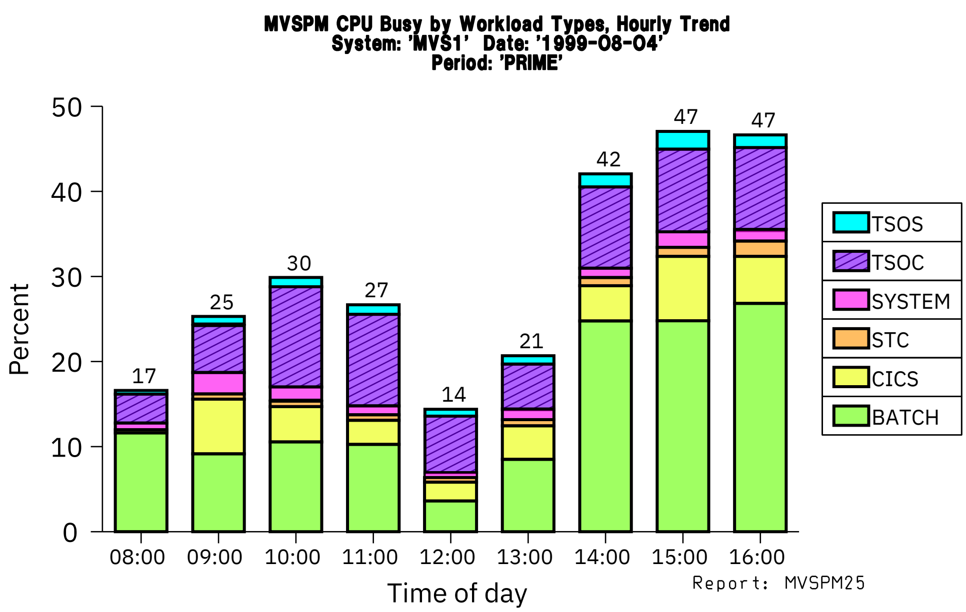

Figure 12. MVSPM CPU Busy by Workload Type, Hourly Trend report

The values in this report are normalized to the percent of system so you can compare this report to the MVSPM Total CPU Busy report. Looking at processor busy by hour by workload type shows the different processor workload patterns. Work is dispatched to the processors based on a priority scheme. When the processor usage runs close to 100%, the lowest-priority work waits. You must understand how long this work can wait before service-level objectives are missed. Operating the system close to maximum capacity requires more careful management to ensure that all business needs are achieved.

A specific workload type often consists of multiple performance groups. The MVSPM CPU Busy by Workload Performance Group Numbers (PGNs), Hourly Trend report shows how much processor time is used by each performance group.

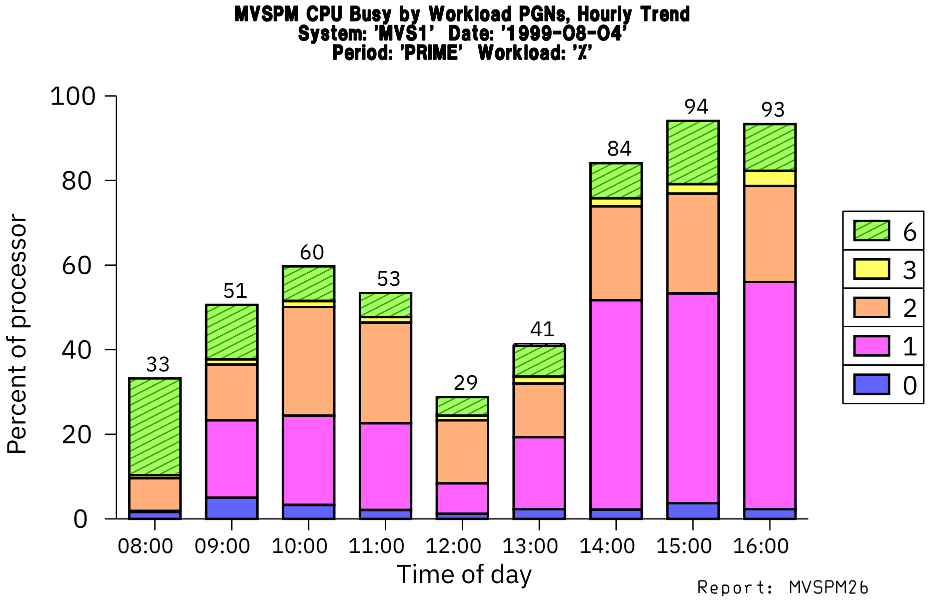

Figure 13. MVSPM CPU Busy by Workload PGNs, Hourly Trend report

High processor busy for a performance group may indicate performance or capacity contention. Find out if one address space is causing this busy or if multiple address spaces are causing it. This detail will help you understand whether you need faster or more processors. When the processor capacity exceeds the available processor speed, the next step may be to split or clone an application.

Analyzing LPAR and PR/SM processor use

The MVSPM CPU Busy Profile Shared LPARs, Hourly Trend report shows how much of the shared physical system is used by all of the nondedicated logical partitions (LPARs).

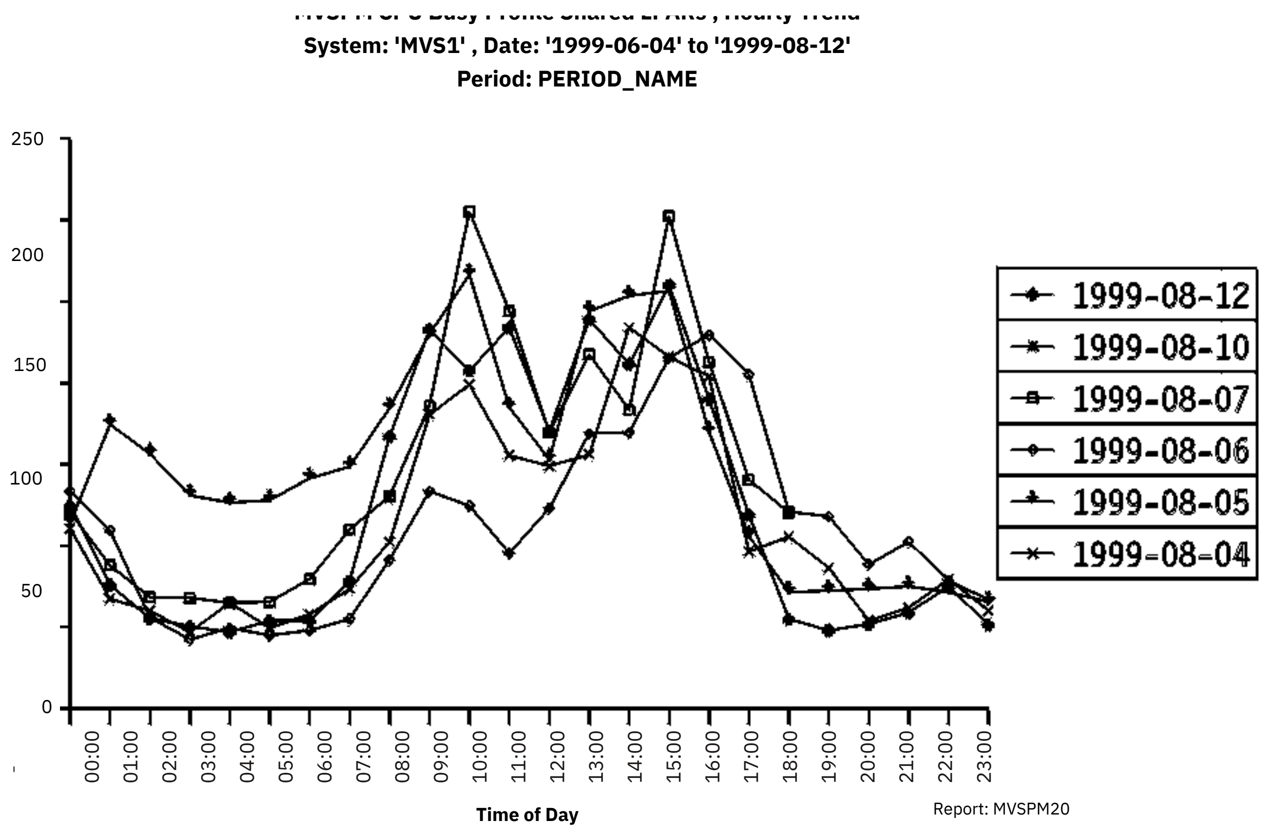

Figure 14. MVSPM CPU Busy Profile Shared LPARs, Hourly Trend report

This report shows the processor busy profile for multiple days to ensure the day chosen for additional reports is a representative day. The total processor busy shown here is based on the portion of the system that is being shared by nondedicated partitions. It does not include the processor busy caused by dedicated partitions, because this processing time is recorded separately. Additional reports break this processor busy out by logical partition.

Notes:

1. If the global performance data control (CONTROL PRF) parameter has been set to N for this system on the Logical Partition Security Parameters Frame (LPSEC), this system can view only the processor usage data for its logical partition. There is no value in the RMF data regarding the setting of this parameter. Therefore, the reports showing LPAR data for all partitions will be incomplete and not reflect the true resource consumption.
2. An LPAR defined with one dedicated processor on a four-processor system restricts 25% of the physical processor from being shared. This restriction would not be reflected in this report. The report reflects only how much of the remaining 75% is being used.

The MVSPM CPU Busy by Shared LPARs, Hourly Trend report shows how much of the available system processor capacity each nondedicated logical partition is using.

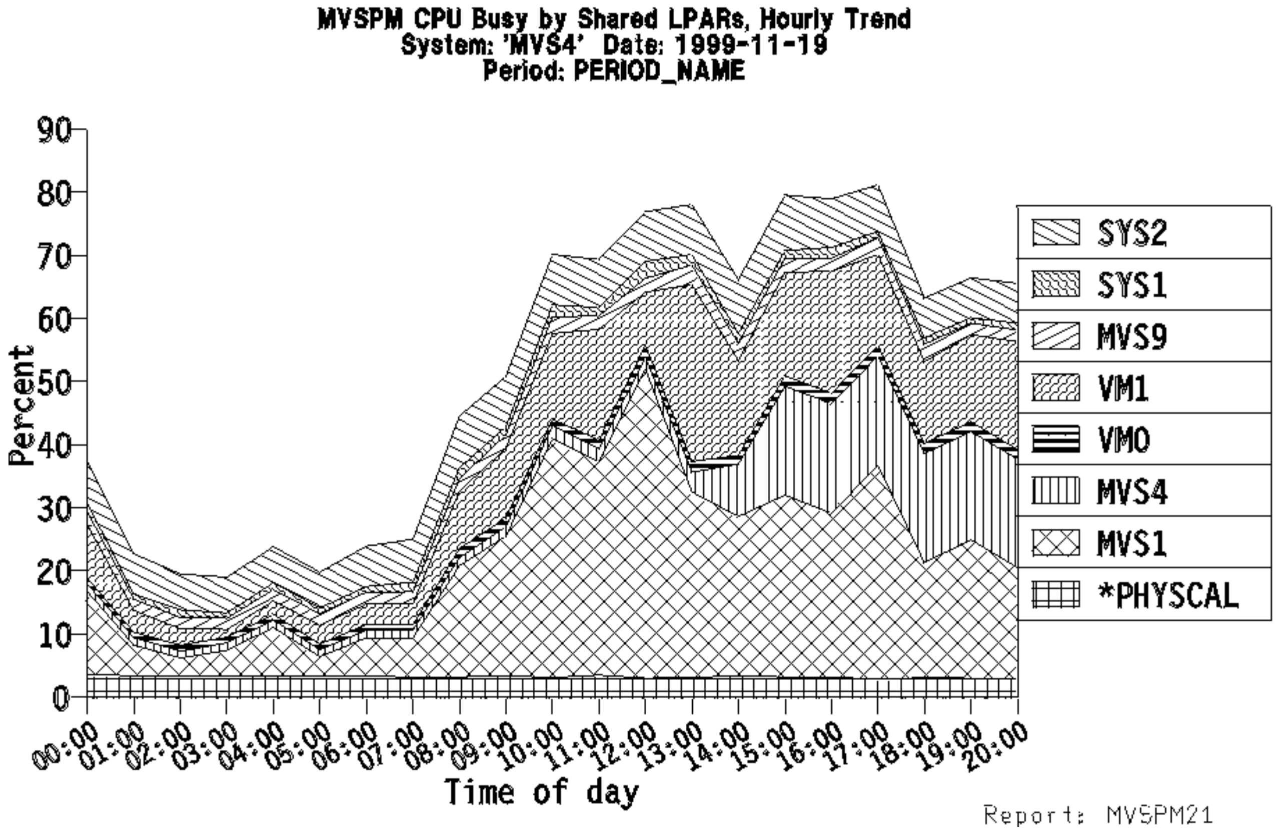

Figure 15. MVSPM CPU Busy by Shared LPARs, Hourly Trend report

Dedicated processors are not included as available processor capacity. The processor busy for each logical partition represents a part of the total available processor resource and not the percent of a logical partition's processor resource.

The MVSPM Percent of Share Used by LPARs, Hourly Trend report shows how much of the allowed share each logical partition is using and is based on the assigned LPAR weights.

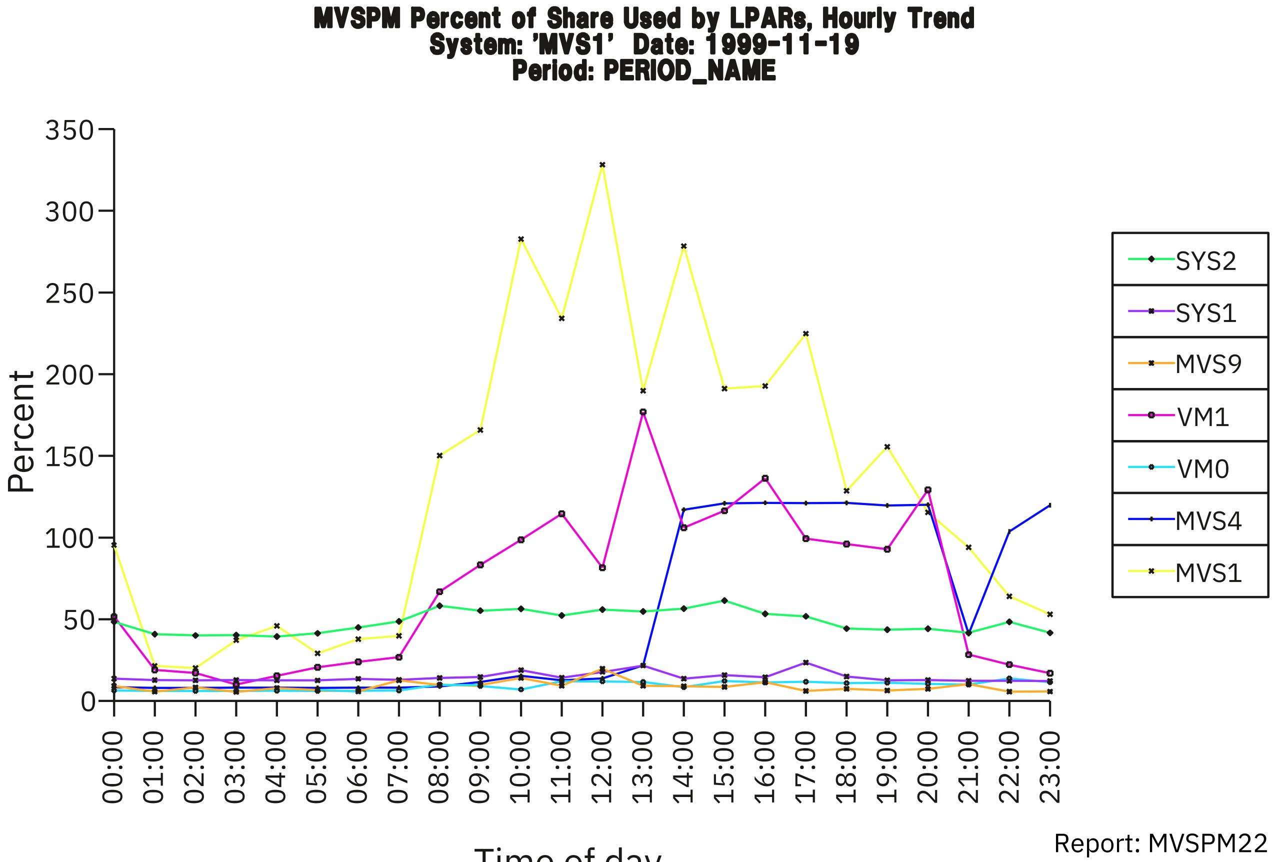

Figure 16. MVSPM Percent of Share Used by LPARs report

You can use this report to determine if the weights have been set properly or if a specific partition consistently exceeds its allotted share. When a partition exceeds its share, it uses the available processor cycles not being used by the other nondedicated partitions. When the partitions require more processor resources, PR/SM reduces the partition that is exceeding its share first. For capacity and performance reasons, you must understand the impact this reduction will have on meeting workload objectives.

LPAR enhancements have provided the capability to isolate the PR/SM processor time. If the report shows a logical partition called PHYSCAL, then this feature is installed and the PR/SM processor busy time can be isolated from the logical partition time. If the report does not contain a logical partition called PHYSCAL, then the PR/SM processor busy time cannot be isolated. PHYSCAL is a special name and cannot be used as an LPAR name.

Analyzing processor storage resources

In the view of a computer system, a user is represented by a collection of control blocks, programs, and working space. When the user enters a transaction, the required application program and data must be retrieved by the system. The location of the application programs and data sets affects both the amount of work the system must do to process the transaction and the response time observed by the user. The performance from both a system view and user view is determined by the speed of access to programs and data.

To be processed by the processor, an instruction or data item must reside in the high-speed processor cache that is integrated with the processor. If the item is not located in the cache, the system must retrieve it from processor (central or expanded) storage. The processor waits while the item is moved into the high-speed processor cache. Because the processor waits, the data transfer is called synchronous. After the item arrives in the cache, the processor continues processing the task.

If the item is not found in processor storage, it must be retrieved from an I/O storage device, such as DASD, through an asynchronous data transfer. The operating system suspends the task being processed and schedules the I/O to retrieve the item. When the I/O completes, the interrupt is processed and the original task is eventually dispatched to run on a processor. This asynchronous retrieval increases the response time of the task.

The figure below illustrates the various levels of processor storage.

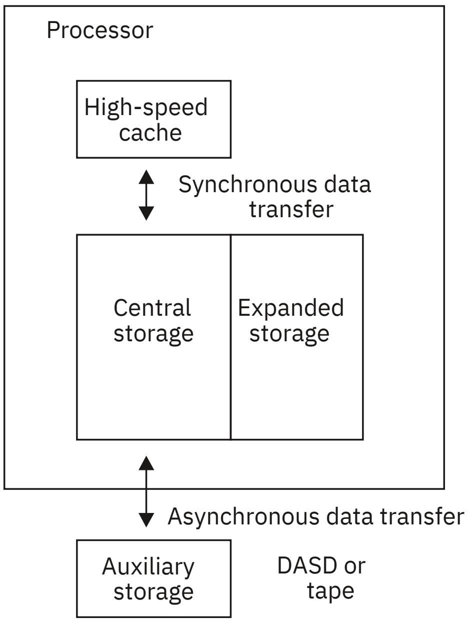

Figure 17. Processor storage

Given this process, having the application and data in processor storage reduces response time and supports consistent response times.

z/OS manages the contents of processor storage in a global least-recently-used (LRU) manner across central and expanded storage. The most frequently referenced virtual storage pages are kept in central storage, the next most frequently referenced pages are kept in expanded storage, and the rest reside in auxiliary storage (page data sets on I/O devices).

Measuring processor storage activity

Data pertaining to processor storage activity is stored in the RMF paging activity record (SMF type 71 record). The SMF type 71 records contain information about the demands made on the system paging facilities and the usage of central and expanded storage during the reporting interval. RMF records the minimum, maximum, and average statistics for both fixed and pageable frames in the common storage area (CSA), link pack area (LPA), system queue area (SQA), local system queue area (LSQA), and private area. It also records the minimum, maximum, and average number of each type of frame in central and expanded storage.

The paging information recorded in SMF type 71 records includes:

- Number of pages swapped in or swapped out
- Number of VIO page-ins and page-outs
- Number of non-VIO page-ins and page-outs
- Total number of pages moved from central to expanded storage
- Total number of pages moved from expanded to central storage
- Total number of pages migrated from expanded storage to auxiliary storage
- Minimum, maximum, and average high unreferenced interval count (UIC)
- Minimum, maximum, and average migration age
- Number of hiperspace page-ins or page-outs
- Number of blocked pages paged in or paged out

- Statistics on shared page groups

SMF type 30 records contain similar paging information on an address space level.

Paging information is important in resource analysis because it identifies whether central or expanded storage is limiting performance.

Analyzing page movement

Excessive paging affects the processor usage, because jobs in central storage that should be using processor time are waiting for data to be moved into central storage. The more jobs you have waiting on paging, the lower your processor usage. Drops in processor usage may indicate storage constraints.

The processor used for storage management runs at a higher priority (performance group 0) than applications. Therefore, this movement may cause additional latent demand when the processor is near capacity and processor storage also is constrained.

To minimize the delay caused by a page-in, avoid the creation of contention on the I/O path to the page data sets.

IBM Z Decision Support includes several predefined reports that show the activity of the paging subsystem. These vary from monthly overview reports to hourly trends of system storage paging rates. These reports can help you determine if too much time is being spent on paging.

Moving from central to expanded storage

The page rate from central to expanded storage is an indicator of how much page movement is occurring. To determine the impact, you must understand the cause of this page movement. Is it because of paging, or swapping, or hiperspace activity?

Note: Hiperspace™ activity that uses the MOVE PAGE instruction is not recorded.

Moving from expanded to central storage

Storage movement from expanded storage to central storage indicates contention for central storage. Pages are being moved out of central storage too quickly. The page rate from expanded to central storage indicates how serious the central storage constraint is.

Moving from expanded to auxiliary storage

Storage movement from expanded to auxiliary storage (pages migrated) indicates that you are using all of your expanded storage. You must find out how many of these pages are being paged back in as page faults (page-ins).

Migration rate indicates how effectively you are using expanded storage. If the page faults are low, then this movement from expanded to auxiliary storage is not an issue. A low migration rate means that pages that are paged out to expanded storage are referenced quickly enough to avoid migration to an external I/O device. Low migration rates mean that the system is avoiding a high paging overhead.

A high migration rate, in contrast, might mean contention for limited expanded storage or that the expanded storage criteria table does not match your workload.

When z/OS moves pages from expanded to auxiliary storage, it must first move the data from expanded to central, and then to auxiliary. So, page migration can cause CPU delay.

Analyzing the demand page-in rate

If excessive paging is occurring, you must determine when it happens and which workloads are experiencing it. The MVSPM Page-ins by Workload Type, Hourly Trend report shows which workloads are experiencing the page faults.

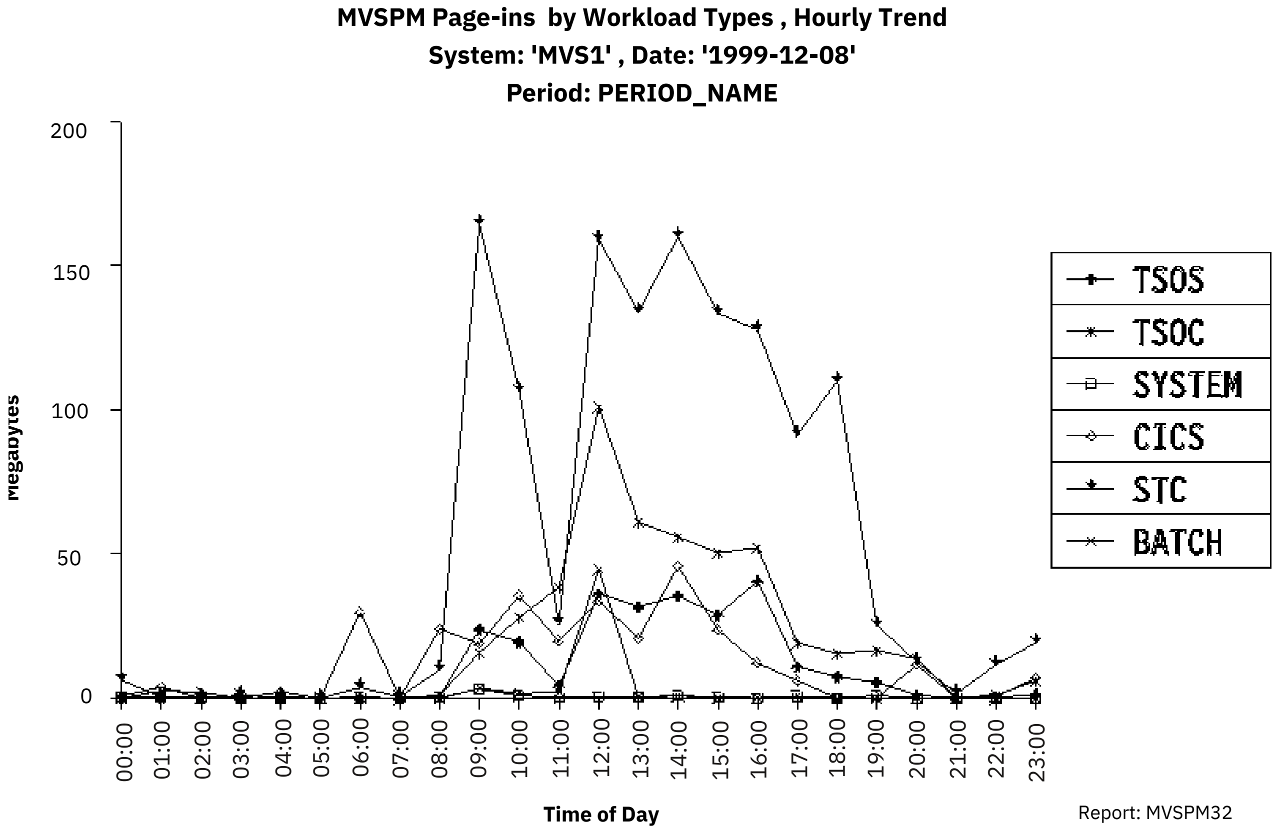

Figure 18. MVSPM Page-ins by Workload Type, Hourly Trend report

fig>

The highest workload delay caused by storage contention occurs when the demand page-in rate is highest.

If high levels of demand paging exist, you might want to further break out this work by performance group to determine which PGN within a specific workload type is experiencing this paging. This analysis shows you the specific type of work experiencing demand page-ins.

You can obtain more information by looking at page I/O response time. To find the impact of paging on a particular workload type, multiply the I/O response time by the number of page faults for that workload. The I/O response time, multiplied by the number of page faults per second per workload type, shows the impact this paging is causing to each workload type.

For workloads that offer transaction reporting, the portion of the response time caused by paging can be estimated by calculating the number of page-ins per transaction and multiplying by the I/O response time. For example, if a workload did 40 page-ins per second and 10 transactions per second, and the average page response time was 30 milliseconds, then:

```
40 page-ins/sec    * 30 milliseconds = 120 milliseconds
     10 transactions/sec
```

120 milliseconds of that workload’s response time was due to paging. If you had subsecond response time, then the time caused by paging represents a minimum of 12% of the response time.

Analyzing page data sets

You can define two kinds of paging data sets in z/OS page data sets and swap data sets. You must define page data sets. Swap data sets are optional. If you have swappable workloads on your system, you can still run with only page data set. You can also have both nonswappable and swappable workloads use swap data sets to ensure that swap activity (usually numerous pages in a swap set) does not interfere with demand paging (usually one page at a time) from online workloads (nonswappable).

The structure of page and swap data sets is different and so is the format of the requests to these data sets. The nomenclature is somewhat misleading: while all nonswap paging goes to page data sets, part of swap paging goes to swap data sets, and part of it goes to page data sets. Thus a one-to-one identification of paging category (that is, paging or swapping) and paging data set type (page or swap) is not possible. The functional division between swap and nonswap paging does not totally correspond to the division between swap and paging data sets.

The MVSPM Page Data Set Type Response Time, Hourly Trend report shows the I/O response time by the hour for each page data set type.

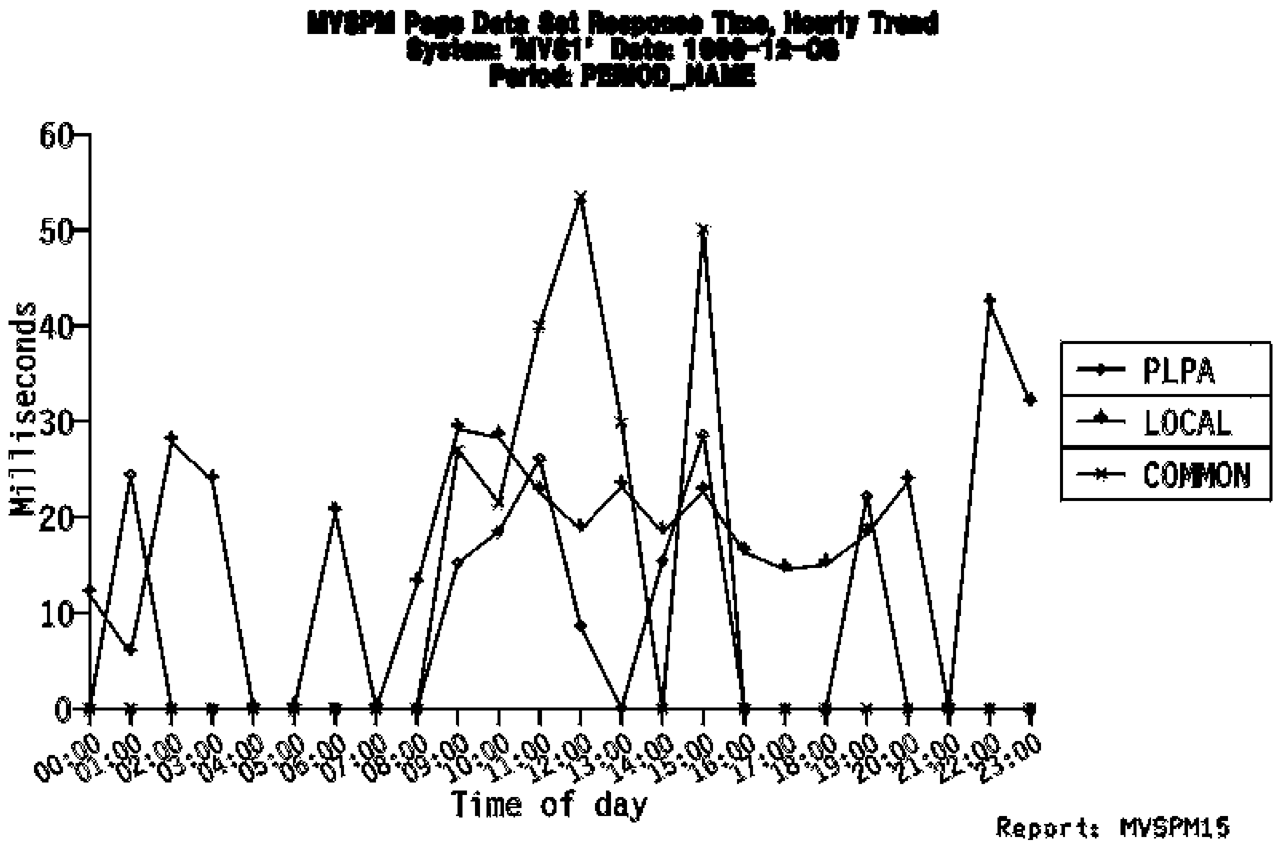

Figure 19. MVSPM Page Data Set Type Response Time, Hourly Trend report

The I/O response time is the average for all devices being used for each page type. Many users have eliminated swap data sets and converted to local page data sets only. This conversion should not impact performance as long as the central storage contention is low. If a high level of storage contention causes a high demand page rate and the swap activity is forced to the same local page data sets, performance delays to the workloads experiencing page faults may occur. If the swap activity is low, there is no impact on the local page data sets. Much of this activity is accommodated by expanded storage, if available.

Analyzing block paging

The block paging function of z/OS uses the sequential or repeatable data reference patterns of many numerically intensive computing applications. It reduces repeated page faults by packaging together pages that are expected to be referenced together in the future and, at page fault time, loading them into central storage as a block rather than one at a time. This function can markedly improve the elongated elapsed times suffered by numerically intensive computing applications when their data has spilled to auxiliary paging devices.

Analyzing who is using processor storage

You can further investigate peak hours of central storage usage. The MVS CS by Workload Type, Peak Hour, Daily Trend report shows which applications were using central storage during a certain hour across multiple days. This report shows if the usage is consistent over time.

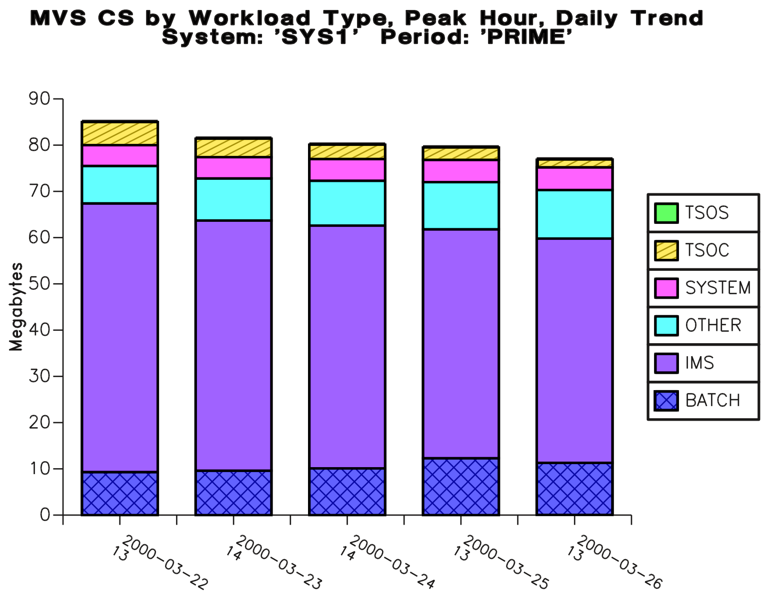

Figure 20. MVS CS by Workload Type, Peak Hour, Daily Trend report

The MVS ES by Workload Type, Monthly Trend report shows expanded storage usage by application. Use this report to see which applications are currently using expanded storage.

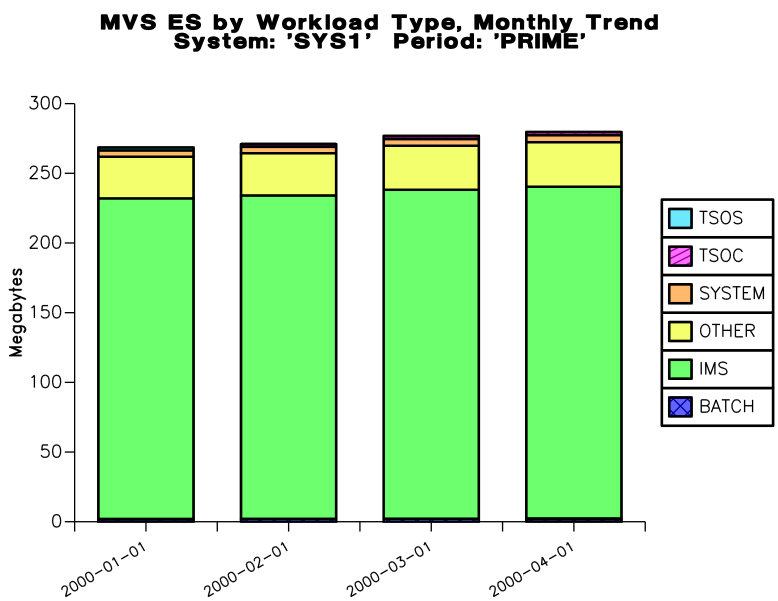

Figure 21. MVS ES by Workload Type, Monthly Trend report

Analyzing storage usage by workload

You can also look at your processor storage usage by workload type. The MVSPM Storage Used by Workload Type, Hourly Trend report shows the number of megabytes of processor (central and expanded) storage allocated to each workload type, based on storage frame seconds.

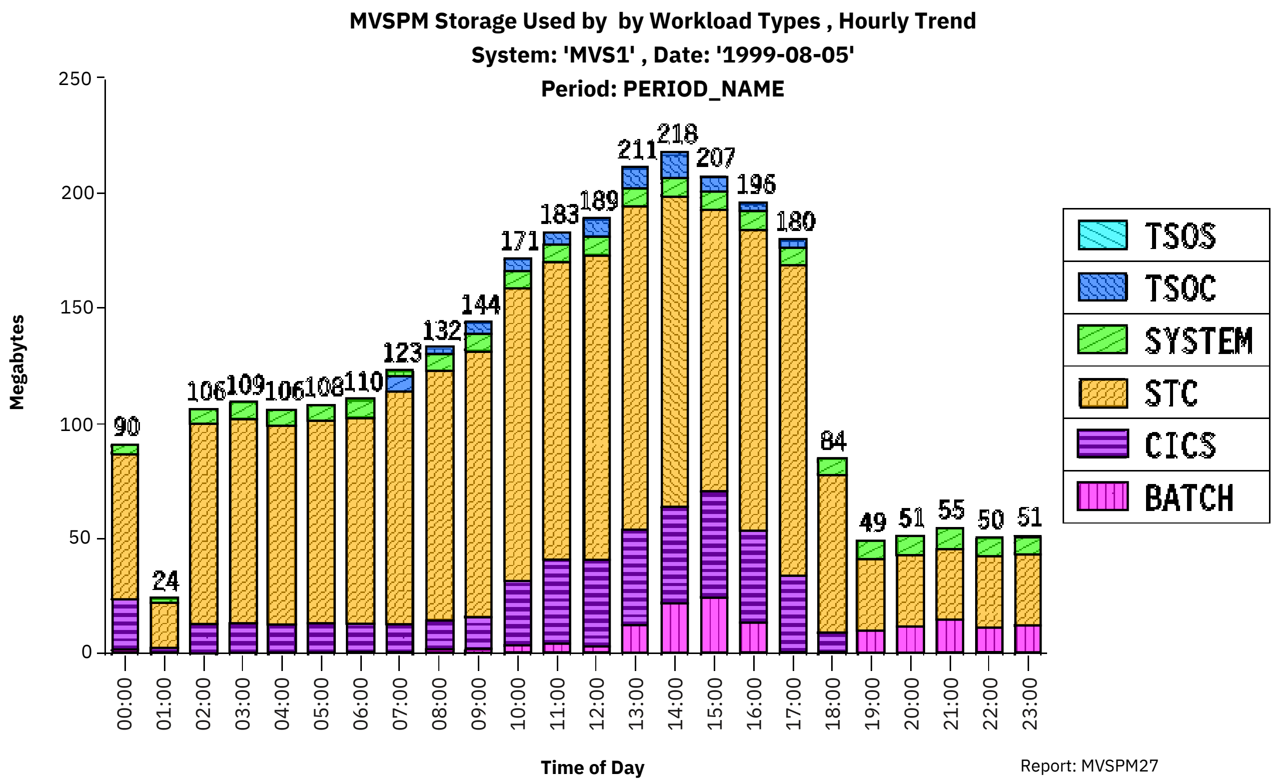

Figure 22. MVSPM Storage Used by Workload Type, Hourly Trend report

Use this report to show which workloads dominate the processor storage. If you are moving workloads to different systems, this report tells you the approximate processor storage required.

You can further break down this information by performance groups within a workload type. The MVSPM Storage Used by Workload PGNs, Hourly Trend report shows the processor storage (central and expanded) used by each performance group number within the specified workload type.

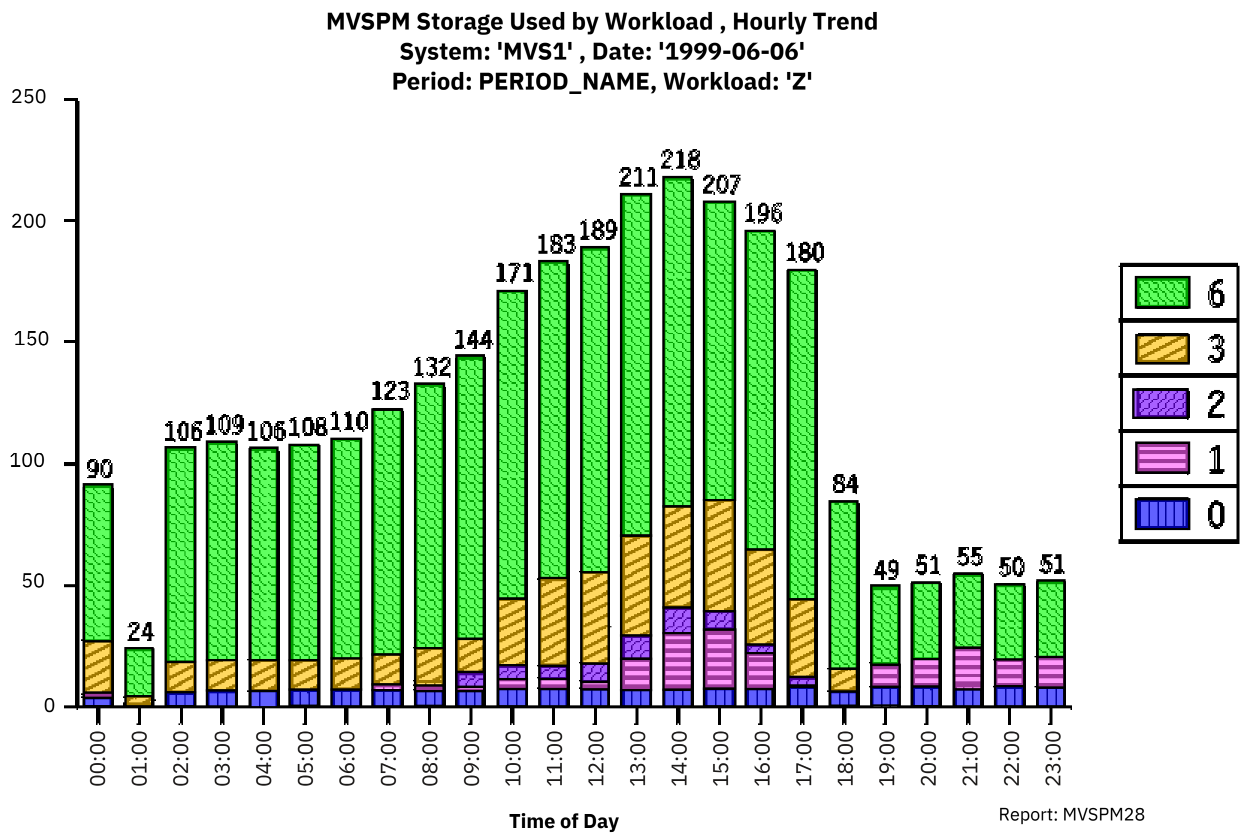

Figure 23. MVSPM Storage Used by Workload PGNs, Hourly Trend report

Providing detail by performance group number is only as effective as the granularity that has been established through the use of the IPS. Having multiple applications running in the same performance group restricts your capabilities.

