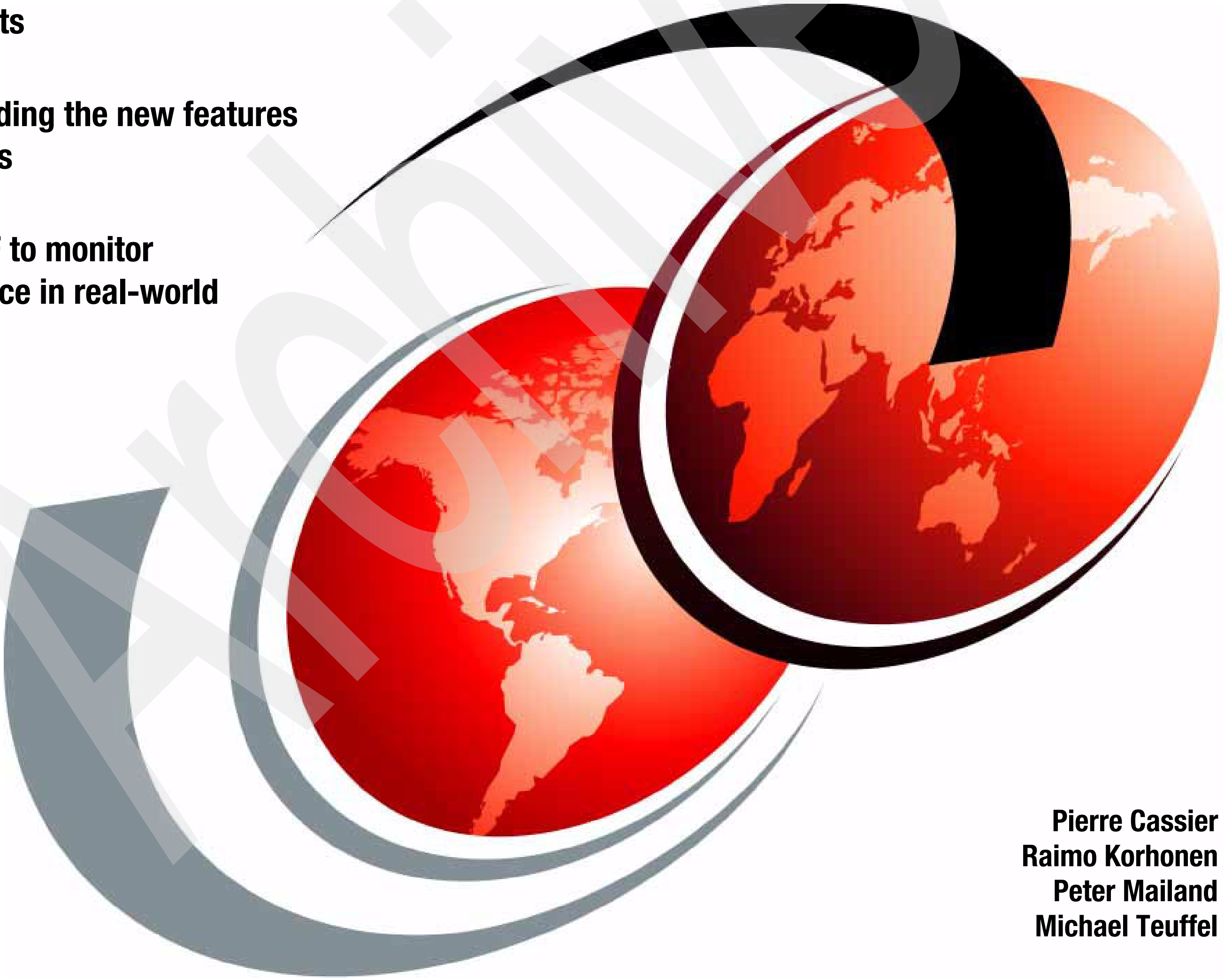
Effective zSeries Performance Monitoring Using Resource Measurement Facility

Setting up and customizing RMF components

Understanding the new features and reports

Using RMF to monitor performance in real-world scenarios



Pierre Cassier
Raimo Korhonen
Peter Mailand
Michael Teuffel

Redbooks

ibm.com/redbooks

International Technical Support Organization

Effective zSeries Performance Monitoring Using Resource Measurement Facility

April 2005

SG24-6645-00

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

First Edition (April 2005)

This edition applies to Version 1, Release 6 of z/OS (product number 5694-A01).

Contents

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.

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 may 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/or 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites 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.

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.

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 illustrates 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.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

@server®
1-2-3®
BatchPipes®
CICS/ESA®
CICS®
DB2®
DFSORT™
Domino®
Enterprise Storage Server®
ESCON®
FICON®
Hiperbatch™
Hiperspace™
ibm.com®
IBM®
IMS™
Language Environment®
Lotus®
MVS™
OS/390®
Parallel Sysplex®
PR/SM™
RACF®
Redbooks (logo) ™
Redbooks™
RMF™
S/360™
Tivoli®
WebSphere®
z/Architecture™
z/OS®
z/VM®
zSeries®

The following terms are trademarks of other companies:

Intel and Intel Inside (logos) are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

Preface

This IBM® Redbook provides a detailed look at Resource Measurement Facility (RMF™), the IBM product designed to simplify management of single and multiple system workloads. RMF gathers data and creates reports that help system programmers and administrators to tune their system optimally, react quickly to system delays, and diagnose and remediate performance problems.

This redbook describes RMF functionality with special emphasis on the newest features, and also presents a review of the older, established components. Detailed instructions for setting up and customizing the components are provided. New features introduced are Spreadsheet Reporter, Distributed Data Server, Linux data gatherer, and Performance Monitor.

A high-level overview of performance analysis concepts is presented, along with a detailed discussion of performance metrics. This information is the foundation for an in-depth look at how RMF can be used to manage systems in the real world. Practical scenarios demonstrate how to use RMF to conduct overall performance evaluations and monitor batch and transactional workloads. The new reporting capabilities are illustrated with numerous examples, in particular those that support the latest workload licensing model, zAAP, UNIX® System Services, WebSphere® Application Server, and Linux.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Poughkeepsie Center. Paola Bari was the Project Leader.

Pierre Cassier is a Certified IT Specialist working for IBM France in Integrated Technology Services. He has more than 20 years of systems programming experience in mainframe environments on MVS™, OS/390®, and z/OS® platforms. His areas of expertise include z/OS, Parallel Sysplex®, WLM, and performance tuning. He teaches WLM and Performance Management courses.

Raimo Korhonen is currently the president of CompMeas Consulting in Finland. He holds a B.Sc. in Electrical Engineering from Tampere Institute of Technology. Raimo joined IBM in 1965 and retired in 1995 as Senior Advisory Systems Engineer. From 1985 through 1995 he worked in the Field Branch Office supporting customers mainly in capacity planning and tuning projects. Prior to that, Raimo was responsible for providing MVS Operating System education at the Customer Education Center.

Peter Mailand is a Software Engineer with IBM System and Technology in Germany. He joined IBM in 1996 as a member of the RMF development team and has 9 years of experience in RMF. His area of expertise is RMF testing and development. This is Peter’s first experience in writing a redbook.

Michael Teuffel has been with IBM Germany for 31 years, working at the Boeblingen Laboratory. He was a Senior IT specialist working in several areas of performance management from S/360™ to z/OS. For many years he was a member of the RMF development team and he performed numerous customer studies in Germany and other countries. Michael retired from IBM in 2001, but continues as a consultant. He has written several books about TSO and z/OS. Michael holds a university doctoral degree in mathematics.

Thanks to the following people for their contributions to this project:

Robert Vaupel from the WLM development team in the IBM Boeblingen Laboratory. Together with Michael Teuffel he is co-author of the book *Das Betriebssystem z/OS und die zSeries®* (The operating system z/OS and the zSeries) which was published in 2004 in Germany.

Richard Conway
International Technical Support Organization, Poughkeepsie Center

Harald Bender
IBM Germany

Oliver Benke
IBM Germany

Laura Blodgett
IBM Poughkeepsie

Pekka Hänninen
IBM Finland

Matthias Gubitz
IBM Germany

Peter Muench
IBM Germany

Nicolas Nardelli
IBM Germany

Alvaro Salla
Consultant

Thanks to the following people for their contribution to the review of this redbook:

Linda August, Harald Bender, Joan Kelley, Alain Maneville, Jim McCoy, Franco Meroni, Judy Rogers, Alvaro Salla, Robert Vaupel, and Kathy Walsh for the technical review

Alison Chandler for the editorial review

Ella Buslovich for the graphical review

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our Redbooks™ to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an Internet note to:

  redbook@us.ibm.com

- Mail your comments to:

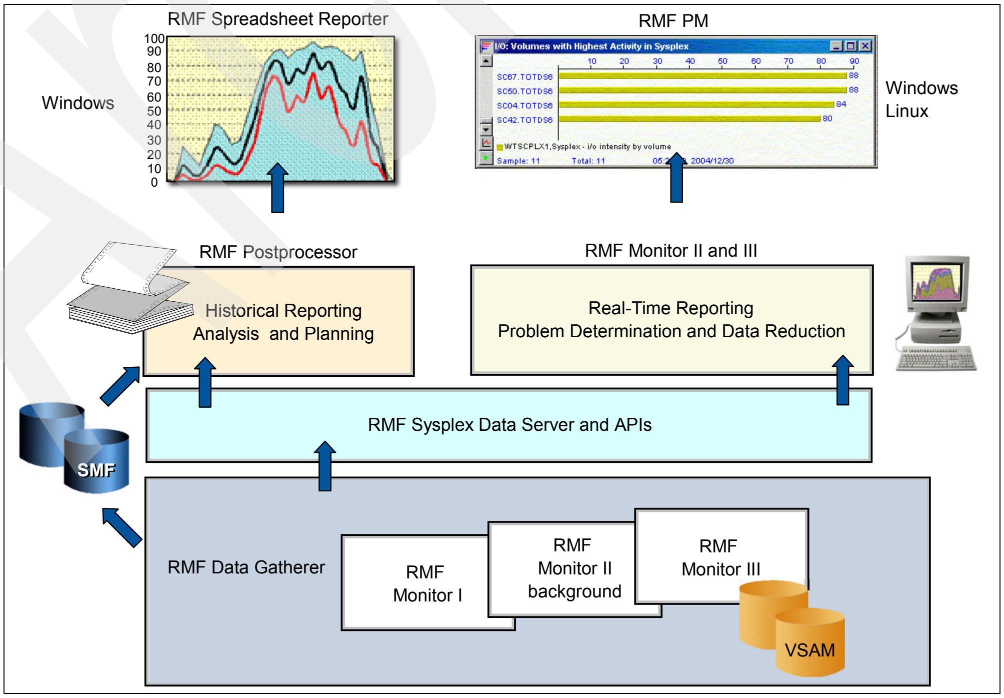
  IBM Corporation, International Technical Support Organization
  Dept. HYJ Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Part 1

RMF components

In this part we introduce the components and elements that are part of the RMF product. Many enhancements have been made to the product in the past few years and this section introduces these new features.

RMF has expanded with respect to system operation and performance that is reported on, and with respect to the interfaces available. While the traditional batch and ISPF reporting remains, flexibility is greatly enhanced with the addition of the Sysplex Data Server and a number of PC-based front ends. A graphical overview of many of the interfaces and functions provided by RMF in z/OS 1.6 is shown in the following figure.



1

RMF data gatherers

In order to be able to report on system performance, RMF must first gather information about both the system and its activity. In this chapter, we discuss the following:

- Long term data gathering with Monitor I and Monitor III
- Snapshot monitoring with Monitor II
- Short term data collection with Monitor III
- Linux data gathering with the RMF Linux data gatherer

1.1 Gathering data

Fundamental for all performance monitoring is data gathering, which is the topic of this chapter. *Resource Measurement Facility (RMF)* is the data gatherer for systems running under z/OS. And now RMF's capabilities have been expanded to include data gathering on Linux systems, too. In this chapter, we discuss the key aspects of gathering and storing data in order to have it available for further processing with the reporter functions of RMF.

1.1.1 Monitors for data gathering

There are three types of time intervals used by RMF monitors for data gathering on z/OS systems:

- Long term data gathering with Monitor I and Monitor III
- Snapshot monitoring with Monitor II
- Short term data collection with Monitor III

In addition, data gathering for Linux is now available.

Long term data gathering and reporting with Monitor I and Monitor III means that gathered data is collected in records of 15- or 30-minute intervals (you can specify other values) for future reporting. You can use these records for long term reporting (for example, a time frame of one day, one month, or more) depending on your requirements.

In contrast, short term data gathering and reporting are defined in seconds or minutes. For Monitor III, the default time interval for gathering or reporting data is 100 seconds, but it can be changed to 60 seconds to have each reporting interval consists of one minute.

Monitor II is defined as *snapshot* monitoring, mainly because of the reporter session, in which you get a report each time you click Enter. Since you are unable to click Enter during a gatherer session, which is also called a "background" session, the length of each gathering interval must be defined explicitly.

The system operator starts all monitors as non-interactive (background) sessions. The operator has a variety of options available to determine the type of data to collect and where it is to be stored. The data gathering functions run independently on each system, but each monitor can be started sysplex-wide by a single operator command. You can run data gathering on each z/OS system, and use the Sysplex Data Server (SDS) to have all the data available on the one system where you run your performance management tasks.

The RMF Linux data gatherer (rmfpms) is similar to Monitor III, a short term data collector, available for Linux on zSeries and Intel®. You need to start rmfpms on each system that you want to monitor. You can analyze the gathered data using RMF Performance Monitoring (RMF PM) or the RMF Web browser interface.

1.1.2 Storing data

RMF stores z/OS data in two types of records:

- All three monitors write SMF records (type 70 – type 79) if you define the appropriate SMF recording options.
- In addition, Monitor III writes VSAM records to in-storage buffers or into RMF-owned VSAM data sets.

The SMF synchronization function ensures that records are written from all monitors in the sysplex for the same intervals.

The RMF Linux data gatherer stores data in a repository directory. The data is automatically archived. Then you can retrieve it for further analysis.

1.1.3 Defining measurement options

There are several ways to define what data should be measured, how it should be measured, and when it should be measured.

> **Tip:** First run your measurements with the default options that RMF offers you without modification – you can apply modifications later if they are necessary.

All measurement options for RMF are defined in parmlib members ERBRMF*xx* (*xx*=00,...), and RMF has defaults for all options that have not been specified in an active parmlib member. During the installation process for RMF, the ERBRMF members are stored in the parmlib. They can be modified, if necessary, and new members can be added. Furthermore, options can be modified during the start procedures for RMF by operator commands. You can find details on this in Chapter 3, “Setup and customization of the traditional favorites” on page 109. The *RMF User’s Guide* contains a detailed description of these parmlib members.

RMF offers a wide spectrum of options to define the scope of data gathering. For all monitors, you have to distinguish among different types of options:

- Which activities to monitor
- The time frame and frequency for monitoring them

Depending on which monitor you use, there are additional options available.

1.1.4 Using the RMF Sysplex Data Server to access data across the sysplex

The RMF Sysplex Data Server (SDS) is a distributed RMF function. It is started as an identical copy on each system of the sysplex. Each copy of the data server communicates with all other copies in the sysplex. RMF uses this sysplex communication method to provide access to distributed RMF measurement data from any point in the sysplex.

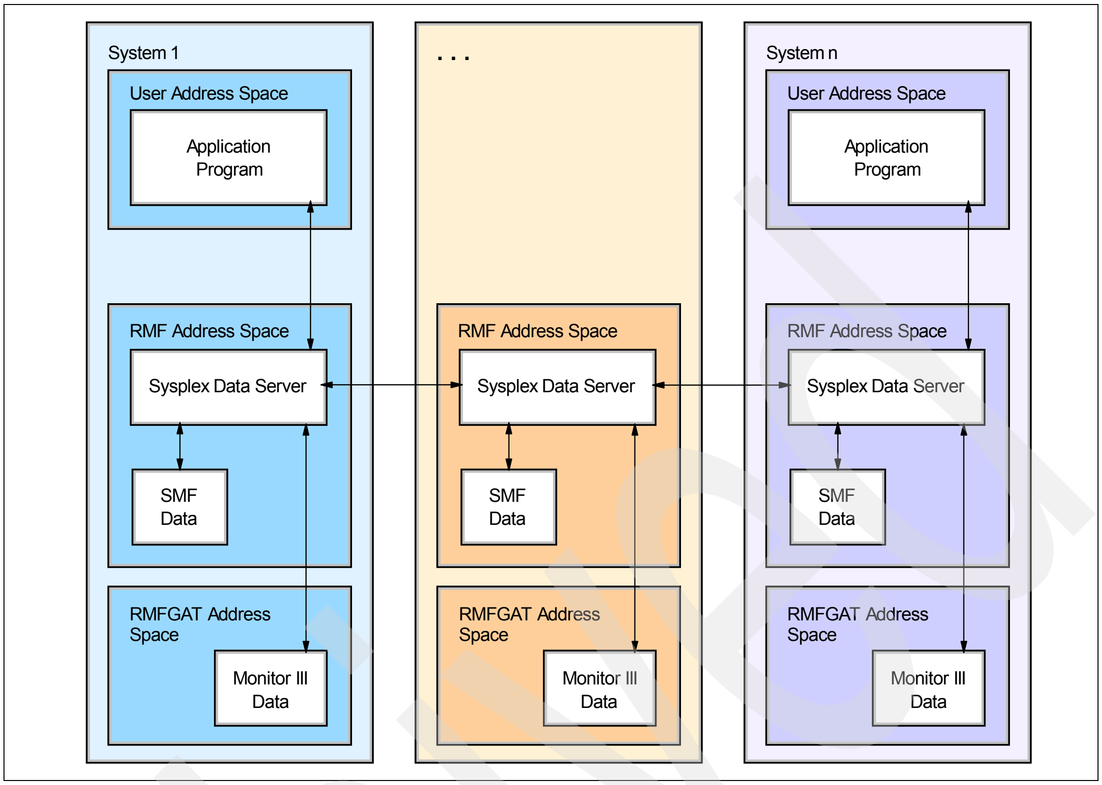


Figure 1-1 RMF Sysplex data server – data flow

SDS is always active when the RMF address space is running. You can access all types of RMF and SMF data collected in the sysplex by using SDS Application Program Interfaces (APIs). These APIs are invoked as callable services by the RMF reporter sessions or by other applications. APIs can access:

- Monitor I, II, and III SMF data
- Monitor III VSAM data
- SMF data of any other type

For further details on these APIs, refer to 2.9, “Using RMF application programming interfaces” on page 107.

To call the RMF services for SMF data, you need authorization to access the SMF data. For details, see “Controlling access to RMF data for the sysplex data services” on page 116.

Sysplex data gathering services for SMF data

RMF stores SMF data in a wraparound buffer. You can choose to create an SDS SMF buffer when you start RMF. You specify the size of the buffer and the types of SMF records to be stored in it as a parameter at RMF startup. The sysplex data services will return SMF data when the buffer exists on at least one system in the sysplex. This system does not need to be the same system that the calling program is running on. SDS returns data only from systems on which data buffers have been created.

Sysplex data gathering services for Monitor III data

You can access data collected by Monitor III data gatherer sessions using the RMF Monitor III Sysplex Data Retrieval Service. Any application program can specify the name of the system from which the Monitor III data is requested. Analogous to SMF data, Monitor III data can be returned from those systems where the Monitor III data gatherer session is active.

Sysplex data gathering services for Monitor II data

Your application program can use this service to create and retrieve Monitor II SMF records (type 79). You do not need to have a Monitor II background session running on the system from which you request the data. Note the difference between this and the Monitor III data gathering service for SMF data, which collects only records created by active monitoring sessions.

> **Note:** It is important to gather data on every system in the sysplex; otherwise, you cannot get a comprehensive report of the total sysplex activity, and the reports will be incomplete.

1.2 Long term data gathering with Monitor I and Monitor III

Monitor I and Monitor III provide long term data collection information about system workload and resource utilization, and include information regarding all the hardware and software components of your system: processor, I/O devices, storage activities and utilization, as well as resource consumption, activity, and the performance of multiple groups of address spaces. Data is gathered for a specific cycle time, and consolidated data records are written at a specific interval time. The default value for data gathering is one second and the default value for data recording is 30 minutes. You can set these values according to your requirements and change them whenever you need to.

1.2.1 Data gathering with Monitor I

Monitor I creates SMF records (type 70–78) for different activities in the system that you need to monitor. Most of these activities are hardware-related, but not all. In addition, Monitor I can produce printed reports at the end of each measurement interval.

The following list summarizes all activities that can be monitored.

CACHE	Cache activity
CHAN	Channel path activity
CPU	Processor activity
CRYPTO	Cryptographic hardware activity
DEVICE	Device activity
ENQ	Contention activity
ESS	Disk system statistics
FCD	FICON® Director activity
IOQ	I/O queuing activity on logical control units
PAGESP	Page data set activity
PAGING	System paging activity
VSTOR	Virtual storage activity
WKLD	System workload
TRACE	Traces specific variables

Data gathering for options CACHE, ESS, and FCD should be performed only on one system in the sysplex to avoid duplicate data; for CACHE and ESS it needs to be the same system.

If you are wondering why you do not see the data gathering options for the Coupling Facility, for UNIX System Services, or XCF, keep in mind that this data is gathered by Monitor III, and not by Monitor I.

The second set of options is required to specify the duration and granularity of the measurements:

CYCLE	Defines the length of the cycle, at the end of which RMF makes sampling observations
INTERVAL	Defines the length of the reporting interval
RECORD	Writes the measured data to the SMF data set
STOP	Defines the desired duration of the Monitor I session
SYNC	Synchronizes the reporting interval with SMF

Last but not least, there are specific options that can be used to tailor Monitor I accordingly:

EXITS	Executes user exits when gathering and reporting
MEMBER	Refers to parmlib member with session options
OPTIONS	Prints option list at the operator console
REPORT	Produces printed interval reports
SYSOUT	Defines output class for printed reports

What is running when no changes have been made

As recommended previously, you can run your measurements without any modifications to the RMF-supplied parmlib member ERBRMF00 (for Monitor I). What does this mean? Most options in the activity list are activated – with some exceptions: NOENQ, NOESS, NOFCD, and NOTRACE (this can be overwritten if you are really interested in that data). The options DEVICE and IOQ mean that activities and I/O queuing for DASD devices will be gathered. If you need to measure additional devices such as tapes or graphical devices, you can specify this directly.

Let’s have a look at the second set of options.

CYCLE(*nnnn*) specifies the length of the cycle at the end of which sampling observations are to be made, where *nnnn* is the number of milliseconds. The valid range is from a minimum of 50 to a maximum of 9999 milliseconds. The default value is 1000 milliseconds; this is equivalent to specifying CYCLE without a parameter.

RMF combines all data that has been gathered for each cycle into records which cover longer time periods, called *report intervals*. These intervals are defined by two options, SYNC and INTERVAL. SYNC specifies whether the interval is to be synchronized with SMF, or on the minute with the RMF interval synchronization mechanism. SYNC(SMF) is the default and specifies that RMF will synchronize its interval using SMF’s global interval and synchronization values. INTERVAL specifies the length of the reporting interval, but is ignored when SMF synchronization is defined.

Important: We highly recommend that you do not change the SYNC option; otherwise, it may cause problems when creating sysplex reports from the data gathered in the different systems of the sysplex.

With the option STOP, you can define the runtime of Monitor I. Typically, Monitor I runs as long as the system is running since NOSTOP is the default value.

The third set of options allows you to perform functions in addition to the system defaults.

The options MEMBER and OPTIONS enable the operator to change options during the RMF start procedure. You can select option MEMBER to use a different parmlib member from ERBRMF00. For OPTIONS, you can be prompted to accept the current option list or modify it. NOPTIONS is the default, which starts RMF without interrupt.

Monitor I offers the capability to create printed reports at each measurement interval. This can be useful when you are testing a system, but it is not designed for a production environment. There is no function that provides Monitor I sysplex-wide reports; these reports can be generated only by the Postprocessor. You can specify this by the option REPORT, while NOREPORT is the default value. In addition, you can define the output class for the reports using SYSOUT(*class*).

In addition, if you do not need to gather the SMF records, you can define this with the option NORECORDS. Otherwise, run with the default option, RECORD, to gather data for further reporting.

With the option EXITS, you can execute user exits during data gathering for reporting.

For those who want to get background information on statistics

Although RMF uses information provided by the system of the measured hardware for some reports (for example, CMBs, LPAR busy, MVS busy), most of the data in the paging, page data set, processor, trace, virtual storage, CPU, I/O queuing, and device activity reports is statistically sampled. According to statistical theory, the accuracy of sampled data increases with the number of samples taken of random events. Therefore, you would expect to observe more precise results with decreased CYCLE time (for a fixed INTERVAL value), or with increased INTERVAL length (for a fixed CYCLE value). For example, 400 samples taken of random, independent events provide a value that with 90% confidence should fall within 4% of the true value; 1,600 samples of random, independent events decrease the expected range of error to 2%, with 90% confidence.

However, pure, statistical predictions are not always applicable to a software measurement tool such as RMF because the assumptions on which they are based (unbiased random, independent samples and an infinite population) might not hold true in an operating environment. Bias might occur because RMF samples internal indications of external system events. Thus, RMF values might not precisely approach the values measured by a hardware measurement tool.

The independence assumption becomes less and less realistic as CYCLE gets very small. As CYCLE gets smaller, each sample is more likely to find the system performing the same functions as in the previous sample; therefore, the new sample adds little additional information. The use of a smaller CYCLE value (while holding INTERVAL constant) should not be significant to accuracy, but any increase in accuracy might be of questionable benefit when compared with the system overhead that is introduced. A reasonable minimum CYCLE value is a function of the timing characteristics of the hardware being measured.

1.2.2 Data gathering services with Monitor III

Monitor III data gathering services write both long term and short term records. The long term monitoring writes SMF records and is described here. The short term monitor writes VSAM records and is described in 1.4, “Short term data collection with Monitor III” on page 11.

Monitor III writes SMF records (type 74) for:

- Coupling Facility activities
- HFS statistics
- OMVS kernel activities
- XCF activities

All these records are written automatically, so there are no options available to define this data. On the other hand, it is impossible to suppress this data gathering.

You can create reports based on these records in the same way you create reports with records that have been gathered by Monitor I.

1.3 Snapshot monitoring with Monitor II

The scope of Monitor II data gathering is predominantly related to single address spaces or resources, giving snapshots of the current status. You can collect data about address space activities and resource consumption. You can also collect data about the processor, DASD volume, and the storage activities and utilization. With Monitor II, it is possible to monitor one specific job or volume continuously.

You can run Monitor II as a background session to create SMF type 79 records. This session is started by the operator, and all options are defined in a parmlib member or by operator commands. (There are also records type 72 available for storage utilization without any Postprocessor reporting option.)

The following list includes the options for gathering information on various activities in the system:

ARD	Address space resource consumption
ARDJ	ARD report for a particular job
ASD	Address space state data
ASDJ	ASD report for a particular job
ASRM	Address space SRM data
ASRMJ	ASRM report for a particular job
CHANNEL	Channel data
DEV	Device data
DEVV	Device data for a specific device
IOQUEUE	I/O queuing data
PGSP	Page/swap data set measurements
SENQ	System enqueue contention
SENQR	System enqueue reserve data
SPAG	System paging activity
SRCS	System real storage/CPU/SRM data

The default parmlib member ERBRMF01 has all options turned off except ASD to gather data for all address spaces. The operator usually has a specific reason, or the need to monitor special activities, to run the Monitor II gatherer session. Therefore, you have to define the options for these measurements explicitly in a parmlib member (either in ERBRMF01 if you want to make this the default, or in another ERBRMFxx member), or with the start command for the gatherer session. For more information, refer to the description in 3.3.1, “Customization of the Monitor II background session” on page 123.

SINTV	Gathering interval
STOP	Session length
DELTA	Presents data as interval deltas

With SINTV and STOP you can define the length of a gathering interval and the length of the session.

DELTA is specific for Monitor II. It reflects the capability to gather data in *delta* mode, which means that certain fields in some reports, such as the processor (CPU) time in the ARD report, reflect values that show the change since the previous interval. The first request for the report shows the value RMF detects at that time; all subsequent invocations of the report show only the change since the previous interval. With option NODELTA, the gatherer is running in total mode and has the cumulative total since the beginning of the Monitor I interval or the creation time of the address space or IPL.

The following list includes the session, reports, and SMF options:

MEMBER	Refers to parmlib member with session options
OPTIONS	Prints option list at the operator console
RECORD	Defines SMF recording
REPORT	Produces printed interval reports
SYSOUT	Defines output class for printed reports
USER	Executes user exits

As mentioned previously, the Monitor II gatherer session is not usually part of a typical production environment; therefore, you probably do not want to use the RMF-supplied parmlib member ERBRMF01 as the default for a gatherer session. You have to define the measurement options according to your requirements. Refer to the *RMF User's Guide* for details, or you can just modify ERBRMF01 or ERBRMF03, where you will see the syntax for all the options.

1.4 Short term data collection with Monitor III

The Monitor III gatherer session has a typical gathering cycle of one second. Consolidated records are written for a range of time (the default is set to 100 seconds). You can collect short term data and continuously monitor the system status to solve performance problems. You get actual performance data (response times, execution velocity) on a very detailed level to use for future comparison with performance policy goals. You can collect data that indicates how fast jobs or groups of jobs are running – this is called *workflow* or *speed*. You also get data that shows how resource-intensive jobs are using processor, DASD devices, and storage – the reports describe this utilization under the term *using*.

There is information about *delays*, which are important indicators of performance problems. This simplifies the comparison of reports created by Monitor I and Monitor III data.

Most data that is available for the Monitor III Reporter is gathered automatically by the gatherer function, but there are also system resources where you can define the level of detail you want for data gathering, such as:

CACHE	Defines cache data gathering
CFDETAIL	Defines data gathering for Coupling Facility structures
HFSNAME	Controls data set recording for z/OS UNIX file systems
IOSUB	Controls data set recording of I/O-subsystem and channel-path activity
OPD	Defines data gathering for OMVS process data
VSAMRLS	Controls data gathering for VSAM RLS activity

If you follow our recommendation to run with the RMF-provided options, you will get data gathered for all of the listed activities with the exception of Coupling Facility structures. CFDETAIL should only be turned on if problem determination data is required; in which case, it can be turned on dynamically using the **`MODIFY RMF`** command.

The following set of options is required to specify the duration and the granularity of the measurements:

CYCLE	Sets the length of the cycle at the end of which RMF samples the data
DATASET	Controls data set recording of sampled data
MINTIME	Specifies the interval at which data samples are summarized
STOP	Sets the duration of the data gatherer interval
SYNC	Synchronizes MINTIME within the hour

WSTOR Sets the size of the RMF local storage buffer

In this list, there are options similar to the options for Monitor I. Some are the same: CYCLE defines by default 1000 milliseconds cycle time, and the default value NOSTOP lets the gatherer run until it is stopped explicitly. SYNC now has nothing to do with SMF synchronization (no SMF records will be written), but now SYNC synchronizes the measurement interval within the hour.

In addition, there are three options that are unique to Monitor III. MINTIME defines the interval at which data samples are summarized. If you don't change this option, the interval is a duration of 100 seconds. The collected data samples for one interval are called a *set-of-samples*. With option DATASET, you control the recording of samples to the VSAM data sets. Detailed instructions for performing this task are in 3.4.1, "Customization of the Monitor III gatherer" on page 124. This option is required if you want to retain the data you have collected for future reporting. If you do not define VSAM data sets, only those samples which are currently available in the RMF local storage buffer can be used for online reporting. The RMF local storage buffer's size is defined by the option WSTOR with a default value of 32 MB.

MEMBER Specifies parmlib members containing session options
OPTIONS Controls display of the current options at the start of a session
RESOURCE Specifies the job entry subsystem (JES) to be used
SYSOUT Specifies the SYSOUT class for gatherer messages

From this list, you recognize MEMBER and OPTIONS – they have the same definitions for Monitor III that they do for Monitor I. SYSOUT is defined with a default value of output class A. Option RESOURCE defines by default JES2 as the job entry subsystem to be used. If your system is running with JES3, you must modify parmlib member ERBRMF04 accordingly.

1.4.1 Common Monitor III report measurements

Monitor III implements a way of measuring performance named Contention Analysis. Using this technique, all address spaces and enclaves have their state sampled in a timely basis. These states are: Using, Delay, Idle, and Unknown. Some of these states are broken into substates. Monitor III also introduced the metric named Workflow%, which relates Using and Delay counters. All this data is captured (sampled and consolidated) by RMFGAT address space. This information is shown in the traditional RMF Monitor III reports, such as: Delay, Workflow exception, Using, and Execution velocity reports.

Using samples

PROC The number of address spaces found using one or more processors (which can be standard CPs or zAAPs). An address space is considered using one or more processors when it has ready work (meaning any ready SRB, interrupted ready task, asynchronous exit routine, or TCB is on the dispatching queue) that could be dispatched by the processor on which the Monitor III data gatherer is running.

DEV The number of address spaces found using one or more devices. An address space is considered using one or more devices when it issues an I/O request. However, because the channel subsystem accepts an I/O request whether the device, control unit, or both are busy or not, the requests might or might not be delayed (queued) in the channel. Therefore, the using requestors for devices may also contain an unknown amount of delay. You must consider this delay when interpreting the workflow value.

Delay samples

PROC The number of address spaces found waiting for a processor (which can be standard CPs or zAAPs). An address space is considered waiting for a processor when it has at least one ready unit of work that is not dispatched. However, the address space with the first ready unit of work is not considered delayed because it would have been using the processor currently being used by Monitor III (if Monitor III was not using the processor). Primary source fields referenced in this calculation are the same as those listed under PROC for using samples.

DEV The number of address spaces found waiting for a measured device. An address space is considered to be waiting for a measured device when at least one I/O queue element in the I/O queue for the device identifies the address space as the issuer of the I/O request but the request is not active. I/O requests queued in the channel for devices are considered to be using the device, and therefore an unknown amount of delay is missing from the delayed requestor count for devices.

In addition, the reports show delay values for storage, enqueues, operator replies, HSM, JES, and XCF.

Address space/enclave workflow (%)

The workflow of an address space or enclave represents how it uses system resources and the speed at which it moves through the system in relation to the maximum average speed at which it could move through the system. The speed at which the system performs the work depends on the simultaneous work requested by other address spaces or enclaves.

A value from 0% to 100% indicates the workflow within the report interval. A low workflow value indicates that it has few of the resources it needs and is contending with other jobs for system resources. A high workflow value indicates that it has all the resources it needs to execute, and that it is moving through the system at a relatively high speed.

For example, a job that would take four minutes to execute, if all the resources it needed were available, would have a workflow of 25% if it took 16 minutes to execute.

The following formula defines the workflow of a single address space:

$$\text{Workflow (\%)} = \frac{\text{\# Using Samples}}{\text{\# Using Samples } + \text{ \# Delay Samples}} \times 100$$

Execution velocity

The execution velocity is a WLM measure and a type of goal. Execution velocity goals define how fast work should run when ready, without being delayed for processor, storage, I/O access, and queue delay. Execution velocity goals are intended for work for which response time goals are not appropriate, such as started tasks, or long running batch work.

$$\text{Execution velocity (\%)} = \frac{\text{\# Using samples}}{\text{\# Using samples + \# Delay samples}} \times 100$$

Execution velocity versus workflow

Although the formula for the execution velocity and the workflow are similar, the resulting value is calculated in a slightly different way. In the execution velocity calculation, only the processor, the storage, and DASD devices are considered; these are the resources which are under control of WLM. The workflow calculation reflects all system components (for example, tape activities or delays caused by tape mounts or HSM). This can lead to having different numbers for these fields in the report for the same address spaces.

Address space using (%)

Jobs getting service from hardware resources (PROC or DEV) are *using* these resources. The use of a certain resource by an address space can vary from 0% to 100%, where 0% indicates no use of the resource during the report interval and 100% indicates that the address space was found to be using the resource in every sample during that period.

$$\text{Using (\%)} = \frac{\text{\# Using Samples}}{\text{\# Samples}} \times 100$$

Address space delay (%)

The delay of an address space represents a job that needs one or more resources but that must wait because it is contending for the resources with other users in the system. The delay of an address space for a specific resource or for all resources can vary from 0% to 100%. A delay of 0% indicates no delay during the report interval, while a delay of 100% represents a job that was found delayed at every sample during that period.

$$\text{Delay (\%)} = \frac{\text{\# Delay Samples}}{\text{\# Samples}} \times 100$$

Resource workflow (%)

The workflow of resources indicates how efficiently users are being served. The speed with which each resource performs the work of all users is expressed as a value from 0% to 100%. A low workflow value represents a large queue of work requests and a large number of delayed jobs, while a high workflow value represents little resource queueing contention and a small number of delayed jobs.

*PROC Workflow(%)

*PROC Workflow is a Monitor III metric that is very valuable in understanding the CPU behavior. In a sense, is more important than utilization itself. The formula is:

$$\text{*Proc\_Workflow} = \frac{\text{\#AS/enclave\_DU\_active}}{\text{(\#AS/enclave\_DU\_active + \#AS/enclave\_DU\_ready)}}$$

- #AS/enclave_DU_active is the average number of ASs and enclaves with dispatchable units active (executing on a logical CPU).
- #AS/enclave_DU_ready is the average number of ASs and enclaves with dispatchable units ready (being delayed by the lack of logical CPUs).

For example, a *PROC Workflow of 40%, discarding the focus on the CPU utilization, is a strong indication of contention. That means that only 40% of the ASs/enclaves that need CPU are getting CPU.

There is a similar metric for I/O, called *DEV Workflow(%).

These Monitor III fields are presented in the Monitor III WFEX report.

1.5 RMF Linux data gatherer

The RMF Linux data gatherer (rmfpms) is a modular data gatherer for Linux. You can analyze the gathered data using RMF PM or the RMF data on demand in a Web browser. rmfpms is also used by the IBM z/VM® Performance Toolkit. rmfpms is a short term data gatherer, similar to Monitor III. You can archive the gathered performance data for later analysis.

rmfpms provides performance metrics for the following performance areas:

- Process level
- Network level
- Apache HTTP server
- File level
- Direct Access Device (DASD) level
- CPU/Memory level

Apache HTTP server performance data is only available if Apache is actually running and properly configured on the monitored image. DASD performance data is only available on zSeries Linux images with Linux kernel 2.4 and activated DASD performance gathering.

You have to install and start rmfpms on each Linux image you want to monitor. After installing rmfpms, you can customize it by adapting default configuration files. The typical gathering cycle interval is 60 seconds, which we recommend that you do not change.

The Linux data gatherer is available via the RMF home page:

http://www.ibm.com/servers/eserver/zseries/zos/rmf/

2

RMF performance data front ends

Once you have collected performance data about your system, you need to display the information.

Use RMF to:

- View historical data when you look back at recent events.
- View real-time data to ensure that current system performance is good.
- Analyze a current performance problem.

To help you understand the RMF display options, this chapter discusses the following:

- Monitor II
- Monitor III
- Postprocessor
- Spreadsheet Reporter
- RMF Performance Monitoring
- RMF Web browser interface

And last, we describe available APIs you can use to get SMF records and Monitor II and Monitor III data from RMF to use with your own applications.

2.1 The RMF performance management menu

The RMF Performance Management menu offers easy access to the reporting capabilities of the Monitor II and Monitor III display sessions and the Postprocessor. Just enter the command:

```
RMF
```

The panel shown in Example 2-1 is returned.

Example 2-1 RMF Performance Management menu

```
                   RMF - Performance Management                 z/OS V1R5 RMF
Selection ===>

Enter selection number or command on selection line.

  1 Postprocessor   Postprocessor reports for Monitor I, II, and III      (PP)
  2 Monitor II      Snapshot reporting with Monitor II                    (M2)
  3 Monitor III     Interactive performance analysis with Monitor III     (M3)

  U USER            User-written applications (add your own ...)          (US)

  R RMFPP           Performance analysis with the Spreadsheet Reporter
  P RMF PM          RMF PM Java Edition
  N News            What's new in z/OS V1R5 RMF

                              T TUTORIAL    X EXIT

  RMF Home Page:    http://www.ibm.com/servers/eserver/zseries/zos/rmf/

         5694-A01 (C) Copyright IBM Corp. 1994, 2003. All Rights Reserved
                    Licensed Materials - Property of IBM
```

From this panel, you can access the RMF reporter session you want by entering on the selection line either:

- The selection number
- The abbreviation shown in parentheses to the right of your choice

Options R and P will not perform any online function because these RMF components are not available in a TSO session, but if you select either R or P, you get information about how to use these workstation functions.

By selecting U, you access previously-defined user-written applications.

Enter T to see a tutorial menu where you can select any RMF component to get more information.

Enter X to leave this panel without starting any reporter.

2.2 Monitor II or Monitor III

Two of the monitors in RMF, Monitor II and Monitor III, show us the existing data in our system. Why use one monitor instead of the other one? Let’s evaluate them.

Both monitors show you what is happening in the system:

- Monitor II shows you the current status of the system at the moment you type the request.
- Monitor III shows you the values captured in the previous minute.

The primary difference between Monitor II and Monitor III is that Monitor II does not report any history, and Monitor III does report historical data. In Monitor III, we can choose the reporting period we want to use to collect historical data.

We recommend Monitor III. It gives you more alternatives and much greater flexibility than Monitor II.

Still, Monitor II is useful for specific situations such as:

- You want to see actions in intervals shorter than 10 seconds, which is the minimum Monitor III reporting period.
- You want to get certain reports or data fields that are only provided in Monitor II:
  - Dispatching priorities of the address spaces (ASD)
  - Sysplex data server report (SDS)
  - Library lists report (linklst, lpalst, apflst)

But, overall, Monitor III is the reporter we recommend, because it can show you not only what is happening in your sysplex or systems right now, but also what happened previously.

2.3 Monitor II

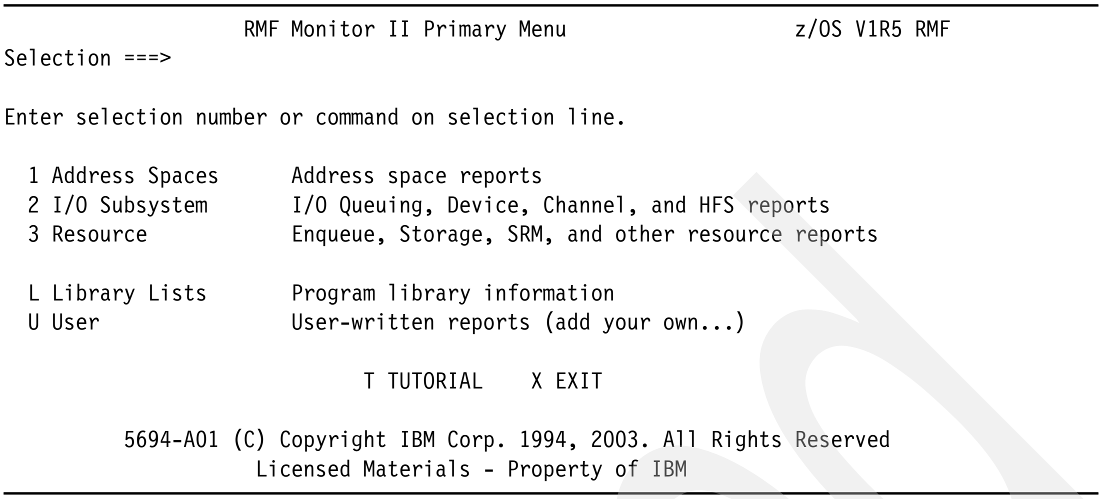
Monitor II tells you what is happening right now on your system, how system resources are used, and how your address spaces are doing. There are several standard reports provided, and you can add your own reports. You cannot see older or historical data. You can only see what is happening right now on your system, or current data.

You can collect data to SMF data sets continuously for Monitor II reports. In this case, you decide beforehand which reports you will produce by specifying them to the Monitor II data gatherer. Later, you can write the reports using the Postprocessor for the period you want to see. This is a useful method, for example, if you want to get information every third second about certain address spaces for one day or perhaps every day.

2.3.1 Using the Monitor II ISPF session

This session describes how to efficiently use the Monitor II ISPF interface that you start from the RMF Performance Management menu. It leads you to the Monitor II Primary Menu shown in Example 2-2 on page 20. (There is also a TSO/E display session for Monitor II available (RMFMON), but it is not covered in detail in this redbook.)

Example 2-2 Monitor II Primary menu

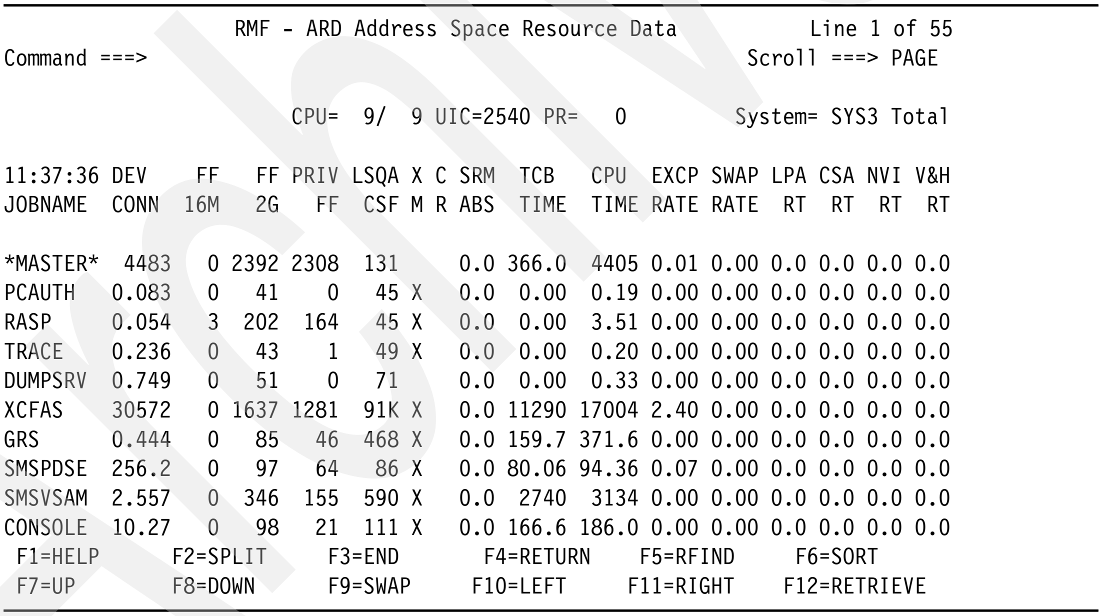


```
                    RMF Monitor II Primary Menu                   z/OS V1R5 RMF
Selection ===>

Enter selection number or command on selection line.

  1 Address Spaces      Address space reports
  2 I/O Subsystem       I/O Queuing, Device, Channel, and HFS reports
  3 Resource            Enqueue, Storage, SRM, and other resource reports

  L Library Lists       Program library information
  U User                User-written reports (add your own...)

                              T TUTORIAL    X EXIT

          5694-A01 (C) Copyright IBM Corp. 1994, 2003. All Rights Reserved
                     Licensed Materials - Property of IBM
```

You have several choices for reports from different reporting groups as you see in the panel.

You can select a report either by navigation through the panel, or directly by using the report command to invoke a report. For example, use the command **ARD** to navigate to the Address Space Resource Data (ARD) report. Example 2-3 shows this report.

Example 2-3 Monitor II ARD report



```
                   RMF - ARD Address Space Resource Data          Line 1 of 55
Command ===>                                                 Scroll ===> PAGE

                        CPU=  9/  9 UIC=2540 PR=   0         System= SYS3 Total

11:37:36 DEV    FF   FF PRIV LSQA X C SRM  TCB   CPU  EXCP SWAP LPA CSA NVI V&H
JOBNAME  CONN  16M   2G   FF  CSF M R ABS  TIME  TIME RATE RATE  RT  RT  RT  RT

*MASTER*  4483   0 2392 2308  131     0.0 366.0  4405 0.01 0.00 0.0 0.0 0.0 0.0
PCAUTH   0.083   0   41    0   45 X   0.0  0.00  0.19 0.00 0.00 0.0 0.0 0.0 0.0
RASP     0.054   3  202  164   45 X   0.0  0.00  3.51 0.00 0.00 0.0 0.0 0.0 0.0
TRACE    0.236   0   43    1   49 X   0.0  0.00  0.20 0.00 0.00 0.0 0.0 0.0 0.0
DUMPSRV  0.749   0   51    0   71     0.0  0.00  0.33 0.00 0.00 0.0 0.0 0.0 0.0
XCFAS    30572   0 1637 1281  91K X   0.0 11290 17004 2.40 0.00 0.0 0.0 0.0 0.0
GRS      0.444   0   85   46  468 X   0.0 159.7 371.6 0.00 0.00 0.0 0.0 0.0 0.0
SMSPDSE  256.2   0   97   64   86 X   0.0 80.06 94.36 0.07 0.00 0.0 0.0 0.0 0.0
SMSVSAM  2.557   0  346  155  590 X   0.0  2740  3134 0.00 0.00 0.0 0.0 0.0 0.0
CONSOLE  10.27   0   98   21  111 X   0.0 166.6 186.0 0.00 0.00 0.0 0.0 0.0 0.0
 F1=HELP     F2=SPLIT     F3=END       F4=RETURN    F5=RFIND     F6=SORT
 F7=UP       F8=DOWN      F9=SWAP     F10=LEFT     F11=RIGHT    F12=RETRIEVE
```

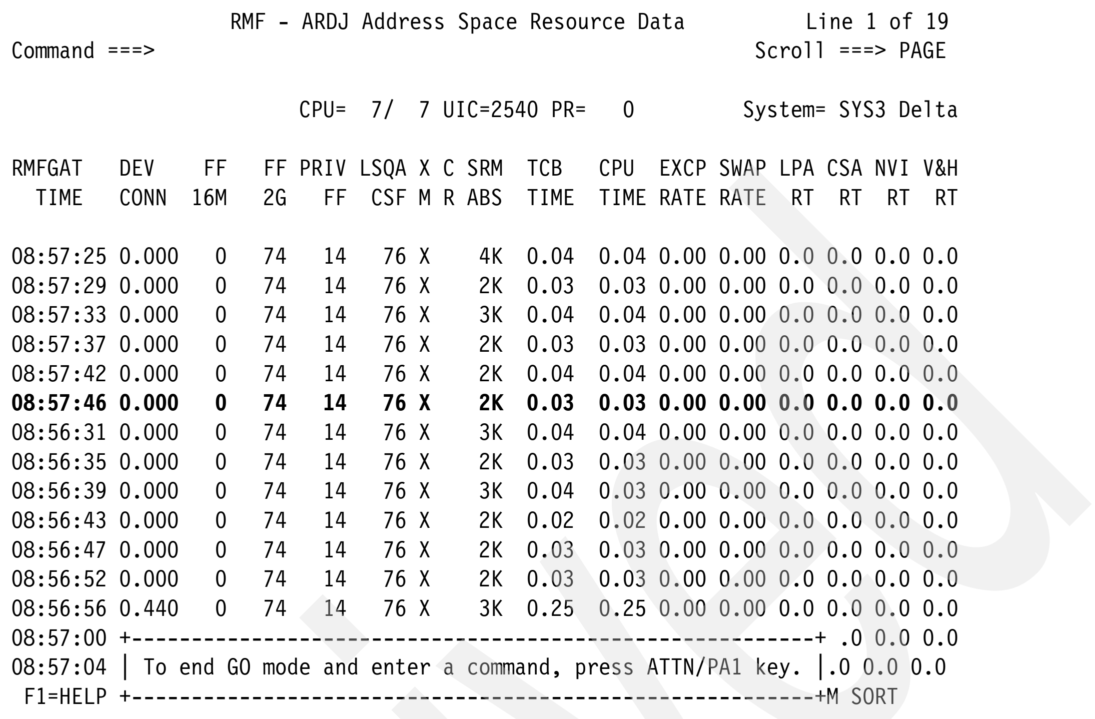
Each time you press Enter, the report is updated with the most current data.

Monitor II offers two types of reports:

Table reports Table reports have a variable number of lines of data such as Example 2-3.

Row reports A row report has only one line of data in the beginning. When you request the report repeatedly by pressing Enter, each request adds one line of data to the display; this is illustrated in Example 2-4 on page 21, which shows the Address Space Resource Data (ARDJ) report for a specific job. The report lines are displayed in wrap around mode.

Example 2-4 Monitor II ARDJ report



```
                  RMF - ARDJ Address Space Resource Data          Line 1 of 19
Command ===>                                                  Scroll ===> PAGE

                        CPU=  7/  7 UIC=2540 PR=   0         System= SYS3 Delta

RMFGAT   DEV    FF   FF PRIV LSQA X C SRM  TCB   CPU  EXCP SWAP LPA CSA NVI V&H
  TIME   CONN  16M   2G   FF  CSF M R ABS  TIME  TIME RATE RATE  RT  RT  RT  RT

08:57:25 0.000   0   74   14   76 X    4K  0.04  0.04 0.00 0.00 0.0 0.0 0.0 0.0
08:57:29 0.000   0   74   14   76 X    2K  0.03  0.03 0.00 0.00 0.0 0.0 0.0 0.0
08:57:33 0.000   0   74   14   76 X    3K  0.04  0.04 0.00 0.00 0.0 0.0 0.0 0.0
08:57:37 0.000   0   74   14   76 X    2K  0.03  0.03 0.00 0.00 0.0 0.0 0.0 0.0
08:57:42 0.000   0   74   14   76 X    2K  0.04  0.04 0.00 0.00 0.0 0.0 0.0 0.0
08:57:46 0.000   0   74   14   76 X    2K  0.03  0.03 0.00 0.00 0.0 0.0 0.0 0.0
08:56:31 0.000   0   74   14   76 X    3K  0.04  0.04 0.00 0.00 0.0 0.0 0.0 0.0
08:56:35 0.000   0   74   14   76 X    2K  0.03  0.03 0.00 0.00 0.0 0.0 0.0 0.0
08:56:39 0.000   0   74   14   76 X    3K  0.04  0.03 0.00 0.00 0.0 0.0 0.0 0.0
08:56:43 0.000   0   74   14   76 X    2K  0.02  0.02 0.00 0.00 0.0 0.0 0.0 0.0
08:56:47 0.000   0   74   14   76 X    2K  0.03  0.03 0.00 0.00 0.0 0.0 0.0 0.0
08:56:52 0.000   0   74   14   76 X    2K  0.03  0.03 0.00 0.00 0.0 0.0 0.0 0.0
08:56:56 0.440   0   74   14   76 X    3K  0.25  0.25 0.00 0.00 0.0 0.0 0.0 0.0
08:57:00 +---------------------------------------------------------+ .0 0.0 0.0
08:57:04 | To end GO mode and enter a command, press ATTN/PA1 key. |.0 0.0 0.0
 F1=HELP +---------------------------------------------------------+M SORT
```

Monitor II offers two modes for the session reports:

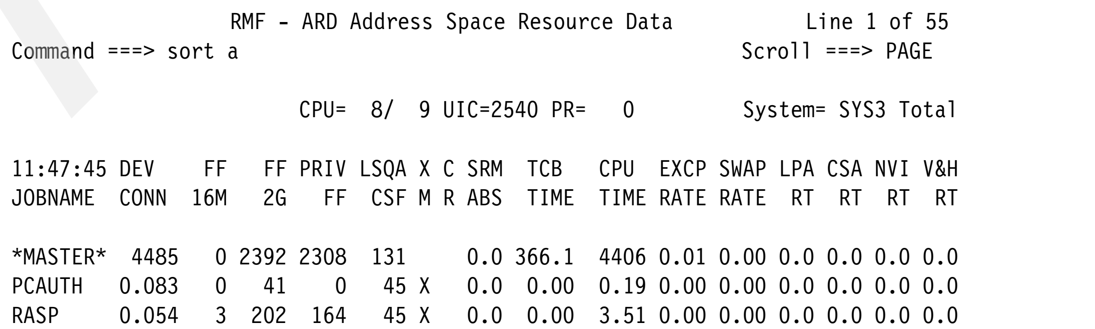
Total mode Depending on the measurement type, each report shows the cumulative total since either the beginning of the Monitor I interval or the creation time of the address space or IPL. To enable total mode use the command **`DELTA OFF`**. Example 2-3 on page 20 shows the ARD report in total mode.

Delta mode A delta mode report shows the incremental change in the activity since the previous version for the report. To enable delta mode use the command **`DELTA ON`**. Example 2-4 shows the ARDJ report in delta mode.

Sorting Monitor II reports

Monitor II offers a powerful sort option for table reports. To sort a table report, type the **`SORT`** command and place the cursor in the column you want to sort, then press Enter. You can use the default ascending sort (**`SORT A`**) or descending sort (**`SORT D`**) option. In Example 2-5, we want to sort the report according to the jobname, so we place the cursor in the `JOBNAME` column, for example on jobname `DUMPSRV`. When you press Enter, Monitor II sorts the lines of the report by the contents of the selected column. You can sort the report by any column. Instead of using the **`SORT`** command, you can use F6.

Example 2-5 How to use the sort feature of Monitor II



```
                  RMF - ARD Address Space Resource Data           Line 1 of 55
Command ===> sort a                                          Scroll ===> PAGE

                        CPU=  8/  9 UIC=2540 PR=   0         System= SYS3 Total

11:47:45 DEV    FF   FF PRIV LSQA X C SRM  TCB   CPU  EXCP SWAP LPA CSA NVI V&H
JOBNAME  CONN  16M   2G   FF  CSF M R ABS  TIME  TIME RATE RATE  RT  RT  RT  RT

*MASTER*  4485   0 2392 2308  131     0.0 366.1  4406 0.01 0.00 0.0 0.0 0.0 0.0
PCAUTH   0.083   0   41    0   45 X   0.0  0.00  0.19 0.00 0.00 0.0 0.0 0.0 0.0
RASP     0.054   3  202  164   45 X   0.0  0.00  3.51 0.00 0.00 0.0 0.0 0.0 0.0
```

```
TRACE    0.236  0   43    1   49 X   0.0  0.00  0.20 0.00 0.00 0.0 0.0 0.0 0.0
DUMPSRV  0.749  0   51    0   71     0.0  0.00  0.33 0.00 0.00 0.0 0.0 0.0 0.0
XCFAS    30580  0 1937 1581  91K X   0.0 11292 17008 2.56 0.00 0.0 0.0 0.0 0.0
GRS      0.444  0   85   46  468 X   0.0 159.8 371.8 0.00 0.00 0.0 0.0 0.0 0.0
```

Remote reporting with Monitor II

You can specify a certain system in a sysplex, and generate a report just for that system. You can specify the system you are using to run your Monitor II session, or another system. You can use the **`SYSTEM`** command, or you can overtype the value of the `System` field in the header of the report panel with the identifier of the desired system.

Finding a text string in Monitor II reports

To find a character string in the report, enter the command:

```
FIND textstring
```

If the string contains blanks, you have to enclose it in quotation marks. You can repeat a previous **`FIND`** command using the command **`RFIND`**.

Report options

You can call some of the Monitor II reports using report options. Example 2-6 shows the report options panel for the Monitor II ARD report. The report options panel allows you to specify additional report parameters, for example, to select a specific workload type that will display in the report.

Example 2-6 Report options of Monitor II ARD report

```
                      RMF Monitor II - Address Space Option
Command ===>

Change or verify parameters. The input entered on this panel applies to
ARD, ASD, and ASRM. To exit press END.

 Class     ===> T_        Specify one of the following workloads:
                          A=All, B=Batch/STC, T=TSO, AS=ASCH, O=OMVS
 Inactive  ===> NO_       Specify YES to include inactive address spaces.
```

To reset the report options to the defaults, enter the **`RESET`** command.

Refreshing a report automatically

To refresh a report automatically, on any report panel, enter the command:

```
GO n
```

n is a decimal integer, and 4 is the default. The report is refreshed automatically every ***n*** seconds. To stop the automatic refresh of the report, press the ATTN or PA1 key.

2.4 Monitor III

Monitor III tells you how well your single system or sysplex is performing, and what is going on. This is presented at different levels:

- Sysplex-wide reports about the workloads, Coupling Facilities, and caching
- System-wide reports about the resources and address spaces

You can see what is happening right now, typically during the last 60 seconds. You can also see what happened recently or you might be able to see what happened the day before yesterday depending to your installation setup. Additionally, you can dynamically change the time frame you want to observe. For example, your actions might be:

- Using 10-minute time frames on one day, travelling backward and forward, to find the most interesting 10-minute period.
- Using one-minute time frames, travelling backward and forward, to find the most interesting one minute-period.
- At that point, it should be easy to locate the system, partition, address space, device, or whatever it is that you want to examine.

Monitor III provides many powerful reports, and we take a closer look at them later in case studies. Here, we are assuming that we can find the point in time and the object that we are looking for, and that we can see the correct report to understand the situation.

Monitor III reports are created from data that is available either in the storage buffers belonging to currently running Monitor III gatherer sessions (this is the most current information), or to VSAM data sets that have been defined for Monitor III to keep the gathered data across gatherer sessions. The reporting interval that you can display with the reporter depends on the size of the buffers and data sets. Information about this topic is in 3.4.1, “Customization of the Monitor III gatherer” on page 124.

As described in 1.2.2, “Data gathering services with Monitor III” on page 9, the Monitor III gatherer writes some data to SMF data sets as well. You can use the Postprocessor for getting reports from this data (see 2.5.1, “Using the Postprocessor to create reports” on page 32).

2.4.1 Using the Monitor III ISPF session

This section describes how to efficiently use the Monitor III ISPF interface that you start from the RMF Performance Management menu. It leads you through the RMF Monitor III Primary Menu in Example 2-7.

Example 2-7 Monitor III Primary menu

```
                   RMF Monitor III Primary Menu                 z/OS V1R5 RMF
Selection ===>

Enter selection number or command on selection line.


  S SYSPLEX         Sysplex reports and Data Index                       (SP)
  1 OVERVIEW        WFEX, SYSINFO, and Detail reports                    (OV)
  2 JOBS            All information about job delays                    (JS)
  3 RESOURCE        Processor, Device, Enqueue, and Storage              (RS)
  4 SUBS            Subsystem information for HSM, JES, and XCF         (SUB)

  U USER            User-written reports (add your own ...)              (US)


                    O OPTIONS    T TUTORIAL    X EXIT

       5694-A01 (C) Copyright IBM Corp. 1986, 2003. All Rights Reserved
                     Licensed Materials - Property of IBM
```

You have several choices of types of reports from different reporting groups.

Example 2-8 shows the Sysplex Summary report. This report displays data from day 10/15/2004 of the sysplex UTCPLXJ8 and includes the time frame 14:14:00 to 14:15:00 because of the specified range of 60 seconds.

Example 2-8 Monitor III Sysplex summary report

```
                     RMF V1R5   Sysplex Summary - UTCPLXJ8       Line 1 of 29

 WLM Samples: 239     Systems: 14 Date: 10/15/04 Time: 14.14.00 Range: 60    Sec

                      >>>>>>>>XXXXXXXXXXXXXXXXXX<<<<<<<<

 Service Definition: WLMDEF01            Installed at: 10/01/04, 14.35.00
      Active Policy: WLMPOL01            Activated at: 10/01/04, 14.35.19

               ------- Goals versus Actuals --------  Trans --Avg. Resp. Time-
               Exec Vel  --- Response Time ---  Perf  Ended  WAIT EXECUT ACTUAL
 Name     T  I  Goal Act  ---Goal--- --Actual--  Indx  Rate   Time   Time   Time

 DISCR    S  D        59                              0.000
 WLMBTCHH S  2    50  56                        0.90  0.000
 CICS     S  2       N/A  0.600 80%        100%  0.50  9.233 0.000  0.032  0.039
 CICSCONV S  3       N/A  10.00 50%        100%  0.70  0.083 0.000  6.506  6.506
 CICSDEFA S  3       N/A  1.000 90%        100%  0.50  2.833 0.000  0.001  0.043
 CICSMISC S  3       N/A  1.000 90%        100%  0.50  1.500 0.000  0.001  0.001
 CICSRGN  S  2    60  67                        0.90  0.000
 FAST     S  2    50  70                        0.72  32.50 0.000  0.020  0.021
  F1=HELP      F2=SPLIT     F3=END       F4=RETURN    F5=RFIND     F6=TOGGLE
  F7=UP        F8=DOWN      F9=SWAP      F10=BREF     F11=FREF     F12=RETRIEVE
```

Summarizing intervals

The command:

```
BREF RANGE=nnnn
```

specifies the time range you want to summarize and present the sampled data for. Valid time range values are 0 to 9999 seconds. Instead of using the **RANGE** command, you can overtype the value of the `Range` field in the header of the report panel.

> **Important:** Ensure that the region size of the TSO session has enough memory when requesting large ranges of data.

Backward and forward referencing

Using the commands **BREF** (backward referencing, F10) and **FREF** (forward referencing, F11), you can do backward and forward referencing. You can also do this by overwriting the fields `Date` and `Time` on the report panels.

Depending on the parameters you specify, you can display data from either:

- The data gatherer's in-storage buffer on any or all of the systems in a sysplex.
- The data gatherer's data sets on any or all systems in the sysplex (this requires that you have defined VSAM data sets for the gatherer sessions).
- Preallocated data sets.

Displaying current range data

To display a report of current data for the length of the current range value, use the command **CURRENT**.

Report options

The report options panels enable you to change the options for individual RMF reports. This allows you to customize reports to select different jobs, resource names, and workflow exceptions to appear in the report displays. You can specify service classes, report classes, and workload groups. The available report options differ, depending on the specific report. To obtain the options panel for a report, specify the command **ROPTIONS** on the command line of the report you wish to change.

Example 2-9 shows the options for the Sysplex Summary report, and you see that you have various selection criteria to use for filtering specific information in the report.

Example 2-9 Report options for the Sysplex summary report

```
                       RMF Sysplex Summary Report Options         Line 1 of 26
Command ===>                                                  Scroll ===> CSR

Select (S), exclude (X) or fill-in groups for the SYSSUM report. Press END.
Selections made here also affect the System Information (SYSINFO) and
the Storage Delay Summary (STORS) report.

  Type          ===> ALL   Type of groups on the report (ALL W S SP R RP)
  Perf. Index   ===> 0.0   Minimum performance index value to include a group
  Importance    ===> 5     Maximum importance to include a group in the report
  Inactive      ===> NO    Show inactive WLM groups (YES NO) in the report

Sel Group    T       Sel Group    T       Sel Group    T       Sel Group    T
    ________ _           ________ _           ________ _           ________ _
    ________ _           ________ _           ________ _           ________ _
 S  *ALL                 BAT_WKL  W           DB_WKL   W           IRDSYSH  W
    OMVS_WKL W           ONL_WKL  W           STC_WKL  W           TSO_WKL  W
    SYSTEM   W           ASCHDEF  S           ASCHHI   S           ASCHLO   S
    BATHI    S           BATLO    S           BATMED   S           BATPIER  S
    CBDEF    S           CBHI     S           CBLO     S           CICSCONV S
    CICSDEF  S           CICSHI   S           CICSLO   S           DDFDEF   S
    DDFHI    S           DDFLO    S           IMSDEF   S           IMSHI    S
    IMSLO    S           IRDSYSH  S           IRDSYSHI S           LIMIT    S
    OMVS     S           OPSDEF   S           OPSHI    S           OPSLO    S
```

To reset the report options back to the defaults, enter the **RESET** command.

Refreshing a report automatically

To refresh a report automatically, enter on any report panel the command **GO**. In GO mode, the report is refreshed automatically every MINTIME. To leave the GO mode and enter the STOP mode, press the ATTN or PA1 key. The STOP mode is the default mode which enables you to navigate among different reports for the same time range by using cursor-sensitive control.

Using cursor-sensitive control

Cursor-sensitive control lets you place the cursor on a field in a tabular report, press Enter, and see another report containing any additional information about the same field. You can move from one RMF report to another without returning to the primary menu or entering specific commands.

Remote reporting with Monitor III

You can specify which individual system in a sysplex you want a report to refer to. You can refer to the system you are using to run your Monitor III session, or another system. You can use the **`BREF SYSTEM=systemname`** command, or you can overtype the value of the `System` field in the header of the report panel with the identifier of the desired system.

Finding a text string in Monitor III reports

To find a character string in the report, enter the command:

```
FIND textstring
```

If the string contains blanks, enclose it in quotation marks. You can repeat a previous **`FIND`** command using **`RFIND`**.

Sysplex considerations

You may have systems in your sysplex with different releases of RMF installed. To avoid problems when reporting Monitor III data, always use an RMF reporter version that is equal to or higher than the highest RMF gatherer version.

User-defined Monitor III reports

RMF provides user exits that enable you to tailor data collection and reporting to the needs of your installation.

You can:

- Add information to a standard Monitor III report
- Sort the information in a standard report in a different order
- Create new reports combining the data that Monitor III gathers in the way you need it

To help you with these steps, use the Monitor III report format definition utility. This utility consists of a series of ISPF panels that allow you to modify the ISPF tables.

To start the report format definition utility, enter the following command from either TSO/E ready mode or within ISPF:

```
RMF UTIL
```

> **Note:** Do not try to access the report format definition utility in split screen mode when you are in an active RMF Monitor III reporter session.

Information to help you customize the reports is contained in Chapter 7 of the *RMF Programmer's Guide*, SC33-7994. Each table shows the fields available in that table, and the name of the report module that creates that table. The macros that map the tables are *not* provided with RMF.

2.4.2 Generating WTO messages

RMF provides exception handling based on the Monitor III metrics. You can use the functionality offered by RMF to issue console messages to the operator, or RMF also enables you to call your own programs. Here we discuss the scenario to set up exceptions and generate Write to Operator (WTO) messages. In addition, you can continue using the WTO messages as triggers for your automation program.

The scenario

The Monitor III Batch reporter is used to provide exception-initiated monitoring for Monitor III metrics. Figure 2-1 shows the scenario.

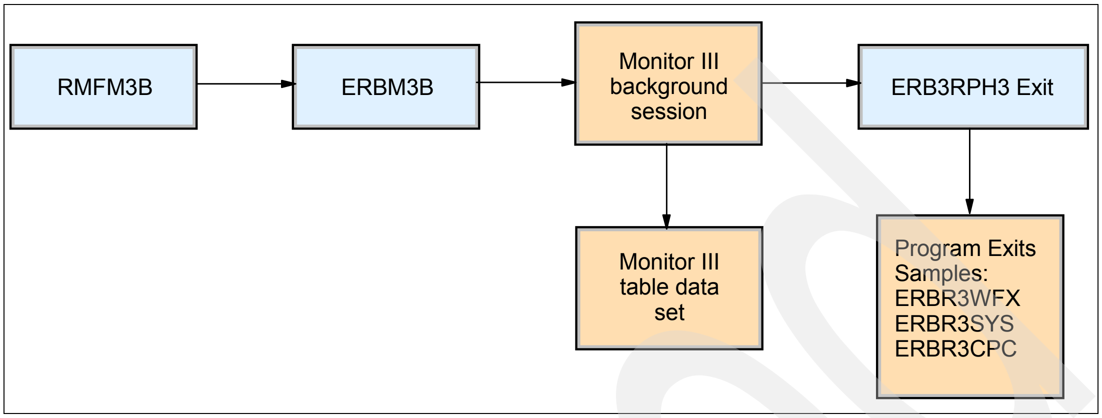

RMFM3B
ERBM3B
Monitor III background session
ERB3RPH3 Exit
Monitor III table data set
Program Exits Samples: ERBR3WFX ERBR3SYS ERBR3CPC

Figure 2-1 The scenario

The flow of actions is as follows:

1. The JCL RMFM3B in SYS1.PROCLIB invokes the REXX EXEC ERBM3B in SYS1.SERBCLS, which controls the Monitor III background session. RMFM3B passes the report parameter to ERBM3B that specifies the report that is used by the Monitor III background session for the exception reporting.
2. ERBM3B passes the control to the Monitor III background session.
3. The Monitor III background session calls the specified report in background mode. The Monitor III background session uses the Monitor III table data set that holds reporter session information, like WFEX options or Monitor III report range.
4. The generic Monitor III reporter phase 3 exit ERB3RPH3 in SYS1.SERBCLS calls the corresponding report exit handler for further processing of the Monitor III reporter data tables. For more information about the available Monitor III reporter tables, see the *RMF Programmer's Guide*.

RMF provides three sample program exits in the SYS1.SERBCLS:

- ERBR3WFX is the WFEX report processing exit handler. It checks for exceptions in the WFEX report, and issues any exceptions found as WTO messages.
- ERBR3SYS is the Monitor III Reporter SYSINFO Phase 3 Sample Exit. It processes the data tables when the SYSINFO report is requested.
- ERBR3CPC is the Monitor III Reporter CPC Phase 3 Sample Exit. It processes the data tables when the CPC report is requested, and issues a WTO message if necessary.

To allocate the Monitor III table data set that holds the report settings, RMF provides the REXX EXEC ERBM3BWX in SYS1.SERBCLS.

The setup

User “Raimo” wants to use the WTO messages to give a warning to the operator when RMF calculates that the system is capped in the next two hours or the four hour msu-value is close to the defined limit. The information is in the following fields of the Monitor III CPC report:

- The remaining time until capping is stored in the field CPCHRMSU.
- The four hour MSU-value is stored in the field CPCHLMSU.

- The defined limit is stored in the field CPCHIMSU.
- The percentage of WLM capping is stored in the field CPCHCAP.

Also, Raimo likes to check the status every five minutes.

User Raimo creates the Monitor III table data set using the REXX EXEC ERBM3BWX in SYS1.SERBCLS, shown in Example 2-10.

1. User Raimo executes the REXX EXEC. This procedure calls the Monitor III WFEX report.
2. Using command **SO**, user Raimo selects the RMF Session Options screen to change the reporting range by typing 300 seconds in the fields Refresh and Range.

- Raimo leaves the Monitor III session, pressing **F3** several times.

The ERBM3BWX procedure allocates a data set named &HLQ.RMFM3B.ISPTABLE, so for the user Raimo, the data set RAIMO.RMFM3B.ISPTABLE is created. The procedure RMFM3B already has an ISPTLIB definition for &HLQ.RMFM3B.ISPTABLE.

Example 2-10 TSO commands and Monitor III screen to create Monitor III table data set

```
exec 'sys1.serbcls(ERBM3BWX)'
RMF table data set 'RAIMO.RMFM3B.ISPTABLE' has been created
Initializing RMF Monitor III environment...
***
                           RMF Session Options

Current option set: WLMPOL   on SYS1
Change or verify parameters. Press END to save and end.

  Mode            ===> STOP       Initial mode (STOP GO)
  First Screen    ===> PRIMARY    Initial screen selection (ex: PRIMARY)
  Refresh         ===> 300        Refresh period (in seconds)
  Range           ===> 300S       Time range 10-9999 sec   (ex: 100S, 100)
                                             1-166 min     (ex: 2M)
...............
...............

Options have been saved to 'RAIMO.RMFM3B.ISPTABLE'
***
```

RMFM3B

The procedure RMFM3B in SYS1.PROCLIB runs the Monitor III reports in batch mode. It is required for exception-initiated monitoring. You need to modify the procedure, according to your environment, or you have to supply the needed input as startup parameters. User Raimo modifies the procedure as follows:

- The HLQ of the RMF libraries, for example for SYS1.SERBLINK, Raimo specifies RMF=SYS1
- The HLQ of the ISPF libraries, for example for ISP.SISPEXEC, Raimo specifies ISPF=ISP
- The report type, which is used for exception reporting, for the CPC report, Raimo specifies REPORT=CPC
- The HLQ for the Monitor III table data set (RAIMO.RMFM3B.ISPTABLE), Raimo specifies HLQ=RAIMO

User Raimo starts this procedure in several systems in the sysplex. Since the procedure allocates and uses a temporary ISPTABLE, a system identification definition is added to the isptable name in two places, as follows:

```
DSN=&HLQ..SYS&SYSCLONE..ERBPHDS3.ISPTABLE
```

Example 2-11 shows an extract of the procedure RMFM3B in SYS1.PROCLIB.

Example 2-11 Extract of RMFM3B

```
//RMFM3B PROC RMF=SYS1,ISPF=ISP,REPORT=CPC,HLQ=RAIMO
//*********************************************************************
//*                                                                   *
//* PROPRIETARY STATEMENT:                                            *
//*    LICENSED MATERIALS - PROPERTY OF IBM                           *
//*    "RESTRICTED MATERIALS OF IBM"                                  *
//*    5694-A01                                                       *
//*    (C) COPYRIGHT IBM CORP. 1996, 2001                             *
//*    STATUS: Z/OS V1R2 RMF (HRM7705)                                *
...
//*********************************************************************
//DELETE   EXEC PGM=IKJEFT01,
//         PARM='DELETE ''&HLQ..SYS&SYSCLONE..ERBPHDS3.ISPTABLE'''
//SYSTSPRT DD SYSOUT=A
//SYSTSIN  DD DUMMY
//SYSPRINT DD SYSOUT=A
//ALLOC    EXEC PGM=IEFBR14
//ERBPHDS3 DD DISP=(NEW,CATLG),
//            DSN=&HLQ..SYS&SYSCLONE..ERBPHDS3.ISPTABLE,
//            UNIT=SYSDA,
//            SPACE=(TRK,(1,1,1)),
//            DCB=(LRECL=80,RECFM=FB,BLKSIZE=3120)
//SYSPRINT DD SYSOUT=A
//ERBM3B   EXEC PGM=IKJEFT01,REGION=4M,DYNAMNBR=90,
//         PARM='ERBM3B &HLQ &REPORT'
//ISPPROF  DD DSN=&&TEMP1,UNIT=SYSDA,SPACE=(TRK,(2,1,2)),
//         DCB=(DSORG=PO,LRECL=80,BLKSIZE=3120,RECFM=FB)
//ISPLOG   DD DSN=&&TEMP2,UNIT=SYSDA,SPACE=(TRK,(5,5)),
//         DCB=(DSORG=PS,LRECL=125,BLKSIZE=129,RECFM=VA)
//SYSPROC  DD DISP=SHR,DSN=&HLQ..RMFM3B.SERBCLS
//         DD DISP=SHR,DSN=&RMF..SERBCLS
.....
```

ERBM3B

The procedure RMFM3B calls the REXX exec ERBM3B in SYS1.SERBCLS. Verify that you use an appropriate BDISPMAX parameter in the procedure ERBM3B. The BDISPMAX parameter helps to deal with possible looping situations. It specifies the maximum number of panel displays that can occur during a session. The default value is 100. If the number specified in BDISPMAX is exceeded, a severe error condition (return code 20) results and an error message, stating that the maximum number of displays is reached, is written to the SYSTSPRT data set. User Raimo changes the value as in Example 2-12.

Example 2-12 Extract of ERBM3B

```
/* REXX *************************************************************/
/*                                                                   */
/*01* MODULE-NAME: ERBM3B                                            */
/*                                                                   */
/*01* DESCRIPTIVE-NAME: RMF Monitor III background setup             */
...
```

```
Trace 0
Parse Upper Arg hlq report .

address "TSO" "PROFILE PROMPT MSGID WTPMSG PREFIX("hlq")"

cmd = "CMD(ERB3RP3I)"
"ISPSTART" cmd                      /* Sets up ERB3RPH3 as phase 3
                                       exit for all reports          */
/* pgm = "PGM(ERB3RCTL)"   Changed by Raimo for this new fix */
pgm = "PGM(ERB3RCTL) BDISPMAX(999999999)"                 /* @OA07834*/
....
```

The report exit

RMF provides a Monitor III Reporter CPC phase 3 sample exit. User Raimo creates his own data set, which the RMFM3B procedure requires as well, and copies the exit to that.

- Raimo creates the new data set RAIMO.RMFM3B.SERBCLS like SYS1.SERBCLS.
- Raimo copies the REXX EXEC ERBR3CPC into this new partitioned data set.

The procedure RMFM3B already has a SYSPROC definition for &HLQ.RMFM3B.SERBCLS.

Example 2-13 shows the modified ERB3CPC REXX EXEC.

Example 2-13 Exit ERBR3CPC

```
/* REXX **************************************************************/
/*                                                                   */
/*01* MODULE-NAME: ERBR3CPC */
/*                                                                   */
/*01* DESCRIPTIVE-NAME: CPC Report Handler                           */
/*                                                                   */
/*01* FUNCTION:                                                      */
...
...
/*********************************************************************/
Trace 0

Arg handler .

ADDRESS ISPEXEC
CONTROL ERRORS RETURN

CapValLimit = "00.0"               /* Set the threshold: capping=0  */
CapTimLimit = "03600"              /* Set the threshold: time=1h    */
MsuValRatio = 0.9                /* MSU threshold: Act4h/4hLim = 0.9 */

CapHdrTxt = "XWZ000I CPC: Local Partition Capping State:"
CapValMsg = "XWZ001I CPC: WLM Capping %:"
CapTimMsg = "XWZ002I Time until Capping (sec):"
MsuWrnMsg = "XWZ003I 4h MSU is close, Ratio:"
WtoLimTxt = "WTO Limit:"

                                   /* Obtain actual values          */
"VGET (cpchcap cpchrmsu cpchlmsu cpchimsu) SHARED"

/* This part just for testing Start */
WtoMsg1= "XWZ009I Testing... hlmsu="cpchlmsu "himsu="cpchimsu
WtoMsg = WtoMsg1 "hcap="cpchcap "hrmsu="cpchrmsu
"SELECT PGM(ERBCSWTO) PARM("WtoMsg")"
```

```
/* This part just for testing End */

hc = 0
/* Format and Write WTOs for Capping%                                */
IF SUBSTR(cpchcap,1,LENGTH(cpchcap)) > CapValLimit THEN DO
                                    /* Capping threshold exceeded ?  */
  WtoMsg = CapHdrTxt
  "SELECT PGM(ERBCSWTO) PARM("WtoMsg")"
  LimitValue = Strip(CapValLimit,"L",'0')
  IF LimitValue = ".0" THEN LimitValue = "0"
  Limit = "(" WtoLimTxt LimitValue ")"
  WtoMsg = CapValMsg cpchcap Limit
  "SELECT PGM(ERBCSWTO) PARM("WtoMsg")"
  hc = 1
END
ELSE DO
/* Format and Write WTOs for projected time until capping            */
  IF SUBSTR(cpchrmsu,1,LENGTH(cpchrmsu)) < CapTimLimit THEN DO
                                    /* Capping time limit reached ?  */
    WtoMsg = CapHdrTxt
    "SELECT PGM(ERBCSWTO) PARM("WtoMsg")"
    LimitValue = Strip(CapTimLimit,"L",'0')
    IF LimitValue = ".0" THEN LimitValue = "0"
    Limit = "(" WtoLimTxt LimitValue ")"
    Wtomsg = CapTimMsg cpchrmsu Limit
    "SELECT PGM(ERBCSWTO) PARM("WtoMsg")"
    hc = 1
  END

  ELSE DO
  /* Format and Write WTOs for close to 4h msu limit                 */
    MsuActRatio = 0
    IF cpchimsu > 0 THEN MsuActRatio = cpchlmsu / cpchimsu
    IF MsuActRatio > MsuValRatio THEN DO
                                    /* Msu 4h ratio reached ? */
      MsuActRatio3 = FORMAT(MsuActRatio,5,3)
      WtoMsg = CapHdrTxt
      "SELECT PGM(ERBCSWTO) PARM("WtoMsg")"
      LimitValue = Strip(MsuValRatio,"L",'0')
      IF LimitValue = ".0" THEN LimitValue = "0"
      Limit = "(" WtoLimTxt LimitValue ")"
      Wtomsg = MsuWrnMsg MsuActRatio3 Limit
      "SELECT PGM(ERBCSWTO) PARM("WtoMsg")"
      hc = 1
    END
  END
END

rc = hc
Exit rc
```

Start WTO generation

To start the RMF WTO exception messages, start the procedure RMFM3B in SYS1.PROCLIB with the console command **`S RMFM3B,HLQ=RAIMO`**.

Example 2-14 shows an extract of the log, where the WTO messages inform the operator that:

- Message XWZ009I informs the operator that this is a test version.
- Message XWZ000I is the heading message.
- The first extract message XWZ003I gives the warning that the 4 hour MSU value is close to the image capacity value, more than 90% of the capacity.
- The second extract message XWZ002I gives the warning that based on the estimate, capping begins in less than 3600 seconds, which is the limit value.
- The third extract message XWZ001I just informs the operator that capping is on.

Example 2-14   RMF WTO exception messages

```
19.57.02 STC24229  +XWZ009I Testing... hlmsu=   34 himsu=   37 hcap= 0.0 hrmsu=14400
19.57.02 STC24229  +XWZ000I CPC: Local Partition Capping State:
19.57.02 STC24229  +XWZ003I 4h MSU is close, Ratio:     0.919 ( WTO Limit: .9 )
20.02.02 STC24229  +XWZ009I Testing... hlmsu=   34 himsu=   37 hcap= 0.0 hrmsu= 8700
20.02.02 STC24229  +XWZ000I CPC: Local Partition Capping State:
20.02.02 STC24229  +XWZ003I 4h MSU is close, Ratio:     0.919 ( WTO Limit: .9 )

20.42.02 STC24229  +XWZ009I Testing... hlmsu=   36 himsu=   37 hcap= 0.0 hrmsu= 3000
20.42.02 STC24229  +XWZ000I CPC: Local Partition Capping State:
20.42.02 STC24229  +XWZ002I Time until Capping (sec):  3000 ( WTO Limit: 3600 )
20.47.02 STC24229  +XWZ009I Testing... hlmsu=   36 himsu=   37 hcap= 0.0 hrmsu= 2400
20.47.02 STC24229  +XWZ000I CPC: Local Partition Capping State:
20.47.02 STC24229  +XWZ002I Time until Capping (sec):  2400 ( WTO Limit: 3600 )

21.42.03 STC24229  +XWZ009I Testing... hlmsu=   37 himsu=   37 hcap=40.0 hrmsu=    0
21.42.03 STC24229  +XWZ000I CPC: Local Partition Capping State:
21.42.03 STC24229  +XWZ001I CPC: WLM Capping %: 40.0 ( WTO Limit: 0 )
21.47.02 STC24229  +XWZ009I Testing... hlmsu=   38 himsu=   37 hcap=41.4 hrmsu=    0
21.47.02 STC24229  +XWZ000I CPC: Local Partition Capping State:
21.47.02 STC24229  +XWZ001I CPC: WLM Capping %: 41.4 ( WTO Limit: 0 )
```

This sample scenario just sends operator commands, but it can also be used by the WTOs to trigger automated tasks such as using IBM Tivoli® System Automation for z/OS.

2.5 Postprocessor

There are two ways to create Postprocessor reports. You can either call the Postprocessor from the RMF Performance Management menu (see Example 2-1 on page 18), or you can use your own JCL and run it as a batch job. If you need Postprocessor data for the Spreadsheet Reporter, you can run the Postprocessor there (see 2.6, “Spreadsheet Reporter” on page 40). In this section, we cover the details you need to know to run your own JCL.

2.5.1 Using the Postprocessor to create reports

The Postprocessor writes reports from the data that is collected and saved to the SMF data sets by:

- Monitor I, SMF record types 70-78
- Monitor II, SMF record type 79
- Monitor III, SMF record type 74
- HTTP Server, SMF record type 103
- Lotus® Domino®, SMF record type 108

This data is written, typically, at 15-minute or 30-minute intervals. The data contains very detailed information about the sysplex, systems, CPCs, and workloads.

Table 2-1 provides an overview about the record types, gatherers and available reports.

Table 2-1 Activities that RMF monitors, and SMF record types

SMF records	Gatherer	Activity
70.1,79.3	Monitor I, II	License Manager, Processor
70.2	Monitor I	Cryptographic hardware
71,79.4	Monitor I, II	Paging
72.2/4,79.3	Monitor I,II	Storage
72.3	Monitor I	Workload activity
73,79.12	Monitor I,II	Channel path
74.1,79.9	Monitor I,II	Device
74.2	Monitor III	XCF
74.3/6	Monitor III	Unix
74.4	Monitor III	Coupling Facility
74.5	Monitor I	Cache
74.7	Monitor I	FICON Director
74.8	Monitor I	Enterprise Storage Server®
75,79.11	Monitor I,II	Page Data set
76	Monitor I	System Counters
77,79.7	Monitor I,II	Enqueue
78.2	Monitor I	Virtual Storage
78.3,79.13/14	Monitor I,II	I/O Queuing
79.1/2/5	Monitor II	Address space
79.6	Monitor II	Reserve
79.8	Monitor II	Transactions
79.15	Monitor II	IRLM long locks
103	non-RMF	HTTP Server
108	non-RMF	Lotus Domino Server

Extracting SMF dump data

You can use the Postprocessor to combine data from one, several, or all of the systems in the sysplex in one report. There are two prerequisites for this:

- Gather the data on all systems.
- Synchronize the gatherers on all systems.

Because SMF produces VSAM data sets and the Postprocessor cannot process VSAM data sets, you must copy SMF records into non-VSAM data sets. Do this by using the IFASMFDP program; for more information see *z/OS MVS System Management Facilities (SMF),* SA22-7630. Using other utilities to copy SMF records often results in truncated or unusable records. You can also use the program to select either specific SMF record types or a specific time range.

The job in Example 2-15 calls the utility IFASMFDP.

- Extract SMF data from SMF data set SYS1.SMFDATA.DUMP1 into SYS1.RMF.SMFDATA.DS01
- Select SMF record types 70-79,103,and 108

Example 2-15 Job that extracts SMF records

```
//BMAIDUMP JOB (DE03141),NOTIFY=BMAI,
//         MSGCLASS=H,MSGLEVEL=(1,1),CLASS=A,REGION=0M
//DUMP     EXEC PGM=IFASMFDP
//SYSPRINT DD SYSOUT=*
//IN       DD DISP=(SHR),DSN=SYS1.SMFDATA.DUMP1
//OUT      DD DISP=(NEW,CATLG),UNIT=SYSDA,SPACE=(TRK,(200,200),RLSE),
//         DSN=SYS1.RMF.SMFDATA.DS01
//SYSIN    DD *
   INDD(IN,OPTIONS(DUMP))
   OUTDD(OUT,TYPE(70:79,103,108))
/*
```

You can specify additional parameters for IFASMFDP, for example:

- To use time range Oct 25, 2004 - Oct 29, 2004

  ```
  DATE(2004299,2004303)
  ```

- To concentrate on main shift from 8:00 - 16:00

  ```
  START(0800)
  END(1600)
  ```

You can also use IFASMFDP to extract data from SMF dump data sets before you start to sort them, or use them as input for the Postprocessor.

Sorting SMF records

RMF needs sorted SMF records by RMF interval start date and interval start time in the data set. If you want to combine SMF records from several data sets, you must merge and sort the records to ensure that you will get correct reports. RMF provides two SORT exits (ERBPPE15 and ERBPPE35) that you should use when running the SORT program. You must use either both exits or neither of them. Using only one of them might render the SMF records unusable. The reason is because ERBPPE15 exit interchange some fields, basically RMF interval start time and date RMDAT/RMFIST with the SMF move-to-buffer time and date fields (SMFDTE/SMFTME) and ERBPPE35 will restore them back.

There is also some preprocessing, that is necessary for SMF 103 and 108 records http and domino) so that they can be processed by RMF Postprocessor. Sorting is necessary for sysplex reporting. RMF Postprocessor may handle single system record input sorted without the exits with the exception of when http or domino reports are involved (type 103, 108 where additionial information are added during the process.

Use the sample job supplied with RMF in SYS1.SAMPLIB(ERBSAMPP) for sorting Postprocessor input, as shown in Example 2-16. Depending on the size of the data sets, you may have to adapt the space parameters.

Example 2-16 Sort job

```
//BMAISORT  JOB (3141,RZ-43),'BMAI',NOTIFY=BMAI,MSGLEVEL=(1,1),
//             CLASS=A,MSGCLASS=Q,REGION=0M
//RMFSORT  EXEC PGM=SORT
//SORTIN   DD DISP=SHR,DSN=SYS1.RMF.SMFDATA.SYS1.DS01
//         DD DISP=SHR,DSN=SYS1.RMF.SMFDATA.SYS1.DS02
//         DD DISP=SHR,DSN=SYS1.RMF.SMFDATA.SYS1.DS03
//         DD DISP=SHR,DSN=SYS1.RMF.SMFDATA.SYS2.DS01
//         DD DISP=SHR,DSN=SYS1.RMF.SMFDATA.SYS2.DS02
//         DD DISP=SHR,DSN=SYS1.RMF.SMFDATA.SYS2.DS03
//SORTOUT  DD  DISP=(NEW,CATLG),SPACE=(TRK,(500,100),RLSE),
//         UNIT=SYSDA,DSN=SYS1.RMF.SMFDATA.SORTED
//SORTWK01 DD  DISP=(NEW,DELETE),SPACE=(CYL,(15,5),RLSE),UNIT=SYSDA
//SORTWK02 DD  DISP=(NEW,DELETE),SPACE=(CYL,(15,5),RLSE),UNIT=SYSDA
//SORTWK03 DD  DISP=(NEW,DELETE),SPACE=(CYL,(15,5),RLSE),UNIT=SYSDA
//SYSPRINT DD  SYSOUT=*
//SYSOUT   DD  SYSOUT=*
//SYSIN    DD  *
    SORT FIELDS=(11,4,CH,A,7,4,CH,A),EQUALS
    MODS E15=(ERBPPE15,36000,,N),E35=(ERBPPE35,3000,,N)
```

Postprocessor reports

RMF offers several types of reports:

- Interval and duration reports

  Here, you use the options REPORTS and SYSRPTS to get single system and sysplex reports. With the additional option DINTV, you create duration reports combining data from several measurement intervals into one report interval.

- Overview report/record

  The OVW option offers you the capability of tailoring summary-like reports according to your requirements. You can create your own single system and sysplex reports that show exactly the information you need for your performance management tasks. In the same way that you can create Overview reports, you can also create Overview records, just by specifying an additional option called OVERVIEW(RECORD).

  You can manually download the Overview records to the workstation for further processing in spreadsheets, or you can use the Spreadsheet Reporter directly to create and submit Postprocessor jobs without logging on to the host system explicitly.

If you have a sysplex with several partitions running in different time zones, you have to adjust the time stamps in all records. RMF provides a utility function for this task (see “Changing the time stamp on SMF RMF records to another time zone” on page 38).

You may have systems in your sysplex with different releases of RMF installed. To avoid problems when creating Postprocessor reports, always use a Postprocessor version that is equal to or higher than the highest Monitor I gatherer version. To ensure that your Postprocessor job is executed on the correct system, you may have to use the SYSAFF option in your job. Example 2-17 shows an extract of a Postprocessor job, where the SYSAFF parameter ensures that the job is executed on system *SC69*.

Example 2-17 Job with SYSAFF option

```
//BMAIPPX JOB (3141,RZ-43),'BMAI',NOTIFY=BMAI,MSGLEVEL=(1,1),
//             CLASS=A,MSGCLASS=Q,REGION=0M
/*JOBPARM SYSAFF=SC69
...
```

The amount of storage needed by a job is specified by the REGION parameter of the job statement. Make sure you either use a large enough region size or use REGION=0M.

Note: The installation exits IEALIMIT or IEFUSI might override the REGION parameter in your job.

You can create Postprocessor reports from either:

- SMF dump data sets: Specify the MFPINPUT DD statement.
- SMF buffer: Omit the MFPINPUT statement. Ensure that you have access to the SMF buffer.

The Postprocessor JCL in Example 2-18 generates reports with the following characteristics:

- Single system reports: CPU Activity report and Summary Interval report
- Sysplex report: Workload Activity report for service classes and periods
- From Oct 25, 2004 - Oct 29, 2004
- Main shift from 8:00 - 16:00
- From SMF dump data set SYS1.RMF.SMFDATA.DS07

Example 2-18   Postprocessor job to create RMF reports

```
//BMAIPPX  JOB (DE03141),NOTIFY=BMAI,
//         MSGCLASS=H,MSGLEVEL=(1,1),CLASS=A,REGION=0M
//RMFPOST  EXEC PGM=ERBRMFPP
//SYSPRINT DD SYSOUT=*
//MFPMSGDS DD SYSOUT=*
//MFPINPUT DD DISP=(SHR),DSN=SYS1.RMF.SMFDATA.DS07
//SYSIN    DD *
  SYSOUT(H)
  SUMMARY(INT)                  /* SUMMARY INTERVAL REPORT */
  STOD(0800,1600)               /* SUMMARY REPORT FROM 08:00-16:00 */
  RTOD(0800,1600)               /* REPORT FROM 08:00-16:00 */
  DATE(2004299,2004303)       /* 25 OCT 2004 - 29 OCT 2004 */
  REPORTS(CPU)                  /* CPU REPORT */
  SYSRPTS(WLMGL(SCLASS,SCPER))  /* WLMGL REPORT FOR SCLASS/SCPER*/
/*
```

Because the data produced by Monitor I and the reports produced by the Postprocessor are so detailed, you will probably print the reports only when you need to look at the systems at a detailed level, and only for the time frame you are studying. In this example, using `SYSOUT(H)`, all reports are written to an output class that is typically available for TSO processing. Then you can route the reports to the printer if you really need them on paper.

Example 2-19 on page 37 is part of the Workload Activity report, which can provide data for all WLM resources in the sysplex. It shows miscellaneous data about the first period of service class TSO.

Example 2-19   Monitor I Workload activity report

```
                                               W O R K L O A D   A C T I V I T Y
                                                                                                                     PAGE  20
      z/OS V1R6               SYSPLEX UTCPLXJ8           DATE 10/14/2004           INTERVAL 30.00.004   MODE = GOAL
                              RPT VERSION V1R5 RMF       TIME 12.30.00

                                        POLICY ACTIVATION DATE/TIME 10/01/2004 14.35.19

REPORT BY: POLICY=WLMPOL01   WORKLOAD=TSO        SERVICE CLASS=TSO        RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=2
                                                 CRITICAL     =NONE

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--   ---SERVICE----   --SERVICE TIMES--   PAGE-IN RATES    ----STORAGE----
AVG     29.01   ACTUAL               1.054   SSCHRT  94.4   IOC      3017K   TCB         617.4   SINGLE     0.0   AVG     1943.52
MPL     29.00   EXECUTION            1.054   RESP     1.8   CPU     48424K   SRB          25.8   BLOCK      0.0   TOTAL   56355.0
ENDED   47460   QUEUED                   0   CONN     1.0   MSO     17420K   RCT          17.9   SHARED     0.0   CENTRAL 56355.0
END/S   26.37   R/S AFFINITY             0   DISC     0.2   SRB      2021K   IIT           2.1   HSP        0.0   EXPAND     0.00
#SWAPS  40073   INELIGIBLE               0   Q+PEND   0.6   TOT     70882K   HST           0.0   HSP MISS   0.0
EXCTD       0   CONVERSION               0   IOSQ     0.0   /SEC    39379    IFA           N/A   EXP SNGL   0.0   SHARED    26.24
AVG ENC  0.00   STD DEV              2.323                                   APPL% CP     36.8   EXP BLK    0.0
REM ENC  0.00                                               ABSRPTN  1358    APPL% IFACP   0.0   EXP SHR    0.0
MS ENC   0.00                                               TRX SERV 1357    APPL% IFA     N/A

GOAL: RESPONSE TIME 000.00.02.000 AVG

         RESPONSE TIME EX   PERF  AVG   --- USING% --- ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--    %
SYSTEM   HHH.MM.SS.TTT VEL% INDX ADRSP   CPU  IFA  I/O  TOT  CPU                               UNKN IDLE  USG  DLY  USG  DLY QUIE

J80      000.00.01.054 57.2  0.5 151.8   0.3  N/A  0.1  0.2  0.2                                2.8 96.7  0.0  0.0  0.0  0.0  0.0

                                       ----------RESPONSE TIME DISTRIBUTION----------
   ----TIME----    --NUMBER OF TRANSACTIONS--    -------PERCENT-------  0    10   20   30   40   50   60   70   80   90   100
   HH.MM.SS.TTT    CUM TOTAL        IN BUCKET    CUM TOTAL   IN BUCKET  |....|....|....|....|....|....|....|....|....|....|
<  00.00.01.000        36895            36895         77.7        77.7  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
<= 00.00.01.200        37447              552         78.9         1.2  >
<= 00.00.01.400        37709              262         79.5         0.6  >
<= 00.00.01.600        38330              621         80.8         1.3  >
<= 00.00.01.800        38566              236         81.3         0.5  >
<= 00.00.02.000        38782              216         81.7         0.5  >
<= 00.00.02.200        39288              506         82.8         1.1  >
<= 00.00.02.400        39457              169         83.1         0.4  >
<= 00.00.02.600        40033              576         84.4         1.2  >
<= 00.00.02.800        40149              116         84.6         0.2  >
<= 00.00.03.000        40296              147         84.9         0.3  >
<= 00.00.04.000        42102             1806         88.7         3.8  >>>
<= 00.00.08.000        46154             4052         97.2         8.5  >>>>>
>  00.00.08.000        47460             1306          100         2.8  >>
```

You can extend the capabilities of the standard Postprocessor reports by creating your own reports. These are called Overview reports, and you can select almost any of the fields available in the Postprocessor reports to use in your Overview reports.

You define each value you want for your report and records with an OVW control statement. A description of all available fields is in chapter 17 of the *RMF User's Guide*, SC33-7990.

The job in Example 2-20 generates an Overview report and Overview records:

- The OVERVIEW statement defines the type of output.
- The OVW statements define the fields to be reported.
- The records will be written to SYS1.RMF.OVWREC.

Example 2-20   Postprocessor JCL to create Overview report and Overview records

```
//BMAIPPX  JOB (DE03141),NOTIFY=BMAI,
//         MSGCLASS=H,MSGLEVEL=(1,1),CLASS=A,REGION=0M
//RMFPOST  EXEC PGM=ERBRMFPP
//SYSPRINT DD SYSOUT=*
//MFPMSGDS DD SYSOUT=*
//MFPINPUT DD DISP=(SHR),DSN=SYS1.RMF.SMFDATA.DS07
//PPOVWREC DD DISP=(NEW,CATLG),SPACE=(CYL,(2,2),RLSE),UNIT=SYSDA,
//         DCB=(RECFM=VB,LRECL=32756,BLKSIZE=32760),
//         DSN=SYS1.RMF.OVWREC
//SYSIN    DD *
   SYSOUT(H)
   ETOD(0800,1000)
   OVERVIEW(REPORT,RECORD)
   OVW(PHYCPUPR(PBUSYL(A04)))
   OVW(LOGCPUPR(LBUSYL(A04)))
   OVW(WGTHAVG(WACTL(A04)))
   OVW(HIBEXVEL(EXVEL(S.BATHI.1)))
   OVW(CICTXPS(TRANS(S.CICSHI.1)))
   OVW(CICTIME(RTIME(S.CICSHI.1)))
   NOSUMMARY                       /* NO SUMMARY REPORT/*
/*
```

The sample job creates the Overview report shown in Example 2-21. In this sample, the data is collected every 10 minutes. A line on the report is displayed for each 10-minute period.

The Overview records contain the same information, one record for each interval. You can download the records to your workstation for further processing with the Spreadsheet Reporter.

Example 2-21 Overview report

```
                                                     R M F   O V E R V I E W   R E P O R T

            z/OS  V1R5               SYSTEM ID SYS1            START 10/18/2004-08.00.00  INTERVAL
00.10.00
                                     RPT VERSION V1R5 RMF      END   10/18/2004-10.00.00  CYCLE 1.000
SECONDS

NUMBER OF INTERVALS 12            TOTAL LENGTH OF INTERVALS 02.00.00

DATE    TIME    INT    PHYCPUPR  LOGCPUPR   WGTHAVG  HIBEXVEL   CICTXPS   CICTIME
MM/DD HH.MM.SS MM.SS
10/18 08.00.00 09.59        3.0      53.6        10     70.5      8.63     2.294
10/18 08.10.00 09.59        5.2      93.2        10     36.1      8.39     2.228
10/18 08.20.00 10.00        5.2      94.0        10     40.7      8.77     2.283
10/18 08.30.00 09.59        3.6      64.9        10     52.6      8.78     2.257
10/18 08.40.00 10.00        3.1      55.1        10     59.0      8.72     2.294
10/18 08.50.00 10.00        4.6      83.1        10     49.6      8.63     2.315
10/18 09.00.00 10.00        5.5      98.6        10     34.6      8.77     2.284
10/18 09.10.00 10.00        5.5      99.7        10     46.2      8.64     2.308
10/18 09.20.00 10.00        5.5      99.6        10     43.1      8.63     2.322
10/18 09.30.00 09.59        5.5      99.6        10     46.4      8.75     2.284
10/18 09.40.00 10.00        5.5      99.6        10     43.9      8.73     2.291
10/18 09.50.00 10.00       10.2      96.8        10     45.4      8.83     2.268
```

> **Tip:** The Spreadsheet Reporter helps you create RMF Postprocessor reports or Overview records without having to log on to the host or write JCL on your own.

Changing the time stamp on SMF RMF records to another time zone

In your sysplex, if you have partitions running in different time zones, they have different GMT time offsets and different time stamps in their SMF records. Therefore, the Postprocessor cannot recognize that these records belong to the same interval when creating reports. You can solve this problem by running program ERBCHGMT. This program changes the GMT time offset and the RMF time stamp, and writes the changed records to a new data set.

We changed the time stamps on the RMF SMF records from GMT minus four hours to GMT plus one hour. We specified parm='+60', which means GMT plus one hour in minutes. This is shown in Example 2-22 on page 39.

Example 2-22   Utility to change time stamp in SMF RMF records

```
//CHGGMT   EXEC PGM=ERBCHGMT,PARM='+60'
//SMFDATA  DD   DISP=SHR,DSN=RAIMO.RMFSMF.BUFDAT
//SMFCHGMT DD   DISP=(NEW,CATLG),SPACE=(CYL,(100,10),RLSE),
//         UNIT=SYSDA,DCB=(RECFM=VBS,LRECL=32760,BLKSIZE=0),
//         DSN=RAIMO.RMFSMF.NEWTIME
//SYSPRINT DD   SYSOUT=*
```

The new data set now has GMT plus one hour. That means, for example, that the RMF time stamp 10:00:00 is changed to 15:00:00, from GMT minus four to GMT plus one. The SMF time stamp remains unchanged.

In SYS1.SAMPLIB, member ERBGMTPP has a sample job to update the records and run some reports using the Postprocessor.

Scanning SMF RMF records

Sometimes it is necessary to investigate the contents of the RMF records on the SMF data set. You can use the ERBSCAN utility to display RMF records directly under ISPF, where you can call it like a TSO command:

```
erbscan  ‘raimo.rmfsmf.bufdat’
```

The result should look like Example 2-23.

Example 2-23   ERBSCAN screen

```
1z/OS V1R5 RMF ERBMFSCN Version 7 (30 Apr 2003) - SCAN SMF dataset

 SMF dataset characteristics:
 RECFM    : VBS
 LRECL    : 32767
 BLKSIZE  : 4096
 DATASETS : 1
 DSNAME(S): RAIMO.RMFSMF.BUFDAT
 DATE/TIME: 2004 October 23     12:32:31.165

1Rec-Num Type    RecLn SMFDate  SMFTime  RMFDate  RMFTime  Int-Len   SMFId/Vers Samples SRCL End-Token        S-C Sysplex  S
 ------- ------- ----- ----------------- ----------------- --------- ---------- ------- ---- ---------------- --- -------- -
       1 070.002   932 2004.293 10:00:00 2004.293 09:50:00 09:59:999 SYS3 712-1       0 0068 BBFDD97D7D800000 S   SYSXPLEX #
       2 070.002   932 2004.293 10:00:00 2004.293 09:50:00 10:00:044 SYS1 715-1       0 0068 BBFDD97D7D800000 S   SYSXPLEX #
       3 070.001 11884 2004.293 10:10:00 2004.293 10:00:00 09:59:997 SYS3 712-1     600 0068 BBFDDBB9B1E00000 S   SYSXPLEX #
       4 +----------------------------------------------------------------------------------------------+ 000 S   SYSXPLEX #
       5 ! Enter 'ERBSHOW <recnum>' in the EDIT command line to analyze the specified record in detail. ! 000 S   SYSXPLEX #
       6 +----------------------------------------------------------------------------------------------+ 000 S   SYSXPLEX #
```

Then you use the second utility ERBSHOW to select the record you are interested in:

```
erbshow 63
```

Example 2-24 shows some lines at the beginning of the record number 63.

Example 2-24   ERBSHOW screen

```
Record Number 63: SMF Record Type 70(1) - RMF CPU Activity
==========================================================
-> SMF record header
====================
   SMF record length      : 11884
   SMF segment descriptor : '0000'X
   SMF system indicator   : '11011111'B
   SMF record type        : 70
   SMF record time        : 11:50:00
   .....
-> RMF header extension
```

```
=======================
   Number of triplets     : 7
   Section  1 offset      : '00000054'X
   Section  1 length      : '0068'X
   Section  1 number      : 1
   Section  2 offset      : '000000BC'X
   Section  2 length      : '0028'X
   .....
-> RMF Product Section (1)
==========================
   #1:   +0000:  712FD9D4 C6404040 40400114 000F0104  *É RMF           *
         +0010:  293F0959 955F0000 00000258 00001000  *   ßn¬      ì     *
         +0020:  40404040 0001000F E9E5F0F1 F0F4F0F0  *        ZV010400*
   .....
```

2.6 Spreadsheet Reporter

The Spreadsheet Reporter is the powerful workstation solution for graphical presentation of long term Postprocessor data. Use the Spreadsheet Reporter to convert your Postprocessor reports to spreadsheet format for further processing with spreadsheet macros. Spreadsheet macros generate representative charts for all performance-related areas. Also, they help you view and analyze performance data at a glance with guidance to drill down to performance problems.

Figure 2-2 displays the CPU utilization for system UIG1 for the selected time range of 10/02/2003 for selected workloads.

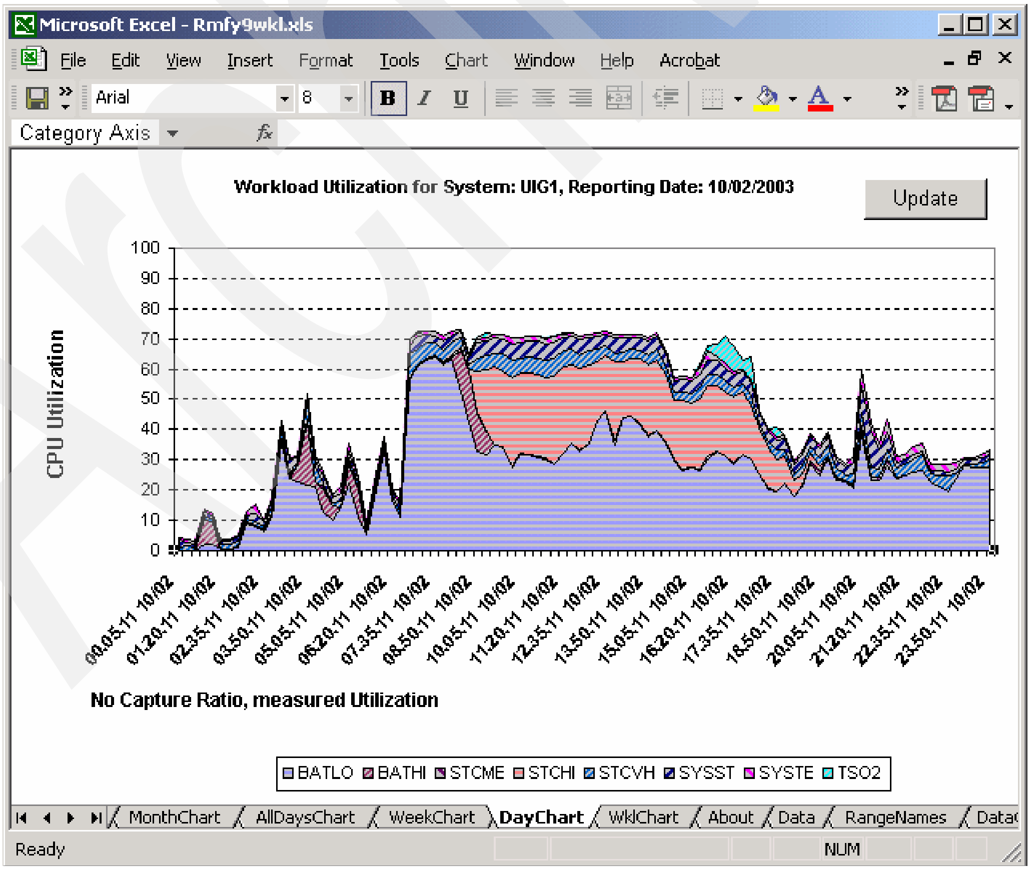


Figure 2-2 Workload Overview spreadsheet macro

The Spreadsheet Reporter offers the following features:

- Spreadsheet Reporter-related resource management with an Explorer-like GUI.
- Fast path to graphical presentation so you can prepare the SMF data in a single step.
- Batch mode enables you to generate input files for the spreadsheets without any GUI interaction, so that report generation is easily automated.

These features are available without having to log on to the host or write your own JCL.

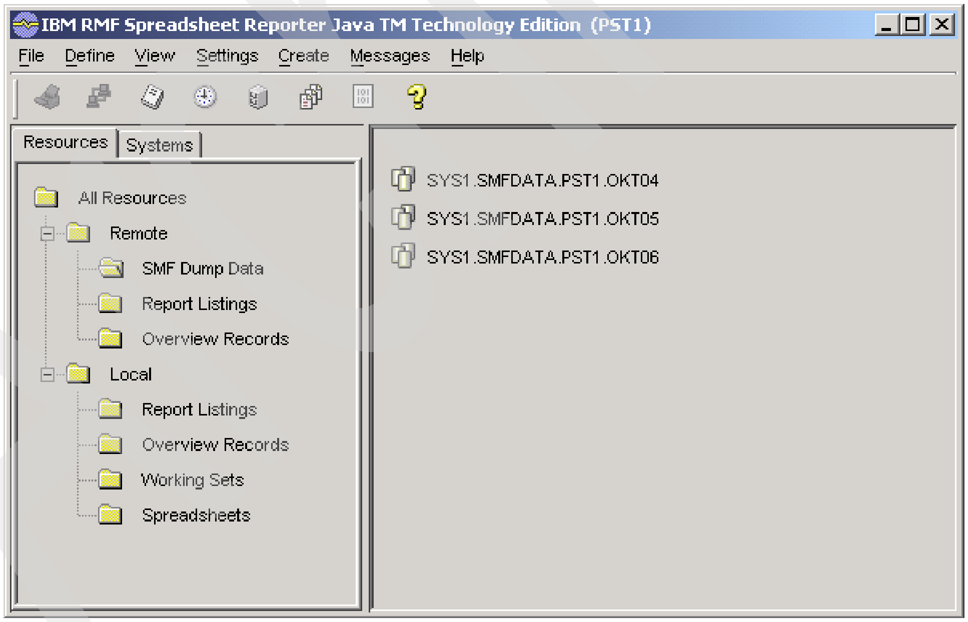
Performance data pulled from SMF records is the basis for z/OS performance analysis and capacity planning. You control the Postprocessor to extract performance measurements from SMF records to produce Postprocessor Report listings and Overview records. The Postprocessor output is downloaded and converted into a working set in spreadsheet format for further processing with the spreadsheet macros.

You can create a working set in one step.

Figure 2-3 shows the Explorer-like user interface of the Spreadsheet Reporter. This is the starting point to work with all available resources.

You can work with remote resources, such as specifying or selecting SMF data sets for creating Overview records or reports, or accessing remote report listings or Overview records to create working sets. You can also work with local resources, such as managing working sets, and accessing local report listings or Overview records to create working sets, and working with the spreadsheet macros.

Here in the Spreadsheet Reporter, you see the defined SMF Dump Data of the current selected system *PST1*. The selected system appears in the window title bar.



SYS1.SMFDATA.PST1.OKT04
SYS1.SMFDATA.PST1.OKT05
SYS1.SMFDATA.PST1.OKT06

Figure 2-3 Spreadsheet Reporter

2.6.1 Spreadsheet Reporter concept

For instructions to install the Spreadsheet Reporter, see “RMF Spreadsheet Reporter” on page 134.

To start the Spreadsheet Reporter, use Start → Programs → IBM RMF Performance Management → RMF Spreadsheet Reporter.

When you use the RMF Spreadsheet Reporter for the first time, you are asked to define the system from which you want to retrieve performance measurements. Click **Yes** to define your system now, as shown in “Defining the system” on page 48 or **No** to define the system later.

Spreadsheet Reporter main window

Figure 2-4 shows the Spreadsheet Reporter main window after you have started it.

The main window consists of two panes:

- **Navigation pane** (left side): Here you navigate to resources and systems that you want to manage.

  Clicking the Resources tab shows you all resource types. You can open a resource type folder containing the corresponding resources. Existing resources are then displayed on the view pane (right side). The resources are organized in a tree hierarchy containing remote and local resources.

  Clicking the Systems tab opens a folder called All Systems. All defined z/OS host systems are displayed in the view pane. If you want to access remote resources, you must first select the system where these resources reside. Otherwise, the list is empty.

- **View pane** (right side): Here you see available resources or systems.

  You can select resources or systems and initiate actions (for example, create a Working Set from a Report listing). Here you can view properties or modify (add/delete) resources or systems.

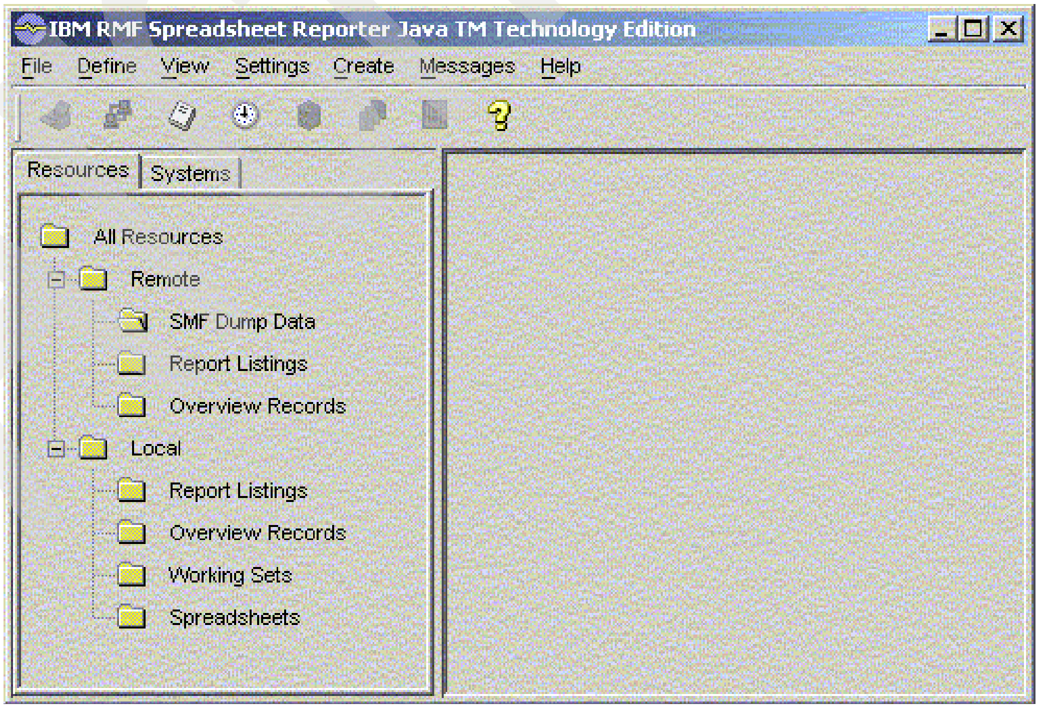


Figure 2-4 Spreadsheet Reporter main window

Spreadsheet Reporter resources

The Spreadsheet Reporter uses a resource-oriented concept. In the main window, the resources are grouped into remote and local resources.

Remote resources are located on the host system; local resources are located on your workstation.

Remote resources

- SMF dump data:

  The Postprocessor extracts the reports or metrics from SMF data to produce Report listings or Overview records. It can use SMF data from two sources:

  – SMF records from SMF dump data sets:

    SMF dump data is usually stored in generation data groups (GDGs). With the Spreadsheet Reporter, you can define these data sets as remote resources of type SMF dump data. For further details on SMF dump data sets, refer to “Extracting SMF dump data” on page 33.

  – SMF records from the SMF buffer:

    When you create Working Sets, Report listings, or Overview records, you determine which type of SMF data the Postprocessor should use. If you do not select any SMF dump data resource, then the Postprocessor automatically extracts the requested data from the SMF buffer.

The following two remote resources are results of Postprocessor batch jobs submitted on the host system, with SMF data specified as input.

Their default names consist of four parts. The first three are:

  – TSO high-level qualifier
  – Prefix D + Julian Day
  – Prefix T + time in HHMMSS format

- Report listings

  The listings are generated from report control statements that specify the reports that you want to examine. The forth qualifier of the name is LISTING.

  For example: IBMUSER.D203.T104615.LISTING

- Overview records

  Overview records are generated from overview control statements that specify the performance metrics that you want to examine. The Spreadsheet Reporter provides several macros that generate these statements for you, so you do not need to know about their syntax. The fourth qualifier of the name is OVWREC.

  For example: IBMUSER.D199.T131456.OVWREC

Local resources

Local resources are all the listings and records that were created on the host system and then transferred to the workstation for further processing. Working sets created on the workstation and spreadsheets are also included.