Analyzing data in memory

Data in memory (DIM) is a strategy to improve system performance by efficiently using all the elements of the IBM storage hierarchy and the latest advancements in hardware and software technology. These statements summarize the DIM recommendations:

- The fastest I/O is no I/O.
- Place as much as you can in processor storage, then access the rest as fast as possible.

You can use several techniques to place programs in processor storage to avoid I/O:

PLPA
The pageable link pack area (PLPA) lets you place programs in common virtual storage. z/OS then manages the processor storage residency of these programs in an LRU manner.

Preload
Some subsystems, most notably CICS and IMS, let application programs be preloaded into their private virtual storage. The residency of these programs falls into the LRU working set management of z/OS.

Virtual fetch
Virtual fetch provides services to create a VIO data set for specified load libraries and to retrieve load modules from the VIO data set. This data set can then reside in expanded storage. IMS uses virtual fetch to hold IMS application programs. When an IMS region requests a particular program, a copy of the program is retrieved from expanded storage and moved into the private virtual storage of the requesting region.

LLA
The library lookaside (LLA) facility uses a virtual lookaside facility (VLF) dataspace to hold the most active modules of linklist and user-specified program libraries. When an address space requests an LLA-managed program that is in the dataspace, the load module is retrieved from VLF instead of from the program library on DASD.

This section describes ways you can get more data into processor storage.

Analyzing dataspace usage

A dataspace is a range of up to 2 gigabytes of contiguous virtual storage addresses that a program can directly manipulate through z/OS instructions. Unlike an address space, a dataspace contains only data; it does not contain common areas or system data or programs. Programs cannot execute in a dataspace, although load modules can reside in a dataspace. To reference the data in a dataspace, a program must be in access register (AR) mode. Up to 15 dataspaces can support an address space at one time.

Using dataspaces minimizes the need to store active data on external devices and increases integrity by separating code from data.

Analyzing hiperspace usage

High performance space (hiperspace) is a data buffer that is backed either in expanded storage only or in expanded storage and auxiliary storage. It can be used for temporary or permanent data. Its maximum size is 2 gigabytes. However, unlike dataspaces, which are limited to 15 active at any time, the number of active hiperspaces for an address space is limited only by address space limits defined in the IEFUSI SMF exit. For response-critical applications requiring large data structures, hiperspaces can provide almost unlimited definable storage, provided that the expanded storage is available.

To analyze the hiperspace usage on your system, create a report using data from the MVSPM_PAGING_H table. You can use data in the ESTOR_HIPER_AVG_MB column to create a report that shows hiperspace usage by hour for multiple days. This report would show the trends throughout the day, if hiperspaces are used, and how much storage is used. You can also create a report using the HS_PAGES_FROM_ES and PAGES_HIPER_TO_ES columns to analyze hiperspace movement to and from expanded storage.

Analyzing LLA/VLF

z/OS offers improved system performance by module placement, LLA and VLF. LLA is a component of z/OS that you can use to increase library performance, system availability, and system usability. LLA lets you eliminate I/O in searching production load libraries. Using LLA, you can eliminate I/O for your often-used (or fetched), read-only production load libraries. LLA also lets you reduce I/O in fetching modules from production load libraries.

VLF is a service offered by z/OS that provides the ability to create and retrieve named data objects, such as members of a partitioned data set in virtual storage. VLF uses dataspaces to keep large amounts of data in virtual storage. For certain applications, VLF reduces I/O for repeated retrieval of members from frequently used data sets and improves response time. Although VLF is an alternative to existing I/O functions, it does not replace these functions.

LLA uses VLF. Without VLF, LLA eliminates I/O for directory search and manages the data sets; with VLF, LLA also reduces the fetch I/O. Note that there are other exploiters of VLF, such as TSO/E and RACF. Carefully evaluate the use of LLA for libraries that are managed by other users of VLF.

The MVSPM VLF Dataspace Usage, Hourly Trend report shows various measurements for VLF for a selected period.

```
                    MVSPM  VLF Dataspace Usage, Hourly Trend
                       System: 'NRD1'  Period: PERIOD_NAME
                               Date: '1999-10-09'


                    Storage           Storage   Storage   Storage    Largest
 VLF                used avg  MAXVIRT used min  used max  used avg   object
class     Time        (%)      (MB)     (MB)      (MB)      (MB)      (MB)
--------  -----  --------  -------  --------  --------  --------  --------
CSVLLA    02:00        92     16.0    14.578    14.961    14.706     2.488
          03:00        91     16.0    14.586    14.602    14.598     2.488
          04:00        92     16.0    14.613    14.668    14.641     2.488
          05:00        91     16.0    13.434    14.992    14.603     2.488
          06:00        84     16.0    13.434    13.434    13.434     2.488
          07:00        89     16.0    13.859    14.297    14.178     2.488
          08:00        87     16.0    12.750    14.984    13.848     2.488
          09:00        92     16.0    14.473    14.809    14.643     2.488
          10:00        16     16.0     0.000     4.832     2.584     2.488
          11:00        36     16.0     5.406     6.148     5.750     2.488
          12:00        41     16.0     6.391     6.867     6.598     2.488
          13:00        57     16.0     8.582     9.242     9.063     2.488
          14:00        62     16.0     9.242    10.879     9.986     2.488
          15:00        79     16.0    12.055    13.609    12.624     2.488
          16:00        85     16.0    13.609    13.625    13.617     2.488
          17:00        85     16.0    13.625    13.633    13.629     2.488
          18:00        85     16.0    13.633    13.633    13.633     2.488
          19:00        85     16.0    13.633    13.633    13.633     2.488
          20:00        85     16.0    13.633    13.633    13.633     2.488
          21:00        85     16.0    13.633    13.633    13.633     2.488
          22:00        85     16.0    13.633    13.633    13.633     2.488
          23:00        85     16.0    13.664    13.664    13.664     2.488


                IBM Z Decision Support Report: MVSPM65
```

Figure 24. MVSPM VLF Dataspace Usage, Hourly Trend report

Note: LLA always shows a hit rate greater than 99%, because it manages which objects VLF handles and it only requests those objects that it passes to VLF.

Analyzing virtual I/O (VIO)

As a means of improving system performance by eliminating much of the overhead and time required to allocate a device and move data physically between main storage and an I/O device, z/OS provides virtual input/output (VIO). A VIO operation uses the system paging routines to transfer data. VIO can be used only for temporary data sets that store data for the duration of the current job; it uses the system paging routines to transfer data into and out of a page data set.

VIO uses paging rather than explicit I/O to transfer data. VIO eliminates the channel program translation and page fixing done by the EXCP driver and by some device allocation and data management overhead. It also provides dynamic allocation of DASD space as it is needed. Another advantage of VIO is that the data set can remain in processor (central or expanded) storage after it is created because RSM attempts to keep the pages in central storage as long as possible.

Analyzing processor storage configuration

Processor storage contention is presented by first showing the amount of processor storage used and the associated storage contention indicators. This measurement is followed by a breakdown, by workload, of processor storage and paging rates. Page activity to auxiliary and expanded storage is also shown.

When storage is overcommitted, the system will thrash, response time increases, and throughput decreases. You have control over whether a job for which there is no room waits until storage is available or enters storage and causes excessive paging. This section discusses the types of storage available and how they can affect your system performance.

Analyzing central storage

When analyzing central storage usage, you must know how your central storage is used. The MVSPM System Central Storage Map, Hourly Trend report shows the central storage allocations by system component.

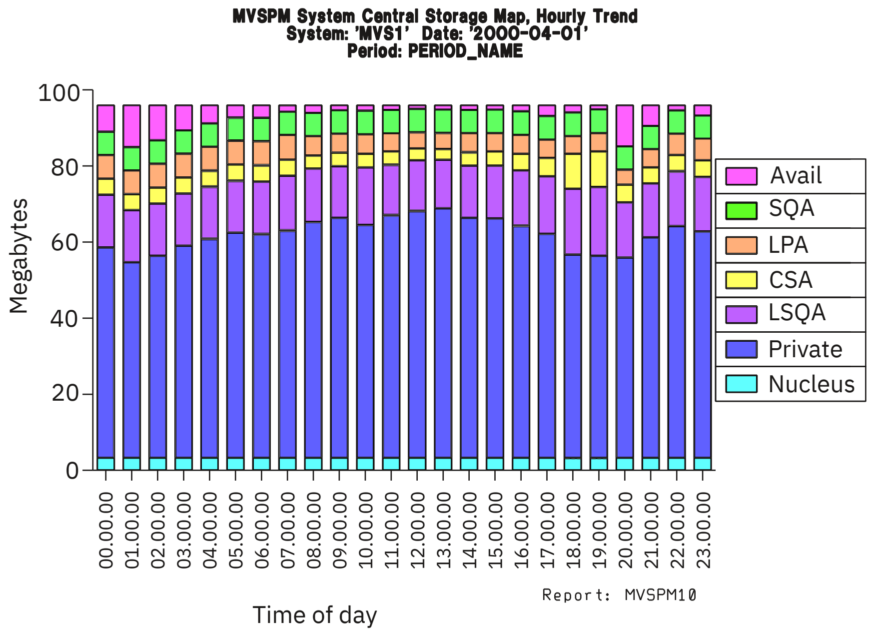

Figure 25. MVSPM System Central Storage Map, Hourly Trend report

This map shows the central storage allocated to these storage areas:

Nucleus
: The nucleus holds the resident part of the z/OS control program. Aside from the control program load module, the nucleus contains the page frame table entries, data extent blocks for z/OS libraries, recovery management support routines, and unit control blocks for the I/O devices.

Common service area (CSA)
: The CSA is addressable by all active programs and is shared by all swapped-in users. The primary advantage of the CSA is to enable inter-address-space communication.

Link pack area (LPA)
: The LPA contains z/OS control program functions (SVC routines), access methods, other read-only system programs, and selected user programs.

System queue area (SQA)
: The SQA contains tables and queues that relate to the entire system.

Local system queue area (LSQA)
: The LSQA contains tables and queues that are unique to a particular address space.

Private area
: The private area of a z/OS address space contains modules and data not shared by other address spaces.

The map also shows the amount of available central storage (Avail). The total of these fields represents the maximum online central storage for the system. Sometimes storage is taken offline. Use this report to verify that the installed central storage is available.

One value to monitor is the average system high UIC. The high UIC can indicate whether there is contention for central storage frames. When the high UIC is relatively low, contention for central storage may be high.

The MVSPM Average High UIC Profile, Hourly Trend report shows the profile for the average high UIC for selected days.

From this report, you can determine when the central storage is most used. A high UIC (255) indicates that users' working sets are not referenced often. A low UIC indicates that users' working sets are referenced frequently.

The UIC indicates the reference pattern of central storage, not the contention for central storage. You could have an average UIC of 1 and a demand page-in rate of 10 per second. You have an extremely low UIC and a low servicing of page faults by the paging I/O, which may happen when page movement is primarily between central and expanded storage. The necessary paging is not causing delays for the user.

A low UIC is not necessarily bad, especially if the system has expanded storage. A low UIC lets you use expanded storage as a fast paging device. The impact a low UIC has on swappable workload is different than its impact on nonswappable workload. If you plan to add new applications, a low UIC indicates that central storage paging will increase. The impact depends on the locality of the data reference pattern for each application. Some applications touch many of the frames of their working data sets, causing a low UIC. What matters is whether the frames the application needs are in central storage or whether they must be retrieved from expanded or auxiliary storage.

The UIC by itself is not the only indicator for storage contention. Page movement rates to and from expanded and auxiliary storage should be examined. When the UIC is close to 255, the need for a fast paging device is low. As the UIC drops, the page movement from central storage increases.

Analyzing expanded storage

If expanded storage is available on your system, pages are moved from central storage to expanded storage as needed. The MVSPM System Expanded Storage Map, Hourly Trend report shows how expanded storage is allocated.

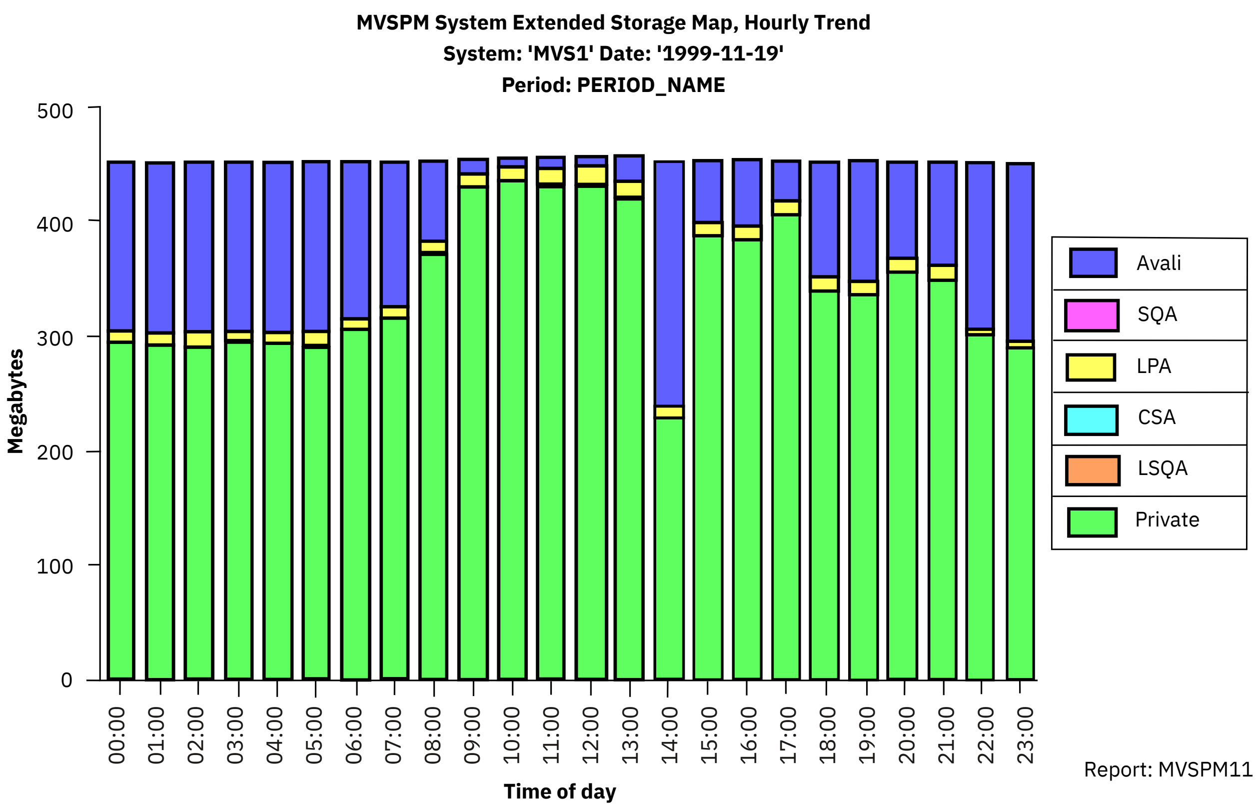

Figure 26. MVSPM System Expanded Storage Map, Hourly Trend report

This report shows the expanded storage allocated to the common storage (CSA), link pack (LPA), system queue (SQA), local system queue (LSQA), and private (Private) areas. It also shows the amount of expanded storage available (Avail). The total of these fields represents the amount of online expanded storage.

Analyzing I/O resources

Many problems reported as poor z/OS performance actually have nothing to do with z/OS. Channel loading, control unit or device contention, data set placement, paging configurations, and shared DASD are the major culprits. This section discusses I/O performance considerations.

Logical control unit (LCU) activity and DASD analysis focuses on the top LCUs and DASD devices on which applications are waiting. An LCU is the set of DASD behind a storage control unit.

The DASD multiprogramming level (MPL) indicates applications waiting. DASD MPL is based on the response time and I/O rate, and indicates the queue depth on a device.

Note: Different configurations of the DASD subsystem affect response time differently.

Measuring I/O

I/O response time is the elapsed time from the execution of the execute channel program (EXCP) macro to the completion of the data transfer, which is indicated by a Channel End/Device End (CE/DE) interrupt. It includes any queue time plus the actual operation. The following figure illustrates the components of I/O response time.

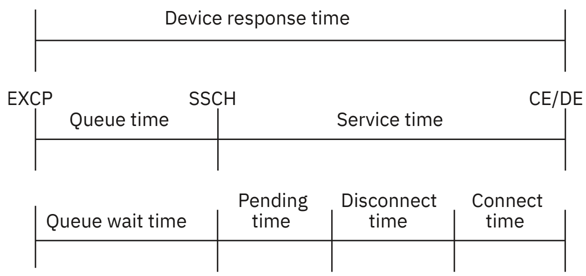

Figure 27. I/O response time components

I/O response times are expressed by component to designate which options may improve response times. I/O response time includes these components:

Queue wait time

The internal z/OS queueing of the I/O operations that are waiting for a previous I/O to the device to complete. Delays in the other service components may cause the queue component to increase, or the queue time may be a function of skewed arrival of I/Os from a particular application. RMF reports queue wait time as IOSQ time.

Pending time

The time from the start of the I/O until the DASD receives it. Pending time can be caused by busy channels and controllers or device busy from another system. If a device is behind a cache controller (non-3990), pending time can also be caused by cache staging (the device is busy during the staging operation) or when using nonenhanced dual copy (3990), the device is busy while writing the data to the primary volume and to the duplex volume if fast copy was not selected.

Connect time

The time actually transferring data between the channel and DASD or channel and cache. Connect time includes:

- The time for Search ID
- The time to transfer data
- Protocol time, which does not overlap the other times

Connect time can also include the time a search operation is occurring between the channel and the DASD or the channel and the cache (usually done to search a directory to find the location of a program module so it can be loaded into storage).

Disconnect time

Disconnect time includes:

- The time for a seek operation.
- Latency—always assumed to be half a revolution of the device.
- Rotational position sensing (RPS) reconnect delay—the time for the set sector operation to reconnect to the channel path.

  This time depends on internal path busy, control unit busy, and channel path busy. If any element in the path is busy, a delay of one revolution of the device is experienced.
- For a 3990 Model 3 cache controller, if the record is not in cache, the time waiting while staging completes for the previous I/O to the device or until one of the four lower interfaces becomes available from either transferring, staging, or destaging data for other devices.

When a device cannot reconnect to the host to transfer data because all paths are busy, it must wait for another revolution.

Using cache control units reduces or eliminates disconnect time. Disconnect time is used as a measurement of cache effectiveness.

Several SMF records contain information about the I/O activity in z/OS:

- SMF type 74 records provide information on usage and queueing at the device level.
- SMF type 72 records provide the I/O service units used by each performance group or service class during an RMF interval. Note: Db2 I/O activity is not currently recorded because Db2 uses the media manager facility.
- SMF type 75 records include I/O activity information that is specific to paging and swapping.
- SMF type 70 records contain a count of I/O interrupts processed.
- SMF type 30 records contain data on which applications are using the DASD.

IBM Z Decision Support collects data from all of these records.

z/OS uses three different measures when recording I/O activity in the SMF log:

- EXCP

  A count of the number of blocks of data transferred. The EXCP is an I/O subsystem driver. To request the movement of data, either the access method or the user program presents information about the operation to the EXCP processor by issuing the EXCP macro instruction. EXCP translates this information into a format understandable to the channel subsystem and invokes the I/O supervisor (IOS).

  SMF maintains counts of these EXCP requests. An EXCP count is a block of data transferred. However, both a 16-byte block and a 32756-byte block count as one EXCP. Also, the I/O access method or I/O driver must count blocks transferred and pass that count to the SMF count routine.
- Start subchannel (SSCH)

  A count of physical I/O operations necessary to transfer the data.

  The start subchannel (SSCH) macro is a privileged instruction issued by the IOS component of z/OS, as a result of a program issuing an EXCP. The access method can take several blocks (EXCPs) from a program, allocate several buffers, and transfer the buffer blocks with one SSCH. Because under the sequential access method, the number of buffers defaulted to five, for QSAM, five EXCPs are counted for every start I/O (SIO). For BSAM I/O, the programmer does the buffering rather than the access method. BSAM usually counts one EXCP per SSCH. For VSAM, however, it is possible to have many SSCHs issued for a single block of data. Although both EXCP and SSCH count I/O operations, they do not map directly to one another.
- I/O connect time

  The length of time that a task was actually transferring data or commands to or from a device.

  The I/O connect time is recorded at the device level in SMF type 74 records. SMF also captures the I/O connect time, like EXCP counts, in the SMF type 30 records.

When an address space executes in cross-memory mode (that is, during either secondary addressing mode or a cross memory call), the EXCP counts or the device connect time interval (DCTI) will be included in the I/O service total. This I/O service is *not* counted for the address space that is the target of the cross-memory reference.

Analyzing the data

The goal of investigating I/O resources is to minimize delays in satisfying I/O requests. A high level of contention within your I/O subsystem can be a concern for responsiveness and throughput. The issue is whether application I/Os are being delayed.

The key to finding and fixing I/O related performance problems is I/O response time (that is, the length of time it takes to complete an I/O operation). I/O response time can have a dramatic effect on performance, particularly with online and interactive subsystems, such as TSO, IMS, and CICS. I/O response time is also the most critical element in batch throughput.

You can monitor the I/O rate of your workload types using the MVSPM Workload Resource Utilization Overview report (see “Figure 6” on page 19). If the I/O rate for a particular workload is high, you must analyze the parts of the I/O subsystem to determine what is causing the delay. Look at the response-time components for these parts of the I/O subsystem:

- Channel paths
- Logical control units
- Devices
- Cash usage

The following sections address the analysis of each of these parts.

Analyzing channel paths

Channel path usage affects response time. As channel path usage increases, the response time for the devices attached to that channel can increase. The impact of channel usage is felt at several points during the execution of an I/O operation.

The higher the channel usage, the greater the probability that the I/O request will be queued. These delays are relatively short and in most cases can be disregarded, because the probability of being delayed is equal to the channel usage. An I/O request is queued if a previous request to a device on the same channel path is in progress. The time for this delay is reported in pending time for the device.

If a channel path to a device shows excessive use, you could define additional paths to the device or introduce a different job mix to produce better performance.

Use the MVSPM Channel Path Busy report to identify potentially critical I/O paths.

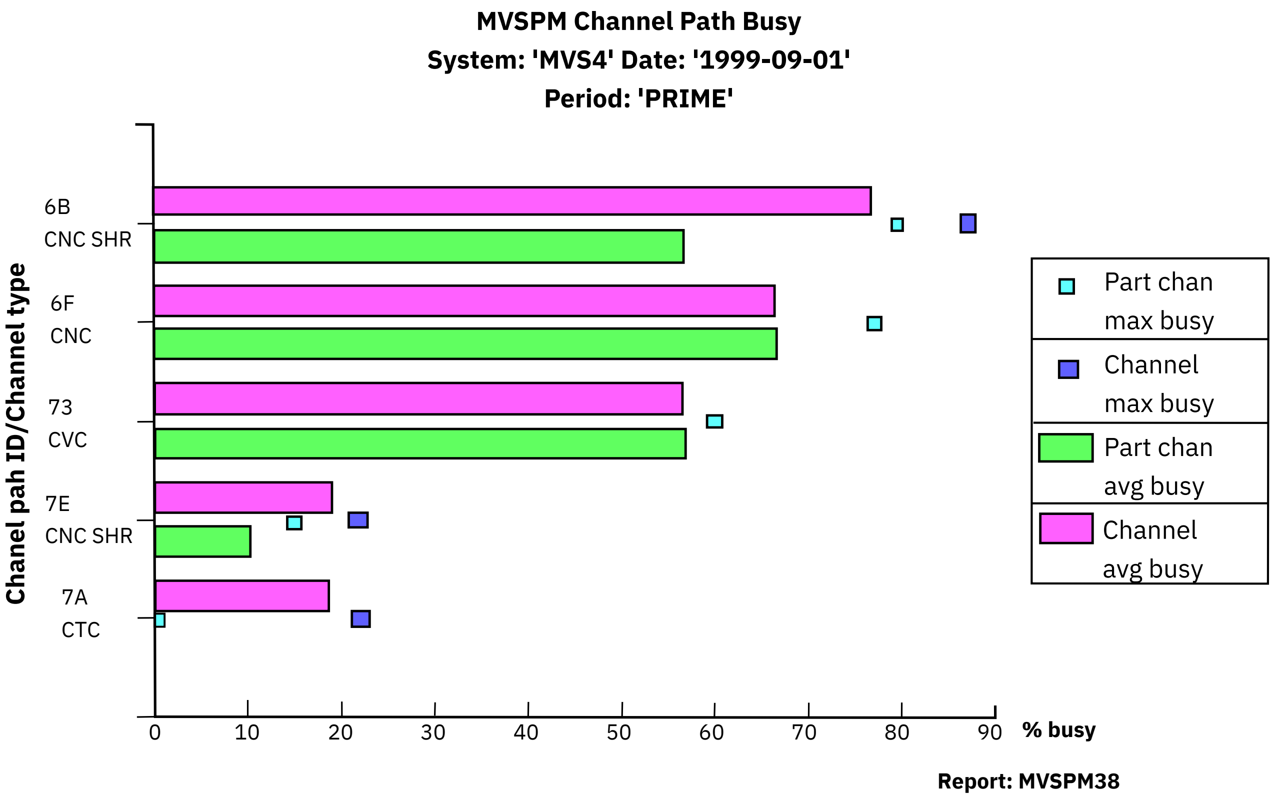

Figure 28. MVSPM Channel Path Busy report

The information on channel path busy is helpful when trying to determine the need for more or less channels. The level of busy for each channel path ID is neither good nor bad, as long as response-time objectives for the related devices and applications are achieved. Busy channel paths may cause an increase in device pend and device disconnect times. These increases will be reflected in the total I/O response time.

Often, the channel path busy will correlate in groups of two to four because they are being serviced by the same control unit. This grouping provides better insight into whether consolidation of control units behind channel paths is reasonable.

Analyzing logical control units

DASD control unit contention, like channel path usage, affects both queue time and service time. It can cause busy conditions that delay the start of an I/O request, and delay reconnects after SEEK and SET SECTOR operations.

The MVSPM I/O Rate for DASD LCUs, Hourly Trend report shows the I/O rate for selected DASD LCUs.

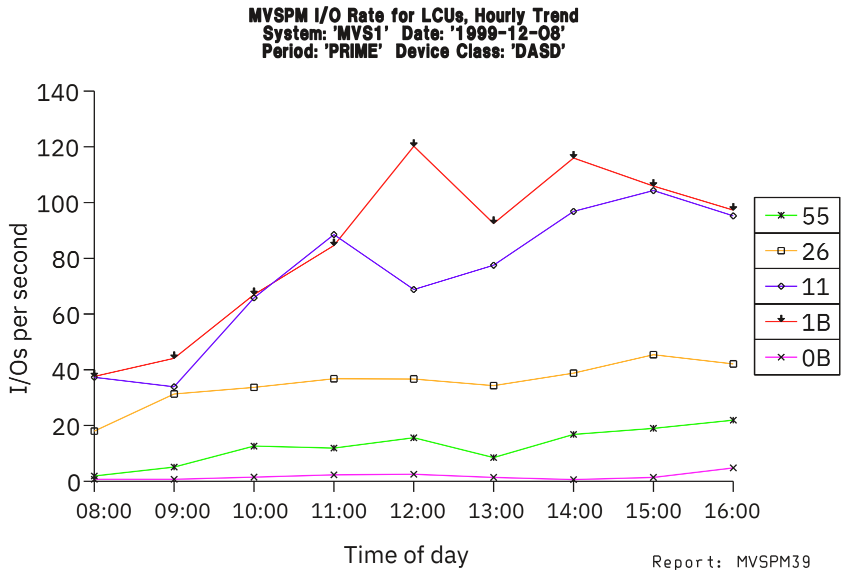

Figure 29. MVSPM I/O Rate for DASD LCUs, Hourly Trend report

Use the information in this report to determine how often and when the I/O is arriving for each of these LCUs. Compare this report to the MVSPM I/O Response Time for DASD LCUs, Hourly Trend report (see Figure 30 on page 46) to demonstrate that high I/O rates do not necessarily reflect high response times.

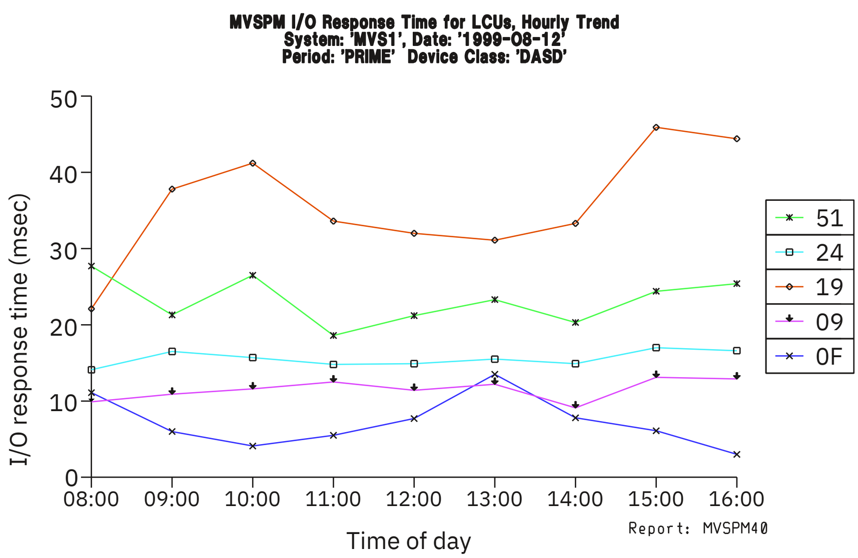

Figure 30. MVSPM I/O Response Time for DASD LCUs, Hourly Trend report

This report shows the average I/O response time for selected DASD LCUs. Use the information in this report to determine how the response time varies over a 24-hour period for each of these LCUs. You should expect the LCUs with faster devices or cached controllers to provide reduced and more consistent response times.

The MVSPM I/O Response Time Components for LCUs report in the following figure shows the I/O response time broken out by components for the top busiest LCUs. Use this report to show which component of the I/O response time is causing the most delay.

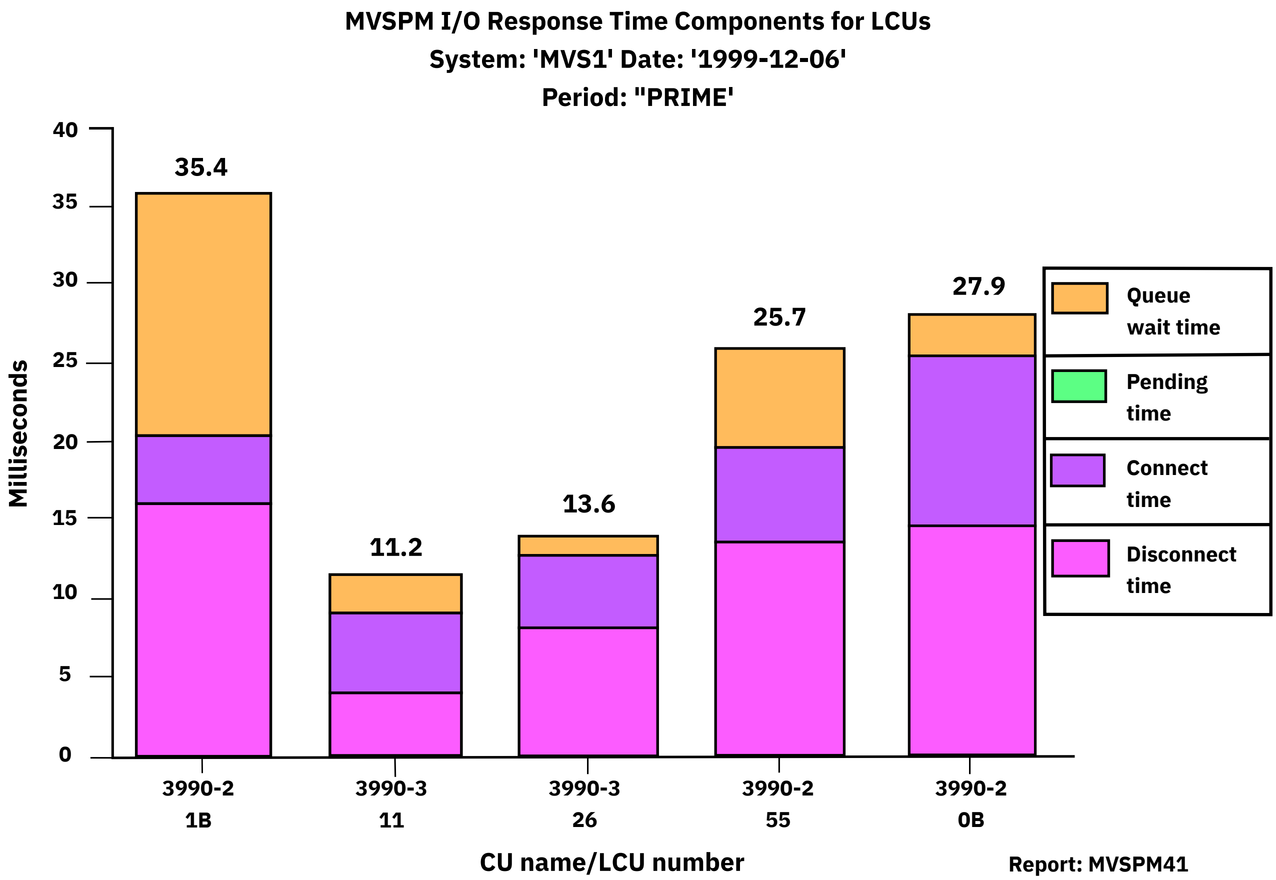

Figure 31. MVSPM I/O Response Time Components for LCUs report

Because queue time increases response time, keep it to a minimum. This is especially true for devices servicing interactive and online applications.

The control unit is busy during data transfer time. While data is being transferred, the device, control unit and channel path are essentially tied together for the duration of the data transfer.

With any RPS device, there is also a short period of time, between the reconnect resulting from a SET SECTOR and the start of the data transfer, when the control unit is busy but the channel is not.

Like channel path usage, the time a request is waiting because of control unit busy is generally much less significant than the revolutions lost because a device could not reconnect due to control unit busy. And like channel usage, these delays due to lost revolutions show up in service time. However, if the queue time due to control unit busy is excessive, look at the response time, including all the elements of queue time, for each device attached to the given control unit to determine where the problem is.

This analysis will probably show one of these conditions:

- Too many high activity devices on a single control unit
- One dominant volume

For the latter condition, a volume that monopolizes a control unit generally has a higher SSCH rate with a low response time while the other devices on that control unit have lower SSCH rates and relatively high response time. You need to determine why the device is dominating the control unit. If multiple data sets are involved, move some to devices in other less active control units.

Analyzing devices

The approach in looking at DASD is not to see which volumes are the busiest but rather to look at which volumes cause the most delays to applications, users, and address spaces.

You should start the analysis with the MVSPM DASD Activity report:

```
                    MVSPM DASD Activity, Overview
                Sysplex: 'SYSPLEX1' Date: '2000-04-13'
                           Period: 'NIGHT '

Unit     LCU  Dev         MVS   DASD I/O-rate Response Que wait Connect Disconn Pending Alloca-  Busy
type     num  num  Volser ID    mpl    (/sec)   (msec)   (msec)  (msec)  (msec)  (msec)  tions    (%)
-------- ---- ---- ------ ---- ----- -------- -------- -------- ------- ------- ------- ------- -----
9345-2  3B   EEC1 DBDC05 MVS3 0.16      3.74     42.4      3.0     2.5     1.9    34.9       0  14.0
9345-2  35   EEC1 DBDC05 MVS1 0.14      3.08     46.6      0.5     2.5     2.6    41.0       0  13.4
33902   2F   CC9C XA0CAT MVS1 0.10      9.00     11.0      0.3     2.7     0.4     7.6       3   9.3
9345-2  3B   EEC1 DBDC05 MVS2 0.10      1.94     50.9      6.1     2.6     3.8    38.5       0   8.0
33902   33   CC9A X80CAT MVS3 0.09     21.83      4.3      0.7     1.3     0.4     1.9       7   7.0
33902   33   CC9B X90CAT MVS2 0.09      5.74     16.3      0.4     3.7     0.4    11.8       3   8.9
33903   11   660F TSOL05 MVS2 0.07      0.94     73.7      1.3     5.9    66.5     0.1      20   0.6
9345-2  3B   EED1 XYZED1 MVS2 0.06      1.39     40.0      3.8    10.4     0.3    25.4       3   5.0
9345-2  35   EED1 XYZED1 MVS1 0.05      1.06     46.5      4.6    11.5     0.3    30.0       3   4.4
33903   11   660F TSOL05 MVS3 0.05      1.83     25.4      1.1     4.8    19.4     0.1      19   0.9


                     IBM Z Decision Support Report: MVSPM05
```

Figure 32. MVSPM DASD Activity Overview report

This report shows, for a selected period, the top devices sorted by DASD MPL (applications waiting). DASD MPL is a better indicator for identifying potential bottlenecks than device busy, which considers only the service time.

The MVSPM No of Users Waiting for Device, Daily Trend report (see Figure 33 on page 48) shows the number of applications waiting on the top devices.

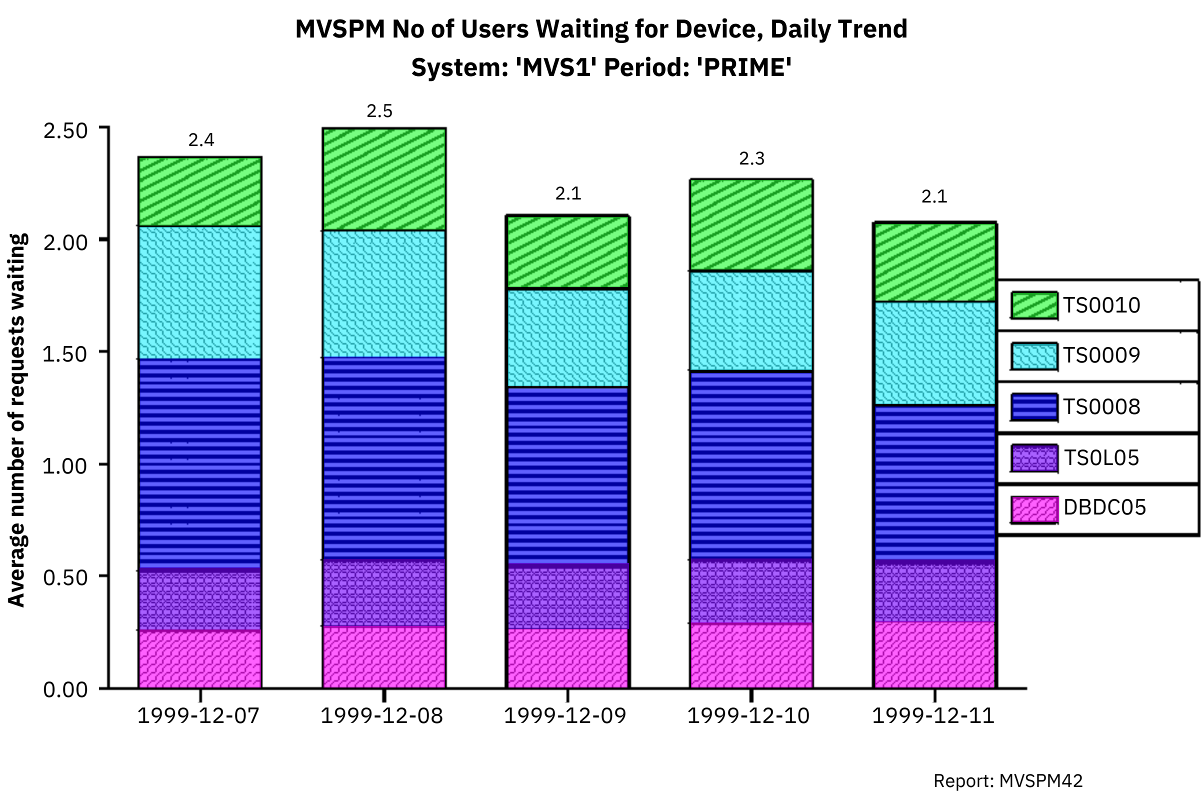

Figure 33. MVSPM No of Users Waiting for Device, Daily Trend report

The number of applications waiting is a factor of response time and the I/O activity. Use it to show the level of device contention and the variation by hour for the selected devices. If applications sharing these devices are queuing internally, this report does not show that. This is an external measurement, and there is no good or bad number. This report shows that there are fluctuations in arrival times of applications across the day. The fluctuations do not show in the MVSPM DASD Activity report , which is based on a shift average.