Their default names consist of four parts. The first three are:

  – System name
  – Prefix D + Julian Day
  – Prefix T + time in HHMMSS format

- Report listings

  The fourth qualifier of the name is lis.

  For example: SYSF.D203.T104615.lis

- Overview records

  The fourth qualifier of the name is rec.

  For example: SYSF.D199.T131456.rec

- Working sets

  Working sets are pulled from Report listings (Report Working Sets) or Overview records (Overview Working Sets), and they are used as input to a spreadsheet macro. When you create a working set directly from SMF dump data, you must know which spreadsheet macro you want to use with it. The reason you need to know ahead of time which spreadsheet macro you are going to use is that certain spreadsheet macros require a Report Working Set for input while other spreadsheet macros require an Overview Working Set. You also need to know the type of macro you are going to use for the correct selection of reports within the report options.

  There is no capability to create a working set derived from both Report listings and Overview records.

  The default names for working sets are comprised of the following parts:

  - Type indicator: (Rpt for a Report Working Set or Ovw for an Overview Working Set)
  - System name
  - Prefix D + Julian Day
  - Prefix T + time in HHMMSS format

  For example: Rpt.SYSF.D203.T103840 or Ovw.SCLM.D143.T144210

- Spreadsheets

  You use spreadsheet macros for the final presentation of the SMF data. Load and view your performance data using the spreadsheet macros into which you input the created working set.

  The Spreadsheet Reporter provides several sample spreadsheet macros to help you view and analyze performance data at a glance. Two examples of available spreadsheets are Workload Activity Trend Report and DASD Activity Report.

Spreadsheet Reporter workflow

With the Spreadsheet Reporter, you can convert Postprocessor output (Report listings or Overview records) to a data format that you can provide as input into spreadsheet macros for graphical presentation, using the following steps:

1. Define a host system to the Spreadsheet Reporter to enable the data transfer between the host and the workstation.
2. On your workstation, create a remote resource of type SMF dump data. This designates an SMF data set on the host.
3. From this resource, create a Working Set on your workstation. You can complete this step with a single action because the Spreadsheet Reporter automatically performs the complex data preparation tasks shown in Figure 2-5.
4. Select a Spreadsheet into which you input the created working set. The result is your desired graphical display of the performance data captured in the original SMF dump data.

You can also create a Working Set from local report listings or overview records by selecting them, and then clicking the **Create Working Set** icon.

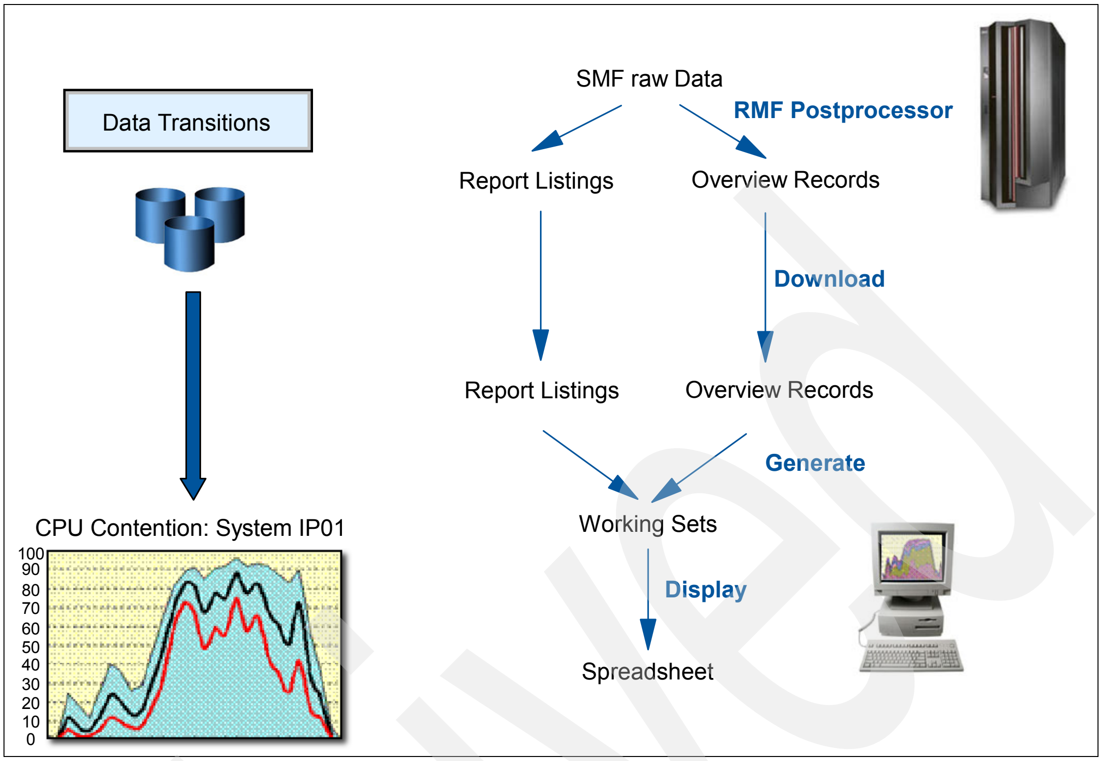


Figure 2-5 Spreadsheet Reporter concept

Spreadsheet macros

RMF provides a set of spreadsheet macros, where each macro represents a performance area. There are macros based on Report Working Sets and macros based on Overview Working Sets. You can create either a Report Working Set or an Overview Working Set, but not a working set that contains both types of data.

You do not need a separate working set for each macro. If the working set contains input from several reports or Overview records, it can be used for several macros. Table 2-2 and Table 2-3 on page 47 list the reports and records that are required for each macro.

How to approach: Start by finding the exact spreadsheet that presents the kind of data you want using the information provided in Table 2-2 and Table 2-3. You have to choose between a Report or an Overview spreadsheet, but this implies that you know exactly what options you want and may need to generate the correct statements for the Overview records.

One suggestion is that initially you spend some time on the spreadsheets with the sample data first to get an idea of what kind of performance data are reported in the spreadsheets.

Macros based on Report Working Sets

Table 2-2 lists the macros that require Report Working Sets as input.

Table 2-2 Spreadsheet macros based on Report Working Sets

Macro	Description	Based on RMF Report	Name
Summary Report	Processes a Summary report and creates analysis summaries and graphics from its data.	Summary report	Rmfn9sum
DASD Activity Report	Analyzes a DASD Activity report and provides summaries for the most frequently used LCUs and DASDs in the system.	DASD Activity report	Rmfr9das
Workload Activity Trend Report	Calculates performance reports and analyzes the system's behavior.	Workload Activity report	Rmfr9wlm
Coupling Facility Trend Report	Provides reports about activities in the coupling facilities.	Coupling Facility Activity report	Rmfr9cf
Cache Subsystem Report	Provides reports about activities in the cache subsystems.	Cache Subsystem Activity report	Rmfr9cac
I/O Subsystem Report	Analyzes DASD Activity reports from several systems and provides summaries for the most frequently used LCUs and DASDs in the sysplex.	Cache Subsystem Activity report, DASD Activity report	Rmfr9mdv
LPAR Trend Report	Analyzes Partition Data reports and provides information about the active partitions in the PR/SM™ environment.	Partition Data report (part of CPU Activity Report)	Rmfr9lp
Tape Mount Report	Displays the tape mounts and the tape activities for one or several systems.	Tape Activity report	Rmfr9tap
Open RMF Report Spreadsheets	Displays all macros you can use for working with a Report Working Set.	All RMF reports	Rmfr9opn
Filter DASD or Cache Reports	Filters devices from large DASD Activity and Cache Activity reports. Use this macro to focus on important, frequently used DASDs or cache subsystems. Use this macro to reduce the amount of data exceeding the spreadsheet format limit. You have to disable the Scratch Extracted RPT Files after Conversion option in the Settings → Options dialog on the General tab.	Cache Subsystem Activity Report, DASD Activity Report	Dasdconv

Macros based on Overview Working Sets

Table 2-3 lists the macros that require Overview Working Sets as input. To create an Overview Working Set, you have to use a specific set of Overview control statements to create the required data. You can use the macro Create Overview Control Statements for this purpose. The macro creates a file with the required overview control statements. The output file of this macro has to be specified as system property.

In addition, the following also offer the capability to generate control statements to create Overview records:

- DASD Activity Report
- Cache Subsystem Report

The advantage of using these macros is that, after you fed the macros with the DASD Activity or Cache Subsystem Report, they are aware of the available resources. For example, the

DASD Activity Report macro offers a list of available devices, where you can select which devices you want to create the required overview control statements. With the Create Overview Control Statement macro, you need to explicitly specify the devices.

Table 2-3 Spreadsheet Macros based on Overview Working Sets

Macro	Description	Based on	Name
LPAR Overview Report	Creates a long term overview about CPU consumption for selected partitions.	SMF 70 Data	Rmfx9cpc
System Overview Report	Creates a summary for one week, by a specified shift, for each hour and every day contained in the data. This allows you to examine data for one week at a glance. The macro also calculates the capture ratio.	SMF 70 Data SMF 71 Data SMF 72.3 Data	Rmfy9ovw
Workload Overview Report	Creates summaries and graphics for a set of selected service classes and workloads of your installation. The macro cannot process more than 27 workloads.	SMF 72.3 Data	Rmfy9wkl
Device Overview Report	Creates a trend report for selected devices of your installation. Note: The macro cannot process more than 30 devices.	SMF 74.1 Data	Rmfx9dev
Cache Subsystem Overview Report	Creates a trend report for selected cache subsystems of your installation. Note: The macro cannot process more than 18 control units or devices.	SMF 74.5 Data	Rmfx9cac
Channel Overview Report	Creates a channel report for selected channels of your installation.	SMF 73 Data	Rmfx9chn
Create Overview Control Statements	Creates OVW statements to generate data for the macros described previously for Overview Working Sets.	no SMF required	Rmx9mak

Attention: Ensure that the SMF dump data sets contain the appropriate SMF records data listed in Table 2-1 and Table 2-2.

Setting the security level for Microsoft® Excel macros

Within Microsoft Excel, you can specify the security level to enable or disable macro execution. To use the Microsoft Excel macros with Spreadsheet Reporter, you must enable these macros for execution.

Therefore, you have to set the security level to Medium or Low.

- Medium means that each time a macro is loaded, Microsoft Excel asks if you want to execute it or not.
- Low means that macros are automatically executed when loaded.

How to change the settings:

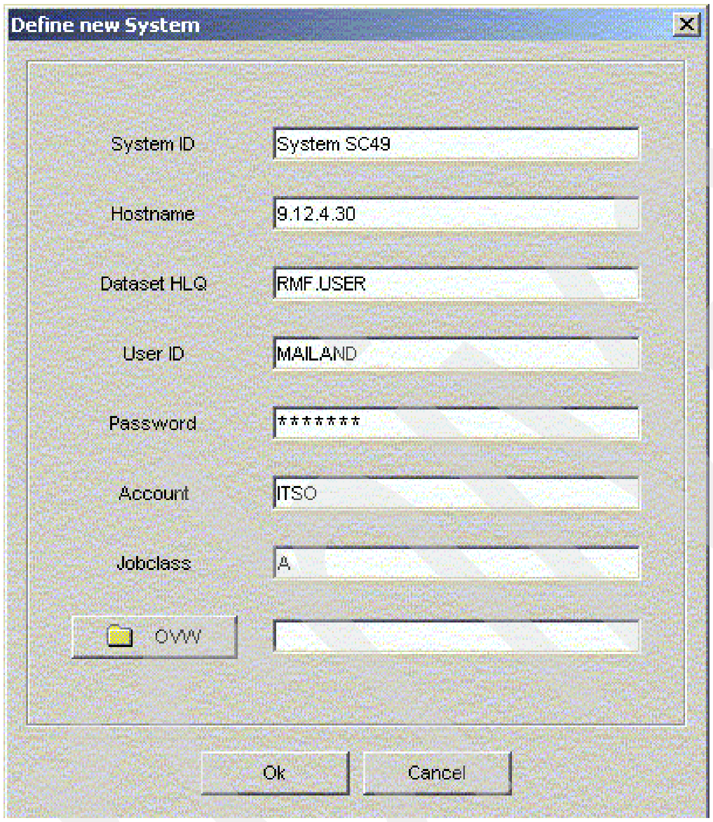
- Microsoft Excel 2000: The default security level is Medium. To change it from a macro's menu bar, use **Tools** → **Macro** → **Security**
- Microsoft Excel 2002: The default security level is High. To change it from a macro's menu bar, use **Tools** → **Options** → **Security** → **Macro Security**.

2.6.2 Getting started

In this section we describe how to work with the Spreadsheet Reporter.

Defining the system

First, we have to define the host system, where we want to execute the Postprocessor and where the SMF dump records are available. Clicking the Systems tab of the view pane, we see the defined systems and we can start defining our system: **Define** → **System** to start the Systems definition dialog, as shown in Figure 2-6.



Define new System

System ID: System SC49
Hostname: 9.12.4.30
Dataset HLQ: RMF.USER
User ID: MAILAND
Password: *******
Account: ITSO
Jobclass: A

Figure 2-6 System definition dialog

We now discuss several important fields.

System ID — Description of the system.

Hostname — The TCP/IP name of the system, either a hostname like XYZ.PROD.IBM.COM or an IP address like 9.164.182.251.
If you do not know your system's hostname, you can retrieve this hostname and the system's TCP/IP address with the TSO command **`hometest`**.

Dataset HLQ — The Spreadsheet Reporter uses remote Postprocessor job executions. During this process, several data sets are allocated. Therefore, the specified HLQ is used, which is usually the user ID. You can change the HLQ. The data sets need to be SMS-managed.

User ID — This is your TSO user ID, used for logon. Ensure that you have sufficient access to RMF Performance data, described in "Controlling access to RMF data for the sysplex data services" on page 116, and that you have access to the specified HLQ.

Account — The Spreadsheet Reporter requires this parameter for Postprocessor JCL generation. Specify the appropriate classification and identification information.

Jobclass — The Jobclass is required for Postprocessor JCL generation.

OVW — The file containing Overview control statements (see Example 2-27 on page 62).

> **Attention:** If you have specified a file containing overview control statements, either an overview report or an overview record is created. The report selection specified on the report options ("Options and Interval settings" on page 51) are then ignored and none of your selected reports are generated.

Also, if you have a file for Overview control statements in the appropriate blank, you cannot get Reports (as in the reports chosen on the Reports tab of the Options window) to run, even if in the Options you specify Reports like the Summary Report and no Overview Records.

Click OK to finish the definition. Now when you check the System tab, you see the defined system. You can change the system definition or the system name. To do so, right-click the system and get the context menu. Click either Rename to change the system name or Properties to get the system definition dialog to change the definitions, shown in Figure 2-7.

If you have selected a system, you see the system name in the title bar. To work with remote resource, you must first select a system.

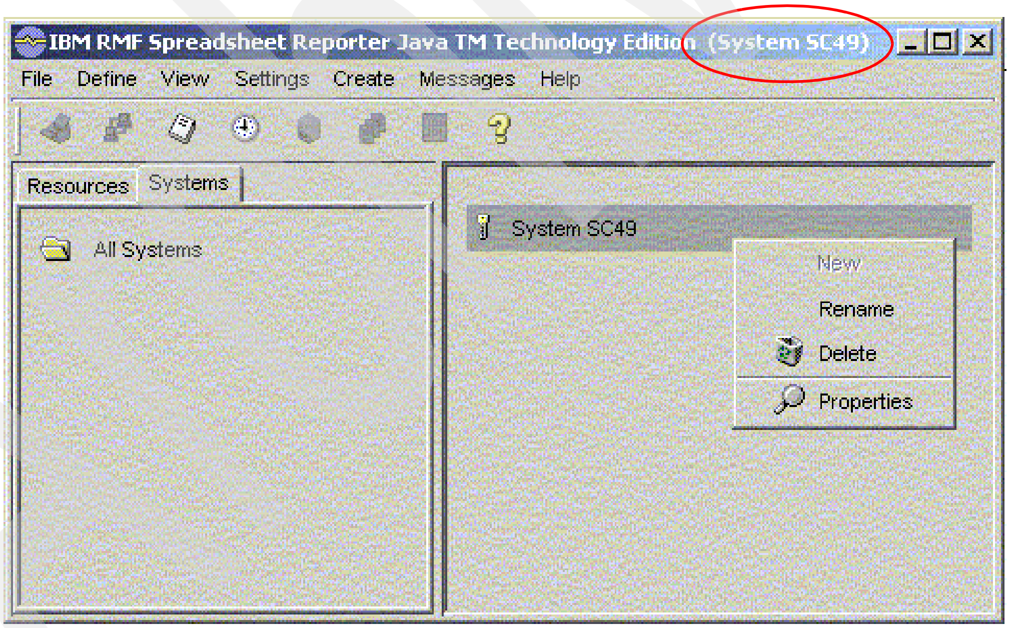


Figure 2-7 Defined system

Defining Postprocessor control statements

There are two ways to define control statements for the Postprocessor:

- Report listings for a Report Working Set

  Using **Settings** → **Options** or clicking the **Specify Options** icon, you get a list of reports where you can make your selection (see "Creating a Report Working Set" on page 51).

- Overview records for an Overview Working Set

  You have to create the appropriate control statements, either on your own, based on the description in the *RMF User's Guide*, or you can use one of the macros:

  – Create Overview Control Statements
  – DASD Activity Report
  – Cache Subsystem Report

  If you want to work with multiple sets of Overview control statements for the same system, you can define multiple copies of the same system with a different System ID, but the same Hostname. In this way, you can work with fixed attachments instead of changing the system properties all the time. Then you have to specify the file containing these statements in the Definition screen of your system.

  Note: You also have to ensure that the output format is Overview records and not reports (see Figure 2-8 on page 52).

JCL customization

Depending on your environment, you may need additional customization regarding the JCL you use or the FTP command sequence.

JCL skeleton

The Spreadsheet Reporter uses a JCL skeleton to create the Postprocessor job. This skeleton JCL RMFPP1.JCL is stored in the connect subdirectory of the Spreadsheet Reporter Program Files installation directory, for example:

C:\Program Files\RMF\RMF Spreadsheet Reporter\Connect\RMFPP1.JCL

Message class

The Postprocessor job uses the default message class H. The job output is later retrieved from the Spreadsheet Reporter. If it cannot retrieve the job output, the generation fails. Depending on your sysplex definition, you have to change the message class. In Example 2-25, we change the message class to T.

Sysplex with different releases

You may have systems in your sysplex with different release levels of RMF installed. To avoid problems when creating Postprocessor reports, always use the Postprocessor release level or version that is equal to or higher than the highest Monitor I gatherer version. To ensure that your Postprocessor job is executed on the correct system, you may have to use the SYSAFF option in your job. In Example 2-25, we added the SYSAFF parameter to ensure that the job is executed on system SC69.

Jobname

The Spreadsheet Reporter generates the jobname for the Postprocessor job by using the user ID and appending “$.” For example, for user MAILAND, the jobname is MAILAND$. You can change the jobname in the job skeleton. In Example 2-25, we change the variable jobname <USER>$ with RMFUSER.

Example 2-25 Extract of modified RMFPP1.JCL

```
//RMFUSER JOB (<ACCT>),<USER>,
//         CLASS=<CLASS>,NOTIFY=<USER>,MSGCLASS=T
/*JOBPARM SYSAFF=SC69
//******************************************************************
//*MAIN USER=<USER>
//******************************************************************
//*        RMF POSTPROCESSING
```

```
//******************************************************************
//*        DELETE MESSAGE AND LISTING DATASETS
//******************************************************************
//DELETE  EXEC PGM=IDCAMS
//SYSIN    DD   *
  DELETE (<PPDSN>) NONVSAM
  DELETE (<HLQ>.RMFPP.MSG) NONVSAM
  ...
```

Add your own FTP commands

After the logon via FTP to the host, the Spreadsheet Reporter executes FTP commands on the host. A subset of this FTP command sequence is stored in the control file:

C:\Program Files\RMF\RMF Spreadsheet Reporter\Work\ftpcmd.txt

Depending on your environment, you may have the requirement to issue a specific FTP command after the connection is established to the host. Therefore, just append your FTP command in the control file, and it is issued after the login. Example 2-26 shows the default control file ftpcmd.txt.

Example 2-26   ftpcmd.txt

```
************************************************************
*       RMF Spreadsheet Reporter Command Extensions       *
*  The command sequence will be executed during FTP login  *
************************************************************
site retpd
site autorecall
```

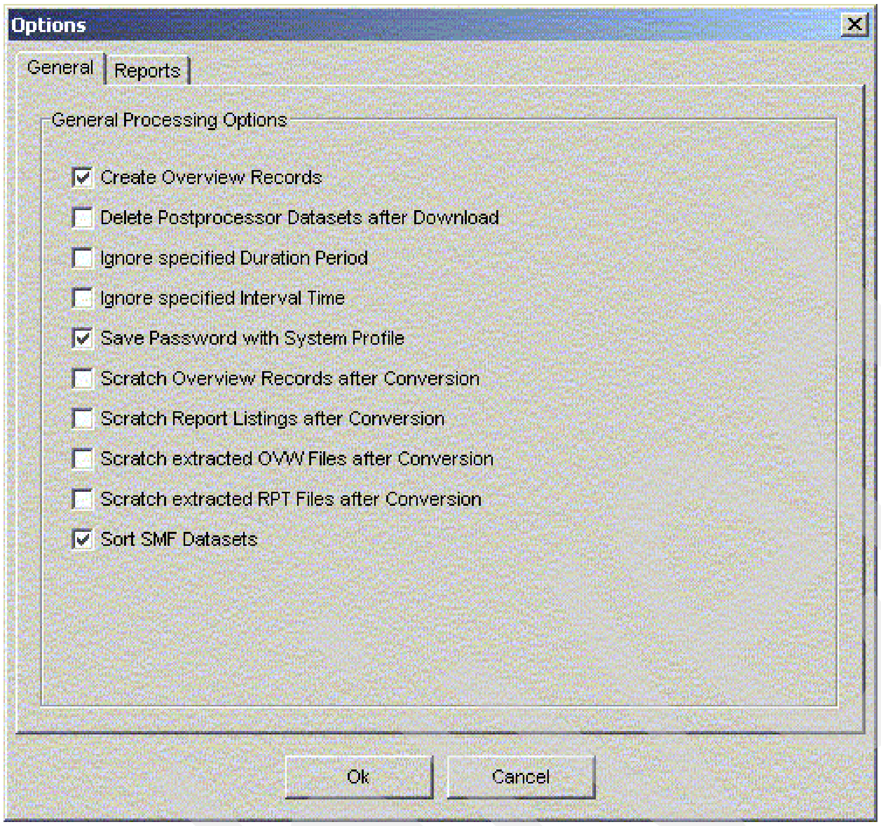
Creating a Report Working Set

After we have defined a system, we are ready to create a Report Working Set. We want to take a look at the workload of our system. Table 2-2 shows that the Workload Activity Trend Report can show us the workload of our system. We also see that this macro uses the Workload Activity report. We check the global Options and Interval settings. And, we need to ensure we have a system selected on the Systems list.

Options and Interval settings

Use **Settings** → **Options** to get the Options dialog, as shown in Figure 2-8.

On the General tab shown in Figure 2-8 we see the general processing options.



General Processing Options

- ☑ Create Overview Records
- □ Delete Postprocessor Datasets after Download
- □ Ignore specified Duration Period
- □ Ignore specified Interval Time
- ☑ Save Password with System Profile
- □ Scratch Overview Records after Conversion
- □ Scratch Report Listings after Conversion
- □ Scratch extracted OVW Files after Conversion
- □ Scratch extracted RPT Files after Conversion
- ☑ Sort SMF Datasets

Figure 2-8 General options

This tab includes the following important settings:

Create Overview Records

If you have defined an Overview control statement file in the system definitions and this option is enabled, an Overview record is created. If you create a Working Set, an Overview Working Set is created. If this option is disabled, but you have defined an Overview control statement file in the system definitions, the Spreadsheet Reporter tries to create a Report Working Set. The creation of the Report Working Set with an Overview Report fails, since the Spreadsheet Reporter cannot convert the Overview Report, but the Overview Reports are available for you for further use in the Working Set.

Ignore specified Duration Period/Interval Time

Ignores the interval settings of Settings → Intervals.

Save Password with System Profile

The password that you specified for a system in dialog Define new System or System Properties is saved but not encrypted. Otherwise you are prompted for the password for all actions that require a host logon.

Scratch extracted RPT Files after Conversion

The Spreadsheet Reporter deletes the local .RPT files after generating Working Sets from Report listings. If you want to use the DASD or Cache Reports macros to filter large DASD Activity and Cache Activity reports, you have to disable this setting.

Scratch extracted OVW Files after Conversion

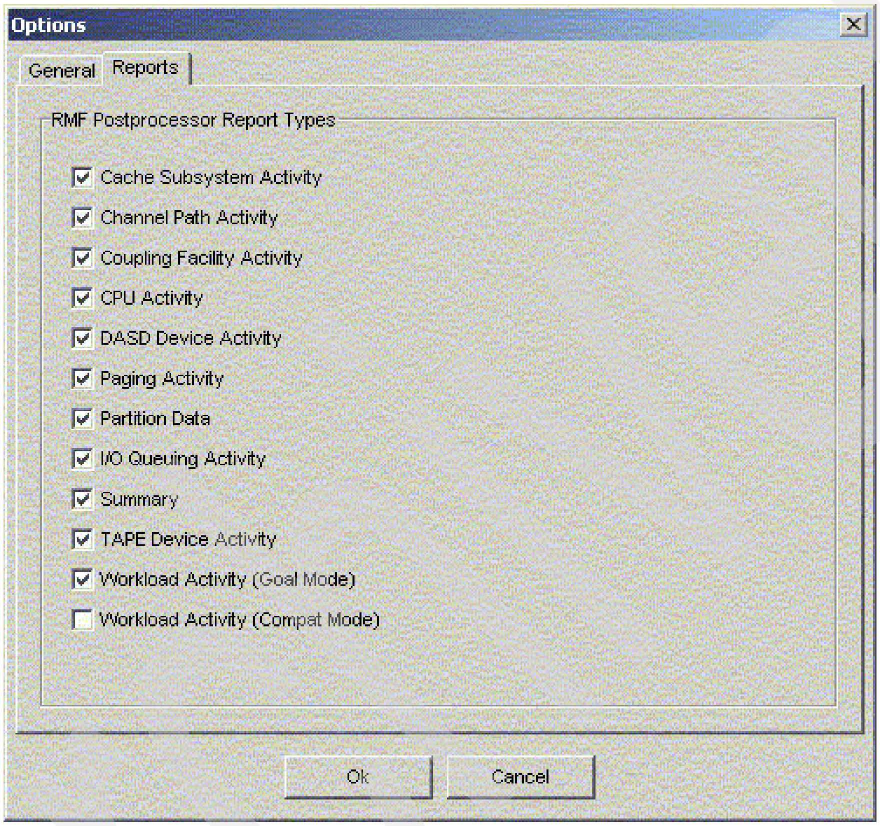
The Spreadsheet Reporter deletes the local .OVW files after generating Working Sets from Overview Records.

Sort SMF Datasets

RMF needs sorted SMF records by RMF interval start date and interval start time in the data set. If you want to combine SMF records from several data sets, you must sort the records to ensure getting correct reports.

On the Reports tab, we see the report options, as shown in Figure 2-9.

Ensure that you request the reports required for the spreadsheet macro you want to use. In Figure 2-9, all reports (except the Workload Activity Compat Mode report) are selected, but this is not always required. With a limited selection of reports, the processing becomes faster.



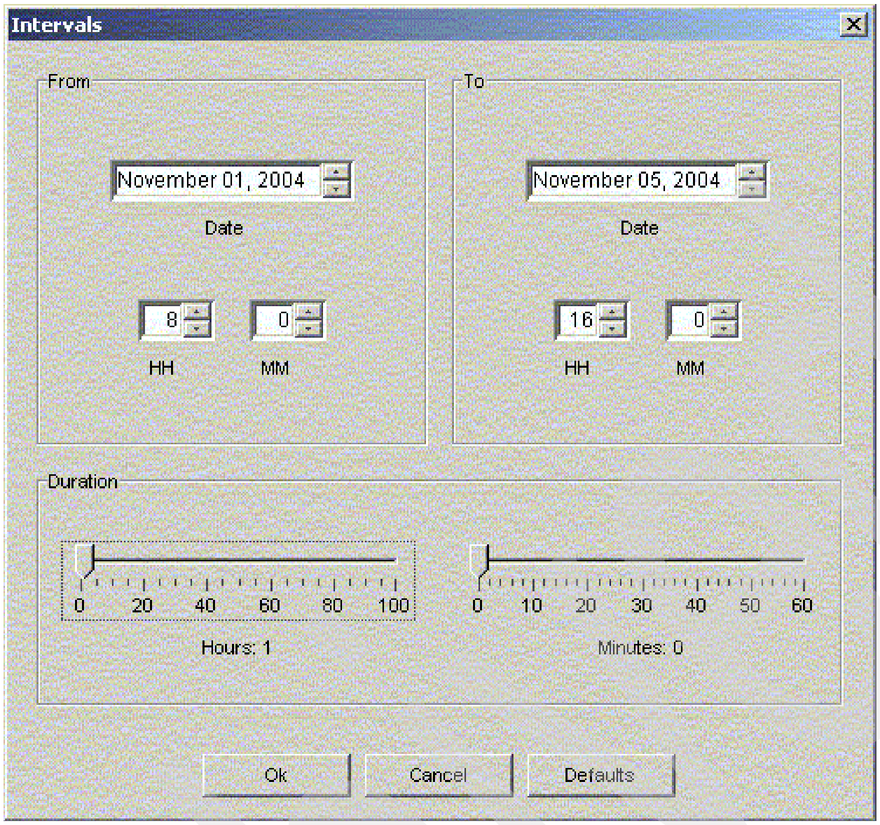
RMF Postprocessor Report Types

☑ Cache Subsystem Activity
☑ Channel Path Activity
☑ Coupling Facility Activity
☑ CPU Activity
☑ DASD Device Activity
☑ Paging Activity
☑ Partition Data
☑ I/O Queuing Activity
☑ Summary
☑ TAPE Device Activity
☑ Workload Activity (Goal Mode)
☐ Workload Activity (Compat Mode)

Figure 2-9 Report options

Select **Settings** → **Intervals** to view the Interval dialog, as shown in Figure 2-10. Here you specify the starting time and ending time of the reporting period for interval or duration reporting. We click **Defaults** to update the interval with the current date. But we also want to take a look from the beginning of the week, so we change the start date. Additionally, we specify that we want to get data only for the time range 8:00 - 16:00.

We want to create a duration report instead of the interval report, with a duration of 1 hour, so we define the length of the duration report in this dialog.



From: November 01, 2004 Date; 8 HH; 0 MM

To: November 05, 2004 Date; 16 HH; 0 MM

Duration: Hours: 1 Minutes: 0

Figure 2-10 Interval settings

Specifying SMF dump data

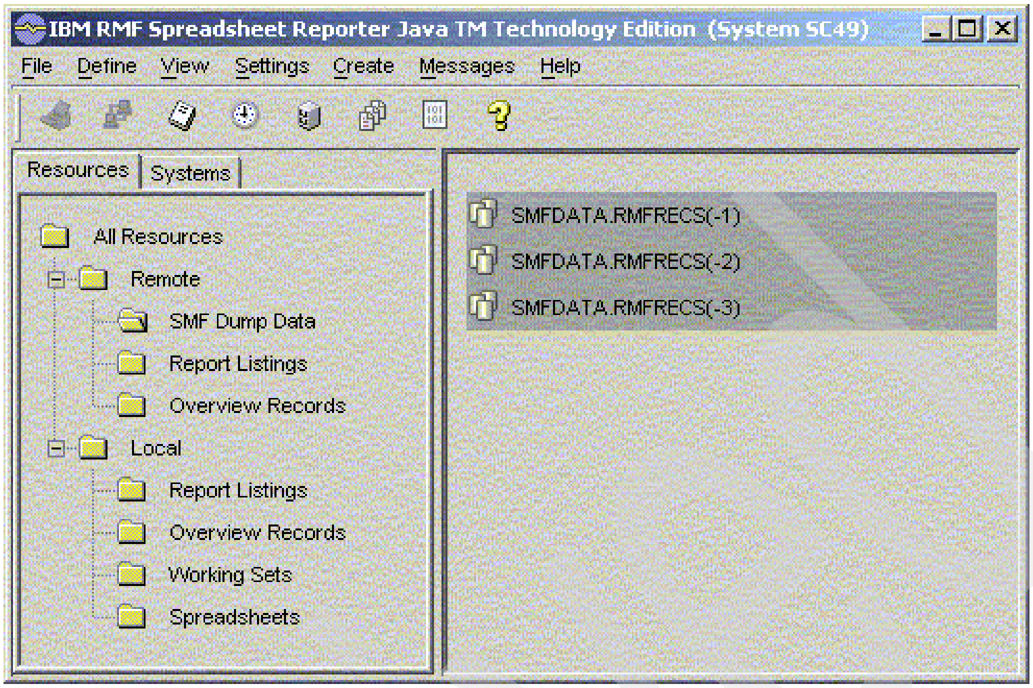
We have to specify the SMF dump data sets, which are typically defined as generation data groups (GDGs), that we want to use for creating the Working Set. Therefore, we switch to the Resource tab and select the remote resource SMF Dump Data.

We have to define our GDG data set as SMF Dump Data, using **Define** → **SMF Dump Data**. We define three GDG data sets, as shown in Figure 2-11:

- SMFDATA.RMFRECS(-1)
- SMFDATA.RMFRECS(-2)
- SMFDATA.RMFRECS(-3)

If you have defined several systems, you have to be sure you have selected the correct system before defining the SMF data.

To rename and delete defined SMF dump data sets, use the context menu of the SMF dump data set.



SMFDATA.RMFRECS(-1)
SMFDATA.RMFRECS(-2)
SMFDATA.RMFRECS(-3)

Figure 2-11 GDG SMF dump data sets

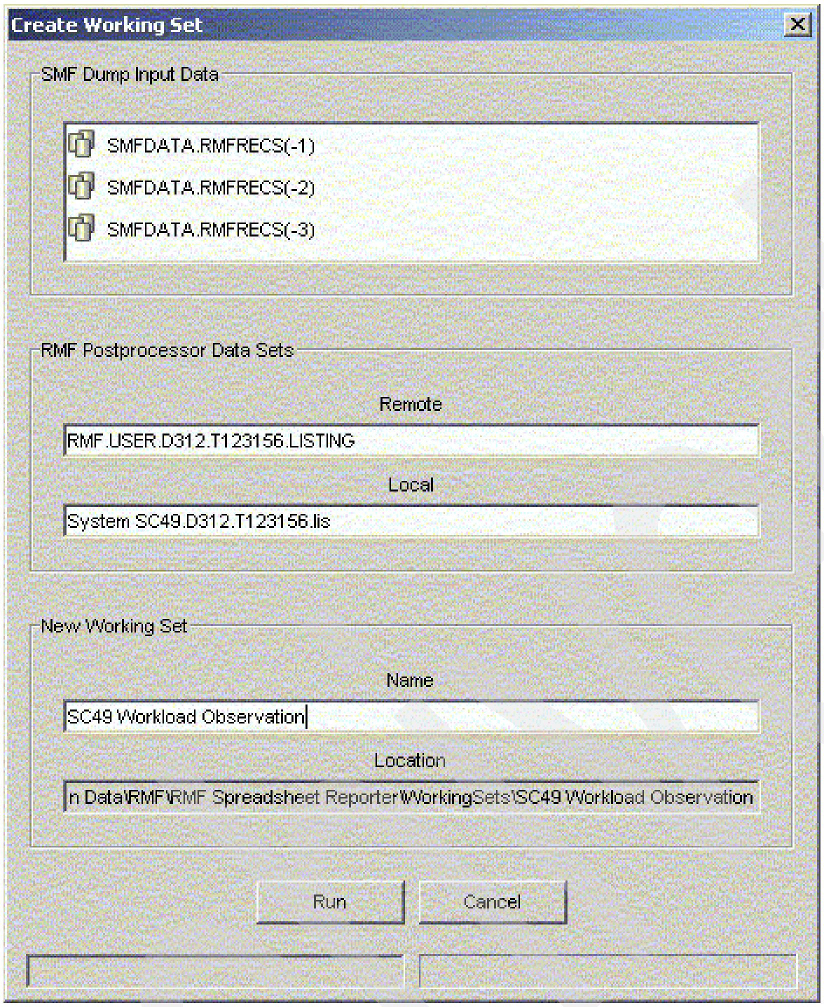
Create Working Set

Now we are ready to create our Report Working Set. We select the defined GDGs and use **Create** → **Working Set**. If you want to use the SMF buffer instead of defined SMF dump data sets, just execute this step without selecting an SMF dump data set.

This brings up the Create Working Set Dialog, as shown in Figure 2-12 on page 56. Here you see at a glance all the important information:

- The used SMF dump input data (empty if SMF buffer is used).
- The name of the remote Report listing, where the defined HLQ is used.
- The name of the local Report listing.
- The name of the Working Set.
- The directory of the Working Set.

If you wish, you can modify the remote/local Report listing name and Working Set. To start the process, click **Run**.



SMF Dump Input Data

SMFDATA.RMFRECS(-1)
SMFDATA.RMFRECS(-2)
SMFDATA.RMFRECS(-3)

RMF Postprocessor Data Sets

Remote: RMF.USER.D312.T123156.LISTING

Local: System SC49.D312.T123156.lis

New Working Set

Name: SC49 Workload Observation

Location: n Data\RMF\RMF Spreadsheet Reporter\WorkingSets\SC49 Workload Observation

Figure 2-12 Create Working Set dialog

After successful execution, the Working Set is created and is now available on the Resource tab as a local Working Set resource, as shown in Figure 2-13. To manage the Working Sets (delete, rename), use the context menu of the Working Set.

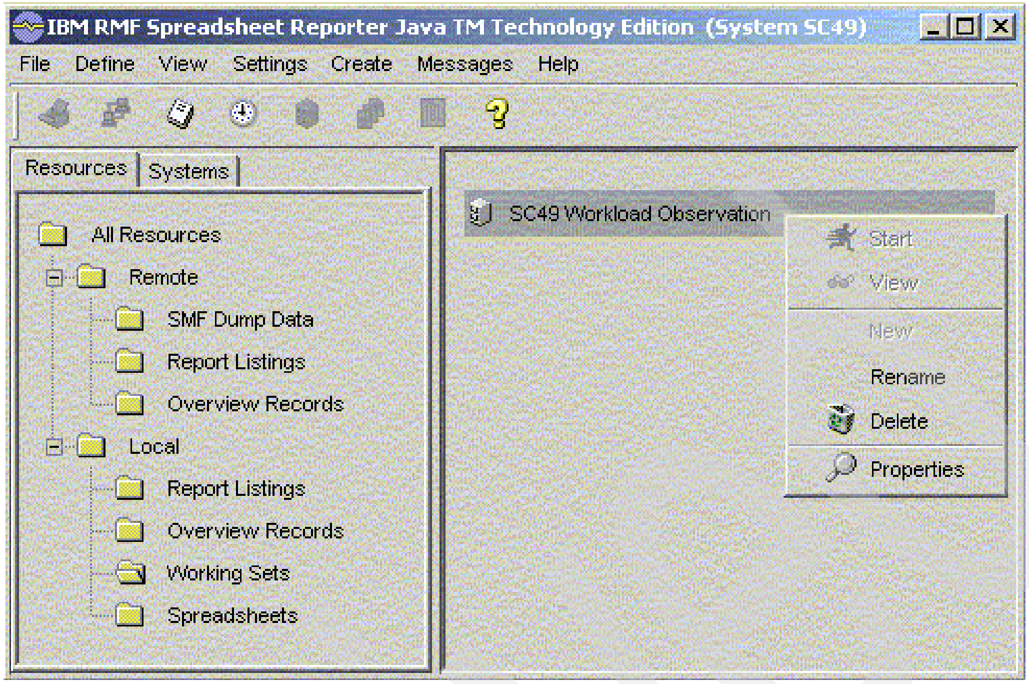


Figure 2-13 Working Set

Messages

After we have created the Report Working Set, we can check messages. Using Messages, we can access:

- JES Joblog
- RMF Postprocessor
- FTP Command log

You can check the RMF Postprocessor messages. They inform you whether the Postprocessor was able to create all the required reports or not. If you think that something is wrong with your Working Set, check the Postprocessor messages; perhaps your required reports were not all generated correctly due to errors.

Using report macros

Now we want to process the Report Working Set with the Workload Activity Trend Report Macro, to analyze the workload.

Therefore, we select the local resource **Spreadsheets**, and start the **Workload Activity Trend Report** macro either by a double-click or by its context menu, as shown in Figure 2-14.

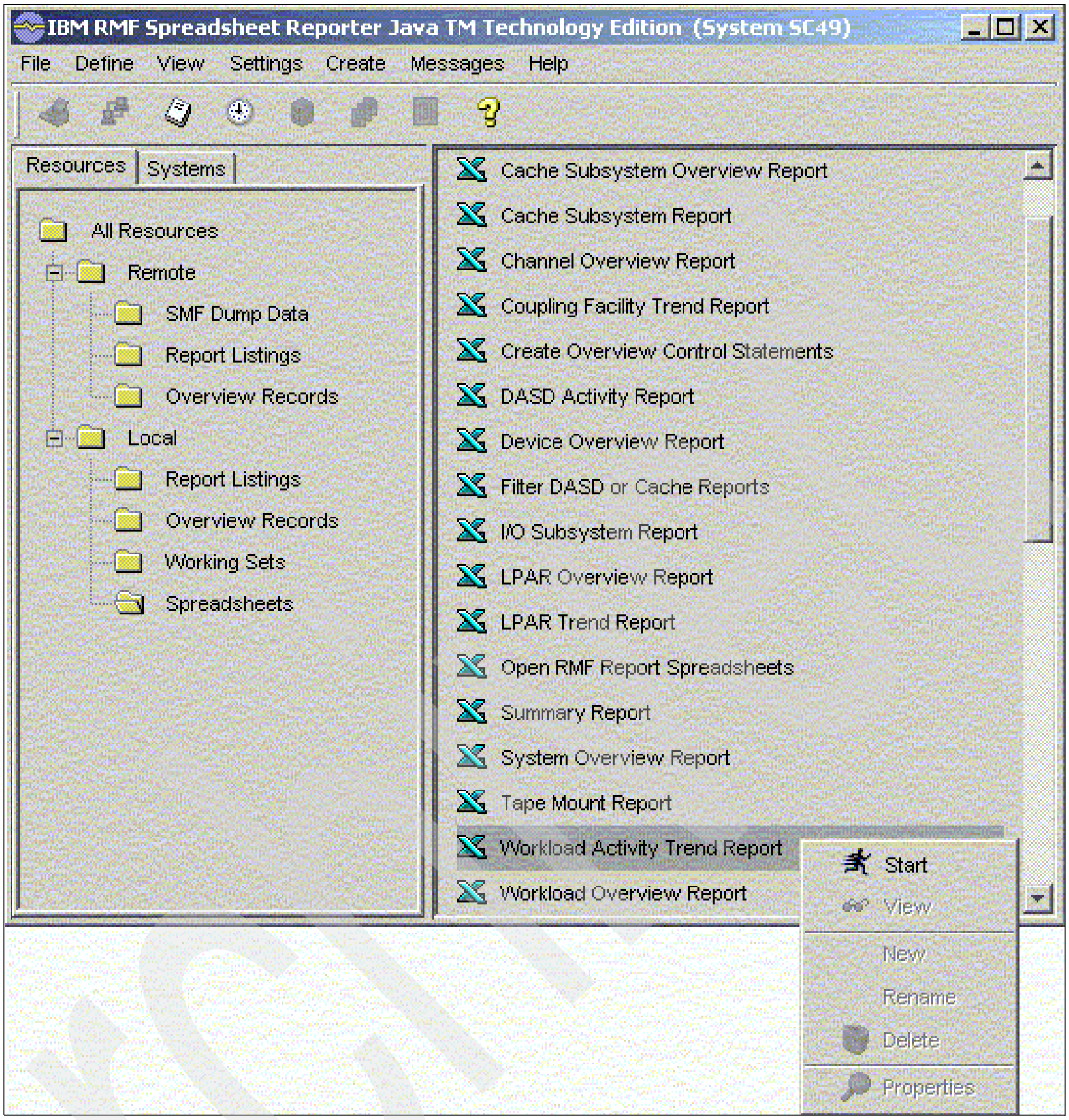


Figure 2-14 Start Workload Activity Trend Report macro

Figure 2-15 shows the Main page of the Workload Activity Trend Report macro. It offers several options:

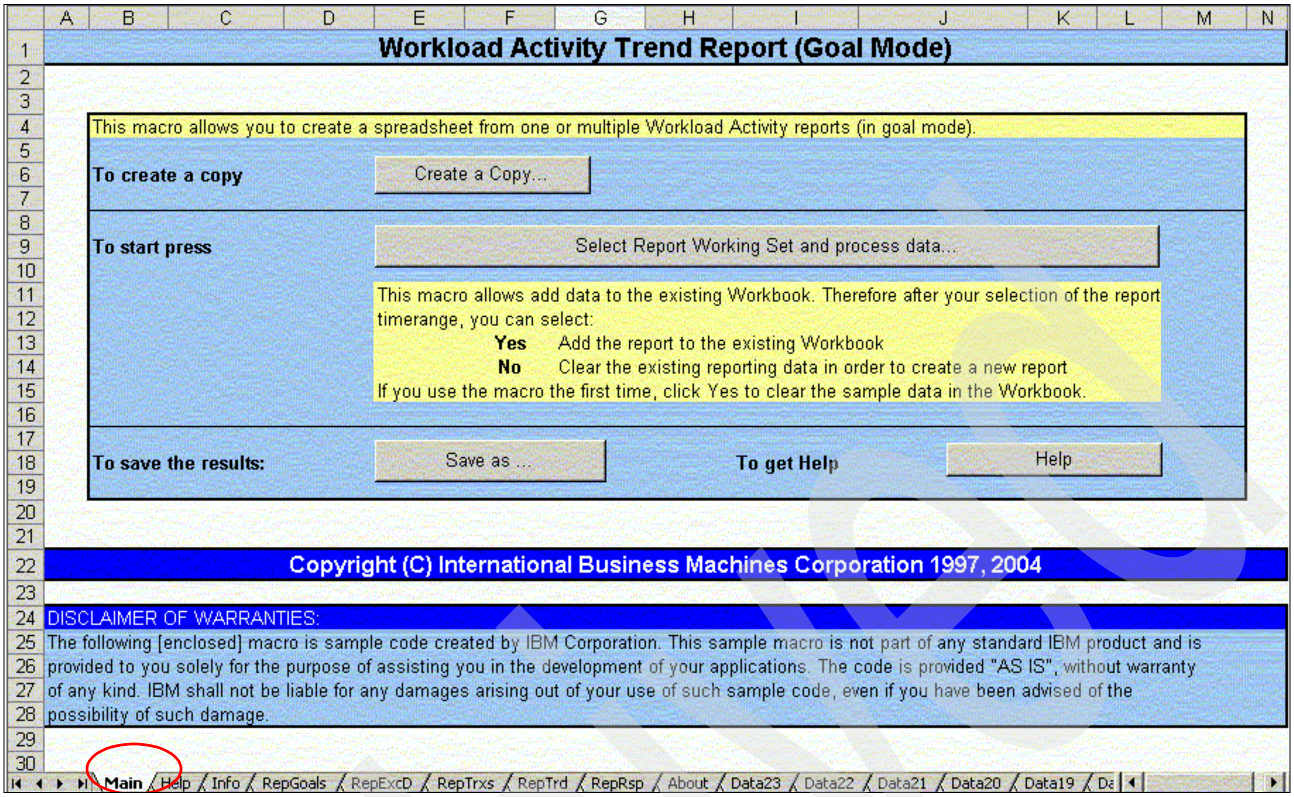
Create a copy — To create a working copy of the data. Use Create a copy to create a copy of the spreadsheet and to keep and protect your data. Otherwise, you may overwrite your current data in the macro when you use the macro again.

Select Report Working Set and process data

Where you start when you want to process your Report Working Set.

Save as — After you have analyzed the data you may want to keep it. You can save the spreadsheet using another name.

Help — Help to use the product.



Workload Activity Trend Report (Goal Mode)

This macro allows you to create a spreadsheet from one or multiple Workload Activity reports (in goal mode).

To create a copy — Create a Copy...

To start press — Select Report Working Set and process data...

This macro allows add data to the existing Workbook. Therefore after your selection of the report timerange, you can select:

Yes	Add the report to the existing Workbook
No	Clear the existing reporting data in order to create a new report

If you use the macro the first time, click Yes to clear the sample data in the Workbook.

To save the results: — Save as ... — **To get Help** — Help

Copyright (C) International Business Machines Corporation 1997, 2004

DISCLAIMER OF WARRANTIES:

The following [enclosed] macro is sample code created by IBM Corporation. This sample macro is not part of any standard IBM product and is provided to you solely for the purpose of assisting you in the development of your applications. The code is provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of such sample code, even if you have been advised of the possibility of such damage.

Figure 2-15 Workload Activity Trend Report macro

We click **Select Report Working Set and process data**.

In the following dialogs we have to select:

1. The Report Working Set to process
2. The systems to process
3. The intervals we want to include (where multiple selection is possible)

After the selection of the intervals, we have the option to:

- Add the data from the Report Working Set to existing data.
- Clear existing data and create a new Workbook.

We select to clear the existing data. You can use this feature to create trend reports, for example, when you start to process the first week of a month with the macro. After you have done your analysis, you save the macro. In the next week, you use the same macro, and add the new data from the current week to the existing data. Thus you can create a trend report.

After we have successfully processed the data, we take a look at the following sheets of the macro:

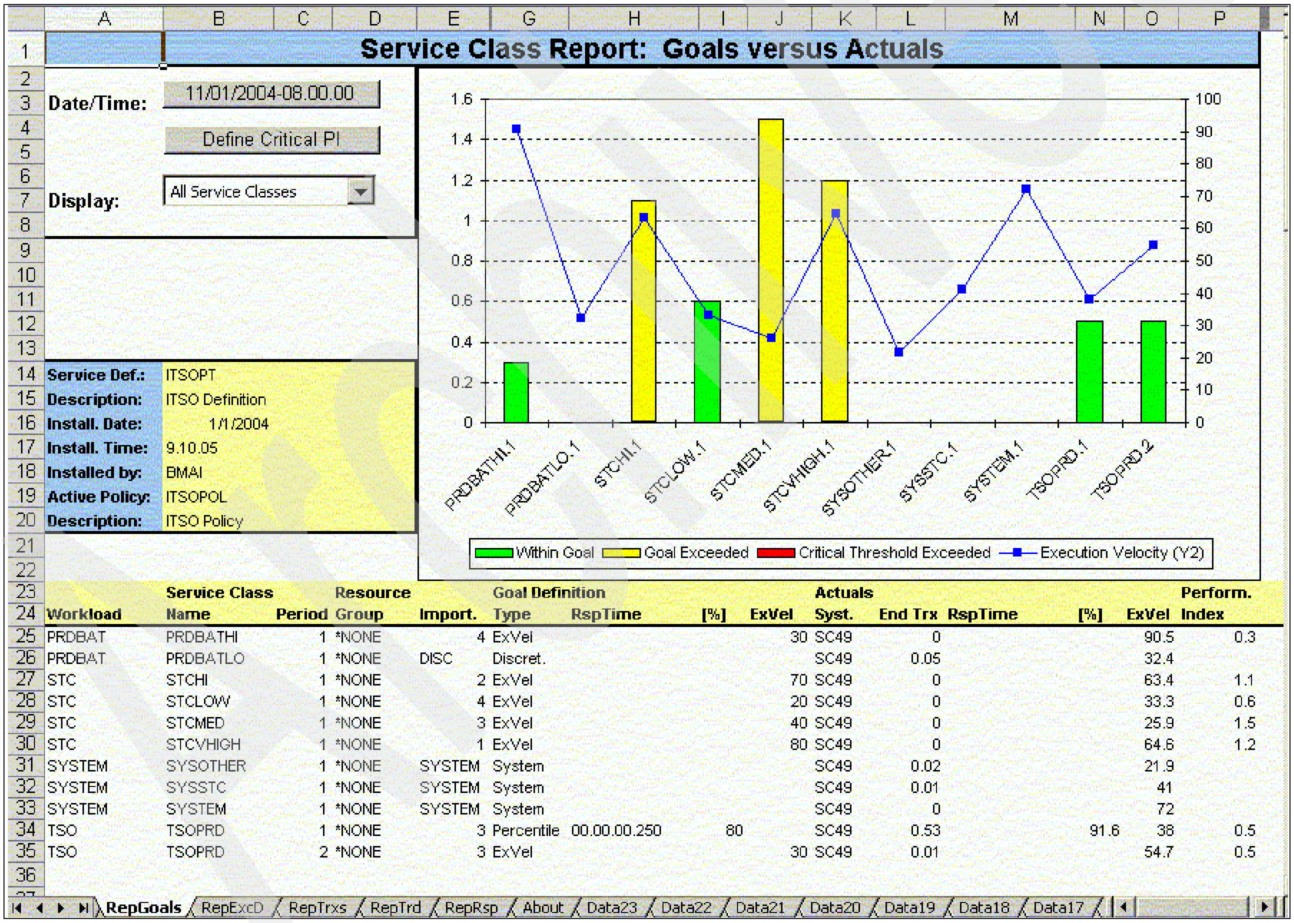
Main	This is the starting point, where we specify the Report Working Set we want to process.
Help	Here we get some basic information.
Info	List of intervals that we have loaded into the spreadsheet.

RepGoals	Here we get a brief overview of the workloads on our sysplex: whether goals could be reached, and some detail information about our service classes, service class periods, and report classes.
RepExcD	An execution delay drilldown of the workloads.
RepTrx	A transaction drilldown of the workloads.
RepTrd	Goal trend report for one selected workload.
RepRsp	Response time distribution for one selected workload.
About	The About sheet contains some general information about the spreadsheet.

All sheets after the About sheet are internal sheets, used by the macro.

Let's take a closer look at the RepGoals sheet. Here we see an overview of the workload of the sysplex on service class level. Three service classes could not attain their goals:

STCHI - period 1	With an PI of 1.1
STCMED- period 1	With an PI of 1.5
STCVHIGH - period 1	With an PI of 1.2



Service Class Report: Goals versus Actuals

Date/Time: 11/01/2004-08.00.00

Display: All Service Classes

Service Def.:	ITSOPT
Description:	ITSO Definition
Install. Date:	1/1/2004
Install. Time:	9.10.05
Installed by:	BMAI
Active Policy:	ITSOPOL
Description:	ITSO Policy

	Service Class		Resource		Goal Definition				Actuals					Perform.
Workload	Name	Period	Group	Import.	Type	RspTime	[%]	ExVel	Syst.	End Trx	RspTime	[%]	ExVel	Index
PRDBAT	PRDBATHI	1	*NONE	4	ExVel			30	SC49	0			90.5	0.3
PRDBAT	PRDBATLO	1	*NONE	DISC	Discret.				SC49	0.05			32.4	
STC	STCHI	1	*NONE	2	ExVel			70	SC49	0			63.4	1.1
STC	STCLOW	1	*NONE	4	ExVel			20	SC49	0			33.3	0.6
STC	STCMED	1	*NONE	3	ExVel			40	SC49	0			25.9	1.5
STC	STCVHIGH	1	*NONE	1	ExVel			80	SC49	0			64.6	1.2
SYSTEM	SYSOTHER	1	*NONE	SYSTEM	System				SC49	0.02			21.9	
SYSTEM	SYSSTC	1	*NONE	SYSTEM	System				SC49	0.01			41	
SYSTEM	SYSTEM	1	*NONE	SYSTEM	System				SC49	0			72	
TSO	TSOPRD	1	*NONE	3	Percentile	00.00.00.250	80		SC49	0.53		91.6	38	0.5
TSO	TSOPRD	2	*NONE	3	ExVel			30	SC49	0.01			54.7	0.5

Figure 2-16 RepGoals sheet

After we have checked the other pages, we want to take a detailed look at the 3 service classes that exceeded their goals: STCHI, STCMED, and STCVHIGH.

The Workload Overview Report macro gives a detailed look at selected workloads.

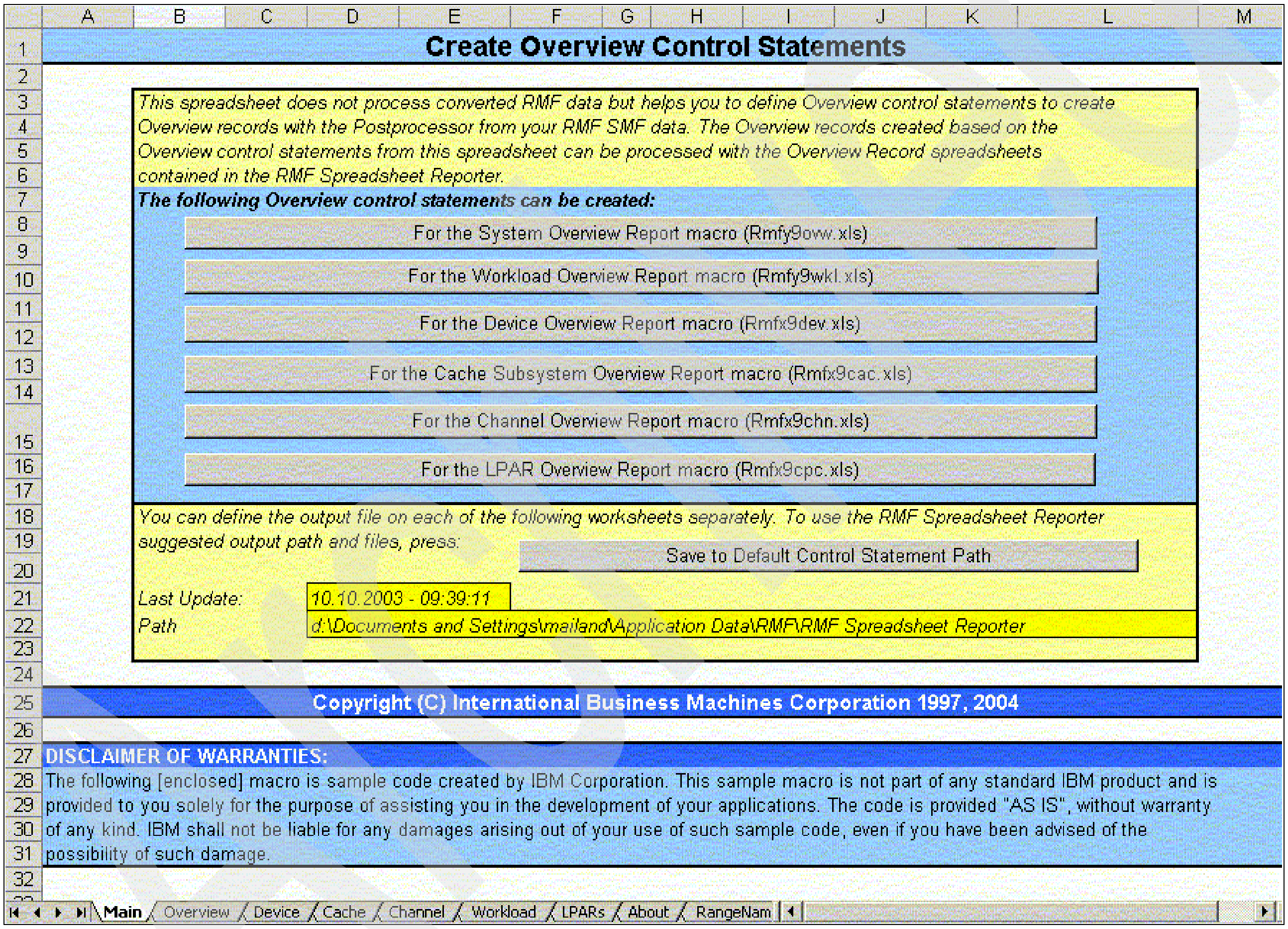
Creating an Overview Working Set

Since the Workload Overview Report macro is based on an Overview Working Set, we need a set of Overview control statements to create it. To do this, we use the Create Overview Control Statements macro.

Create Overview Control Statements macro

This macro creates the set of Overview control statements needed by the Overview macros.

Launch the Create Overview Control Statements macro from the local spreadsheet resources. On the Main page, shown in Figure 2-17, you choose for which Overview macro you want to create the set of Overview control statements. We click **For the Workload Overview Report macro (Rmfy9wkl.xls)**.



Create Overview Control Statements

This spreadsheet does not process converted RMF data but helps you to define Overview control statements to create Overview records with the Postprocessor from your RMF SMF data. The Overview records created based on the Overview control statements from this spreadsheet can be processed with the Overview Record spreadsheets contained in the RMF Spreadsheet Reporter.

The following Overview control statements can be created:

You can define the output file on each of the following worksheets separately. To use the RMF Spreadsheet Reporter suggested output path and files, press:

Last Update: *10.10.2003 - 09:39:11*

Path *d:\Documents and Settings\mailand\Application Data\RMF\RMF Spreadsheet Reporter*

Copyright (C) International Business Machines Corporation 1997, 2004

DISCLAIMER OF WARRANTIES:

The following [enclosed] macro is sample code created by IBM Corporation. This sample macro is not part of any standard IBM product and is provided to you solely for the purpose of assisting you in the development of your applications. The code is provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of such sample code, even if you have been advised of the possibility of such damage.

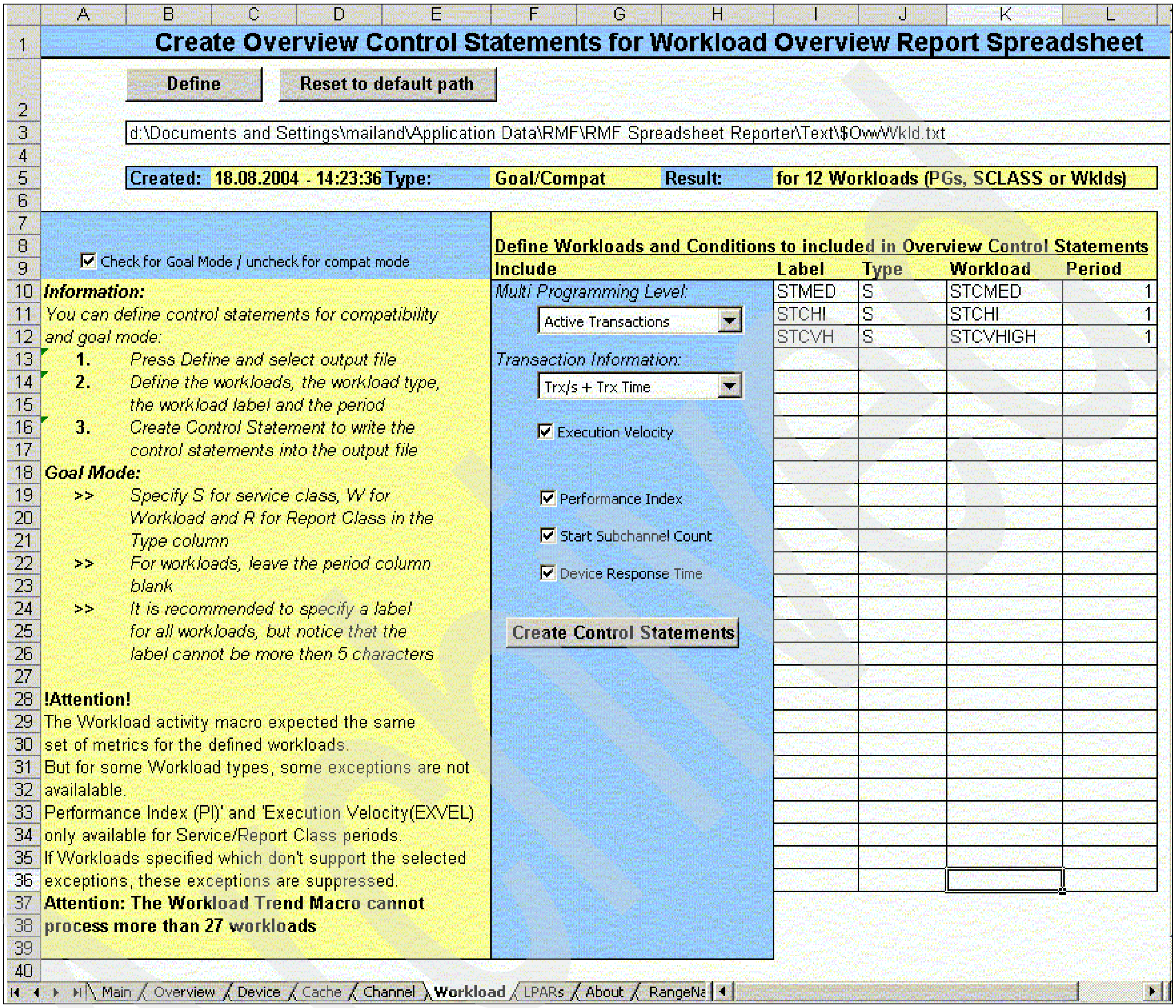
A
B
C
D
E
F
G
H
I
J
K
L
M
For the System Overview Report macro (Rmfy9ovw.xls)
For the Workload Overview Report macro (Rmfy9wkl.xls)
For the Device Overview Report macro (Rmfx9dev.xls)
For the Cache Subsystem Overview Report macro (Rmfx9cac.xls)
For the Channel Overview Report macro (Rmfx9chn.xls)
For the LPAR Overview Report macro (Rmfx9cpc.xls)
Save to Default Control Statement Path
Main
Overview
Device
Cache
Channel
Workload
LPARs
About
RangeNam

Figure 2-17 Create Overview Control Statements macro

On the Workload sheet, shown in Figure 2-18, we now have to define the workloads we want to look at. We have to specify:

Label	Description label, maximum 5 characters.
Type	S for service class, R for report class, and W for Workload.
Workload	The workload name, for example, service class name.
Period	Specify if you want to focus on a period; otherwise, leave the field empty.

We create a definition for our three service classes. Since this is the first time we use the macro, we click **Reset to default path** to set the initial path. After we have finished, we click **Create Control Statements**.



Create Overview Control Statements for Workload Overview Report Spreadsheet

d:\Documents and Settings\mailand\Application Data\RMF\RMF Spreadsheet Reporter\Text\$OvwWkld.txt

Created: 18.08.2004 - 14:23:36 Type: Goal/Compat Result: for 12 Workloads (PGs, SCLASS or Wklds)

☑ Check for Goal Mode / uncheck for compat mode

Information:

You can define control statements for compatibility and goal mode:

1. Press Define and select output file

2. Define the workloads, the workload type, the workload label and the period

3. Create Control Statement to write the control statements into the output file

Goal Mode:

>> Specify S for service class, W for Workload and R for Report Class in the Type column

>> For workloads, leave the period column blank

>> It is recommended to specify a label for all workloads, but notice that the label cannot be more then 5 characters

!Attention!

The Workload activity macro expected the same set of metrics for the defined workloads. But for some Workload types, some exceptions are not availalable. Performance Index (PI)' and 'Execution Velocity(EXVEL) only available for Service/Report Class periods. If Workloads specified which don't support the selected exceptions, these exceptions are suppressed.

Attention: The Workload Trend Macro cannot process more than 27 workloads

Define Workloads and Conditions to included in Overview Control Statements

Include

Multi Programming Level:

Active Transactions

Transaction Information:

Trx/s + Trx Time

☑ Execution Velocity

☑ Performance Index

☑ Start Subchannel Count

☑ Device Response Time

Label	Type	Workload	Period
STMED	S	STCMED	1
STCHI	S	STCHI	1
STCVH	S	STCVHIGH	1

Figure 2-18 Workload definition sheet

Now a file is created that contains all required Overview control statements. Example 2-27 shows the created file for the three service class periods. For a further description of the required statements for Overview macros, refer to the *RMF User's Guide*, SC33-7990.

Example 2-27 Set of Overview control statements

```
OVW(CPUSTMED(APPLPER(S.STCMED.1)))
OVW(EXPSTMED(EXCPRT(S.STCMED.1)))
OVW(MPLSTMED(TRANSAVG(S.STCMED.1)))
OVW(TOTSTMED(TRANS(S.STCMED.1)))
OVW(RTMSTMED(RTIME(S.STCMED.1)))
OVW(EVLSTMED(EXVEL(S.STCMED.1)))
OVW(GPISTMED(PI(S.STCMED.1)))
OVW(SCHSTMED(SSCHRT(S.STCMED.1)))
OVW(RSPSTMED(RESP(S.STCMED.1)))
OVW(CPUSTCHI(APPLPER(S.STCHI.1)))
OVW(EXPSTCHI(EXCPRT(S.STCHI.1)))
OVW(MPLSTCHI(TRANSAVG(S.STCHI.1)))
OVW(TOTSTCHI(TRANS(S.STCHI.1)))
OVW(RTMSTCHI(RTIME(S.STCHI.1)))
OVW(EVLSTCHI(EXVEL(S.STCHI.1)))
OVW(GPISTCHI(PI(S.STCHI.1)))
OVW(SCHSTCHI(SSCHRT(S.STCHI.1)))
OVW(RSPSTCHI(RESP(S.STCHI.1)))
OVW(CPUSTCVH(APPLPER(S.STCVHIGH.1)))
OVW(EXPSTCVH(EXCPRT(S.STCVHIGH.1)))
OVW(MPLSTCVH(TRANSAVG(S.STCVHIGH.1)))
OVW(TOTSTCVH(TRANS(S.STCVHIGH.1)))
OVW(RTMSTCVH(RTIME(S.STCVHIGH.1)))
OVW(EVLSTCVH(EXVEL(S.STCVHIGH.1)))
OVW(GPISTCVH(PI(S.STCVHIGH.1)))
OVW(SCHSTCVH(SSCHRT(S.STCVHIGH.1)))
OVW(RSPSTCVH(RESP(S.STCVHIGH.1)))
OVW(NUMPROC(NUMPROC))
OVW(CPUBUSY(CPUBSY))
OVW(APPLPER(APPLPER(POLICY)))
OVW(EXCPRT(EXCPRT(POLICY)))
```

To create an Overview Working Set, we have to attach the Overview control statement file to the system. But wait – when we do this, afterwards we will have to edit the system definition again to remove the attachment in order to create a standard Report Working Set as before. Thus, for the creation of Overview Working Sets, we recommend defining a second system with identical attributes. Then, in the system definition dialog, we can specify the Overview control statement file exclusively for this system. This allows us to rerun the Overview Working Set creation later without changing the attachments all the time. Figure 2-19 now shows our two systems:

- System SC49 for Report Working Set handling
- System SC49 Overview Working Set for Overview Working Set handling

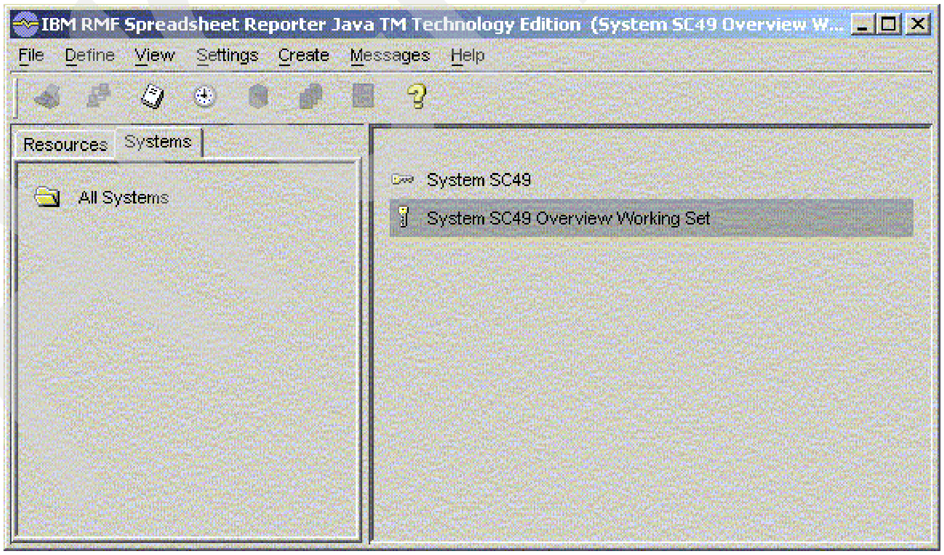

IBM RMF Spreadsheet Reporter Java TM Technology Edition (System SC49 Overview W...
File
Define
View
Settings
Create
Messages
Help
Resources
Systems
All Systems
System SC49
System SC49 Overview Working Set

Figure 2-19 System definition for Overview Working Set

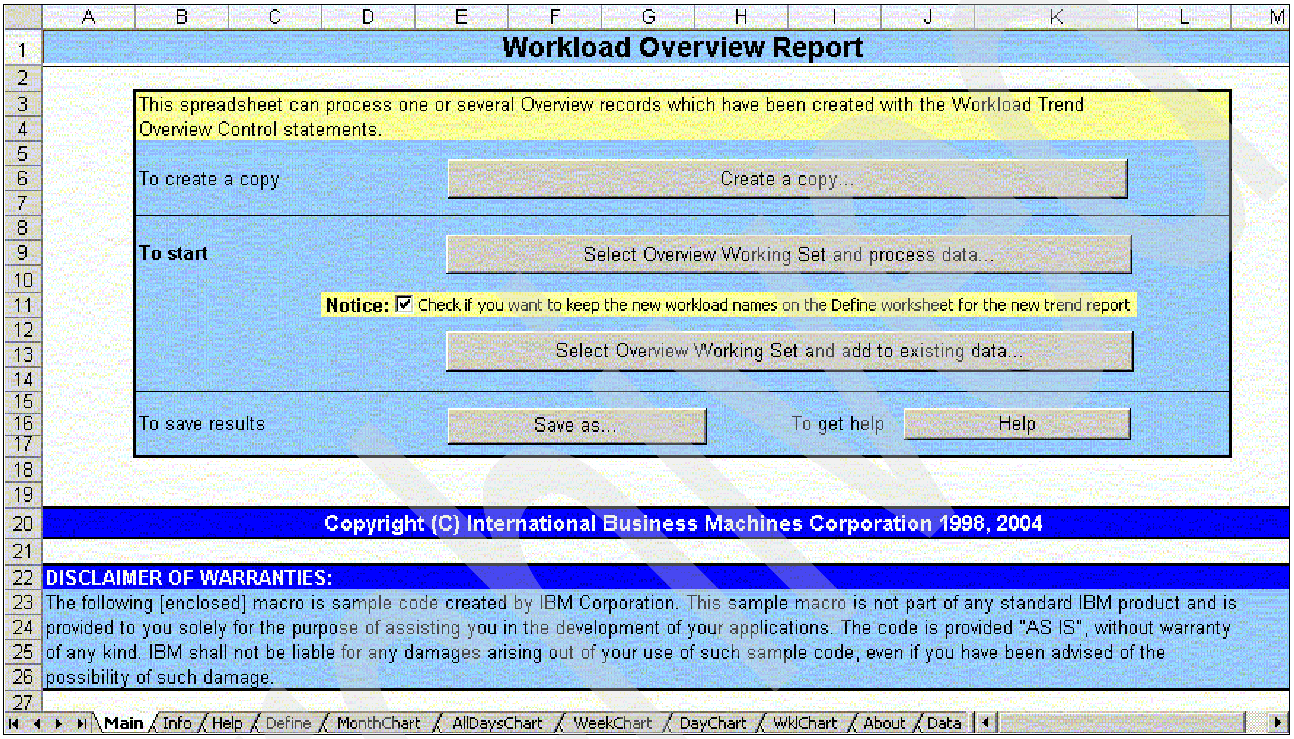
Next we select the newly defined system and switch to the Resource tab. At the Resource tab, we define again the SMF Dump Data resources for the new system. We use the same GDGs that we discussed in “Specifying SMF dump data” on page 54.

We select the three GDG dump data sets, and create the Overview Working Set using **Create** → **Working Set**. The Create Working Set dialog is the same as we used for the creation of the Report Working Set. We click **Run** to start the process.

Now the new Overview Working Set is added to the local Working Sets resources.

Using the Overview macro

To work with the Workload Overview Report macro, select the local Spreadsheet resource and start the macro. Figure 2-20 shows the Main sheet, which looks similar to the Main sheet of the Workload Activity Trend Report macro and has the same functionality.



Workload Overview Report

This spreadsheet can process one or several Overview records which have been created with the Workload Trend Overview Control statements.

To create a copy

To start

Notice: ☑ Check if you want to keep the new workload names on the Define worksheet for the new trend report

To save results

To get help

Copyright (C) International Business Machines Corporation 1998, 2004

DISCLAIMER OF WARRANTIES:

The following [enclosed] macro is sample code created by IBM Corporation. This sample macro is not part of any standard IBM product and is provided to you solely for the purpose of assisting you in the development of your applications. The code is provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of such sample code, even if you have been advised of the possibility of such damage.

Figure 2-20 Workload Overview Report macro - Main sheet

We click **Select Overview Working Set and process data**.

In the following dialogs we have to select:

1. The Overview Working Set to process
2. The system to process
3. The intervals we want to process

After we have successfully processed the data, we can take a look at the other sheets to analyze the data.

This macro offers powerful long-term analysis capabilities. We can create charts on a monthly, weekly, and daily basis. Now we look at the AllDaysChart, shown in Figure 2-21.

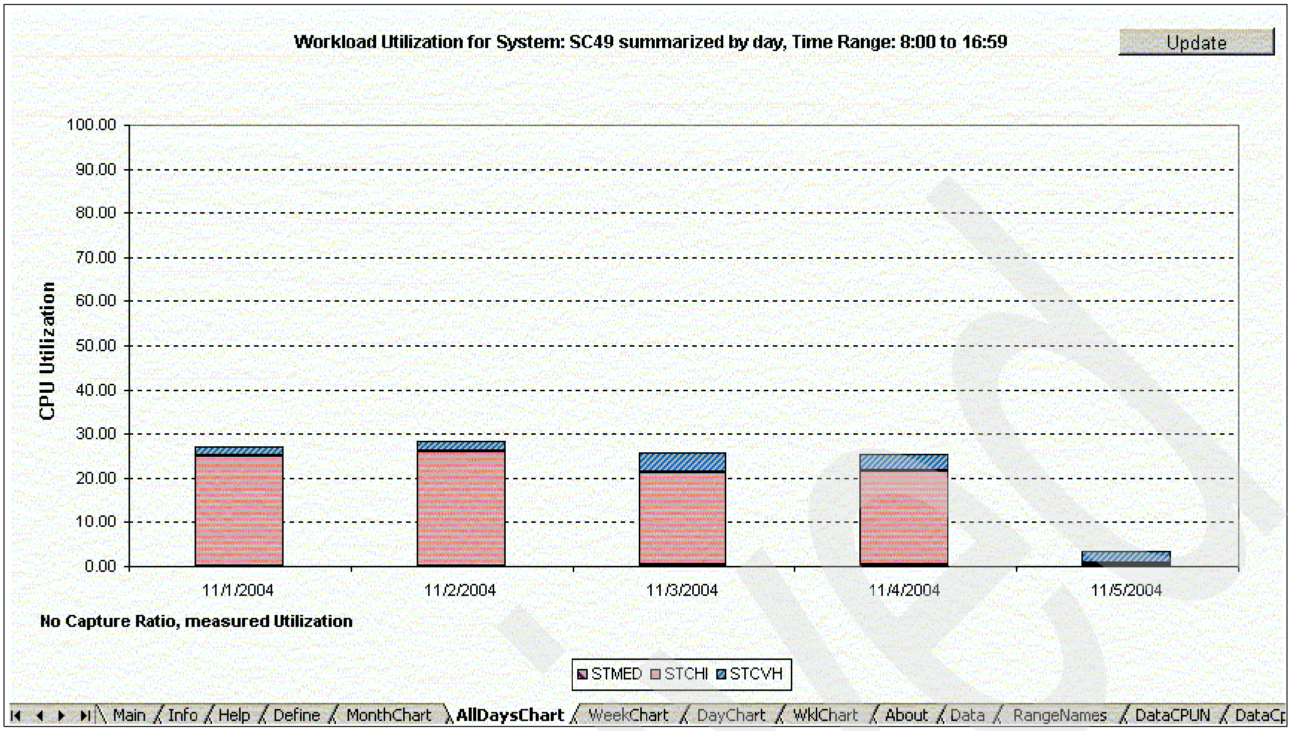


Figure 2-21 Workload Overview Report macro: DayChart sheet

Here we see that service class STCHI is the important one. It uses about five to ten times more CPU service than STCVH. And the CPU utilization of STCMED is almost not visible, compared to the other service classes. Now it is interesting to take a look at data from the last month, to see if we have the same situation. If this is the case, then we should take care about STCHI and keep on tracking that service class.

We decide we want an automatic solution that creates the Overview Working Set on a weekly base without interaction. Then, we can just start our analysis instead of creating the Overview Working Set each week. Therefore, the batch mode of the Spreadsheet Reporter is the right solution for our problem.

Batch mode

The RMF Spreadsheet Reporter provides a collection of procedures that allow you to generate Working Sets in batch mode without any GUI interaction. They are located in the installation directory. The default is:

C:\Program Files\RMF\RMF Spreadsheet Reporter

Batch procedures

Several batch procedures are supplied:

- Jclgen.bat
  This procedure generates the JCL for a job to run the Postprocessor on the host. It contains variables, for example, mfpinput for the SMF data sets, which you have to specify according to your needs. Refer to the *RMF User's Guide* for details.

  Invocation:

  ```
  jclgen option
  ```

where *option* can be one of the following:

sort Sorts the SMF data sets. If you use this option, the mfpinput parameter in Jclgen.bat points to a file containing the names of the required SMF data sets.

nosort Does not sort the SMF data sets. As with option sort, specify the names of the required SMF data sets with the mfpinput parameter.

buffer Takes the SMF data from the RMF Sysplex Data Server's SMF buffer.

- Collect.bat
  This procedure performs the complete SMF data collection as well as the download to the workstation. That is, it submits the job on the host and downloads the resulting Postprocessor output (Report Listing or Overview Record) to the workstation. It contains variables, which you have to modify according to your needs.

  Invocation:

  ```
  collect hostname password type
  ```

  hostname Name of the host where you want to connect

  password Password for the TSO user specified with variable user

  Type Type of Postprocessor output:

  -r if you want to collect data for a Report Listing or

  -o for an Overview Record

  If FTP errors occur during file transmission, the corresponding messages are written into the file ...\Work\ftp.log in the installation directory.

- CreateRptWSet.bat
  This procedure generates a Working Set from an existing local Report Listing.

  Invocation:

  ```
  CreateRptWSet listing dir name
  ```

  listing Path and filename of the Report Listing

  dir Working Set directory

  name Working Set name

- CreateOvwWSet.bat
  This procedure generates a Working Set from an existing Overview Record data set on the workstation.

  Invocation:

  ```
  CreateOvwWSet ovwrec dir name
  ```

  ovwrec Path and filename of the Overview Record file

  dir Working Set directory

  name Working Set name in double quotes.

Using scenario

In this scenario, we automate the creation of an Overview Working Set. To do this, we use:

- Collect.bat to create the Overview records and download them to the workstation
- CreateOvwWSet.bat to create an Overview Working Set from the Overview record

We want to create an Overview Working Set with the following characteristics:

- Covering the week from 11/08/2004 - 11/12/2004

- From 8:00 to 16:00
- Using the set of overview control statements in the file C:\Program Files\RMF\RMF Spreadsheet Reporter\Work\OVW_statements.txt
- The SMF Dump Data is specified in the file C:\Program Files\RMF\RMF Spreadsheet Reporter\Work\SMF_Dump_Data_Sets.txt
- For user MAILAND, using HLQ RMF.USER
- On our system SC49, with IP address 9.12.6.22
- The remote Postprocessor data set to create on the host is RMF.USER.NEW.RMFREP
- The Postprocessor data set is downloaded to C:\Program Files\RMF\RMF Spreadsheet Reporter\Work\local_ovwrecord.rec
- The JES joblog is downloaded to C:\Program Files\RMF\RMF Spreadsheet Reporter\Work\rmfpp.msg
- The RMF messages are downloaded to C:\Program Files\RMF\RMF Spreadsheet Reporter\Work\jes.joblog
- Using the JCL skeleton file C:\Program Files\RMF\RMF Spreadsheet Reporter\Work\rmfpp1.jcl
- The new Overview Working Set name is MyWSet
- In the directory C:\Documents and Settings\TOT133\Application Data\RMF\RMF Spreadsheet Reporter\WorkingSet\MyWSetDir

Figure 2-28 shows an extract of the changed Collect.bat, according to our requirements. We start the Collect.bat using the DOS command:

```
collect 9.12.6.22 password -o
```

Example 2-28 Extract of collect.bat

```
@echo off
@echo.
@echo **********************************************************
@echo * RMF Spreadsheet Reporter - Remote Job Execution Engine *
@echo **********************************************************

set classpath=.;.\jre\bin
set path=.\jre\bin;%path%
set workpath=.\work

rem JCL variable Parameters following here
rem --------------------------------------
set in=%workpath%\rmfpp1.jcl
set out=%workpath%\rmfpp2.jcl
set acct=ACCT
set class=A
set date=DATE(11082004,11122004)
set time=RTOD(0800,1600)
set hlq=RMF.USER
set user=MAILAND
set ppdsn=NEW.RMFREP
set sysin=%workpath%\OVW_statements.txt
set mfpinput=%workpath%\SMF_Dump_Data_Sets.txt
rem ---------------------------
rem End JCL variable Parameters

rem FTP variable Parameters following here
rem --------------------------------------
set log=%workpath%\jes.joblog
set ppfile=%workpath%\local_ovwrecord.rec
set msg=%workpath%\rmfpp.msg
rem ---------------------------
rem End FTP variable Parameters
...
```

To create an Overview Working Set from the downloaded Overview record, we use the batch file with the DOS command.

```
createovwwset "C:\Program Files\RMF\RMF Spreadsheet Reporter\Work\local_ovwrecord.rec" 
"C:\Documents and Settings\TOT133\Application Data\RMF\RMF Spreadsheet 
Reporter\WorkingSets\MyWSetDir" "MyWSet"
```

To automate these steps, we create a new batch file that executes the batch files with the required parameters. Figure 2-29 shows our batch file `generate_ovw_wset.bat` that calls `collect.bat` and `CreateOvwWSet.bat` with the appropriate parameters.

Now we could also schedule the execution of the `generate_ovw_wset.bat` file using the Microsoft Windows® Task scheduler.

Example 2-29 generate_ovw_wset.bat file

```
collect 9.12.6.22 password -o
createovwwset "C:\Program Files\RMF\RMF Spreadsheet Reporter\Work\local_ovwrecord.rec" 
"C:\Documents and Settings\TOT133\Application Data\RMF\RMF Spreadsheet 
Reporter\WorkingSets\MyWSetDir" "MyWSet"
```

Troubleshooting

In this section, we describe two common problems that might appear and what can be done to solve them:

- Problems with the connection to the host system
- Problems with listing data sets that are longer than allowed

Remote Postprocessor execution

If you cannot get the Postprocessor job to run on the host system, you may not have met all of the required prerequisites, described in "RMF Spreadsheet Reporter" on page 134. To check that, use the FTP command line verification. The FTP command line verification checks whether you can connect via FTP to your host and submit a job. If the command line verification fails, this is an indicator that your setup is not correct, and you have to check this with your local support. The log of the FTP command line verification may help them identify the problem. If this happens, you could use the Deferred mode described on page 69.

For FTP command line verification, use a simple IEFBR14 job, as shown in Example 2-30.

Example 2-30 IEFBR14 example job

```
//MAILAND JOB (ITSO),MAILAND,
//         CLASS=A,MSGCLASS=A
//*****************************
//STEP1    EXEC PGM=IEFBR14
//
```

Save the IEFBR14 example job and start a DOS session.

Example 2-31 shows the following sequence:

- Create the file C:\IEFBR14.JCL
- Upload the file C:\IEFBR14.JCL
- Use the quote command to set the file type to JES for job submission.
- Start the job and try to receive the job output into the file `iefbr14.joblog`.
- The job output failed. This means that we are not able to download the job output, for example, JES messages. The Spreadsheet Reporter also tries to receive the job output in order to get additional information about the run. The next step is to check prerequisites and verify our system settings. You may not have authorization to make the required changes, for example, changing the FTP.DATA member to modify the JES time-out. If you do not have the authorization to make the required changes, you can use the deferred mode of the Spreadsheet Reporter.

Example 2-31 FTP DOS command line verification

```
C:\>notepad IEFBR14.JCL

C:\>ftp 9.12.4.58
Connected to 9.12.4.58.
220-FTPD1 IBM FTP CS V1R5 at pst1.itso.ibm.com, 02:45:32 on 2004-11-09.
220 Connection will close if idle for more than 5 minutes.
User (9.12.4.58:(none)): mailand
331 Send password please.
Password:
230 MAILAND is logged on.  Working directory is "MAILAND.".


ftp> put IEFBR14.JCL
200 Port request OK.
125 Storing data set MAILAND.IEFBR14.JCL
250 Transfer completed successfully.
ftp: 417 bytes sent in 0.00Seconds 417000.00Kbytes/sec.

ftp> quote site filetype=jes
200 SITE command was accepted

ftp> get iefbr14.jcl iefbr14.joblog
200 Port request OK.
125-Submitting job iefbr14.jcl FIXrecfm 80
125 When JOB20199 is done, will retrieve its output
550 JesPutGet aborted, job not found

ftp> quit
221 Quit command received. Goodbye.
```

Deferred mode

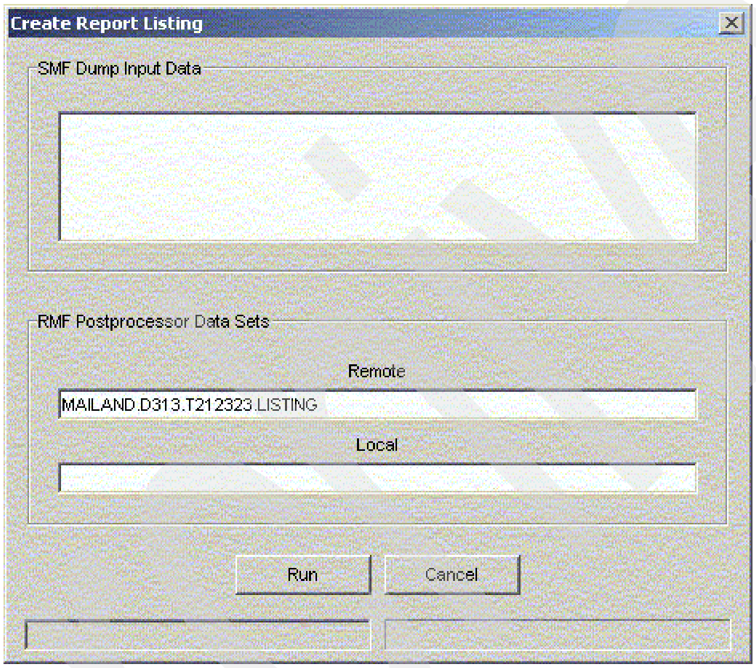
Sometimes you may not have the authorization required to make all the modifications necessary to satisfy the prerequisites for the Spreadsheet Reporter, such as the JES time-out, for example. Therefore, to get around this, use the Spreadsheet Reporter in a deferred mode.

With the deferred mode, first you create the Postprocessor data set on the host. You just submit the job, without retrieving the output. After the job is finished and the output is created, you can now transfer the output to your workstation for further processing, to create a Working Set.

The deferred mode works like this:

Instead of using **Create** → **Working Set**, use **Create** → **Report Listing** or **Create** → **Overview Record**.

Within Create Report Listing or Create Overview Record, you now specify the name of the remote and local Postprocessor data sets. Figure 2-22 shows the Create Report Listing dialog. If you don't specify a name for the local Postprocessor data set, the Spreadsheet Reporter creates only the remote Postprocessor data set. It only submits the JCL to the host and doesn't wait for the end of the job execution to download the job output. The Postprocessor data set is added to the remote resources, either to Report Listings or Overview Records.



Remote: MAILAND.D313.T212323.LISTING

Figure 2-22 Create Report Listing dialog

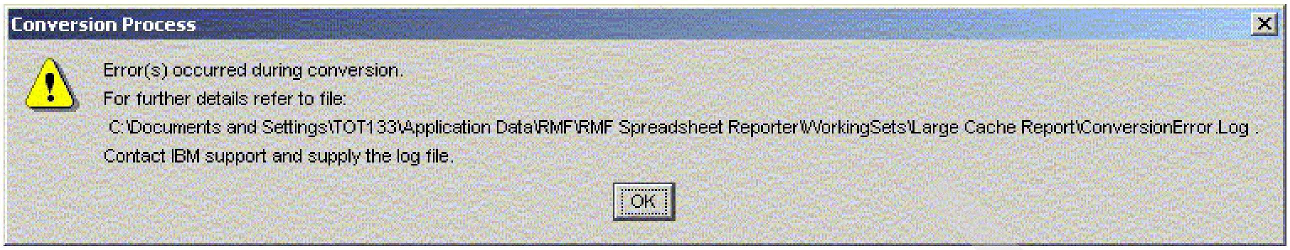
To receive the output later, just select the remote Report Listing or Overview Records and execute **Create** → **Working Set**.

Or you can also use **File** → **Transfer** to download the remote Report Listing or Overview Record to your workstation. It is then added to the local resources. There you can select the local Report Listing or Overview Record, and use **Create** → **Working Set**.

The JES and RMF message output is not retrieved in the deferred mode, so in case of problems you have to check them on the host. The RMF messages are saved into the data set hlq.RMF.MSG.

Large DASD/Cache reports conversion error

The Spreadsheet Reporter converts Postprocessor listings into spreadsheet format. This format has a limitation of 8192 rows per file. If you currently have DASD Activity or Cache Activity Postprocessor reports with a large number of devices, you may reach this limit and the spreadsheet converter cannot convert your report. In this situation, you receive an error message as in Figure 2-23.


Conversion Process

Error(s) occurred during conversion.
For further details refer to file:
C:\Documents and Settings\TOT133\Application Data\RMF\RMF Spreadsheet Reporter\WorkingSets\Large Cache Report\ConversionError.Log .
Contact IBM support and supply the log file.

Figure 2-23 Conversion error

In that case, you have to check the file ConversionError.Log in the Working Set directory as shown in the error message for further information.

In the ConversionError.Log in Example 2-32, the error messages ERB9065A and ERB9066A tell us about the problem (We have reached the limit) and suggest a solution to our problem (Use the Filter DASD or Cache Reports macro).

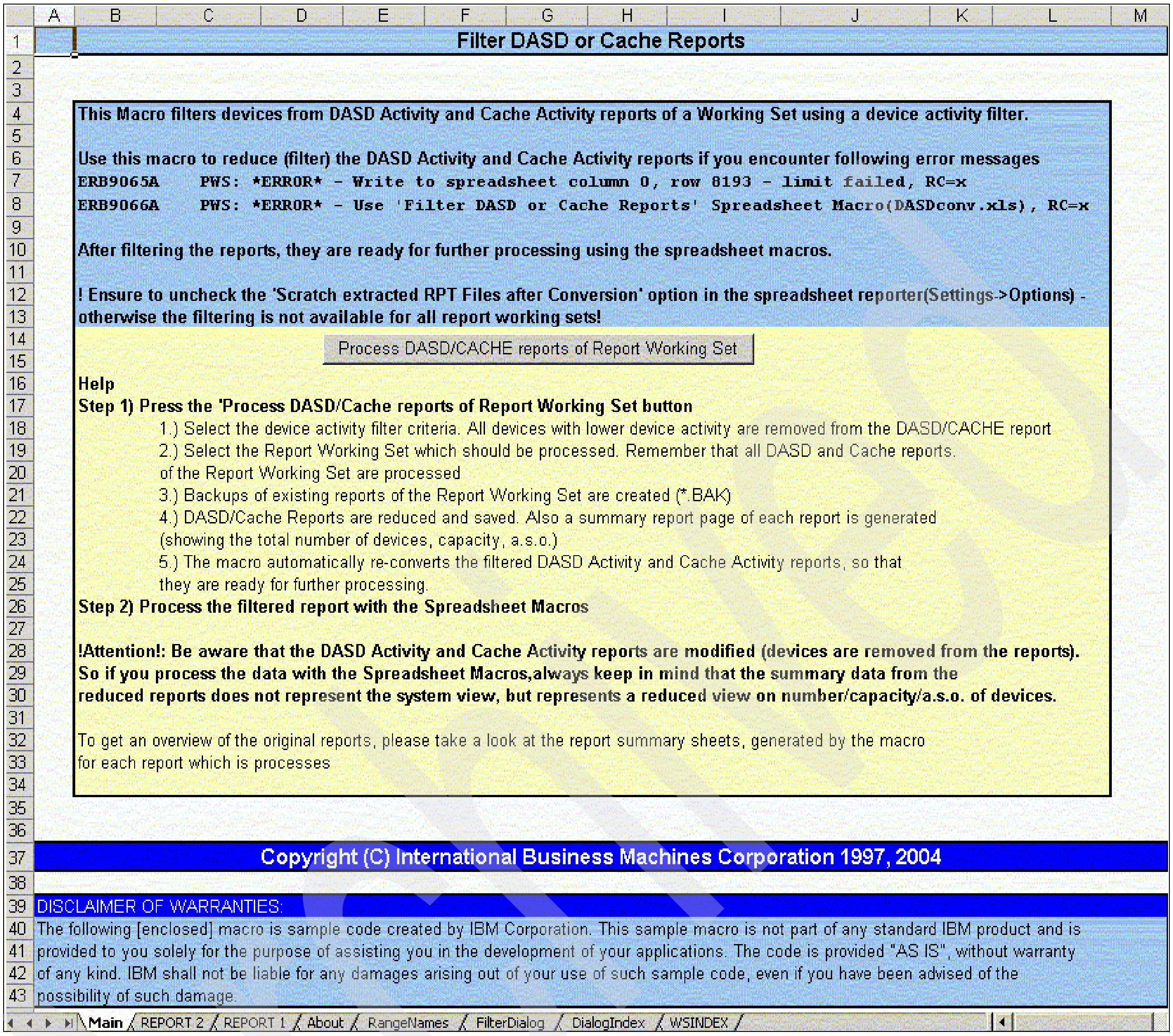
Example 2-32 ConversionError.Log

```
RMF Spreadsheet Reporter  V5.1.3  z/OS V1R5
Copyright (c) 1994, IBM Corporation - all rights reserved

If the error persists, contact IBM support and supply this log and the file:
C:\Documents and Settings\TOT133\Application Data\RMF\RMF Spreadsheet
Reporter\WorkingSets\Large Cache Report\D0000001.RPT

CMD: C:\Program Files\RMF\RMF Spreadsheet Reporter\RMF2SC\rmf2scw.exe /q /nw /f
C:\Documents and Settings\TOT133\Application Data\RMF\RMF Spreadsheet
Reporter\WorkingSets\Large Cache Report\D0000001.RPT /l C:\Documents and
Settings\TOT133\Application Data\RMF\RMF Spreadsheet Reporter\WorkingSets\Large Cache
Report\D0000001.WK1

ERB9065A    PWS: *ERROR* - Write to spreadsheet column 0, row 8193 - limit failed, RC=2
                 Tue Nov 09 08:24:42 2004
ERB9066A    PWS: *ERROR* - Use 'Filter DASD or Cache Reports' Spreadsheet
Macro(DASDconv.xls), RC=2
                 Tue Nov 09 08:24:42 2004
ERB9082A    PWS: *ERROR* - Writing .WK1 values 1 failed, RC=2
                 Tue Nov 09 08:24:42 2004
ERB9010A    PWS: *ERROR* - Failed writing .WK1 file from Report, RC=2
                 Tue Nov 09 08:24:42 2004
RMF Spreadsheet Reporter  V5.1.3  z/OS V1R5
Copyright (c) 1994, IBM Corporation - all rights reserved
```

To use the Filter DASD or Cache Reports macro, select the local Spreadsheet resource, and start the macro.

The primary purpose of this macro is to focus on the important devices, the devices which are really used the most (since you have such a large number of devices, you might not be really interested in every device). Devices with little or no activity are not important for the analysis. Therefore, this macro uses an I/O activity rate filter to reduce the number of devices. The default filter is 0.1 I/O per second.

On the Main sheet, shown in Figure 2-24, we click **Process DASD/CACHE reports of Report Working Set**.



Filter DASD or Cache Reports

This Macro filters devices from DASD Activity and Cache Activity reports of a Working Set using a device activity filter.

Use this macro to reduce (filter) the DASD Activity and Cache Activity reports if you encounter following error messages

```
ERB9065A    PWS: *ERROR* - Write to spreadsheet column 0, row 8193 - limit failed, RC=x
ERB9066A    PWS: *ERROR* - Use 'Filter DASD or Cache Reports' Spreadsheet Macro(DASDconv.xls), RC=x
```

After filtering the reports, they are ready for further processing using the spreadsheet macros.

! Ensure to uncheck the 'Scratch extracted RPT Files after Conversion' option in the spreadsheet reporter(Settings->Options) - otherwise the filtering is not available for all report working sets!

Help

Step 1) Press the 'Process DASD/Cache reports of Report Working Set button

1.) Select the device activity filter criteria. All devices with lower device activity are removed from the DASD/CACHE report

2.) Select the Report Working Set which should be processed. Remember that all DASD and Cache reports. of the Report Working Set are processed

3.) Backups of existing reports of the Report Working Set are created (*.BAK)

4.) DASD/Cache Reports are reduced and saved. Also a summary report page of each report is generated (showing the total number of devices, capacity, a.s.o.)

5.) The macro automatically re-converts the filtered DASD Activity and Cache Activity reports, so that they are ready for further processing.

Step 2) Process the filtered report with the Spreadsheet Macros

!Attention!: Be aware that the DASD Activity and Cache Activity reports are modified (devices are removed from the reports). So if you process the data with the Spreadsheet Macros,always keep in mind that the summary data from the reduced reports does not represent the system view, but represents a reduced view on number/capacity/a.s.o. of devices.

To get an overview of the original reports, please take a look at the report summary sheets, generated by the macro for each report which is processes

Copyright (C) International Business Machines Corporation 1997, 2004

DISCLAIMER OF WARRANTIES:

The following [enclosed] macro is sample code created by IBM Corporation. This sample macro is not part of any standard IBM product and is provided to you solely for the purpose of assisting you in the development of your applications. The code is provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of such sample code, even if you have been advised of the possibility of such damage.

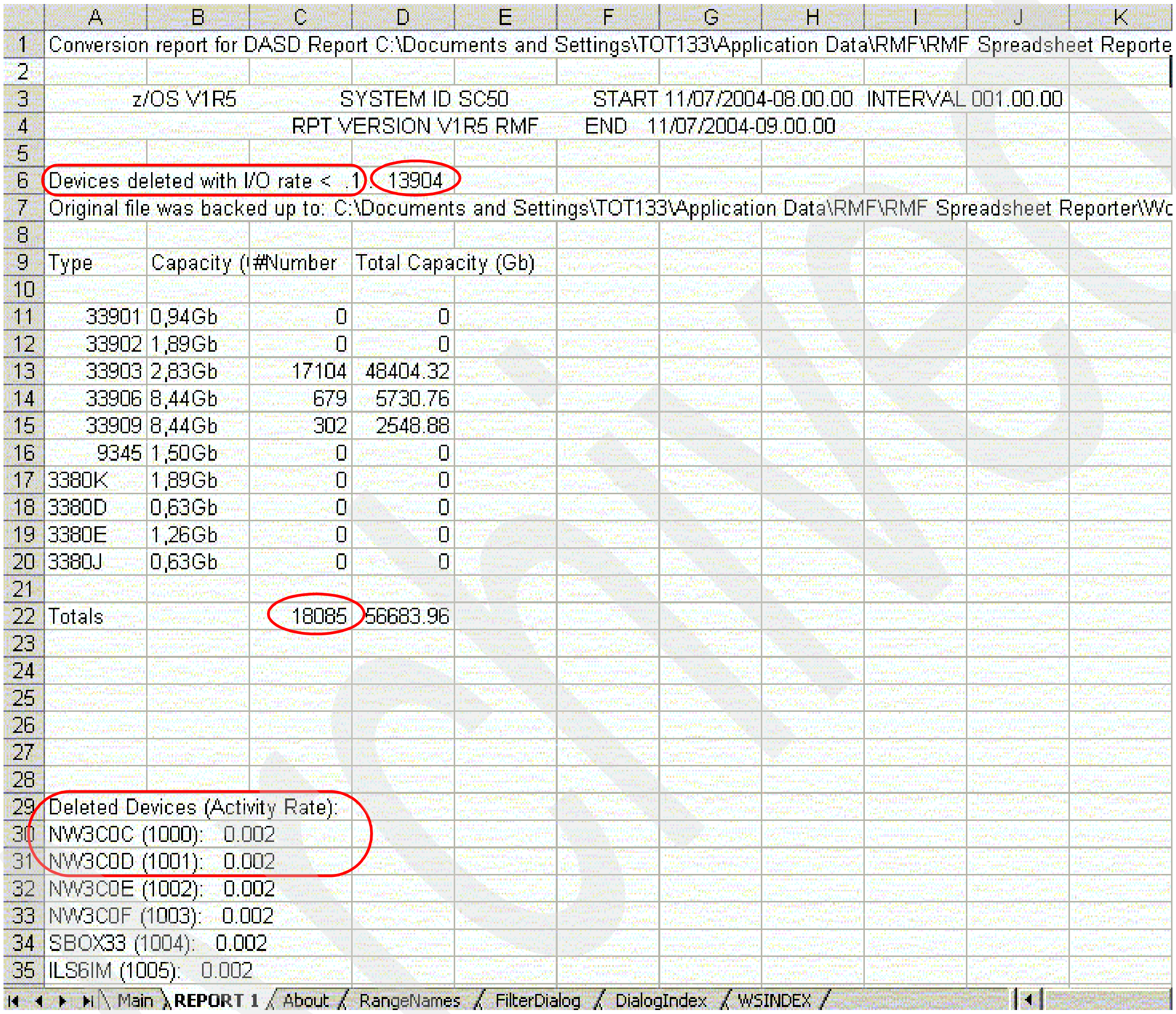
Figure 2-24 Filter DASD or Cache Reports macro

We have to:

1. Select the device activity filter criteria. All devices with lower device activity are removed from the DASD/CACHE report.
2. Select the Report Working Set to process. Remember that all DASD and Cache reports of the Report Working Set are processed.
3. Create Backups (*.BAK) of existing reports of the Report Working Set.
4. DASD/Cache Reports are reduced and saved. Also, a summary report page of each report is generated, showing the total number of devices, capacity, and so forth.
5. The macro automatically reconverts the filtered DASD Activity and Cache Activity reports, so that they are ready for further processing.
6. Process the filtered report with the spreadsheet macros.

The Filter DASD or Cache Reports macro creates a summary report for each processed DASD Activity or Cache Activity report. Figure 2-25 shows a summary report for one processed DASD report. The report contained initially 18085 devices. We have used a filter of 0.1 I/O per second, so all devices with less activity are dropped. We saw a reduction of devices—in total 13904 devices were dropped—because of filtering. So now only 4181 devices are left. There are still many devices to focus on, but we have dropped a lot of devices which are really not of interest for our analysis.

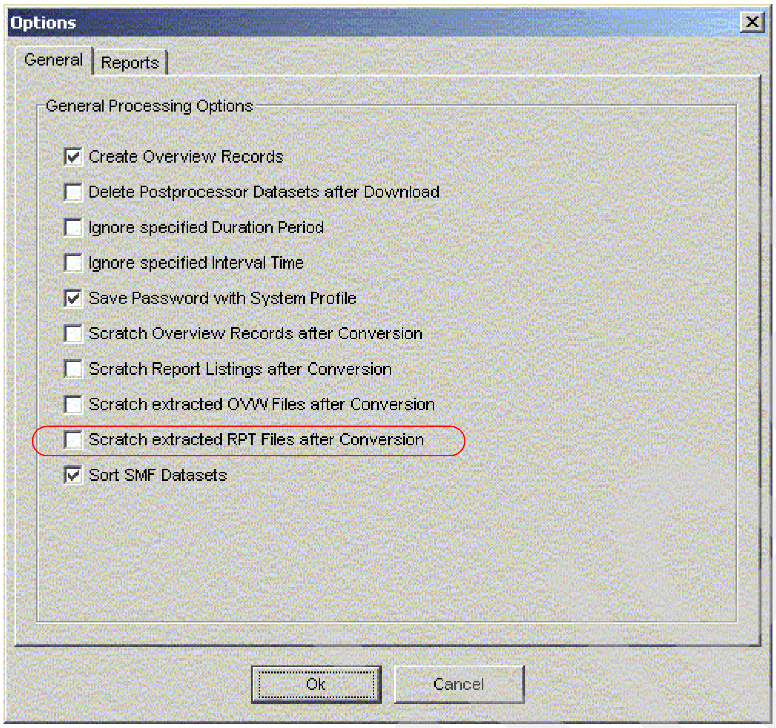
Additionally, you see a list of the dropped devices and their I/O activity in the lower part of the report.



	A	B	C	D	E	F	G	H	I	J	K
1	Conversion report for DASD Report C:\Documents and Settings\TOT133\Application Data\RMF\RMF Spreadsheet Reporte										
2											
3	z/OS V1R5		SYSTEM ID SC50			START 11/07/2004-08.00.00			INTERVAL 001.00.00		
4			RPT VERSION V1R5 RMF			END 11/07/2004-09.00.00					
5											
6	Devices deleted with I/O rate < .1			13904							
7	Original file was backed up to: C:\Documents and Settings\TOT133\Application Data\RMF\RMF Spreadsheet Reporter\Wc										
8											
9	Type	Capacity (	#Number	Total Capacity (Gb)							
10											
11	33901	0,94Gb	0	0							
12	33902	1,89Gb	0	0							
13	33903	2,83Gb	17104	48404.32							
14	33906	8,44Gb	679	5730.76							
15	33909	8,44Gb	302	2548.88							
16	9345	1,50Gb	0	0							
17	3380K	1,89Gb	0	0							
18	3380D	0,63Gb	0	0							
19	3380E	1,26Gb	0	0							
20	3380J	0,63Gb	0	0							
21											
22	Totals		18085	56683.96							
23											
24											
25											
26											
27											
28											
29	Deleted Devices (Activity Rate):										
30	NW3C0C (1000): 0.002										
31	NW3C0D (1001): 0.002										
32	NW3C0E (1002): 0.002										
33	NW3C0F (1003): 0.002										
34	SBOX33 (1004): 0.002										
35	ILS6IM (1005): 0.002										

Figure 2-25 Summary report

You can also use the Filter DASD or Cache Reports macro to filter your reports to focus on important devices. Therefore, you need to disable the option “Scratch extracted RPT Files after Conversion,” as shown in Figure 2-26.



Options

General | Reports

General Processing Options

☑ Create Overview Records

□ Delete Postprocessor Datasets after Download

□ Ignore specified Duration Period

□ Ignore specified Interval Time

☑ Save Password with System Profile

□ Scratch Overview Records after Conversion

□ Scratch Report Listings after Conversion

□ Scratch extracted OVW Files after Conversion

□ Scratch extracted RPT Files after Conversion

☑ Sort SMF Datasets

Figure 2-26 Spreadsheet Reporter Options panel

2.7 RMF Performance Monitoring

RMF Performance Monitoring Java™ Technology Edition (RMF PM) enables you to monitor the performance of your z/OS host or Linux image (zSeries or Intel) from a workstation. You can manage z/OS sysplexes and Linux images from a single point of control by monitoring the resources of the corresponding system.

On z/OS, RMF PM takes its input data from a single data server on one system in the sysplex, which gathers the data from the RMF Monitor III on each image. This function is called the Distributed Data Server (DDS). If you want to monitor several sysplexes, each one needs to have an active DDS. RMF PM supports the complete set of metrics provided by the RMF Monitor III gatherer. This information includes general performance data, performance data for jobs, performance data for systems running in goal mode, and workload-related performance data such as:

- WLM workloads
- WLM service classes and periods
- WLM report classes

On Linux images or guests, RMF PM takes its input data from the RMF Linux data gatherer (rmfpms) running on the image which is being monitored. To monitor several Linux systems, each system must have an active data gatherer. For Linux, general performance data is provided, as well as Apache performance data.

You have the flexibility to create unique views that monitor the performance of your system. You can display real-time and historical data as bar charts. Additionally, you can combine data from different resources. Once you have created these views, you can save them as PerfDesks. With PerfDesks, you create a set of DataViews customized to your monitored systems. DataViews sample performance data into one or more Series displayed as bar charts. You can also save the data from the DataViews into spreadsheet format, for further processing with spreadsheet applications.

You simply open the PerfDesk and start it whenever you want to view performance data in your monitored system again from the same perspective. Once you have defined a set of PerfDesks, you can export them and share them with other RMF PM users. Also, RMF PM offers powerful analysis functions to drill down problems.

Figure 2-27 on page 75 shows the user interface of RMF PM. On the left side, you see the resource structure of the selected sysplex. On the right side, an open PerfDesk with three active DataViews is displayed. In the first DataView, you see a single metric, the number of users and active users. The second and the third DataView display list metrics, showing the performance index by service class periods and the volumes of the sysplex by I/O intensity.

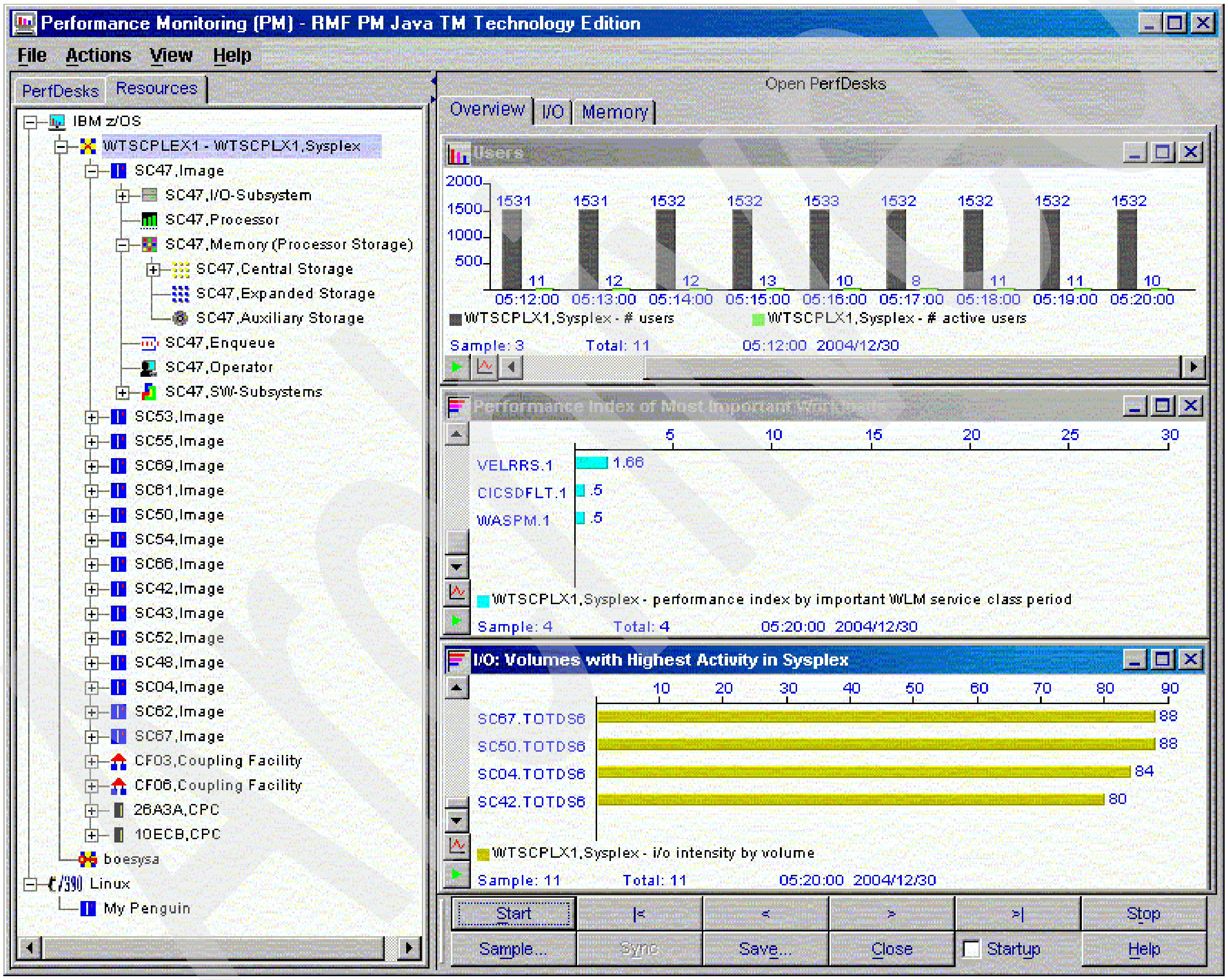


Figure 2-27 RMF Performance Monitoring

RMF PM is available for Microsoft Windows and also for Linux. For the Linux version, check the RMF home page.

2.7.1 Using RMF PM

For instructions on how to install RMF PM, see “Installing RMF PM under Microsoft Windows” on page 147 or “Installing RMF PM under Linux on your workstation” on page 148.

To use RMF PM, you must have an active DDS on a z/OS system or an active rmfpms on a Linux system. For information about the setup, see “RMF Distributed Data Server” on page 136 and “RMF Linux data gatherer” on page 138.

Also, ensure that your TSO user ID has sufficient access to RMF performance data, described in “Controlling access to RMF data for the sysplex data services” on page 116.

To start RMF PM on Microsoft Windows, use **Start** → **Programs** → **IBM RMF Performance Management** → **RMF PM**.

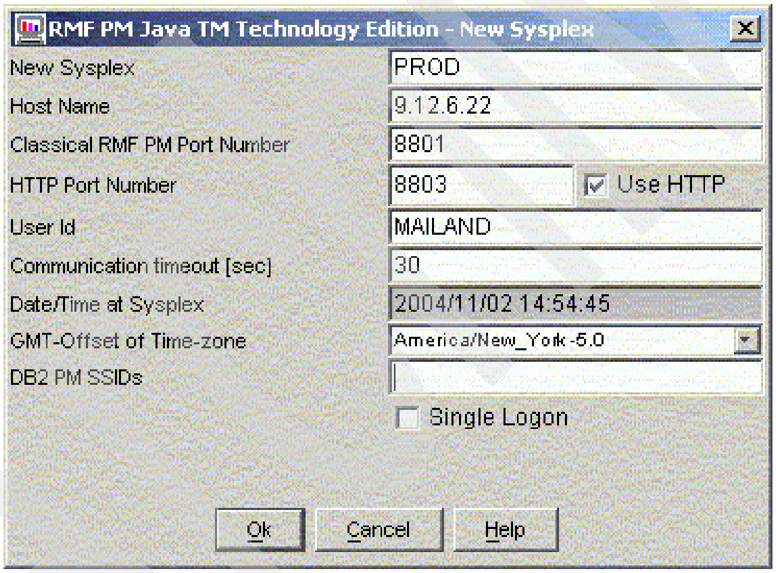
To start RMF PM on Linux, execute the script *rmfpm* in the *rmfpm/* directory with the command **`rmfpm/rmfpm`**.

Here we discuss RMF PM on Microsoft Windows. However, RMF PM on Microsoft Windows is similar to RMF PM on Linux in both usage and functionality.

The Sysplex definition dialog

When you start RMF PM the first time, you have to define the sysplex that you want to monitor and where you have an active data gatherer running.

Figure 2-28 shows the Sysplex definition dialog. The Linux Image definition dialog is nearly the same, but some z/OS-specific definitions, like the DB2® PM SSIDs definition, are missing.



New Sysplex	PROD
Host Name	9.12.6.22
Classical RMF PM Port Number	8801
HTTP Port Number	8803 ☑ Use HTTP
User Id	MAILAND
Communication timeout [sec]	30
Date/Time at Sysplex	2004/11/02 14:54:45
GMT-Offset of Time-zone	America/New_York -5.0
DB2 PM SSIDs	
	☐ Single Logon

Figure 2-28 Sysplex definition dialog

The meanings of the important fields are as follows:

Sysplex
: Description of the sysplex.

Host name
: The TCP/IP name of the host where DDS is active, either a hostname like XYZ.PROD.IBM.COM or an IP address like 9.164.182.251.
If you do not know your system’s hostname, you can retrieve this hostname and the system’s TCP/IP address with the TSO command **`hometest`**.

Classical RMF PM Port Number/HTTP Port Number
: Enter the Session port Number (default is 8801) and the HTTP port number (default is 8803) of the DDS Server on the host.

Ensure that the ports you use are open for communication and not blocked by your Workstation or network firewall.

User ID
Your TSO user ID, used for logon. Ensure that you have sufficient access to RMF Performance data, described in “Controlling access to RMF data for the sysplex data services” on page 116.

DB2 PM SSID
From RMF PM, you can start DB2 PM modules. Use **Actions** → **DB2 PM** → **Statistics/Threads/Exceptions** to start them. You have to specify each DB2 PM SSID locally configured on your workstation and associated with the TSO user ID given in this dialog.