You can also create this report to show the top devices based on the prime shift average DASD MPL value for the focus day. The report displays these devices over all days in the measurement to show if delays occur consistently across the measurement period. This report shows that there may be fluctuations in arrival times of applications across days.

The report shows which applications will benefit if you improve the performance of a specific device.

The MVSPM I/O Response Time Components, DASD Devices report shows the I/O response time broken out by component for the top devices based on the DASD MPL level.

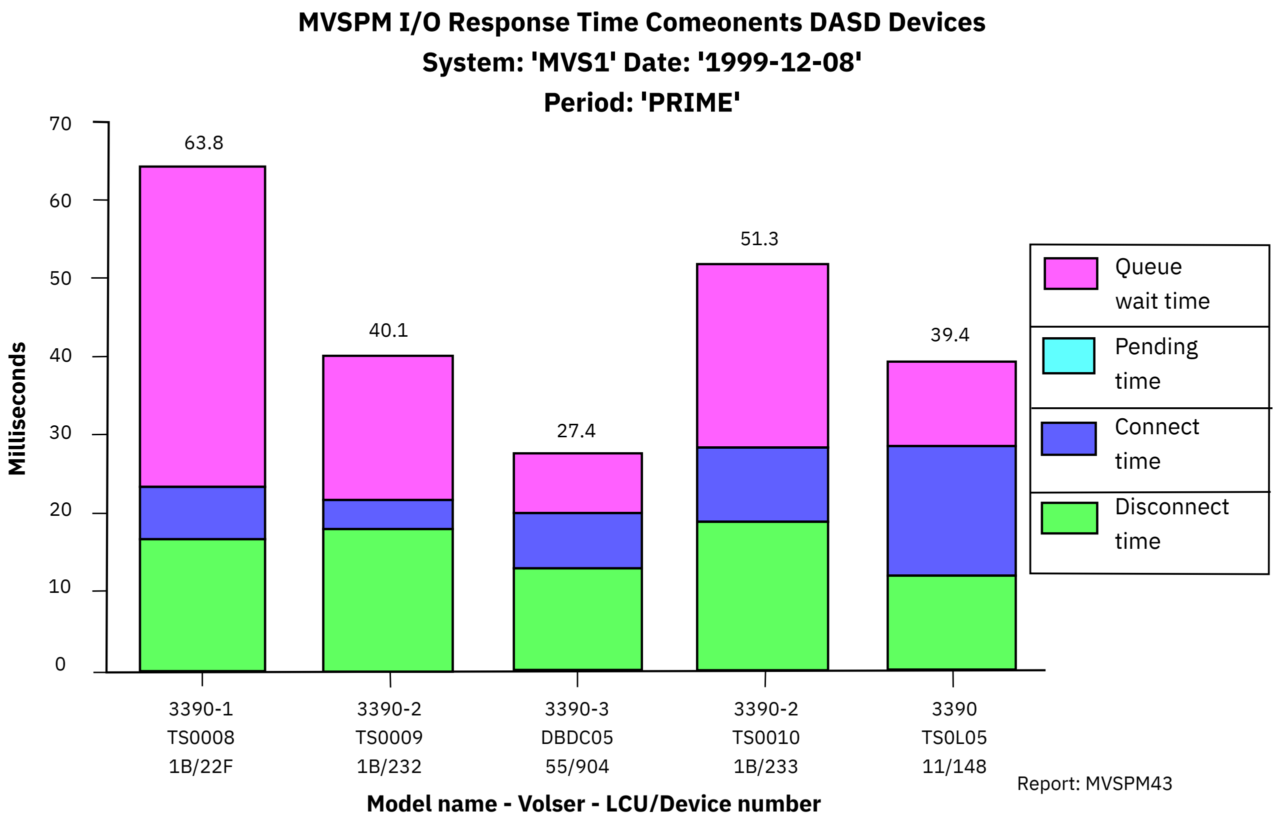

Figure 34. MVSPM I/O Response Time Components, DASD Devices report

The devices selected are the top devices from the MVSPM DASD Activity report . The report shows the DASD I/O times for the volumes with the highest DASD MPL. These volumes might not have the highest response times.

Note: High device usage by itself is not necessarily undesirable—the significant measurement is the average SSCH response time for the device.

You can use this data to show which component of the I/O response time is causing the most delay. When evaluating I/O technologies, use this report to determine the benefits of faster devices and channels, cache controllers, and additional I/O paths.

After identifying the devices with the highest DASD MPL, you can create the MVSPM I/O Response Time for Device, Hourly Trend report to show response time components for the device by the hour to see which time of day has the highest contention, and which component has the highest amount of delay.

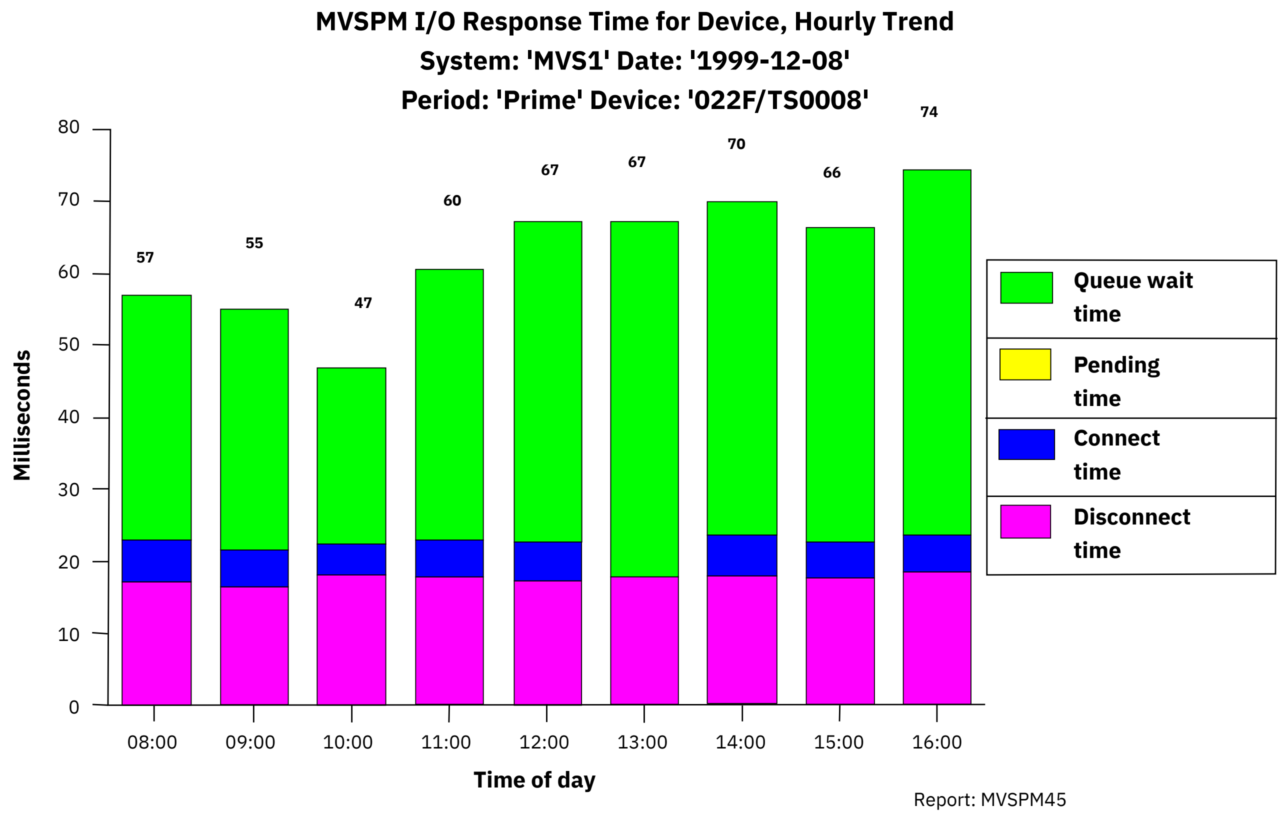

Figure 35. MVSPM I/O Response Time for Device, Hourly Trend report

For DASD volumes that take advantage of caching, there is no disconnect time for cache hits or DASD fast write hits. Caching also decreases the device connect time.

If SMF type 30 records are available and interval recording is turned on, you can create a report that shows the applications using the device. Do this to show the importance of the impact the device is having on these applications and to determine the options that will improve the service of this device.

Using the z/OS Performance Management (MVSPM) component, you can sort devices used by a specific application (in this example, TSO is the focus workload) to find the volumes with the highest activity. The MVSPM I/O Rate for Devices by Workload Type report shows the I/O rates for the device addresses with the heaviest activity for the specific workload type.

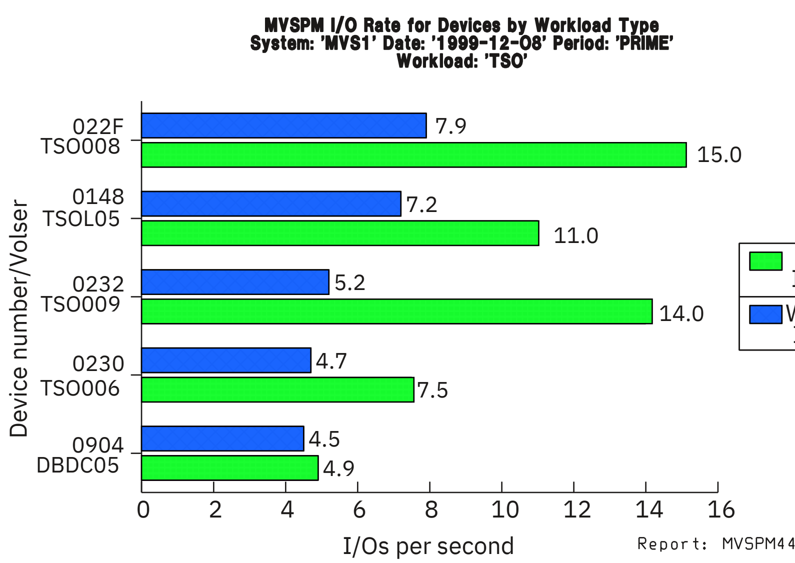

Figure 36. MVSPM I/O Rate for Devices by Workload Type report

Use this report to find important devices regarding I/O rate for an application and to determine if other applications are using the same volumes.

All workloads are not alike when sharing DASD with multiple applications. Sometimes one application will get in the way of another. For example, if batch and TSO share a DASD device, TSO applications could experience sporadic response time delays. What happens is that a batch job gets started, and it runs for 5 minutes. While it is running, the batch job keeps the device busy, therefore causing TSO to wait.

You can create a report that shows the response time components for the top DASD used by TSO to determine what is causing delays on the devices.

High device usage by itself is not necessarily a problem. It can be a problem when it causes excessive queue times, particularly for response-oriented subsystems like IMS and TSO. So, the measure of whether a device's performance is good or bad is not usage, but response time. High device usage can be one of the causes for excessive queue time. The higher the device usage, the more likely the next request will be queued. High device usage does not necessarily imply high service time. A DASD device with high usage frequently has reasonable device service time.

Analyzing cache usage

Cache is the area within the DASD controller where data for read and write access is stored. This data would otherwise be accessed directly from a DASD volume. Data read from a cache control unit significantly reduces the I/O time over data access from a DASD volume. Data found in cache is called a read hit. Hit I/O operations do not disconnect from the channel, so there are no RPS delays and the entire mechanical motion is eliminated, because there is no seek or latency.

When the data is not found in cache, a read miss occurs. The optimum situation is to have a read hit with every read operation. This yields a 100% hit ratio. The hit ratio is the number of read hits compared with the total number of read operations. The higher the ratio, the better the cache device is being used. A high hit ratio indicates that when data is accessed, it was found in cache, which eliminates the mechanical delays (for example, RPS, SEEK, and latency delays) associated with DASD access.

The 3990 Model 3 extended function cache controller allows data write I/Os to be cached using the DASD fast write (DFW) function. DASD fast write operations write the data to both cache and nonvolatile storage (NVS). It is then automatically scheduled to be transferred (or destaged) to DASD without waiting for I/O to complete. A write hit occurs when DFW is enabled and the data to be updated is resident in cache. If the data to be updated is not found in cache, a write miss occurs and the data write is direct to DASD. Then the controller stages that record along with the rest of the track into cache, so that subsequent I/Os will result in a write hit.

The 3990 Model 3 extended function cache controller also includes dual copy processing. The dual copy feature maintains identical copies of two (paired) DASD volumes on two different devices running under the same subsystem.

To determine which volumes or data sets would benefit from caching, look for devices and data used by the system's key applications. Consider enabling all volumes for caching and then note the exceptions that cause excessive staging (misses).

Also consider whether data has good locality of reference. Locality of reference refers to the location of data on DASD. If the data is scattered among numerous tracks, each track must be brought into the cache as it is needed. If volumes and data sets have poor locality of reference, they can actually run slower if they are enabled for caching. Consider disabling caching for devices with a hit percent less than 30%.

Other possible candidates to benefit from caching include security system data sets (for example, RACF), link libraries, load libraries, TSO/ISPF data sets, catalogs, indexed VTOCs, and DFSMShsm™ control data sets.

Use the MVSPM Highest Staging Rate Volumes, Daily Overview report to help identify the volumes with a majority of staging without any hit operations.

In addition, the 3990 hardware provides Enterprise Systems Connection (ESCON®) capability. ESCON adapters provide for the attachment of fiber optic cables and the use of ESCON serial protocols for communicating between channels and storage control. Using fiber optic cables lets an installation support DASD and tape devices at longer distances. ESCON provides flexibility in installing and configuring DASD, tape, and storage controls. The performance objective of the 3990 Model 3 ESCON support is to provide subsystem performance comparable to that of parallel channels at short distance and only slightly degraded performance at distances up to 9 kilometers (5.6 miles).

ESCON channels provide higher data transfer rates. However, only cache hits can take advantage of these higher rates.

The MVSPM Cache Hit Rates by CU, Hourly Trend report shows the total cache hit rate to the cache controllers.

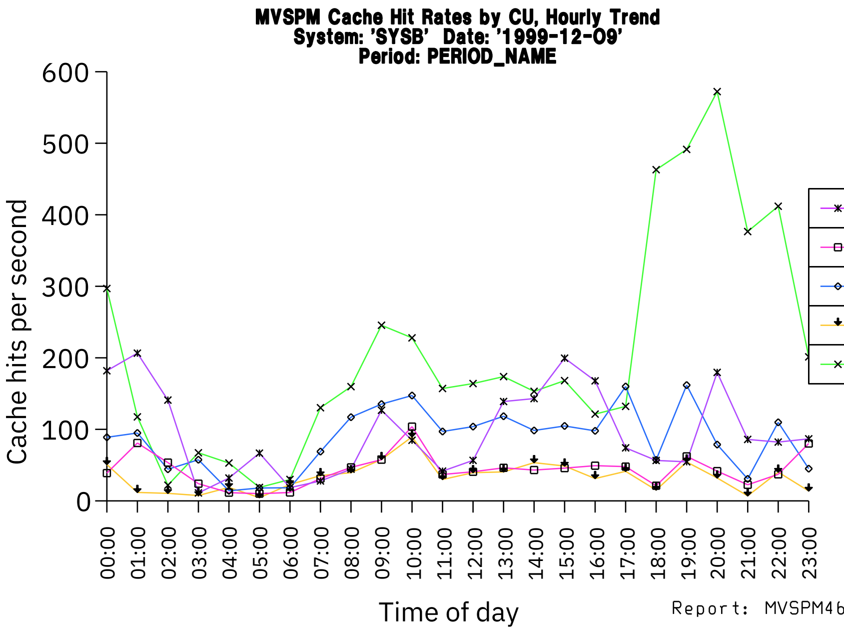

Figure 37. MVSPM Cache Hit Rates by CU, Hourly Trend report

The report shows the cache hit rate for selected cache controllers. The cache hit rate consists of normal cache hits (random reads), sequential cache hits (sequential reads), DASD fast write hits, and cache fast write hits. When I/Os are resolved in cache, the channel does not disconnect from the device. So, there is no disconnect time associated with the I/O operation. A high percentage of I/Os resolved in cache leads to lower response times for devices.

Note: If a controller is shared, the data reflects the aggregate rate of all systems accessing that controller.

To analyze the cache usage, you can break down the cache hit rate into components. The MVSPM DFW Hit Rates by CU, Hourly Trend report shows the DASD fast write hit rate, a subset of the cache hit rate.

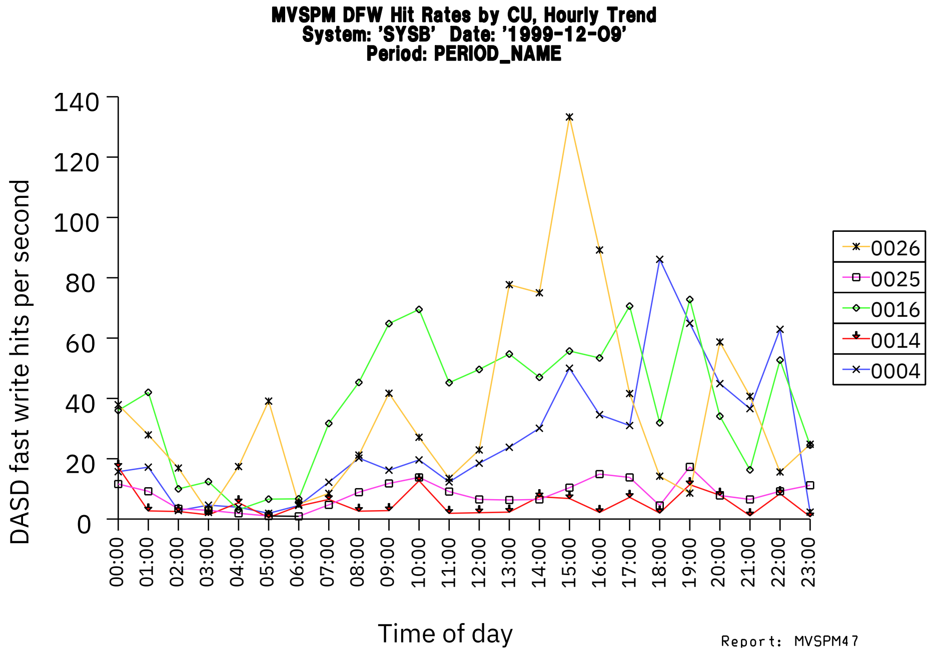

Figure 38. MVSPM DFW Hit Rates by CU, Hourly Trend report

This report provides insight into the number of write I/Os being resolved successfully in the cache by DASD fast writes. No activity on a controller indicates that either the feature does not exist or DASD fast write is not turned on.

A key indicator of the effectiveness of your cache controllers is the cache staging rate. The MVSPM Staging Rates by CU, Hourly Trend report shows the cache staging rate for selected cache controllers.

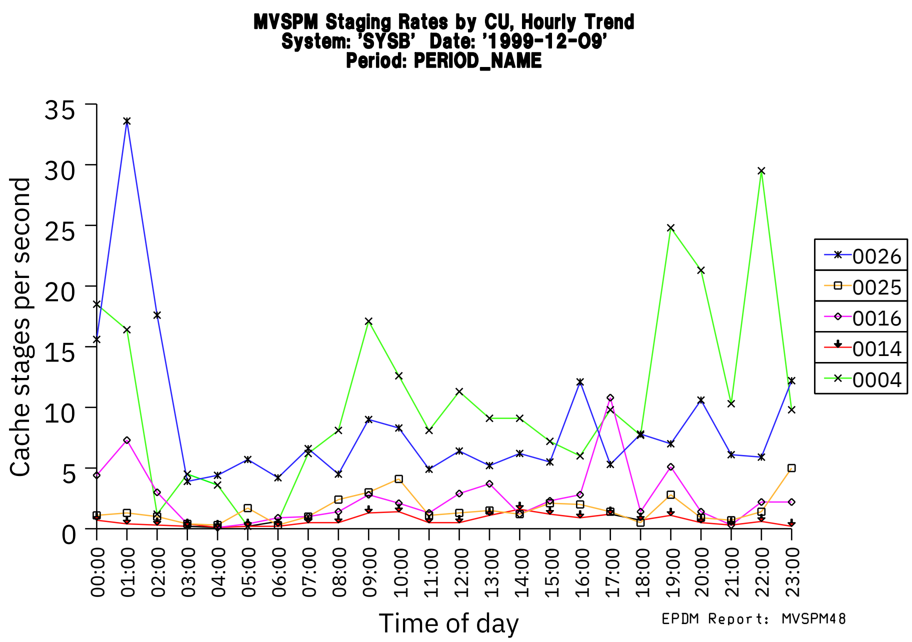

Figure 39. MVSPM Cache Controller Staging Rates report

The cache staging rate indicates how many of the DASD read and fast write operations are not being successfully resolved in the cache. When the data is not located in the cache, it must be staged into the cache so it will be available on a subsequent I/O operation. This I/O staging rate can be caused by an overcommitted cache, an insufficient amount of cache storage, or a bad locality of reference for the data being cached.

The MVSPM Overall Effectiveness by CU, Daily Overview report shows the effectiveness of the cache. Compare the number of operations resolved in the cache to the number of those that were not.

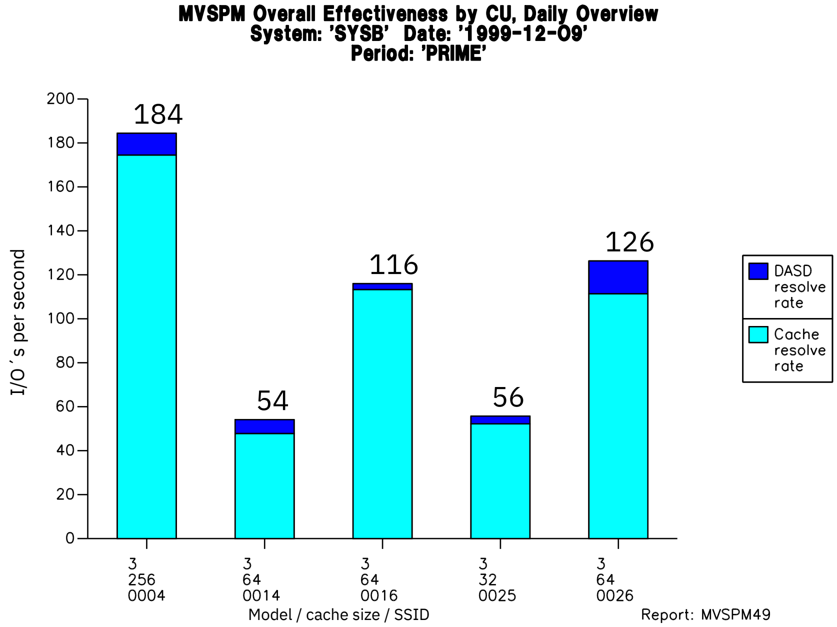

Figure 40. MVSPM Overall Effectiveness by CU, Daily Overview report

This report shows how the I/O operations are being satisfied in the cache subsystem. The cache resolve rate accounts for all I/O operations that resulted in a cache hit, including normal and sequential cache hits, DASD fast write (DFW) hits, and cache fast write hits.

The DASD resolve rate accounts for all I/O operations not satisfied in cache. These operations include staging, DASD fast write retry, inhibit, bypass, non-DFW write through, and operations to devices turned off to the cache.

The MVSPM Highest Hit Rate Volumes, Daily Overview report shows the top DASD volumes based on the total cache hit rate across selected cache controllers.

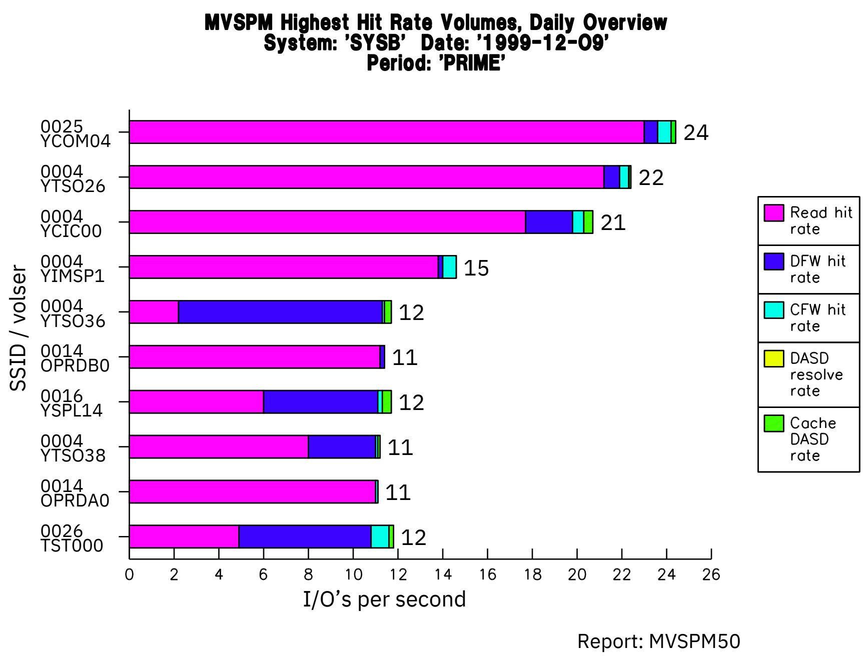

Figure 41. MVSPM Highest Hit Rate Volumes, Daily Overview report

For each volume, the report shows the I/O rate for each type of cache hit operation plus a summary of the cache miss operations. These I/O rates are given:

Read hit rate
: The I/O rate of normal and sequential cache hits.

DFW hit rate
: The I/O rate of DASD fast write hits.

CFW hit rate
: The I/O rate of cache fast write hits.

DASD resolve rate
: The I/O rate of all I/O operations not resolved in cache.

Cache DASD rate
: The I/O rate to the secondary volume of a duplex pair for volumes using the dual copy option of the 3990, or anticipatory destage from cache to DASD so nonvolatile storage (NVS) can be freed up.

This report provides insight into how the I/O operations for the volume are being satisfied within a specific cache subsystem.

The MVSPM Highest Staging Rate Volumes, Daily Overview report shows the I/O rate for each type of cache miss operation and cache hits and asynchronous operations:

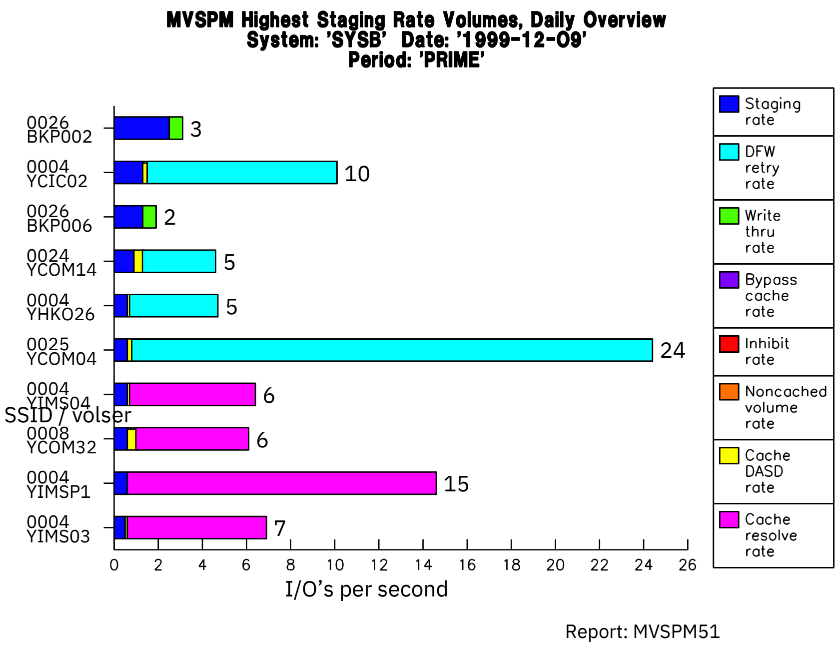

Figure 42. MVSPM Highest Staging Rate Volumes, Daily Overview report

This report gives these I/O rates:

Staging rate
: The I/O rate of read misses and DFW misses. The record was not found in the cache. It is accessed from DASD, and that record with the rest of the track is staged into the cache.

DFW retry rate
: The I/O rate of DASD fast write retries. The record is in cache and is about to result in a cache hit; however, nonvolatile storage (NVS) is full. The operation is not retried, but written directly to DASD. DFW retries are also called DFW bypasses or forces.

Write thru rate
: The I/O rate of write through to DASD. These are write operations for 3880-23s and writes to devices behind 3990-3s that are not enabled for DFW.

Bypass cache rate

The bypass mode was specified in the define extent. Examples of bypass include Db2 prefetch, ICKDSK, paging, DFSORT™ SORTWK if CFW not active, WRITECHECK data sets opened for OUTPUT or UPDATE, and DFDSS on write operations. The I/O is sent directly to DASD, the track is not promoted, and the record is invalidated in cache. If DFW is active for the device and the updated record has not been destaged, the destage must first be completed.

Inhibit rate

The inhibit mode was specified in the define extent and the record was not in the cache. Examples of inhibit mode include DFDSS for reads and DFSMS Dynamic Cache Management for maybe cache and never cache storage classes. The I/O will retrieve the record in cache if it exists there. If the record is not in cache, the I/O is sent directly to DASD, but no staging of the rest of the track occurs. If inhibit mode was specified and the result was a cache hit, the I/O would fall into one of the cache hit categories. Inhibit includes only the misses, which do not result in staging.

Noncached volume rate

The I/O rate to devices disabled to cache operations.

Cache DASD rate

The I/O rate to the secondary volume of a duplex pair for volumes using the dual copy option of the 3990, or anticipatory destage from cache to DASD so nonvolatile storage (NVS) can be freed up.

Cache resolve rate

The I/O rate of all I/O operations that resulted in the data being resolved in the cache.

You can conclude that the higher the staging rate, the higher the cache contention. You should note whether the volumes are in the same caching subsystem. This report might indicate pervasive cache contention, where more effective caching is required. You could turn some volumes off, implement DFSMS Dynamic Cache Management Enhancement, or install larger cache memory.

Analyzing enqueue contention

Some resources periodically build up queues of one or more requestors waiting to use the resource. The time a requestor spends waiting on a resource can affect system throughput and response time. So, you must monitor resources to identify which applications might be delayed because of enqueue contention.

Measuring the data

The enqueue activity record (SMF type 77 record) provides information about resources that periodically build up queues of one or more requestors waiting to use the resource. The information in this record can be very helpful in locating resources that consistently cause bottlenecks. Contention is reported for those resources to which access is controlled by jobs that issue ENQ and DEQ macro instructions.

The data-gathering technique used to compile the information provided in the type 77 record differs from the techniques used for other SMF records in that samples are taken only when enqueue contention begins and ends. This technique greatly increases the accuracy of the data, because measurements are taken only when an event occurs that affects the resource contention activity.

Analyzing the data

The MVSPM Applications Waiting on ENQs, Worst Case report shows the number of applications and system functions waiting for the major enqueued resources.

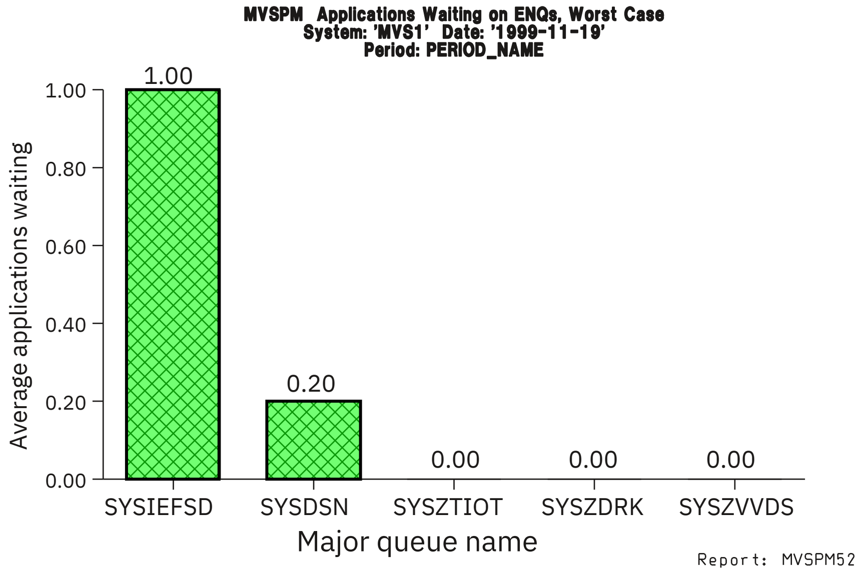

Figure 43. MVSPM Applications Waiting on ENQs, Worst Case report

Use this report to identify contention for serialized resources. The z/OS enqueue mechanism provides integrity by serializing user access requests to these resources. The number of applications waiting represents the number of users who are swapped in and cannot be dispatched because the resource they need is being held by another user.

Some applications use major names to control their users. This may have an impact on the interpretation of this report. Each major name displayed shows the number of users waiting or using this resource and being controlled by the application. To understand the difference, you must define each major name to either a system function or an application requirement.

Sometimes, when there are other bottlenecks in the system, you may see MPL spikes on the enqueue resource queue, due to a sudden release of multiple users that were being help up. When a job on one system holds onto a DASD device and work on another system needs that device, a whole flurry of requests often go out to enqueued resources, such as RACF, when the second system gets access to the DASD.

Once you have identified a critical resource, such as a serially reusable resource that can be requested on either an exclusive or shared basis, your organization can improve the situation in many ways. You could change the hardware configuration to release device bottlenecks, change data set placement, reschedule jobs to improve throughput, or specify again the enqueue residence value (ERV) installation tuning parameter to give more processor time to the holder of the resource.

Chapter 4. Analyzing DFSMS storage management data

With growing needs for additional storage resources, most organizations cannot afford to waste storage on DASD volumes. You must monitor storage usage, reclaim unused storage, and manage data set migration and deletion. This chapter discusses using the DFSMS™ component to monitor your storage usage.

Before DFSMS, most allocation of DASD space was done on a volume basis. For example, if a project required space, a whole volume was added to the pool of the project. So, volume-level reporting was generally sufficient.

In today's environment, device capacity has increased substantially, so it no longer makes economic sense to allocate storage on a volume basis. Persistence in doing so results in:

- Under utilization of storage
- User awareness of physical volumes
- Volume-dependent applications
- User involvement in storage management
- Data growth restrictions
- More frequent out-of-space incidents
- Device migration complications

In a system-managed environment, volumes belong to a storage group. Performance, availability, and management requirements are automatically assigned to a data set by associating a storage class and a management class during allocation, recall, or recovery. A storage group is assigned to a data set based on characteristics such as data set size, whether it is to be cached, or whether it is to be temporary or permanent. Consequently, as volume awareness is eliminated, it becomes necessary to produce meaningful, volume-independent reports for DASD capacity planning and storage management.

While system-managed storage provides a more flexible environment for managing storage, the DFSMS component gives an organization with the data necessary to monitor, plan, analyze, track, and report storage. The DFSMS component provides a wealth of information in a variety of reports at different levels of detail.

Measuring storage usage

While system-managed storage provides a more flexible environment for managing storage, the DFSMS data collection facility (DCOLLECT) furnishes you with the data necessary to monitor and track storage usage. The DCOLLECT log contains data from catalogs, volume table of contents (VTOCs), and DFSMShsm™.

IBM Z Decision Support processes these records from the DCOLLECT log:

- Active data set record (type D)

  This record provides data relating to space usage, data set attributes, and indicators for data sets on the selected volumes and storage groups.
- Volume information record (type V)

  This record provides statistics and information on volumes for which data is collected.
- VSAM association record (type A)

  This record provides specific information relating to virtual storage access method (VSAM) data sets residing on the selected volumes and storage groups.
- Migration data set record (type M)

This record provides information on space usage and data set attributes for data sets migrated by DFSMShsm.

- Data set backup version record (type B)

  This record provides information on space usage and data set attributes for all versions of data sets backed up by DFSMShsm.

- DASD capacity planning record (type C)

  This record provides information and statistics for volumes managed by DFSMShsm: primary volumes and migration level 1 (ML1) volumes.

- Tape capacity planning record (type T)

  This record provides statistics for tape volumes managed by DFSMShsm.

Managing storage usage

This section discusses the overall picture of your storage usage. By looking at the overall picture, you can:

- Monitor trends
- Identify problem areas
- Plan for future needs

Managing storage usage trends

To analyze your storage resources, find out what you have currently and how it is being used. By tracking and monitoring storage usage, you can more effectively provide storage to support the users.

Managing DASD storage

The DFSMS Total DASD Usage, Monthly Trend report lets you track DASD storage usage compared with the amount that has been installed. The report shows the average allocation on the DASD volumes the DCOLLECT facility has processed on a monthly basis and the total capacity of the volumes. The report shows an overview of the storage situation of the system.

You can see if the amount of DASD installed is less than the amount planned. You supply the planned storage information in a lookup table (DFSMS_PLANNED_CAP) so you can track whether the DASD is being installed according to the projected plan for the year. Refer to the *System Performance Feature Reference Volume 1* for information on updating the lookup table.

Investigate situations where the amount installed and the amount planned are not the same. If the amount installed is less than the amount planned, this could indicate that:

- Volumes are offline.
- There are problems with the volumes.
- The DASD is not being installed in a timely manner.

You can quickly determine whether storage usage is increasing or decreasing and at an expected or unexpected rate. If the storage usage deviates significantly from the plan, further investigation is required. You must determine which group or situation is responsible for the increase or decrease.

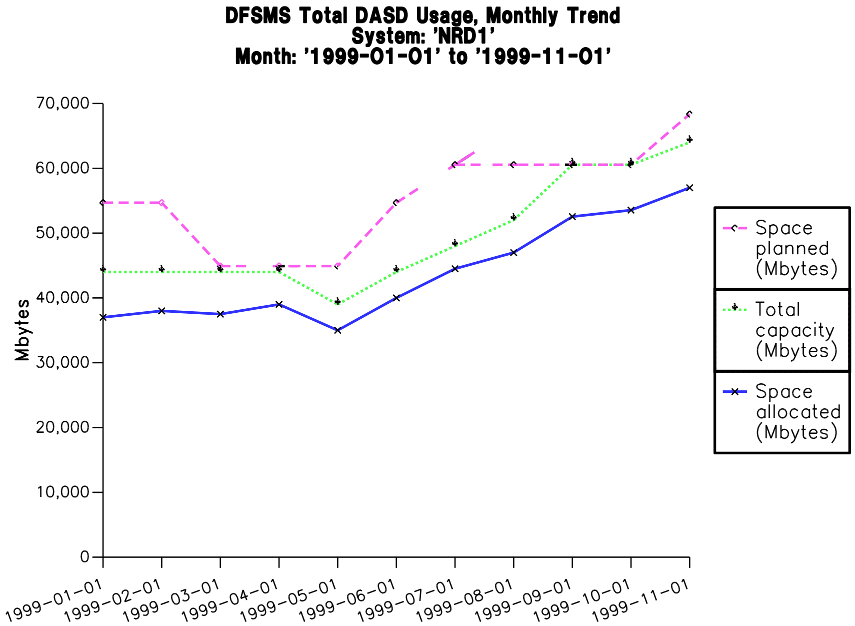

Figure 44. DFSMS Total DASD Usage, Monthly Trend report

Note: To allocate data across all of your DASD is impractical, so draw a line somewhere below total capacity. This line would then represent a more acceptable target against which to measure total allocated space.

If the DFSMS Total DASD Usage, Monthly Trend report shows possible problems requiring further investigation, follow these steps:

1. Display the DFSMS Projects Allocating Most DASD report or the DFSMS Total Storage Used by Project report. If any projects need additional investigation, go to step 2.
2. Display information for the projects. Use the DFSMS Allocated/Used/Unused Storage, Daily Trend report to determine if the projects are tracking to target. If the projects are allocating more storage than planned, go to step 3.
3. Contact the relevant project department to see what can be done to bring the project back on track.

Managing total storage

As storage is installed and used, find out which storage category is occupying it.