Single Logon
Check this if you are using the same user ID and password for several sysplexes. RMF PM assumes you are using the same user ID and password combination for all sysplexes which have this checkbox selected, so you do not need to type in your password every time.

We recommend that you use the supplied default values for the Ports and the time-out definition.

Click **OK** to finish your definition. Now the sysplex is added to the resource view, as shown in Figure 2-29. In this figure, we also see the default Linux System My Penguin below the Linux resource.

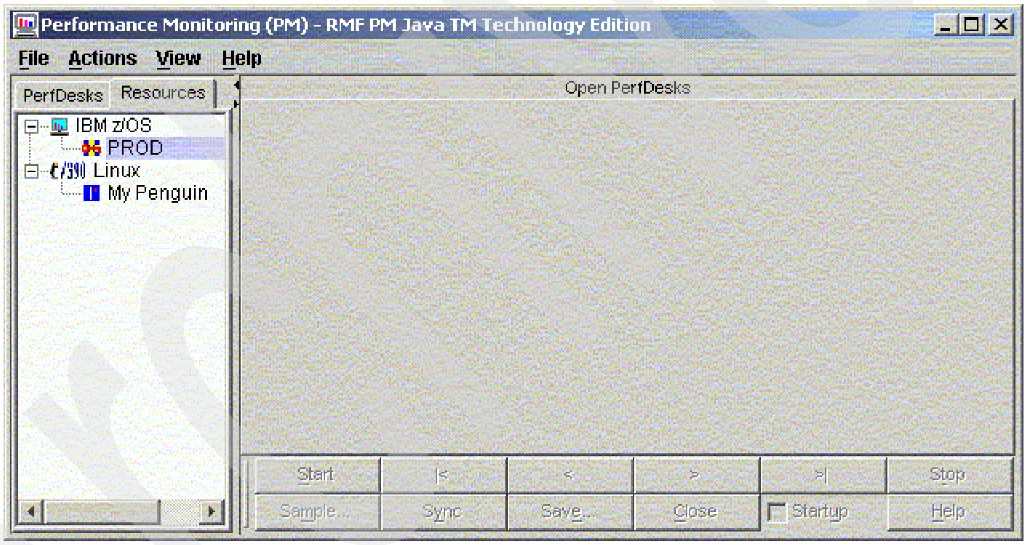


Figure 2-29 RMF PM - Resource view

Working with the sysplex can be described very quickly:

- To connect to the sysplex, select **File** → **Open** → **Sysplex** or just double-click the Sysplex resource **PROD**. On the logon dialog, enter your password and click **OK**.
- To modify your sysplex settings, select **File** → **Change Settings** → **Sysplex** or the context menu of the sysplex resource.
- To define an additional sysplex, select **File** → **New** → **Sysplex**.

The main window

After you have started RMF PM, you get the main window of RMF PM, as shown in Figure 2-30.

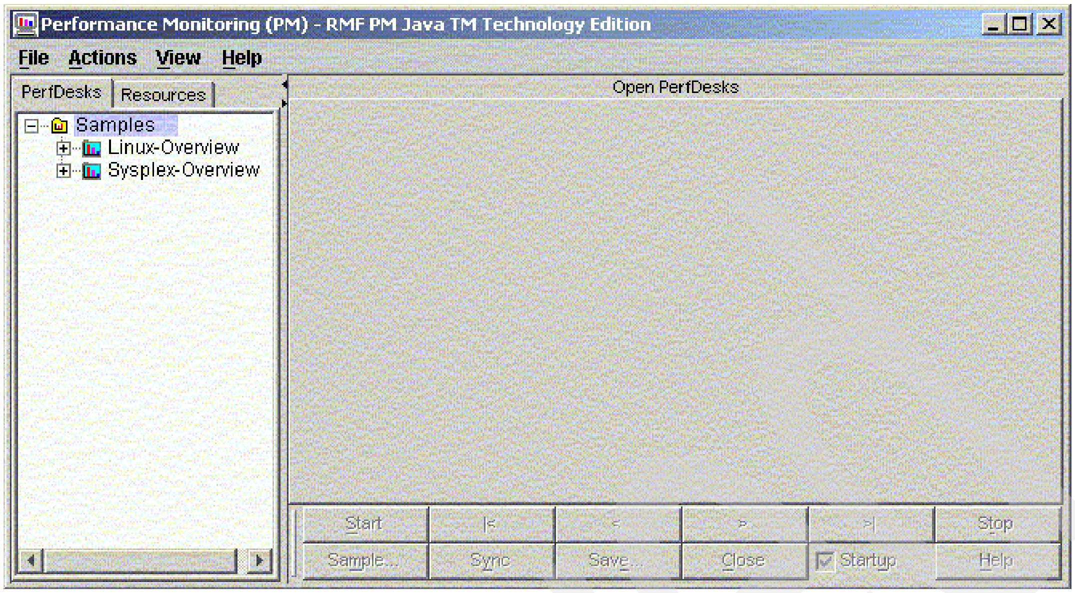


Figure 2-30 RMF PM main window

There are two views:

- The view on the right is the Open PerfDesks view.
- The view on the left with two tabs contains the PerfDesks view and the Resources view.

The performance data in RMF PM is displayed in DataViews. In Figure 2-30, you see the default PerfDesks Linux-Overview and Sysplex-Overview in the PerfDesks view.

These two PerfDesks are saved in the PerfDesk folder Samples. You can create PerfDesk folders and store your own PerfDesk into different folders to structure your PerfDesks. If you open a PerfDesk, it is displayed in the Open PerfDesks view. The Resources view contains the resource structure of your system.

Overview PerfDesks

The Sysplex-Overview and Linux-Overview PerfDesks provide a high-level health check overview of your system. You cannot modify these predefined PerfDesks.

The Sysplex-Overview PerfDesk for z/OS displays the following perspectives and their associated metrics:

Processor	% processor utilization by MVS image
I/O	i/o intensity by volume
WLM	performance index by WLM service class period
Sysplex	# users by MVS image

The Linux-Overview PerfDesk for zSeries Linux displays the following perspectives and their associated metrics:

Processor	CPU % time by process and total processor utilization
Filesystem	% used and size of file systems
Storage	Page fault rate and resident memory size by process
Apache HTTP Server	Apache HTTP Server access statistic

These predefined overview PerfDesks are started automatically. Figure 2-31 shows the Sysplex-Overview PerfDesk for z/OS.

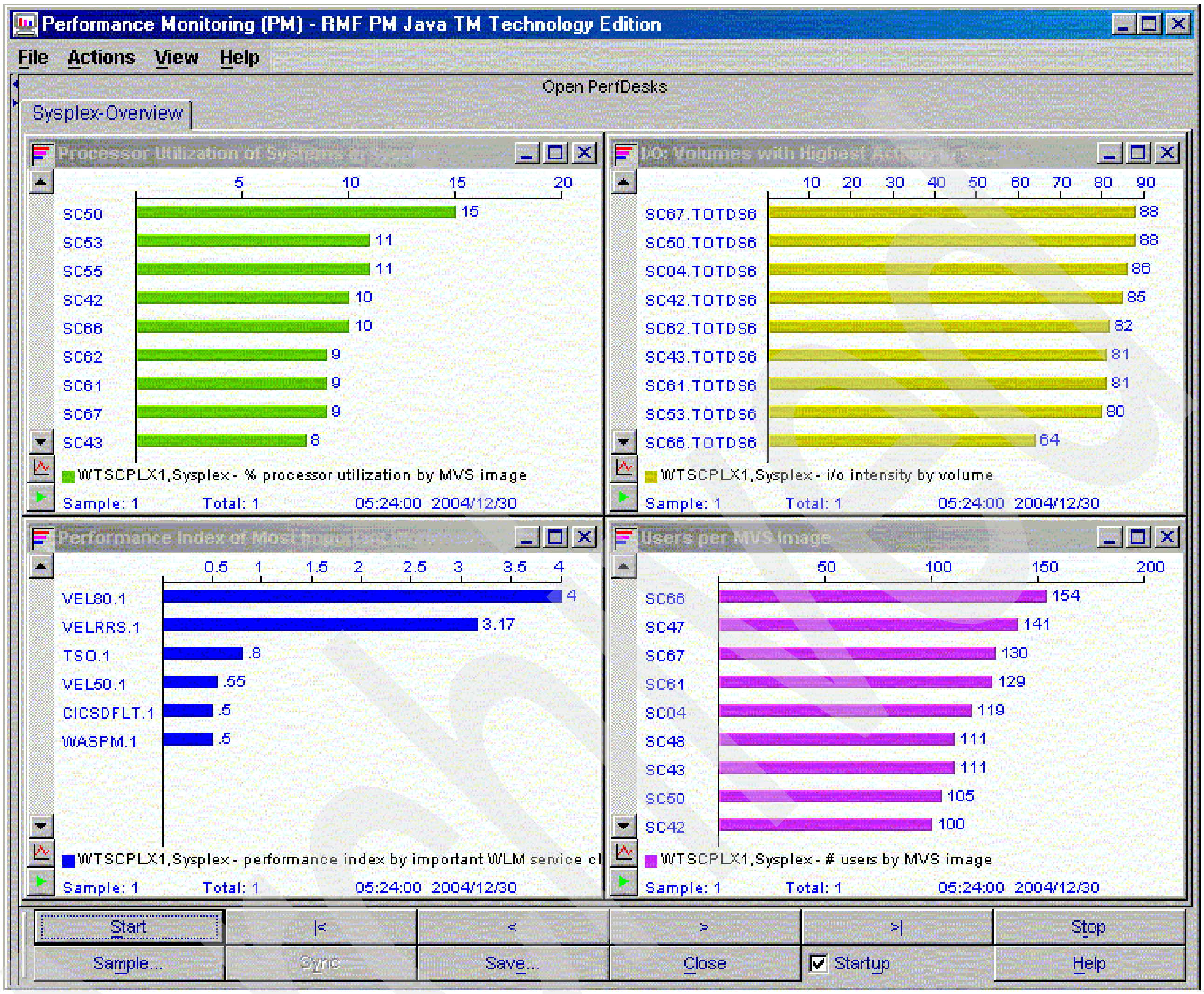

Performance Monitoring (PM) - RMF PM Java TM Technology Edition
File
Actions
View
Help
Open PerfDesks
Sysplex-Overview
Processor Utilization of Systems in Sysplex
5 10 15 20
SC50 15
SC53 11
SC55 11
SC42 10
SC66 10
SC62 9
SC61 9
SC67 9
SC43 8
WTSCPLX1,Sysplex - % processor utilization by MVS image
Sample: 1 Total: 1 05:24:00 2004/12/30
I/O Volumes with Highest Activity
10 20 30 40 50 60 70 80 90
SC67.TOTDS6 88
SC50.TOTDS6 88
SC04.TOTDS6 86
SC42.TOTDS6 85
SC62.TOTDS6 82
SC43.TOTDS6 81
SC61.TOTDS6 81
SC53.TOTDS6 80
SC66.TOTDS6 64
WTSCPLX1,Sysplex - i/o intensity by volume
Sample: 1 Total: 1 05:24:00 2004/12/30
Performance Index of Most Important ...
0.5 1 1.5 2 2.5 3 3.5 4
VEL80.1 4
VELRRS.1 3.17
TSO.1 .8
VEL50.1 .55
CICSDFLT.1 .5
WASPM.1 .5
WTSCPLX1,Sysplex - performance index by important WLM service cl
Sample: 1 Total: 1 05:24:00 2004/12/30
Users per MVS Image
50 100 150 200
SC66 154
SC47 141
SC67 130
SC61 129
SC04 119
SC48 111
SC43 111
SC50 105
SC42 100
WTSCPLX1,Sysplex - # users by MVS image
Sample: 1 Total: 1 05:24:00 2004/12/30
Start
|<
<
>
>|
Stop
Sample...
Sync
Save...
Close
☑ Startup
Help

Figure 2-31 Sysplex Overview PerfDesk

Actions are available through the menu bar, but also often via context menus of the items.

Resource structure

After you have successfully established connection to your system, you can browse the resource structure of your sysplex in the resource view, as shown in Figure 2-32 on page 81.

RMF PM represents your computing system as a composition of resources with associated metrics and attributes.

The following list shows the resource tree structure for z/OS:

Sysplex

- MVS Image
  - I/O-Subsystem
    - Storage Subsystem (SSID)
    - Logical Control Unit (LCU)
    - Channel
    - Volume
  - Processor
  - Storage
    - Central Storage
    - CSA
    - ECSA
    - SQA
    - ESQ
    - Expanded Storage
    - Auxiliary Storage
  - Enqueue
  - Operator
  - SW-Subsystem
    - JES
    - HSM
    - XCF
- Coupling Facility
  - Coupling Facility Structure
- CPC
  - LPAR

The following list shows the resource tree structure for Linux:

Linux image

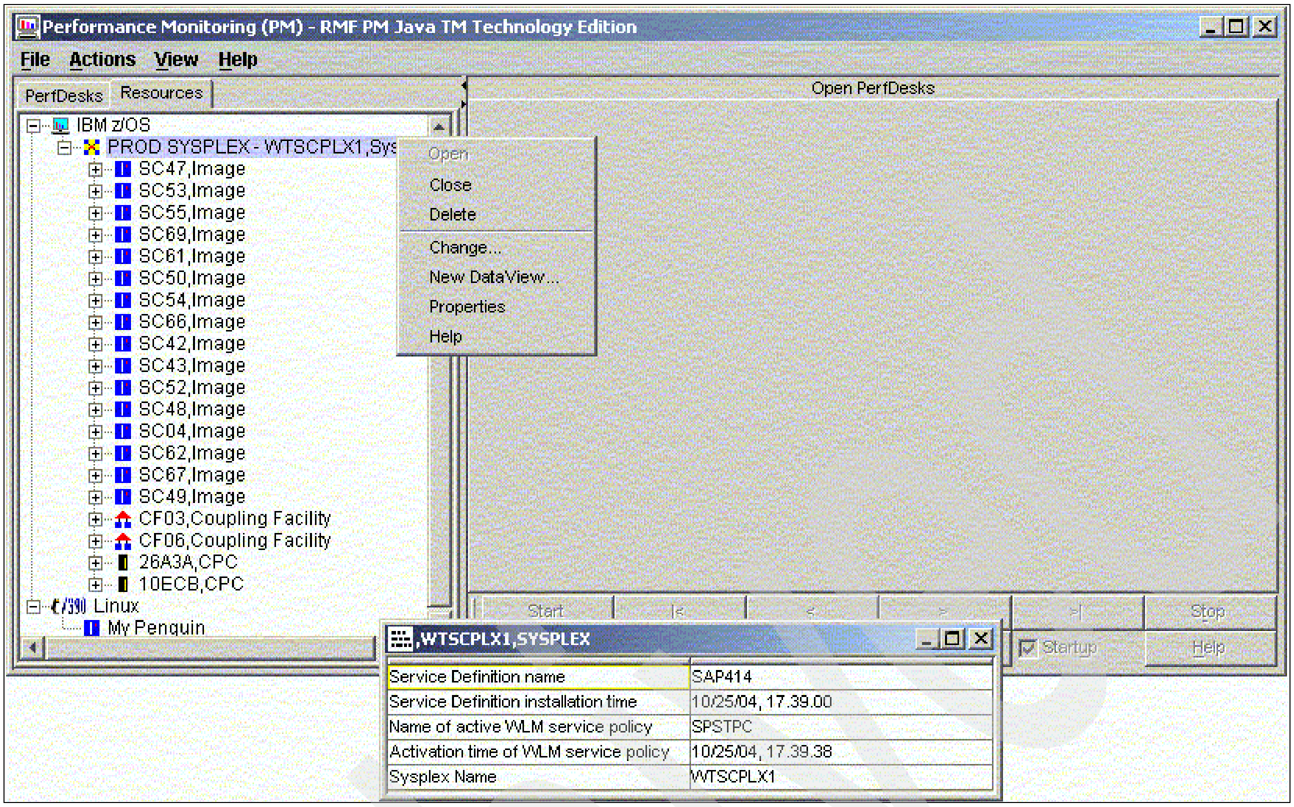
- Memory
- Network
- CPU
- Filesystem

Resource attributes

To get attributes for a resource, you have to select the resource, get the context menu, and select the Properties menu item. A new window comes up and displays the attributes for the resource. For example, in Figure 2-32 on page 81, the attributes for the selected sysplex PROD resource are displayed.

As sysplex attributes, we see:

- The active service definition and the installation time
- The active WLM policy name and installation time
- The sysplex name



Service Definition name	SAP414
Service Definition installation time	10/25/04, 17.39.00
Name of active WLM service policy	SPSTPC
Activation time of WLM service policy	10/25/04, 17.39.38
Sysplex Name	WTSCPLX1

Figure 2-32 Attributes of Sysplex Resource

PerfDesks and DataViews

After we have explored our system, we are now interested in performance data. Typically, each resource is associated with various metrics. A metric can either consist of exactly one value that is tied to one resource (single metric) or a list of name/value pairs (list metrics). These metrics are displayed as bars in DataViews, and one DataView can display up to nine metrics.

The PerfDesk contains the DataViews. You can save the PerfDesk so that the definitions are available in your next RMF PM session. PerfDesks are saved in the PerfDesk folders. The PerfDesk Folders enable you to structure your PerfDesks.

You can import and export PerfDesks to share your definitions with other colleagues who are also using RMF PM.

User scenario

Now we show the functionality of RMF PM in a user scenario where we are going to investigate a problem. We have complaints from users that their jobs are not performing very well. There are some jobs delayed in our sysplex, and we have to find out which jobs are delayed, and why.

Creating a DataView

First we create a PerfDesk, using **File** → **New** → **PerfDesk**. In the PerfDesk Dialog, we specify the PerfDesk name, `Workload Check`. Additionally, we want to save the PerfDesk in our own PerfDesk Folder, so we create one by clicking **Create folder** and name it `Wkl Folder`.

Now we have created the PerfDesk Workload Check, and it is stored in the PerfDesk Folder Wkl Folder.

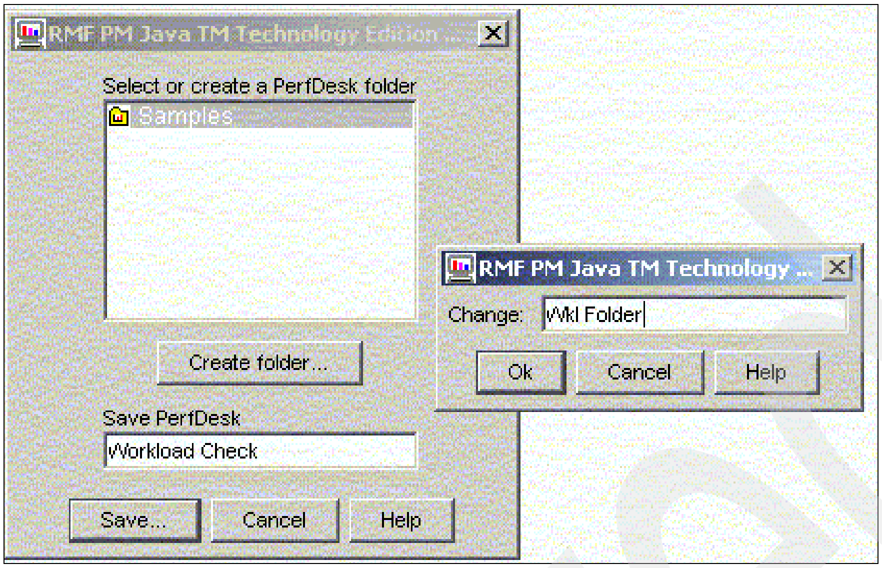
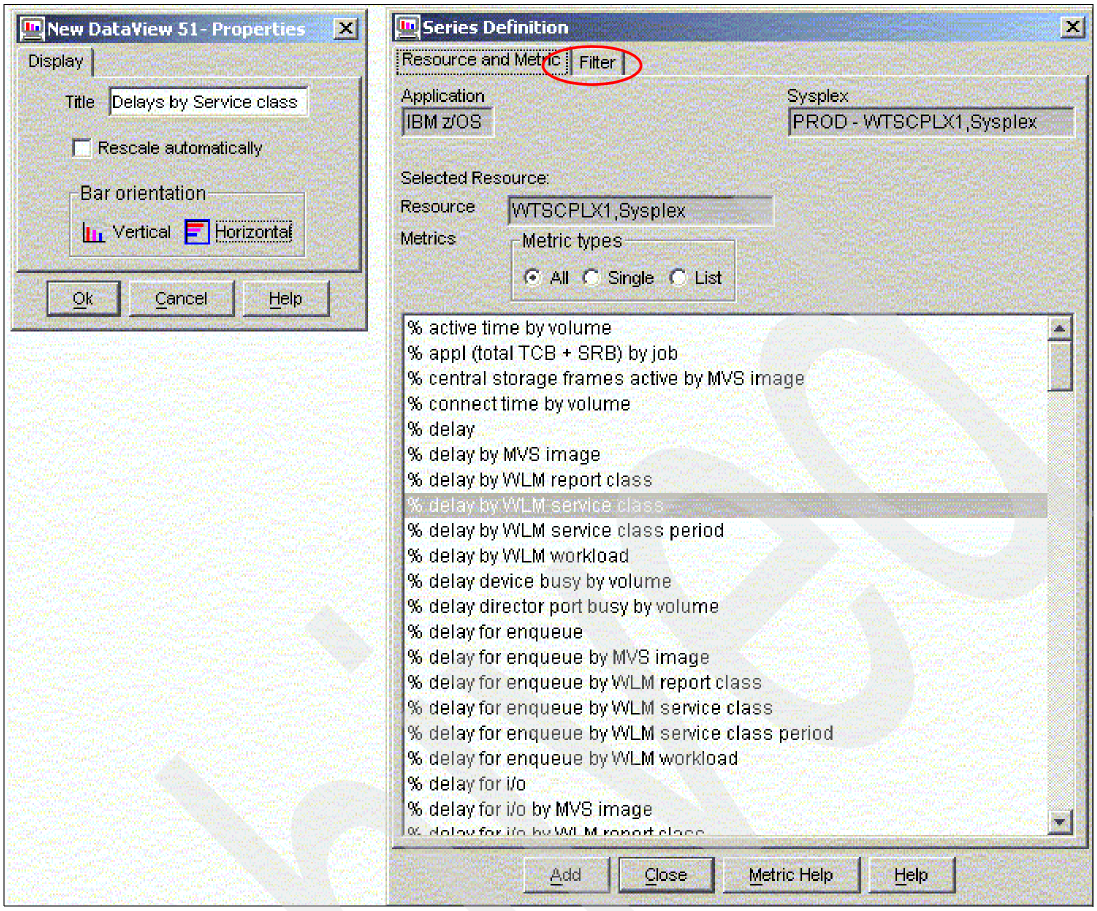

RMF PM Java TM Technology Edition ...
Select or create a PerfDesk folder
Samples
Create folder...
Save PerfDesk
Workload Check
Save...
Cancel
Help
RMF PM Java TM Technology ...
Change:
Wkl Folder
Ok
Cancel
Help

Figure 2-33 Create PerfDesk and PerfDesk Folder

We want to look for delays in the sysplex, so we create a DataView that helps us find where the delays are in our Sysplex. The list metric `% delay by WLM service class` gives us this information on a sysplex level. To create a DataView, we select the resource **Sysplex PROD** in the resource view and select **File** → **New** → **DataView**, or we select the context menu of the selected resource and select the **DataView** menu item.

The DataView Properties dialog is displayed, as shown in Figure 2-34. There we specify the title of the DataView. We have the option to display the data as vertical or horizontal bars. Vertical bars represent single metrics better, and horizontal bars are better for displaying list metrics. So we choose to display our data (list metrics) as horizontal bars. We click **OK** to get the Series Definition dialog.

On the Series Definition dialog, we choose the metric `% delay by WLM service class`. We click **Add** to add the selected metric to the DataView. While it is possible to add more metrics to a DataView, we focus on this metric. To get more information about a selected metric, click **Metric Help**. Depending on the metric type, additional tabs are available where we can specify a Workscope, a filter, or both. For our selected metric, we only can specify a filter for the service class name. But since we don't know where to focus, we do not set a filter. We click **Close** to finish our selection.



Title: Delays by Service class

□ Rescale automatically

Bar orientation: Vertical [✓ Horizontal]

Application: IBM z/OS

Sysplex: PROD - WTSCPLX1,Sysplex

Selected Resource:

Resource: WTSCPLX1,Sysplex

Metrics — Metric types: ● All ○ Single ○ List

% active time by volume
% appl (total TCB + SRB) by job
% central storage frames active by MVS image
% connect time by volume
% delay
% delay by MVS image
% delay by WLM report class
% delay by WLM service class
% delay by WLM service class period
% delay by WLM workload
% delay device busy by volume
% delay director port busy by volume
% delay for enqueue
% delay for enqueue by MVS image
% delay for enqueue by WLM report class
% delay for enqueue by WLM service class
% delay for enqueue by WLM service class period
% delay for enqueue by WLM workload
% delay for i/o
% delay for i/o by MVS image
% delay for i/o by WLM report class

Figure 2-34 Create a DataView

Now we have successfully created our first DataView. To start data gathering for the DataView, we click **Start** on the lower button bar or click the **Start/Stop** button in the DataView. In Figure 2-35, we see the gathered data for our metric in the DataView.

- Result: We see a delay of 20% for the service class VEL50 at the time 09:25:00 (displayed below the graph). This service class VEL50 is our candidate for further investigation.

Tip: You can maximize the Open PerfDesk view by clicking the arrows on the bar that separates the two views.

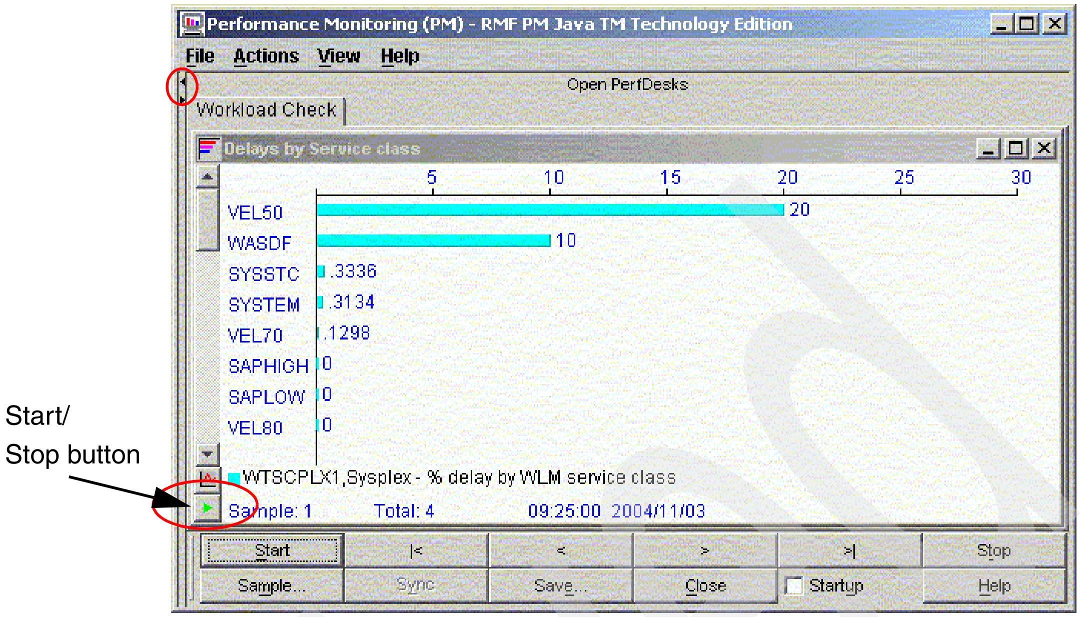


Figure 2-35 %delay by WLM service class

Now we want to see whether the load is focused on one system in our sysplex. Therefore, we select again our sysplex resource and create a new DataView called "`% delay by image`."

We want to look for delays in the sysplex of the service class VEL50. The metric `% delay by MVS image` on sysplex level shows that. So we select the sysplex resource and create a new DataView. On the Series Definition dialog, we select the metric **% delay by MVS image**. For this metric, we have the option to define a specific workscope and additionally a filter, as shown in Figure 2-36.

On the Workscope tab of the Series Definition dialog, we specify that we want to focus on the service class VEL50.

Depending on the metric, you can specify the following workscope:

- Job
- Workload
- Service class and period
- Report class
- Total - no special workscope is used

If you specify a workscope, you could either enter the workscope name or select the workscope from the list. To refresh the workscope list, click **Refresh**.

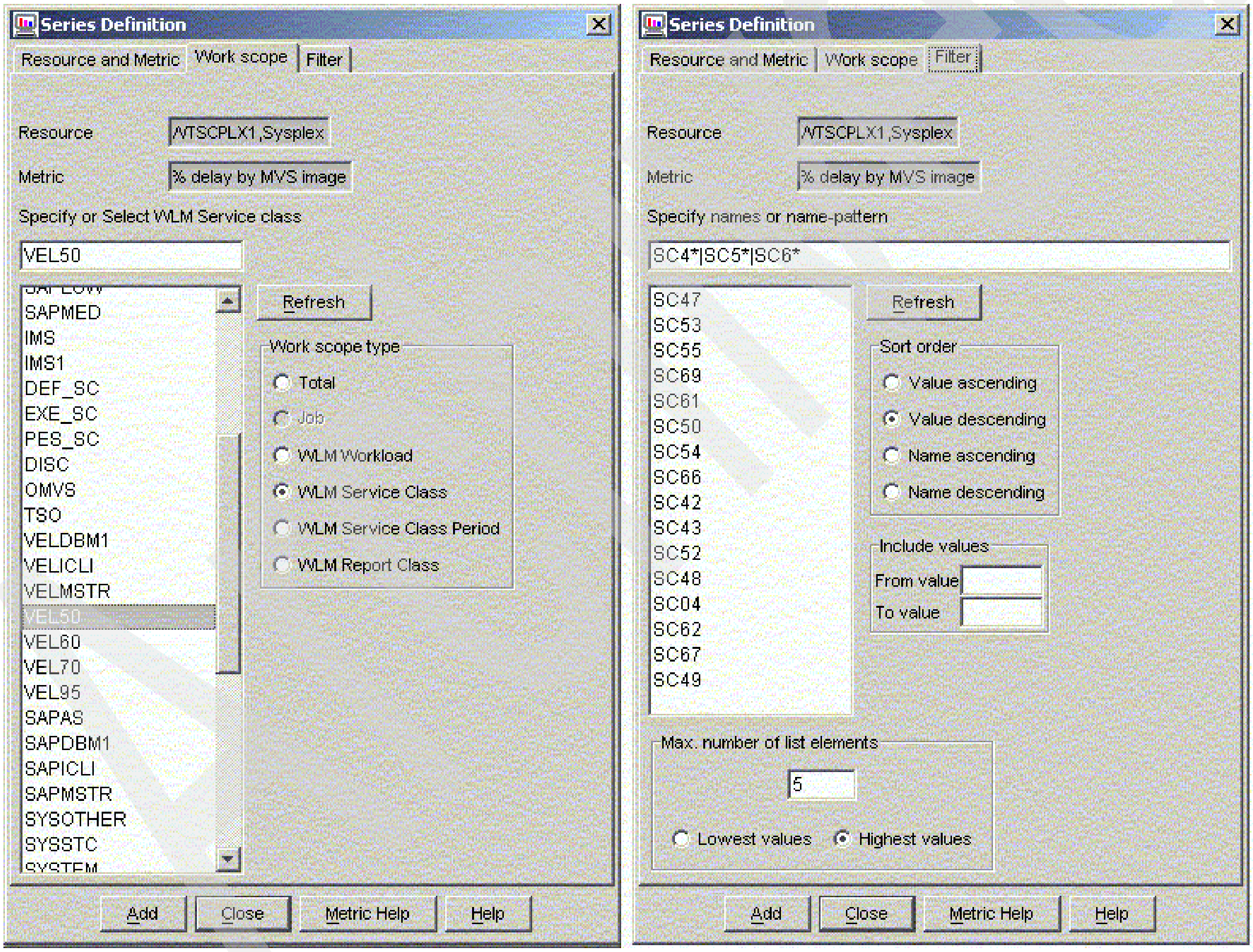
On the Filter tab, we can set several filter options. We want to see only the top five, so that we can focus specifically on them; we specify the maximum number of list elements to five. Additionally, we want to limit the view to our production system, which uses names SC4*, SC5*, and SC6*, so we specify the filter **`SC4*|SC5*|SC6*`**.

Depending on the metric, you can specify the following filter options:

- Name pattern - Optionally, one or more name patterns in the form of a simple expression used as a filter, with:
  - ? - one character
  - * - zero, one, or several characters

If the name contains an asterisk (*) for example, *MASTER*, you have to precede each * by a back slash. To refresh the filter list, click **Refresh**.

- Value bounds – optionally, you can define an upper bound and a lower bound for the values to be displayed.
- Sort order – allows you to order the values in the list of name-value pairs of the list metrics:
  - Value ascending
  - Value descending
  - Name ascending
  - Name descending
- Max. numbers of list elements – for restricting the length of the list of name-value pairs of list metrics:
  - n Highest values
  - n Lowest values



Series Definition — Work scope

Resource: /WTSCPLX1,Sysplex

Metric: % delay by MVS image

Specify or Select WLM Service class: VEL50

Work scope type: ● WLM Service Class

Series Definition — Filter

Resource: /WTSCPLX1,Sysplex

Metric: % delay by MVS image

Specify names or name-pattern: SC4*|SC5*|SC6*

Sort order: ● Value descending

Max. number of list elements: 5 ● Highest values

Figure 2-36 Setting Workscope and Filter option

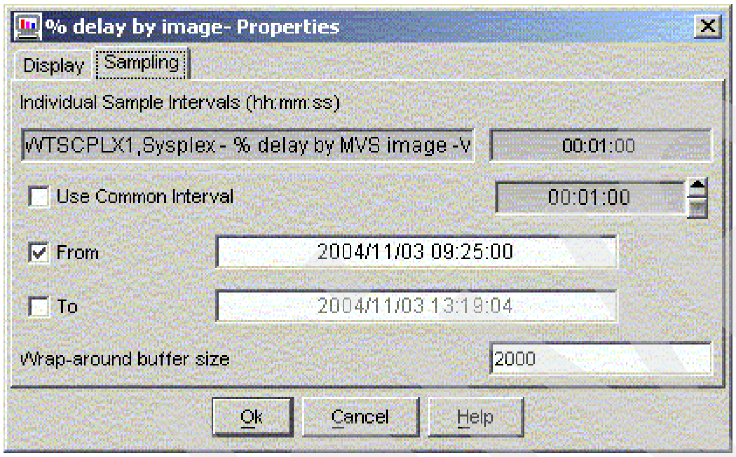
We click **Done** to finish the customization.

Tip: To arrange the DataViews select **View** → **Tile DataViews**.

Sampling historical data

Well, we saw at time stamp 09:25:00 that service class VEL50 has a delay of 20%. But now enough time has passed that we have created our second DataView. How can we get data for the metric `% delay by MVS image` in the past? We use the sampling function of RMF PM to recall historical data. To do so, the data gatherer, either the DDS or rmfpms, must have the historical data available in storage or preallocated data sets.

Selecting **File** → **Change Settings** → **DataView**, we get the properties of the selected DataView, and we can specify the timeframe to sample data. We specify that we want to gather data beginning at the time stamp 9:25:00, as shown in Figure 2-37. We click **OK** to finish our definition.



Individual Sample Intervals (hh:mm:ss)

WTSCPLX1,Sysplex - % delay by MVS image -V 00:01:00

□ Use Common Interval 00:01:00

☑ From 2004/11/03 09:25:00

□ To 2004/11/03 13:19:04

Wrap-around buffer size 2000

Figure 2-37 Sample historical data

After we have specified the timeframe for historical data sampling, we start the data gathering for the DataView.

- Result: In Figure 2-38, we see that the main delays (37%) occur on System SC48. The other systems show zero delays, so we have to focus now on System SC48.

After we have started sampling of our second DataView, the reporting data is out of sync. To synchronize data reporting of all DataViews in the PerfDesk to the time of the selected DataView, click **Sync** in the lower button bar.

Startup PerfDesk

To save the changes to the PerfDesk, either click **Save** in the lower button bar or click **File** → **Save PerfDesk**.

We could also specify the PerfDesk as Startup PerfDesk, by enabling the Startup checkbox on the lower button bar. A Startup PerfDesk is automatically started each time you start RMF PM.

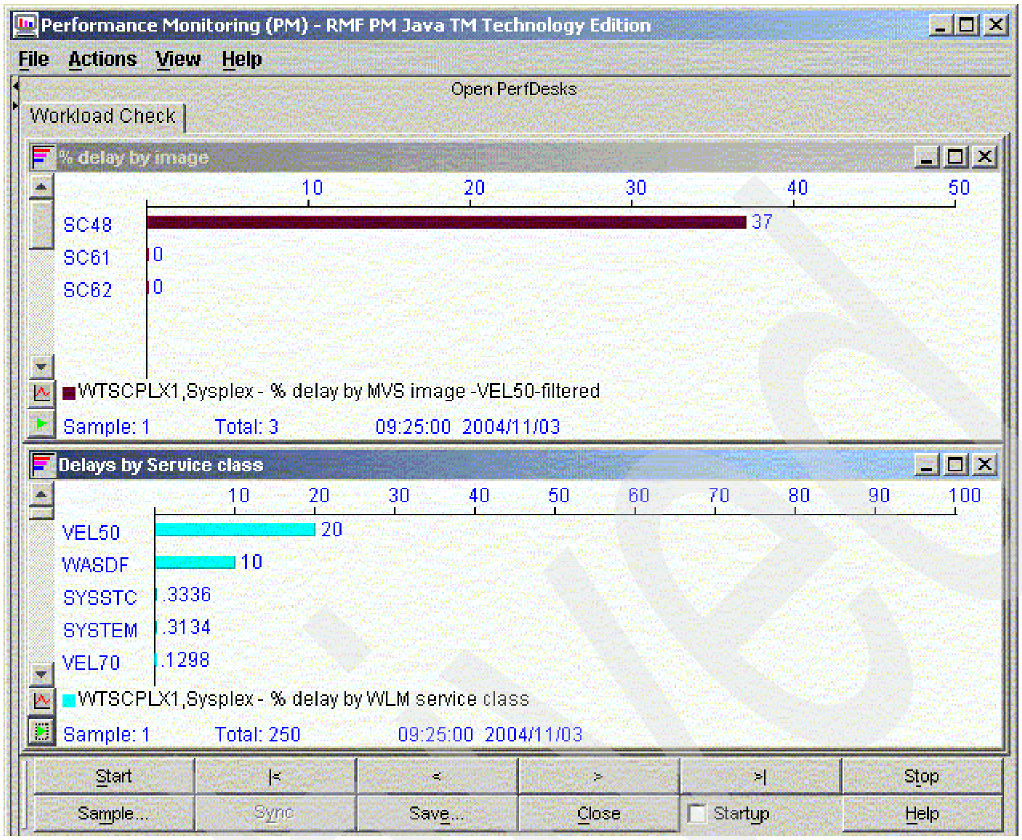
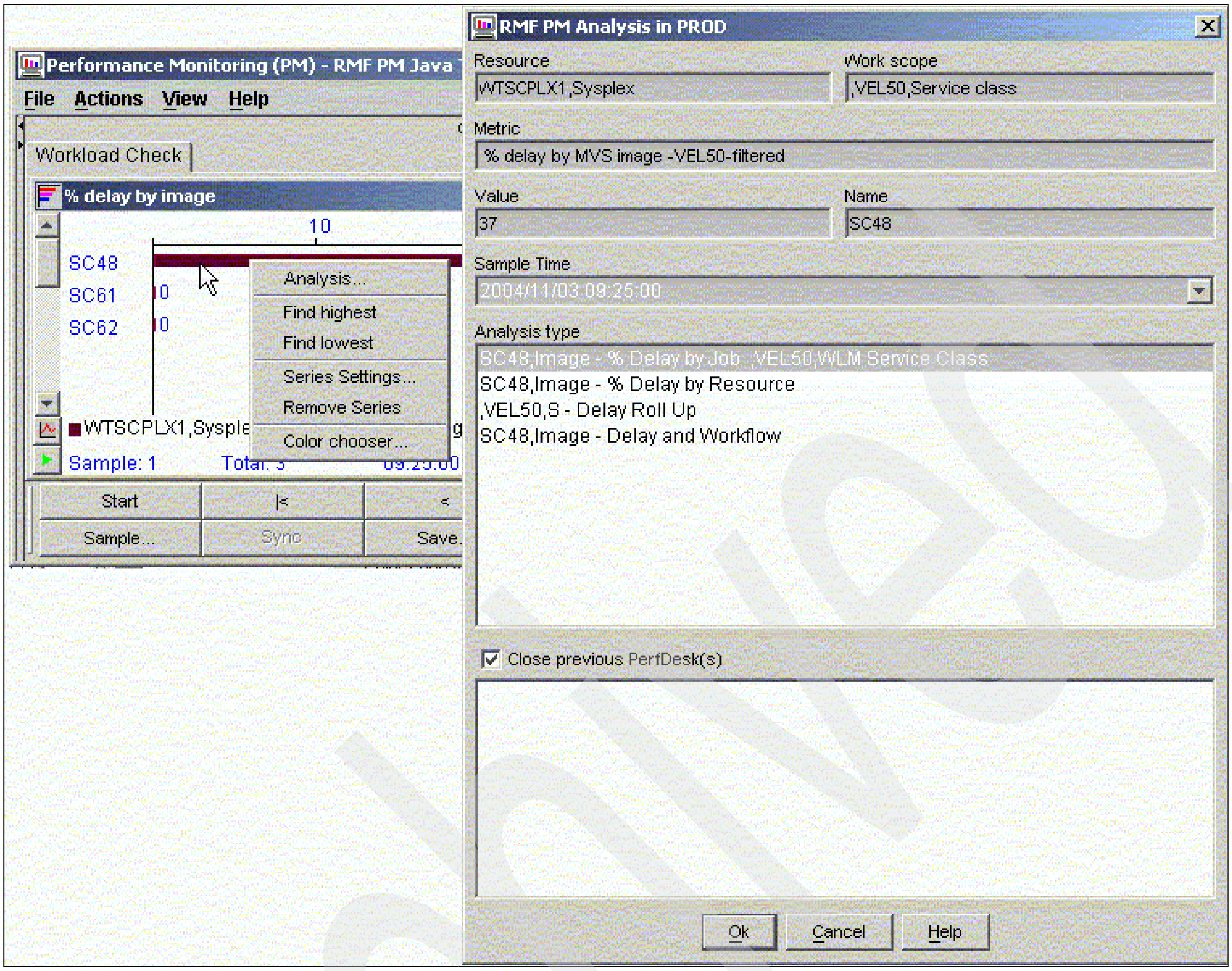


Figure 2-38 Historical data sampling

PerfDesk analysis

Now we know that the problem is on system SC48 where service class VEL50 encounters delays, as we see in Figure 2-38. But we need a deeper drilldown, and for that RMF PM offers powerful advanced analysis functionality.

To analyze the problem at 09:25:00 of system SC48 where we have 37% delay, we have to get the context menu of the bar showing 37% delay in the DataView and select the menu item **Analysis**, as shown in Figure 2-39. The analysis dialog offers several predefined analysis PerfDesks. We select the Job drill down to see the `% delay by Job` on system SC48 for WLM service class VEL50 by job and click **OK**.



Resource: WTSCPLX1,Sysplex

Work scope: ,VEL50,Service class

Metric: % delay by MVS image -VEL50-filtered

Value: 37

Name: SC48

Sample Time: 2004/11/03 09:25:00

Analysis type:

SC48,Image - % Delay by Job ,VEL50,WLM Service Class
SC48,Image - % Delay by Resource
,VEL50,S - Delay Roll Up
SC48,Image - Delay and Workflow

☑ Close previous PerfDesk(s)

Figure 2-39 RMF PM advanced analysis function

- Result:

  On the analysis PerfDesk in Figure 2-40, we see the jobs that are delayed: JESJOP3, JESJOP1, JESJOP2, JESJOP6 and JESJOP9 are heavily delayed, between 29% and 40%.

  Possible reasons for the delay are:

  – I/O delays
  – Queuing delays
  – Memory delays
  – CPU delays
  – Subsystem delays
  – Operator delays

.

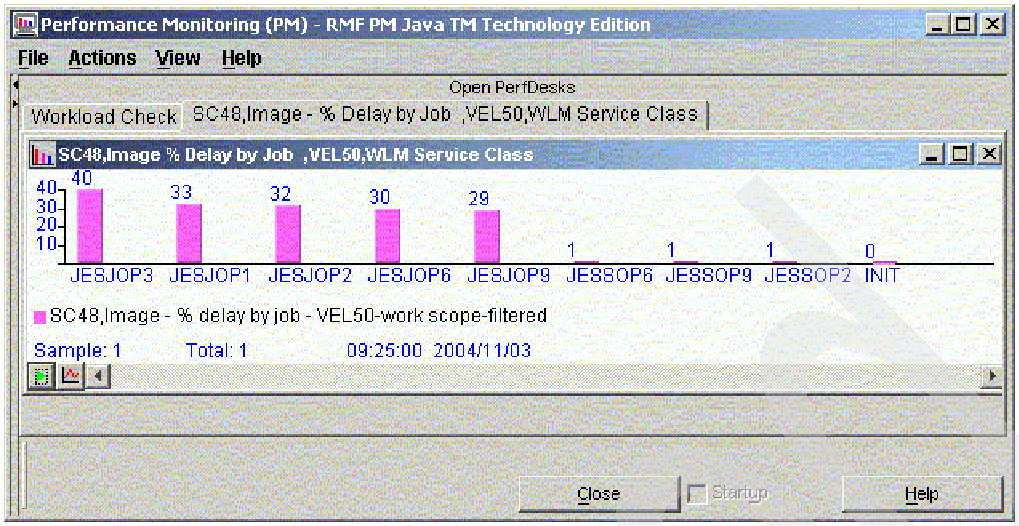

Performance Monitoring (PM) - RMF PM Java TM Technology Edition
File Actions View Help
Open PerfDesks
Workload Check
SC48,Image - % Delay by Job ,VEL50,WLM Service Class
SC48,Image % Delay by Job ,VEL50,WLM Service Class
40
30
20
10
40
33
32
30
29
1
1
1
0
JESJOP3 JESJOP1 JESJOP2 JESJOP6 JESJOP9 JESSOP6 JESSOP9 JESSOP2 INIT
SC48,Image - % delay by job - VEL50-work scope-filtered
Sample: 1
Total: 1
09:25:00 2004/11/03
Close
Startup
Help

Figure 2-40 Analysis for delays on SC48

So we start over and do the analysis for the job JESJOP3 with the 40% delay at time stamp 09:25:00. As the next analysis step, we choose the `% Delay by Resource` analysis PerfDesk.

- Result:

  In Figure 2-41, we see the main reason for the delay – at 9:25:00 there is a 40% processor delay.

  We continue to gather samples to verify that this is consistent through most of the intervals.

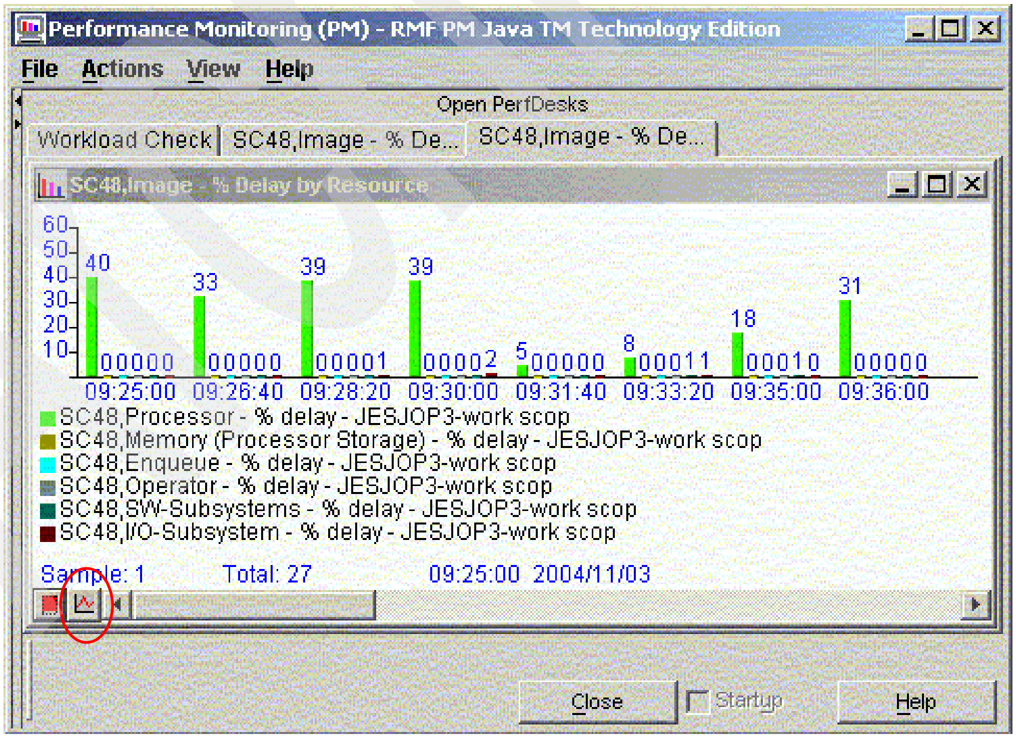

Performance Monitoring (PM) - RMF PM Java TM Technology Edition
File Actions View Help
Open PerfDesks
Workload Check
SC48,Image - % De...
SC48,Image - % De...
SC48,Image - % Delay by Resource
60
50
40
30
20
10
40
33
39
39
5
8
18
31
00000
00000
00001
00002
00000
00011
00010
00000
09:25:00 09:26:40 09:28:20 09:30:00 09:31:40 09:33:20 09:35:00 09:36:00
🟩 SC48,Processor - % delay - JESJOP3-work scop
🟨 SC48,Memory (Processor Storage) - % delay - JESJOP3-work scop
🟦 SC48,Enqueue - % delay - JESJOP3-work scop
⬛ SC48,Operator - % delay - JESJOP3-work scop
⬛ SC48,SW-Subsystems - % delay - JESJOP3-work scop
🟥 SC48,I/O-Subsystem - % delay - JESJOP3-work scop
Sample: 1
Total: 27
09:25:00 2004/11/03
Close
Startup
Help

Figure 2-41 Analysis of Job JESOP3

Further samples show the same result, that we have heavy processor delays. It looks like this is a processor constraint. Other delays like memory, enqueue, I/O, Operator, or Subsystems are minimal or close to zero.

As a result of the analysis that we have performed with RMF PM, we know about the processor constraint in system SC48. Now it is a matter of understanding if we can solve the constraint by rearranging the workload definitions.

Tip: To identify highs and lows in a DataView, select **Actions** → **Find highest values in Series** or **Actions** → **Find lowest values in Series**.

Exporting data to spreadsheets

You can export RMF PM data into spreadsheets to create charts in an easy way.

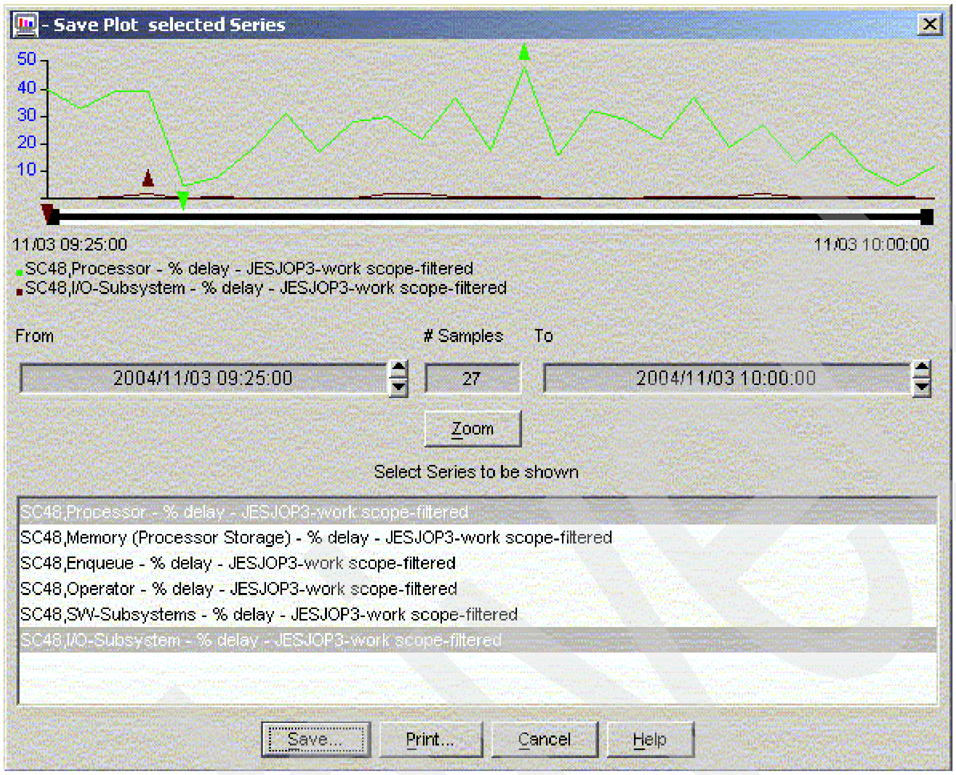
RMF PM can save the data into spreadsheet format for further processing. To save the data of a DataView use the **Plot/Save Series** menu item in the DataView's context menu or click the **Plot/Save** button, as shown in Figure 2-41.

Within the Save/Plot selected Series dialog, you can view Series plots and optionally save a variety of Series of a DataView.

You can select:

- One Series from a list of Series
- Multiple Series if the Series contain Single-Values
- Zoom a range (from one time stamp to another time stamp)
- Series for one or more value names (on Value-Lists only) selectable from two Value Name lists, sorted by highest maximum values and by highest average value in descending order.

In Figure 2-42, we selected a subset of the metrics. Additionally, you see the peaks and lows of a metric. To export the data into spreadsheet format, click **Save**.



🟩 SC48,Processor - % delay - JESJOP3-work scope-filtered
🟥 SC48,I/O-Subsystem - % delay - JESJOP3-work scope-filtered

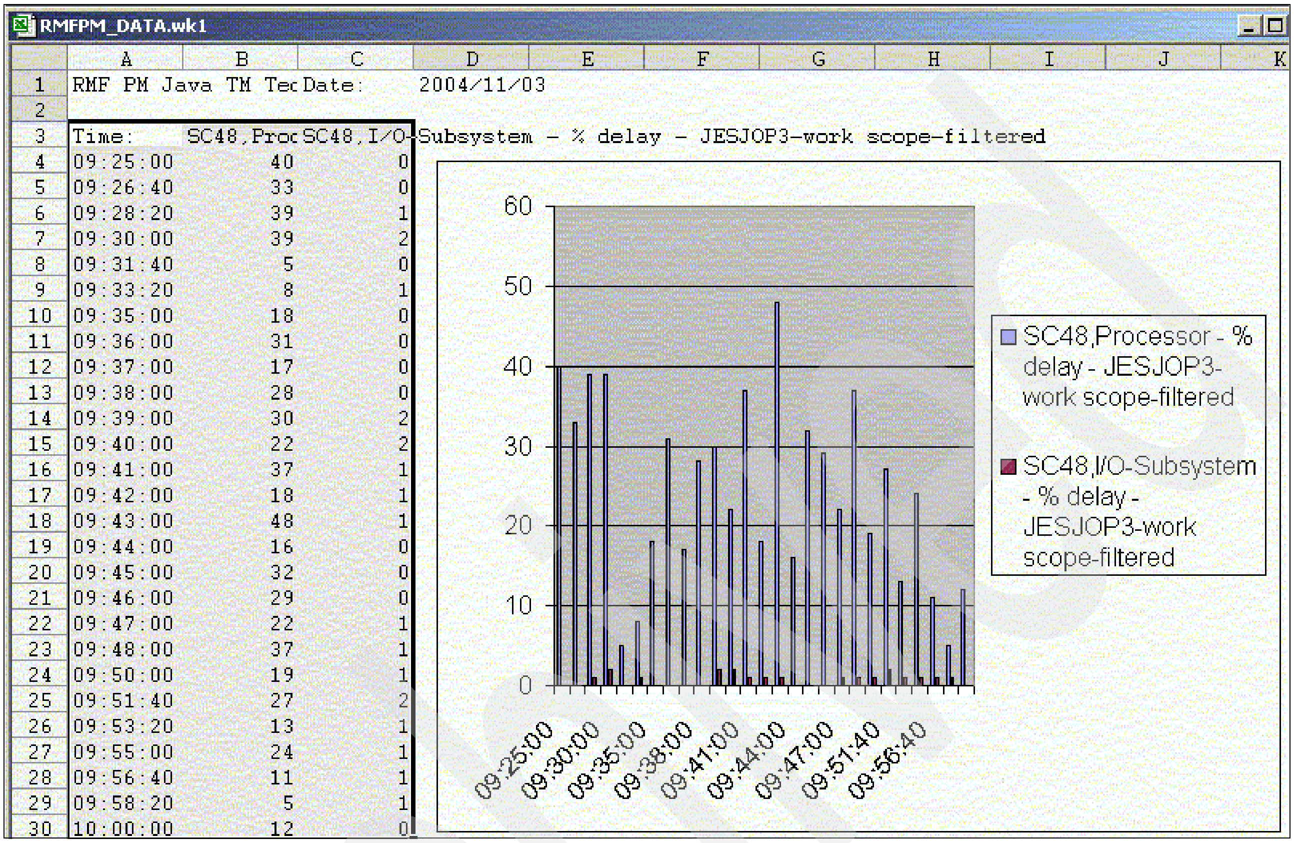
From: 2004/11/03 09:25:00
Samples: 27
To: 2004/11/03 10:00:00

Select Series to be shown

SC48,Processor - % delay - JESJOP3-work scope-filtered
SC48,Memory (Processor Storage) - % delay - JESJOP3-work scope-filtered
SC48,Enqueue - % delay - JESJOP3-work scope-filtered
SC48,Operator - % delay - JESJOP3-work scope-filtered
SC48,SW-Subsystems - % delay - JESJOP3-work scope-filtered
SC48,I/O-Subsystem - % delay - JESJOP3-work scope-filtered

Figure 2-42 Save RMF PM into Lotus 1-2-3® WK1 Format

After we have saved the data, it is easy to create a chart, for example, using Microsoft Excel. We load the saved data into Microsoft Excel and select the data range with the header lines (A3 to C30). Using **Insert** → **Chart**, we create a chart in an easy way as in Figure 2-43.



RMFPM_DATA.wk1

	A	B	C	D
1	RMF PM Java TM Tec		Date:	2004/11/03
2				
3	Time:	SC48,Proc	SC48,I/O-Subsystem – % delay – JESJOP3-work scope-filtered	
4	09:25:00	40	0	
5	09:26:40	33	0	
6	09:28:20	39	1	
7	09:30:00	39	2	
8	09:31:40	5	0	
9	09:33:20	8	1	
10	09:35:00	18	0	
11	09:36:00	31	0	
12	09:37:00	17	0	
13	09:38:00	28	0	
14	09:39:00	30	2	
15	09:40:00	22	2	
16	09:41:00	37	1	
17	09:42:00	18	1	
18	09:43:00	48	1	
19	09:44:00	16	0	
20	09:45:00	32	0	
21	09:46:00	29	0	
22	09:47:00	22	1	
23	09:48:00	37	1	
24	09:50:00	19	1	
25	09:51:40	27	2	
26	09:53:20	13	1	
27	09:55:00	24	1	
28	09:56:40	11	1	
29	09:58:20	5	1	
30	10:00:00	12	0	

E
F
G
H
I
J
K
60
50
40
30
20
10
0
09:25:00
09:30:00
09:35:00
09:38:00
09:41:00
09:44:00
09:47:00
09:51:40
09:56:40
SC48,Processor - % delay - JESJOP3-work scope-filtered
SC48,I/O-Subsystem - % delay - JESJOP3-work scope-filtered

Figure 2-43 RMF PM data in a spreadsheet

Transferring PerfDesk configurations

RMF PM customization is saved on the workstation; for example, all the PerfDesk definitions can be saved. If you need to work at another workstation, and you need to use the PerfDesks and DataViews that you have created to monitor your system, you can transfer them easily to another workstation.