The DFSMS Total Storage Used by Project report tracks the total amount of allocated storage by these DFSMShsm categories:

DFSMShsm category
    Report column

Active (L0)
    Space allocated

Level 1 migration (ML1)
    Mspace DASD

Level 2 migration (ML2)
    Mspace tape

Backup
    Bspace DASD Bspace tape

DFSMS Total Storage Used by Project
System: 'NRD1'
Date: '1999-10-02'

Project	Total storage (Mbytes)	Space allocated (Mbytes)	Mspace DASD (Mbytes)	Mspace tape (Mbytes)	Bspace DASD (Mbytes)	Bspace tape (Mbytes)	Migration ratio (%)	Backup ratio (%)
ABC11	437	90	74	0	273	0	12,0	11,5
ABC12	114	34	10	0	69	0	4,3	5,2
ABC13	381	101	277	0	2	0	77,5	1,2
ABC14	1 354	384	662	0	308	0	61,7	6,3
ABC15	103	69	5	0	29	0	11,1	6,0
ABC16	6	2	1	0	3	0	73,0	48,6
ABC18	874	9	370	0	495	0	57,4	48,2
ABC19	13 768	2 123	3 499	0	8 147	0	33,0	24,0
ABC2	57	5	10	0	41	0	45,0	19,2
ABC22	336	1	126	0	210	0	64,0	18,8
ABC3	356	150	5	0	201	0	7,9	10,8
ABC4	3 328	485	2 288	0	555	0	18,0	17,1
ABC5	3 936	26	1 809	0	2 101	0	52,6	22,9
ABC6	39	5	9	0	25	0	59,7	43,0
ABC9	896	83	138	0	675	0	22,5	7,0

IBM Z Decision Support Report: DFSMS05

Figure 45. DFSMS Total Storage Used by Project report

You can use the information in this report to plan storage requirements for workload transfers of projects to other systems, and for disaster recovery planning.

The report may be useful in identifying which storage category is increasing or decreasing the most.

The report also shows the relative size between the different categories. These proportions can be expected to change as you move to a fully system-managed environment.

Managing storage allocation

In a system-managed environment, you must track the amount of storage being allocated by storage class. The storage class defines the performance and availability characteristics of a data set. Hence, this type of information helps you in planning storage hardware requirements.

The DFSMS Total Storage Used by Storage Class report tracks the amount of storage allocated for each storage class on a daily basis. Space used by data that is not system-managed is reported in the non-SMS class.

DFSMS Total Storage Used by Storage Class
System: 'MVS1'
Date: '2000-10-02'

Storage class	Total storage (Mbytes)	Space allocated (Mbytes)	Mspace DASD (Mbytes)	Mspace tape (Mbytes)	Bspace DASD (Mbytes)	Bspace tape (Mbytes)	Migration ratio (%)	Backup ratio (%)
	2 747	7	0	0	2 739	0	0,0	16,6
SNMN	22 818	3 430	9 285	0	10 103	0	30,3	19,7
SYQR	421	129	0	0	292	0	0,0	17,6

IBM Z Decision Support Report: DFSMS06

Figure 46. DFSMS Total Storage Used by Storage Class report

Isolating storage problems

The DFSMS component provides a series of reports that let you monitor storage based on a snapshot of the system or specific projects. You can use these reports with the trend reports to identify and isolate problems and to help you better understand how storage is used.

Managing projects using most storage

Many organizations have at least one very large system or processor that supports a variety of applications or projects. In this environment, you must identify the projects that are using the most storage, because either a change in the system or in the project might have dramatic effects. These projects usually demonstrate the largest growth rates.

The DFSMS Projects Allocating Most DASD report identifies the heaviest ten storage users of a system. The report shows the space allocated on primary volumes.

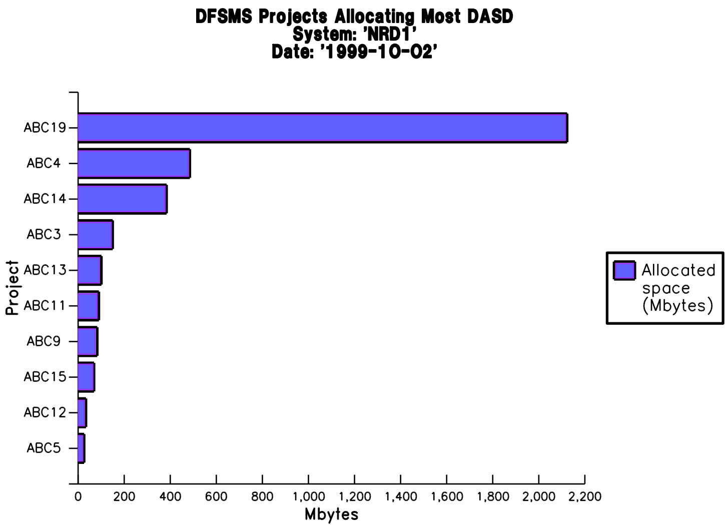

Figure 47. DFSMS Projects Allocating Most DASD report

Once you identify the top storage users of a system, you can easily create a report of the storage usage of these projects on a daily and monthly basis. You can detect unexpected fluctuations in the storage usage of the project. Discuss these exceptions with your project departments.

Managing wasted storage

You also must know the amount of reclaimable storage on a system. Reclaimable storage is wasted storage. Wasted storage results from:

- Inactive data not being migrated
- Poor blocking
- Data that should reside on tape
- Space that is allocated and not used

To ensure that DASD is being used cost-effectively, monitor wasted storage information. Organizations that are storage-constrained and cannot install more storage can use this information to reclaim some of this wasted storage.

The DFSMS Allocated/Used/Unused Storage, Daily Trend report tracks the used storage to the allocated storage of a system. The allocated storage is the amount of storage on primary (L0) volumes and does not include migration or backup storage.

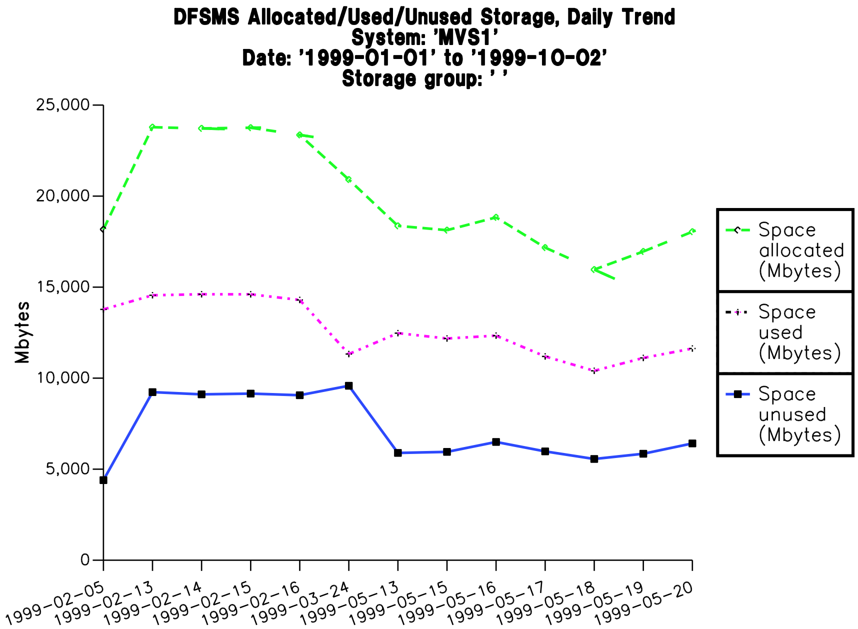

Figure 48. DFSMS Allocated/Used/Unused Storage, Daily Trend report

The DFSMS Projects Wasting Most DASD report identifies the ten projects that could benefit most from reclaiming unused space.

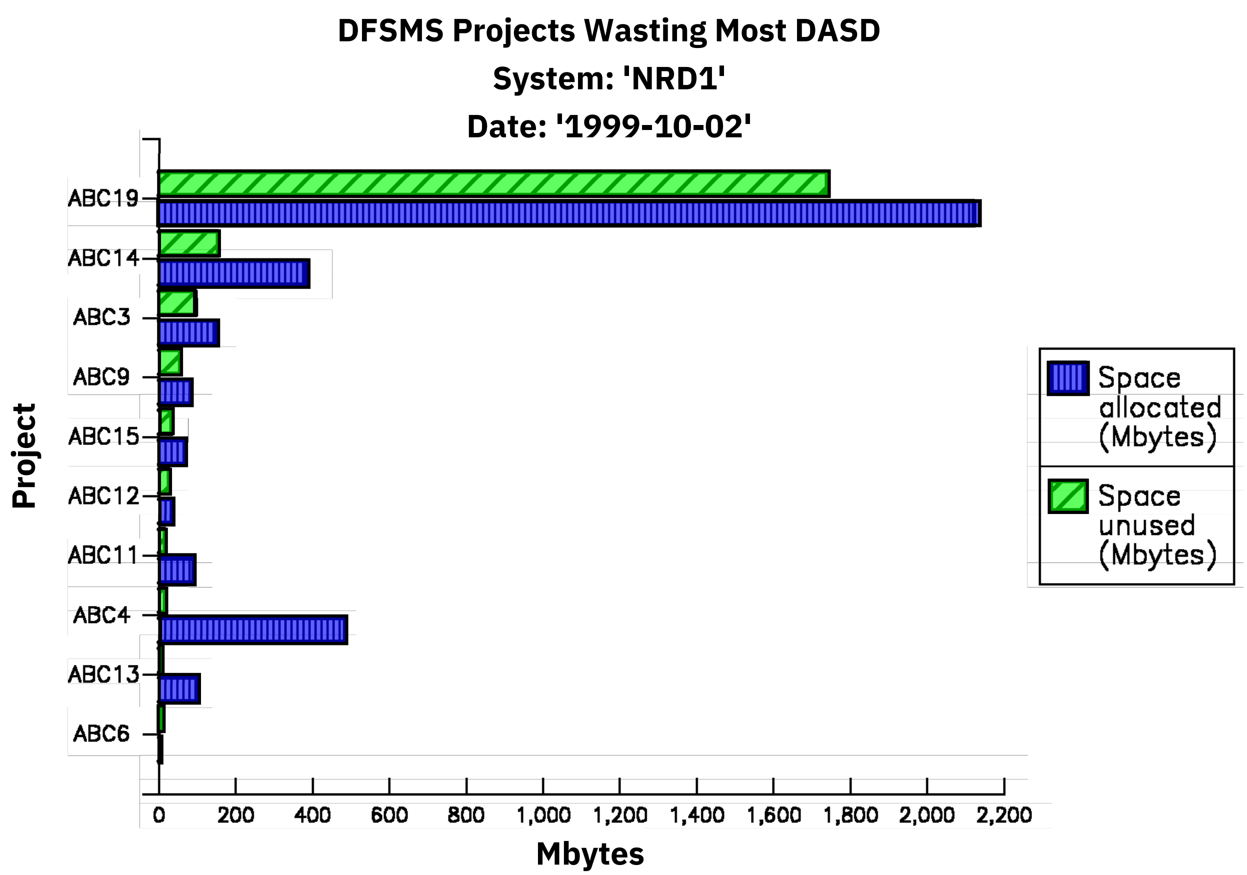

Figure 49. DFSMS Projects Wasting Most DASD report

Over-allocation of data sets is common in many organizations; it wastes considerable primary storage. This over-allocation is a result of users not understanding their storage requirements, not modifying their

JCL as the storage hardware changes, and not coding the release parameter (RLSE) in their JCL. This wasted storage can be eliminated in a system-managed environment by specifying PARTIAL RELEASE = YES or PARTIAL RELEASE = COND in the management class definition. When YES is specified, DFSMShsm unconditionally releases the unused storage during the daily storage management process. When COND is specified, DFSMShsm releases storage only if there is a nonzero secondary storage allocation defined.

By examining the DFSMS Total Storage Used by Project report, you can determine if over-allocation is a problem. If this is the case, find out which projects have the most unused storage. These project departments should be contacted so they may release the wasted storage or so you can ensure that the proper management class has been assigned. You can use DCOLLECT output or the Interactive Storage Management Facility (ISMF) to generate lists of data sets with unused storage.

You need to look only at the few projects that have the most unused storage, because normally, a small number of projects account for most of the wasted space.

Managing free capacity and storage group thresholds

Ensuring that the user community has sufficient storage for new data set allocation is a primary responsibility of the storage administrator. Your SLAs should document the expected amount of free storage in each storage group or pool. So, you must set a threshold indicating when available storage reaches the minimum level for each storage group or pool, and take action when the available storage drops below this threshold.

The DFSMS Free Space by Storage Group, Monthly Trend report shows the average free space in your storage groups. Use this report to monitor the free space situation in your storage groups. When the available free space drops below a threshold, either increase the total capacity by adding more DASD to exposed storage groups or initiate cleanup actions against your storage groups.

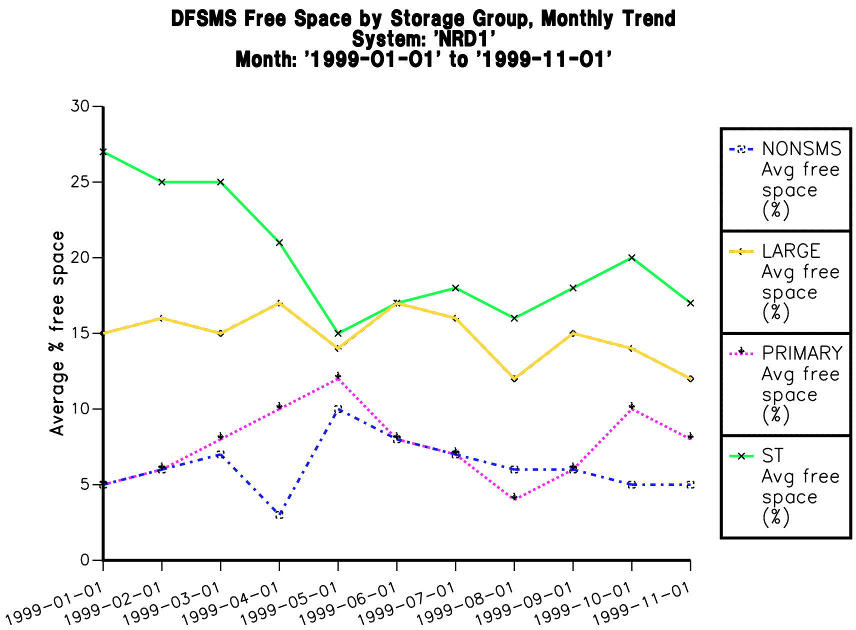

Figure 50. DFSMS Free Space by Storage Group, Monthly Trend report

The DFSMS Free Space by Storage Group, Daily Trend report shows the daily average free space for each storage group. This type of trend report is particularly useful in a non-system-managed environment where the threshold migration of DFSMShsm has not been implemented. Organizations having storage problems can use this type of report to identify periods of high or increased allocation.

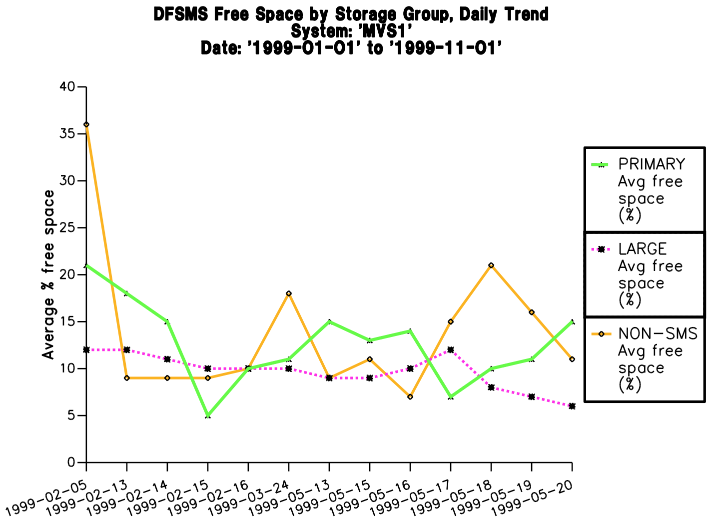

Figure 51. DFSMS Free Space by Storage Group, Daily Trend report

Besides the previous report, you should create daily reports based on free space of a volume or the low (target) threshold of a storage group to monitor storage and, with the DFSMShsm component of DFSMS, to help prevent out-of-space (E37, B37, or D37) abend situations. Exception reporting is the recommended method, because it eliminates excessive manual involvement. Non-system-managed volumes can be reported on by volume serial number or by volume pools.

Using migration threshold reporting

System-managed or DFSMShsm-managed volume free space and storage capacity reporting should be based on threshold management. You use thresholds to balance storage allocation across a pool. You can monitor storage in this environment by reporting on these indicators:

- The amount of data not migrated but eligible for migration
- If DFSMShsm was not able to meet the low or target threshold value of the volume.

DFSMShsm collects this information and records it in the DCOLLECT DASD Capacity Planning record. DFSMShsm creates records for each L0 and ML1 volume. Using IBM Z Decision Support, you can report on threshold information at the storage group or pool level, besides the volume level.

The low or target threshold for a volume is the minimum amount of data that you want to maintain on the volume. During daily space management for L0 volumes, DFSMShsm migrates data only until the low or target threshold value is met. On ML1 volumes, DFSMShsm migrates all data that is a candidate for ML2 migration.

For L0 volumes, by reporting on the amount of storage not migrated (because the low threshold has been met) but eligible for migration, you can track at the volume or pool level whether there is sufficient capacity, and forecast future requirements. When the amount of migratable storage is decreasing, you must plan to add more storage. If the amount of migratable storage is increasing, there may be an excess of storage. Investigate the situation further before removing any volumes.

The DFSMS Data Not Migrated but Eligible, Daily Trend report shows the percentage of storage that is not migrated in each storage group, but is eligible for migration because the target threshold was reached. This report shows that there may be too much storage in the TEST storage group, because the volumes are meeting the target threshold and more than 15% of the data is still eligible for migration but not migrated.

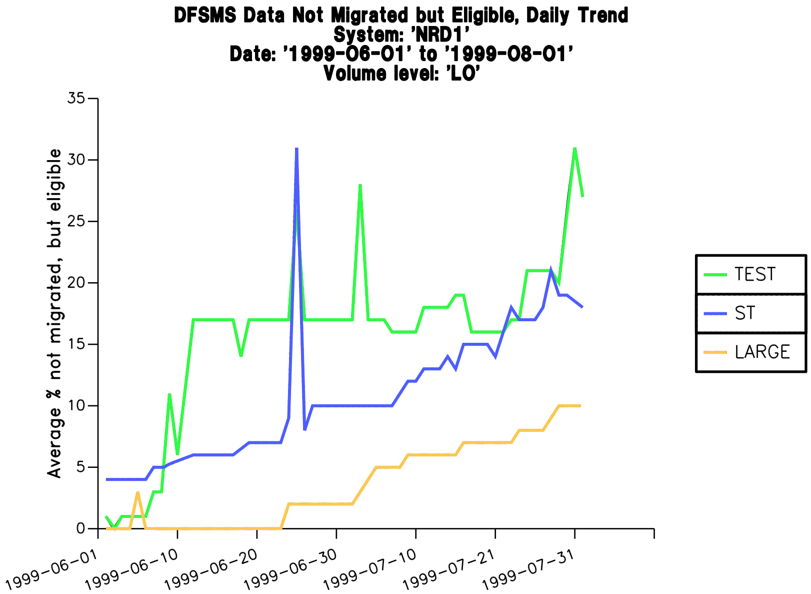

Figure 52. DFSMS Data Not Migrated but Eligible, Daily Trend report

If you find that you typically migrate all eligible data (the average percentage of data not migrated but eligible for migration is near zero), track your storage group or pool capacities through exception reports, based on whether DFSMShsm was able to meet low or target threshold value of the volume. If DFSMShsm does not meet the target threshold value, the storage group or volume is out of capacity. If you added a storage group or pool name through a lookup table, you can summarize the report at this level.

The DFSMS Storage Groups Meeting Target Threshold report tracks when the target threshold values for each storage group was met. A value of less than zero indicates that the storage group did not meet the target level.

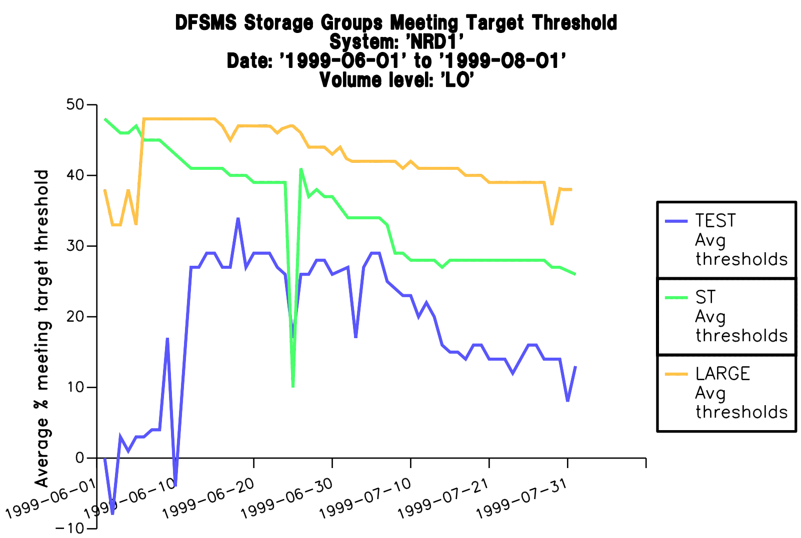

Figure 53. DFSMS Storage Groups Meeting Target Threshold report

Assuming that the threshold values are properly set (that is, a lower value would cause migration or recall thrashing), volumes, storage groups, or pools that did not meet their target should be considered as out of capacity.

When a storage group fails to meet the desired target threshold occupancy value, the storage administrator knows the number of volumes and the serial numbers of particular volumes having storage problems. The DFSMS Volumes Not Meeting Target Threshold report lists the volumes meeting or not meeting the threshold value. The right-most column is a calculated value:

```
target occupancy - occupancy after daily space management
```

This value indicates the percentage of storage occupied over the low threshold value. The report is sorted on the right-most column, with the worst volume at the top of the list.

```
                                  DFSMS Volumes Not Meeting Target Threshold
                                                System: 'MVS1'
                                               Date: '2003-12-10'


                                                                                        Occup         Occup      Threshold
                                                       Target       Trigger            before         after         met/
        Storage                                       occupancy    occupancy            dsm           dsm         not met
Volser   group                                          (%)          (%)                (%)           (%)           (%)
------  -----------------------------  -----------  -----------  -----------  -----------  -----------
TST003  TEST                                    50           70           67           67          -17
TST005  TEST                                    50           70           61           60          -10
TST004  TEST                                    50           70           54           54           -4
PRI004  ST                                      60           80           72           63           -3
TST000  TEST                                    50           70           54           52           -2
PRI006  ST                                      60           80           64           59            1
PRI008  ST                                      60           80           60           59            1
PRI013  ST                                      60           80           49           48           12
PRI010  ST                                      60           80           47           47           13
LRG000  LARGE                                   50           70           31           31           19


         IBM Z Decision Support Report: DFSMS14
```

Figure 54. DFSMS Volumes not Meeting Target Threshold report

Managing backup and migration storage

The DFSMS Migration and Backup Activities, Daily Trend report shows the number of data sets that are migrated and backed up each day. Use this report to monitor the amount of migration and backup work performed by DFSMShsm each day and to identify any extraordinarily busy days when lots of migration and backup activities occur.

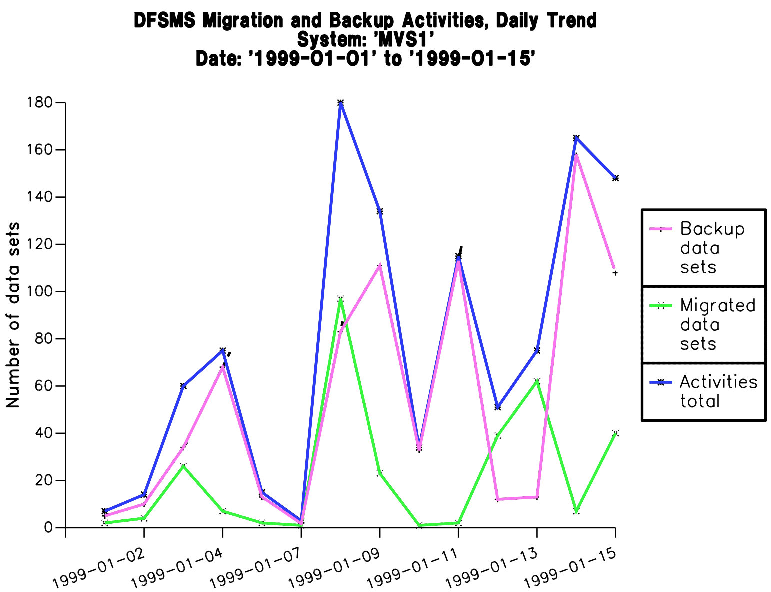

Figure 55. DFSMS Migration and Backup Activities, Daily Trend report

As storage is installed, find out which categories of data, active versus inactive, are using it and how it affects tape usage.

The DFSMS Active/Migrat/Backup Storage, Monthly Trend report tracks the total amount of space allocated by these categories:

- Space allocated

- Migration space on DASD
- Migration space on tape
- Backup space on DASD
- Backup space on tape

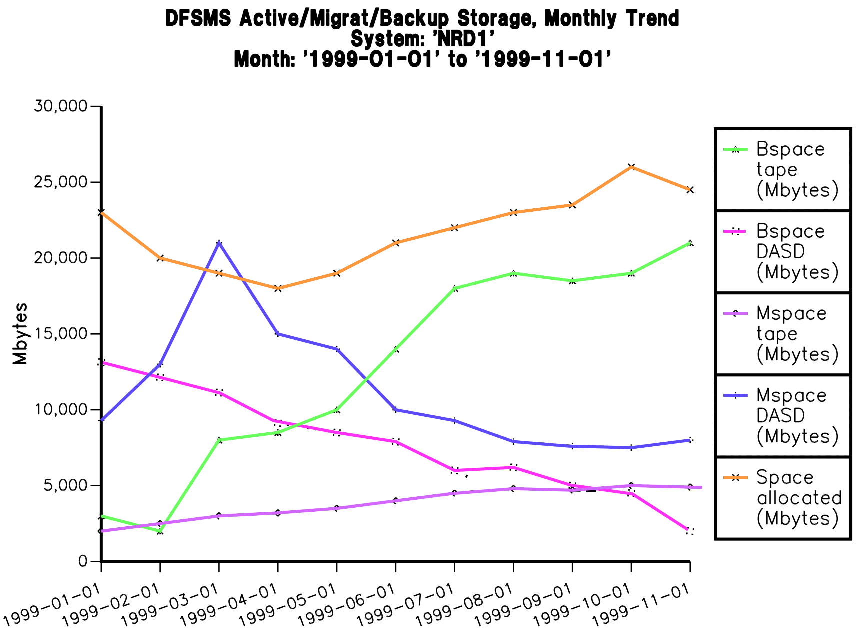

Figure 56. DFSMS Active/Migrat/Backup Storage, Monthly Trend report

Use this report to identify which storage category is increasing or decreasing the most. The report also shows you the relative size between the different categories.

Managing migration

DFSMShsm automatically migrates data from active (L0) volumes to migration level 1 (ML1) and then to migration level 2 (ML2) according to values set by the organization. You must monitor your migration volumes to ensure that the number of days that data sets reside on a particular migration volume is optimally set for the storage classes in your organization.

The DFSMS Storage Used for Migration on ML1/ML2 report shows the daily trend for the total amount of migration storage for the selected storage class. The report shows how much of this migration storage is ML1 and how much is ML2 storage.

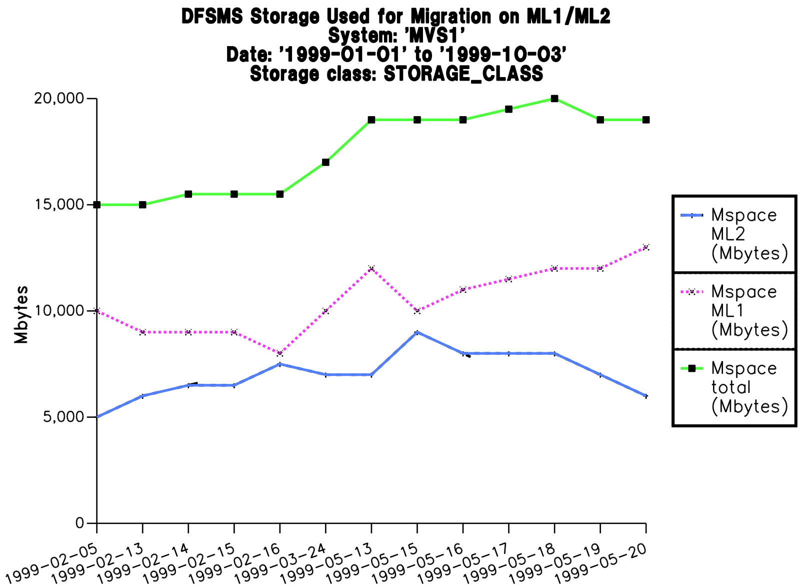

Figure 57. DFSMS Storage Used for Migration on ML1/ML2 report

Also consider monitoring the target and trigger migration values to ensure their proper setting. The DFSMS Target/Trigger Migration Values report shows the average daily trend for target and trigger occupancy percentages (low and high) thresholds for the volumes in a selected storage group. The report also shows the average occupancy percentage of data that is eligible for migration, but was not migrated because the target threshold was met during the migration process.

Managing backup data sets

DFSMShsm can back up data to either DASD or tape. You should monitor the amount of precious DASD used for backup data sets. The DFSMS Projects Using Most DASD for Backup report shows the ten projects in your organization that allocate the most storage on DASD for backup versions.

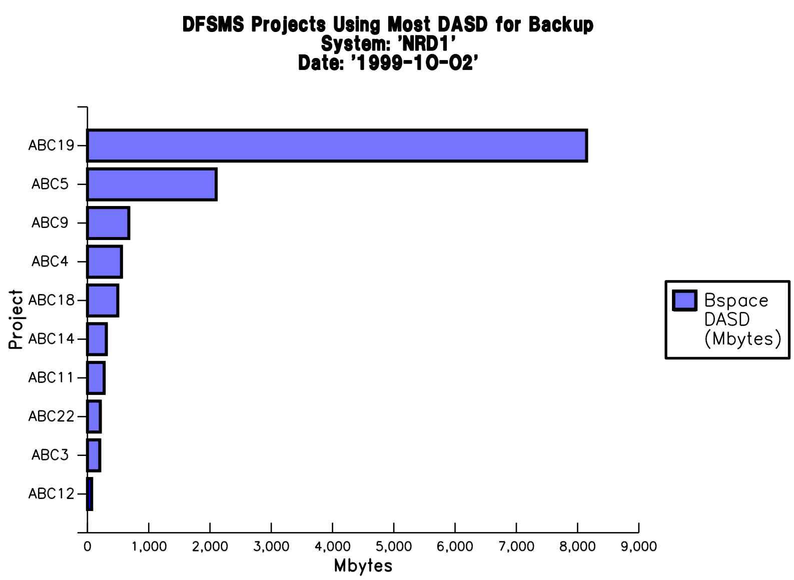

Figure 58. DFSMS Projects Using Most DASD for Backup report

Managing DFSMShsm tape usage

DFSMShsm uses tapes for migration, backup, and dump processing. The DFSMS Tape for Migration/Backup/Dump, Daily Trend report tracks DFSMShsm tape usage on a daily basis.

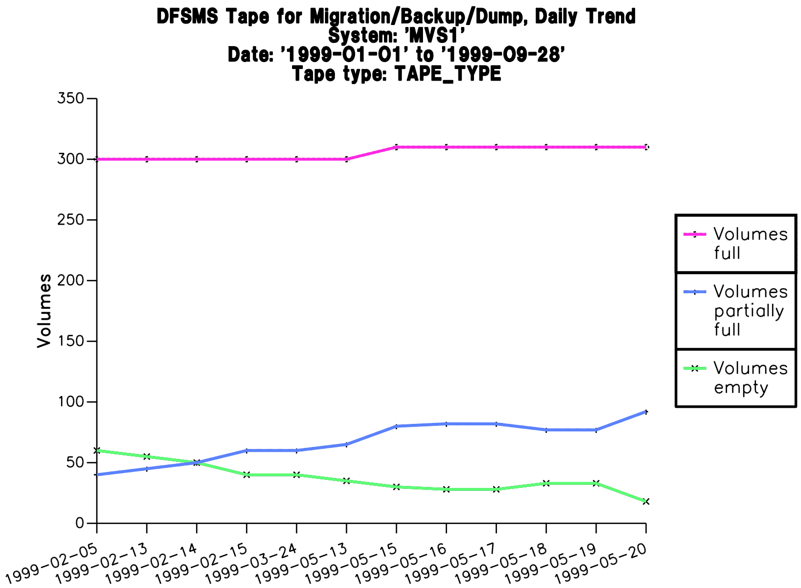

Figure 59. Migration/Backup/Dump, Daily Trend report

The report tracks the total number of tapes in the DFSMShsm library and the number of fully and partially used tape volumes. Use this information to monitor the use of tapes by DFSMShsm. You can determine the status and size of your DFSMShsm tape pool. When too many tape volumes are only partially full, you can schedule a DFSMShsm tape recycling job. Normally, DFSMShsm periodically performs this recycling job, but the period defined for it may have to be revised based on this report.

Likewise, if the number of unused tapes becomes too few, consider adding more tape volumes to the DFSMShsm tape pool.

The information on tape usage may be valuable during a conversion to system-managed storage because DFSMShsm will migrate, back up, and dump more data to tape. If the number of fully used volumes is increasing, order more tapes.

As data on the ML2 and backup volumes becomes obsolete, the percentage of valid data on the tape volumes decreases. DFSMShsm recycle processing consolidates the valid data and eliminates the superseded data on these tapes. If the number of partially used tapes is increasing, review the SETSYS ML2RECYCLEPERCENT and RECYCLEPERCENT commands in the ARCCMDxx PARMLIB member when DFSMShsm is started. This is because they control the amount of data that is valid on ML2 and backup volumes.

Managing exceptions

Problem situations occur in most organizations for a variety of reasons. IBM Z Decision Support supplies an exceptions report that can help you identify some of these potential problem situations before they cause real problems. You can then take corrective action and reduce or eliminate disruption to the user's environment.

The DFSMS DASD Exceptions report lists the exceptions that occurred during a selected period:

```
                             DFSMS DASD Exceptions
                                System: 'MVS1'
                         Date: '2000-02-13' to '2000-02-20'


            System                              Exception
Date          ID    Area      Severity  description
----------  ------  --------  --------  --------------------------------------------
.
.
2000-02-13  MVS1    STORAGE   04        Free VIRs for volume TS0004 low: 71
2000-02-14  MVS1    STORAGE   04        Free extents for volume TS0004 low: 23
2000-02-15  MVS1    STORAGE   04        Free space in volume TS0003 low: 0
2000-02-16  MVS1    STORAGE   04        Free DSCBs for volume TS0003 low: 3033
2000-02-17  MVS1    STORAGE   04        Fragment index for volume TS0003 high: 710
2000-02-18  MVS1    STORAGE   04        Free VIRs for volume TS0003 low: 72
2000-02-19  MVS1    STORAGE   04        Free extents for volume TS0003 low: 14
2000-02-20  MVS1    STORAGE   04        Free space in volume TS0002 low: 3
.
.

               IBM Z Decision Support Report: DFSMS10
```

Figure 60. DFSMS DASD Exceptions report

This report identifies potential problems with volumes. For example, the number of free extents on volume TS0004 is 23. This may indicate that there are only a few small free extents left on the volume. However, it could also mean that there are 23 large free extents, which is good. If, at the same time, you get a warning about the number of free data set control blocks (DSCBs), you may soon be in a situation when these extents cannot be used for allocation.

To prevent such a situation, use the DFSMS VTOC Information by Volume report to identify potential problem volume table of contents (VTOC) candidates. The report lists this information for each volume:

- Fragmentation index
- Number of free extents
- Number of free DSCBs
- Number of free VTOC index records (VIRs)
- Percentage of free storage
- Free space in megabytes

The following sections discuss the implications of these measurements.

```
                       DFSMS VTOC Information by Volume
                            System: 'MVS1'
                          Date: '2000-02-05'
                             Volser: 'TSOL__'


                                                               Free        Free
           Fragment       Free         Free         Free       space       space
Volser       index      extents        DSCBs        VIRs        (%)      (Mbytes)
------  -----------  -----------  -----------  -----------  -----------  -----------
TSOL01          514          257        3 028          257           24          442
TSOL02          541          199        3 346          264           18          329
TSOL03          489          171        3 255          268           19          358
TSOL04          637          225        3 063          256           16          300
TSOL05          242          100        3 158          262           19          363
        ===========  ===========  ===========  ===========  ===========  ===========
Average         485          190        3 170          261           19          358


                  IBM Z Decision Support Report: DFSMS03
```

Figure 61. DFSMS VTOC Information by Volume report

Managing free DSCB exceptions

Effective use of DASD storage requires a properly sized VTOC. The number of DSCBs in the VTOC determines the number of data sets that can reside on the volume. Use the DFSMS VTOC Information by Volume report to monitor the DSCBs for your volumes. Free storage remaining on the volume cannot be used when a VTOC is 100% full.

Managing free VIRs exceptions

Besides a properly sized VTOC, effective use of DASD storage also requires a properly sized VTOC index. The VIR, which is one of the record types in the index, contains pointer entries to each format-1 DSCB in the VTOC. So, when the VIRs are exhausted, data set allocation is inhibited, and remaining free storage cannot be used. Use the DFSMS VTOC Information by Volume report to monitor free VIR exceptions.

Managing fragmentation indexes and free extents

Because of the nature of allocation algorithms and the frequent creation, extension, and deletion of data sets, the free storage on DASD volumes becomes fragmented. This leads to inefficient use of DASD storage, an increased number of storage-related abends, and performance degradation caused by excessive DASD arm movement.

Space fragmentation on a volume is measured by a fragmentation index. The higher the value, the more severe the fragmentation. Volumes with a fragmentation index above an user-determined value are candidates for defragmentation. The DCOLLECT type V record contains the fragmentation index. You can use this data to monitor volume fragmentation and automatically schedule a DFDSS DEFRAG (or equivalent function) job. Many organizations already have a volume DFDSS DEFRAG process established through regularly scheduled batch jobs. Use the DFSMS VTOC Information by Volume report to monitor the fragmentation indexes to ensure that the DFDSS DEFRAG jobs are being run regularly and frequently enough.

Chapter 5. Analyzing Db2 performance

This chapter discusses issues that can affect the performance of your Db2 subsystem. It describes the data used by the Db2 component of IBM Z Decision Support and how you analyze that data to identify constraints and contention in your Db2 subsystem.

The System Performance feature Db2 component provides a set of predefined reports that you can use to monitor the performance of your Db2 subsystem. When analyzing performance data, consider that almost all symptoms of poor performance will be magnified when there is contention. For example, if there is a slowdown in DASD, these problems could occur:

- Bottleneck of transactions for data set activity
- I/O and locking waits
- Paging constraints

There will also be more transactions in the system. This causes greater processor overhead and greater demand for virtual storage and processor storage.

In such situations, the system shows heavy use of all its resources. However, it is actually experiencing typical system stress, and there is a constraint that has yet to be found. To find the constraint, you must find what is really affecting task life. Always start by looking at the overall system before you decide that you have a specific Db2 problem.

Within Db2, the performance problem is either poor response time or an unexpected and unexplained high use of resources. You should first check things such as total processor usage, DASD activity, and paging. You must look at the system in some detail to see why tasks are progressing slowly, or why a given resource is being heavily used. Use the z/OS System and z/OS Performance Management components to investigate the usage of the system resources .

First get a picture of task activity, list only the task traces, and then focus on particular activities. For example, specific tasks or a specific time period. For a response-time problem, you might want to look at the detailed traces of one task that is observed to be slow, a problem for which there may be several possible reasons. For example, the users might be trying to do too much work with certain tasks, work that clearly takes time, and the system simply cannot do all the work within the expected time with the available resources.

Another possibility is that the system is constrained by processor storage. Therefore, the tasks progress more slowly than expected because of paging interrupts. These constraints show up as delays between successive requests recorded in the Db2 trace.

Alternatively, many Db2 tasks could be waiting because there is contention for a particular resource. For example, there might be a wait on a particular data set or a certain application might cause many tasks to enqueue the same item.

The performance of your Db2 subsystem, and what you monitor, depends on the version of Db2 being used. This chapter describes some of the possible performance issues. For more information on Db2 and specific information for your release, refer to the appropriate version of the Db2 Universal Database for OS/390 and z/OS: Administration Guide and Reference.

The following table describes the contents of this chapter:

Table 3. Contents of Chapter 5

This section ...	Describes ...
Chapter 5, "Analyzing Db2 performance," on page 75	• The operation of a Db2 subsystem. • Db2 database organization and the Distributed Relational Data Architecture (DRDA®). • The key terms used for Db2.

Table 3. Contents of Chapter 5 (continued)

This section ...	Describes ...
"Analyzing Db2 response time" on page 82	Performance issues within a single Db2 subsystem that can affect user response time (elapsed times, I/O activities, I/O suspensions, and lock activities).
"Analyzing database buffer pools" on page 91	• How database buffer pools can affect your Db2 subsystem performance. • The IBM Z Decision Support reports that you can use to monitor the efficiency of your buffer pools.
"Analyzing EDM pools" on page 95	IBM Z Decision Support reports that you can use to monitor the efficiency of your EDM pools.
"Analyzing Db2 processor usage" on page 95	The Db2 component report that shows processor usage by Db2 system address space.
"Analyzing Db2 performance in a distributed environment" on page 96	Performance issues unique to a Db2 distributed environment.

Understanding the Db2 environment

Db2 is a relational database management system (DBMS) that runs under the z/OS operating system. A DBMS manages data stored in databases. You can use Db2 with a variety of tools to access its databases directly and you can develop application programs that access its databases.

A DBMS lets you control and use the information stored on a computer system more effectively and efficiently than you can with standard file processing. Programs that use standard file processing facilities to access and store data issue statements such as READ and WRITE to do file input and output functions. These statements invoke operating system services called access methods. An access method handles details of the physical location and structure of the data. Several access methods are available for MVS, including:

- Virtual storage access method (VSAM)
- Queued sequential access method (QSAM)
- Basic partitioned access method (BPAM)

Although an access method shields a program from the data's physical format, the program is still closely tied to the physical structure. The program has to keep track of the complete structure of the data in each record, even if it needs only a small part of the data. The access method is not aware of the content of a record so, if the record structure changes, the program must be changed even if the change has nothing to do with the program's function.

When a program uses a DBMS such as Db2, the DBMS has the complete picture of the physical and logical structure of the stored data. The application program can deal with the data on a logical level, and leave the physical data management concerns to the DBMS. The following figure shows the layers of software in a DBMS:

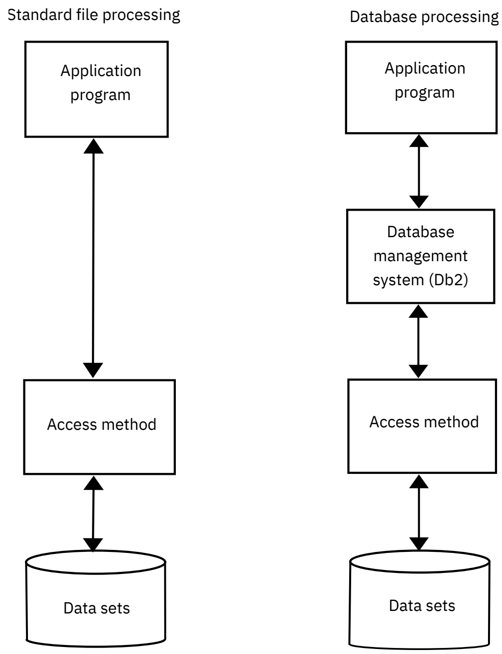

Figure 62. DBMS software

The program does not need descriptions of the physical data sets that contain database data and it does not need descriptions of any data element that it does not need to perform its task. Application programs that use a DBMS are data independent. If the physical structure of the stored data changes, no programs are affected. If the logical content of the data changes, only the programs that use the changed elements are affected. A DBMS provides more control over the integrity of the data.

A DBMS centralizes control over a large collection of data, so it coordinates access to the data by many programs concurrently, and controls what happens to them. Db2 lets more than one application access the same data at essentially the same time. This concurrent access could cause these types of problems with data integrity:

Lost updates
: Without concurrency control, two processes, A and B, might both read the same row from the database, and both calculate new values for one of its columns, based on what they read. If A updates the row with its new value, and then B updates the same row, A's update is lost.

Access to uncommitted data
: Without concurrency control, process A might update a value in the database and process B might read that value before it was committed. Then, if A's value is not later committed, but backed out, B's calculations are based on uncommitted (and possibly incorrect) data.

Unrepeatable reads
: Some processes require this sequence of events: Process A reads a row from the database, then goes on to process other SQL requests. Later, process A reads the first row again and must find the same values it read the first time. Without control, process B could have changed the row between the two reads.

To prevent these situations from occurring, Db2 uses locks to control concurrency. A lock associates a Db2 resource with an application process in a way that affects how other processes can access the same resource. The process associated with the resource is said to hold or own the lock. Db2 uses locks to ensure that no process accesses data that has been changed, but not yet committed, by another process.

Db2 also uses logging to ensure the integrity of a database. Db2 logs all activity to a data set. The active log always contains the most recent log records. The archive log contains older data, which is moved out of the active log to make room for more current data.

With these logs, Db2 ensures that it can recover from possible errors. During restart of the Db2 subsystem, Db2 processes the log in a forward direction to apply all REDO log records. Db2 then scans the log in a backward direction to apply UNDO log records for all aborted changes. So, Db2 can recover all database changes in the event of a failure and ensure the integrity of the data.

Understanding Db2 and z/OS

Db2 operates as a subsystem of z/OS. Db2 utilities run in the batch environment. Applications that access Db2 resources can run in batch, under TSO, under IMS, or under CICS.

Application programs are typically used in organizations where the same inquiry, data capture, and update tasks are performed many times a day.

Ad hoc systems, such as the Query Management Facility (QMF™), are also widely used. With QMF, a user can enter a database request at a terminal and get a response from Db2. QMF issues the instructions that Db2 needs to respond and then displays the results on the screen.

Understanding Db2 data organization

In a relational database, data is perceived to exist in one or more tables, each consisting of a specific number of columns and a number of unordered rows. Each column in a row is related in some way to the other columns. Thinking of data as a collection of tables gives you an easy way to visualize stored data and lets you explain your needs in easy-to-understand terms.

Db2 accesses data by referring to its content instead of its location or organization in storage. The rows of a relational table have no fixed order. The order of the columns, however, is always the order in which you specified them when defining the table.

A Db2 subsystem contains Db2 objects such as databases, tables, table spaces, storage groups, views, indexes, and so on.

Understanding Db2 objects

The elements that Db2 manages can be divided into two broad categories:

Data objects
Accessed under the user's direction and by which the user's data (and some system data) is organized.

System objects
Controlled and accessed by Db2.

Db2 objects follow this hierarchy:

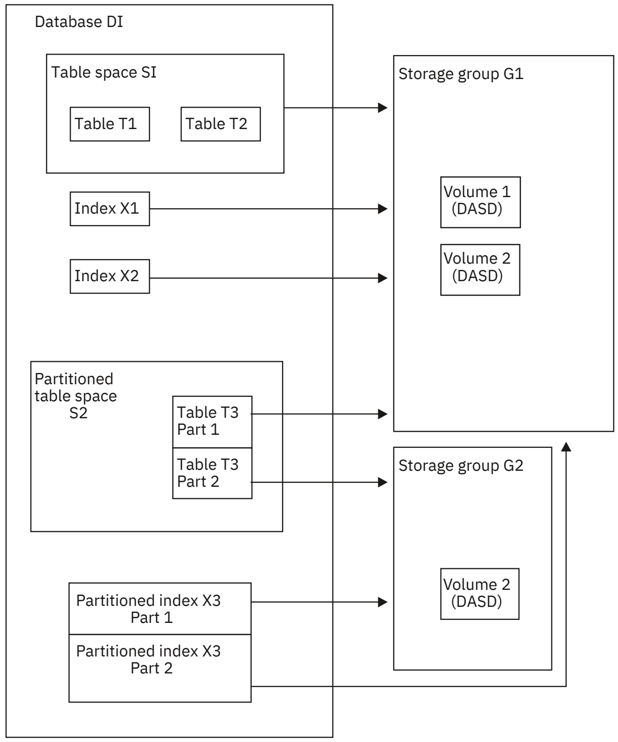

Figure 63. Db2 objects hierarchy

Db2 data objects include:

Databases

In Db2, a database is a collection of tables and associated indexes. A single database may contain all the data associated with one application or with a group of related applications. Collecting data into one database lets you start or stop access to all the data in one operation, or grant authorization for access to all the data as a single unit.

Storage groups

A Db2 storage group is a set of volumes on DASD that hold the data sets in which tables and indexes are actually stored.

Table spaces

A table space is one or more data sets in which one or more tables are stored.

Tables

All data in a Db2 database is presented in tables, which are collections of rows all having the same columns. When you create a table in Db2, you define an ordered set of columns.

Indexes

An index is an ordered set of pointers to the data in a Db2 table, stored separately from the table. Except for changes in performance, users of the table are unaware that an index is being used. Db2 decides whether or not to use the index to access the table. You can use indexes to improve performance and ensure uniqueness. In many cases, access to data is faster with an index than without.

Views

A view is an alternate way of representing data that exists in one or more tables. A view can include all or some of the columns from one or more tables. A view looks just like a table and can be used as though it were a table. Db2 does not store any data for the view itself. Only the view definition is stored in the Db2 catalog.

Db2 system objects include:

Db2 catalog

The Db2 catalog consists of tables of data about everything defined to the Db2 system. When you create, alter, or drop any object, Db2 inserts, updates, or deletes rows of the catalog that describe the object and tell how it is related to other objects. So, the catalog contains metadata, which is data about data.

Db2 directory

The Db2 directory, which contains Db2 internal control data, is used by Db2 during its normal operation. Among other things, the directory contains the database descriptors (DBDs) which is a replica of the catalog information but in a format suitable for Db2 at run time. It also contains the compiled access paths for the applications. The directory also contains recovery and utility status information.

The directory consists of a set of Db2 tables stored in four table spaces. Unlike the catalog, the directory cannot be accessed by SQL.

Active and archive logs

As data changes and significant events occur, Db2 records them in log records. In the case of failure, it uses the data in these log records to recover.

Db2 writes each log record to a pair of DASD data sets called the active log. When the active log is full, Db2 copies the contents of the active log to a DASD, mass storage subsystem (MSS), or magnetic tape data set called the archive log.

Bootstrap data set (BSDS)

The BSDS contains information required to start Db2. The BSDS contains:

- An inventory of all active and archive log data sets known to Db2
- A wrap-around inventory of all recent Db2 checkpoint activity
- The distributed data facility (DDF) communication record
- Conditional restart information
- Buffer pool specifications

Buffer pools

On systems that have the prerequisite hardware and software, Db2 maintains two levels of storage for each buffer pool:

Virtual buffer pools

These are areas of virtual storage used to temporarily store pages of table spaces or indexes. When an application program must access a row of a table, Db2 retrieves the page containing that row and places the page in a buffer. If the row is changed, the page must be written back to the table space on DASD. If the needed data is already in the buffer, the program need not wait for it to be retrieved from DASD. The result is improved performance. Db2 buffer pools are heavy users of central and expanded storage.

Hiperpools

You can use hiperspaces to extend Db2’s virtual buffer pools. Db2 cannot directly manipulate data that resides in a hiperspace, but it can transfer the data from hiperspace into a regular Db2 buffer pool much faster than it could retrieve it from DASD. Hiperpools use expanded storage or hiperspace. Hiperpools are optional.

EDM pool

The EDM pool is a pool of central storage used for database descriptors and application plans. The EDM pool contains:

- Database descriptors (DBDs)
- Skeleton cursor tables (SKCTs)
- Cursor tables (CTs) or copies of the SKCTs
- Skeleton package tables (SKPTs)
- Package tables (PTs) or copies of the SKPTs

- An authorization cache block for each plan that is currently executing.

Communications database (CDB)

The CDB, automatically created during the installation and migration processes, is a user-maintained collection of tables used by the DDF to map Db2 objects to VTAM objects. It also holds security information and communication characteristics. Db2 uses the CDB to get information about communicating with other Db2 subsystems, or other remote locations that support distributed relational database architecture (DRDA).

Data definition control support (DDCS) database

The DDCS database is a user-maintained collection of tables used by data definition control support to restrict the submission of specific Db2 database definition language (DDL) statements to selected application identifiers (plans or collections of packages).

Locks

Db2 lets more than one program access the same data at essentially the same time. This is known as concurrency. To control concurrency and prevent inconsistent data, Db2 uses locks.

Resource limit specification table (RLST)

The RLST contains the limits of the amount of time permitted for the execution of dynamic SELECT, UPDATE, DELETE, and INSERT SQL statements. You can establish a single limit for all users, different limits for individual users, or both. In Db2 Version 2 Release 3 or later, the RLST can also contain a flag indicating whether BIND should be allowed.

For complete information on Db2, refer to the *Db2 Universal Database for OS/390 and z/OS: Administration Guide and Reference*.

Understanding Db2 I/O operations

Db2 uses these read mechanisms:

Synchronous read

Normal synchronous read is used when just one or a few consecutive pages are retrieved.

Sequential prefetch

Sequential prefetch is performed concurrently with other operations of the application program. Sequential prefetch brings pages into the buffer pool before they are required, and reads several pages with a single I/O operation.

List sequential prefetch

List sequential prefetch prefetches data pages accessed through nonclustered indexes. It is also used for access by multiple indexes.

Write operations are usually performed concurrently with user requests. Updated pages are queued by data set until they are written when:

- A checkpoint is taken
- The percentage of update pages in a buffer pool exceeds the vertical deferred write threshold value
- The percentage of unavailable pages in a buffer pool exceeds the deferred write threshold value

Up to thirty-two 4KB or four 32KB pages can be written in a single I/O operation, if all pages are within a range of 150 4KB pages. A high number of updated pages written in each I/O is desirable.

Understanding distributed data

The distributed data facility (DDF) lets an application program connected to one Db2 subsystem access data at a remote relational database instance that supports IBM's Distributed Relational Data Architecture (DRDA). Alternatively, if the connection is between two Db2 subsystems, Db2 can use private connections for distributed data. DDF also allows local access of data from an application executing in a remote application requester environment that supports DRDA. The following figure gives an overview of DDF support:

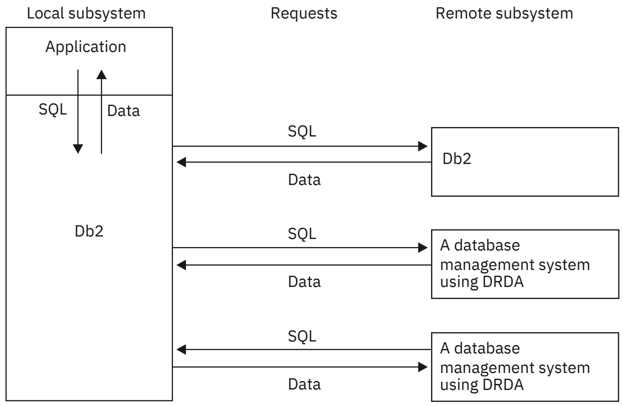

Figure 64. Overview of distributed database support

For an application process that accesses several different Db2 subsystems, you must often distinguish between local and remote subsystems. A local subsystem is the subsystem to which an application process is first connected and authorized. It is the subsystem from which the application process originates Db2 requests. All other subsystems, in this context, are remote. Remote does not necessarily mean far away. You can access distributed data among two or more Db2 subsystems running at the same user site, even on the same processor. In that case, the subsystem you are directly connected to is still called local. All others, wherever they might be, are remote.

Understanding DRDA

DRDA is IBM's architecture that lets any IBM or non-IBM product that adheres to its rules access distributed relational data. DRDA provides the necessary connectivity to and among relational database managers operating in similar or dissimilar operating environments.

DRDA describes, through protocol models, the necessary interchanges between an application and a remote relational database to perform these functions:

- Establish a connection between an application and a remote relational database (RDB).
- Bind an application's host language variables and SQL statements to a remote RDB.
- Execute SQL statements on behalf of an application in a remote RDB, returning both data and a completion indicator to the application.
- Maintain consistent logical unit of work boundaries between an application and an RDB.

All of IBM's relational database management systems support DRDA.

Analyzing Db2 response time

Response times from several areas of Db2 provide an important indication of how well Db2 is running. The Db2 component provides several predefined reports that give you the information necessary to assess the response times of your Db2 applications. These reports show the response times by application, an overview of response time, and trends in response time. You can use these reports to see if you are meeting your service-level objectives.

Measuring Db2 response time

To correctly monitor Db2 response time, you must understand how it is reported. The following figure shows how some of the main measures relate to the flow of a transaction.

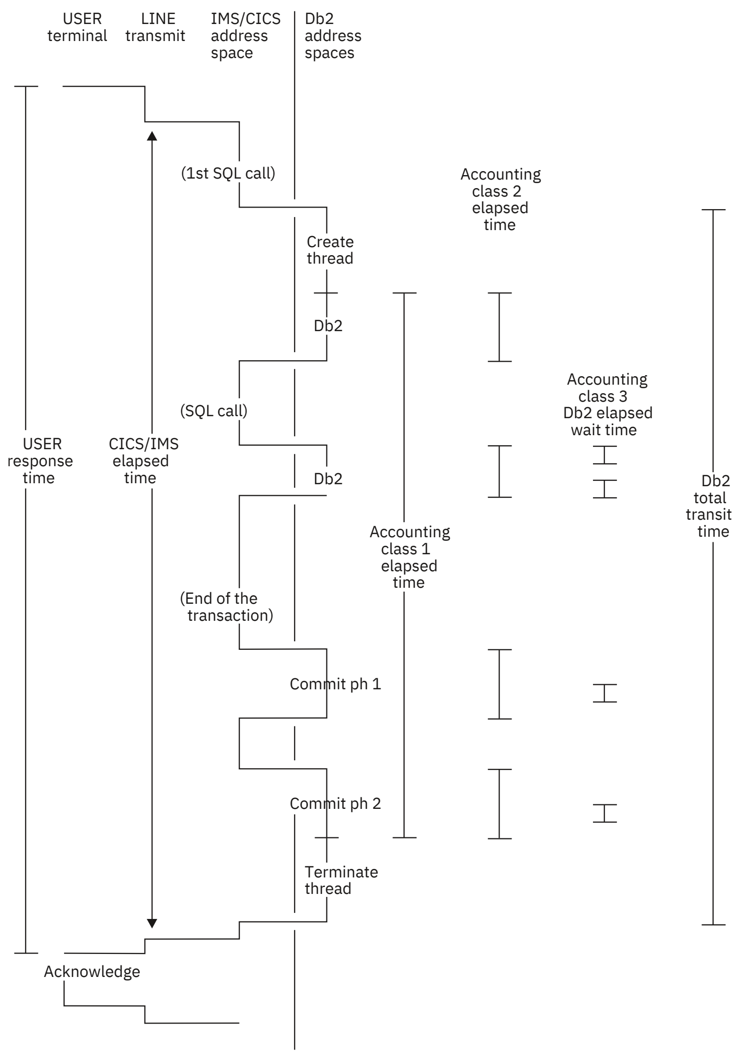

Figure 65. Transaction response times

These times can be distinguished:

- User response time

  This is the response time, from the moment the user presses the Enter key until the first response appears back at the terminal.

- Db2 accounting elapsed times

  These times are collected in the accounting records. They are taken over the accounting interval between the point where Db2 starts to execute the first SQL statement, and the point preceding thread termination or reuse by a different user (signon). The interval excludes the time spent in creating and terminating a thread.

  Db2 records these elapsed times:

  Class 1 elapsed time
  This time is always presented in the accounting record and shows the duration of the accounting interval. It includes the time spent in Db2 and time spent in the front end.

  Class 2 elapsed time
  Class 2 elapsed time, produced only if the accounting class 2 trace is active, counts only the time spent in the Db2 address space during the accounting interval. It represents the sum of the times from any entry into Db2 until the corresponding exit from Db2. It is also referred to as the time spent in Db2. If class 2 is not active for the duration of the thread, the class 2 elapsed time may not reflect the entire Db2 time for the thread, but only the time when the class 2 trace was active.

Class 3 elapsed time

Class 3 elapsed time, produced only if the accounting class 3 trace is active, is elapsed wait time in Db2. Possible causes of wait time include:

- Synchronous I/O suspension time
- Lock/latch suspension time
- Asynchronous read suspensions
- Asynchronous write suspensions
- Service task suspensions
- Archive log mode time

- Db2 total transit time

  Regarding an SQL transaction or query, the total transit time is the elapsed time from the beginning of create thread, or the signon of another authorization ID when reusing the thread, until either the end of the thread termination, or the signon of another authorization ID.

  When a thread is reused, the transit time may include idle time. Unlike the accounting class 1 and 2 times, it includes time spent in create and terminate thread.

The following figure shows the breakdown of elapsed times:

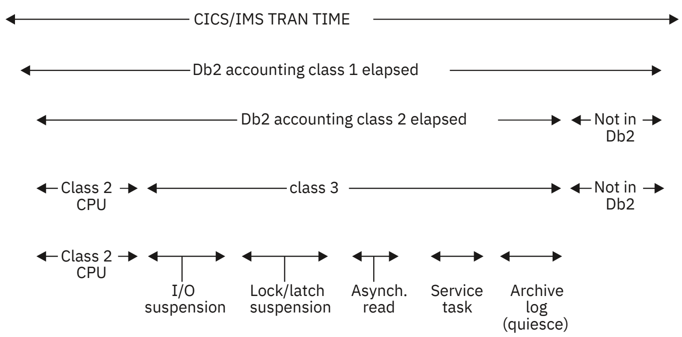

Figure 66. Accounting response time diagram

The elapsed time spent in each of the processing areas provides the basis for beginning the tuning of an application. You can determine the IMS/CICS transaction elapsed time from reports created by the CICS Performance feature and the IMS Performance feature of IBM Z Decision Support. You can determine the accounting class 1, class 2, and the total transit time from the Db2 component.

To have correct average times, accounting class 2 and class 3 traces must be active for the entire accounting period. If these classes are started and stopped within an application run, Db2 records only the time when the classes are active, thereby creating a discrepancy with the real Db2 times.

You can analyze the different areas of elapsed times, make comparisons, and determine the possible problem area more easily.

Analyzing elapsed times

You should start the analysis of your Db2 subsystem by determining if you are meeting service-level objectives. The DB2 Application Response Times, Detail report shows the average, maximum, and minimum elapsed times for each application.

```
                     DB2 Application Response Times, Detail
                         System: 'MVS1'  DB2 ID: 'DB2A'
                      Date: '1999-11-19'  to  '2000-03-09'


                             Elapsed    Elapsed   Elapsed
Application                  seconds    seconds   seconds       Commit   Abort
   name      Profile           avg        max       min         count    count
------------ ----------- ---------- ----------  -------- ------------  ------
P.R.         QUERY             9.39      23.27      0.16        13218      18
QMF          QUERY             7.83      25.20      0.13         7876       8
SPUFI RR     QUERY             4.60      21.14      0.31           73       5


                IBM Z Decision Support Report: DB201
```

Figure 67. DB2 Application Response Times, Detail report

If an application is not receiving the appropriate service level, you must determine why the elapsed time is high. The DB2 Transaction Statistics, Detail report in the following figure shows the response-time components for each Db2 plan.

```
                                         DB2 Transaction Statistics, Detail
                                          System: 'MVS1'  DB2 ID: 'DB2A'
                                        Date: '2000-01-11'  to  '2000-03-09'


                                                     DB2                Lock    Service  Archive  Archive
          Elapsed    CPU    DB2 CPU  CPU(SRB)  CPU(SRB)  IO wait   latch     wait   logwait  logread   Drain
          seconds  seconds  seconds  seconds   seconds  seconds  seconds  seconds  seconds  seconds  seconds      Commit   Abort
DB2 plan    avg      avg      avg      avg       avg      avg      avg      avg      avg      avg      avg         count    count
------------------ -------- --------  --------  -------- -------- -------- -------- -------- -------- --------  ---------  ------
DB2PM211     0.88    0.006    0.004     0.000     0.000    0.000    0.000    0.000    0.000    0.000    0.000      50983       0
DRLPLAN     21.68    7.332    4.019     0.009     0.000    0.017    0.019    0.000    0.000    0.000    0.000       7098     123
QMF310      12.90    0.680    0.165     0.004     0.000    0.001    0.001    0.000    0.000    0.000    0.000       4227      62
DSNUTIL      0.46    0.028    0.000     0.003     0.000    0.000    0.000    0.000    0.000    0.000    0.000       1391      11
DISTSERV  2128.04    0.647    0.004     1.164     0.002    0.001    0.000    0.000    0.000    0.000    0.000         95       0
DSNESPRR     6.15    0.115    0.002     0.002     0.000    0.000    0.000    0.000    0.000    0.000    0.000         55       4
                                                                                                          =========  ======
                                                                                                              63849     200

                                IBM Z Decision Support Report: DB204
```

Figure 68. DB2 Transaction Statistics, Detail report

To determine what is causing delays for the plan to which the application belongs, examine these columns in the report:

Elapsed seconds avg

This is the average class 1 elapsed time. Compare this with the CICS or IMS transit times. You can determine the CICS and IMS transaction elapsed times from reports created by the CICS Performance feature and the IMS Performance feature of IBM Z Decision Support.

Differences between these CICS or IMS times, and the Db2 accounting times arise mainly because the Db2 times do not include:

- Time before the first SQL call
- Db2 create thread
- Db2 terminate thread

Differences can also arise due to thread reuse in CICS or IMS, or through multiple commits in CICS. If the class 1 elapsed time is significantly less than the CICS or IMS time, check reports that show CICS transaction performance or application response times, or IMS transaction performance. For a description of the CICS or IMS reports available through IBM Z Decision Support, refer to the CICS Performance Feature Guide and Reference or the IMS Performance Feature Guide and Reference.

Elapsed time can occur:

- In Db2 in signon, create, or terminate thread
- Outside Db2 in CICS or IMS processing

For CICS, the transaction could have been waiting outside Db2 for a thread.

Db2 CPU seconds avg
This is the average processor time spent in Db2. The problem might be due to Db2 or IMS Resource Lock Manager (IRLM) traces, or to a change in access paths. When using CPU time for analyzing Db2 performance, use only TCB time.

I/O wait seconds avg
This is the average I/O wait time during the reporting interval. This value includes:

Synchronous I/O suspension time
This is the total application wait time for synchronous I/Os. Check the total number of synchronous read I/Os reported.

If the number of read I/Os is higher than expected, check for:

- A change in the access path to data. Here, the number of GETPAGE requests is higher than expected.

  A GETPAGE is a logical read of one 4K page. If the page is in the buffer pool, it is retrieved from the buffer pool, so DASD I/O is unnecessary. It is always desirable to have required data in the buffer pools.
- Changes in the application.
- A system-wide problem in the database buffer pool.
- A system-wide problem in the EDM pool.

If I/O time is greater than expected, and not caused by more read I/Os, check for:

- Synchronous write I/Os.
- I/O contention (usually each synchronous read I/O typically takes from 15 to 25 milliseconds, depending on the DASD device).

Asynchronous read suspension time
This is the accumulated wait time for read I/O done under a thread other than the current thread:

- Sequential prefetch
- List prefetch
- Sequential detection
- Synchronous read I/O performed by a thread other than the one being reported

Asynchronous write suspension time
This is the accumulated wait time for write I/O done under a different thread:

- Asynchronous write I/O
- Synchronous write I/O performed by a thread other than the one being reported

Lock latch seconds avg
This is the average time spent in lock/latch management within Db2. This shows contention for Db2 resources. If contention is high, check the DB2 Transaction Statistics, Detail report and the DB2 Database Service Statistics, Detail report .

Service wait seconds avg
This is the average wait time due to synchronous unit switching, which lets Db2 switch from one execution unit to another. For example, waits for opening data sets are included.

Archive log wait seconds avg
This is the average time the thread was suspended due to ARCHIVE LOG MODE(QUIESCE) command processing.

To have correct average times, accounting classes 2 and 3 must be active for the entire accounting period. If these classes are started and stopped within an application run, Db2 records only the time when the classes are active, thereby creating a discrepancy with the real Db2 times.

You can also look at the distribution of response-time components by application profiles. The DB2 Application Profiles, Trend report shows the response-time components for a specific profile as a percentage of the total elapsed time (class 1) over a specific time period:

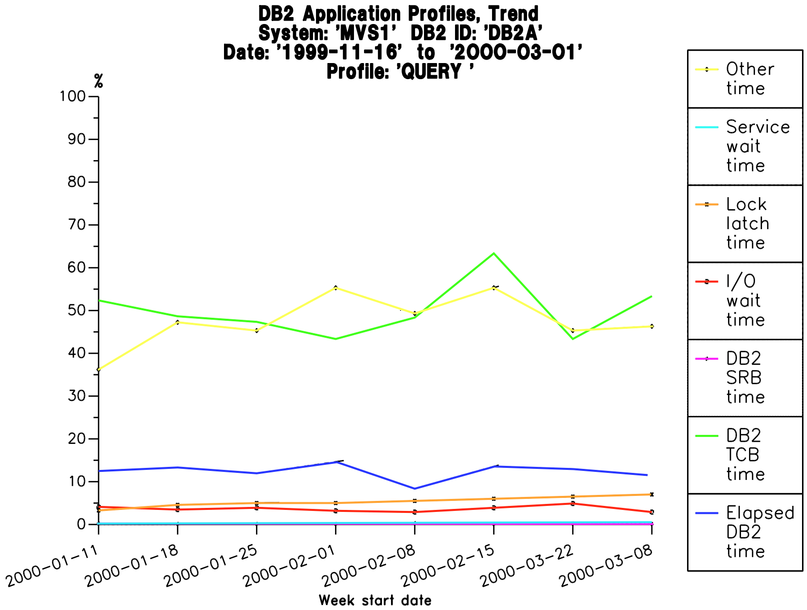

Figure 69. DB2 Application Profiles, Trend report

You can improve the response time and throughput of your Db2 applications by reducing:

- The number of I/O operations
- The time needed to perform I/O operations
- The amount of processor resources consumed

The following sections describe ways to accomplish these goals.

Analyzing I/O activities

You can improve the performance of your Db2 applications by reducing the number of I/O operations. Possible reasons why unnecessary I/O operations are occurring include:

- An application is not using indexes effectively.

  Using multiple indexes may reduce the number of I/O operations, because the indexes are scanned and the answer set of record identifiers is found before table pages are read. Only the rows of the table that answer the query are accessed. By retrieving only those rows necessary to satisfy a query, you can minimize unnecessary I/O.

- Inefficient access paths are selected for subsequently bound application plans and packages.

  This problem especially occurs on:

  – Table spaces that contain frequently accessed tables
  – Tables involved in a join or sort
  – Tables against which SELECT statements having many search arguments are performed
  – Tables with many rows

  Use the Db2 RUNSTATS utility to collect statistics about Db2 objects. Db2 uses these statistics during bind to choose the path in accessing data. If you do not use the Db2 RUNSTATS utility, Db2 could select an inefficient access path, resulting in unnecessary I/O operations.

- Not enough free space is allocated.

  When data is loaded, PCTFREE reserves a percentage of each page as free space, which is used later when inserting or updating data. If no free space is available to hold the additional data, it must be stored on another page. When several records are physically located out of sequence, performance suffers.

By specifying sufficient free space, you gain certain advantages. However, there are a few disadvantages. The advantages are:

- Faster placement of row by space algorithm
- Better clustering of rows (giving faster access)
- Fewer overflows (overflows occur only with variable-length rows and in one level)
- Less frequent REORGs needed
- Less information locked in a page
- Fewer index page splits

The disadvantages are:

- More DASD occupied
- Less information per I/O
- More pages to scan
- Possibly more index levels

You should weigh the advantages against the disadvantages for your particular organization.

- Buffer pools are not large enough for your workload.

  Considering the central and expanded storage that you have available, you could improve the performance of your applications and queries by making the buffer pool as large as possible. Larger buffer pool sizes result in fewer I/O operations, and so, faster access to your data. A large buffer pool can also prevent I/O contention for the most frequently used DASD volumes. When a Db2 sort is used during a query, a large buffer pool reduces contention on devices containing the temporary table spaces.

Analyzing I/O wait time

Slow I/O can also affect your Db2 response time. Possible causes of slow I/O include:

- Insufficient temporary table spaces.

  If your applications require large concurrent sorts and you do not have enough temporary table spaces allocated, response time suffers.
- Frequently used data sets located on slow DASD devices.
- Frequently used data sets located on the same DASD device.
- Insufficient primary allocation for frequently used data sets (that is, data sets in multiple extents).

You must monitor the average I/O elapsed time for read I/O and write I/O. A high average I/O elapsed time indicates that I/O contention is occurring. The requested I/O is accessing data from a busy data set, volume, or control unit and is continually being suspended.

For a volume or control unit level I/O problem, use the z/OS Performance Management (MVSPM) component to determine which component in the path is the bottleneck. If the problem is at this level, the z/OS system programmers should be involved.

High average I/O elapsed times may also occur when there is heavy processor contention, without an existing I/O contention problem.

The I/O wait time is the total application wait time for Db2 I/O operations that are related to synchronous read I/O, asynchronous reads, and asynchronous writes. An unexpectedly high I/O suspension time may be associated with higher synchronous read I/Os and synchronous write I/Os.

The DB2 Buffer Pool Statistics, Detail report shows the number of reads and writes performed on each buffer pool.

```
                             DB2 Buffer Pool Statistics, Detail
                               System: 'MVS1'  DB2 ID: 'DB2A'
                          Date: '2000-01-19'  Period(s): ('PRIME')


                                                    Seq
                                          Seq     prefetch    List
 BP                 Get       Sync     prefetch    pages    prefetch    Buffer     Pages      Async  Dataset
 ID Time           pages      reads    requests    read     requests    updates    written    writes  opens
--- ----- ----------- --------- --------- --------- --------- --------- --------- --------- -------
  0 08.00      502272       576      1422      8412         2    167936      8379       473      33
    09.00     2759424       879     34178      4330        19    509207     15459       955      22
    10.00     1194496      2386     41965     21670         1    172015     27516       994       0
    11.00      501504      3322      6913     16254        80    173381     17154       769       9
    12.00     2695424      1936     33497      3612        16    209910      4994       441       4
    13.00      400896       409     17455      4320        52     35425      8694       324       0
    14.00     2403072       758     17160      1007         5     68959      2631       320      43
    15.00     2571264      1588      8082      6592        23    148817      9707       785     134
    16.00     1404928       325      1768      1761        23    108096      4755       517     102
          ----------- --------- --------- --------- --------- --------- --------- --------- -------
 BP0 tot:    14433280     12179    162440     67958       221   1593746     99289      5578     347

          =========== ========= ========= ========= ========= ========= ========= ========= =======
   Total:    14433280     12179    162440     67958       221   1593746     99289      5578     347

                      IBM Z Decision Support Report: DB215
```

Figure 70. DB2 Buffer Pool Statistics, Detail report

Check the average number of synchronous read I/Os . If the I/O suspension time is high and the average number of synchronous read I/Os is higher than expected:

- There may have been a change in the access path for the transaction. For example, the access path was through an index, and now is through an alternate index, causing more synchronous read I/Os. The change in access path may also be reflected by an increase in the average number of GETPAGE requests for a transaction. Check on the access path.
- The number of I/Os may be higher than expected if Db2-enforced referential integrity is used by the application.
- There could be a system-wide EDM pool problem, with the EDM pool being too small.
- There could be a system-wide buffer pool problem, with the buffer pool being too small.

  Some prefetched pages could be written over in the buffer pool before an application has a chance to process them, if the buffer pool is not large enough. The application is then forced to read in the pages synchronously.

If the I/O suspension time is high and the average number of synchronous read I/Os is not higher than expected:

- Check the number of synchronous write I/Os. Ideally, there would be no synchronous write I/Os.
- High I/O suspension times with no corresponding increase in either read I/Os or write I/Os indicate that either I/O contention or processor contention is occurring.
  - If the problem is I/O contention, the application is accessing data from a busy data set, volume, or control unit and is continually being suspended.
  - If the problem is processor contention, then after a read I/O or write I/O completes, the application is not being dispatched until much later. Check the MVS dispatching priority of the applications.
- Applications might be waiting due to log buffer force-write.

  The wait time that the application incurs when writing the log buffers to the active log is included in the I/O suspension elapsed time. This occurs at commit time, when the log buffers are forced to the active log. For two-phase commit, the application is forced into waits for the I/O to the active log to complete twice. For one-phase commit, only one wait is incurred.

  The DB2 System Statistics, Overview report shows log manager efficiency:

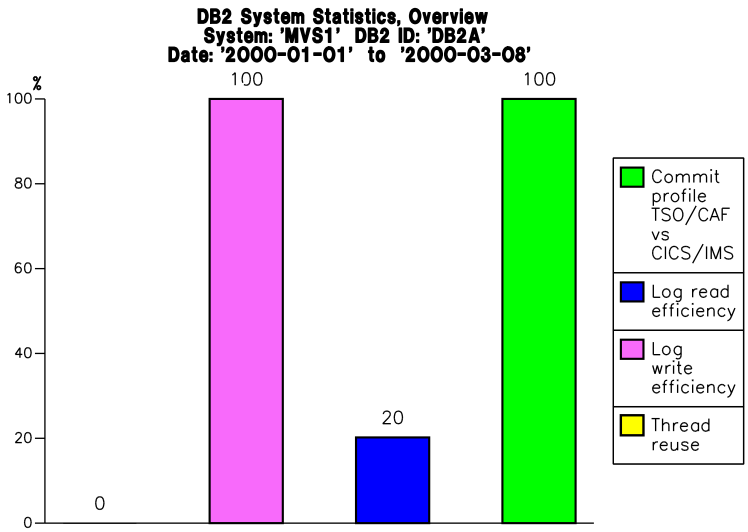

Figure 71. DB2 System Statistics, Overview report

The log write efficiency is the percentage of total log writes that were force writes. The log read efficiency is the percentage of the total number of log reads from the log output buffers.

Ensure that the active log is on a DASD path that is not subject to I/O contention. Check the elapsed time for a log write. If it appears high, check the path usage by using reports from the z/OS Performance Management (MVSPM) component.

Analyzing lock activities

Db2 allows more than one application process to access the same data at essentially the same time. This is called concurrency. To control such undesirable effects as lost updates, access to uncommitted data, and unrepeatable reads, concurrency must be controlled. Db2 uses locks to provide concurrency control. A lock associates a Db2 resource with an application process in a way that affects how other application processes can access the same resource. The application process associated with the resource is said to hold or own the lock. Db2 uses locks to ensure that no application process accesses data that has been changed, but not yet committed, by another application process.

If an application attempts to access a resource that is already locked by another application, Db2 suspends the application. Lock suspension time increases the response time of an application.

If the lock/latch suspension time is significant, check the number of suspensions, whether the suspensions were lock, latch or other, and any timeout/deadlock occurrences. The DB2 Database Services Statistics, Detail report shows the number of lock requests, deadlocks, lock suspensions, lock timeouts, and lock escalations.

DB2 Database Services Statistics, Detail
System: 'MVS1' DB2 ID: 'DB2A'
Date: '2000-01-19' Period(s): ('PRIME')

Time	SQL stmts DML	SQL stmts DDL	SQL stmts control	Plan alloca- tions	Binds (plans + pkgs)	Rebinds (plans + pkgs)	Lock requests	Dead- locks	Lock suspen- sions	Lock time- outs	Lock escala- tions
08.00	207018	0	0	129	0	0	170576	0	0	0	0
09.00	1380067	23	5	221	0	0	711376	0	2	0	0
10.00	158480	0	0	97	0	0	48496	0	0	0	0
11.00	198293	12	3	233	0	0	167216	1	4	0	0
12.00	1105511	29	0	197	0	0	1024144	0	23	5	0
13.00	15191	0	0	73	0	0	138800	0	13	8	0
14.00	49482	257	0	166	0	0	1600352	0	14	9	0
15.00	185172	182	38	244	0	0	1883536	1	20	2	0
16.00	146239	231	8	90	0	0	915632	0	0	0	0
	3445453	734	54	1450	0	0	6660128	2	76	24	0

IBM Z Decision Support Report: DB208

Figure 72. DB2 Database Services Statistics, Detail report

If there have been many lock suspensions, determine the reason for the suspensions and on which Db2 object the suspensions occurred. After making these determinations, you can consider solutions based on the type of Db2 object and the SQL activity of the applications being suspended.

If there have been timeouts or deadlocks, then there is a contention problem where locks are not being released quickly enough before the IRLM timeout cuts in. You must determine which applications were holding the locks required.

If there has been lock escalation, either shared or exclusive, then this also must be investigated. Lock escalation should occur infrequently, if at all, and should be avoided if application concurrency is desired. Once lock escalation occurs, you can expect to see timeouts occurring for any applications accessing the table depending on the type of lock escalation, whether shared or exclusive.

Once you determine which applications are involved in lock contention, you can use several methods to reduce the locking problems, including changing the lock type on a table, increasing the amount of free space per page to reduce lock contention problems, adding more commits to an application, or changing the application. Also, you can perform system-wide tuning of IRLM.

Note: The number of locks taken might be higher than expected if Db2-enforced referential integrity is used in the application.

Analyzing database buffer pools

Buffer pools are areas of virtual storage that temporarily store pages of table spaces or indexes. When an application program accesses a row of a table, Db2 searches for that page in the buffer pool. If the requested data is already in a buffer pool, the application program need not wait for it to be retrieved from DASD. Avoiding the I/O needed to retrieve data from DASD results improves performance.

If the row is changed, the data in the buffer pool must be written to DASD eventually. That write operation might be delayed until Db2 takes a checkpoint, or until Db2 must use the space for a more frequently used page. Until that time, the data can be read or changed without a DASD I/O operation.

You can change the size and other characteristics of a buffer pool at any time while Db2 is running, using the ALTER BUFFERPOOL command. You can also get online statistics of your Db2 buffer pools using the DISPLAY BUFFERPOOL command. These commands are available only with Db2 Version 3 Release 1.

Depending on your hardware and software configuration, Db2 can maintain two levels of storage for each buffer pool:

- The virtual buffer pool, the first level of storage, is backed by central storage, expanded storage, or auxiliary storage.
- The hiperpool, the second level of storage, uses expanded storage or hiperspace. Hiperpools are optional. Hiperpools are available only for Db2 Version 3 Release 1.

Virtual buffer pools hold the most frequently accessed data. Hiperpools serve as a cache for data that is accessed less frequently. When a row of data is needed from a page in a hiperpool, the entire page is read into the corresponding virtual buffer pool.

Because DASD read operations are not required for accessing data that resides in hiperspace, response time is shorter than for DASD retrieval. Using hiperpools can reduce transaction and query response times.