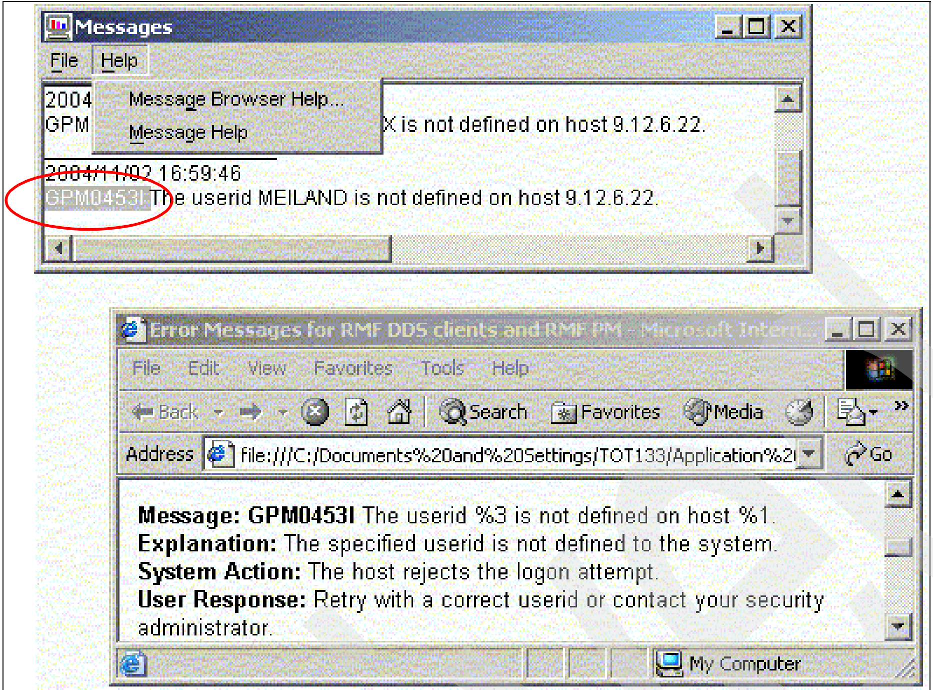
RMF PM offers import and export functions to save the configuration data into a file. To import or export a PerfDesk, select **File** → **Export PerfDesks** and **File** → **Import PerfDesks**. To import or export PerfDesk Folders, use the context menu **Import Folder** or **Export Folder** of the PerfDesk Folder in the PerfDesks view.

2.7.2 Troubleshooting

Here we discuss how to efficiently use the RMF PM message browser, and how to get additional information when you encounter an error with RMF PM.

RMF PM message browser

If an error occurs during data retrieval or data connection, the message browser pops up and displays a message to the user. Each message has a time stamp and a message ID. To get additional help regarding the message, select the message ID in the message window, and click **Help** → **Message Help**. The message explanation is displayed in a new browser window, as shown in Figure 2-44.



2004

GPM X is not defined on host 9.12.6.22.

2004/11/02 16:59:46
GPM0453I The userid MEILAND is not defined on host 9.12.6.22.

file:///C:/Documents%20and%20Settings/TOT133/Application%2

Message: GPM0453I The userid %3 is not defined on host %1.
Explanation: The specified userid is not defined to the system.
System Action: The host rejects the logon attempt.
User Response: Retry with a correct userid or contact your security administrator.

Figure 2-44 RMF PM Message browser

RMF PM log files

RMF PM writes several log files that supply additional information to the RMF change team when an error occurs.

The following log files are saved:

- gpmcom.err
- pm.log
- pm_err.log

They are stored in the private subdirectory directory of the RMF PM Application Data directory. For example, for user TOT133:

C:\Documents and Settings\TOT133\Application Data\RMF\RMF Performance Monitoring\private

Keep these log files available when you report an error.

2.8 RMF Web browser interface

Instead of using the RMF PM, you can also just access RMF data via a Web browser. The DDS and the data gatherer on Linux provide a Web front end to online performance data. Using a Web browser that can display XML documents with XSL style sheets (like Mozilla 1.4 or above, Netscape 7.0 or above, or Microsoft Internet Explorer 5.5 or above), you have access to more than 600 z/OS performance metrics.

You can monitor the resources of the corresponding system, and additionally save important metrics into your own performance overview. Every time you log on, you can switch to this saved performance overview.

Using the RMF data on demand, you can browse your configuration and display available performance metrics.

2.8.1 Using RMF Web browser interface

After you have customized the DDS on your z/OS host or rmfpms on your Linux image (see“RMF Distributed Data Server” on page 136, “RMF Linux data gatherer” on page 138), you are ready to use the RMF Web browser interface.

To export performance data from the RMF Web browser interface into spreadsheets, you need Microsoft Windows XP and Microsoft Excel 2002 or higher.

Note: Verify that the HTTP port (default 8803) defined during customization of the DDS or rmfpms is open for communication. Security applications like firewalls may also block this type of communication.

Welcome screen

The DDS behaves like an HTTP server. You can open your Web browser and simply type the URL:

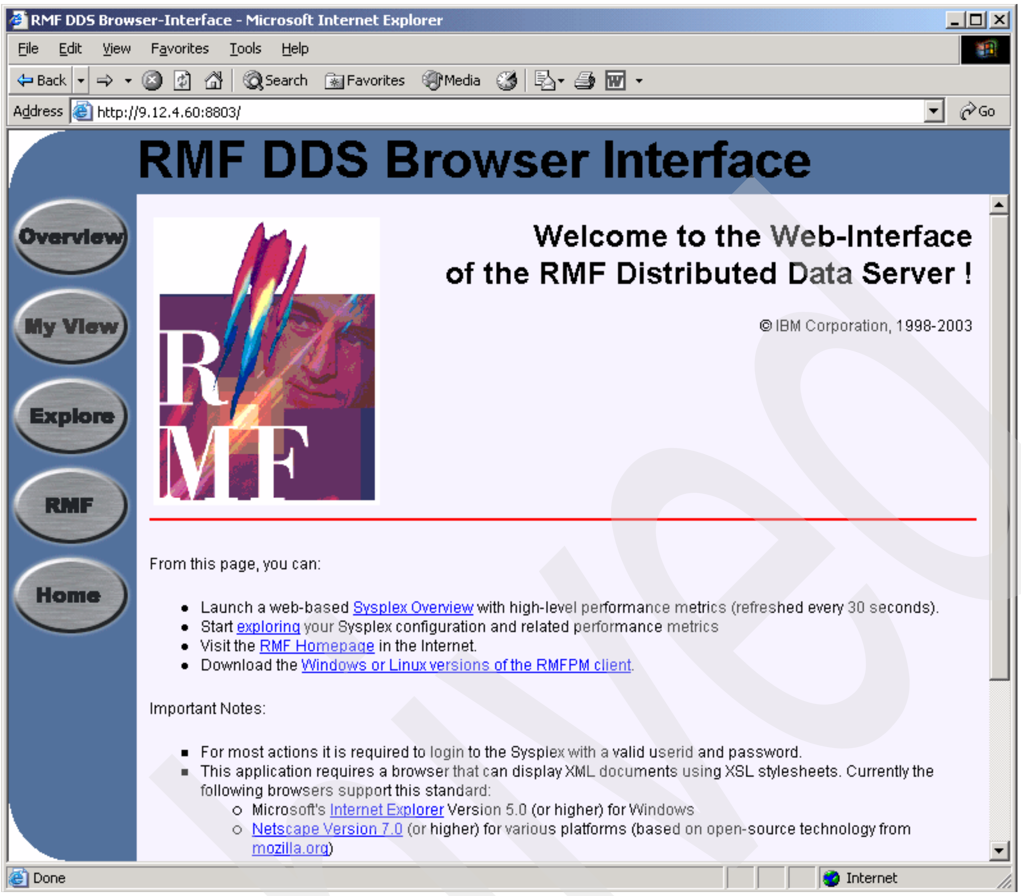
```
http://<hostname>:8803
```

where the *hostname* is the hostname or IP address of the host where the DDS or rmfpms is installed. After you have successfully connected to the rmfpms or DDS, the browser displays the Welcome screen, as in Figure 2-45.

The Welcome screen offers these functions:

- Overview - Shows you a health check overview of your system.
- My View - Views where you can see your preferred metrics.
- Explore - Explores the resources and performance metrics of your system.
- RMF - Leads you to the RMF Home page.
- Home - Leads you back to the welcome screen.

Depending on the customization of your DDS, you need your TSO user ID and password to log on to use Overview, My View, and the Explore functions. Ensure that you have sufficient access to RMF Performance data, described in “Controlling access to RMF data for the sysplex data services” on page 116.


RMF DDS Browser-Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://9.12.4.60:8803/
Go
Overview
My View
Explore
RMF
Home
Done
Internet

RMF DDS Browser Interface

Welcome to the Web-Interface of the RMF Distributed Data Server !

© IBM Corporation, 1998-2003

From this page, you can:

- Launch a web-based Sysplex Overview with high-level performance metrics (refreshed every 30 seconds).
- Start exploring your Sysplex configuration and related performance metrics
- Visit the RMF Homepage in the Internet.
- Download the Windows or Linux versions of the RMFPM client.

Important Notes:

- For most actions it is required to login to the Sysplex with a valid userid and password.
- This application requires a browser that can display XML documents using XSL stylesheets. Currently the following browsers support this standard:
  - Microsoft's Internet Explorer Version 5.0 (or higher) for Windows
  - Netscape Version 7.0 (or higher) for various platforms (based on open-source technology from mozilla.org)

Figure 2-45 DDS welcome screen

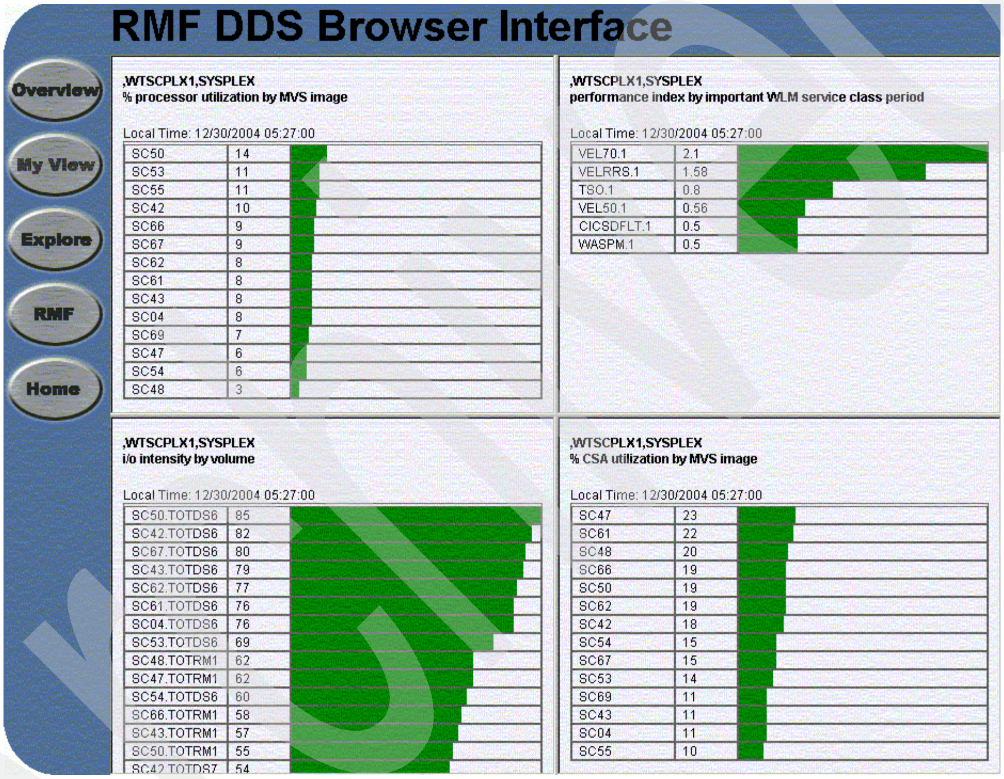
Performance overview

Click **Overview** to see the predefined high-level health check of your system.

Figure 2-46 shows the Overview for z/OS that displays the following perspectives and their associated metrics:

Processor	% processor utilization by MVS image
I/O	i/o intensity by volume
Storage	% CSA utilization by MVS image
WLM	performance index by WLM service class period

You cannot modify the predefined Overview. To create a view with your preferred set of metrics, use the function described in “My View function” on page 101.



,WTSCPLX1,SYSPLEX
% processor utilization by MVS image

Local Time: 12/30/2004 05:27:00

SC50	14
SC53	11
SC55	11
SC42	10
SC66	9
SC67	9
SC62	8
SC61	8
SC43	8
SC04	8
SC69	7
SC47	6
SC54	6
SC48	3

,WTSCPLX1,SYSPLEX
performance index by important WLM service class period

Local Time: 12/30/2004 05:27:00

VEL70.1	2.1
VELRRS.1	1.58
TSO.1	0.8
VEL50.1	0.56
CICSDFLT.1	0.5
WASPM.1	0.5

,WTSCPLX1,SYSPLEX
i/o intensity by volume

Local Time: 12/30/2004 05:27:00

SC50.TOTDS6	85
SC42.TOTDS6	82
SC67.TOTDS6	80
SC43.TOTDS6	79
SC62.TOTDS6	77
SC61.TOTDS6	76
SC04.TOTDS6	76
SC53.TOTDS6	69
SC48.TOTRM1	62
SC47.TOTRM1	62
SC54.TOTDS6	60
SC66.TOTRM1	58
SC43.TOTRM1	57
SC50.TOTRM1	55
SC42.TOTDS7	54

,WTSCPLX1,SYSPLEX
% CSA utilization by MVS image

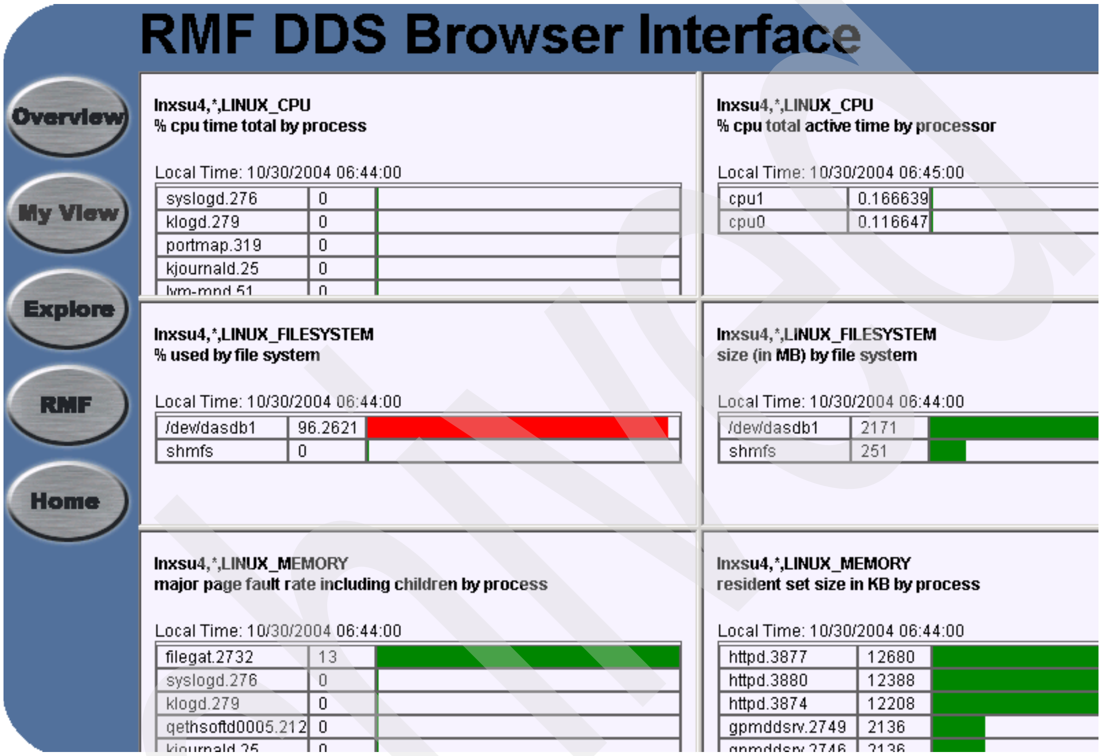
Local Time: 12/30/2004 05:27:00

SC47	23
SC61	22
SC48	20
SC66	19
SC50	19
SC62	19
SC42	18
SC54	15
SC67	15
SC53	14
SC69	11
SC43	11
SC04	11
SC55	10

Figure 2-46 Sysplex Overview

Figure 2-47 shows the predefined Overview for zSeries Linux that displays the following perspectives and their associated metrics:

Processor	CPU % time by process and total processor utilization
Filesystem	% used and size of file systems
Storage	Page fault rate and resident memory size by process
Apache HTTP Server	Apache HTTP Server access statistic



RMF DDS Browser Interface

lnxsu4,*,LINUX_CPU
% cpu time total by process

Local Time: 10/30/2004 06:44:00

syslogd.276	0	
klogd.279	0	
portmap.319	0	
kjournald.25	0	
lvm-mpd.51	0	

lnxsu4,*,LINUX_CPU
% cpu total active time by processor

Local Time: 10/30/2004 06:45:00

cpu1	0.166639	
cpu0	0.116647	

lnxsu4,*,LINUX_FILESYSTEM
% used by file system

Local Time: 10/30/2004 06:44:00

/dev/dasdb1	96.2621	🟥
shmfs	0	

lnxsu4,*,LINUX_FILESYSTEM
size (in MB) by file system

Local Time: 10/30/2004 06:44:00

/dev/dasdb1	2171	🟩
shmfs	251	🟩

lnxsu4,*,LINUX_MEMORY
major page fault rate including children by process

Local Time: 10/30/2004 06:44:00

filegat.2732	13	🟩
syslogd.276	0	
klogd.279	0	
gethsoftd0005.212	0	
kjournald.25	0	

lnxsu4,*,LINUX_MEMORY
resident set size in KB by process

Local Time: 10/30/2004 06:44:00

httpd.3877	12680	🟩
httpd.3880	12388	🟩
httpd.3874	12208	🟩
gpmddsrv.2749	2136	🟩
gpmddsrv.2746	2136	🟩

Figure 2-47 Linux image Overview

Explore your system

DDS represents your computing system as a composition of resources with associated metrics and attributes.

The following list shows the resources tree structure for z/OS:

Sysplex

- MVS Image
  - I/O-Subsystem
    - Storage Subsystem (SSID)
    - Logical Control Unit (LCU)
    - Channel
    - Volume
  - Processor
  - Storage
    - Central Storage
      CSA
      ECSA
      SQA
      ESQ
    - Expanded Storage
    - Auxiliary Storage
  - Enqueue
  - Operator
  - SW-Subsystem
    - JES
    - HSM
    - XCF
- Coupling Facility
  - Coupling Facility Structure
- CPC
  - LPAR

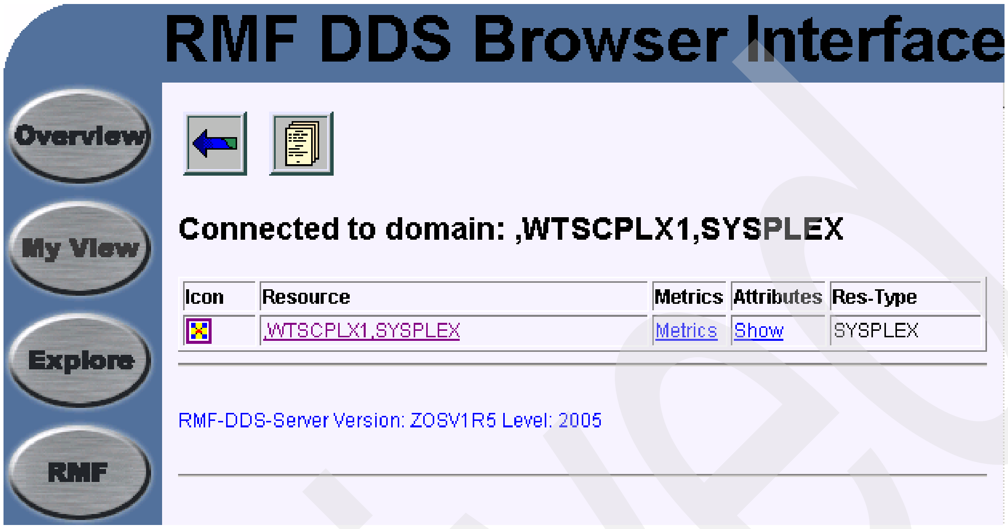
The following list shows the resources tree structure for zSeries Linux and Intel Linux:

Linux image

- Memory
- Network
- CPU
- Filesystem

Let’s explain this by using an example.

For z/OS, the top-level resource is the sysplex, and other resources are children or grand-children of the sysplex. So the z/OS images are children of the sysplex. And the z/OS image resource got associated metrics, like the number of active users or % workflow by WLM report class. Figure 2-48 shows the sysplex resource.



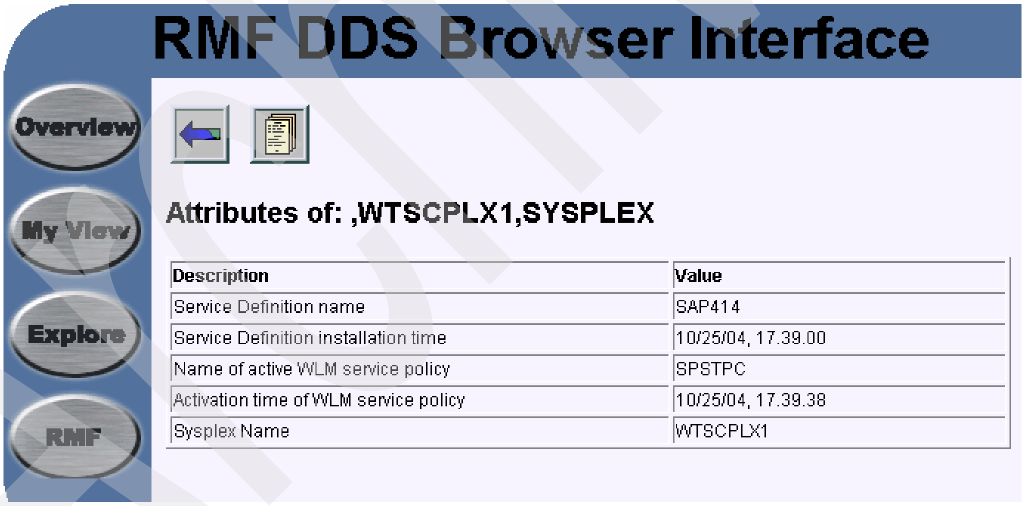
Connected to domain: ,WTSCPLX1,SYSPLEX

Icon	Resource	Metrics	Attributes	Res-Type
▣	,WTSCPLX1,SYSPLEX	Metrics	Show	SYSPLEX

RMF-DDS-Server Version: ZOSV1R5 Level: 2005

Figure 2-48 Sysplex resource

To see the attributes of the sysplex resource, click the hyperlink **Show** in the attributes column. Figure 2-49 shows the attributes of the sysplex resource.

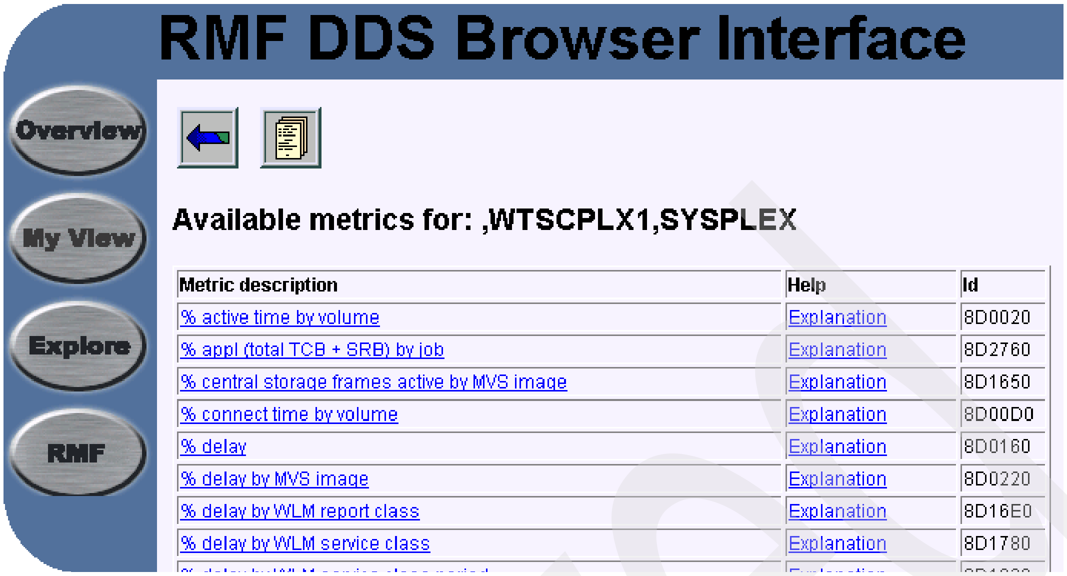


Attributes of: ,WTSCPLX1,SYSPLEX

Description	Value
Service Definition name	SAP414
Service Definition installation time	10/25/04, 17.39.00
Name of active WLM service policy	SPSTPC
Activation time of WLM service policy	10/25/04, 17.39.38
Sysplex Name	WTSCPLX1

Figure 2-49 Sysplex attributes

To see the metrics available for the sysplex resource, click the **Metrics** hyperlink. Figure 2-50 shows an extract of the available metrics for the sysplex resource. You get help for a metric when you click on the **Explanation** hyperlink in the Help column.

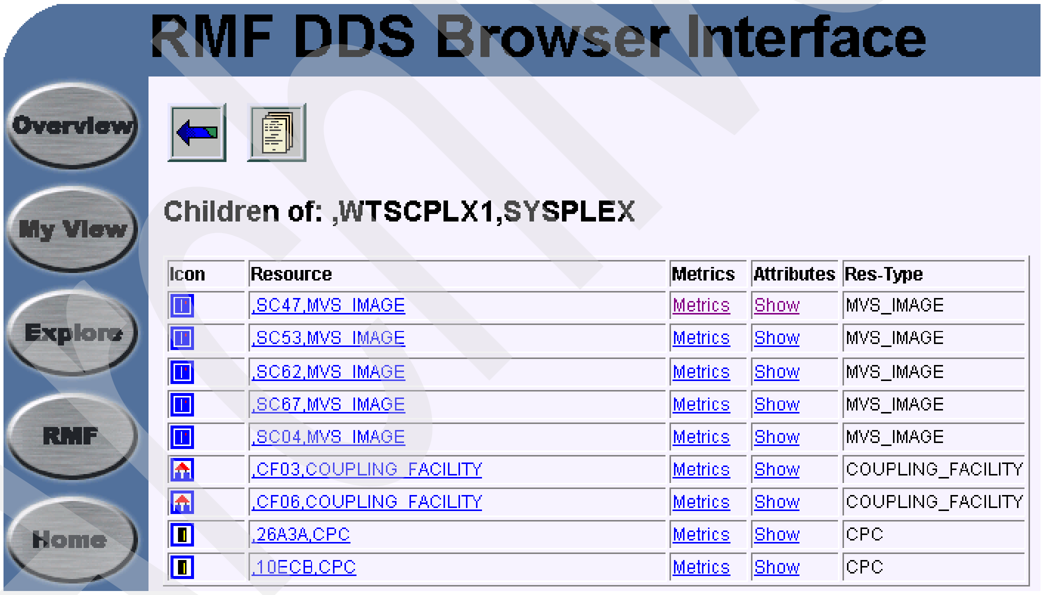


Available metrics for: ,WTSCPLX1,SYSPLEX

Metric description	Help	Id
% active time by volume	Explanation	8D0020
% appl (total TCB + SRB) by job	Explanation	8D2760
% central storage frames active by MVS image	Explanation	8D1650
% connect time by volume	Explanation	8D00D0
% delay	Explanation	8D0160
% delay by MVS image	Explanation	8D0220
% delay by WLM report class	Explanation	8D16E0
% delay by WLM service class	Explanation	8D1780

Figure 2-50 Metrics for the Sysplex resource

To explore the children resources of the sysplex, click the hyperlink **,WTSCPLX1,SYSPLEX**. Figure 2-51 shows the children resources for the sysplex resource.

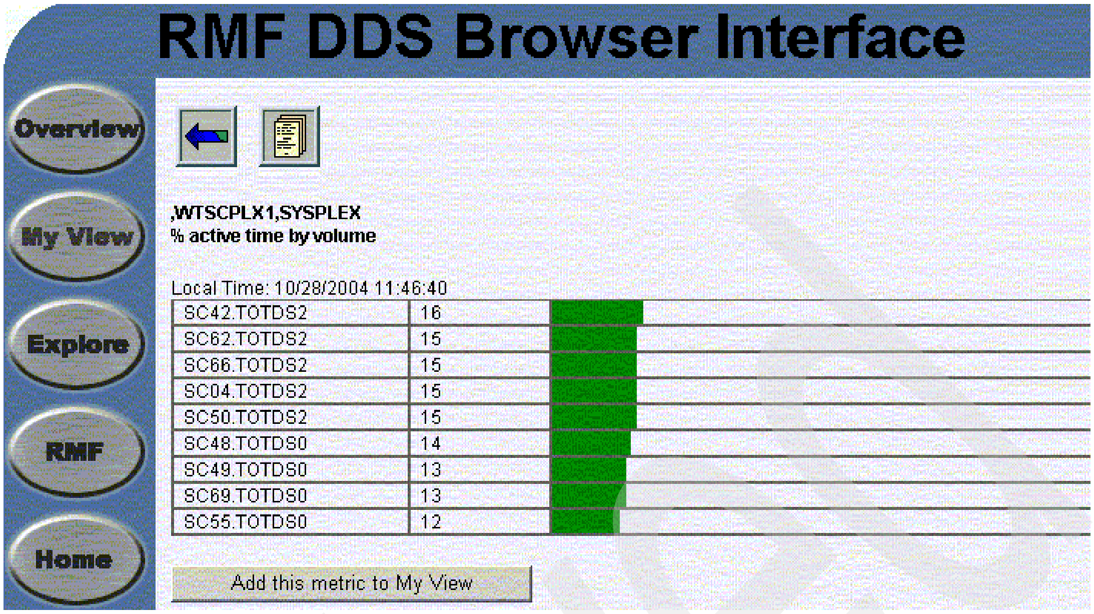


Children of: ,WTSCPLX1,SYSPLEX

Icon	Resource	Metrics	Attributes	Res-Type
	,SC47,MVS_IMAGE	Metrics	Show	MVS_IMAGE
	,SC53,MVS_IMAGE	Metrics	Show	MVS_IMAGE
	,SC62,MVS_IMAGE	Metrics	Show	MVS_IMAGE
	,SC67,MVS_IMAGE	Metrics	Show	MVS_IMAGE
	,SC04,MVS_IMAGE	Metrics	Show	MVS_IMAGE
	,CF03,COUPLING_FACILITY	Metrics	Show	COUPLING_FACILITY
	,CF06,COUPLING_FACILITY	Metrics	Show	COUPLING_FACILITY
	,26A3A,CPC	Metrics	Show	CPC
	,10ECB,CPC	Metrics	Show	CPC

Figure 2-51 Children resources of sysplex resource

Typically, each resource is associated with various metrics. A metric can either consist of exactly one value that is tied to one resource or a list of name/value pairs. To view a metric, just select the metric on the metric list. For example, click on the hyperlink **% active time by volume** shown in Figure 2-50 to see the metric shown in Figure 2-52.



,WTSCPLX1,SYSPLEX
% active time by volume

Local Time: 10/28/2004 11:46:40

SC42.TOTDS2	16
SC62.TOTDS2	15
SC66.TOTDS2	15
SC04.TOTDS2	15
SC50.TOTDS2	15
SC48.TOTDS0	14
SC49.TOTDS0	13
SC69.TOTDS0	13
SC55.TOTDS0	12

Figure 2-52 Sysplex metric % active time by volume

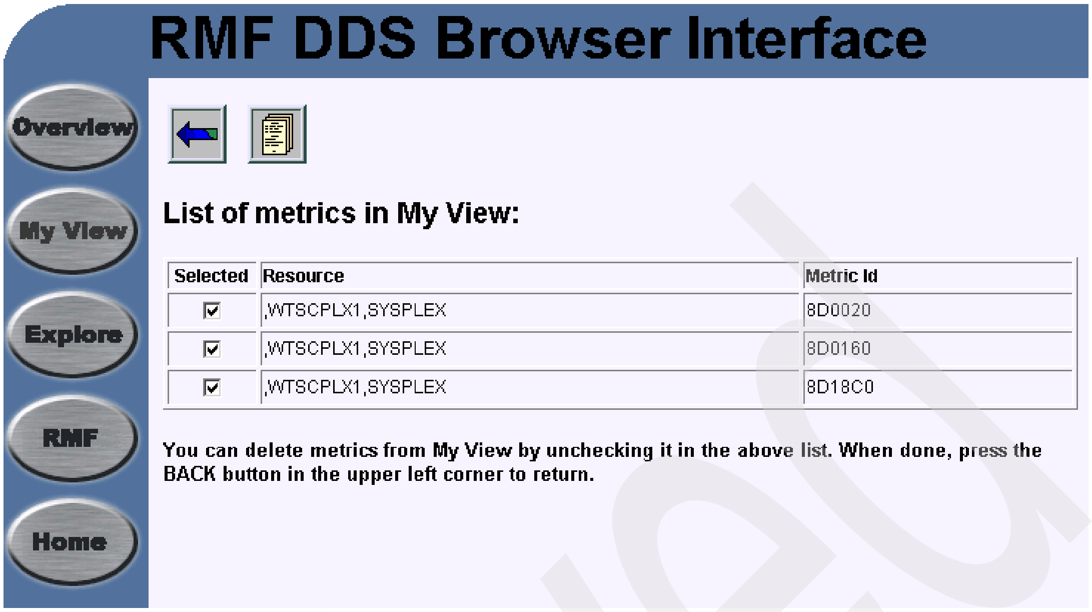
Click **Add this metric to My View** to add the metric to the set of metrics that are available when you use the My View function. This is explained in detail in the next section.

The metric view is automatically refreshed. The time stamp of the RMF interval time is displayed above the metrics. At the moment, when a Monitor III collection interval (MINTIME) has completed, the window is automatically refreshed with the new data.

My View function

This function allows you to define a set of metrics and save it to My View. The My View set of metrics is saved as a cookie on your workstation. Whenever you use the RMF data via a Web browser on that workstation again, you can recall the metrics which you have specified for My View.

To add a metric, click **Add this metric to My View** in the metric view, like Figure 2-50. After you have clicked the button, you see Figure 2-53. Here you can manage the metrics of My View. You see all the metrics that you have added to My View. To delete one of them, uncheck the metric you want to delete and use the back button.



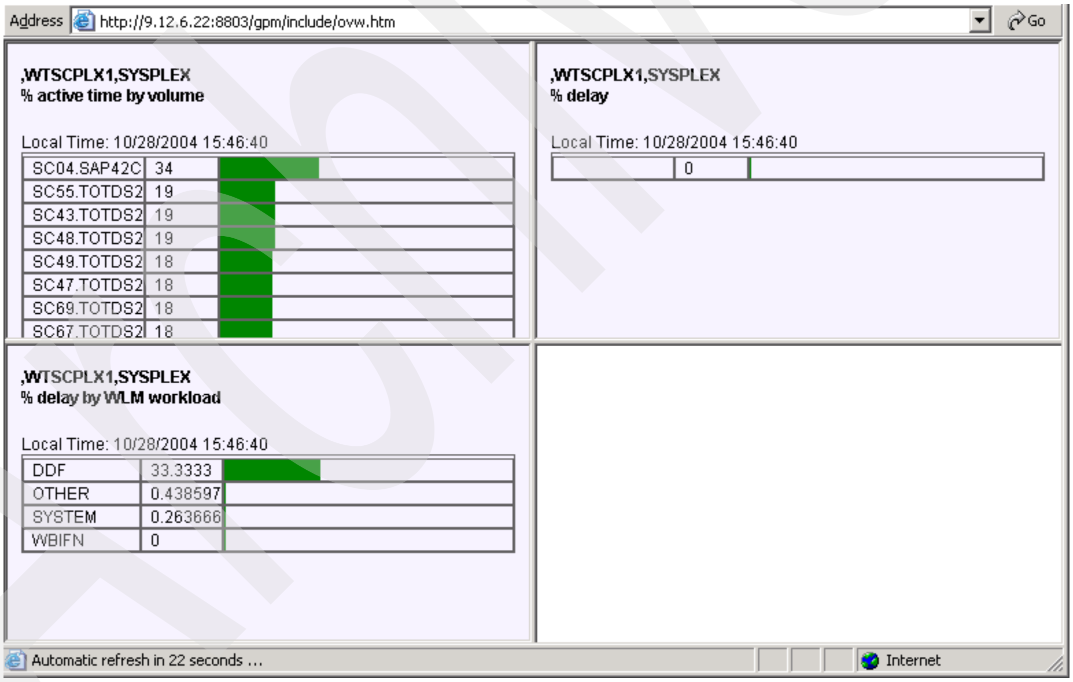
List of metrics in My View:

Selected	Resource	Metric Id
☑	,WTSCPLX1,SYSPLEX	8D0020
☑	,WTSCPLX1,SYSPLEX	8D0160
☑	,WTSCPLX1,SYSPLEX	8D18C0

You can delete metrics from My View by unchecking it in the above list. When done, press the BACK button in the upper left corner to return.

Figure 2-53 Manage My View

Click **My View** and open the view in a new window, as shown in Figure 2-54.



http://9.12.6.22:8803/gpm/include/ovw.htm

,WTSCPLX1,SYSPLEX
% active time by volume

Local Time: 10/28/2004 15:46:40

SC04.SAP42C	34
SC55.TOTDS2	19
SC43.TOTDS2	19
SC48.TOTDS2	19
SC49.TOTDS2	18
SC47.TOTDS2	18
SC69.TOTDS2	18
SC67.TOTDS2	18

,WTSCPLX1,SYSPLEX
% delay

Local Time: 10/28/2004 15:46:40

	0

,WTSCPLX1,SYSPLEX
% delay by WLM workload

Local Time: 10/28/2004 15:46:40

DDF	33.3333
OTHER	0.438597
SYSTEM	0.263666
WBIFN	0

Automatic refresh in 22 seconds ...

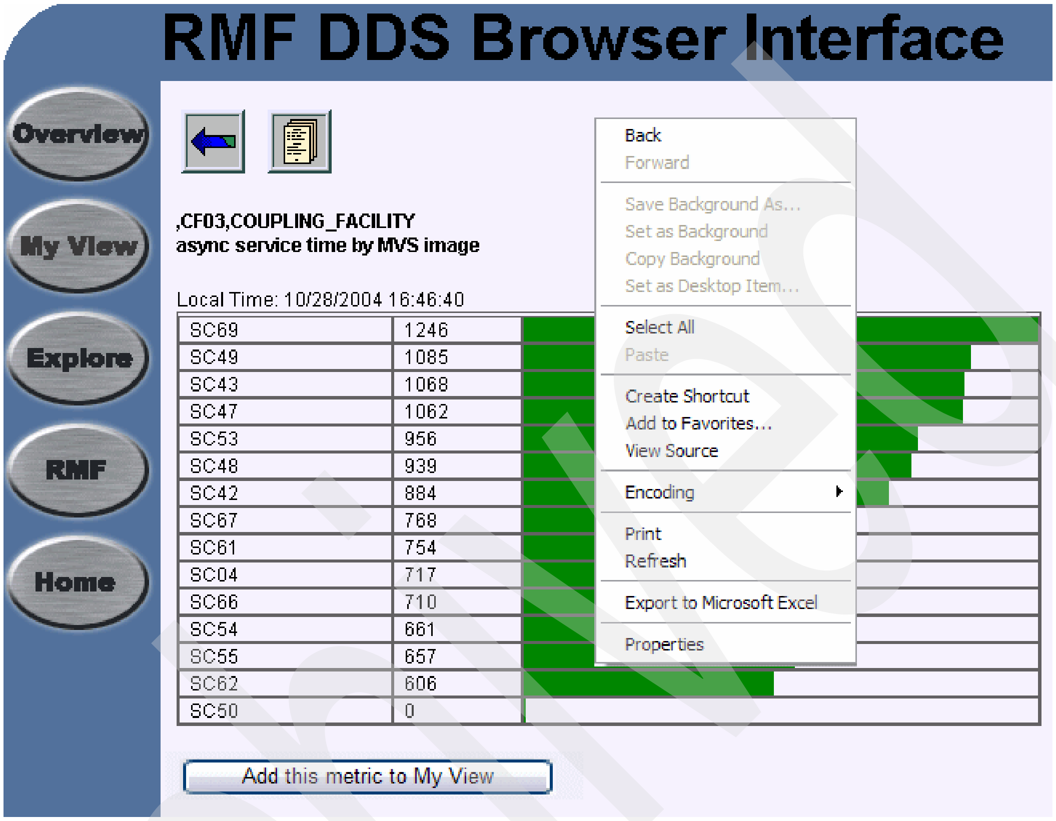
Figure 2-54 My View

As mentioned, the My View is saved as a cookie on your workstation. So if you want to use the My View function, ensure that the security settings of your Web browser and your security program (for example, firewalls) allow cookies.

Save data into spreadsheets

Once the performance data is displayed in your browser window, you can export it to spreadsheet applications by means of standard functions. This is supported by Microsoft Windows XP and Microsoft Excel 2002 or higher.

You can choose any monitoring window as a starting point, and then select the action **Export to Microsoft Excel** from the context menu of the free surface within this window. In Figure 2-55, we export the data from the metric `async service time by MVS Image` from the CF CF03.



,CF03,COUPLING_FACILITY
async service time by MVS image

Local Time: 10/28/2004 16:46:40

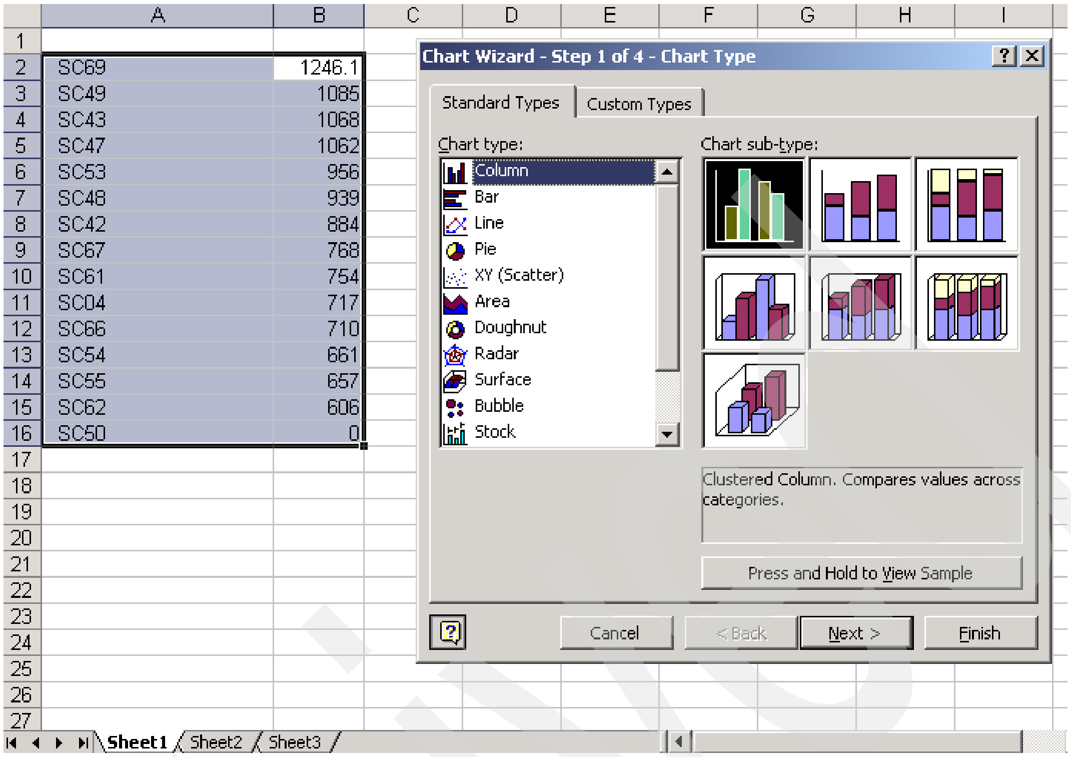
SC69	1246
SC49	1085
SC43	1068
SC47	1062
SC53	956
SC48	939
SC42	884
SC67	768
SC61	754
SC04	717
SC66	710
SC54	661
SC55	657
SC62	606
SC50	0

Figure 2-55 Export data from the metric view

As a result of the export function, Microsoft Excel is started and the data is exported to Microsoft Excel, as shown in Example 2-56. To create a chart, select the data range (cell A2 to cell B16) and click **Insert** → **Chart** to bring up the Chart Wizard.

The Chart Wizard offers four steps where you can customize your chart.

In Step 1, you can select the type of chart you want to create, such as bar charts or line charts. We use the default selection, the bar chart.



	A	B
1		
2	SC69	1246.1
3	SC49	1085
4	SC43	1068
5	SC47	1062
6	SC53	956
7	SC48	939
8	SC42	884
9	SC67	768
10	SC61	754
11	SC04	717
12	SC66	710
13	SC54	661
14	SC55	657
15	SC62	606
16	SC50	0

Chart Wizard - Step 1 of 4 - Chart Type

Chart type: Column

Clustered Column. Compares values across categories.

Figure 2-56 Chart Wizard: Step 1

Click **Next** to get to Step 2. In Step 2, click the Series tab and enter a name for the Series in the Name field, as shown in Figure 2-57.

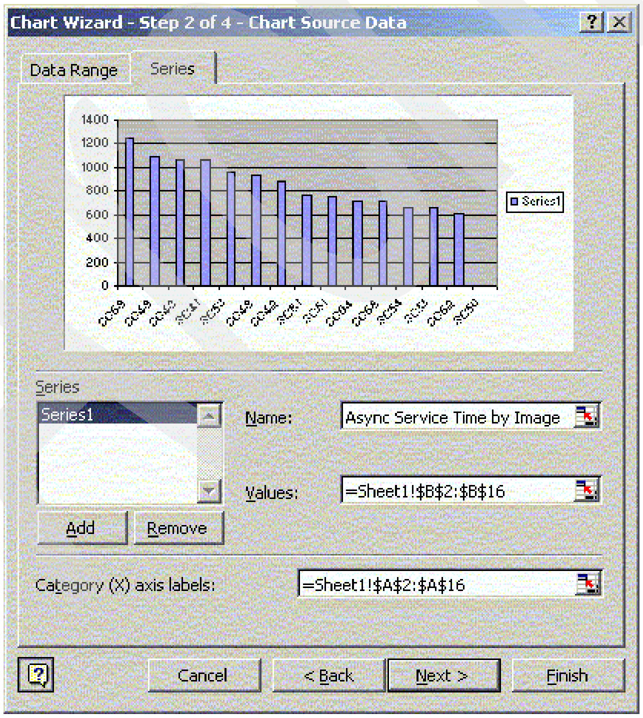


Chart Wizard - Step 2 of 4 - Chart Source Data

Series: Series1

Name: Async Service Time by Image

Values: =Sheet1!$B$2:$B$16

Category (X) axis labels: =Sheet1!$A$2:$A$16

Figure 2-57 Chart Wizard: Step 2

Click **Next** to get to Step 3. Here you can enter the Chart title, the Category (X) axis description and the Value (Y) axis description, as shown in Figure 2-58.

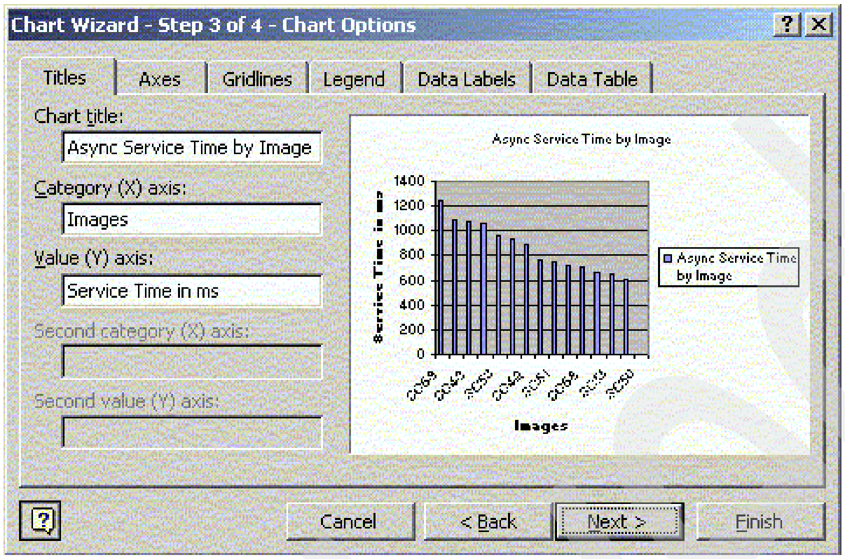


Chart title: Async Service Time by Image

Category (X) axis: Images

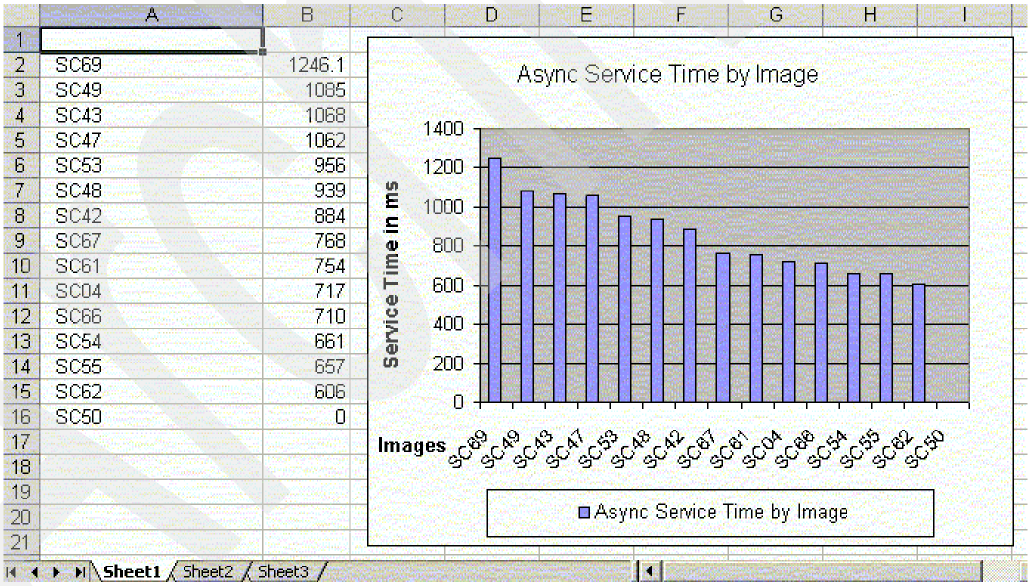
Value (Y) axis: Service Time in ms

Second category (X) axis:

Second value (Y) axis:

Figure 2-58   Chart Wizard: Step 3

Click **Next** to get to Step 4. Here you specify whether you want to create the chart on the same sheet or on a new sheet. Finally, click **Finish** to create the chart. Figure 2-59 shows an example chart.

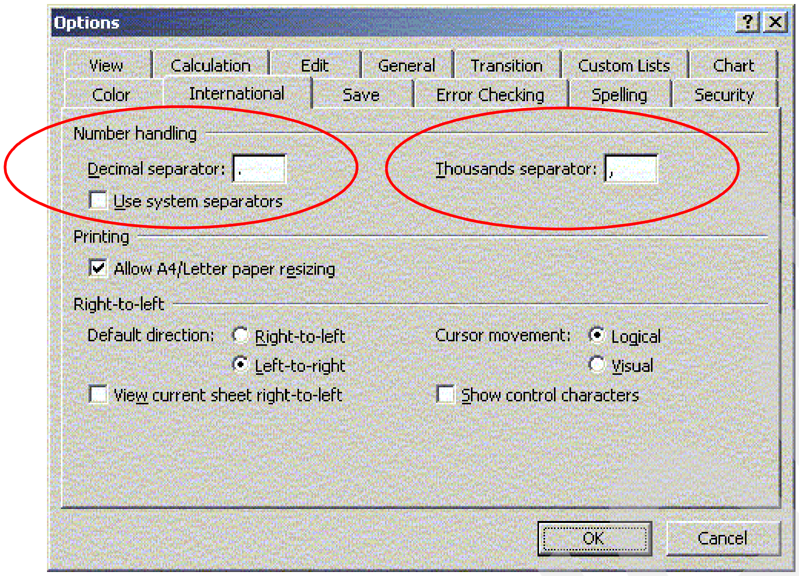


	A	B
1		
2	SC69	1246.1
3	SC49	1085
4	SC43	1068
5	SC47	1062
6	SC53	956
7	SC48	939
8	SC42	884
9	SC67	768
10	SC61	754
11	SC04	717
12	SC66	710
13	SC54	661
14	SC55	657
15	SC62	606
16	SC50	0
17		
18		
19		
20		
21		

Figure 2-59   Microsoft Excel Chart

Problem with decimal separator

The exported data uses the period as a decimal separator. If your regional settings of your systems define another decimal separator, like the comma, Microsoft Excel cannot handle the imported data correctly. Therefore, you have to change the decimal separator setting in Microsoft Excel. Click **Tools** → **Options** and select the International tab in the Options dialog window. Uncheck the option “Use system separator” and change the Decimal separator to the period. Also change the Thousands separator to the comma. Click **OK** to apply the settings.



Number handling

Decimal separator: .

Thousands separator: ,

□ Use system separators

Printing

☑ Allow A4/Letter paper resizing

Right-to-left

Default direction: ○ Right-to-left ● Left-to-right

Cursor movement: ● Logical ○ Visual

□ View current sheet right-to-left

□ Show control characters

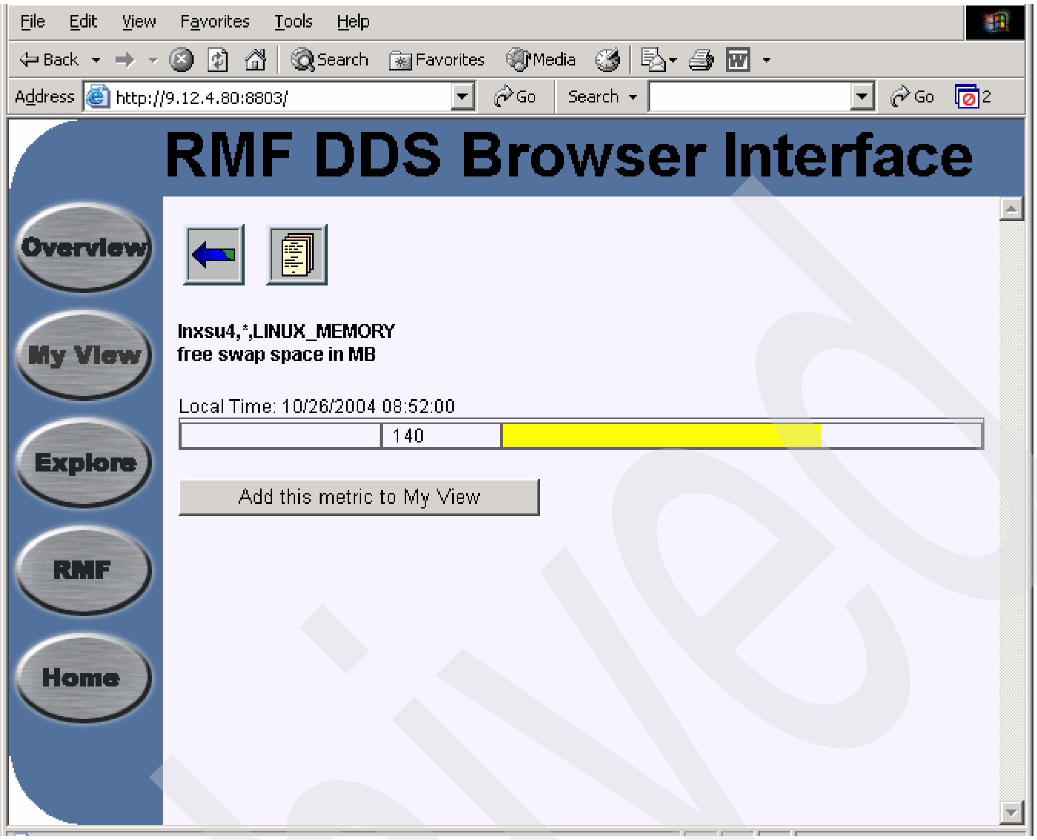
Figure 2-60 Change Decimal separator in Microsoft Excel

Exception definitions for Linux

Within rmfpms you have the option to define exceptions for metrics. You define the critical trigger and the warning trigger, using thresholds. When the metric reaches the critical or warning threshold, RMF PM and RMF Web browser interface are able to display the situation. If the metric matches the warning state, the metric value is displayed in the color yellow. If the metric matches the critical state, the metric value is displayed in the color red.

Figure 2-61 shows the result using the RMF Web browser interface when the metric `free swap space in MB` has reached the warning trigger, so it is displayed in the warning color (yellow).

The exceptions are defined in the gpmexusr.ini file for rmfpms. For details refer to “The gpmexusr.ini file” on page 141.


File Edit View Favorites Tools Help
Back
Search
Favorites
Media
Address http://9.12.4.80:8803/
Go
Search
Go
RMF DDS Browser Interface
Overview
My View
Explore
RMF
Home

lnxsu4,*,LINUX_MEMORY
free swap space in MB

Local Time: 10/26/2004 08:52:00

140

Add this metric to My View

Figure 2-61 Metric reached the warning trigger

2.9 Using RMF application programming interfaces

While RMF provides many ways to access the performance information it gathers, there are always situations where you either need information that is not accessible using one of the existing interfaces, or you want to use RMF information in one of your own programs. For these situations, RMF provides a number of application programming interfaces (APIs). This section describes the following:

- Getting sysplex-wide SMF records ERBDSQRY and ERBDSREC
- Getting sysplex-wide Monitor II information using ERB2XGDS
- Getting sysplex-wide Monitor III information using ERB3XDRS

We explain how to use these data services with the help of RMF-provided sample programs and exit routines. For more information, refer to the *RMF Programmer's Guide*, SC33-7994.

ERBDSQRY data service gives you a directory of the available SMF records in the RMF data buffers, for the timeframe you specify. This directory is for each system in the sysplex. It gives you a token for each available record. Based on this information, you can then request all or some of the records you need from RMF, using data service ERBDSREC.

ERBDSREC data service returns the SMF records that you required. You give a token that you have gotten from the data service ERBDSQRY for each record you require.

ERB2XDGS data service returns Monitor II SMF records from the RMF data buffer to you. These are type 79 records. You specify which subtypes you require by calling a data reduction exit routine, which RMF provides or you can write your own.

ERB3XDRS data service returns data from the Monitor III buffer or from the VSAM data set to you. To get these “sets of samples,” you specify the timeframe for them, and whether you want to get the sets of samples sequentially per system, or combined for the entire sysplex. A description of all the available fields is in the *RMF Programmer’s Guide*, Chapter 7.

2.9.1 Calling services ERBDSQRY, ERBDSREC, ERB2XGDS, and ERB3XDRS

This is a brief introduction on how to call RMF data services. Use them with the help of RMF-provided sample programs and exit routines. For more information and description of the APIs, refer to *RMF Programmer’s Guide*, Chapter 2.

We describe the general structure to access information using the sample programs and exit routines provided by RMF in this section. All modules are in SAMPLIB in source format, except the data reduction exits ERB2XSMF and ERB3XSOS, which are “dummy” exits that you do not need to modify.

ERBDSMP1 is a sample program, written in C language. It calls the data services ERBDSQRY and ERBDSREC.

- ERBDSQRY data service returns the tokens for each SMF record available, for the time frame you specify. These tokens are then used when calling the ERBDSREC service.
- ERBDSREC data service receives the tokens from you, and it returns a record for each token, to the data area provided by you.

ERBDSMP2 is a sample program, written in C language. It calls the data service ERB2XDGS.

- ERB2XDGS data service returns the required Monitor II SMF records. Additionally, this data service calls the exit routine ERBDSMX2, which is written in Assembler.
  - ERBDSMX2 is the data reduction exit routine provided by RMF. It is called for each system you have, before data is moved to the answer area. By default it does not reduce the data, but it can be used as a sample, which you can adapt for data reduction.
  - ERB2XSMF could be used instead of ERBDSMX2. This exit routine is provided by RMF and copies the complete data gathered by the Monitor II data gatherer (SMF record type 79) to the answer area. ERB2XSMF has no exit parameters.

ERBDSMP3 is a sample program, written in C language. It calls the data service ERB3XDRS.

- ERB3XDRS data service returns the sets of samples, according to the timeframe you specify. You get the sets of samples either sequentially per system, or combined for the sysplex. Additionally, this calls exit ERBDSMX3, written in Assembler.
  - ERBDSMX3 is the data reduction exit routine provided by RMF. It is called for each system you have, before the data (a set of samples) is moved forward. It does not reduce the amount of data, but it can be used as a sample, which you can modify.
  - ERB3XSOS is a data reduction exit routine provided by RMF. It is called for each system you have, before the data (a set of samples) is moved forward. It does not make any reductions. You could use this instead of ERBDSMX3, if you do not make any reductions.