Backing buffer pools with hiperpools also reduces central storage contention between buffer pools and other virtual resources, because hiperspaces are backed only by expanded storage.

A database buffer pool can have these page types:

In-use pages
: These pages are currently being read or updated. The data they contain is unavailable for use by other applications.

Updated pages
: These are pages whose data has been changed but which have not yet been written to DASD.

Available pages
: These pages can be considered for new use, to be overwritten by an incoming page of new data. Both in-use and updated pages are unavailable in this sense; they are not considered for new use.

Db2’s use of a buffer pool or hiperpool is governed by several preset values called thresholds. Each threshold is a level of use which, when exceeded, causes Db2 to take some corrective action. The level of use is usually expressed as a percentage of the total size of the buffer pool or hiperpool.

Db2 has two types of thresholds:

Fixed thresholds

You cannot change these thresholds. Monitoring buffer pool usage includes noting how often those thresholds are reached. If they are reached too often, the remedy is to increase the size of the buffer pool or hiperpool. Increasing the size, though, can affect other buffer pools, depending on the total amount of space available for your buffers.

The fixed thresholds are more critical for performance than the variable thresholds are. Generally, you want to set buffer pool sizes large enough to avoid reaching any of these thresholds, except occasionally, when performance is not critical.

The fixed thresholds are:

Immediate write threshold (IWTH) - 97.5%
: This threshold is checked whenever a page is to be updated. If it has been exceeded, the updated page is written to DASD as soon as the update is completed. The write is synchronous with the SQL request; that is, the request waits until the write has been completed and the two operations are not carried out concurrently. Avoid reaching this threshold, because it has a significant effect on processor usage and on I/O operations.

Data management threshold (DMTH) - 95%
: This threshold is checked before a page is read or updated. If the threshold has been exceeded, Db2 accesses the page in the buffer pool once for each row that is retrieved or updated in that page.

Sequential prefetch threshold (SPTH) - 90%
: When this threshold is reached, sequential prefetch is inhibited until more buffers become available. Operations that use sequential prefetch, such as those using large and frequent scans, are affected. Avoid reaching this threshold, because it seriously limits performance.

Deferred write threshold (DWQT) - 50%
: This threshold is a percentage of the virtual buffer pool that might be occupied by unavailable pages, including both updated pages and pages in use. If the percentage of unavailable pages in the virtual buffer pool exceeds the threshold, Db2 schedules write operations for enough data sets to decrease the number of unavailable buffers to 10% below the threshold.

Vertical deferred write threshold (VDWQT) - 10%
This threshold is expressed as a percentage of the virtual buffer pool that might be occupied by updated pages from a single data set. If the percentage of updated pages for the data set exceeds the threshold, Db2 schedules writes for that data set.

Variable thresholds

You can change some thresholds directly, using the ALTER BUFFERPOOL command. These variable thresholds are available only with Db2 Version 3 Release 1:

Sequential steal threshold
This threshold is a percentage of the virtual buffer pool that might be occupied by sequentially accessed pages. If the threshold is exceeded, Db2 tries to steal a buffer holding a sequentially accessed page rather than one holding a randomly accessed page.

Hiperpool sequential steal threshold (HPSEQT)
This threshold is a percentage of the hiperpool that might be occupied by sequentially accessed pages. The effect of this threshold on the hiperpool is essentially the same as that of the sequential steal threshold on the virtual pool.

Virtual buffer pool parallel sequential steal threshold (VPPSEQT)
This threshold is a portion of the virtual buffer pool that might be used to support parallel I/O operations.

Deferred write threshold (DWQT)
This threshold is a percentage of the virtual buffer pool that might be occupied by unavailable pages, including both updated pages and pages in use. If the percentage of unavailable pages in the virtual buffer pool exceeds the threshold, Db2 schedules write operations for enough data sets to decrease the number of unavailable buffers to 10% below the threshold.

Vertical deferred write threshold (VDWQT)
This threshold is expressed as a percentage of the virtual buffer pool that might be occupied by updated pages from a single data set. If the percentage of updated pages for the data set exceeds the threshold, Db2 schedules writes for that data set.

Use the DB2 Buffer Pool Statistics, Overview report to analyze the efficiency of your buffer pools:

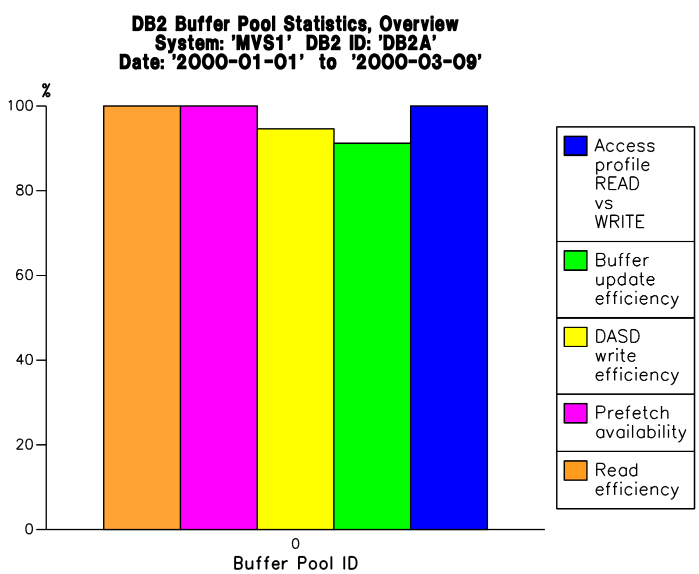

Figure 73. DB2 Buffer Pool Statistics, Overview report

The read efficiency shows the hit ratio within the buffer pool (that is, how often Db2 finds the required page in the buffer pool). The prefetch availability is the availability of the sequential prefetch mechanism. This value should always be close to 100%. The DASD write efficiency shows the percentage of the total pages written that were synchronous and asynchronous write operations. The access profile shows how much of the Db2 load comes from logical reading (GETPAGE requests). A high value indicates high logical reading activity. A low value indicates high logical update activity. Use this report to determine how your buffer pools are used.

The DB2 Buffer Pool Exceptions report shows you when problems occur with your buffer pools:

```
                           DB2 Buffer Pool Exceptions
                         System: 'MVS1'  DB2 ID: 'DB2A'
                      Date: '2000-01-01'  to  '2000-03-09'


                                Defer
                                write      Sync     Seq        No
                          BP    thrsh      write  prefetch    write      No
Date        Time          ID   reached     I/Os   disabled    engine  workfile
----------  --------      --  --------  --------  --------  --------  --------

2000-01-14  07.56.40       0         0       231         0         0         0
            14.25.24       0         8         0         0         0         0
            16.24.49       0         2         0         0         0         0
                              --------  --------  --------  --------  --------
       2000-01-14 total:            10       231         0         0         0


2000-01-19  10.28.21       0       157         0         0         0         0
            11.59.53       0        53         0         0         0         0
            13.57.33       0        59         0         0         0         0
            14.28.55       0         3         0         0         0         0
            15.27.05       0        18         0         0         0         0
            15.57.03       0        24         0         0         0         0
                              --------  --------  --------  --------  --------
       2000-01-19 total:           314         0         0         0         0


2000-03-02  13.09.09       0       182         0         0         0         0
            13.39.09       0        83         0         0         0         0
            14.09.22       0        87         0         0         0         0
            14.40.34       0        22         0         0         0         0
            16.39.14       0         2         0         0         0         0
            22.07.14       0        15         0         0         0         0
            22.40.31       0        13         0         0         0         0
                              --------  --------  --------  --------  --------
       2000-03-02 total:           404         0         0         0         0

                              ========  ========  ========  ========  ========
                  Total:           728       231         0         0         0


                 IBM Z Decision Support Report: DB219
```

Figure 74. DB2 Buffer Pool Exceptions report

The report shows the number of times the deferred write threshold was reached for the buffer pool.

The report also shows the number of immediate (synchronous) write I/Os for the buffer pool. Ideally, there should be no synchronous write I/Os. A synchronous write I/O can occur:

- At the checkpoint when updated pages are scheduled to be written. If a page is being updated when a checkpoint occurs, and at the next checkpoint, it is also being updated, then the application updating the page is forced to do a synchronous write I/O. Look at the DB2 System Parameters report to determine the checkpoint frequency and then determine the number of checkpoints that occurred in the reporting interval. If the checkpoints were occurring more than once every fifteen minutes and it is a normal workload that is occurring, the checkpoint frequency should be decreased.
- Synchronous writes also occur when the buffer pool immediate write threshold of 97.5% is reached. This threshold should rarely be reached, if at all, and indicates that the buffer pool size should be increased.

The report also shows the number of times the sequential prefetch was disabled because of unavailable buffers for the buffer pool, the number of times no write engine was available for the asynchronous write I/O for the buffer pool, and the number of times the work files required were too many and could not be created because of insufficient buffer pool resources.

The exceptions shown in this report all affect the efficient operation of buffer pools. When problems occur in the buffer pools, the applications experience delays in response time.

Analyzing EDM pools

You should design the EDM pool to contain:

- The CTs, PTs, and DBDs in use
- The SKCTs for the most frequently used applications
- The SKPTs for the most frequently used applications
- The DBDs referred to by these applications
- The cache blocks for your plans that have caches

By designing the EDM pool this way, you can avoid allocation I/Os, which can represent a significant part of the total number of I/Os for a transaction. You can also reduce the processing time needed to check whether users attempting to execute a plan are authorized to do so.

An EDM pool that is too small causes:

- Increased I/O activity in DSNDB01.SCT02, DSNDB01.SPT01, and DSNDB01.DBD01
- Increased response times, because of loading the SKCTs, SKPTs, and DBDs
- Fewer threads used concurrently, because of lack of storage

An EDM pool that is too large might use more central storage than necessary.

Use the DB2 EDM Pool Statistics, Overview report to monitor the usage of your Db2 EDM buffer pools. The report shows buffer usage and hit ratios for DBDs, cursor tables, and package tables:

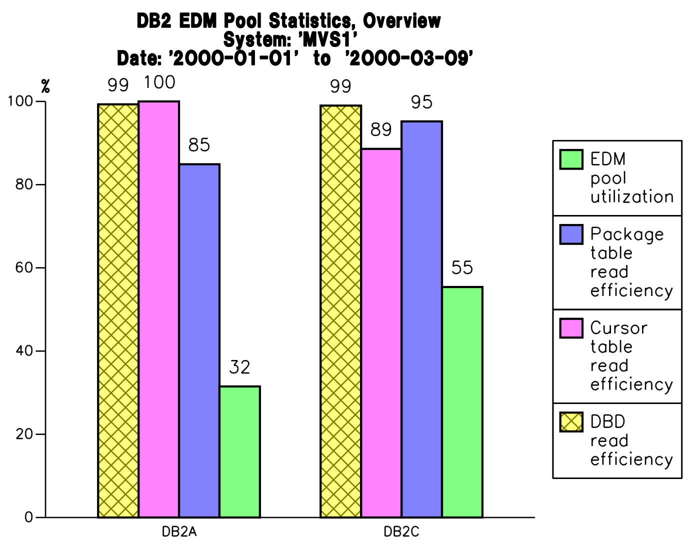

Figure 75. DB2 EDM Pool Statistics, Overview report

The report shows the read efficiency for DBDs, cursor tables, and package tables. The read efficiency shows how often these objects are already in the EDM pool, thus avoiding extra I/O operations. The report also indicates how the EDM pools are being used (that is, the percentage used for DBDs, the percentage used for cursor tables, and the percentage used for package tables).

When pages are needed for the EDM pool, any pages that are available are allocated first. If the available pages do not provide enough space to satisfy the request, pages are taken from an inactive SKCT, SKPT, or DBD. If there is still not enough space, an SQL error is sent to the application program.

Analyzing Db2 processor usage

Many factors affect the amount of processor resources that Db2 consumes. You can reduce Db2 consumption of the processor resources by:

- Reusing threads
- Using efficient caching of authorizations in all environments to eliminate Db2 authorization checks
- Reducing the amount of overhead processing by suppressing traces, using fixed-length columns, reducing locking, and using more efficient access paths

Use the DB2 System CPU % by Address Space, Overview report to monitor the relative processor usage by system address spaces for your Db2 subsystems.

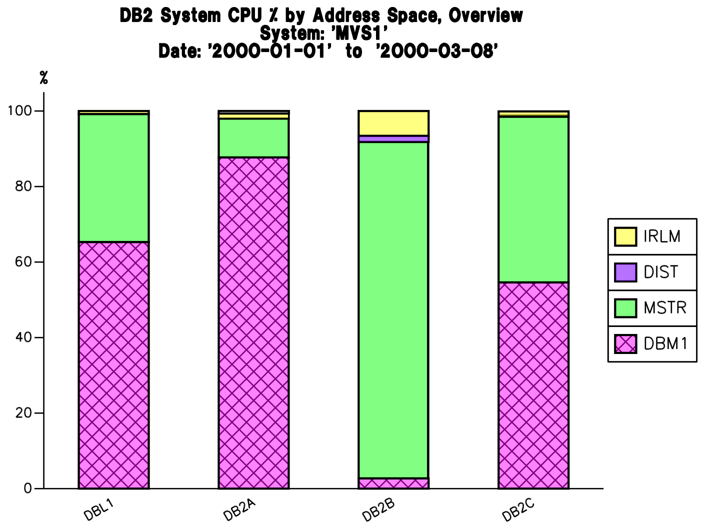

Figure 76. DB2 System CPU % by Address Space, Overview report

The report shows the relative CPU usage for these address spaces:

- Db2 database service (DBM1) address space
- Db2 system services (MSTR) address space
- Db2 distributed data facility (DIST) address space
- IMS Resource Lock Manager (IRLM) address space

Analyzing Db2 performance in a distributed environment

An SQL request sent to a remote system can sometimes take longer to execute than the same request, accessing tables of the same size, on the local Db2 subsystem. The principal reasons for this potential increase in execution time are:

- Overhead processing, including startup, negotiating session limits (change number of sessions processing)
- The time required to send messages across the network

Some aspects of overhead processing (for instance, network processing) are not under Db2 control.

Measuring Db2 response time in a distributed environment

Db2 accounting records are created separately at the requester and each server. Events are recorded in the accounting record at the location where they occur.

The following figure shows the relationship of the accounting class 1 and class 2 times and the requester and server accounting records.

This figure is a very simplified picture of the processes that go on in the serving system. It does not show block fetch statements and is only applicable to a single row retrieval.

The elapsed times referred to in the header are:

- (1)—SERV

The amount of elapsed time spent at the server between the actual receipt of the SQL statement until the answer is sent to VTAM. (Not applicable to DRDA connections.)

- (2)—REQ

  The amount of elapsed time spent at the requester between the send of the SQL statement and the receipt of the answer from the server.

- (3)—Server Cls 1

  The elapsed time from the creation of the database access thread until the termination of the database access thread.

- (4)—Server Cls 2

  The elapsed time to process the SQL statement and the commit at the server.

- (5)—Server Cls 3

  The amount of time the serving Db2 system spent suspended waiting for locks or I/O.

- (6)—Requester Cls 3

  The amount of time the serving Db2 system spent suspended waiting for locks or I/O.

- (7)—Requester Cls 2

  The elapsed time from when the application passed the SQL statement to the local Db2 system until return. This is considered in Db2 time.

- (8)—Requester Cls 1

  The elapsed time from the creation of the allied thread until the termination of the allied thread.

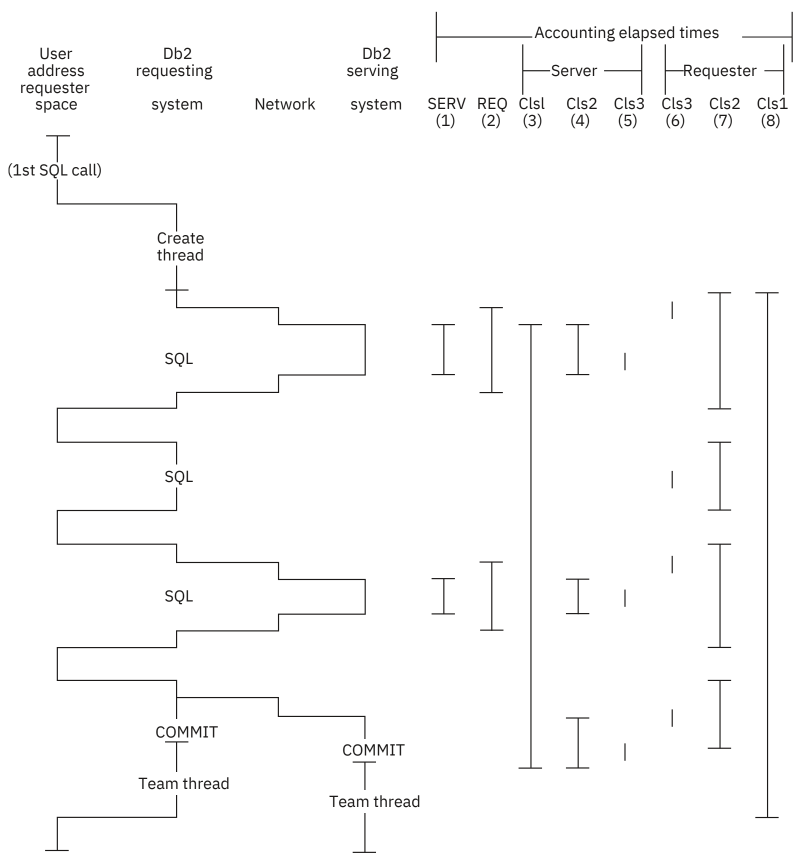

Figure 77. Elapsed times in a distributed environment as reported by Db2

Analyzing Db2 response time in a distributed environment

The above figure highlights that the class 2 in Db2 elapsed time at the requester includes the time that elapsed while the requester was waiting for a response from the server. To remove that time, subtract the requester elapsed time (see the REQ column in the figure) from the class 2 elapsed time.

However, the class 2 processing time (TCB and SRB) at the requester does not include processing time at the server. To determine the total class 2 processing time, add the class 2 time at the requester to the class 2 time at the server.

Likewise, add the GETPAGE counts, prefetch counts, locking counts, and I/O counts of the requester to the equivalent counts at the server. For private Db2 connections, SQL activity is counted at both the requester and server. For DRDA connections, SQL activity is counted only at the server.

The accounting distributed fields for each serving or requesting location are collected from the viewpoint of this thread communicating with the other location identified. For example, SQL sent from the requester is SQL received at the server. These distributed fields from the requester and the server should not be added together.

Several fields in the distributed section merit specific attention. The number of VTAM conversations is reported in several fields. The count of the number of conversation requests queued during allocation is reported as CONVERS_QUEUED in the IBM Z Decision Support Db2 component tables. There is a count of successful conversation allocations (CONVERS_ALLOCATED). You can use the difference between allocations and successful allocations to identify a session resource constraint problem. VTAM and network parameter definitions are important factors in the performance of Db2 distributed processing. For more information, refer to the *VTAM Network Implementation Guide*.

Bytes sent, bytes received, messages sent, and messages received are recorded at both the requester and the server. They provide information on the volume of data transmitted. However, because of the way distributed SQL is processed, more bytes may be reported as sent than are reported as received.

The number of SQL statements bound for remote access is the number of statements dynamically bound at the server for private Db2 connections access. This field is maintained at the requester and is reported as BOUNDS_FOR_REMOTE in the Db2 component tables.

To determine the percentage of the rows transmitted by block fetch, compare the total number of rows sent, to the number of rows sent in a block fetch buffer, which is reported as BLOCK_FETCH_ROWS in the Db2 component tables. The number of rows sent is reported at the server, and the number of rows received is reported at the requester. Block fetch can significantly affect the number of rows sent across the network.

Because of the way distributed SQL is processed, there may be a small difference in the number of rows reported as sent compared with those reported as received. However, a significantly lower number of rows received may indicate that the application did not fetch the entire answer set. This is especially true for private Db2 connections.

Chapter 6. Analyzing VM performance

This chapter describes the performance analysis you can perform using VM data collected by IBM Z Decision Support. This data is VM accounting data and VM performance data, which is generated by the VM monitor facility and the VM Performance Reporting Facility (VMPRF).

This chapter suggests a top-down approach to system management. After establishing service-level agreements with your user groups, you must ensure that the users are receiving the service they require. The key indicator of service level is response time. Performance analysis should start with the user response times. If you detect a problem with the response times, you then investigate the system resources to determine what is causing the delay.

This chapter describes the three major host resources: processor, processor storage, and I/O. Network resources also affect VM system performance.

For information on analyzing the delays caused by networks, refer to these books:

- *Network Performance Feature Installation and Administration*
- *Network Performance Feature Reference*
- *Network Performance Feature Installation and Administration*

The discussion in each category does not cover all the aspects of a computer system and environment that might influence that category's measurement and analysis. Exactly what affects a transaction's response time is always dependent on the circumstances at the time the event occurs or when the measurement sample is taken.

By analyzing reports and data for each category, you can identify whether a performance problem exists. If it does, you can identify more closely the source of the problem and what you can adjust to overcome it.

Using VM accounting and z/VM Performance data

IBM Z Decision Support collects two types of VM data. The VM accounting component uses accounting data generated by the VM operating system. The z/VM Performance component processes data from the VM Performance Reporting Facility and z/VM Performance Toolkit.

Using z/VM Performance data

z/VM Performance Toolkit and VMPRF are performance analysis tools that create performance reports from VM monitor data. They provide data on system performance and resource usage and also for data tables and reports for the z/VM Performance component. By analyzing z/VM Performance component reports, you can detect z/VM performance constraints.

The z/VM Performance component collects data from VMPRF or z/VM Performance Toolkit SUMMARY files. The SUMMARY files contain key performance measurements. z/VM Performance Toolkit and VMPRF create the SUMMARY files from data collected from the VM monitor facility.

Use VMPRF summary records to collect performance data on z/VM systems up to version 4.4; use z/VM Performance Toolkit extended summary records to collect performance data on z/VM systems version 5.1, or later.

These steps describe where the data originates and how it becomes available to IBM Z Decision Support:

1. The VM monitor, which is part of the VM operating system, creates monitor records in a discontiguous saved segment (DCSS) in central storage.
2. The utility program MONWRITE moves the data from the DCSS to tape or, more commonly, DASD.
3. z/VM Performance Toolkit or VMPRF reads the monitor data from tape or DASD and creates SUMMARY files.

4. The IBM Z Decision Support administrator transfers the SUMMARY files from VM to the z/OS system running IBM Z Decision Support.
5. The IBM Z Decision Support collect step reads the SUMMARY files and stores the data in the IBM Z Decision Support database.

Understanding the VM monitor facility

The VM monitor facility provides a mechanism for collecting system performance data. Based on the analysis to be done, you can control the amount and nature of the data to be collected. The VM monitor facility collects two fundamental types of data: event data and sample data. This data is collected using different collection techniques, which are explained in detail in following sections.

The data collected by the VM monitor is in these CP-defined data domains (classes):

System domain
: Contains system resource usage data. This domain contains only sample data. The general-system data domain is always automatically enabled when monitor sampling starts and remains active until monitor sampling stops.

Monitor domain
: Contains information about system configuration (paging, storage, I/O, processors, and other areas). It also provides information about the operation of the monitor itself. This domain contains sample and event data. The monitor domain is always automatically enabled when monitor sampling or event recording starts and remains active until the monitor stops.

User domain
: Contains data relating to users logging on and off, scheduling status, virtual I/O, and vector usage. This domain contains sample and event data.

Scheduler domain
: Contains information of queue manipulation relating to the scheduler and a user’s use of the system (with terminal reads and writes). It monitors the flow of work through the scheduler and indicates resource allocation strategies of the scheduler and dispatcher. This domain contains only event data. Note: Because of the large amount of data this domain can generate, do not collect scheduler domain data.

Seek domain
: Contains records that are generated each time a SEEK command is found in a channel command word (CCW). This domain contains only event data. Note: Because of the large amount of data this domain can generate, you should limit the number of devices for which you collect seek data.

I/O domain
: Contains counters to monitor I/O requests, error recovery, interrupts, and other information for real devices. This domain contains sample and event data. Note: Because VMPRF analyzes only DASD data, you should collect I/O domain data for CLASS DASD only.

Processor domain
: Contains data on system locks, simulation, interprocessor signalling, and other statistics related to processor usage. This domain contains sample and event data.

Storage domain
: Contains system statistics for utilization characteristics of real, virtual, expanded, and auxiliary storage. This domain contains sample and event data.

Application data domain
: Contains application data copied from a virtual machine’s storage when this storage has been declared to CP for collecting the data generated by the application program in that virtual machine. This domain contains sample and event data.

Event-driven data collection

Event-driven data collection produces one monitor event record for each selected system event at the time the event is detected. The volume of data produced is determined by the frequency of the selected system events.

Several event-driven domains exist, each providing data related to a particular area of system activity. With the exception of the monitor domain, each domain can be started and stopped separately. Additional controls further restrict events to those of specific interest. For example, in the I/O domain, you may specify device numbers, types, and classes to be included or excluded from monitoring. In the user data domain, you can include or exclude specific users from event-driven records created.

Sample data collection

Sample data collection produces monitor records at the end of each interval. The records contain two types of data: single-sample data and high-frequency data. Most domains have their own high-frequency sample data or single-sample data that is sampled as requested. For the I/O and user domains, only specified devices or specified user IDs are sampled.

Single-sample data

This data is collected once during the time interval, and only at the end of that time interval. Some of this data represents the status of the system at the time the data was collected; other data is accumulations of counters, states, or elapsed times collected at the end of each time interval since sampling started.

A user ID must be present in two consecutive monitor samples before VMPRF reports any resource use by that user. The average logon session must be at least two monitor samples intervals before z/VM Performance's user resource usage reports are reasonably accurate. The combination of short logon sessions and long monitor intervals results in large amounts of processor time, paging, and I/O that are never accounted for in the summary files.

Some of the I/O counters are halfwords and roll over after 65 536 events. You must sample these counters at least once every 5 minutes to avoid loss of data (assuming an I/O rate of 100 I/Os per second). The more frequently you sample the counters, the more accurate the resulting analysis. The default sample interval of 1 minute is sufficient for most organizations. Very large organizations may find that a 5-minute sample interval provides a usable, but more manageable amount of data. However, long sample intervals are usually accompanied by a loss of precision, especially in the user domain.

High-frequency data

This data is collected at a higher rate than it is recorded. For each domain or group of pertinent data within the domain, a separate counter exists that represents the number of times particular data was sampled.

Each high-frequency counter contains a cumulative count of frequency values that represent system states that you can monitor. Each time a certain domain is sampled, the monitor adds the data value to a counter. The counter is always incremented unless high-frequency sampling is stopped and restarted, which sets the sampling counters to zero. The monitor places high-frequency data in monitor records only when a single-sample interval expires.

You need at least 30 high-speed samples per sample interval to have statistically significant results. The default 60-second sample interval and 2-second high-frequency sampling rate provide the required number of samples. If you change either the sample interval or the high-frequency sampling rate, you must maintain or increase the number of high-frequency samples per sample interval to maintain data significance.

Using VM accounting data

VM generates this accounting data for each virtual machine:

- Virtual machine processor usage
- Dedicated device usage

- Temporary disk usage
- Logons and autologs
- Successful and unsuccessful attempts to link to minidisks
- VTAM service machine link activity
- Terminal identification data

VM accounting data is a useful cross-check against user data reported by z/VM Performance. If the VM monitor sample interval is long and the average logon session is short, the user resource usage reported in the VM accounting data is more accurate than the user resource usage reported by z/VM Performance.

Refer to *System Performance Feature Reference Volume II* for information on how the VM accounting component processes the data collected. It processes the data collected to produce graphical and tabular reports.

After collecting records (possibly on the user ID DISKACNT), transmit or load the data to the z/OS system to be collected by IBM Z Decision Support.

To find more information about how z/VM records accounting information and how you can control its activities, refer to *z/VM CP Planning and Administration*.

Analyzing response time

The key indicator of performance is user response time. When users start complaining about the response time that they see, you know that a problem in your system or the network is causing delays.

Resource contention in the host affects a user's response time. These resources include the processor, processor storage, I/O subsystem, communication with service machines, and network and line delays. Network and line delays may be a significant part of total end-user response time, but there is no way to measure them directly within VM.

z/VM provides system-wide transaction rates and response times, as collected by the VM scheduler. z/VM also provides response times by user class and individual users. You can also compute the approximate response plus think-time command-time cycle using this formula:

```
command cycle = active users / total transaction rate
```

where the total transaction rate is the sum of the trivial, nontrivial, and QuickDsp transaction rates.

However, none of the readily available measures of user response time, including internal transaction response time, reflect response time as seen at the user's terminal or are consistent. For example, the classification of transactions as trivial or nontrivial constantly changes. The classification varies by workload. A transaction that is trivial in one workload may be nontrivial in another workload. The response times recorded by VMPRF also do not reflect dependencies on service machines.

The best indicators and most reliable measures of end-user response time are the length of the dispatch and eligible lists, and the number of users in SVM wait on the VTAM/VSCS servers.

A virtual machine usually experiences delays when it is in the dispatch list with a transaction in progress and waiting for resources. The only resources that can delay a user in the eligible list are storage and paging. Most systems perform best when tuned to run with no users in the eligible list. Users can also wait for resources while in the dormant list. The only explicitly measured delay is SVM wait (waiting for a service machine), but users can be in the dormant list, have transactions in progress, and be waiting on I/O or DASD paging.

The performance characteristics of an operating system depend on factors such as the choice of hardware, the total number of users on the system, the functions being performed by the system, and the way the system parameters are set up.

The interrelationship of these factors primarily affects system performance:

- Overloaded user DASD

- Underutilization of minidisk cache or controller cache
- The speed and number of paging devices
- The amount of auxiliary storage made available
- Real storage size
- Real processor speed
- Characteristics of the workload being processed in each virtual machine
- The number of virtual machines logged on concurrently

Possible causes of delay are:

- Waiting to start or excessive input queue time
- Waiting for the processor while in storage
- Excessive demand paging
- Waiting for I/O to complete (including delays due to operational constraints)
- Queue contention for CMS or batch work

There is no order in which you must investigate these delays. The best way to identify a starting point is to look at the number of users waiting for various system resources. Investigate the largest source of delay first. Consider using this set of priorities:

1. I/O
2. Processor storage (central and expanded)
3. Processor
4. Communications with service machines

Perform the analysis for the system as a whole, for individual service machines in the critical path for response time, and for whatever user classes are important to you.

A well-tuned system should have few storage, paging, or I/O constraints. You can experience situations where the processor is only 40% utilized, and is 60% in wait state. You might then conclude that the system does not have enough load for the processor to work at 100% utilization. But this conclusion might be far from reality, because users might not be able to execute on the processor due to constraints in the system. Most users on the dispatch list may be waiting for I/O or paging, and some users may be on the eligible list (waiting for storage).

Thus, you should analyze the reason for wait states and verify how much the system is in idle state, without any users waiting in the dispatch list or eligible list.

Analyzing I/O resources

Most problems reported as poor VM performance are actually a result of I/O. Channel loading, control unit or device contention, file placement, and shared DASD have potential as performance constraints.

This section shows you the most important indicators of bottlenecks on DASD.

Measuring I/O response time

The key to finding and fixing I/O-related performance problems is DASD response time: the length of time it takes to complete an I/O operation. Response time can have a dramatic effect on performance, particularly with online and interactive subsystems, such as CMS.

The following figure illustrates how DASD response time is defined.

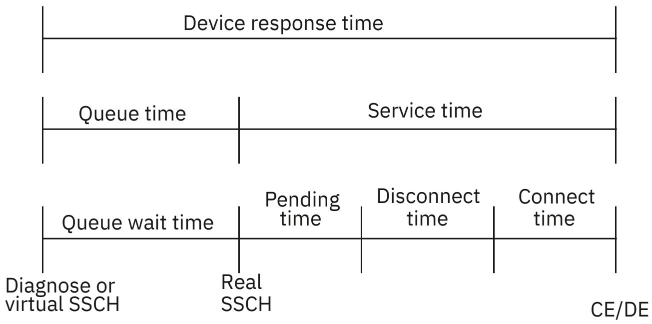

Figure 78. DASD response-time components

DASD response time is the elapsed time from the DIAGNOSE instruction at the start subchannel (SSCH) instruction to the completion of the data transfer, which is indicated by a channel end/device end (CE/DE) interrupt. It includes any queue time plus the actual I/O operation. *Service time* is the elapsed time from the successful SSCH instruction to the data transfer completion. It includes seek time, any rotational delays, and data transfer time. Service time plus queue time equals *response time*.

The above figure shows these DASD response-time components:

Queue wait time

This is the internal VM queueing of the I/O operations that are waiting for a previous I/O to the device to complete. Queue time represents the time spent waiting on a device. Delays in the other service components may cause the queue component to increase, or the queue time may be a function of skewed arrival of I/Os from a particular application.

Pending time

This is the time from the start of the I/O until the DASD receives it. The pending time indicates the channel path usage. A high pending time indicates that the channel or logical control unit is busy. Pending time can be caused by busy channels and controllers or device busy from another system.

If a device is behind a cache controller (non-3990), pending time can also be caused by cache staging (the device is busy during the staging operation). When using nonenhanced dual copy (3990), the device is busy while writing the data to the primary volume and to the duplex volume if fast copy was not selected.

Disconnect time

Disconnect time includes:

- The time for a seek operation.
- Latency, always assumed to be half a revolution of the device.
- Rotational position sensing (RPS) reconnect delay, the time for the set sector operation to reconnect to the channel path.

  This time depends on internal path busy, control unit busy, and channel path busy. If any element in the path is busy, a delay of one revolution of the device is experienced.
- For a 3990 Model 3 cache controller, if the record is not in cache, the time waiting while staging completes for the previous I/O to the device or until one of the four lower interfaces becomes available from either transferring, staging, or destaging data for other devices.

When a device cannot reconnect to the host to transfer data because all paths are busy, it must wait for another revolution.

Using cache control units reduces or eliminates disconnect time. Disconnect time is used as a measurement of cache effectiveness.

Connect time

This is the time actually spent transferring data between the channel and DASD or channel and cache. Connect time can also include the time a search operation is occurring between the channel and the

DASD or the channel and the cache (usually done to search a directory to find the location of a program module so it can be loaded into storage).

A high connect time indicates that you are using large blocks to transfer the data. This can be a problem if you mix small blocks and large blocks. The small blocks may have to wait on the larger ones to complete, thus causing a delay.

Analyzing I/O devices

Often inadequate system or virtual machine performance is due to problems in the DASD subsystem. The DASD subsystem has a big impact on overall system performance.

High device usage is not necessarily a problem. It can be a problem when it causes excessive queue lengths, particularly for response-oriented subsystems. Therefore, the measure of good or bad performance is not usage but response time. High device usage can be one of the causes for excessive queue time. The higher the device usage, the more likely the request will be queued. High device usage does not necessarily imply high service time. You can often find a DASD volume with high usage and reasonable device service time. High device usage by itself is not necessarily undesirable, the significant measurement is the average SSCH response time for the device.

Several factors can affect response time. High usage of channel paths, control units, or other devices can reduce the chances for a nonqueued I/O operation. High usage of channel paths, control unit contention, and poor device seek patterns can further increase response time, after the I/O operation begins.

To find I/O bottlenecks, look for devices with a lot of queueing. If a device can handle a high I/O rate with little or no queueing, it is not a bottleneck. Devices cached in expanded storage or in the controller can handle a high I/O rate without queueing.

Evidence of I/O bottlenecks includes:

- Paging queued on the paging devices
- I/O requests queued on nonpaging devices (shown by device response times that exceed service response times)
- Users in paging wait (DASD paging or system I/O), instruction simulation wait (DIAGNOSE I/O), or I/O active (nonsynchronous I/O started using SIO, start I/O fast release (SIOF), or SSCH)

High device busy, a low I/O rate, and high disconnect time usually indicate excessive channel or control unit use. High device busy, a low I/O rate, and high connect time indicate long data transfers and are probably not a problem.

The approach in looking at DASD is not to see which volumes are the busiest, but rather to look at which volumes are causing the most delays to virtual machines. The z/VM DASD with Longest Queues report shows the devices with the longest queues.

```
                    z/VM DASD With Longest Queues
                  System: 'XYZVM   ' Period: ('PRIME')
                           Date: '2000-12-28'
                    Time: '08.00.00' to '20.00.00'

                                SSCH                Response  Service
               SSCH+RSCH       queued    SSCHrate    time      time
VOLSER          (count)       (count)   /second     (msec)    (msec)
------   ---------------   -----------  --------   --------  -------
VMX240            101863           132       4.7       48.2     47.0
MNK7A9             15593            83       0.6       22.3     18.3
EXPSUP             62379            58       2.3       20.7     19.6
MDSK21            652830            34      20.8       14.8     14.7
VMXA2R             55907            31       1.8       15.8     14.7
EQL002             80128            25       2.5       17.8     17.2
MNK7AA             19435            25       0.7       15.0     14.0
VMX25B             22582            17       1.6       42.4     41.6
EQL001            102132            11       3.3       17.2     17.0
MDSK15             19869             7       0.7       22.2     21.7

       IBM Z Decision Support Report VMPRF_D3
```

Figure 79. z/VM DASD With Longest Queues report

This report shows the response time and service time for each DASD volume. The difference between the response time and service time indicates how much queueing there is in the system on the device.

The report contains this information:

VOLSER
The DASD volume serial number.

SSCH+RSCH (count)
The device usage. This value shows how many requests were sent to this device.

SSCH queued (count)
Number of requests queued because the device was busy by another application.

SSCHrate/second
The number of start and resume subchannel requests for the device per second.

Response time (msec)
I/O response time for the request, in milliseconds.

Service time (msec)
The time needed to service the request, in milliseconds. When looking at the service time, determine whether the value is reasonable for that particular device and configuration. Service time is the sum of pending time, disconnect time, and connect time. You must determine a threshold for service time. If a device exceeds that threshold, look at more detailed reports.

Your busiest devices should take advantage of cache controllers. The response time will be high if the device is not cached or if cache is ineffective.

Service time for a device should be less than one revolution. If it exceeds this, you probably have a problem. The service time should be close to the connect time.

Note: DASD devices attached to integrated adapters on the ES/9221 processors will always show service time, response time, and usage equal to zero. These adapters do not provide the channel measurement information used by z/VM to obtain this information.

Preferred guests (V=R and V=F) with dedicated I/O devices should use SIE I/O assist and should have a low virtual I/O rate and a high capture ratio. DASD used with I/O assist will not have a reported service time. To report a service time, VM would have to take the device out of I/O assist to issue an I/O request to report a value. DASD performance benefits from going into I/O assist and not coming out of it. If you see transitions in and out of I/O assist, it could indicate a hardware error and may cause performance problems.

For applications that use *BLOCKIO (for example, the shared file system or SQL), contention appears as queueing on a device, instead of instruction simulation wait on the processor. *BLOCKIO is asynchronous.

If a particular device is experiencing high SSCH rates and poor response time, you might want to move some data to a less-used device to distribute the workload. The z/VM Least Used or not Used DASD Devices report lists the DASD devices with no or low utilization.

```
                  z/VM Least Used or not Used DASD Devices
                    System: 'XYZVM    ' Period: ('PRIME')
                             Date: '2000-12-28'
                       Time: '08.00.00' to '09.00.00'

                                  SSCH                Response  Service
                 SSCH+RSCH       queued    SSCHrate     time      time
VOLSER            (count)        (count)   /second     (msec)    (msec)
------   ---------------   -----------  --------  --------  -------
VMX4AA                28             0       0.0       0.8      0.8
VMX4AB                28             0       0.0       0.8      0.8
VMX4A1                28             0       0.0       0.8      0.8
VMX4A3                28             0       0.0       0.8      0.8
VMX4A9                28             0       0.0       0.8      0.8
MDSK42                30             0       0.0      20.0     20.0
MDSK46                34             0       0.0      16.8     16.8
MDSK28                35             0       0.0      19.1     19.1
E56LEE                38             0       0.0      23.0     23.0
VMPG03                41             0       0.0      19.2     19.2

            IBM Z Decision Support Report: VMPRF_D4
```

Figure 80. z/VM Least Used or not Used DASD Devices report

When investigating DASD performance, consider these things:

- Is the device a CMS minidisk? Is it blocked at 4K? Is it cached in expanded storage? Is it shared with another physical system image? A CMS minidisk that is not shared with another system image, and that is blocked at 4K should be cached in expanded storage. Reads from the minidisk cache can replace most of the real I/O to the device.
- Is minidisk cache effective? The average lifetime of a block in the minidisk cache should be equal to think time plus two standard deviations plus response time.
- Are devices used by preferred guests dedicated to them? I/O by a preferred guest to a dedicated device is handled by SIE and is never seen by VM. The service and response time of that device must be measured by the guest.
- Are devices that should be handled by SIE I/O assist really being handled by SIE I/O assist?
- Are cached controllers being used effectively?
- Is the shared file system in use? Do the devices with high activity or queueing belong to an SFS file pool?

Analyzing processor storage resources

To analyze your processor storage, you must understand processor storage usage.