Part 2

Setting up and customizing RMF components

In this part we describe the customizing of the RMF components. We customize the traditional RMF components Monitor I, Monitor II, and Monitor III after installation.

For the new RMF workstation components, we describe the installation and the customization of the RMF Spreadsheet Reporter, RMF Performance Monitoring, and the Distributed DataServer.

3

Setup and customization of the traditional favorites

This chapter describes the customization of the following RMF components after installation:

- Monitor I (ERBRMF00 member)
- Monitor II (ERBRMF01 member)
- Monitor III (ERBRMF04 member)
- Sysplex Data Server

3.1 RMF customization

In order to use RMF, you have to do several customization tasks. In 3.1.1, “Basic customization” on page 112, we explain tasks such as library authorization and specifying the priority of RMF. If you are already familiar with basic customization tasks, you can skip this section and start with “Advanced RMF configuration” on page 115, where we describe advanced customization tasks, such as enabling extended CPMF mode to get extended channel path measurements.

3.1.1 Basic customization

This section describes the basic customization to enable RMF.

Define RMF library authorization

All RMF load modules reside in the two libraries SYS1.SERBLINK and SYS1.SERBLPA. You have to add SYS1.SERBLINK to your link list and APF list, and SYS1.SERBLPA to your LPA list.

It is possible to make this change with or without an IPL of the system.

Activate RMF with IPL

We recommend that you use your PROGxx members instead of using the LNKLSTxx member for the link list concatenation and APF authorization.

We recommend that you use several PROGxx members to do this in a structured way, where each member has a dedicated function:

- One PROGxx member handles the APF authorization.
- One PROGxx member defines a link list.
- One PROGxx member adds libraries to the link list.
- One PROGxx member activates the link list.

First we add SYS1.SERBLINK to the APF list to become APF authorized.

Example 3-1 PROGxx member handles the APF authorization

```
APF FORMAT(DYNAMIC)
APF ADD
    DSNAME(SYS1.SERBLINK)
    VOLUME(******)
...
```

In Example 3-1, VOLUME(******) indicates that SYS1.SERBLINK resides on the SYSRES. Otherwise, you have to specify the volume where the SYS1.SERBLINK data set resides.

Next we define the link list LNKLST00.

Example 3-2 PROGxx member defines a link list

```
LNKLST Define Name(LNKLST00)
...
```

Now we add SYS1.SERBLINK to the already defined link list LNKLST00, as shown in Example 3-3.

Example 3-3 PROGxx member adds libraries to the link list

```
LNKLST ADD NAME(LNKLST00)
DSNAME(SYS1.SERBLINK)
...
```

Finally, we activate the previously defined link list LNKLST00.

Example 3-4 PROGxx member activates the link list

```
LNKLST Activate Name(LNKLST00)
```

We also add SYS1.SERBLPA to the LPA list, as shown in Example 3-5.

Example 3-5 Extract from LPALSTxx

```
SYS1.SERBLPA,
...
```

RMF requires all of these definitions. The next time we IPL the system, these changes become active.

Activate RMF without an IPL

We can do this configuration without an IPL. In this situation, we use one PROGxx member, which creates a copy of the current linklist, and add the RMF libraries.

The PROGPM member in Example 3-6 shows a sample definition. Ensure that the linklist RMFLNKST is currently not defined. If it is already defined to the system, you have to undefine the linklist, using the `LNKLST UNDEFINE NAME(RMFLNKST)` option.

To add the LPA modules from the SYS1.SERBLPA to the LPA, we have to use the **`LPA ADD`** command.

> **Attention:** Be careful using the `UPDATE` command. Updating an address space with a running program might cause an error in getting a module or by locating an incorrect copy of the module. The system does not verify the validity of the data for `UPDATE`.

Example 3-6 PROGPM member that enables RMF without an IPL

```
APF FORMAT(DYNAMIC)
APF ADD
    DSNAME(SYS1.SERBLINK)
    VOLUME(******)
LNKLST DEFINE NAME(RMFLNKST) COPYFROM(CURRENT)
LNKLST ADD NAME(RMFLNKST)
       DSNAME(SYS1.SERBLINK) ATTOP
LNKLST Activate Name(RMFLNKST)
LPA ADD MASK(*) DSNAME(SYS1.SERBLPA)
LNKLST UPDATE,JOB=*
```

Now use the SET PROG console command to activate the PROGPM member.

```
SET PROG=PM
```

For more information about adding libraries to the link, APF, and LPA lists, with or without an IPL, see *z/OS MVS Initialization and Tuning Reference,* SA22-7592. For information about the syntax of the SETPROG command, see *z/OS MVS System Commands,* SA22-7627.

Specifying priority for RMF

It is required that RMF and RMFGAT (when started) have the second-highest priority in the system, next to the system address spaces. Use the WLM application to put RMF and RMFGAT in service class SYSSTC because its dispatching priority is always above any installation-defined service class. If the priority is too low, RMF is not dispatched when its interval time expires. If this problem occurs, data collection for jobs running with higher priority can be incomplete, or the system cannot perform any event processing. This could also result either in incorrect measurement reports, or in common storage shortages which might lead to an IPL.

> **Important:** Always ensure that RMF and RMFGAT have the second-highest priority in the system.

Ensure linkage to language environment

Two components of RMF, the Postprocessor and the DDS, use the services of the Language Environment®. They need access to the data set SYS1.SCEERUN. There are two ways to provide this access:

- The recommended way is to include the data set SYS1.SCEERUN in the link list of the system on which RMF is running. No further action is required when starting the separate components.
- If you do not wish to include SYS1.SCEERUN in the link list, you must specify SYS1.SCEERUN as the STEPLIB of the job step that starts the component. The sample jobs can be found in SYS1.SAMPLIB member ERBSAMPP to start the Postprocessor, and member GPMSERVE in SYS1.PROCLIB to start the DDS.

Check the program properties table

z/OS provides two default entries in the program properties table (PPT) for the RMF modules, ERBMFMFC and ERB3GMFC. We recommend that you run with the defaults provided in the PPT, or the results are unpredictable. The default entries include:

- Non-swappable
- System task
- No protection key
- No processor affinity

Any user modifications to those entries require you to specify a PPT entry for ERBMFMFC and ERB3GMFC in a SCHEDxx parmlib member, which must include the RMF defaults and user overrides.

Example 3-7 contains an extract of the SCHED00 member, with the default PPT definitions for ERBMFMFC and ERB3GMFC.

Example 3-7 Extract of SCHED00 member

```
PPT       PGMNAME(ERBMFMFC)
          NOSWAP
          SYST
          NODSI
          AFF(NONE)
          NOPREF
PPT       PGMNAME(ERB3GMFC)
          NOSWAP
          SYST
          NODSI
          AFF(NONE)
```

```
         NOPREF
```

Note: Do not specify a protection key for these entries.

IPL with the CMB parameter

If you intend to monitor devices other than tape and DASD, you must IPL with the CMB system parameter and describe the number of extra measurement blocks required. RMF requires one extra measurement block for each extra device number to be monitored.

CMB is an IEASYSxx parameter. This CMB parameter specifies the maximum number of devices that you plan to dynamically add and measure plus the number of non-DASD/Tape devices. In Example 3-8, you see the CMB parameter with CMB=10000, which is a good value. If you are running on a system that exploits the z990 channel subsystem, the CMB parameter in IEASYSxx is ignored and the function is replaced with the measurement data in ECMB.

Example 3-8 Extract of an IEASYSxx parmlib member

```
CLOCK=&CLOCK.,
 CLPA,
 CMB=10000,
...
```

See *z/OS MVS Initialization and Tuning Reference* for more information on this parameter.

3.1.2 Advanced RMF configuration

This section discusses additional configuration steps.

Ensure common storage tracking

To ensure that the Postprocessor Common Storage report (STORC) provides complete data, it is required that VSM common storage tracking is active. The DIAGxx member controls the common storage tracking. In Example 3-9, the DIAG01 member enables common storage tracking, which is the default.

Example 3-9 DIAG01 parmlib member

```
 VSM TRACK CSA(ON) SQA(ON)
```

To activate the DIAG01, issue the console command **`SET DIAG=01`**.

If VSM common storage tracking is not active, one of the messages ERB617I, ERB618I, or ERB619I indicates that the report may be incomplete for some jobs.

Assign started task procedures to user IDs

It is possible to assign a unique user ID for each started task procedure, or one user for all the tasks.

We recommend that you define a unique user for each started task procedure. Here we show you the definition using Security Server (RACF).

First we have to create a group with an OMVS group ID (GID). We name it RMFGRP.

In Example 3-10, we define three user IDs: RMF, RMFGAT, and GPMSERVE, which have the RMFGRP as a default group ID. The RMFGRP default group must have an OMVS group ID (GID, and each user must have an OMVS user ID (UID). We choose GID 1000 and UID 1001,1002, and 1003.

Ensure that the UIDs and GID are currently not in use by other users. The GID and UIDs are used to control OMVS file access. If you specify two users with the same UID, each user can access the files of the other user.

Example 3-10 Definition of default group and three User IDs

```
ALG RMFGRP OMVS(GID(1000))
ADDUSER RMF DFLTGRP(RMFGRP) OMVS(UID(1001) HOME(‘/’))
ADDUSER RMFGAT DFLTGRP(RMFGRP) OMVS(UID(1002) HOME(‘/’))
ADDUSER GPMSERVE DFLTGRP(RMFGRP) OMVS(UID(1003) HOME(‘/’))
```

After we define the three User IDs, we assign the started tasks supplied by RMF to these User IDs., as shown in Example 3-11. After the assignment of the user IDs, we refresh the RACF option.

Example 3-11 Assign started tasks to user IDs

```
RDEFINE STARTED RMF.* STDATA(USER(RMF) TRUSTED(YES))
RDEFINE STARTED RMFGAT.* STDATA(USER(RMFGAT) TRUSTED(YES))
RDEFINE STARTED GPMSERVE.* STDATA(USER(GPMSERVE) TRUSTED(YES))
SETROPTS RACLIST(STARTED) REFRESH
```

Controlling access to RMF data for the sysplex data services

You must have RACF authorization to use applications that call sysplex data services to access data from the RMF Sysplex Data Server’s SMF buffer.

RMF has defined a RACF resource profile of class FACILITY called ERBSDS.SMFDATA to control access to SMF data in the SDS SMF buffers. Authorize every user that accesses the SMF records in this SMF buffer.

- ERBSDS.SMFDATA controls access to SMF data in the SMF buffer by the ERBDSQRY service (Query Available Sysplex SMF Data) or the ERBDSREC service (Request Sysplex SMF Record Data). One application using these services is the Postprocessor when the SMF records are retrieved directly from the SMF buffers. Another application using these services is the data gatherer of the Monitor II ILOCK command.

RMF does not perform mandatory access checks for Monitor II data (accessed by the ERB2XDGS service) and Monitor III set of samples data (accessed by the ERB3XDRS service). If you want to protect this data, define RACF resource profiles called ERBSDS.MON2DATA and ERBSDS.MON3DATA in the FACILITY class. If you do not define a profile, RACF does not restrict any user ID from invoking the mentioned sysplex data services:

- ERBSDS.MON2DATA controls access to Monitor II SMF type 79 data by the ERB2XDGS service. For example, a Monitor II reporter session invokes this service when reporting about another system in the sysplex.
- ERBSDS.MON3DATA controls access to Monitor III set of samples data by the ERB3XDRS service, for example, the Distributed Data Server as server address space for users of RMF PM calls this service. If this profile is defined, you must authorize the TSO user ID of RMF PM users. Also, a Monitor III reporter session calls this service when sysplex-wide reports are requested.

If the same group of users takes advantage of all RMF sysplex data services, you can work with the generic profile ERBSDS.*.

In Example 3-12 for Security Server (RACF), we set up the resource profiles ERBSDS.SMFDATA, ERBSDS.MON2DATA, and ERBSDS.MON3DATA, and grant access to these data sources for the user ID MAILAND.

Example 3-12 TSO commands for Security Server (RACF) resource profiles setup

To activate the resource class FACILITY:

```
SETROPTS CLASSACT(FACILITY) GENCMD(FACILITY) GENERIC(FACILITY)
```

To define the profiles:

```
RDEFINE FACILITY ERBSDS.SMFDATA UACC(NONE)
RDEFINE FACILITY ERBSDS.MON2DATA UACC(NONE)
RDEFINE FACILITY ERBSDS.MON3DATA UACC(NONE)
```

To grant the userid of the application program READ access:

```
PERMIT ERBSDS.SMFDATA CLASS(FACILITY) ID(MAILAND) ACC(READ)
PERMIT ERBSDS.MON2DATA CLASS(FACILITY) ID(MAILAND) ACC(READ)
PERMIT ERBSDS.MON3DATA CLASS(FACILITY) ID(MAILAND) ACC(READ)
```

Activate changes:

```
SETROPTS REFRESH RACLIST(FACILITY)
```

You get the error message such as Example 3-13 when you try to access the protected resource ERBSDS.SMFDATA. We used a Postprocessor job to create an RMF report using SMF data from the SMF buffer. This failed because the user ID MAILAND has no access to resource ERBSDS.SMFDATA.

Example 3-13 Error message trying to access protected resource

```
11.39.30 JOB25131  ICH408I USER(MAILAND ) GROUP(SYS1    ) NAME(MAILAND
   472              ERBSDS.SMFDATA CL(FACILITY)
   472              INSUFFICIENT ACCESS AUTHORITY
   472              ACCESS INTENT(READ   )  ACCESS ALLOWED(NONE   )
```

When using RMF PM and your TSO user ID was not authorized to access the protected resource, you receive following error message by RMF PM:

```
GPM0456I The userid MAILAND is not authorized to retrieve RMF performance data.
```

Do one of the following if you want to prevent unauthorized access to the sysplex data services:

- Define the profiles ERBSDS.SMFDATA, ERBSDS.MON2DATA, and ERBSDS.MON3DATA to the FACILITY class to protect access to the related sysplex data services,
- Work with the generic profile ERBSDS.* and have generic profile checking active.

Customizing RMF control session

IBM provides the cataloged procedure that is necessary to start RMF. The procedure is stored in SYS1.PROCLIB(RMF), and you can modify it according to your requirements. The RMF control session is the base for data gathering for Monitor I, II, and Monitor III.

The modified start procedure in Example 3-14 shows where, instead of the default ERBRMF00 parmlib member, the parmlib member ERBRMF10 is used for Monitor I gatherer. The SMF buffer is initialized to the size of 256 MB, and it is specified that RMF SMF records type 70 to 79 and 79(15) are to be stored cumulatively in the wraparound SMF buffer.

Example 3-14 Modified RMF start procedure in SYS1.PROCLIB

```
//IEFPROC EXEC PGM=ERBMFMFC,REGION=0M,
//         PARM='(MEMBER(10),SMFBUF(SPACE(256M),RECTYPE(70:79,79(15))))'
```

The RMF Sysplex Data Server is always active when the RMF address space is running.

To start RMF, issue the console command **START RMF**.

Example 3-14 shows how RMF is started with the modified start procedure.

Example 3-15 Start RMF

```
S RMF
IRR812I PROFILE RMF.* (G) IN THE STARTED CLASS WAS USED 363
        TO START RMF WITH JOBNAME RMF.
$HASP100 RMF      ON STCINRDR
IEF695I START RMF      WITH JOBNAME RMF      IS ASSIGNED TO USER RMF
   , GROUP STCGROUP
$HASP373 RMF STARTED
IEF403I RMF - STARTED
ERB100I RMF: ACTIVE
IEE252I MEMBER ERBRMF00 FOUND IN SYS1.PARMLIB
ERB450I RMF: SMF DATA BUFFER INITIALIZED
```

Startup automation

To start RMF automatically after you IPL the system, you can include the RMF start command and the required parameters in your active COMMNDxx parmlib member.

The extract of the COMMND00 parmlib member in Example 3-16 shows how RMF is automatically started, using parmlib member ERBRMF10 for Monitor I gatherer and creating the SMF buffer, designating the size of 256 MB and specifying that RMF SMF records type 70 to 79 and 79(15) are to be stored in it.

Example 3-16 Extract of COMMND00 member

```
COM='D D,T,AUTODSN=ALL'
COM='S RMF,,,(MEMBER(10),SMFBUF(SPACE(256M),RECTYPE(70:79,79(15))))'
...
```

3.2 Monitor I

This section discusses how to control the Monitor I data gatherer. The dependencies to SMF are covered as well.

For a detailed discussion about the Monitor I option, refer to *RMF User's Guide*, Chapter 11.

3.2.1 Relationship to SMF

We recommend that the Monitor I is synchronized to SMF. This is specified in the ERBRMFxx member for Monitor I via the **SYNC(SMF)** option.

To customize SMF, change the SMFPMRxx parmlib member. Use the same SMF interval (**INTVAL(xx)**) and the same synchronization (**SYNCVAL(00)**) for all systems in the sysplex. An SMF interval of 15 minutes gives you good flexibility for the creation of Postprocessor reports (**INTVAL(15)**).

You need to specify that the RMF SMF records are written to SMF dump data sets. This is done by the **SYS(TYPE(xx))** and **SUBSYS(TYPE(xx))** statements.

The RMF SMF records are SMF Record type 70-79. You may also want to include the HTTP Server SMF Record type 103 and Lotus Domino Server SMF Record type 108, since RMF also is able to process them. To have copies of SMF records type 79 subtype 15 available in the SMF buffer of the Sysplex Data Server, you have to activate the exits IEFU83, IEFU84, and IEFU85. These copies are needed for Monitor II IRLM long lock reporting.

The SMFPRM00 parmlib member in Example 3-17 specifies the recommended options.

Example 3-17 Extract of SMF SMFPRM00 parmlib member

```
   INTVAL(15)                /* SMF INTERVAL IS 15 MINUTES       */
    SYNCVAL(00)               /* INTERVAL STARTS ON THE HOUR      */
SYS(TYPE(70:79,103,108), /* RMF SMF records*/
   EXITS(IEFU83,IEFU84,IEFU85,IEFACTRT, /* Exit for 79(15) */
      IEFUJV,IEFUSI,IEFUJP,IEFUSO,IEFUTL,IEFUAV),
      INTERVAL(SMF,SYNC),NODETAIL)
SUBSYS(STC,EXITS(IEFU29,IEFU83,IEFU84,IEFU85,IEFUJP,IEFUSO,
       IEFACTRT),
       INTERVAL(SMF,SYNC),
       TYPE(70:79,103,108))
...
```

Tip: Although not related to RMF, consider during the setup of your SMF member, that collecting record type 92 and 99 can generate a lot of data and your installation should turn the collection of this data off. Furthermore, if you are not using measured usage licensing, then record type 89 can be turned off as well.

3.2.2 Monitor I customization considerations

The gatherer options are stored in an ERBRMFxx parmlib member. ERBRMF00 is the default parmlib member. You can adapt this member to your needs. For further details about the data gatherer options, see *RMF User's Guide*, Chapter 11.

In this section we discuss some additional details we have to consider.

Cache and ESS data gathering

Cache controller data is gathered by individual device address. There is no indication of which system in the sysplex initiates a recorded event. Therefore, it is possible to gather data on any system sharing the cached devices.

To avoid having duplicated data, you should gather cache activity data on one system only. In addition, ESS and cache should be set on the same system where you are collecting the data. That system should have connectivity to all SSIDs. To suppress the gathering of cache data, specify the NOCACHE option.

But if you have dedicated devices and an asymmetric hardware definition, you need to gather cache control data on more than one system. The Cache Subsystem Activity Postprocessor report created from the gathered data from one system, doesn't cover dedicated devices of other systems. So to get the complete picture of your sysplex, you have to look at the Cache Subsystem Activity Reports from several systems.

Since RMF has no sysplex control over the gatherer options, it cannot automatically deselect cache gathering on all but one system. But you can use the symbolics feature offered by z/OS.

Let us say we have a sysplex with four systems, A01, A02, A03, and A04. We decide to enable the cache gathering only on system A01.

Therefore, we define a sysplex-wide symbol &CACHEOPT as NOCACHE in the IEASYMxx member, as shown in Example 3-18. For the system A01, we define also the symbol &CACHEOPT, but as CACHE. This means that the &CACHEOPT sysplex symbol is resolved as NOCACHE in the sysplex systems, but on system A01 we override it with system symbol &CACHEOPT with CACHE. And we do the same for the ESS option.

Example 3-18 Extract of IEASYMxx

```
SYSDEF          SYSCLONE(&SYSNAME(3:2))
                SYMDEF(&CLOCK='VM')       /* USE CLOCKVM   */
                SYMDEF(&SMFPARM='00')     /* POINT TO SMFPRM00 */
                SYMDEF(&SSNPARM='00')     /* POINT TO IEFSSN00 */
                SYMDEF(&BPXPARM='FS')     /* SYSPLEX FILE SHARING */
                SYMDEF(&CACHEOPT='NOCACHE')
                SYMDEF(&ESS='NOESS')
SYSDEF          HWNAME(SCZP702)
                   LPARNAME(A01)
                   SYSDEF(&CACHEOP='CACHE')  /* CACHE FOR A01 */
                   SYSDEF(&ESS='ESS')        /* ESS FOR A01  */
                   SYSNAME(SYS1)
...
```

Now we use the defined symbols &CACHEOPT and &ESS on one common Monitor I member ERBRMFxx, as in Example 3-19. If this member is used for Monitor I, the data gathering options CACHE and ESS are only enabled on LPAR A01.

Example 3-19 Extract of Monitor I parmlib member

```
          CPU                          /* COLLECT CPU STATISTICS       */
          &CACHEOPT
          &ESS
          PAGING                       /* COLLECT PAGING STATISTICS    */
...
```

Note: If you want to activate option ESS, you have to perform the same procedure as with option CACHE. Both options need to be active on the same system.

LPAR data gathering

If you want to gather information about other LPARs in terms of Partition Data report and CPC report, you must enable the Performance Data Control of the Logical Partition Security profile on the HMC.

3.2.3 Starting Monitor I

IBM provides the cataloged procedure which is necessary to start RMF. The procedure is stored in SYS1.PROCLIB(RMF), and you can modify it according to your requirements.

Example 3-20 shows how to start RMF using the standard procedure that uses the default ERBRMF00 parmlib member and without the SMF buffer. The RMF Sysplex Data Server is always active when the RMF address space is running. When the RMF control session is started, Monitor I is started by default.

Example 3-20 Start RMF using default options

```
START RMF
IRR812I PROFILE RMF.* (G) IN THE STARTED CLASS WAS USED 363
        TO START RMF WITH JOBNAME RMF.
$HASP100 RMF      ON STCINRDR
IEF695I START RMF      WITH JOBNAME RMF      IS ASSIGNED TO USER RMF
   , GROUP STCGROUP
$HASP373 RMF STARTED
IEF403I RMF - STARTED
ERB100I RMF: ACTIVE
IEE252I MEMBER ERBRMF00 FOUND IN SYS1.PARMLIB
ERB100I ZZ : ACTIVE
```

"Customizing RMF control session" on page 117 shows how you can customize the standard procedure.

To start RMF using the modified parmlib member, ERBRMF10, for Monitor I, use the start command with parameters as Example 3-21 shows.

Example 3-21 Start RMF using ERBRMF10 parmlib member and SMF buffer

```
START RMF,,,(MEMBER(10))
IRR812I PROFILE RMF.* (G) IN THE STARTED CLASS WAS USED 363
        TO START RMF WITH JOBNAME RMF.
$HASP100 RMF      ON STCINRDR
IEF695I START RMF      WITH JOBNAME RMF      IS ASSIGNED TO USER RMF
   , GROUP STCGROUP
$HASP373 RMF STARTED
IEF403I RMF - STARTED
ERB100I RMF: ACTIVE
IEE252I MEMBER ERBRMF10 FOUND IN SYS1.PARMLIB
ERB100I ZZ : ACTIVE
```

To check to see if the RMF control session is active and which data gatherers are active, use the display command, as in Example 3-22.

Example 3-22 Control Session display command

```
MODIFY RMF,DISPLAY
ERB211I RMF: ACTIVE SESSIONS - III,AA,ZZ
```

To verify the current options and that Monitor I is running, use the display command for a specific session, as in Example 3-23.

Example 3-23 Monitor I display command

```
MODIFY RMF,DISPLAY ZZ
ERB305I ZZ: PARAMETERS
ERB305I ZZ:   VSTOR(S)  -- DEFAULT
ERB305I ZZ:   NOESS  -- DEFAULT
ERB305I ZZ:   NOFCD  -- DEFAULT
ERB305I ZZ:   CRYPTO  -- DEFAULT
ERB305I ZZ:   CACHE  -- DEFAULT
ERB305I ZZ:   IOQ(NONMBR)  -- MEMBER
```

```
ERB305I ZZ:   IOQ(NOGRAPH)  -- MEMBER
ERB305I ZZ:   IOQ(NOCOMM)  -- MEMBER
ERB305I ZZ:   IOQ(NOUNITR)  -- MEMBER
ERB305I ZZ:   IOQ(NOCHRDR)  -- MEMBER
ERB305I ZZ:   IOQ(DASD)  -- MEMBER
ERB305I ZZ:   IOQ(NOTAPE)  -- MEMBER
ERB305I ZZ:   SYSOUT(Z)  -- MEMBER
ERB305I ZZ:   NOOPTIONS  -- MEMBER
ERB305I ZZ:   REPORT(REALTIME)  -- MEMBER
ERB305I ZZ:   RECORD  -- MEMBER
ERB305I ZZ:   NOEXITS  -- MEMBER
ERB305I ZZ:   CYCLE(1000)  -- MEMBER
ERB305I ZZ:   NOSTOP  -- MEMBER
ERB305I ZZ:   SYNC(SMF)  -- MEMBER
ERB305I ZZ:   NOENQ  -- MEMBER
ERB305I ZZ:   NOTRACE  -- MEMBER
ERB305I ZZ:   PAGESP  -- MEMBER
ERB305I ZZ:   DEVICE(NONMBR)  -- MEMBER
ERB305I ZZ:   DEVICE(NOGRAPH)  -- MEMBER
ERB305I ZZ:   DEVICE(COMM)  -- MEMBER
ERB305I ZZ:   DEVICE(NOUNITR)  -- MEMBER
ERB305I ZZ:   DEVICE(NOCHRDR)  -- MEMBER
ERB305I ZZ:   DEVICE(DASD)  -- MEMBER
ERB305I ZZ:   DEVICE(NOTAPE)  -- MEMBER
ERB305I ZZ:   DEVICE(NOSG)  -- MEMBER
ERB305I ZZ:   WKLD -- MEMBER
ERB305I ZZ:   CHAN  -- MEMBER
ERB305I ZZ:   PAGING  -- MEMBER
ERB305I ZZ:   CPU  -- MEMBER
```

Modify Monitor I gathering option

If you want to change a specific option, you can also use the modify console command, as Example 3-24 shows.

Example 3-24 Enabling the FCD gathering

```
MODIFY RMF,MODIFY ZZ,FCD
ERB104I ZZ: MODIFIED
```

How to stop the RMF Control session

To stop the RMF Control session and Monitor II and Monitor III gatherer, simply use the purge command as in Example 3-25.

Example 3-25 Extract of the console log when purging RMF

```
STOP RMF
ERB102I AA: TERMINATED
ERB803I III: MONITOR III DATA SET SUPPORT TERMINATED
ERB102I III: TERMINATED
ERB102I ZZ: TERMINATED
ERB451I RMF: SMF DATA BUFFER TERMINATED
ERB102I RMF: TERMINATED
```

Parmlib member of Monitor I

RMF ships the ERBRMF00 default parmlib member for Monitor I. If you modify the member, we recommend that you keep the following settings:

- SYNC(SMF) option, to keep RMF synchronized with SMF interval.
- Ensure that cache control data is gathered only on one system. You can use symbols, described in “Cache and ESS data gathering” on page 119.
- RECORD option, to write measured data to SMF records.

If you customize the member ERBRMF00 in SYS1.PARMLIB, keep in mind that with an upgrade to the next release, your changed member is overwritten by the new ERBRMF00 member. Therefore. we recommend that you create a copy of the member before customizing it.

3.3 Monitor II

This section discusses how to set up the Monitor II background session to collect SMF data to create Postprocessor reports.

For a detailed discussion about the Monitor II background session option, check *RMF User’s Guide*, Chapter 12.

3.3.1 Customization of the Monitor II background session

The Monitor II background session collects SMF records for archiving and processing with the Postprocessor. RMF provides the ERBRMF01 default parmlib member for Monitor II.

The following options should be reviewed:

- The supplied parmlib member includes the option STOP(30M), that means that after 30 minutes, the background gathering will stop. So to keep your background session running, use the NOSTOP option.
- Use the NOOPTIONS parameter at the start of a session to avoid printing an options list at the operator’s console, or when modifying options to avoid requiring an operator to reply.
- To collect SMF records, use the RECORD parameter.
- Monitor II offers two modes for the session reports: Delta mode and total mode, controlled by the DELTA/NODELTA parameter.
  - Total mode: The report shows the cumulative total since the beginning of the Monitor II interval.
  - Delta mode: The report shows the change in the activity since the previous request for the report.
- Data collection for the ILOCK - IRLM Long Lock Detection report is initiated by the operator, who enters the following command at the console for one system in the sysplex:

  ```
  MODIFY irlmid,RUNTIMEO
  ```

  The command is propagated automatically to all other systems.

Monitor II writes SMF Type 79 records. Specify them in SMFPRMxx parmlib member if you want to collect them, as shown in “Relationship to SMF” on page 118. Ensure also that the user has sufficient authorization, as described in “Controlling access to RMF data for the sysplex data services” on page 116.

A discussion of how to create Postprocessor reports is in 2.5, “Postprocessor” on page 32.

If you customize the member ERBRMF01 in SYS1.PARMLIB, keep in mind that with an upgrade to the next release, your changed member is overwritten by the new ERBRMF01 member. Therefore, we recommend that you create a copy of the member before customizing it.

3.3.2 Starting the Monitor II background session

In order to start the Monitor II background session, you need to have the RMF control session running.

Use the modify console command to start the Monitor II background session. The default parmlib member ERBRMF01 is used when no other member is specified as start parameter. Example 3-26 shows how to start the Monitor II background session and use the ERBRMF20 parmlib member.

Example 3-26 Start Monitor II background session

```
MODIFY RMF,START AA,MEMBER(20)
IEE252I MEMBER ERBRMF20 FOUND IN SYS1.PARMLIB
ERB100I AA: ACTIVE
```

In order to adjust parameters, you can also modify a previously started Monitor II background session. In Example 3-27, the ARD option is enabled using the modify console command. In this case, we used AA, but any two character string can be used for the Monitor II background session.

Example 3-27 Modify the Monitor II background session options

```
MODIFY RMF,MODIFY AA,ARD
 ERB104I AA: MODIFIED
```

You can view the current active Monitor II background session options using the display console command:

```
MODIFY RMF,DISPLAY AA
```

3.4 Monitor III

Here we describe how to set up the Monitor III gatherer.

For a detailed discussion about the Monitor III gatherer option, refer to *RMF User’s Guide*, Chapter 13.

3.4.1 Customization of the Monitor III gatherer

The gatherer options are stored in an ERBRMFxx parmlib member. The ERBRMF04 is the default parmlib member. You can adapt this member to your needs. Here we discuss some additional details that we would like you to consider during your customization. The Monitor III gatherer can store the gathered data in a buffer, and also in VSAM data sets. For example, you can keep data available for a week for later examination. You can also archive these VSAM data sets for later processing.

NOOPTIONS parameter

Use the NOOPTIONS parameter at the start of a session to avoid printing an options list at the operator’s console, or when modifying options to avoid requiring an operator to reply.

Monitor III gatherer MINTIME

The MINTIME option specifies, in seconds, the length of the gathering interval. At the end of this interval, the data gatherer combines all samples it has gathered into a set of samples. The data reporter can summarize the combined samples at the end of each MINTIME interval. The default is 100, but we recommend that you use 60 seconds. The smallest time interval that the data reporter can report on is 10 seconds, but this increases the gathering overhead. Keep in mind, the MINTIME controls the data interval for the Monitor III ISPF Session, RMF PM, and the RMF data on-demand in a Web browser.

Important: For sysplex reporting, use the same MINTIME value for all systems in the sysplex to provide correct sysplex reporting.

z/OS UNIX file systems data gathering

This data gathering is required to create the File System Statistics part of the HFS Postprocessor report. You have to specify the UNIX HFS names you want to monitor.

For example, in order to monitor the following HFS:

```
ZOSR04.OMVS.ROOT
ZOSR04.OMVS.VAR
ZOSR04.OMVS.ETC
```

Specify the following parameter:

```
HFSNAME(ADD(ZOSR06.OMVS.ROOT))
HFSNAME(ADD(ZOSR06.OMVS.VAR))
HFSNAME(ADD(ZOSR066.OMVS.ETC))
```

VSAM RLS activity

The VSAMRLS option controls the collection of VSAM RLS activity data. When you specify VSAMRLS or allow the default value to take effect, activity data is gathered for VSAM RLS by storage class. In addition, you can specify data set masks to collect data by VSAM spheres, too. You can specify up to 25 different data set masks at a time.

Let’s assume that you want to gather data for the following VSAM clusters:

```
PROD.IMS.FILEA
PROD.IMS.IMS1
PROD.IMS.IMS2
SSHR.CICS.CICS1.FILEA
SSHR.CICS.CICS1.FILEB
APPL1.IMS.FILEA
```

Specify the following parameter:

```
VSAMRLS(ADD(PROD.IMS.*)
VSAMRLS(ADD(SSHR.CICS.CICS1.FILE*)
VSAMRLS(ADD(APPL1.IMS.FILEA)
```

Note: Since VSAM RLS Activity by VSAM spheres is a sysplex-wide report, activate the same set of data set masks on all systems in the sysplex.

Cache data gathering

There is a data loss issue with collecting CACHE data only on one system in a sysplex. If that system goes down for some reason, then the cache data is lost. It may be better that the cache data collected is the same on all the systems and that you choose to collect the cache data on one, some, or all systems depending upon your installation tolerance for data loss. The Washington System Center recommendation is to collect cache data on two systems.

Coupling Facility data gathering

The CFDETAIL parameter controls the collection of data about the Coupling Facility (CF). If this option is active, detailed data about activities in the structures (LIST, LOCK, and CACHE) of the CF are stored in the set of samples and are available in the Coupling Facility Activity report. The default is NOCFDETAIL, which suppresses detailed data gathering for the Coupling Facility. To start collection, specify CFDETAIL when starting or modifying the Monitor III session. With CFDETAIL, a large amount of data is gathered that enables you to get many details about the usage of each structure in the coupling facility. Consider that this data gathering is done only on one member of the sysplex. This is called sysplex master gathering and reduces performance overhead on non-master members and the amount of data in SMF records. The RMF Sysplex Data Server determines internally which member of the sysplex is the master.

There are scenarios where it is important for you to determine which member of the sysplex becomes the master system, for example, when not all systems are connected to all CFs, but you want to have details from all CFs. Then it is necessary that the master is connected to all CFs. With the following procedure, you can select the master system by manual intervention:

1. Select a system which is connected to all CFs. Choose one system, where the RMF release is the highest available release of RMF in your sysplex.
2. Stop the Monitor III data collector RMFGAT temporarily on all systems, except the one determined in step 1.
3. Restart RMFGAT on these systems now. The system selected in step 1 is the master system until it becomes offline or RMFGAT is stopped on this system.

For a complete Coupling Facility Activity report, we recommend that you combine data from all of the systems in the sysplex. If data from one or more systems is missing, the Structure and Subchannel Activity sections of the report are incomplete. In addition, the PRIM (primary) and SEC (secondary) indicators of duplexed structures might be missing in the Usage Summary section because this information is gathered only on one member of the sysplex (sysplex master gathering).

Monitor III VSAM data sets

The Monitor III data gatherer writes records (sets of samples) to a large storage buffer, or optionally, to user-defined VSAM data sets. Without the VSAM data sets, the data is overwritten as soon as the buffer is filled. If you define VSAM data sets, you can save large amounts of information, and you can reuse the VSAM data sets as RMF continuously records data over time.

You can define up to 100 data sets for use with the data gatherer. Define at least two data sets because the gatherer deletes all data in a data set before writing to it, so a single data set is emptied immediately after it is filled. RMF is limited to keep indices for about 1100 sets of samples in one VSAM data set, regardless of its physical size.

If you have a MINTIME of 60 seconds, this means that if the VSAM data set is big enough to keep 1100 sets of samples, it can keep at least 18 to 20 hours. The size of one set of samples depends on the sysplex/system configuration, for example, the number of devices, channels, defined workloads, number of actual delayed jobs, and so forth.

You have to specify the name of the VSAM data set in the Monitor III parmlib member. To keep one common member, we recommend using names with symbolic substitution. For example, data set names with a symbolic for the system name, such as SYS1.RMF.&SYSNAME..DS1 can be used for multiple systems in your sysplex.

You can use the Clist ERBVSDEF, shipped in SYS1.SERBCLS, to define VSAM data sets.

- Try first a size of 60 cylinders per data set.
- Define two data sets for one system. Use the ERBVSDEF TSO command, as follows:

```
erbvsdef 'sys1.rmf.sys1.ds1' vsamvol(hg670a) tracks(900)
erbvsdef 'sys1.rmf.sys1.ds2' vsamvol(hg670a) tracks(900)
```

- Add the data sets to Monitor III using these console commands:

```
f rmf,f III,ds(add(sys1.rmf.sys1.ds1))
f rmf,f III,ds(add(sys1.rmf.sys1.ds2))
```

- Then start to use them by issuing the console command:

```
f rmf,f III,ds(start)
```

To verify that your VSAM data set is an appropriate size, check the Data Set report of Monitor III Sysplex reports. The Data Index report in Example 3-28 shows the date and time of the earliest and latest data stored in each data set.

Example 3-28 Monitor III Data index report

```
                        RMF V1R5   Data Index - SYSXPLEX          Line 1 of 14
Command ===>                                                     Scroll ===> CSR

Samples: 60      System: SYS1  Date: 11/23/04  Time: 09.26.00  Range: 60    Sec

       ----Begin/End----
System --Date-- --Time-- -DDNAME- ---------------Data Set Name----------------

SYS1   11/23/04 07.30.00 SYS00333 SYS1.RMF.SYS1.DS1
       11/23/04 09.20.00
SYS1   11/23/04 09.20.00 SYS00334 SYS1.RMF.SYS1.DS2
                09.27.00             * * *      Currently active      * * *
```

- In our case, one VSAM data set covers about 110 minutes (07.30.00-09.20.00). That is, with a MINTIME of 60 seconds, 110 sample sets. Therefore, we will use data sets 10 times larger for 1100 sample sets. That is 600 cylinders each.
- To have data for seven days, which is 1080 minutes or sample sets, we define 10 data sets per system, 600 cylinders each.

Then, we need to update the data sets definitions in the Monitor III, as follows:

- Stop the use of the data sets using an operator command:

```
f rmf,f III,ds(stop)
```

- Delete the data sets using ISPF 3.4:

```
delete 'sys1.rmf.sys1.ds1'
delete 'sys1.rmf.sys1.ds2'
```

- Define 10 data sets per system using ISPF 6. Following is an example of one of them:

```
erbvsdef 'sys1.rmf.sys1.ds1' vsamvol(hg670a) tracks(9000)
...
erbvsdef 'sys1.rmf.sys1.ds10' vsamvol(hg670a) tracks(9000)
```

Example 3-29 shows an extract of a Monitor III parmlib member. The VSAM data sets are added, using the **DATASET(ADD(...))** parameter. Finally, you have to enable the data set recording, using the **DATASET(START)** parameter. Since the symbol &SYSNAME is used, you can use it as a common member. You see how the data set name is resolved in the Monitor III Data Index report in Example 3-28.

Example 3-29 Extract of Monitor III parmlib member

```
DATASET(ADD(SYS1.RMF.&SYSNAME..DS1))
DATASET(ADD(SYS1.RMF.&SYSNAME..DS2))
...
DATASET(ADD(SYS1.RMF.&SYSNAME..DS10))
DATASET(START)
...
```

For further information about how to transfer Monitor III VSAM data sets to other systems, refer to “Transmitting Monitor III data to another location” on page 129.

3.4.2 Monitor III in-storage buffer

RMF stores the set of samples collected during a MINTIME in its own local storage buffer. If you specify data set recording during a session, RMF copies each set of samples from the local storage buffer to the currently active data set for the session. Common data items for a set of samples (such as jobname or device name) are held in tables to reduce the amount of local storage needed.

The WSTOR (value) parameter specifies, in megabytes, the maximum size of RMF’s local storage buffer for the data gatherer. The default is 32 MB. In large configurations, with a large number of devices, this value might not be enough. If you see that the number of sets of samples in the storage is too small, you can increase the WHOLD parameter.

Note: You cannot modify the **WSTOR(*value*)** option by the session command **MODIFY**.

The DATASET((WHOLD(value)) allows you to specify, in megabytes, a storage value that controls page releases in the local storage buffer, specified by the WSTOR() parameter. RMF begins to wrap around the buffer once the WSTOR value is exceeded and page releases occur. We recommend using the same size as the RMF local storage buffer, specified using the WHOLD option. WHOLD should be set equal to the WSTOR value only for environments without any storage constraints. The default is 7 megabytes.

When you use the Monitor III gatherer and want to analyze data in the past, it is good to have the data available in the in-storage buffer. If the data is not in the in-storage buffer, RMF needs to access the Monitor III VSAM data sets. So it is a performance issue. To avoid having to access the VSAM data sets, it is helpful to keep a few hours in the in-storage buffer.

3.4.3 Starting the Monitor III gatherer

To start the Monitor III gatherer, start the RMF control session by the console command **START RMF**. For further details, refer to “Starting Monitor I” on page 120.

Use the modify console command to start the Monitor III gatherer. The default parmlib member ERBRMF04 is used when no other member is specified as start parameter. Example 3-30 shows how to start the Monitor III session and use the ERBRMF30 parmlib member.

Example 3-30 Start Monitor III gatherer

```
F RMF,S III,MEMBER(30)
```

```
IEE252I MEMBER ERBRMF30 FOUND IN SYS1.parmlib
ERB115I START RMFGAT MONITOR III SESSION III
IRR812I PROFILE RMFGAT.* (G) IN THE STARTED CLASS WAS USED 529
        TO START RMFGAT WITH JOBNAME RMFGAT.
$HASP100 RMFGAT   ON STCINRDR
IEF695I START RMFGAT   WITH JOBNAME RMFGAT   IS ASSIGNED TO USER
RMFGAT  , GROUP OMVSGRP
$HASP373 RMFGAT   STARTED
IEF403I RMFGAT - STARTED
ERB105I III: DATA GATHERER ACTIVE
ERB100I III: ACTIVE
```

If you customize the member ERBRMF04 in SYS1.PARMLIB, keep in mind that with an upgrade to the next release, your changed member is overwritten by the new ERBRMF04 member. Therefore, we recommend that you create a copy of the default member before customizing it.

3.4.4 Using preallocated data sets

In this situation, the reporter retrieves data only from the preallocated data sets to the local reporter session, independent of any gatherers that are running on the various systems. It is possible to preallocate data sets created on different systems. The Data Index report shows all data available in all the data sets, with the respective system-ID.

To preallocate a data set, use the ALLOC TSO command. The following TSO commands allocate the data sets SYS1.SAVED.SYS1.DS1 and SYS1.SAVED.SYS1.DS2:

```
ALLOC FI(RMFDS00) DA(‘SYS1.SAVED.SYS1.DS1‘) SHR
ALLOC FI(RMFDS01) DA(‘SYS1.SAVED.SYS1.DS2‘) SHR
```

If you are allocating data sets from a sysplex, it is of key importance that you allocate all data sets of the sysplex to enable complete reporting. You can allocate only VSAM data sets which do not belong to an active Monitor III Gatherer session.

To free this allocation use the TSO command **free** as follows:

```
free file(rmfds00)
free file(rmfds01)
```

3.4.5 Transmitting Monitor III data to another location

You may need to transmit some of your Monitor III data sets to another user at a different location to perform additional investigation. When you plan to transfer data, keep in mind that a large amount of data may require time and may exceed your installation’s limits.

Use the following steps to transmit Monitor III data:

1. Unload your Monitor III VSAM data set SYS1.RMF.SYS1.DS1 to a sequential data set using TSO command **erbv2s** provided by RMF:

   ```
   erbv2s ‘sys1.rmf.sys1.ds1’ ‘user1.rmf.sys1.ds1.seq’ tracks(9000)
   ```

   This command uses unitname SYSDA. If your installation does not have SYSDA defined, you can use the IDCAMS **repro** command to perform this function.

2. Use TSO command **transmit** to transmit the sequential data set to the target destination:

   ```
   transmit (wtscplx1.user2) dataset(rmf.sys1.ds1.seq)
    .......
   ```

```
 INMX034I WARNING: 5400000 records transmitted.  Your installation limit is
9999999
 INMX000I 0 message and 13200 data records sent as 5446984 records to
WTSCPLX1.USER2
 INMX001I Transmission occurred on 10/23/2004 at 11:14:26.
```

3. User2 uses TSO command **receive** to receive the sequential data set:

```
receive
 Dataset USER1.RMF.SYS1.DS1.SEQ from USER1 on ????????
 Enter restore parameters or 'DELETE' or 'END' +
volume(bigfre)
 Restore successful to dataset 'USER2.RMF.SYS1.DS1.SEQ'
 Receipt notification unsuccessful +
 --------------------------------------------------------
 No more files remain for the receive command to process.
```

4. User2 uses TSO command **erbs2v** to reload the data to a VSAM data set:

```
erbs2v ‘user2.rmf.sys1.ds1.seq’ ‘sys1.rmf.sysx.ds1’ vsamvol(colvol)
```

Now User2 has your Monitor III VSAM data set. A further step is required to define the data set to the RMF application using the TSO **alloc command** as follows:

```
alloc file(rmfds00) da(‘sys1.rmf.sysx.ds1’) shr
```

Now User2 can start Monitor III and can view the same data as User1 and perform investigations.

As soon as User2 doesn’t need any further access to User1’s data, User2 should remember to free this allocation using the TSO command **free** as follows:

```
free file(rmfds00)
```

When User2 restarts Monitor III, User2 sees the active Monitor III gatherer data again.

3.5 Sysplex Data Server

The RMF Sysplex Data Server (SDS) is started when the RMF control session is started. RMF is able to store its SMF record images in a wraparound buffer, the SDS SMF buffer. You can control the size of this buffer and the SMF record types that RMF writes to it using the SMFBUF option.

You need to explicitly specify the SMF record type 79 subtype 15 for Monitor II IRLM long lock reporting.

To start RMF with SMF buffer, use the start command with parameter, as in Example 3-31. The SMF buffer is initialized with a size of 256 MB and it is specified that the RMF SMF records type 70 to 79 and 79(15) are written to it.

Example 3-31 Start RMF with an SMF buffer

```
START RMF,,,(SMFBUF(SPACE(256M),R(70:79,79(15))))
IRR812I PROFILE RMF.* (G) IN THE STARTED CLASS WAS USED 363
        TO START RMF WITH JOBNAME RMF.
$HASP100 RMF      ON STCINRDR
IEF695I START RMF      WITH JOBNAME RMF      IS ASSIGNED TO USER RMF
   , GROUP STCGROUP
$HASP373 RMF STARTED
IEF403I RMF - STARTED
```

```
ERB100I RMF: ACTIVE
IEE252I MEMBER ERBRMF00 FOUND IN SYS1.PARMLIB
ERB450I RMF: SMF DATA BUFFER INITIALIZED
ERB100I ZZ : ACTIVE
```

Tip: The Sysplex Data Server can maintain all SMF record types.

It is also possible to start or modify the SMF buffer after RMF is already started, as shown in Example 3-32.

Example 3-32 Start or modify the SMF buffer

```
MODIFY RMF,SMFBUF(SPACE(256M),R(70:79,79(15)))
ERB454I RMF: SMF DATA BUFFER SPACE MODIFICATION STARTED
ERB455I RMF: SMF DATA BUFFER RECTYPE MODIFICATION COMPLETED
```

4

Setup and customization of the new RMF facilities

This chapter discusses the steps required to get each of the “new” RMF facilities up and running, and describes the additional customization that is possible with each one:

- The RMF Spreadsheet Reporter
- The RMF Distributed Data Server
- The RMF Linux data gatherer
- RMF Performance Monitor Java edition

4.1 RMF Spreadsheet Reporter

The Spreadsheet Reporter is the powerful Microsoft Windows workstation solution for graphical presentation of long term Postprocessor data. This section describes the installation.

4.1.1 Prerequisites

We recommend that you fulfill the following prerequisites:

- Operating system: Microsoft Windows ME, 2000, or XP
- Spreadsheet program:
  You can use any spreadsheet program that is compatible with the Lotus WK1 format. However, in order to use the spreadsheet macros shipped with the Spreadsheet Reporter, one of the following products is required:
  - Microsoft Excel 2000, 2002, or 2003
  - Lotus 1-2-3 Version 9.7 or higher
- Level of access:
  If you are the administrator of this workstation, you install the application files for the Spreadsheet Reporter to the default directory C:\Program Files\RMF\IBM RMF Spreadsheet Reporter. Otherwise, you have to specify a different installation directory where you have write access.
- If you want to use the SMF buffer, ensure that your TSO user ID has sufficient access to RMF Performance data, described in “Controlling access to RMF data for the sysplex data services” on page 116.
- The Spreadsheet Reporter uses active FTP mode to exchange data with the host. Ensure this communication is allowed by firewalls, and that the FTP server on the host is running.

  To support active mode FTP, ensure that the following communication ports from the host are opened:
  - FTP server's port 21 from anywhere (Client initiates connection).
  - FTP server's port 21 to ports > 1024 (Server responds to client).
  - FTP server's port 20 to ports > 1024 (Server initiates data connection to the client).
  - FTP server's port 20 from ports > 1024 (Client responds to Server).
- To avoid FTP-JES time-outs, check the JESPUTGETGO option in your FTP.DATA member.
- The Spreadsheet Reporter submits a Postprocessor job and retrieves the job output. The FTP JESINTERFACELEVEL statement in FTP.DATA specifies the FTP-to-JES interface. When FTP JESINTERFACELEVEL 2 is specified, ensure that you have the proper access to RACF® resource class JESSPOOL to retrieve the job output. The ability to submit jobs is controlled by the JESJOBS SAF resource. For more information on SDSF security, refer to *z/OS SDSF Operation and Customization,* SA22-7670.

  To avoid JES output being deleted from JES2 Spool after JOB termination, verify your OUTCLASS definitions. The output class H is used as default by the Spreadsheet Reporter and needs the definition: OUTCLASS(H) OUTDISP(HOLD,HOLD).
- The Spreadsheet Reporter starts remote Postprocessor job execution. During this process, several data sets are allocated. You have to choose an HLQ for these data sets that is SMS managed.

4.1.2 Installation

The Spreadsheet Reporter is related to the version of the Postprocessor. Whenever there are changes in the Postprocessor that affect the Postprocessor reports, you need the appropriate Spreadsheet Reporter version that can process the changed reports. The Spreadsheet Reporter is compatible with older Postprocessor versions, but may have problems with newer Postprocessor versions.

If there are changes to reports introduced by a new z/OS version or a maintenance fix, you need to upgrade your Spreadsheet Reporter version. The new Spreadsheet Reporter version is available in the updated SERBPWSV distribution library and on the RMF Home page.

The code of the Spreadsheet Reporter is available as member ERB9R2SW of the SERBPWSV distribution library. Download ERB9R2SW.EXE as a binary file from this SERBPWSV host data set.

Tip: We recommend that you get the latest version of the Spreadsheet Reporter, which is available on the RMF Home page at:

`http://www.ibm.com/servers/eserver/zseries/zos/rmf/`

Start the installation by executing the file: ERB9R2SW.EXE. The dialog guides you through the installation.

Different installation types are available:

- Typical: Install all application files and all spreadsheet macros for Microsoft Excel.
- Compact: Install only the application files, without any spreadsheet macros. Select this installation type if you just want to use the Spreadsheet Reporter as a remote Postprocessor execution and download utility.
- Custom: You can choose the components that you want to install:
  - Spreadsheet Reporter application files
  - Lotus 1-2-3 spreadsheet macros
  - Microsoft Excel spreadsheet macros

Specify the directory where you want to install the Spreadsheet Reporter. The default is: C:\Program Files\RMF\RMF Spreadsheet Reporter.

Specify the resource directory where you want the Spreadsheet Reporter to place the resources, for example, Report Listings, macros, or Working Sets. You can accept the default or specify another folder. The default is: C:\Documents and Settings\Administrator\Application Data\RMF\ RMF Spreadsheet Reporter.

Note: You cannot change the resource directory later, so be certain that you have sufficient disk space on the target drive.

4.1.3 Starting the Spreadsheet Reporter

To start the Spreadsheet Reporter, select **Start** → **Programs** → **IBM RMF Performance Management** → **RMF Spreadsheet Reporter**.

For details about using Spreadsheet Reporter, refer to “Spreadsheet Reporter” on page 40.

4.2 RMF Distributed Data Server

The Distributed Data Server (DDS) is a single data server on one system in the sysplex. The DDS gathers data from Monitor III distributed on all systems in the sysplex. If you want to monitor several systems in a sysplex, you must set up a DDS host session on that system in the sysplex with the highest RMF release. Each system of the sysplex must have an active and synchronized Monitor III gatherer.

If you want to monitor several sysplexes, each one needs to have an active DDS and an active and synchronized Monitor III gatherer. For information on how to start and customize Monitor III gatherer, refer to “Customization of the Monitor III gatherer” on page 124.

RMF PM takes its input data from the DDS. You can also access the performance data from the DDS using the RMF data on demand in a Web browser.

4.2.1 Setting up the Distributed Data Server

This section describes how to set up and start the DDS.

General setup tasks, like authorization of the RMF libraries and the assignment of user IDs to the started tasks, are discussed in “Basic customization” on page 112 and “Advanced RMF configuration” on page 115.

Considerations for z/OS UNIX level of security

If the BPX.DAEMON FACILITY resource is defined, your system has z/OS UNIX security and can exercise more control over your superusers. Because DDS runs as a daemon, it must have access to the BPX.DAEMON facility. You need to define all programs loaded by GPMSERVE to PROGRAM CONTROL. Example 4-1 shows the definitions for Security Server (RACF) for the defined user ID GPMSERVE, which is assigned to the started task GPMSERVE.

Example 4-1 TSO commands for Security Server (RACF)

```
PERMIT BPX.DAEMON CLASS(FACILITY) ID(GPMSERVE) ACCESS(READ)
RDEFINE PROGRAM GPM* ADDMEM('SYS1.SERBLINK'\//NOPADCHK) UACC(READ)
RDEFINE PROGRAM ERB* ADDMEM('SYS1.SERBLINK'\//NOPADCHK) UACC(READ)
RDEFINE PROGRAM CEEBINIT ADDMEM('CEE.SCEERUN'\//NOPADCHK) UACC(READ)
RDEFINE PROGRAM IEEMB878 ADDMEM('SYS1.LINKLIB'\//NOPADCHK) UACC(READ)
SETROPTS WHEN(PROGRAM) REFRESH
```

Customization

On all systems that you want to monitor, you need to start the Monitor III gatherer with identical MINTIME and SYNC options. For customization details of the Monitor III gatherer, refer to “Customization of the Monitor III gatherer” on page 124.

Also, make sure that the following prerequisites are met on your z/OS host:

- UNIX System Services is configured.
- TCP/IP under UNIX System Services is configured and active.

RMF provides a default parmlib member GPMSRV00, shown in Example 4-2. You can tailor this according to your needs, but we recommend you use the standard GPMSRV00 member.

Example 4-2   GPMSRV00 default parmlib member

```
CACHESLOTS(4)       /* Number of timestamps in CACHE     */
DEBUG_LEVEL(0)      /* No informational messages     */
SERVERHOST(*)       /* Dont bind to specific IP-Address */
MAXSESSIONS_INET(5) /* MaxNo RMF PM clients     */
SESSION_PORT(8801)  /* TCP/IP port number RMF PM     */
TIMEOUT(0)          /* No timeout     */
DM_PORT(8802)       /* Port Number for DM requests     */
DM_ACCEPTHOST(*)     /* Accept from all IP-addresses     */
MAXSESSIONS_HTTP(20) /* MaxNo of concurrent HTTP requests */
HTTP_PORT(8803)     /* Port number for HTTP requests     */
HTTP_ALLOW(*)       /* Mask for hosts that are allowed */
HTTP_NOAUTH()       /* No server can access without auth.*/
```

Ensure that the ports used to access DDS are open for communication:

- RMF PM uses the SESSION_PORT (default 8801) and can optionally also use the HTTP_PORT (default 8803).
- RMF data on demand in a Web-browser uses the HTTP_PORT (default 8803).

Example 4-3 shows how to start the DDS using the default GPMSERVE procedure of SYS1.PROCLIB. If no member parameter is specified, the default parmlib member GPMSRV00 is used.

Example 4-3   Starting the DDS

```
S GPMSERVE
IRR812I PROFILE GPMSERVE.* (G) IN THE STARTED CLASS WAS USED 292
        TO START GPMSERVE WITH JOBNAME GPMSERVE.
$HASP100 GPMSERVE ON STCINRDR
IEF695I START GPMSERVE WITH JOBNAME GPMSERVE IS ASSIGNED TO USER GPMSERVE
   , GROUP SYS1
$HASP373 GPMSERVE STARTED
IEF403I GPMSERVE - STARTED
IEE252I MEMBER GPMSRV00 FOUND IN SYS1.IBM.PARMLIB
GPM060I RMF DISTRIBUTED DATA SERVER READY FOR COMMANDS
```

DDS host trace

DDS allows you to set up a trace for further problem determination. You have to modify the GPMSERVE start procedure. Replace the SYSOUT and SYSPRINT statements either with data set definitions or sysout class definitions, as in Example 4-4.

Example 4-4   Modified GPMSERVE procedure

```
//GPMSERVE PROC MEMBER=00
//STEP1    EXEC PGM=GPMDDSRV,REGION=0M,TIME=1440,
//         PARM='TRAP(ON)/&MEMBER'
//GPMINI   DD   DISP=SHR,DSN=SYS1.SERBPWSV(GPMINI)
//GPMHTC   DD   DISP=SHR,DSN=SYS1.SERBPWSV(GPMHTC)
//CEEDUMP  DD   DUMMY
//SYSPRINT DD DISP=(NEW,CATLG),UNIT=SYSDA,SPACE=(TRK,(1000,500),RLSE),
//          DSN=SYS1.RMF.PTRACE
//SYSOUT   DD DISP=(NEW,CATLG),UNIT=SYSDA,SPACE=(TRK,(1000,500),RLSE),
//         DSN=SYS1.RMF.OTRACE
//         PEND
```

Use the **modify** command with parameter TRACEON to enable the trace, as shown in Example 4-5. To stop the trace, use the TRACEOFF parameter.

Example 4-5 Enable the DDS trace

```
F GPMSERVE,TRACEON
GPM052I TRACE IS NOW ON
F GPMSERVE,TRACEOFF
GPM052I TRACE IS NOW OFF
```

The DDS trace produces a lot of output. Therefore, you should ensure that your trace data sets are big enough and that you only create the trace when it is really necessary.

Also, the user assigned to the GPMSERVE started task must have sufficient authorization. For details, refer to “Assign started task procedures to user IDs” on page 115, “Controlling access to RMF data for the sysplex data services” on page 116 and “Considerations for z/OS UNIX level of security” on page 136.

4.3 RMF Linux data gatherer

The RMF Linux data gatherer (rmfpms) is a modular data gatherer for Linux on zSeries and Intel Linux. This section describes the installation and the customization of rmfpms and the environment. You can access the performance data from rmfpms using RMF PM or the RMF Web browser interface. The rmfpms performance data is automatically archived for reuse later.

4.3.1 Installation

rmfpms is available via the RMF Home page at:

http://www.ibm.com/servers/eserver/zseries/zos/rmf/

You have to get the appropriate version for your Linux image and install rmfpms on each Linux image you want to monitor.

rmfpms is available for:

- zSeries Linux
  - Kernel 2.4 - 31 bit
  - Kernel 2.6 - 31 bit
  - Kernel 2.4 - 64 bit
  - Kernel 2.6 - 64 bit
- Intel Linux
  - Kernel 2.4
  - Kernel 2.6

There are several versions of rmfpms available for your combination of Linux kernel, GCC libraries, and for 31 bit and 64 bit. For new Linux distributions, use the Kernel 2.4 or Kernel 2.6 version.

Example 4-6 shows you how to get the version of a Linux distribution.

Example 4-6 Linux version

```
lnxsu4:~ # cat /proc/version
 Linux version 2.4.21-217-default (root@s390z07) (gcc version 3.2.2) #1 SMP Wed
May 5 10:29:13 UTC 2004
```

If you have problems with your Linux distribution or don't know which version to use, contact the RMF team on the RMF Home page.