The z/VM Processor Storage Activity report provides information on the paging activity of all online processors.

```
                 z/VM Processor Storage Activity, Hourly
                   System: 'XYZVM   ' Period:  ('PRIME')
                             Date: '2000-01-04'

                           Expanded  Expanded  Expanded    DASD      DASD
      Processor  Fastpath   paging   page-ins  page-out  page-ins  page-out
Hour   address     (%)       /sec      /sec      /sec      /sec      /sec
----  ---------  --------  --------  --------  --------  --------  --------
  13          0     94.95      3.63      2.57      1.06      0.83      0.00
              1     93.33      9.32      2.58      6.74      1.97      3.42
              2     92.38      3.59      2.28      1.31      0.68      0.00
              3     94.75      3.92      1.53      2.39      0.66      0.00
              4     95.25      2.17      1.19      0.98      0.47      0.00
              5     95.63      1.97      0.82      1.16      0.46      0.00

  14          0     96.60      5.25      2.55      2.71      1.50      0.00
              1     93.87     15.49      3.46     12.03      2.86     12.76
              2     93.86      3.76      1.87      1.89      0.90      0.00
              3     94.56      2.65      1.65      1.00      0.89      0.00
              4     96.50      1.91      1.12      0.79      0.60      0.00
              5     88.94      1.65      1.13      0.53      0.79      0.00

  15          0     94.86      3.90      2.14      1.76      0.90      0.00
              1     93.10     12.38      2.84      9.55      1.99      4.75
              2     94.33      3.73      2.43      1.30      0.72      0.00
              3     94.95      3.04      1.50      1.54      0.69      0.01
              4     96.52      1.99      1.46      0.54      0.61      0.00
              5     94.14      3.35      1.01      2.34      0.56      0.00


              IBM Z Decision Support Report: VMPRF_P4
```

Figure 81. z/VM Processor Storage Activity, Hourly report

The Fastpath % column shows the percentage of page-ins, moving the contents of an expanded storage block to central storage, that were handled by the page fault fastpath. The Expanded paging/sec, Expanded page-ins/sec, and Expanded page-out/sec columns show movement between central and expanded storage. The DASD page-ins/sec and DASD page-out/sec columns indicate page movement from central to auxiliary storage. These page reads and writes require I/O, which can affect user response time.

Analyzing central storage

The basic objective in tuning central storage is to keep in storage the pages used by a virtual machine while the user has a transaction in progress. You also want to keep the pages in storage while the user thinks about the next transaction, so that they are still resident when the user enters the next command. You want to keep pages resident long enough to avoid paging.

You can improve central storage usage by using saved segments. Saved segments let users share storage. For example, a 1MB segment of code used by all users on a system with 1000 users requires 1MB of storage, if the code is in a saved segment and shared by the users. The code would require 1000MB of storage if each user had a separate copy.

You can also improve central storage use by using saved segments to share a common read-only minidisk’s file directory table.

Analyzing expanded storage

How VM uses expanded storage depends on system activity. If the system spends more time waiting on cache-eligible minidisk I/O than it spends on paging, the system uses expanded storage primarily for minidisk cache. Consider configuring VM so that all expanded storage is used for minidisk cache, and none of it is used for paging.

Note: If you want to dedicate all of expanded storage to minidisk cache, specify ALL on the CP RETAIN command. With ALL, CP sets a flag so it never pages to expanded storage.

Analyzing paging activity

The z/VM Processor Page and Spool Activity, Hourly report provides information about paging activity.

```
                 z/VM Processor Page and Spool Activity, Hourly
                     System: 'XYZVM    ' Period:  ('PRIME')
                                Date: '2000-01-04'

                        Page        Page        Spool       Spool
       Processor        read        write       read        write
Hour   address        (count)     (count)     (count)     (count)
----   ---------    ----------  ----------  ----------  ----------
  13           0         2975           1           3           0
               1         7099       12325       39765       24668
               2         2440           0           5           0
               3         2382           3           3           0
               4         1694           3           1           0
               5         1642           0           2           0

  14           0         5406           0           1           0
               1        10307       45918       40760       26325
               2         3238           2           1           0
               3         3189           1           1           0
               4         2145           0           0           0
               5         2861           2           0           0

  15           0         3239           0           0           0
               1         7167       17095       45342       29778
               2         2581           1           0           0
               3         2468          25           4           0
               4         2194           2           0           0
               5         2011           2           1           0


       IBM Z Decision Support Report: VMPRF_P3
```

Figure 82. z/VM Processor Page and Spool Activity, Hourly report

The z/VM User Paging and Spooling, Monthly report provides more details about paging activity.

```
                     z/VM User Paging and Spooling, Monthly
                       System: 'XYZVM    ' Period:  ('PRIME')
                             Month: '2000-01-01'

                              DASD          DASD
User       User               page          page           Spool         Spool
class       ID                reads         writes         reads         writes
--------   --------   ------------  ------------  ------------  ------------
ADMIN      AABABLD              12             0             0             0
           AASADMIN              0             0             0           167
           ADCOCK                0             0             0             7
           AHLGREN               0             0          1229           499
           ALLMOND               1             0           237            32
           AMMERMAN              0             0          1686            32
           ANDERSON              0             0          5341           530
           AUTOLINK            176            23             1             6
           AVAIL                 0             0            42            18
           BASSETT              20             0            86            36

       IBM Z Decision Support Report: VMPRF_U2
```

Figure 83. z/VM User Paging and Spooling, Monthly report

Most service machines should have paging rates closely approaching zero. Pages should be reserved for VTAM and other service machines until the paging rates are close to zero.

For most users, the amount of time they spend in page wait is a better indicator of a paging bottleneck than the actual paging rate.

Time spent in page wait includes waiting on system I/O (for example, reading the CP directory, updating the warm start area, and doing spool I/O). You might have a paging bottleneck on a SYSRES volume that has no paging and spooling areas.

Page read activity to spool volumes without page write activity probably comes from saved segments (DCSSs and NSSs). A significant amount of activity indicates that the usage count for saved segments is dropping to zero, the pages are being purged from the paging areas, and subsequent references must be satisfied by page reads from the spool areas.

Under normal conditions, most activity on the spool area should be spool reads and writes.

Analyzing processor resources

The best way to identify a processor bottleneck is to look at the number of users waiting for the processor.

To identify and reduce nonproductive processor time, divide processor time into categories, measure how much time is used in each category, and identify those categories that might be reduced or eliminated to improve performance without adversely affecting system functions. To do this, you must first understand how VM measures and reports processor time.

Measuring processor time

The following figure illustrates a sample division of processor time with the overlap of different categories.

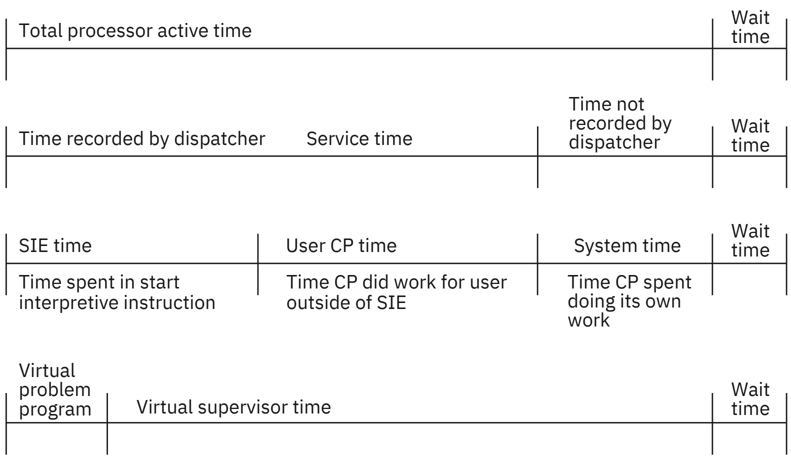

Figure 84. Processor time

Processor time includes these categories:

Dispatch time
The processor time required to complete the task, from first receiving the task to sending the result to the user.

Wait time
The time a task was in the wait state.

SIE time
The time spent in start interpretive instruction.

User CP time
The system time spent on behalf of the user.

System time
The time spent by the system doing system management overhead tasks. This time is not charged to any specific user.

Virtual problem program
The time that the processor service took to perform the user task.

Virtual supervisor time
The time used for supervisor services.

LPAR time
LPAR partition management time. This time is visible only as the difference between the total processor time reported by LPAR and the processor time measured within the partition.

Many processor complexes consist of several individual processors connected to each other. The operating system managing the work on these processors usually has access to all the processors for both work and measurement purposes. The VM monitor knows how many processors that are physically or logically (in the case of PR/SM) made available to the operating system and stores the wait times for each processor in the same record at the end of each measurement interval.

If a processor complex includes four individual processors, all physically available to the VM system, then four measures of processor wait time are in each VM monitor record at the end of each interval.

z/VM calculates total processor busy as (total system and user time for all processors) / (total online time for all processors). This calculation does not include LPAR partition management overhead.

Analyzing processor activity

You might conclude that 100% processor usage means trouble, but this is not always true. A system can run at 100% utilization without causing any problem in system throughput or in transaction response time. If you have 100% processor usage, try to determine whether the throughput is starting to decrease (VIO rate decreasing or emulation time decreasing) or whether the response time is deteriorating (average response time of transactions increasing for the same number of transactions).

To detect a processor bottleneck, look at the total number of users in the dispatch list, the number waiting for the processor, and the number waiting for other resources. If the number of users waiting for the processor is consistently the largest number of the users in the dispatch and eligible lists, then you have a processor bottleneck. Examine the user state samples for individual users. If processor wait is the most frequent wait state, that user may be constrained by the availability of processor time.

To look at total processor utilization, look at either processor wait time or busy time. The z/VM Processor Usage Distribution, Hourly report shows the processor wait time (System wait) for all processors in your processor complex.

```
                  z/VM Processor Usage Distribution, Hourly
                     System: 'XYZVM    ' Period: ('PRIME')
                             Date: '2000-01-04'

                      Busy   Processor   Processor   Processor      System
      Processor       time      user     emulation     system        wait
Hour   address        (%)      (sec)       (sec)       (sec)        (sec)
----  ---------    -------  ----------  ----------  ----------  ----------
   9          0      18.75         610         432          66        2823
              1      29.55         765         370         298        2151
              2      16.65         542         382          58        2910
              3      13.86         453         322          46        3030
              4      10.78         355         253          33        3160
              5       8.82         294         220          23        3242

  10          0      18.99         625         447          59        2803
              1      34.26         939         464         295        1869
              2      17.59         581         419          52        2867
              3      14.58         484         351          41        2996
              4      11.44         382         278          30        3128
              5       9.26         311         232          22        3220

  11          0      17.42         568         395          59        2865
              1      34.82         955         463         298        1859
              2      15.48         505         349          52        2948
              3      12.65         416         289          40        3070
              4      10.81         361         267          28        3146
              5       7.87         263         193          20        3272


              IBM Z Decision Support Report: VMPRF_P1
```

Figure 85. z/VM Processor Usage Distribution, Hourly report

You should note that processor utilization of 100% tells you nothing about bottlenecks. Good response time can cause heavy processor consumption in z/VM. Minidisk cache in expanded storage eliminates waiting for minidisk I/O, which lets users perform more transactions in a given period. More transactions increase processor usage during that time. Good response time also tends to reduce the demand for central storage, which in turn reduces the amount of processor time the system spends managing central storage and paging. Good response time also reduces the number of users with transactions in progress, and reduces the amount of processor time the system spends scheduling and dispatching users.

Heavy processor consumption could also indicate that you are using all the cycles you have. It is an indication of a processor problem only if you do not have desirable performance.

If system wait time is not high, focus the analysis on how processor time is used, with the objective of reducing or eliminating nonproductive processor time. You should then investigate:

- Central storage and other aspects of the paging subsystem, if significant processor time is being spent on block paging
- I/O (with the intent of reducing exceptions), if significant processor time is being spent on I/O interrupts that do not page
- Reduction of supervisor services such as SVCs

If wait time is high, determine why the system is waiting. Processor wait time has three possible causes:

- I/O delays
- Queues on serially reusable resources and internal locks
- Insufficient work in the system

Significant sources of queue contention often appear for file access. This contention contributes to I/O delays.

Although the system as a whole appears to behave normally, you should examine individual processors in the system. Note that when a VM system consists of more than one processor, one processor is the master processor. This processor is typically much busier than the other processors in the system.

The z/VM Processor Usage Distribution, Hourly report shows that each processor is busy within a normal range of activity. Busy time does not indicate any problems in performance. If the percentage were higher than 90% for an extended period, you should investigate storage and paging indicators to see if insufficient central or expanded storage is causing paging problems. You should also investigate the I/O subsystem to determine if significant processor time is spent on interrupts that do not page or swap.

You investigate processor problems by focusing on individual elements of performance. You should verify that data in these fields is within acceptable limits:

Busy time (%)
: Total user time and system overhead time divided by online time.

Processor user (sec)
: User processor time, in seconds. It includes time that CP spends on behalf of users doing such things as instruction simulation, diagnose I/O, spool I/O, or page translation.

Processor emulation (sec)
: Total virtual, emulation mode time, in seconds. Also expressed as time in start interpretive execution (SIE) mode.

Processor system (sec)
: System processor time, in seconds. This result represents CP overhead.

System wait (sec)
: Total processor wait time, in seconds.

Besides processor usage, you must look at the capture ratio. The z/VM User Real and Virt Processor Usage, Monthly report lists each user ID and the capture ratio for that user.

```
                z/VM User Real and Virt Processor Usage, Monthly
                       System: 'XYZVM   ' Period:  ('PRIME')
                                Month: '2000-01-01'

                           Busy     Virtual     Capture     Active       Logon
User      User             time      time        ratio       time         time
class      ID              (sec)     (sec)        (%)       (hours)      (hours)
--------  --------  ---------  ---------  ---------  ----------  ----------
ADMIN     AABABLD         202        197      97.50        0.73        0.73
          AASADMIN          5          3      64.10        0.25        0.25
          ADCOCK           37         34      91.70        0.07        0.25
          AHLGREN          15         12      78.86        0.45        0.50
          ALLMOND          29         26      90.28        0.12        0.25
          AMMERMAN         33         27      80.75        0.47        0.74
          ANDERSON         26         21      79.64        0.37        0.50
          AUTOLINK        555        544      97.98        6.98        7.98
          AVAIL             9          8      88.61        0.03        0.20

          IBM Z Decision Support Report: VMPRF_U1
```

Figure 86. z/VM User Real and Virt Processor Usage, Monthly report

A low capture ratio may be caused by using lots of system services, such as IUCV and VMCF. A high paging rate under z/VM is usually the result of using the fast path to page in from expanded storage, which charges the paging processor time as virtual (emulation) time and increases the capture ratio.

The z/VM System Instruction Rate, Hourly report provides information about the load on the system.

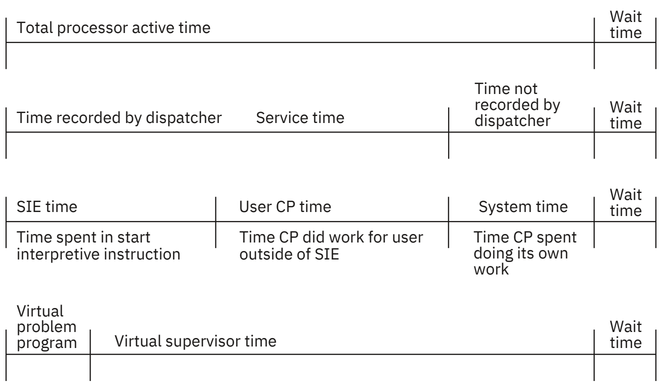

Figure 87. z/VM System Instruction Rate, Hourly report

The report shows these system instruction rates:

Diagnose rate/sec
The number of IBM-supplied DIAGNOSE instructions executed, per second

Simulation rate/sec
The number of simulated instructions executed, per second

SIGP interrupt rate/sec
The number of external signal processor (SIGP) interrupts issued by this processor, signalling another processor, per second

External interrupt rate/sec
The number of external processor interrupts, per second

Analyzing who is using the processor

Besides looking at the overall processor usage, you must also look at who is using the processor the most. The z/VM Heaviest Users of the Processor, Monthly report provides information on resource usage by individual users.

```
                 z/VM Heaviest Users of the Processor, Monthly
                     System: 'XYZVM    ' Period:  ('PRIME')
                              Month: '2000-01-01'

          Processor      Busy     Virtual      DASD        DASD
User        usage        time       time        IO        paging
 ID          (%)        (sec)      (sec)     (count)       /sec
--------  ---------  ---------  ---------  ---------  ----------
GARNER        38.29        689        667      37067        0.02
CPDOCS        33.21        359        351      13151        0.00
PATTONW       12.16         95         73      21274        0.00
LAING         11.58         28         27        597        0.00
CONSORTI      11.52        623        588      29524        0.05
LAPIDUS       10.31         17          9       9086        0.00
VTAM           8.06       2888        641          4        0.01
AABABLD        7.64        202        197       7087        0.00
HARRISL        6.86         58         56       1008        0.00
NETVIEW        4.56       1635        646    1638426        0.01
DIRMAINT       4.39        472        314     254950        0.00
MVSNM1         4.33       3101       1774      89655        0.12
TEMP404        4.30         75         44       6583        0.03
TRKING         4.23        305        270      32959        0.01
NEADE          4.22         76         74       3046        0.00
STAMMER        4.20        605        593       3832        0.00
ADCOCK         4.10         37         34       1581        0.00
NENAA          3.95         36         30       3354        0.02
MVSNM2         3.95       2827       1536      28349        0.08
SMSSYS         3.88       1390       1221      65521        0.64

       IBM Z Decision Support Report: VMPRF_U4
```

Figure 88. z/VM Heaviest Users of the Processor, Monthly report

Look at these fields:

Processor usage (%)
Processor usage for this user (or user class) expressed as a percentage of the sum of elapsed time for all online processors.

Busy time (sec)
Total processor usage in seconds for this user (or user class).

Virtual time (sec)
Virtual processor usage in seconds for this user (or user class).

DASD IO (count)
Number of page-ins and page-outs to DASD on behalf of this user. This value is the DASD paging rate.

DASD paging/sec
The rate of pages per second on behalf of this user.

For the best performance, ensure that the minor time slice does not expire while a user is between I/O operations. Make sure the minor time slice is equal to, or up to two or three times as long as, the processor busy time divided by the DASD I/O count. This value is the processor milliseconds used per I/O request. Setting the minor time slice larger than this value gives the user the processor time needed to go from the end of one I/O to the start of the next I/O without having storage taken. Setting the time slice is especially useful for guests.

Analyzing communications with service machines

This section describes the possible delays caused by communications with service machines.

Service machines are typically *black box* virtual machines. Usually, no user has direct access to them and they exist only to perform set tasks. Because the service machines perform set tasks, by definition they are transaction driven. They must provide a certain level of service to the user. Otherwise, user response times might be adversely affected.

Because users interact with service machines in day-to-day activities, such as OFFICE calendar and document handling, an important contributor to user response time is often the VTAM service machine in a predominantly network environment. Performance problems can be related to some constraint in a service machine.

VM has a special monitoring option for virtual machines that are service machines. The CP directory parameter SVMSTAT of the OPTION statement lets you mark a virtual machine as a service machine. The z/VM IUCV and VMCF Usage by User Class, Monthly report shows which user classes interact with other user IDs.

```
                  z/VM IUCV and VMCF Usage by User Class, Monthly
                       System: 'XYZVM    ' Period:  ('PRIME')
                                Month: '2000-01-01'

                 IUCV         IUCV         IUCV         VMCF         VMCF         VMCF
User            receive       fail         send        receive       fail         send
class           (count)      (count)      (count)      (count)      (count)      (count)
--------    ----------   ----------   ----------   ----------   ----------   ----------
ADMIN           671005           35       711225         2049            0         1003

         IBM Z Decision Support Report: VMPRF_U9
```

Figure 89. z/VM IUCV and VMCF Usage by User Class, Monthly report

To investigate user performance problems, look at the number of users in SVM wait. On an individual user basis, look at the amount of time a user spends in SVM wait. z/VM Performance Toolkit X'FC41' and VMPRF record type 41 contain information on the number of times a user is found in SVM wait and the number of times the SVM wait flag was left on at the end of a transaction and had to be reset by the scheduler.

Ideally, the servers should be idle (waiting for work) and the clients should spend none of their time in SVM wait. If a user is spending time in SVM wait, find out which service machine the user is using and investigate that service machine to determine what is causing the delay.

To determine if a service machine is causing a delay, find out if the service machine has messages queued. z/VM Performance Toolkit X'FC41' and VMPRF record type 41 contain data on the IUCV receive queue and the VMCF queue for a service machine. To use this data, you must modify the record definition to extract the extra fields and then modify the update definition to store the additional data in a IBM Z Decision Support table.

If a service machine does have messages queued, determine what is causing delays to the service machine. A service machine that consistently has messages queued is being asked to do too much. You might be able to increase the server capacity by increasing its share, giving it reserved pages to reduce paging, or using some form of caching to reduce the time required for DASD I/O.

In some cases, one service machine relies on another service machine. For example, a user might want SQL access on a remote system. Here, the user sends a message to VTAM, VTAM sends a message to the APPC/VM VTAM support (AVS) service machine, and AVS sends a message to the virtual machine on the remote system. A response message then travels back to the original user in the opposite direction. When more than one service machine is involved, investigate each service machine to determine where the user might experience delays. Find the one with the worst queues, that spends the most time waiting on resources, and investigate it. If that is the cause of delay, stop investigating. If not, repeat the process with the next most likely candidate.

Many response-time problems are a direct result of mistuning VTAM or VSCS. Give these service machines high shares, reserved pages to avoid I/O, and QuickDsp.

Analyzing LPAR usage

This section refers to the report and tables available to IBM Z Decision Support only starting from z/VM version 5.1.

In a PR/SM environment you can divide a physical machine into Logical Partitions (LPARs), so that multiple operating systems can run simultaneously. By partitioning the disk, you make better use of the available hardware resources, provided that the partitions are properly defined. In particular, you need to assign or reserve the appropriate number of physical and logical processors to each partition. This can greatly affect the processing resources used by the partition.

To monitor daily LPAR usage, use the z/VM System MIPS Capacity by LPAR report described in the figure below. This report shows you the processing resources used by each LPAR both in terms of MIPS and of percentage with respect to the overall processor capacity. Note, however, that the MIPS available to a particular processor is not measured by the VM Monitor, but needs to be defined in the VMPRF_MIPS_CPU lookup table.

```
                        z/VM System MIPS Capacity by LPAR
                       System: 'XYZVM   ' Period:  ('PRIME')
                                Month: '2000-01-01'

              IUCV        IUCV        IUCV        VMCF        VMCF        VMCF
User         receive      fail        send       receive      fail        send
class        (count)     (count)     (count)     (count)     (count)     (count)
--------  ----------  ----------  ----------  ----------  ----------  ----------
ADMIN         671005          35      711225        2049           0        1003

       IBM Z Decision Support Report: VMPRF_U9
```

Figure 90. z/VM System MIPS Capacity by LPAR report

Other information can be collected from the VMPRF_LPARS_H,_D,_M tables and VMPRF_LPARS_HV,_DV,_MV views.

Chapter 7. Analyzing system operations and security

This chapter describes the System Performance Feature components that provide reports for operations and for problem and change management.

Analyzing JES and NetView messages

The message analysis/automation component of the System Performance feature collects data from the JES2 and JES3 SYSLOG files and the NetView log file and produces reports based on this data.

The message analysis/automation component helps you to identify which messages are candidates for suppression or automation.

Use the MAA Most Frequent Nonsuppressed Messages, Daily report to analyze the most frequent nonsuppressed messages. Look at the message text and decide if this is an important message. If you find the message unimportant from an operational point of view, then you can put this message ID in your message processing facility (MPF) list.

```
            MAA Most Frequent Nonsuppressed Messages, Daily
               JES Complex: 'JES2COMP'  Period: 'PRIME '
                  Date: '2000-01-15' to '2000-01-15'


   Message        Message
     ID            count   Message text
   ----------  ----------  -----------------------------------------
   CCC0000           1279   SE '08.00.03 JOB00038 $HASP165 SE51045L
   010/001            536  *010/001 -K -KIS-ENTER PARAMETERS (AUTO=
   IEA989I            471   IEA989I SLIP TRAP ID=X33E MATCHED
   HASP308            334   $HASP308 VPWPROD  ESTIMATED TIME EXCEED
   HASP530            329   $HASP530 FI72340  ON L5.ST1         1,0
   IEF196I            257   IEF196I IEF237I JES2 ALLOCATED TO SYSLO
   EDG6627A           196  *EDG6627A MA M 052B V(K10900) R(K10900)
   IOS000I            188   IOS000I 52B,89,NCA,02,0600,,**,,INIT348
   EDG6642I           185   EDG6642I VOLUME K10900 LABELLED SUCCESS
   EDG6622I           183   EDG6622I VOLUME K10900 INITIALIZATION S

           IBM Z Decision Support Report: MAA07
```

Figure 91. MAA Most Frequent Nonsuppressed Messages, Daily report

The MAA Messages by Route Code (JES2), Daily report is useful in determining the importance and destination for messages. Route code 1 is, for example, assigned to action messages and displayed upon the master console. Use this report to find messages that are candidates for automation by NetView.


```
              MAA Messages by Route Code (JES2), Daily
              JES Complex: 'JES2COMP'  Period: 'PRIME '
                         Date: 2000-01-15


                         Messages   Single    Multi    Lines per
Route          Message   of total    line      line    multiline
code            count      (%)     messages  messages   message
----------   --------   --------  --------  --------  ---------
   SUPP         12229       71.6     11855       374          5
00               2245       13.1      2233        12          9
08                923        5.4       919         4          2
-OTHER-           621        3.6       339       282          3
03                383        2.2       383         0          0
0102              355        2.1       355         0          0
11                135        0.8       135         0          0
ALL                94        0.6        94         0          0
02                 35        0.2        34         1          2
0810               18        0.1        18         0          0
0208               10        0.1        10         0          0
0211                8        0.0         8         0          0
1011                8        0.0         8         0          0
010207              6        0.0         6         0          0
01                  6        0.0         6         0          0

          IBM Z Decision Support Report: MAA10
```

Figure 92. MAA Messages by Route Code (JES2), Daily report

Use the MAA Messages Passed to NetView (JES2), Daily report to analyze the messages most frequently passed to NetView. If you have messages passed to NetView for no reason, you should update your MPF list to prevent this.

```
              MAA Messages Passed to NetView (JES2), Daily
                JES Complex: 'JES2COMP'  Period: 'PRIME '
                   Date: '2000-01-15' to '2000-01-15'


             Automation
Message       message
  ID           count     Message text
----------   ----------  -----------------------------------------
HASP373             738   $HASP373 SE58186  STARTED
HASP395             710   $HASP395 SE51045L ENDED
010/001             536  *010/001 -K -KIS-ENTER PARAMETERS (AUTO=
IEF403I             476   IEF403I SE51045B - STARTED - TIME=08.03
IEF404I             455   IEF404I SE51045L - ENDED - TIME=08.00.0
HASP308             334   $HASP308 VPWPROD  ESTIMATED TIME EXCEED
HASP530             329   $HASP530 FI72340  ON L5.ST1         1,0
IEF125I             262   IEF125I SE58186 - LOGGED ON - TIME=08.0
IEF126I             214   IEF126I SE58186 - LOGGED OFF - TIME=08.
EDG6627A            196  *EDG6627A MA M 052B V(K10900) R(K10900)

           IBM Z Decision Support Report: MAA05
```

Figure 93. MAA Messages Passed to NetView (JES2), Daily report

Analyzing Tivoli Workload Scheduler for z/OS (OPC) performance

This section describes how to use IBM Z Decision Support to monitor a system that plans and controls its work with Tivoli Workload Scheduler for z/OS.

Tivoli Workload Scheduler for z/OS has its own reports which you use to check for errors when creating and extending plans. The IBM Z Decision Support reports are more useful for looking at long-term trends.

Before collecting Tivoli Workload Scheduler for z/OS data, ensure that your current plan extension batch job allocates the EQQTROUT data set to a log and is not a dummy data set, as it is in the sample job. Schedule the IBM Z Decision Support collection to run after the job that extends the current plan.

You can use the Tivoli Workload Scheduler for z/OS (OPC) component reports to report on jobs that have run on computer workstations, for example, listing the most common error codes. However, you can get similar information using the z/OS System (MVS) component reports. The Tivoli Workload Scheduler for z/OS reports can give you extra information:

- Reports about other workstation types, such as job setup workstations.
- Reports by application owner rather than by job name.
- Error codes—not only job completion codes, but also error codes assigned by Tivoli Workload Scheduler for z/OS itself, which include errors detected by Tivoli Workload Scheduler for z/OS, such as errors resulting from JCL variable substitution.

Using the sample reports to check service levels

Use these reports:

OPC Number of Reruns, Monthly Trend (OPC07)
Reruns are an indication of failing jobs, and this report gives a global view of the system. More detailed analysis of reruns (by application) is possible with these two reports:

- OPC Reruns by Operation Number, Worst Case (OPC14)
- OPC Reruns per Application, Worst Case (OPC06)

OPC Number of Jobs Processed, Monthly Trend (OPC10)
This report gives the number of jobs processed, both those controlled by Tivoli Workload Scheduler for z/OS and those outside.

OPC Missed Deadline by Application Owner ID (OPC12)
Missed deadlines by application owner and priority can be part of a service level agreement. You can get some advance warning of missed deadlines for a critical job by looking at the OPC Operations for Specific Job Name (OPC05) report, which shows the elapsed time trend with any errors that have occurred.

OPC Late-Job Statistics by Application Owner ID (OPC13)
This report gives the number of late jobs, but also gives the numbers of failed and deleted jobs.

Using the sample reports to check performance

Use this report:

OPC Tracking Times by Event Type, Daily trend (OPC11)
This report highlights delays in the passing of events to Tivoli Workload Scheduler for z/OS.

Using the sample reports to check problems in Tivoli Workload Scheduler for z/OS planning

Use these reports:

OPC Missed-Feedback Operations, in Percent (OPC09)
This report highlights operation durations that were outside the range for feedback (so the duration in the database was not adjusted). This can indicate exceptionally high durations (in which case, the problem should also be highlighted on the service level reports) or a problem with the original estimated duration being incorrect, resulting in the real duration being always outside the permitted range for feedback.

Too many missed feedback events are also a sign of too restrictive feedback options, either at the application or at the global (JTOPTS) level.

OPC MCP Events per Caller, Monthly Overview (OPC08)
This report shows alterations to the current plan and the source of the alterations. Too many dialog changes, for example, can be a symptom of poor planning, and some operator intervention could perhaps be avoided by changing definitions in the database.

Analyzing RACF

The Resource Access Control Facility (RACF) controls access to protected resources on both z/OS and z/VM systems.

RACF helps meet security needs by letting you:

- Identify and verify users.
- Authorize users to access the protected resources.
- Control the means of access to resources.
- Log and report various attempts of authorized and unauthorized access to protected resources.
- Administer security to meet security goals.

RACF provides these functions; the installation defines the users and the resources to be protected.

A specific RACF user, called the security administrator, has the responsibility to define users and resources to RACF. (Alternatively, the security administrator can assign other people to do some of this defining.) Besides defining what resources to protect (DASD data sets, minidisks, tape data sets, DASD volumes, tape volumes, terminals, and so on), the security administrator can define and grant the authorities by which users access the protected resources. Thus, the security administrator establishes the access rules that RACF uses to control the user-resource interaction within the organization.

RACF retains information about the users, resources, and access authorities in *profiles* on the *RACF database* and refers to the profiles when deciding which users should be permitted access to protected system resources.

Identifying and verifying users

RACF uses a user ID to identify the person who is trying to gain access to the system, then uses the password to verify the authenticity of that identity. RACF uses the concept of only one person knowing a particular user ID-password combination to verify user identities and to ensure personal accountability. Each user is also assigned to one or more groups. A group is a collection of RACF users who share common access requirements to protected resources or who have similar attributes within the system. Groups reduce the administrative work of maintaining individual access authorities.

Authorizing users to access resources

Having identified and verified the user, RACF then controls interaction between the user and the system resources. RACF controls not only which users may access resources, but also in what way the user may access them, such as for reading or for updating. RACF also can authorize *when* a user can access resources, either by time or day.

Logging and reporting

Having identified and verified the user, and limited access to resources, RACF records the events where attempted user-resource interaction has occurred. An organization can use logging and reporting to alert management to anticipated user activities and system events and to variances from the expected use of the system.

Administering security

Because the security requirements at every data processing organization differ, RACF lets an organization meet its own unique security objectives. RACF lets an organization administer security in several ways:

- Flexible control of access to protected resources
- Protection of user-defined resources
- Choice of centralized or decentralized administration of profiles
- Transparency to end users
- Exits for user-written routines

- Data security monitor
- RACF report writer
- RACF Database Unload and SMF Unload utilities

RACF operation

To visualize how RACF works, picture RACF as a layer in the operating system that verifies users' identities and grants user requests to access resources.

Assume, for example, that a user has been identified and verified to the RACF-protected system, and now wants to modify an existing RACF-protected resource. After the user issues a command to the system to access the resource, a system resource manager (such as data management) processes the request. Part of the resource manager's processing, when RACF is installed, is to ask RACF if this resource is protected. If it is protected, verify that the user can access it, and if requested, modify it. RACF checks various profiles to verify that the user can access the resource and to determine if the user has the required authorization to modify the contents. RACF then returns the results of its check to the resource manager. The resource manager, based on what RACF indicates, either grants or denies the request.

The following figure shows how RACF interacts with the operating system to allow access to a protected resource. The operating system-RACF interaction to identify and verify users is similar in approach.

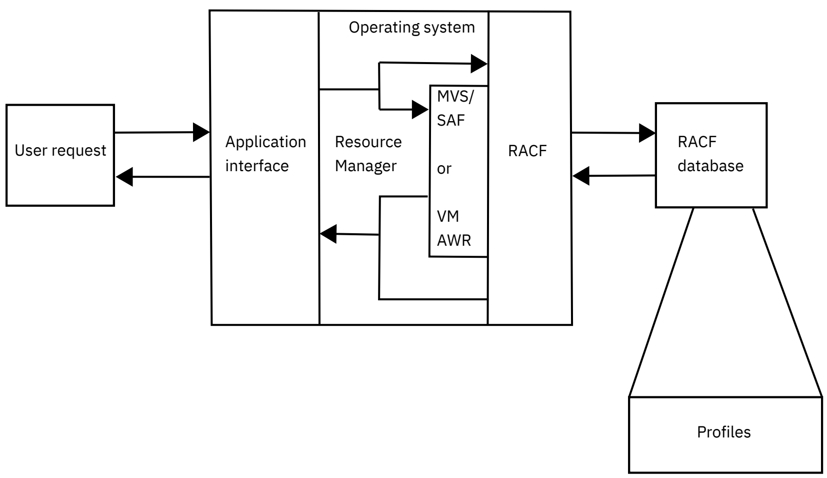

Figure 94. RACF and its relationship to the operating system

Measuring RACF use

The RACF component gets its input from SMF type 80 and type 81 records. Type 80 records are produced during RACF processing. RACF writes a type 80 record when one of these events occurs:

- Unauthorized attempts to enter the system
- Authorized accesses or unauthorized attempts to access RACF-protected resources
- Authorized or unauthorized attempts to modify profiles on a RACF database
- Successful or unsuccessful partner LU verification

Type 81 records are produced at every RACF initialization and contain information such as database name and password rules setup.

Analyzing the data

The RACF component is divided into two subcomponents: RACF Activity and RACF Configuration.

RACF Activity
The RACF Activity subcomponent provides several reports that are useful for auditing the use of RACF. The reports are a complement to the RACF report writer. The IBM Z Decision Support reports list information for:

- Logon/job failures
- RACF command failures
- RACF special user commands
- RACF auditor user commands
- RACF operations user access
- Resource access failures
- Resource accesses
- RACF initialization

You can use these reports to audit the use of your RACF system. The RACF Resource Access Failures report (2) shows which user IDs have unsuccessfully attempted to access RACF-protected resources.

```
                                             RACF Resource Access Failures
                                                    System: 'MVS1'
                                           Date: '2003-12-08' to '2003-12-09'
                                               Minimum security level: 0


Responsible           Sec-  Resource                        Gen- User     Access  Access             Event
   user     Class    level   name                           eric  ID      request allowed Date       desc
----------- -------- ----- ------------------------------- ---- -------- ------- ------- ---------- --------------------
?           DATASET      0 CCC.BEX.CLIST                   Y    ROGER    READ    NONE    2003-12-08 WARNING MSG ISSUED
            DATASET      0 IN.GDDM.SMPACDS                 Y    WILLIAM  ALTER   READ    2003-12-08 INSUFFICIENT AUTH
            DATASET      0 MAP.SLVV1R1.PLSV                N    CORNALE  ALTER   CONTROL 2003-12-08 INSUFFICIENT AUTH
            DATASET      0 PS.V3R2.FORMATS                 Y    WIELATH  UPDATE  READ    2003-12-08 INSUFFICIENT AUTH
                                                                                                    --------------------
                                                                                                  *                    4

PRODTEST    DATASET      5 PROD.TEST2.LOAD                 N    JERRY    READ    NONE    2003-12-08 INSUFFICIENT AUTH
            DATASET      5 PROD.TEST2.LOAD                 N    JERRY    READ    NONE    2003-12-08 INSUFFICIENT AUTH
                                                                                                    --------------------
                                                                                                  *                    2

                                                                                                    ====================
                                                                                                                       6


                                 IBM Z Decision Support Report: RACF06
```

Figure 95. RACF Resource Access Failures report

The RACF auditor can also use the RACF component reports to monitor who is issuing auditor user commands and special user commands. The RACF AUDITOR User Commands—Auditor report (3) lists the users who used AUDITOR commands within the given time frame.

```
                   RACF AUDITOR User Commands - Auditor Report
                                  System: 'MVS1'
                       Date: '2003-12-08' to '2003-12-09'


Command    User ID    Resource name                    Date        Time
--------   --------   ------------------------------   ----------  --------
SETROPTS   SVENNEG    -                                2003-12-08  08.50.46

               IBM Z Decision Support Report: RACF04
```

Figure 96. RACF AUDITOR User Commands - Auditor Report

RACF Configuration

The RACF Configuration subcomponent provides several reports that are useful for having a detailed picture of the RACF database setup. It contains data and report for:

- RACF users
- RACF groups
- RACF users/groups connections
- RACF database and access lists
- Resource general resources and access lists

You can use these reports to have a detailed image of your RACF system at a specific date. For example, the RACF Number of Profiles in the Database is an overview report that provides information about the general RACF database contents:

```
              RACF Number of Profiles in the Database
               Date: '2005-04-09' System: 'MVS1'

Profile
Type           Count
----------   --------
DATASET          220
GENRES           440
GROUP            190
USEr             125

     IBM Z Decision Support Report: RACFC01
```

Figure 97. Example of a RACF Number of Profiles in the Database report

After looking at an overview report, you can search in more detail. For example, to search for the users who have specific privileges, you can use the RACF Users with Particular Privileges:

```
           RACF Users with Particular System Privileges
               Date: '2005-04-09' System: 'MVS1'

User         User              <--------- Privileges --------->
ID           Name              Special    Operations    Auditor
----------   ----------------- ---------  -----------   -------
User1        User no. 1        Yes        No            No
User2        User no. 2        No         No            No
User3        User no. 3        No         No            No

      IBM Z Decision Support Report: RACFC03
```

Figure 98. System Privileges report

Chapter 8. Analyzing WebSphere MQ for z/OS performance

This chapter describes the WebSphere MQ for z/OS component, which provides reports on WebSphere MQ for z/OS.

WebSphere MQ for z/OS messaging software enables business applications to exchange information across different operating system platforms in a way that is straightforward for programmers to implement.

The IBM WebSphere MQ for z/OS range of products provides application-programming services that enable application programs to communicate with each other by using messages and queues. This form of communication, called commercial messaging, provides assured, once-only delivery of messages.

You can use WebSphere MQ for z/OS to separate application programs so that the program sending a message can continue processing without having to wait for a reply from the receiver. If the receiver or the communication channel to it is temporarily unavailable, the message can be forwarded at a later time. WebSphere MQ for z/OS also enables you to provide acknowledgements of messages received.

The programs that comprise a WebSphere MQ for z/OS application can be running on different computers, on different operating systems, and at different locations.

Because the applications are written using a common programming interface known as the Message Queue Interface (MQI), applications developed on one platform can be transferred to other platforms.

What the message manager does

In WebSphere MQ for z/OS, queues are managed by a component called the queue or message manager. The queue manager provides messaging services for the applications, and processes the MQI calls that the applications issue. The queue manager ensures that messages are put on the correct queue or that they are routed to another queue manager.

MQSeries Message Manager Statistics, Daily report

The following figure shows the MQSeries Message Manager Statistics, Daily report, which shows daily statistics for the operation performed by the message manager:

```
                       MQSeries Message Manager Statistics, Daily
                                       System: All
                                       Date: All


                                                                       Close
MQSeries   MQOPEN   MQCLOSE    MQGET     MQPUT     MQPUT1     MQINQ    MQSET     handle
   Id     requests  requests  requests  requests  requests  requests  requests  requests
--------  --------  --------  --------  --------  --------  --------  --------  --------
 VICA       173284    173217    177491     86661     86303         0         0         0
            302530    292377    329909    156498    155575         0         0         0
            219418    219289    261234    109717    119524         0         0         0
                 0         0         0         0         0         0         0         0
             20199     19993     10099     10099         0         0         0         0
                 0         0         0         0         0         0         0         0


                   IBM Z Decision Support Report: MQS04
```

Figure 99. MQSeries Message Manager Statistics, Daily report

What the buffer manager does

The buffer manager is the WebSphere MQ for z/OS component that handles the movement of data between DASD and virtual storage. The buffer pools are areas of WebSphere MQ for z/OS virtual storage that are reserved to satisfy the buffering requirements for WebSphere MQ for z/OS queues. To manage your buffers efficiently, you must consider the factors that affect the buffer pool I/O operations and also the statistics associated with the buffer pool.

The following figure shows the MQSeries Buffer Manager Statistics, Daily report, which shows daily statistics for the buffer manager.

```
                          MQSeries Buffer Manager Statistics, Daily
                                         System: All
                                          Date: All


              Buffer   Average   Average  Getpage  Getpage  Getpage     Synch    Asynch    Times
MQSeries       Pool    number     avail      new   current   notfnd      page      page    buffer
   id           id     buffer    buffer requests  requests  requests    write     write   unavail
--------  --------  --------  --------  --------  --------  ---------  ------   -------  --------
VICA             0      1050      1029         0    642229        33       19         0         0
                        1050       925        65    798527       101       13         0         0
                        1050      1032         0    628889        33       20         0         0
                        1050       257         0        31         0        0         0         0
                        1050       171         0     26484         0        1         0         0
                        1050      1024         0    173758        33        0         0         0

                 1     10500      6613         0    274804      5839        0         0         0
                       10500       948      5050     48576         0        1         0         0
                       10500         0         0         0         0        0         0         0
                       10500      6869    116827   1188797      6001     1591         0         0
                       10500      5714    161121   1561257     12127     5850         0         0
                       10500      4718     91590   1179647      5940     5439         0         0


                       IBM Z Decision Support Report: MQS06
```

Figure 100. MQSeries Buffer Manager Statistics, Daily report

What the data manager does

The data manager is the WebSphere MQ for z/OS component that handles the links between messages and queues. It calls the buffer manager to process the pages that have messages on them.

What the log manager does

The log manager is the WebSphere MQ for z/OS component that handles the writing of log records. Log records are essential for maintaining the integrity of the system if there is a system or media failure.

The MQSeries Log Manager Statistics, Daily report shows daily statistics for the log manager.

```
             MQSeries Log Manager Statistics
                     System: 'MVSY'
                   Date: 2000-02-14


               Log       Log
MQSeries     write      read        Wait      Read
   Id      requests  requests      count   archive
--------   --------  --------   --------  --------
VICA        3062112         0          0         0


IBM Z Decision Support Report: MQS07
```

Figure 101. MQSeries Log Manager Statistics, Daily report

Collecting accounting information

There are three reports that give you WebSphere MQ for z/OS accounting information related to CICS, IMS, or z/OS. These reports provide information about the CPU time spent processing WebSphere MQ for z/OS calls and counts of the number of MQPUT and MQGET requests for messages of different sizes at the user level.

The following are samples of daily accounting reports for CICS, IMS, and z/OS information related to WebSphere MQ for z/OS:

MQSeries CICS Accounting, Daily report

```
                      MQSeries CICS Accounting, Daily
                               System: MVS
                             Date: 2000-02-13


MQSeries  Job       Connect   Transact  Tran     CPU      MQPUT     MQGET
   Id     Id         name        Id     Name   seconds    total     total
--------  --------  --------  --------  ----  --------  --------  --------
VICA      PAICE     IYAYECIC  PAICE                  0         0         0
                    IYAYECIC  CICSUSER  CKTI         0         0         0
                    IYAYECIC  CICSUSER  GP19         2         0       828
                    IYAYECIC  CICSUSER  PP15         2       833         0



                 IBM Z Decision Support Report: MQS01
```

Figure 102. MQSeries CICS Accounting, Daily report

MQSeries IMS Accounting, Daily report

```
                            MQSeries IMS Accounting, Daily
                                     System: All
                                     Date: All


                                                 IMS  IMS
MQSeries  Job       Connect   Transact  Account      PST  PSB          CPU       MQPUT      MQGET
   Id     Id         name        Id      token       id   name       seconds     total      total
--------  --------  --------  --------  ------------ ----- -------  ---------  ---------  --------
VICA                VDF5CHIN            N/A          ADD  QAA              0        123       120

          FISHERD   VDF5NPPG  FISHERD   N/A          SDD  WSS             28        128       123
                    VDF5P     FISHERD   N/A                38            138        134
                    VDF5PPG   FISHERD   N/A                39            139        134
                    VDF5NUOW  FISHERD   N/A                46            146        142
                    VDF5NP    FISHERD   N/A                53            153        153
                    VDF5PUOW  FISHERD   N/A                58            158        158
                    VDF5CHIN  FISHERD   N/A               396            196        199


                         IBM Z Decision Support Report: MQS02
```

Figure 103. MQSeries IMS Accounting, Daily report

MQSeries z/OS Accounting, Daily report

```
                         MQSeries Accounting, Daily
                               System: 'MV25'
                                  Date: All


MQSeries  Job       Connect   Transact    CPU       MQPUT     MQGET
   Id     Id         name        Id     seconds     total     total
--------  --------  --------  --------  --------  --------  --------
VDF5                VDF5CHIN                   0         0         0

          FISHERD   VDF5NPPG  FISHERD         28     24300     24300
                    VDF5P     FISHERD         38     21300     21300
                    VDF5PPG   FISHERD         39     24300     24300
                    VDF5NUOW  FISHERD         46     93000     93000
                    VDF5NP    FISHERD         53     42600     42600
                    VDF5PUOW  FISHERD         58     96000     96000
                    VDF5CHIN  FISHERD        396    211072    863974

          PAICE1    VIDFMSTR  PAICE1         559    656000         0


             IBM Z Decision Support Report: MQS03
```

Figure 104. MQSeries z/OS Accounting, Daily report

Collecting monitoring information

The following reports give WebSphere MQ monitoring information:

- MQSeries Short on Storage and Log manager counts, Daily Report
- MQSeries Wait and Suspend Times, Daily Report
- MQSeries Message Sizes, Daily Report
- MQSeries Data Manager Indicators, Daily Report

The following are samples of daily monitoring reports related to WebSphere MQ for z/OS:

MQSeries Short on Storage and Log manager counts, Daily Report

```
                MQSeries Short On Storage & Log Mngr Counts, Daily
                                  System: 'MV41'
                                 Date: 2005-07-27


                                                   Count
              Count        Count      Count     of Waits    Reads from     No. of
   MQSeries   of SOS       of SOS     of SOS     Unaval    Archive Logs  Checkpoints
      Id     Reductions  bit set on   Abends     Buffer     % of Total    per hour
   --------  ----------  ----------  ------   --------   ------------  -----------
   MQ07               0           0        0          4             0            0


                 IBM Z Decision Support Report: MQS10
```

Figure 105. Example of an MQSeries short on storage and log manager counts, Daily report

MQSeries Wait and Suspend Times, Daily Report

```
                 MQSeries Wait and Suspend Times, Daily
                                  System: 'MV41'
                                 Date: 2005-07-27


            Time    Elapsed    Times    Elapsed    Times    Elapsed    Times
            Wait   Wait Time   Susp.   Wait Time   Susp.   Wait Time   Susp.
MQSeries    Log      Susp.     During    Susp.     During    Susp.     During
   Id      Writes    MQGET     MQGET     MQPUT     MQPUT    MQPUT1     MQPUT1
--------  -------  ---------  -------  ---------  -------  ---------  -------
MQ07            0          1       46         86      106          0        0

MQ08            0          0        3          0        0          0        0


                   IBM Z Decision Support Report: MQS11
```

Figure 106. Example of an MQSeries Wait and Suspend Times, Daily report

MQSeries Message Sizes, Daily Report

```
                         MQSeries Message Sizes, Daily
                                  System: 'MV41'
                                 Date: 2005-07-27


          Expired                                                Kilo     Kilo
            Msg.    Max.Size  Min.Size  Max.Size   Min.Size    Bytes    Bytes
MQSeries  Handled   Msg Read  Msg Read  Msg Write  Msg Write   Written   Read
   Id     in MQGET   MQGET    MQGET     MQPUT      MQPUT       MQPUT1    MQGET
--------  --------  --------  --------  ---------  ---------  -------  ------
MQ07             1   4451857    954458    4608187    1113875    25107   24236

MQ08             6     16469     13320      12490      12418       13     292


                  IBM Z Decision Support Report: MQS12
```

Figure 107. Example of an MQSeries Message Sizes, Daily report

MQSeries Data Manager Indicators, Daily Report

```
                         MQSeries Data Manager Indicators, Daily
                                  System: 'MV41'
                                 Date: 2005-07-27


                                            Full Count
          Count of   Max Req   No. of    Coupling    Avg. Time    Avg. Time
MQSeries  Page Gets   Queue     Req      Facility   for IXLLSTE  for IXLLSTM
   Id     from Disk   Depth   Requeues   Structure     Calls        Calls
--------  ---------  -------  --------  ----------  -----------  -----------
MQ07              0        1         0           0            0            0

MQ08              0        1         0           0            0            0


                  IBM Z Decision Support Report: MQS13
```

Figure 108. Example of an MQSeries Data Manager Indicators, Daily report

Chapter 9. Analyzing WebSphere Application Server (WAS) performance

This chapter provides a brief overview of the WebSphere platform to help the reader become more confident with its terminology and concepts used in this book and, also, to provide monitoring and tuning information. E-business in the z/OS environment consists of a number of components both on the z/OS operating system and on the WebSphere application Server. The z/OS operating system provides the supporting infrastructure, such as directory services (LDAP), persistent storage (Db2), security (RACF), Workload Manager (WLM), and other subsystems. The performance of these subsystems can impact the performance of the WebSphere Application Server environment. The following overview briefly discusses the various components of both z/OS operating system and the Websphere Application Server that can be tuned to improve the performances of the e-business environment.

Overview

One of the strengths of running the WebSphere Application Server on a zSeries server is the z/OS environment itself. The z/OS operating system allows you to scale your web server both horizontally and vertically. Using the z/OS Workload Manager (WLM), additional WebSphere Application servers can be started to meet a sudden increase in demand. The z/OS SecureWay™ Server also provides a robust and secure environment that protects your e-business and other corporate assets. Because they provide functions that are used by WebSphere, a number of components in the z/OS operating system environment can be tuned to both decrease response time and increase the overall throughput of web-enabled applications:

- TCP/IP
- UNIX System Services
- Hierarchical File System
- LDAP
- Workload Manager
- SSL and Security
- LOGGER/resource recovery services (RRS)
- JVM

These components are described in the following sections.

TCP/IP

The TCP/IP protocol is used as the basis for communication between the clients and the HTTP server. It is also used to communicate between the System Management User Interface and System Management Server and other portions of the WebSphere Application Server environment.

UNIX System Services

UNIX System Services (USS) provide the overall environment needed to run the WebSphere Application Server environment and the Java™ applications that run within it. The tight integration of the UNIX platform into the z/OS operating system creates a stable, secure foundation for an e-business, and allows you to use the portability of Java applications. They can run both Web Server and backend applications on the same physical hardware, and can eliminate the latency and overhead caused by inter-platform communication.

File system

The file system is used to store the Java code, the static files and the configuration files needed for running the application servers. It provides means of separating applications and their related files into logical groupings of files and directories.

LDAP

LDAP Directory services are used to provide a method for finding Java objects, such as the servlets and the Enterprise Java Beans (EJBS) from the basis of e-business reusability on the Web. WebSphere Application server V4.01 uses an LDAP server as a naming registry. The data that is used by LDAP is stored in Db2 tables. Both the naming server and the IR server use LDAP DLLs to directly store data in Db2. The applications and related files are actually stored in Db2 tables managed by the System Management Server. The SM server interfaces directly with Db2. The actual J2EE Application Servers use the Java Naming Directory Interface (JNDI) to LDAP to access Java objects.

Workload Manager

The Workload Manager (WLM) is used to automatically scale the number of available servers and balance the workload between them. This functionality provided by the z/OS operating system, increases flexibility for WebSphere running on the zSeries platform.

SSL and Security

SSL and security are needed in the actual Web-enabled environment, and IBM zSeries and z/OS work together to lessen the impact of providing security by using Hardware Cryptographic Co-processors and the z/OS Secureway Server to provide security with a minimum of overhead.

LOGGER/RRS

WebSphere for z/OS requires the use of the resource recovery services Attach facility (RRSAF) of Db2, which in turn requires RRS to be set up. RRS provides services to authorize resource managers, such as database programs and communication managers that manage distributed transactional communications. Registration services, context services, and resource recovery services (RRS) are three separate z/OS components, but it is sometimes useful to think of them as a single function called recoverable resource management services (RRMS), the z/OS syncpoint manager.

JVM

The Java Virtual Machine (JVM) exists in each of the server address space that is used to run the Java applications. Ensure that you have the latest level maintenance because constant improvements in performance are made within each release level. Another important factor is how Java memory is defined and its consequent effect on garbage collection. The JVM has thresholds that it uses to manage its storage. When the thresholds are reached, garbage collection (GC) is invoked to release any unused storage. GC has a significant effect on Java performance.

WebSphere Application Server environment

As well as the operating system components, there are also a number of components and subcomponents of the Web Sphere Application Server environment. It may help to visualize the WebSphere Application Server environment as consisting of the following components and subcomponents:

- Basic environment
- HTTP Server/WebSphere Application Server Plug-in
- Daemon Server
- System Management Servers
- Naming Servers

- Interface Repository Servers
- Application Servers
- J2EE Application Servers with the HTTP/HTTPS transport handler option
- Managed-object framework (MOFW) Application Servers
- System Management User Interface (SMUI)

Application Servers

In addition to the Basic Environment, there can be one or more of the following application servers:

- J2EE Application Server
- MOFW Server (only WebSphere Version 4)

WebSphere for z/OS run time includes servers for both J2EE and CORBA applications, through its J2EE server and MOFW servers respectively. The primary focus for WebSphere for z/OS, however, is its J2EE server, which supports both Enterprise Java Beans and Web components that conform to the J2EE specifications and packaging standards published by Sun Microsystems. These two types of J2EE application component run on a WebSphere for z/OS J2EE server, and can use both:

- The application programming interfaces (APIs) and services that the Java 2 Standard Edition (J2SE) Software Development kit (SDK) provides.
- Enterprise services such as Java Database Connectivity (JDBC), Java Naming and Directory Interface (JNDI), and the Java transaction Service (JTS) and API (JTA)

The J2EE specifications dictate which APIs and services each type of application component may use, and the environment in which they must run. Although both Enterprise beans and Web applications may run in a separate type of container within the J2EE server. Enterprise beans run in the EJB container, and Web applications run in a Web container. These two containers in the WebSphere for z/OS J2EE server conform to the J2EE specifications for runtime environments.

This diagram represents how the WebSphere for z/OS product is initially configured in a monoplex on z/OS. The names of individual WebSphere for z/OS components match the names your installation system programmers have used when first installing the WebSphere for z/OS product. The runtime servers are labelled *server instance*. Server instances comprise of address spaces that actually run code. A *server*, instead, is a logical grouping of replicated server instances. Servers allow you to partition workloads into separate server instances, but still refer to them as a single unit. This is particularly important in sysplex environments, where each system in the sysplex might be running a replicated server instance, but clients outside the sysplex address them as a single server. The client does not know which server instance is actually doing work on its behalf, in fact a subsequent work request from the client may, due to workload balancing, be served by a different server instance in the sysplex.

The following list shows the main WebSphere Application Server components:

1. A full WebSphere for z/OS runtime environment includes several servers: the Daemon, System Management, Naming, and Interface Repository server instances. Although not directly part of WebSphere for z/OS, the run time requires a Lightweight Directory Access Protocol (KLDAP) server.
2. WebSphere for z/OS provides a J2EE server instance for J2EE application components: Enterprise Beans and Web applications. J2EE servers contain at least one EJB container and one Web container. The EJB container manages Enterprise Beans, while the Web container manages Web applications (servlets and JavaServer Pages). The WebSphere for z/OS run time includes a run time that runs in the HTTP server address space. Clients can access Web applications in the following ways:
   - By configuring an HTTP or HTTPS Transport Handler for each J2EE server to receive HTTP or HTTPS requests directly into the control region for that server. Using this option removes the requirement to have an IBM HTTP Server for z/OS configured with the Local Redirector plug-in to route to the Web container.
   - By using the IBM HTTP Server for z/OS in conjunction with the local Redirector plug-in. Using this transport allows the IBM HTTP Server for z/OS to serve as the entry point for HTTP requests into the sysplex. The plug-in routine then forwards the requests to the Web container for processing.

3. WebSphere for z/OS also provides a Managed Object Framework (MOFW) server instance (only WebSphere Version 4), which provides a run time environment for CORBA-compliant components.
4. The runtime server instances use other z/OS functions, such as z/OS UNIX, and TCP/IP. Part of installing WebSphere for z/OS includes configuring these functions for use at run time.

Monitoring Performance

Monitoring performance starts by collecting the appropriate information and then analysing it. The primary goal of tuning is to adjust system resource levels to manage performance to an acceptable level as defined by Service Level Agreement (SLA). For more information, refer to *Defining service-level objectives*.

Typically, monitoring results in reports showing the following:

- Processor utilization by workload type
- Processor storage allocation by workload type
- I/O activity by workload type
- I/O subsystem response time analysis
- Information mirroring levels of resource contention.

SMF Records

SMF records can be collected using IBM Z Decision Support to capture information about the performance of the system and now of the WebSphere Application Environment. After the SMF data has been collected, it has to be interpreted and analysed. To determine what tuning should be done, you first need to measure the performance of your WebSphere Application Server Environment. Important sources of information you can use to help you accomplish this task are the table, view and report objects provided by IBM Z Decision Support.

Once you have identified a particular bottleneck, you can attempt to correct the imbalance or, if the system is already at maximum throughput, take other steps to increase the capacity (that is, increase CPU, storage, I/O paths, and so on). There are a number of SMF record types that can be collected by IBM Z Decision Support to monitor the performance of your e-business environment:

- Record type 30 Address Space- accounting and availability information
- Record type 92 USS HFS information
- Record type 103 HTTP Server information
- Record type 100 Db2 statistics, transaction data collected at event monitoring points
- Record type 101 Db2 accounting, resources used during a transaction
- Record type 102 Db2 performance
- Record type 115 and 116 WebSphere MQ Statistics
- Record type 118 TCP/IP Statistics
- Record type 119 TCP/IP Statistics

The following SMF record types can also be collected and analysed to monitor the overall performance of your system:

- Record type 70 RMF Processor Activity
- Record type 71 RMF Paging Activity
- Record type 72 RMF Workload activity and Storage Data
- Record type 73 RMF Channel Path Activity
- Record type 74 RMF Activity of Several Resources
- Record type 75 RMF Page Data Set Activity
- Record type 76 RMF Trace Activity

- Record type 77 RMF Enqueue Activity
- Record type 78 RMF Monitor I Activity
- Record type 79 RMF Monitor II Activity

SMF type 30 records Process Accounting and Availability data

Type 30 records are written to provide system-wide accounting information on a periodic basis. SMF type 30 records provide resource usage information for traditional z/OS work units such as TSO/E session, started task, or batch job. They are all well suited for the WebSphere Application Server. The WebSphere accounting information can be found in the tables of the IBM Z Decision Support Resource Accounting Component.

```
 REPORT           DRL7.RAFADDRLOG                 LINE 17     POS 1      79

     A                          A             A       A         A         A
 TIMESTAMP                   STCNAME        STEPNO  ACCT1     ACCT2     ACCT3
+--------------------------++--------++-----------++--------++--------++------+
 2002-12-19-09.39.41.670000  IBWJSR2               1  -         -         -
 2002-12-19-11.59.16.290000  IBWJSR2S              1  -         -         -

  A         A         A       A         A          A         A
 ACCT4     ACCT5     SMFID  RACFGRP    USERID    PGMNAME   USERFLD
+--------++--------++-----++---------++--------++--------++--------+
 -         -         S1E    CBACGRP1   CBACRU1   BBOCTL    I
 -         -         S1E    CBASGRP1   CBASRU1   BBOSR     I

      A           A           A           A           A          A
   TCBTIME     SRBTIME     HPTTIME     IIPTIME     RCTTIME   NDISKBLKS
+----------++----------++----------++----------++----------++----------+
  5.441E+03   3.710E+00   0.000E+00   0.000E+00   0.000E+00   0.000E+00
  2.215E+01   2.300E-01   0.000E+00   0.000E+00   0.000E+00   2.000E+00
```

Also the z/OS System (MVS) component table MVS_ACCNT23_PGM_T is populated with rows related to WebSphere Application Server.

For charge back purposes, you can configure your J2EE server by account,by deploying all applications that belong to the same customer into the same generic server. It, then, collects SMF type 30 records at server address space. This assumes your customers are groups of users and that you do not have to charge back to individual users. When a program issues a fork, the activity of the child process is not included in the accounting data for the parent process. That is, when the program issues a fork , the child process has its own SMF type 30 records. The parents' type 30 records do not include the activity of the child process. However, you can associate the parent process with the child process by using the following:

- The process user ID, specified in the SMF30OUI field
- The process user group, specified in the SMF30OUG field
- The process session ID(SID), specified in the SMF30OSI field
- The process parent ID, specified in the SMF30OPP field

The IBM Z Decision Support MVS_OMVSADIS_T and MVS_OMVSADDR_T tables of the z/OS System (MVS) component provide information about:

- Job name
- Program name of UNIX service process
- CPU time
- UNIX Process ID
- Directory Blocks
- File system block read/write
- Read/Write for Network socket
- Total service request

For address spaces that use OpenMVS callable services such as WebSphere for z/OS, a repeating section for resource availability is provided in SMF record type 30. Collection of resource availability data is triggered by the following events:

- Job step termination
- Invoking one of the EXEC functions (substep termination)
- SMF type 30 interval processing
- Invoking z/OS process clean-up (undub)

The resource availability is traced by the already existing AVAILABILITY_x tables and relative reports.

```
                 MVS Availability not within Target, Daily Trend
                       Date: '2002-12-19' to '2002-12-19'


                       MVS                    Up in     Up in
           Sysplex   system   Resource Resource schedule  schedule Objective
Date        name       ID       type     name    (hour)     (%)       (%)
---------- -------- -------- -------- -------- -------- --------- ---------
2002-12-19 UTCPLXBS S1E      WAS      IBWJSR2      1.48      16.5      95.0


                 IBM Z Decision Support Report: MVS74
```

SMF type 70-79 RMF records

Understanding resource usage by workload is the first step to effective performance management. Before you go down to the application level, you should first obtain a view of all workloads in your system to understand your system behavior and try to identify where the problem is. RMF records, SMF type 70-79, provide the system data you need for system and workload analysis. In the workload activity record (SMF record 72), RMF records information about workloads grouped into categories that you have defined in the Workload Manager policy. They can be workloads, service classes, or they can be logically grouped into reporting classes. IBM Z Decision Support provides a very large number of tables, views and reports objects populated by RMF records.

SMF type 92 HFS records

Type 92 records are written at a UNIX System Services level and they provide information about the file system.To collect information about the activity of a USS mounted file system, you must collect SMF type 92 subtypes 5 (unmount) records at the time the file system is mounted and at the time the file system is unmounted. To collect information about the activity of a specific USS file when it is opened and at the moment the file is closed. IBM Z Decision Support MVSPM_OMVS_FILE_H and MVSPM_OMVS_MOUNT_H tables provide detailed information on these metrics.

Note: When running an e-business workload on z/OS, a very large number of HFS files may be opened and closed. Recording of SMF type 92 subtypes 10 and 11 records may lead to a significant reduction performance because of the large number of records being written to the SMF data sets. Unless you have a specific reason, it is recommended that you suppress the recording of SMF type 92 subtypes 10 and 11 records.

SMF type 103 - HTTP Server records

SMF type 103 records are used to monitor the IBM HTTP Server. Though information recorded here pertains only to activity occurring in the HTTP server address space, it is a good source of information. There are two subtypes for HTTP Server type 103 records:

- Subtype 1 Configuration information
- Subtype 2 Performance information

Configuration record data is taken from the server configuration file, httpd.conf, and is written after the HTTP server is fully initialized. Performance record data is accumulated continuosly and written at intervals defined in the httpd.conf by the SMFRecordingInterval directive. IBM Z Decision Support tables INTCON_CONF_X and INTCON_PERF_x contain the following metrics

- Number of threads currently used
- Maximum number of threads defined
- Number of requests
- Number of requests errors
- Number of responses
- Number of responses discarded
- Number of bytes received
- Number of bytes sent
- Number of unknown type bytes
- Number of timeouts

More in general, refer to the following:

- *Analyzing z/OS performance*
- *Analyzing DFSMS storage management data*
- *Analyzing Db2 performance*
- *Analyzing WebSphere MQ for z/OS performance*

WebSphere information

In addition to this and other information, you can now also collect and analyse information that is unique to the WebSphere Application Server. WebSphere for z/OS has its own SMF (type 120) records, which collect additional domain-specific information for WebSphere for z/OS. All this information can be used to report system reliability, analyze configuration, schedule work, identify system resource usage, and perform other performance-related tasks on customer organization. The goal is to create a proactive process where problems are identified before users start complaining. The first step in this process is to define business objectives and identify a way to measure them. If the objectives are not being met, then actions can be initiated to isolate and identify the problem. Once the cause of the problem has been identified, you can decide whether it is a tuning issue or the result of a lack of resources.

SMF type120 - Application Server Container records

SMF type 120 records are generated from WebSphere Application Server servers in full blown configuration. They can be as detailed as one record per transaction (although this might effect the performance). They are mainly used to gather application statistics rather than charge back (for example: CPU time is not recorded in SMF type 120 records). Type 120 SMF records contain information about both the WebSphere Application Server and Container performance and usage. If you wish to record activity at the application level on your WebSphere Application server, SMF records type 120 are one primary source of information.

As stated earlier, the general recommendation is not to turn on specific SMF records unless you are actually going to use them. If knowledge of WebSphere application performance is important, we recommend that you enable the recording of SMF 120 records. You may find that running with SMF type 120 records in production is appropriate, as these records give information specific to WebSphere applications such as response time for J2EE artifacts, bytes transferred, and so forth. If you do choose to run with SMF type 120 records enabled, we recommend that you use interval SMF records rather than activity records. If your choice is also to generate Server or Container interval 120records , it is probably a good idea to synchronize the record 120 intervals for SMF, RMF recording interval. Synchronizing recording intervals for SMF, RMF, and WebSphere components will allow for consistent system workload and application performance information.

Tuning

Tuning your system for the WebSphere Application Server assumes that your basic underlying z/OS system is already performing at acceptable service levels. Information on Performance and tuning for WebSphere on z/OS can be found in the *WebSphere Application Server: Performance Monitoring and Tuning* manual.

Chapter 10. Analyzing WebSphere Message Broker performance

This chapter describes the WebSphere Message Broker component, the terminology, and which reports provide details on Message flow accounting and statistics.

Overview

WebSphere Message Broker enables information packaged as messages to flow between different business applications. Messages can be routed from sender to recipient based on the content of the message. A message flow describes the operations to be performed on the incoming message, and the sequence in which they are carried out.

Each message flow consists of a series of steps used to process a message. The steps are defined in message flow nodes. The work of routing and transforming messages takes place in a broker. A broker is a set of execution processes that hosts one or more message flows to route, transform, and enrich messages. Brokers contain a number of execution groups; processes in which message flows are run. An execution group is a named grouping of message flows that have been assigned to a broker. The broker enforces a degree of isolation between message flows in distinct execution groups by ensuring that they execute in separate address spaces, or as unique processes.

Message flow accounting and statistics details

The details that are available are:

Message flow statistics

A message flow is a sequence of processing steps that execute in the broker when an input message is received.

One record is created for each message flow in an execution group. Each record contains the following details:

- Message flow name and UUID
- Execution group name and UUID
- Broker name and UUID
- Start and end times for data collection
- Type of data collected (snapshot or archive)
- CPU and elapsed time spent processing messages
- CPU and elapsed time spent waiting for input
- Number of messages processed
- Minimum, maximum, and average message sizes
- Number of threads available and maximum assigned at any time
- Number of messages committed and backed out
- Accounting origin


```
                         Daily accounting data for Message Flows
                                MVS system:'MVD0'
                                Broker    :'MQ06BRK'
                                Exec Group:'ECOMPUTE1'
                                Date      :'2006-03-21'


  Message            Total        Total          Total   Total Elapsed        Number        Total
  Name            CPU Time    Wait Time   Elapsed Time       Wait Time   of Messages   Input Size
  --------------  --------    ---------   ------------   -------------   -----------   ----------
  fcompute1              0        16673              0             123             0            0


                  IBM Z Decision Support Report: WMB001
```

Figure 109. Example of Daily accounting data for Message Flows report

Thread statistics

One record is created for each thread assigned to the message flow. Each record contains the following details:

- Thread number (this has no significance and is for identification only)
- CPU and elapsed time spent processing messages
- CPU and elapsed time spent waiting for input
- Number of messages processed
- Minimum, maximum, and average message sizes

```
                        Daily accounting data for Threads
                           MVS system  :'MVD0'
                           Broker      :'MQ06BRK'
                           Exec Group  :'ECOMPUTE1'
                           Message Flow:'fcompute1'
                           Date        :'2006-06-21'


                 Total Messages          Total      Total       Total   Total Elapsed          Total
      Thread No       in Thread   Elapsed Time   CPU Time   Wait Time       Wait Time   Message Size
      ---------  --------------   ------------   --------   ---------   -------------   ------------
             24             144          51628       5792      172942      5568430000          13776


                           IBM Z Decision Support Report: WMB003
```

Figure 110. Example of Daily accounting data for Threads report

Node statistics

A message flow node is a processing step in a message flow.

One record is created for each node in the message flow. Each record contains the following details:

- Node name
- Node type (for example MQInput)
- CPU time spent processing messages
- Elapsed time spent processing messages
- Number of times that the node is invoked
- Number of messages processed
- Minimum, maximum, and average message sizes

```
                              Daily accounting data for Nodes
                                 MVS system  :'MVD0'
                                 Broker      :'MQ06BRK'
                                 Exec Group  :'ECOMPUTE1'
                                 Message Name:'fcompute1'
                                 Date        :'2006-06-21'


   Node                    Total         Total        Total        Number of         Number of
   Name                 CPU Time  Elapsed Time  No Messages  Input Terminals  Output Terminals
   -------------------  --------  ------------  -----------  ---------------  ----------------
   inputNode..........      1271         24163           12                0                 9
   outputFailure......         0             0            0                3                 6
   outputTrue.........      1625          1651           12                3                 6


                                 IBM Z Decision Support Report: WMB002
```

Figure 111. Example of Daily accounting data for Nodes report

Terminal statistics

A terminal is the point at which one node in a message flow is connected to another node.

One record is created for each terminal on a node. Each record contains the following details:

- Terminal name
- Terminal type (input or output)
- Number of times that a message is propagated through this terminal

```
                       Daily accounting data for Terminals
                           MVS system  : 'MVD0'
                           Broker      : 'MQ06BRK'
                           Exec Group  : 'ECOMPUTE1'
                           Message Flow: 'fcompute1'
                           Node        : 'inputNode'
                           Date        : '2006-06-21'


                                   Type of       Total Number
               Terminal Name      Terminal     of Invocations
               -------------      --------     --------------
               catch                Output                  0
               failure              Output                  0
               out                  Output                 12


                       IBM Z Decision Support Report: WMB004
```

Figure 112. Example of Daily accounting data for Terminals report (partial view)

Appendix A. Support information

If you have a problem with your IBM software, you want to resolve it quickly. IBM provides a number of ways for you to obtain the support you need.

- Searching knowledge bases: You can search across a large collection of known problems and workarounds, Technotes, and other information.
- Obtaining fixes: You can locate the latest fixes that are already available for your product.
- Contacting IBM Software Support: If you still cannot solve your problem, and you need to work with someone from IBM, you can use a variety of ways to contact IBM Support.

Contacting IBM Support

This section describes how to contact IBM Support if you have been unable to resolve a problem with IBM Z Decision Support.

Before contacting IBM Support, your company must have an active IBM software maintenance contract, and you must be authorized to submit problems to IBM. The type of software maintenance contract that you need depends on the type of product you have. For more information, refer to the IBM Support website at the following links:

IBM Support
: https://www.ibm.com/mysupport/s/

IBM Z Support
: https://www.ibm.com/support/pages/ibm-enterprise-support-and-preferred-care-options-ibm-z

To contact IBM Support to report a problem (*open a case*), follow these steps:

1. Determine the business impact.
2. Describe the problem and gather information.
3. Submit the problem report.

Determining the business impact

When you report a problem to IBM, you are asked to supply a severity level. Therefore, you need to understand and assess the business impact of the problem that you are reporting. Use the following criteria:

Severity 1
: The problem has a *critical* business impact. You are unable to use the program, resulting in a critical impact on operations. This condition requires an immediate solution.

Severity 2
: The problem has a *significant* business impact. The program is usable, but it is severely limited.

Severity 3
: The problem has *some* business impact. The program is usable, but less significant features (not critical to operations) are unavailable.

Severity 4
: The problem has *minimal* business impact. The problem causes little impact on operations, or a reasonable circumvention to the problem was implemented.

Describing the problem and gathering information

When describing a problem to IBM, be as specific as possible. Include all relevant background information so that IBM Support specialists can help you solve the problem efficiently. To save time, know the answers to the following questions:

- What software versions were you running when the problem occurred?
- Do you have logs, traces, and messages that are related to the problem symptoms? IBM Support is likely to ask for this information.
- Can you re-create the problem? If so, what steps were performed to re-create the problem?
- Did you make any changes to the system? For example, did you make changes to the hardware, operating system, networking software, product-specific customization, and so on.
- Are you currently using a workaround for the problem? If so, be prepared to explain the workaround when you report the problem.

Submitting the problem

You can submit your problem to IBM Support in either of the following ways:

Online
: Go to https://www.ibm.com/mysupport/s/, click on **Open a case**, and enter the relevant details into the online form.

By email or phone
: For the contact details in your country, go to the IBM Support website at https://www.ibm.com/support/. Look for the tab on the right and click **Contact and feedback** > **Directory of worldwide contacts** for a list of countries by geographic region. Select your country to find the contact details for general inquiries, technical support, and customer support.

If the problem you submit is for a software defect or for missing or inaccurate documentation, IBM Support creates an Authorized Program Analysis Report (APAR). The APAR describes the problem in detail. Whenever possible, IBM Support provides a workaround that you can implement until the APAR is resolved and a fix is delivered. IBM publishes resolved APARs on the IBM Support website, so that other users who experience the same problem can benefit from the same resolution.

Bibliography

IBM Z Decision Support publications

The IBM Z Decision Support library contains the following publications and related documents.

The publications are available online in the IBM Knowledge Center at the following link, from where you can also download the associated PDF:

https://www.ibm.com/support/knowledgecenter/SSH53X_1.9.0

- *Administration Guide and Reference*, SC27-9055

  Provides information about initializing the IBM Z Decision Support database and customizing and administering IBM Z Decision Support.
- *AS/400 System Performance Feature Guide and Reference*, SC27-9060

  Provides information for administrators and users about collecting and reporting performance data generated by AS/400 systems.
- *CICS Performance Feature Guide and Reference*, SC27-9057

  Provides information for administrators and users about collecting and reporting performance data generated by Customer Information Control System (CICS®).
- *Distributed Systems Performance Feature Guide and Reference*, SC27-9059

  Provides information for administrators and users about collecting and reporting performance data generated by operating systems and applications running on a workstation.
- *Guide to Reporting*, SC27-9066

  Provides information for users who display existing reports, for users who create and modify reports, and for administrators who control reporting dialog default functions and capabilities.
- *IMS CSQ Performance Feature Guide and Reference*, SC27-9058

  Provides information for administrators and users about collecting and reporting performance data generated by Information Management System (IMS).
- *Language Guide and Reference*, GI13-4376

  Provides information for administrators, performance analysts, and programmers who are responsible for maintaining system log data and reports.
- *Messages and Problem Determination*, GC27-9056

  Provides information to help operators and system programmers understand, interpret, and respond to IBM Z Decision Support messages and codes.
- *Network Performance Feature Installation and Administration*, SC28-3205

  Provides information for network analysts or programmers who are responsible for setting up the network reporting environment.
- *Network Performance Feature Reference*, SC28-3206

  Provides reference information for network analysts or programmers who use the Network Performance feature.
- *Network Performance Feature Reports*, SC28-3207

  Provides information for network analysts or programmers who use the Network Performance feature reports.
- *Resource Accounting for z/OS*, SC28-3208

  Provides information for users who want to use IBM Z Decision Support to collect and report performance data generated by Resource Accounting.

- *System Performance Feature Guide*, SC28-3209

  Provides information for performance analysts and system programmers who are responsible for meeting the service-level objectives established in your organization.
- *System Performance Feature Reference Volume I*, SC27-9062

  Provides information for administrators and users with a variety of backgrounds who want to use IBM Z Decision Support to analyze z/OS, z/VM®, zLinux, and their subsystems, performance data.
- *System Performance Feature Reference Volume II*, SC27-9063

  Provides information for administrators and users with a variety of backgrounds who want to use IBM Z Decision Support to analyze z/OS, z/VM, zLinux, and their subsystems, performance data.
- *Usage and Accounting Collector User Guide*, SC27-9064

  Provides information about the functions and features of the Usage and Accounting Collector.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM‘s application programming interfaces.

If you are viewing this information in softcopy form, the photographs and color illustrations might not display.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. For a current list of IBM trademarks, refer to the *Copyright and trademark information* at https://www.ibm.com/legal/copytrade.

Glossary

A

administration
A IBM Z Decision Support task that includes maintaining the database, updating environment information, and ensuring the accuracy of data collected.

administration dialog
A set of host windows used to administer IBM Z Decision Support.

C

collect
A process used by IBM Z Decision Support to read data from input log data sets, interpret records in the data set, and store the data in Db2 tables in the IBM Z Decision Support database.

compatibility mode
A mode of processing, in which the IEAIPS*xx* and IEAICS*xx* members of SYS1.PARMLIB determine system resource management.

component
An optionally installable part of a IBM Z Decision Support feature. Specifically in IBM Z Decision Support, a component refers to a logical group of objects used to collect log data from a specific source, to update the IBM Z Decision Support database using that data, and to create reports from data in the database.

control table
A predefined IBM Z Decision Support table that controls results returned by some log collector functions.

D

data table
A IBM Z Decision Support table that contains performance data used to create reports.

DFHSM
See *DFSMShsm*.

DFSMShsm
Data Facility Storage Management Subsystem hierarchical storage management facility. A functional component of DFSMS/MVS™ used to back up and recover data, and manage space on volumes in the storage hierarchy.

DFSMS
Data Facility Storage Management Subsystem. An IBM licensed program that consists of DFSMSdfp™, DFSMSdss™, and DFSMShsm.

E

environment information
All of the information that is added to the log data to create reports. This information can include data such as performance groups, shift periods, installation definitions, and so on.

G

goal mode
A mode of processing where the active service policy determines system resource management.

I

internal data type
A data type used within IBM Z Decision Support during the processing of data.

J

Java virtual machine
A software implementation of a central processing unit that runs compiled Java code (applets and applications).

K

key columns
The columns of a Db2 table that together constitute the key.

key value
Value that is used to sort records into groups.

L

log collector
A IBM Z Decision Support program that processes log data sets and provides other IBM Z Decision Support services.

log collector language
IBM Z Decision Support statements used to supply definitions to and invoke services of the log collector.

log data set
Any sequential data set that is used as input to IBM Z Decision Support.

log definition
The description of a log data set processed by the log collector.

log procedure
A program module that is used to process all record types in certain log data sets.

lookup expression
An expression that specifies how a value is obtained from a lookup table.

lookup table
A IBM Z Decision Support Db2 table that contains grouping, conversion, or substitution information.

P

processor time
A measure of the time that a job or task controls the processor.

purge condition
Instruction for purging old data from the IBM Z Decision Support database.

R

record definition
The description of a record type contained in the log data sets used by IBM Z Decision Support, including detailed record layout and data formats.

record procedure
A program module that is called to process some types of log records.

record type
The classification of records in a log data set.

repeated section
A section of a record that occurs more than once, with each occurrence adjacent to the previous one.

report definition language
: IBM Z Decision Support statements used to define reports and report groups.

report group
: A collection of IBM Z Decision Support reports that can be referred to by a single name.

reporting dialog
: A set of host or workstation windows used to request reports.

resource group
: A collection of network resources that are identified as belonging to a particular department or division. Resources are organized into groups to reflect the structure of an organization.

resource information
: Environment information that describes the elements in a network.

S

section
: A structure within a record that contains one or more fields and may contain other sections.

service class
: A group of work which has the same performance goals, resource requirements, or business importance. For workload management, you assign a service goal and optionally a resource group, to a service class.

source
: In an update definition, the record or Db2 table that contains the data used to update a IBM Z Decision Support Db2 table.

sysplex
: A set of z/OS systems communicating and cooperating with each other through certain multisystem hardware components and software services to process customer workloads.

system table
: A Db2 table that stores information that controls log collector processing, IBM Z Decision Support dialogs, and reporting.

T

target
: In an update definition, the Db2 table in which IBM Z Decision Support stores data from the source record or table.

threshold
: The maximum or minimum acceptable level of usage. Usage measurements are compared with threshold levels.

IBM Z Decision Support database
: A set of Db2 tables that includes data tables, lookup tables, system tables, and control tables.

U

update definition
: Instructions for entering data into Db2 tables from records of different types or from other Db2 tables.

V

view
: An alternative representation of data from one or more tables. A view can include all or some of the columns contained in the table on which it is defined.

Index

SC28-3209-00