No *root* user ID is needed to install and start the rmfpms. Download the appropriate rmfpms from the RMF Home page and unpack it, using the Linux command:

tar xvfz *archive-name*
This installs rmfpms into a new *rmfpms* subdirectory.

Installation scenario

User *mailand* uses Linux kernel 2.4 on 64 bit. User mailand needs to download the file rmfpms_s390x_kernel26.tgz. It is recommended to store it in the home directory, so mailand saves the file into /home/mailand/rmfpms_s390x_kernel24.tgz.

To unpack the file, mailand uses the command:

```
tar xvfz rmfpms_s390x_kernel24.tgz
```

It creates an rmfpms directory and unpacks the files to this directory. Now rmfpms is successfully installed in the directory /home/mailand/rmfpms.

4.3.2 Customization

rmfpms uses three configuration files:

- .rmfpms_config in the rmfpms directory.
- gpmsrv00.ini in the rmfpms/bin directory.
- gpmexusr.ini in the rmfpms/bin directory.

You can customize the default configuration file to your needs.

If you do not install rmfpms in your home directory, or if you choose a different repository directory for the rmfpms performance data, you need to customize the .rmfpms_config configuration file.

We recommend not to have the rmfpms data repository in the root file system. If you have started the **enable_autostart** script and you have a separate disk for the /var file system, this is done automatically. This directory is used to store all historical information. It can use about 20 KB or more per day. We recommend that you remove historical data, otherwise you will fill up some Linux file systems completely and run out of space on your system. Remember that the operating system itself needs to access the root file system in order to run. Also, most Linux applications need write access to their own directories; otherwise, they may terminate because there is no space left on the device.

In the file gpmsrv00.ini, you specify rmfpms settings such as the communication port that you use.

rmfpms provides exception handling. You can specify thresholds for metrics in the gpmexusr.ini configuration file. RMF PM and RMF Web browser interface also can show you visually when the threshold is reached.

The .rmfpms_config configuration file

Example 4-7 shows the default .rmfpms_config configuration file.

Example 4-7 Configuration file .rmfpms_config

```
# rmfpms_config - included in rmfpms bash shell script
#
# 11/14/2000, 07/26/2001 Oliver Benke
# (c) IBM Deutschland Entwicklung GmbH, IBM Corp.
#
# configuration parameters
export IBM_PERFORMANCE_REPOSITORY=$HOME/rmfpms/.rmfpms
export IBM_PERFORMANCE_HOME=$HOME/rmfpms/bin/
export IBM_PERFORMANCE_MINTIME=60
export LD_LIBRARY_PATH=$IBM_PERFORMANCE_HOME:$LD_LIBRARY_PATH
export APACHE_ACCESS_LOG=/var/log/httpd/access_log
export APACHE_SERVER=localhost
export APACHE_SERVER_PORT=80
```

If you decide to install rmfpms in a directory that is not your home directory, you have to modify the variables that use the $HOME symbol in the configuration file.

- IBM_PERFORMANCE_REPOSITORY points to the repository files in the rmfpms/.rmfpms directory. Performance data and log files are written into the repository directory. We recommend that you have the rmfpms data repository directory on its own disk drive.

  Example scenario: The disk /dev/dasdx is mounted to the mountpoint /rmfdata, so we must specify:
  `IBM_PERFORMANCE_REPOSITORY =/rmfdata`
- IBM_PERFORMANCE_HOME points to the rmfpms binary files in the rmfpms/bin directory.

We recommend that you use the default data gatherer interval of 60 seconds, specified by the option IBM_PERFORMANCE_MINTIME.

If you want to access Apache HTTP server performance data, you have to verify the following parameters:

- APACHE_ACCESS_LOG points to the location of the Apache log file.
- APACHE_SERVER is the host name of the Apache server.
- APACHE_SERVER_PORT is the port of the Apache server; the default is 80.

You need to enable the Apache HTTP server statistic data gathering, described in “Additional configuration” on page 143. RMF currently does not have the capability to monitor more than one Apache HTTP server.

The gpmsrv00.ini file

The gpmsrv00.ini configuration file contains a subset of the parameters of GPMSRVxx parmlib member of the DDS. Example 4-8 shows the default gpmsrv00.ini configuration file with the supported parameters.

Example 4-8 Default gpmsrv00.ini configuration file

```
CACHESLOTS(4)                /* Number of timestamps in CACHE     */
DEBUG_LEVEL(3)                  /* All informational messages        */
SERVERHOST(*)                   /* Don't bind to specific IP-Address */
MAXSESSIONS_HTTP(20)            /* MaxNo of concurrent HTTP requests */
HTTP_PORT(8803)                 /* Port number for HTTP requests     */
HTTP_ALLOW(*)                   /* Mask for hosts that are allowed   */
HTTP_NOAUTH()                   /* No server can access without auth.*/
```

Ensure that the HTTP port (default 8803) used by the RMF PM and RMF data on demand in a Web browser to access rmfpms is open for communication.

We recommend that you use the default configuration. For detailed information about the parameters, refer to *RMF User's Guide*, SC33-7990.

The gpmexusr.ini file

Within the gpmexusr.ini file in the rmfpms/bin directory, you can define exceptions for metrics. If this file does not exist, you have to create it. You define the *critical* trigger and *warning* trigger, using thresholds. When the metric reaches the critical or warning threshold, RMF PM and RMF data on demand in a Web browser are able to display the situation. If the metric matches the warning state, the metric value is displayed in the color yellow. If the metric matches the critical state, the metric value is displayed in the color red.

The definition is created in an XML format.

You need to specify:

Exception type entry User type.

Exception entry id and name for your exception.

Metric entry Metric id in hex format for which you want to define the exception. You can also use multiple metrics.

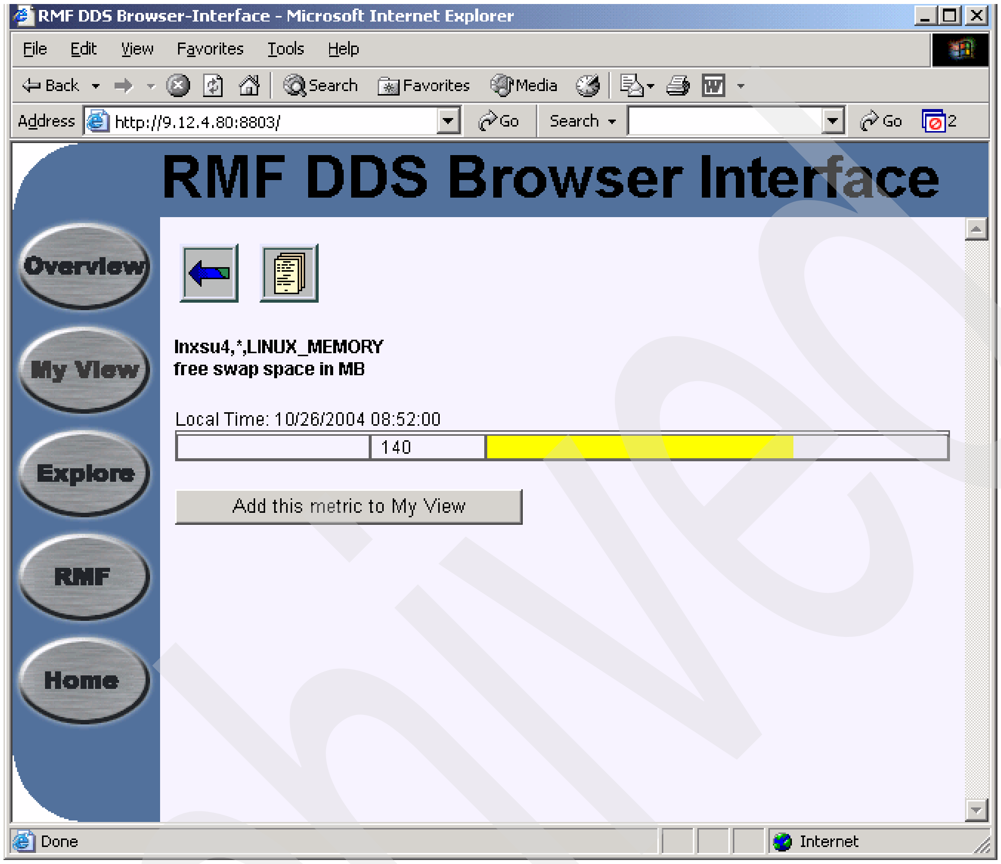
Trigger entry Critical and warning trigger type. Specify a lower threshold (le = less or equal) or an upper threshold (ge = greater or equal) to compare to the metric.

Example 4-9 shows the definition of a user exception in the file rmfpms/bin/gpmexusr.ini. The User ID is MyException and the name is Free Swap Space. The exception is specified for the metric `0x405090, free swap space in MB`. The trigger type *warning* is reached when the metric is less than or equal to the threshold 150 MB. The trigger type *critical* is reached when the metric is less than or equal to the threshold 50 MB.

Example 4-9 Exception definition file gpmexusr.ini

```
<exceptions type="user">
  <exception id="MyException" name="Free Swapp Space">
    <metrics>
      <metric>0x405090</metric>
    </metrics>
    <triggers>
      <trigger type="critical" le="50"/>
      <trigger type="warning"  le="150"/>
    </triggers>
  </exception>
</exceptions>
```

Figure 4-1 shows the result using RMF data on demand in a Web browser when the metric `free swap space in MB` has reached the warning trigger (140 MB is less than or equal to warning trigger of 150 MB), so it is displayed in the warning color (yellow).



RMF DDS Browser-Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://9.12.4.80:8803/
Go Search Go
RMF DDS Browser Interface
Overview
My View
Explore
RMF
Home
Add this metric to My View
Done
Internet

lnxsu4,*,LINUX_MEMORY
free swap space in MB

Local Time: 10/26/2004 08:52:00

140

Figure 4-1 Metric matches exception

To get an XML document with the complete list of available metrics and the metric id, open your Web browser and simply type the URL:

```
http://<hostname>:8803/gpm/config/index.xml
```

where the *hostname* is the host name or IP address of the host where the DDS or rmfpms is installed. Example 4-10 shows an extract of the index.xml file for Linux.

- The ibmgpm:metric entry describes the metric:
  - The id tag specifies the metric id in hex format.
  - The description tag provides a description.

Example 4-10 Extract of the index.xml for Linux

```
<ibmgpm:metric id="402050" description="% cpu idle time" metric-type="SINGLE" workscopes="G" helpurl=""
helpid="9132" listtype="" />
  <ibmgpm:metric id="4020A0" description="% cpu idle time by processor" metric-type="LIST" workscopes="G"
helpurl="" helpid="9132" listtype="C" />
  <ibmgpm:metric id="402030" description="% cpu time in kernel mode" metric-type="SINGLE" workscopes="G"
helpurl="" helpid="9134" listtype="" />
...
```

Additional configuration

You need additional customization to enable the performance data gathering of your Apache HTTP server and of DASDs on zSeries.

Apache HTTP server configuration

In the httpd.conf file of your Apache HTTP server, you need to enable the statistics module in order to make the data available in rmfpms. Depending on your distribution, you may also need to adapt the sysconfig file named apache.

You have to customize the httpd.conf configuration file, which is stored in the directory /etc/httpd. Load, add the statistics module, and enable access to the statistics data, as shown in Example 4-11.

Example 4-11 Extract of configuration file httpd.conf

```
# Example:
# LoadModule foo_module libexec/mod_foo.so
LoadModule status_module      /usr/lib/apache/mod_status.so
...
#  Reconstruction of the complete module list from all available modules
#  (static and shared ones) to achieve correct module execution order.
#  [WHENEVER YOU CHANGE THE LOADMODULE SECTION ABOVE UPDATE THIS, TOO]
ClearModuleList
AddModule mod_status.c
...
#
  Allow server status reports, with the URL of http://servername/server-status
# Change the ".your-domain.com" to match your domain to enable.
#
# Note: apache is started (by /etc/init.d/apache) with -D STATUS if
# HTTPD_SEC_ACCESS_SERVERINFO is set to "yes" in
# /etc/sysconfig/apache.

<IfDefine STATUS>
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from none
    Allow from localhost
</Location>
```

For some distributions, you may need to edit the /etc/sysconfig/apache configuration file and set `HTTPD_SEC_ACCESS_SERVERINFO` to `yes`, as shown in Example 4-12.

Example 4-12 Extract of configuration file apache

```
#
# /etc/sysconfig/apache:
#
# enable status module (yes|no)
#
HTTPD_SEC_ACCESS_SERVERINFO=yes
...
```

Monitoring DASD statistics on zSeries mainframes

You can enable DASD performance gathering on zSeries by entering the command with root authority:

```
echo set on > /proc/dasd/statistics
```

To turn it off, enter the command:

```
echo set off > /proc/dasd/statistics
```

If DASD performance gathering is turned off, the following metrics are not available:

- dasd io average response time per request
- dasd io average response time per sector
- dasd io requests per second

4.3.3 Start RMF Linux data gatherer

To start rmfpms, you can use the rmfpms shell script in the rmfpms/bin directory with **start** parameter, as in Example 4-13. You need to start rmfpms on each Linux image you want to monitor.

Example 4-13 Start rmfpms

```
mailand@lnxsu4:~> pwd
/home/mailand
mailand@lnxsu4:~> rmfpms/bin/rmfpms start
Starting performance gatherer backends ...
 DDSRV: RMF-DDS-Server/Linux-Beta (Sep 10 2004) started.
 DDSRV: Functionality Level=2.008
 DDSRV: Reading exceptions from gpmexsys.ini and gpmexusr.ini.
DDSRV: Server will now run as a daemon process.
done!
```

To end rmfpms, use the stop parameter to call the rmfpms shell script:

```
rmfpms/bin/rmfpms stop
```

You may get an error message if you want to restart the rmfpms after abnormal termination. The error message tells you that the gatherer backends were already started. The reason for this error message is that some files may have been left open if rmpfpms was abnormally terminated. Therefore, you have to start rmfpms using the **restart** parameter, as shown in Example 4-14.

Example 4-14 Restart rmfpms

```
mailand@lnxsu4:~> rmfpms/bin/rmfpms start
Performance gatherer backends already started
mailand@lnxsu4:~>
mailand@lnxsu4:~> rmfpms/bin/rmfpms restart
Stopping performance gatherer backends ...
rmfpms/bin/rmfpms: line 33: kill: (28497) - No such process
gpmddsrv: no process killed
done!
rmfpms/bin/rmfpms: line 47: /proc/dasd/statistics: Permission denied
Starting performance gatherer backends ...
 DDSRV: RMF-DDS-Server/Linux-Beta (Sep 10 2004) started.
 DDSRV: Functionality Level=2.008
 DDSRV: Reading exceptions from gpmexsys.ini and gpmexusr.ini.
DDSRV: Server will now run as a daemon process.
done!
```

To get the status of rmfpms, use the command:

```
rmfpms/bin/rmfpms status
```

You can start a few of the gatherer modules instead of all of them, if you do not need all of them. To do this, change the rmfpms/bin/rmfpms startup shell script.

Here is the list of all the data gathers:

- apachegat: Collect Apache HTTP server data
- dasdgat: Collect DASD data
- filegat: Collect file system data
- gengat: Collect system data
- netgat: Collect network data
- procgat: Collect process data

Log files are written into the log subdirectory of the repository directory, for example:

/home/mailand/rmfpms/.rmfpms/logs

Start rmfpms at system IPL/boot time

We recommend that you start rmfpms automatically at system startup.

To do this, execute the script **enable_ autostart** in the rmfpm directory. The script installs rmfpms in the directory /opt/rmfpms and the performance repository in /var/opt/rmfpms/.rmfpms/archive. The script uses the chkconfig utility of your Linux distribution to start rmfpms automatically at system startup time.

If you need to customize rmfpms, you have to edit the configuration files in /opt/rmfpms.

To enable autostart, change to the rmfpms installation directory by using the command:

```
./enable_autostart
```

Storing and archiving data

We store the performance data in a subdirectory of the rmfpms repository directory. All performance data for one day is stored in one subdirectory, named according to the date. For example, the data from 10/16/2004 is stored in rmfpms/.rmfpms/20041016.

We store the archived data in the archive subdirectory of the rmfpms repository directory, for example, rmfpms/.rmfpms/archive. RMF provides two shell scripts for easy archiving of performance data. The tools are located in the rmfpms/bin subdirectory. The archive shell script is called automatically every 1600 gatherer intervals. The default time between two archive runs with the default MINTIME of 60 seconds is 1600 minutes, which is 26:40 hours.

You can retrieve the archived data for further analysis using RMF PM or RMF data on demand in a Web browser.

To avoid running out of space in the file system because of the archived data, you should delete older archived data. You can delete historical data either manually or by using the script rmfpms/bin/delete_old_perfdata, as shown in Example 4-15.

Example 4-15 Script delete_old_perfdata

```
#!/bin/bash
#
# delete_old_perfdata
#
# delete rmfpms performance data which is older than 7 days
#
# 10/28/2004 Oliver Benke
# (c) IBM Deutschland Entwicklung GmbH, IBM Corp.
#
#
# change to rmfpms bindir
if [ "$0" = "delete_old_perfdata" ]
then
        bindir=$PWD
else
        temp=$0XIXBXM
        bindir=${temp%\/delete_old_perfdataXIXBXM}
fi
cd $bindir
#
# read config file
source ../.rmfpms_config
#
echo "The following files are now deleted: "
#
# delete all files at least <ctime> days old (in the example, at least 7
# days old)
find $IBM_PERFORMANCE_REPOSITORY/archive -name "*.tgz" -ctime +5 -print -exec rm
'{}' \;
#
echo "... done.”
```

To automate this process, use **cron**. Example 4-16 shows an extract of the file crontab (default /etc/crontab, but this depends on your distribution), where the /home/mailand/rmfpms/bin/delete_old_perfdata script is called on a weekly basis on Monday at 1:00 am. You must have root access to change the file /etc/crontab.

Example 4-16 Extract of crontab

```
SHELL=/bin/sh
PATH=/usr/bin:/usr/sbin:/sbin:/bin:/usr/lib/news/bin
MAILTO=root
# check scripts in cron.hourly, cron.daily, cron.weekly, and cron.monthly
#
# Execute delete_old_perfdata on weekly base each monday at 1:00
0 1 * * 1 root /home/mailand/rmfpms/bin/delete_old_perfdata
...
```

Retrieve rmfpms performance data

To retrieve rmfpms performance data, use the rmfpms_unarchive tool. The command syntax is:

```
rmfpms/bin/rmfpms_unarchive yyyymmdd
```

where *yyyy* is the year, *mm* is the month, and *dd* is the day.

For example, you can retrieve the performance data from 10/16/2004 with the command:

```
rmfpms/bin/rmfpms_unarchive 20041016
```

Now rmfpms can access the data and you can use RMF PM to analyze it. For details, refer to "Sampling historical data" on page 86.

Tracing rmfpms

Standard logging messages are written to the rmfpms/.rmfpms/logs directory. If this information is not sufficient, you can enable the rmfpms debug mode.

First change the DEBUG_LEVEL(x) option in gpmsrv00.ini to DEBUG_LEVEL(3).

To enable the trace, use the following commands:

- **`export ICLUI_TRACETO=stderr`**
- **`export GPMDAEMON=0`**
- **`rmfpms/bin/rmfpms restart`**

Error messages are now written directly to the console.

4.4 RMF Performance Monitor

RMF Performance Monitoring Java Technology Edition (RMF PM) enables you to monitor the performance of your z/OS host or Linux image (zSeries or Intel) from a workstation. You can manage z/OS sysplexes and Linux images from a single point of control by monitoring the resources of the corresponding system.

RMF PM is available for Microsoft Windows and Linux. For the Linux version, check the RMF Home page. This section describes the setup of both versions.

RMF PM uses the DDS on the z/OS sysplex, and rmfpms on Linux, to retrieve its performance data. For details on how to set up and start the DDS, refer to "RMF Distributed Data Server" on page 136; for details on how to set up and start the rmfpms, see "RMF Linux data gatherer" on page 138.

Also, ensure that the users of RMF PM have sufficient access authority, by checking "Controlling access to RMF data for the sysplex data services" on page 116.

4.4.1 Installing RMF PM under Microsoft Windows

To install RMF PM, be sure that you meet the required prerequisites.

Prerequisites

We recommend that you fulfill these prerequisites for the client:

- Microsoft Windows 2000, ME, or XP.
- Netscape Navigator 4.6 or higher, or Microsoft Internet Explorer 5.0 or higher.
- Level of access:
  For Microsoft Windows 2000 or later, in order to install and use RMF PM, you must have at least Standard User level access (Microsoft Windows XP: Computer Administrator account type) or higher (see the Microsoft Windows user management facilities). If you

are in the Restricted User level access group (Microsoft Windows XP: Limited account type), you will encounter problems during the installation and use of RMF PM.

Installation

We recommend that you check the RMF Home page for the latest RMF PM version:

```
http://www.ibm.com/servers/eserver/zseries/zos/rmf/
```

RMF PM is also available on the host data set SYS1.SERBPWSV(GPMWINV2). Receive the file in binary mode and save it as gpmwinv2.exe.

The file gpmwinv2.exe is a self-extracting installation program that installs all the components of RMF PM onto your workstation.

Start RMF PM under Microsoft Windows

To start RMF PM, select **Start** → **Programs** → **IBM RMF Performance Management** → **RMF PM**.

For details about using RMF PM, refer to “RMF Performance Monitoring” on page 74.

4.4.2 Installing RMF PM under Linux on your workstation

Download the RMF PM Linux version to your workstation from the RMF home page:

```
http://www.ibm.com/servers/eserver/zseries/zos/rmf/
```

Download the file and install it using the command:

```
tar xvfz gpmlinpm.tgz
```

This installs RMF PM into a new rmfpm subdirectory on your workstation. You do not need root authority to install and start RMF PM.

For example, user mailand downloads the file gpmlinpm.tgz and stores it in his home directory /home/mailand/gpmlinpm.tgz.

He unpacks it, and it creates an rmfpm subdirectory and unpacks the files to this directory. Now rmfpm is successfully installed in the directory /home/mailand/rmfpm.

Start RMF PM under Linux on your workstation

To start RMF PM under Linux on the workstation, use the script rmfpm in the rmfpm/ directory.

For example, to start RMF PM when it is installed in /home/mailand/rmfpm, use the command:

```
/home/mailand/rmfpm/rmfpm
```

For details about using RMF PM, refer to “RMF Performance Monitoring” on page 74.

Part 3

Using RMF to answer your performance questions

In this part we describe how you can use the RMF components to monitor performance in your installation. We start with a discussion of performance in general, and identify the multiple elements you should monitor in order to control your environment.

Transactional and batch workload are analyzed and discussed, and we also cover WLC considerations, IRD, Crypto, WebSphere Application Server and Linux workload.

5

Performance analysis

Performance analysis is one of the most important areas of research on all existing platforms. Measuring, comparing with standards, and tuning are actions of a world where proactivity and creativity reduce response time for transactions, increase the throughput of batch jobs, and consequently reduce processing data costs.

Before we get into discussing how RMF reports performance measurements, in this chapter we provide an overview of the mechanisms utilized to measure and manage performance. In particular, we discuss:

- An overview of performance analysis
- CPU and I/O performance metrics
- Workload Manager (WLM) highlights

5.1 Defining performance

Is your system running fine? What does fine means? Are you happy with your system performance? Is the machine capacity able to face the next peak? Why did the batch window close so late yesterday? There are a lot of questions which need to be answered. Because of that we start with some basics on the topic in this section.

What is a performance problem?

There are many views on what constitutes a performance problem. Most of them revolve around unacceptably high response times or resource usage, which we can collectively refer to as *pain*. The need for performance investigation and analysis is, for example, indicated by:

- Bad or erratic response time
  - Service level objectives being exceeded
  - Users complaining about slow response compared with previous days
  - Unexpected changes in response times or resource utilizations
- Other indicators showing stress
  - RMF Monitor III Workflow/Exceptions
  - System resource indicators (for example, paging rates, DASD response)
  - Expected throughput (transactions per second) on the system not being attained

Ultimately, you have to decide for yourself whether a given situation is a problem worth pursuing or not. This differs installation by installation and it is based on your own experience and the knowledge of your system. We simply assume for the following discussions that you are trying to relieve some sort of numerically quantifiable pain in your system.

Generally, a performance problem is the result of some workload not getting the resources it needs to complete in a timely manner. Or, the resource is obtained, but is not fast enough to provide the desired response time.

5.1.1 Service level agreements (SLA)

The human view of the performance of a system is often subjective, emotional, and difficult to manage. However, meeting the business needs of the users is the reason the system exists. To match business needs with subjective perceptions, the concept of Service Level Agreements (SLAs) was introduced. The SLA is a contract that objectively describes and enforces such measurables as:

- Average transaction response time for network, I/O, CPU, or total
- The distribution of these response times (for example, 90% TSO trivial at less than 0.2 of a second)
- System availability metrics

SLA is a fundamental step for performance analysis and capacity planning disciplines.

5.2 Performance analysis overview

Performance analysis is the technique used to enforce in your IT systems the performance goals defined in the SLA.

Performance analysis consists of two parts, illustrated in Figure 5-1:

- *Performance administration* is executed by the service level administrator with the objective to *define* the rank of goals by importance of the transactions and later to analyze the *reports* to verify the performance results. The service level administrator is responsible for defining the installation's performance goals based on business needs. This explicit definition of workloads and performance goals is called a *service definition*.
- *Performance management* consists of the following tasks:
  - *Management,* with the objective of allocating data processing resources (CPU, I/O, and storage) to transactions according to their SLA. The expression *allocating resources* implies determining the transaction *priority* in the queues accessing such resources. This priority determines for how long the transaction spends time in such queues.
  - *Monitoring* is the task of verifying if the objectives of management are reached and reacting accordingly.

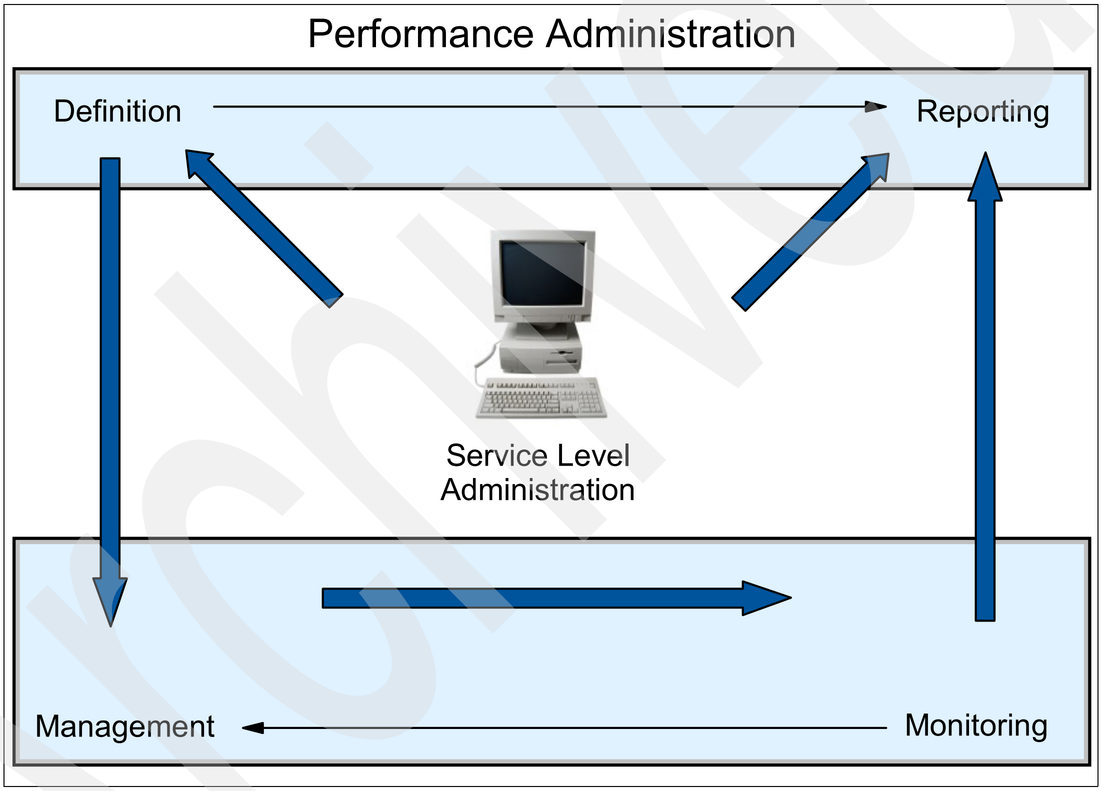


Figure 5-1 Performance analysis

5.2.1 Performance management

In this book we focus on the performance management aspects. Performance management includes an heuristic ongoing cycle of: measuring, planning, change, and waiting for results. It is very important to wait between the changing and the measuring. Time is needed in order for the system to assimilate the changes. *Heuristic* here means to reach a moving target state. In WLM goal mode, the performance management tasks are partially executed by the WLM code based on installation-defined goals.

Concerning measuring, WLM provides the capability to collect performance data (including delays) in the context of the service definition. The performance data is available to reporting and monitoring products so that they can use the same terminology.

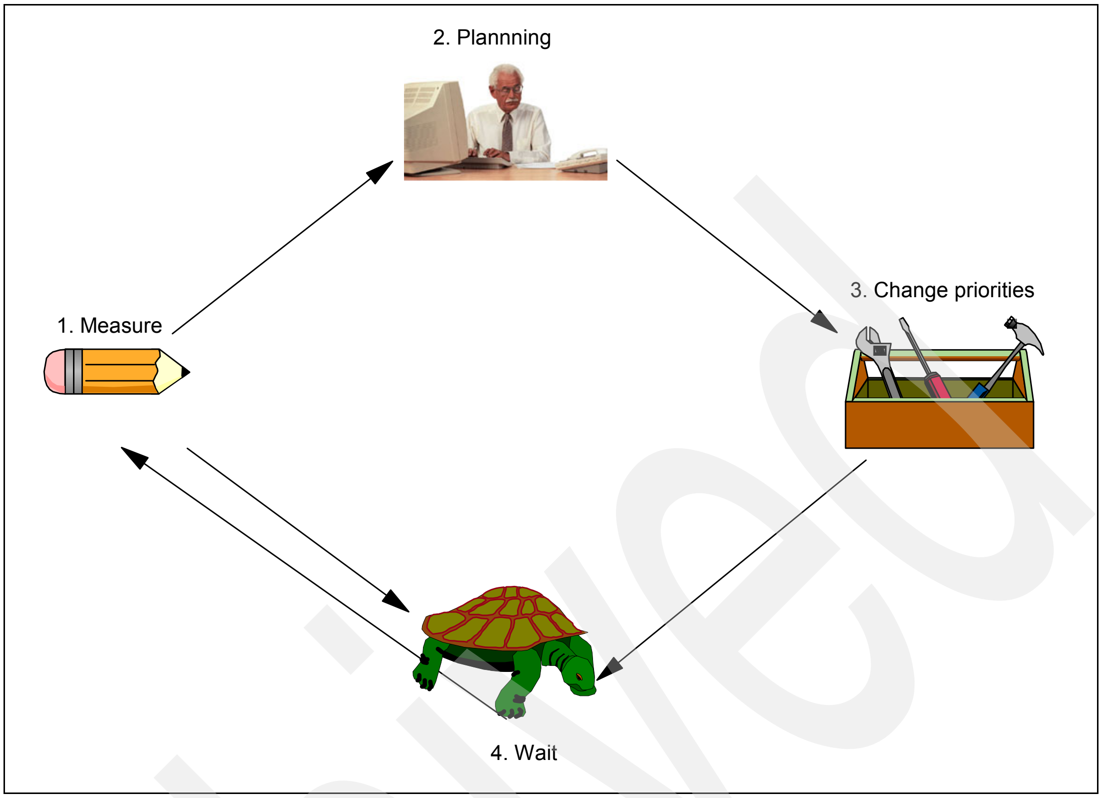


Figure 5-2 Performance management

As a general statement, we can say that there are three main ways to solve performance management problems, meaning the conflicts between reality and goals stated in the SLA:

- *Buy:* You can simply buy more resources (done by the installation).
- *Tune:* Tuning your system makes more effective and efficient use of existing resources. (This may be done by the installation or by a WLM resource adjustment routine).
- *Steal:* You can "steal" the resources from a less critical transaction by modifying priorities in the queues. (This may be done by the installation or by a WLM policy adjustment routine).

If none of these options are technically or financially possible, it is necessary to change the target expectations. You may have experienced the situation where you complete an extensive performance analysis only to conclude that no further tuning or stealing of resources can be done. One of the goals of this book is to assist you in determining whether you have reached this point.

5.2.2 General performance management metrics

In performance management there are some metrics used to verify the system behavior with respect to SLA and performance in general. *System* here means the following resources: CPUs (including ICFs, IFLs, zAAPs, SAPs) and channels. The following metrics apply to all such resources:

- Average transaction response time (Tr)
- External throughput rate (ETR)
- Resource Utilization (U)

In your performance management activity, you might also want to consider the concept of *saturation design point* (SDP). For details, see “Saturation design point (SDP)” on page 159.

Average transaction response time

A *transaction* is a business unit of work implemented to solve a problem for the business. It is produced by the interaction of a user (online or not) and the system. So, the transaction time has two pieces: computer response time and the user thinking time.

The computer response time is made up of:

- Host response time
- Network time
- Other servers time

Response time for an online transaction is the time between when the user hits the Enter key and when a screen with the required full data is presented. The time in other servers (your workstation, for example) is due to Web applications, for example where a servlet or JSP returns an HTML page including many GIFs. These issues are discussed in detail in 9.6, “Monitoring WebSphere Application Server workload” on page 301.

Response time for a batch job transaction is the time between the JOB submit and the final execution of the last job step; the time for sysout printing and purging are not included.

Response time in general is made of: service time (the time actual work is done) and waiting time (the time waiting for resource):

$$\text{Tr} = \text{Ts} + \text{Tw}$$

Similarly for computer response time, transaction service time (Ts) and transaction wait time (Tw) are made up of the individual resource service and wait times, shown in Figure 5-3 and defined by the following equations:

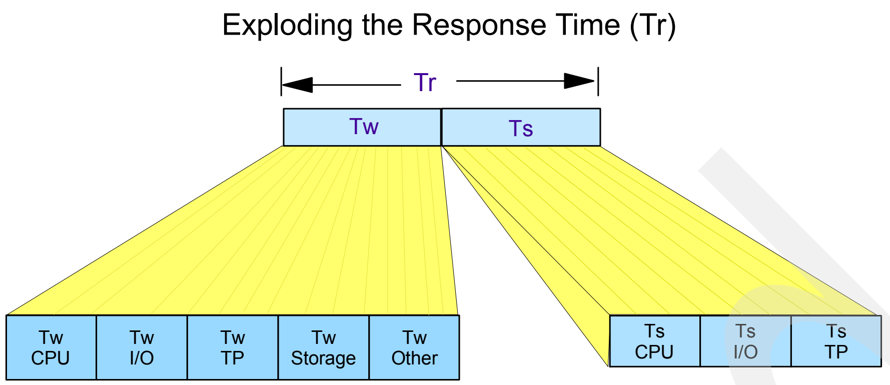
$$\text{Ts} = \text{Ts (CPU)} + \text{Ts (I/O)} + \text{Ts (TP)} + \text{Ts (Other)}$$

$$\text{Tw} = \text{Tw (CPU)} + \text{Tw (I/O)} + \text{Tw (TP)} + \text{Tw (Storage)} + \text{Tw (Other)}$$

You should understand that a high Tw is not a cause for unacceptable Tr, but a consequence of high activity in the server.

Another way of splitting response time (Tr) in the host is by subsystems. For example, in a CICS® transaction, the response time is composed of a number of factors, including the following:

- TCP/IP time
- CICS processing time
- DB2 access time
- Web server processing time
- UNIX System Services time
- JES2/JES3 time



Tw (CPU): Time with ready dispatchable units, but not enough dispatching priority.
Tw (I/O): Time with I/Os delayed in UCB or channel subsystem.
Tw (TP): Time in the VTAM or TCP/IP queues.
Tw (Storage): Time suffering a page fault or being swapped-out.
Tw (Other): Time delayed by operator, ENQ, data set recall, server AS service.
Ts (CPU): Time with dispatchable units executing CPU.
Ts (I/O): Time with I/Os being executed (connected or disconnected).
Ts (TP): Time being transported in the network.

Figure 5-3 Average transaction response time

Transaction response time is the best way to characterize the performance of a transaction, because it is directly connected to business needs and it is easily understood by people. The response time measured by z/OS and reported by RMF is the host response time portion only.

If the Tw is zero and the SLA is not accomplished, then you need to decrease the Ts. This can be done either by buying more resources on the processors (CPU, channel, SAP) or improving the logic of the application program processing the transaction.

There is a simple relationship between time in the queue (Tw) and the average length of the queue, and it is obtained by sampling. For more details, refer to “Little’s law” on page 324.

On top of presenting the response time in a multitude of reports, RMF offers other ways of viewing response times:

- Monitor III (and WLM) samples the state of address space and enclaves, keeping the result in state counters:
  - Using state is the sampled view of service time (Ts)
  - Delay state is the sampled view of wait time (Tw)
- Workflow and Execution Velocity are metrics based in the Using and Delay sampled numbers. These measurements provide a detailed view of what happens in the system, and they can help you understand system performance. You can see which address spaces, enclaves or workloads are performing well and which ones are not. Use these measurements to learn the reasons for performance problems occurring in workloads or resources. They are described in Chapter 1.4.1, “Common Monitor III report measurements” on page 12.
- Example 5-1 shows a sample of the Monitor III Group Response Time report where you can see several time components of a transaction response time. For example, the OMVS

service class transactions have, on average, a response time of 19.39 secs. This response time is broken into the details, such as 6.46 secs using CPU, 0.64 secs using DEV (I/O), 0.64 secs being delayed by CPU, and so forth.

Example 5-1 Monitor III Group Response Time

```
                        RMF V1R5   Group Response Time

 Samples: 100     System: AQTS  Date: 12/22/04  Time: 16.18.20  Range: 100   Sec

 Class: OMVS         Period: 1    Description:
 Primary Response Time Component: Delayed for unmonitored reasons

                                                TRANS    --- Response Time ----
 WFL    Users    Frames   Vector   EXCP   PGIN   Ended    -- Ended TRANS-(Sec) -
  %    TOT  ACT   %ACT     UTIL    Rate   Rate   Rate       WAIT  EXECUT  ACTUAL
  91    16    1      1        0   772.4    0.0   0.110     0.003   19.39   19.39

                          -AVG USG-   -------------Average Delay--------------
                   Total   PROC  DEV   PROC   DEV  STOR  SUBS  OPER   ENQ OTHER
 Average Users     2.130   0.71 0.07   0.07  0.00  0.00  0.00  0.00  0.01  1.31
 Response Time ACT 19.39   6.46 0.64   0.64  0.00  0.00  0.00  0.00  0.09  11.9

                     ---STOR Delay---  ---OUTR Swap Reason---  ---SUBS Delay---
                     Page  Swap  OUTR    TI    TO    LW    XS   JES   HSM   XCF
 Average Users       0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
 Response Time ACT   0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
```

External throughput rate (ETR)

ETR is a measurement of the number of ended transactions per elapsed time, as pictured in the following formula:

$$\text{ETR} = \frac{\text{Number of Transactions}}{\text{Elapsed time}}$$

ETR is also called *transaction rate.* ETR is associated with a system's throughput.

In the Workload Activity report, the elapsed time is the RMF interval. In Example 5-2, the interval set was 30 minutes (1800 seconds), during which 1879 transactions from service class TSOHIGH ended, causing an ETR of 1.04 (field END/S in the report).

Example 5-2 Workload Activity report (extract)

```
TRANSACTIONS (Service Class TSOHIGH)
   AVG       4.42
   MPL       4.39
   ENDED     1879
   END/S     1.04
   #SWAPS    3154
   EXCTD        0
   AVG
   ENC       0.32
   REM
   ENC       0.12
   ENC       0.01
```

There is also one formula relating the Tr with ETR (see “Little’s law” on page 324 for more details):

$$\mathrm{ETR} = \mathrm{N}/(\mathrm{Tt} + \mathrm{Tr})$$

Where:

N Is the average number of users generating transactions (logged on)
Tt Is the average thinking time of these users
Tr Is the average computer response time

Some considerations regarding this formula:

- The variables that more intensively affect the ETR are N and Tt due to their usual numeric values. Therefore, SLAs based on ETR are difficult to attain because the only variable that the IT department can directly manage is Tr.
- Any bottleneck, internal or external to the system, effects the ETR. Examples include: I/O constraints, tape mount delay, and paging delay.

Resource utilization

Resource utilization measures how much a resource delivers service in a timely basis. The general formula is:

$$\mathrm{U\%} = \mathrm{Busy\_Time} \times 100/\mathrm{Elapsed\_Time}$$

We may derive (for just one server):

$$\mathrm{Busy\_Time} = \mathrm{Ts} \times \#\ \mathrm{Transactions}$$

Replacing Busy_Time in the first formula:

$$\mathrm{U\%} = \mathrm{Ts} \times \#\ \mathrm{Transactions} \times 100/\mathrm{Elapsed\_time}$$

$$\mathrm{U\%} = \mathrm{Ts} \times \mathrm{ETR} \times 100$$

This formula shows that the U increases when Ts or ETR or both elements are raised.

For example, if in a bank branch office with only one teller there is an average arrival rate of 2 clients/minute with a service time (Ts) of 0.25 minute, we can say that the teller average utilization is U% = 0.25 * 2 * 100 = 50%

- The product: Ts * ETR is also called *Traffic*.
- The product: Tr * ETR is called *Intensity*.

For example, you can see the formula applied to the sample in Example 5-3 with the RMF CPU Activity report showing a z/OS image with four logical CPUs. Each CPU utilization is pictured in the column LPAR BUSY TIME PERC. In this case the average CPU utilization is 64.22%.

Example 5-3 CPU Activity report

```
                                   C P U   A C T I V I T Y   R E P O R T

z/OS V1R5                SYSTEM ID SYS1                      DATE 02/09/2004      INTERVAL 13.00.004
                          RPT VERSION V1R5 RMF               TIME 16.30.00       CYCLE 1.000 SECONDS

 CPU     2064

 CPU     ONLINE TIME   LPAR BUSY    MVS BUSY    CPU SERIAL    I/O TOTAL         %I/O INTERRUPTS
 NUMBER  PERCENTAGE    TIME PERC    TIME PERC   NUMBER        INTERRUPT RATE    HANDLED VIA TPI
 3       100.00        64.24        100.0       3A1536        413.0              0.64
 4       100.00        64.21        100.0       3A1536        410.9              0.66
```

```
5        100.00        64.19        100.0        3A1536        410.6                0.60
6        100.00        64.22        100.0        3A1536        415.6                0.66
TOTAL/AVERAGE          64.22        100.0                       1650                0.64
```

You can also use the Markov's equation, relating Tw and U. Refer to “Markov’s equation” on page 325 for more details.

Putting this on a graphic where the service time is constant, you can observe that when the U% is greater than the knee point on your curve, the wait time increases drastically.

In a z/OS system, the consequence is that if the key business transactions (highest dispatching priorities) and the z/OS tasks use more than the knee point on the curve, then the Tw for these transactions starts to increase, causing performance degradation (high Tr) in those critical workloads. If not, the rest of the capacity can be consumed by other workloads up to 100%, without impacting the business. Refer to “*PROC Workflow(%)” on page 14 for further details.

Saturation design point (SDP)

CPU is either in busy state or in wait state. In other words, either it is 100% busy or 0% busy. Figure 5-4 shows this fact for a duration of a few minutes. The shaded area indicates that the CPU is busy. Performance monitors group these states in specific amounts of time, for example, one minute intervals. In Figure 5-4, the left portion of the graphic shows a minute time scale. In the first minute, the average utilization is 40% and in the second minute it is 65%.

Usually, these values are grouped in intervals of hours. An hourly average is calculated and presented to human analysis (50%, 80%, and 90% in the right part of the graphic). As you decrease the granularity (from one minute to one hour) you may lose key details. The major question becomes: “If my hourly utilization is 75%, how often did I reach 100% in the *per minute* average within this hour?”

To answer this question, the concept of *saturation design point* (SDP) was developed. SDP is the maximum average utilization in a larger interval (one hour, for example), where the installation is sure that in the smaller interval (one minute, for example) the average utilization was never 100%. If the current average is greater than the SDP, then the 100% busy was reached in the smaller interval. The formula for the hour/minute relationship is:

$$\text{SDP\%} = \text{Average Hourly Utilization} \times 100 / \text{Peak Minutely Utilization}$$

Imagine that, in some hour interval, the Average Hourly Utilization was 80%, and, in that hour interval, the highest minute average peak was 92%. Then:

SDP% = 80 *100 / 92 = 87%

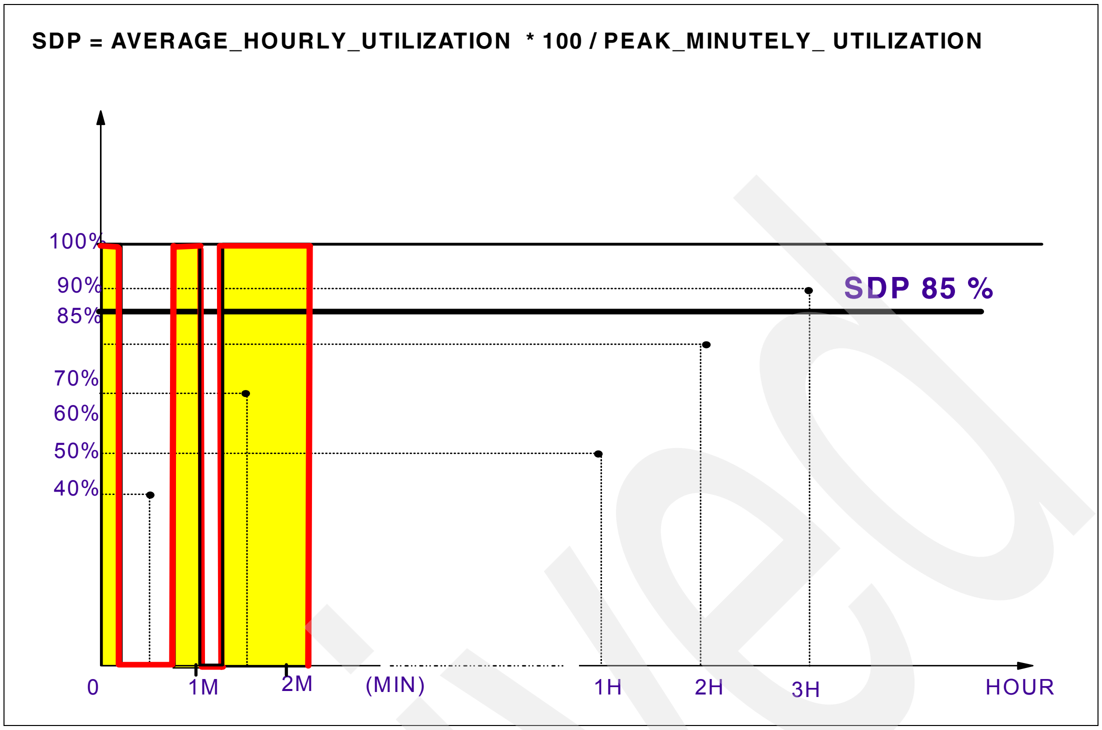


Figure 5-4 Utilization curve with SDP

Observing the formula, it is clear that when SDP is equal to the Average Hourly Utilization, then at least in one minute, the 100% minute average was reached.

In the graphic, with the SDP worth 85%, then when hour Utilization is 90% (as in the third hour), there is the guarantee that some minute average within that hour reached the 100% mark.

If a customer knows its SDP using theoretical (usually 70%) or experimental approaches, they can use this information in:

- Performance management to avoid the hour average exceeding the SDP
- Capacity planning for sizing the processors to not exceed the SDP limit

5.2.3 Processor performance metrics

Some metrics used in performance management and capacity planning to evaluate the processor speed capacity are the following:

- CPU Time
- MIPS
- MFLOPS
- CP service units - SRM Constant – MSU
- Internal Throughput Rate (ITR)
- Large Systems Performance Reference (LSPR) - RPP

CPU time

CPU time of a transaction is the processing time consumed to execute the program logic of the transaction. In z/OS, the program is associated with a dispatchable unit, a task (TCB) or a service request (SRB) in an address space or enclave basis. The transaction CPU time is measured by z/OS and is called captured time.

The RMF Workload Activity report shown in Example 5-4 shows several components of the CPU time. These time components are covered in depth in “z/OS CPU time considerations” on page 168.

Example 5-4 RMF Workload report extract

```
---SERVICE----   --SERVICE RATES--
 IOC     56040    ABSRPTN       598
 CPU      1680K   TRX SERV      594
 MSO      2938K   TCB         271.3
 SRB     56695    SRB           9.2
 TOT      4731K   RCT           4.4
 /SEC     2626    IIT           5.4
                  HST           0.0
                  APPL %       16.1
```

One of the installation’s targets in performance management is to reduce such amounts. The CPU time value can also be theoretically derived through the formula:

$$\text{CPU time} = \text{Cycle\_Time} \times \text{CPAI} \times \text{Path\_Length}$$

Let’s discuss the elements of this formula.

Cycle time

Cycle time can be defined in one of the following ways:

- Time to execute the fastest instruction of the CPU instruction set.
- Time elapsed between two consecutive pulses generated by the internal clock in the processor. Numerically it is the inverse of the frequency that is measured in MHz.

$$\text{CPU time} = \text{Path length} \times \frac{Cycles}{Instructions} \times \text{Cycle time}$$

  – Path length depends on the instruction set and the compiler.

  – Cycles/instruction depends on the design (for example, microcoded instructions).

  – Cycle time depends on the technology (for example, CPU model).

Thus, cycle time by itself is not a complete indicator of CPU speed.

CPAI - Cycle per average instruction

CPAI is the number of cycles, on average, needed to execute one instruction. CPAI depends very much in the set of executed instructions. There are very simple instructions executed in just one cycle. The z990 can even execute more than one instruction in one cycle - this property is called *superscalar*. On the other hand, there are complex instructions that need many cycles in order to execute, thereby biasing the CPAI to a higher value.

The multiplication of: CPAI x Cycle_Time produces the average time to execute one instruction.

Path length

Path length is the number of instructions needed to execute the transaction program.

Obviously, if you have less path length, this produces a lower CPU time. Path length depends on the following two factors:

- Processor architecture, which defines the set of instructions
- Quality of the compiler that generates the object code

The goal of an architecture is to offer powerful instructions in order to implement a program with a small path length.

Millions of instructions per second (MIPS)

This term depends on cycle time and number of cycles per instructions. It is only valid for comparison if the instruction sets are the same. In common usage today, MIPS are used as a single number to reflect relative processor capacity. As with any single-number capacity indicator, your experience may vary considerably. These numbers are often based on vendor-provided announcements.

By the way, MIPS means *millions of instructions per second*. There is no MIP, and the term MIP does not exist. This is often used erroneously. A small processor has a speed of 1 MIPS, not 1 MIP.

Using MIPS to compare processors

Using MIPS as a comparison between two processors is fair when the two processors execute exactly the same set of instructions executing the same logic associated with the same transaction program.

Today, MIPS is used as a single number to reflect relative processor capacity. CPU time and MIPS have the following differences when used to compare processors:

- CPU time is *measured* and MIPS is *imagined* (often based on vendor claims).
- CPU time varies when the transaction switches between processors of different speeds; MIPS *consumed* by one transaction should not vary with the speed of processor.

MFLOPS

MFLOPS mean *millions of floating point instructions per second.* This value is used only in numerically intensive computing (scientific/technical). There is one important benchmark available in the industry that measures MFLOPS performance, the LINPACK benchmark. Here, many computer companies are in a tough battle to have the world-wide fastest processor. You can find details at:

```
www.linpack.com
```

CPU Service Units - SRM Constant – MSU

CPU Service Units is a metric used by z/OS to measure the CPU consumption by transactions executing under z/OS dispatchable units such as tasks (TCB) or service requests (SRB).

CPU Service Unit should be more or less *repetitive* and only includes the *productive* CPU execution. These terms have the following meanings:

- Productive - The consumption of CPU due to overheads such as page fault and swapping is not accounted for in the transaction CPU Service Unit.
- Repetitive - The same value is roughly measured for the same transaction (executing the same logic with the same amount of I/O records) in any CPU.

The System Resource Manager (SRM) constant is a number derived by product engineering to normalize the speed of a CPU, so that a given amount of work would report the same service unit consumption on different processors.

The relationship between CPU SUs and SRM constant is illustrated in the following formula:

$$\text{Service units} = \text{CPU seconds} \times \text{SRM constant}$$

Service Units (SU) are typically used by the System Resource Manager as a base for resource management.

Note: The SUs delivered in an LPAR configuration are dependent on the number of logical CP defined in that LPAR and not on the whole machine capacity.

The SRM constants are contained in internal tables within z/OS and are published in the *z/OS MVS Initialization and Tuning Reference*, SA22-7592. The SUs are reported in many Monitor I and Monitor III reports, or you can find them at:

http://www.ibm.com/servers/eservers/zseries/srm/

Many installations that have to charge the users will do so on the basis of SUs. This is a reasonable approximation of relative capacity. But, of course, any single-number metric for processor capacity comparison can be misleading; therefore, LSPR numbers, based on your own workload mix, are the best basis to use.

The power of a system can also be characterized in terms of MSUs (*millions of service units per hour*). These numbers are published in the LSPR documentation.

Internal throughput rate (ITR)

ITR determines the capacity of a processor in terms of the number of transactions per CPU second.

$$\text{ITR} = \text{Number of Transactions} / \text{CPU Time}$$

ITR is a function of:

- CPU speed
  The faster the CPU, the higher the ITR.
- Operating system
  The better the operating system, the higher the ITR.
- Transaction workload characteristics
  Short trivial transactions produce higher ITR.

If we make constant the operating system (z/OS, for example) and the transaction workload variable, we can use ITR to measure the CPU speed in terms of transactions. ITR is more suitable for such comparison than ETR because ETR depends on a much larger number of variables. Refer to “External throughput rate (ETR)” on page 157 for details about ETR.

IBM calculates ITRs for several workloads and machines using the Large Systems Performance Reference (LSPR) methodology. ITR provides a reliable basis for measuring capacity and performance. The relationship between ETR and ITR is:

$$\text{ETR} = \text{ITR} \times \text{CPU utilization}$$

Then, when the CPU utilization is 100%, the ETR is equal to ITR.

Using ITRs, you can define an ITR ratio (ITRR) between two processors:

$$\text{ITRR} = \frac{\text{ITR(Proc2)}}{\text{ITR(Proc1)}}$$

Large Systems Performance Reference (LSPR)

Although CPU time consumed by one transaction is the best metric to measure the speed of a CPU, the formula seen in “CPAI - Cycle per average instruction” on page 161 is not practical because path length and CPAI hardly can be measured in your workload.

To address that problem, IBM implemented a benchmark method to estimate the CPU capacity needed for your workload. A *benchmark* is a well-defined set of measurements for specific applications. A benchmark can be online transaction processing, batch jobs, WebSphere applications, and others. Due to the fact that benchmarks are based on applications (rather than just using a MIPS value for a processor, for example), it is easier for installations to understand benchmark results, and then project these results onto their own environments.

The IBM benchmark project is called *Large Systems Performance Reference (LSPR)*. It measures ITRs of different machines with different workload types and different operating systems (z/OS, z/VM, VSE, ACP/TPF). The idea behind LSPR is that in most cases it is not useful to characterize a processor by only one number (or a set of numbers for different workloads), but instead by comparing two processors. This means that if you have installed one specific processor and you want to upgrade to another one, you can use the LSPR numbers to see how much more capacity you can get. LSPR tables always show ITRR values, one selected processor is defined as base with the value ITRR=1.0, and the values for all other processors show the relative performance compared to the base processor. All the capacity numbers are relative to the zSeries 2084-301. Table 5-1 shows the ITRR values for some z990 processors.

Table 5-1 LSPR table z990

Processor	#CP	Mixed	CB-L	CB-S	WASDB	OLTP-W	OLTP-T
2084-301	1	1.00	1.00	1.00	1.00	1.00	1.00
2084-302	2	1.90	1.95	1.80	1.90	1.93	1.92
2084-303	3	2.76	2.88	2.55	2.79	2.82	2.80
2084-304	4	3.60	3.79	3.28	3.66	3.68	3.64
...	...	...	...	...	...	...	...
2084-315	15	10.78	12.55	8.75	12.11	11.11	10.29
2084-316	16	11.24	13.23	9.02	12.78	11.60	10.65
Ratios for 17 way and above are based on running multiple OS images.							
2084-317	17	11.80	13.96	9.37	13.52	12.23	11.18
2084-318	18	12.36	14.69	9.71	14.26	12.87	11.70
...	...	...	...	...	...	...	...
2084-331	31	19.58	24.18	14.17	23.87	21.10	18.48
2084-332	32	20.13	24.91	14.51	24.61	21.73	19.00

These ratios represent IBM's assessment of relative processor capacity in an unconstrained environment for the specific benchmark workloads and system control programs specified in the tables. Ratios are based on measurements and analysis. The data shown in the table is based solely on IBM's measurements and analysis of the processors in the tables.

Each individual LSPR workload is designed to focus on a major type of activity, such as interactive, on-line database, or batch. The LSPR does not focus on individual pieces of work, such as a specific job or application. Instead, each LSPR workload includes a broad mix of activity related to that workload type. Focusing on a broad mix can help assure that resulting capacity comparisons are not skewed. Some of the workloads are:

- OLTP-T - Traditional On-line Workload

- OLTP-W - Web-enabled On-line Workload
- WASDB - WebSphere Application Server and Database
- CB-L - Commercial Batch Long Job Steps
- CB-S - Commercial Batch Short Job Steps

Relative Processor Power (RPP) is the average ITR ratio of the workloads described by the Mixed column in Table 5-1. The Mixed workload consists of an equal mix of OLTP-T, OLTP-W, WASDB, CB-S and CB-L.

RPP is normalized to a base machine. So, if a given transaction takes a certain amount of RPPs, you can estimate how much it would consume on a different machine by using the formula:

$$\text{Utilization (CPU new)} = \frac{\text{RPP(CPU old)}}{\text{RPP(CPU new)}} \times \text{Utlization (CPU old)}$$

In summary, various numbers may commonly be used as rough indicators of CPU capacity. Remember that most of these are very rough approximations only, and your actual capacity may vary significantly. To get the most accurate assessment of CPU capacity for your workload, differences in workload processing characteristics must be taken into account using a methodology such as that described in the Large Systems Performance Reference. You can find the most current version at:

http://www.ibm.com/servers/eserver/zseries/lspr/zSeries.html

In addition, there is a recommendation to not use the LSPR ratios directly, but rather to build workload mixes based on the work your installation is running. There is a flash with the methodology of how to define a workload mix when building capacity sizing plans that gives information on when it is appropriate for use. You can find the document at:

http://www.ibm.com/servers/eserver/zseries/lspr/lsprmixwork.html

Additional information can be found in the following WSC document:

http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS135

5.2.4 I/O performance and metrics

As we did with CPU, now we visit some performance management considerations and metrics associated specifically with an I/O operation.

As CPU speed increases, the I/O response time (I/O Tr) becomes the determinant factor in the average transaction response time. The I/O response time is shown in the formula:

$$\text{I/O Tr} = \text{I/O Ts} + \text{I/O Tw}$$

The average I/O response time is one of the fields shown in Example 5-5, the Shared Device Activity report.

Example 5-5 Shared Device Activity report

```
SMF            DEVICE    AVG  AVG    AVG  AVG  AVG  AVG  AVG    %      %     %     AVG
 DEV  DEVICE   VOLUME PAV  SYS  IODF LCU  ACTIVITY  RESP IOSQ   CMR  DB   PEND DISC CONN   DEV    DEV   DEV   NUMBER
 NUM  TYPE     SERIAL      ID   SUFF      RATE      TIME TIME   DLY  DLY  TIME TIME TIME   CONN   UTIL  RESV  ALLOC

 8004 33903    PAVTS1      *ALL           4043.964   1.0  0.0   0.2  0.0   0.4  0.0  0.7  33.03  33.03   0.0   8.9
                        8  SYS1  29  00A9 1139.045   1.0  0.0   0.2  0.0   0.4  0.0  0.7   9.33   9.33   0.0   3.0
                        8  SYS2  29  00A9 1466.329   1.0  0.0   0.2  0.0   0.4  0.0  0.7  11.96  11.96   0.0   3.0
                        8  SYS3  29  00A9 1438.591   1.0  0.0   0.2  0.0   0.4  0.0  0.7  11.74  11.74   0.0   2.9
```

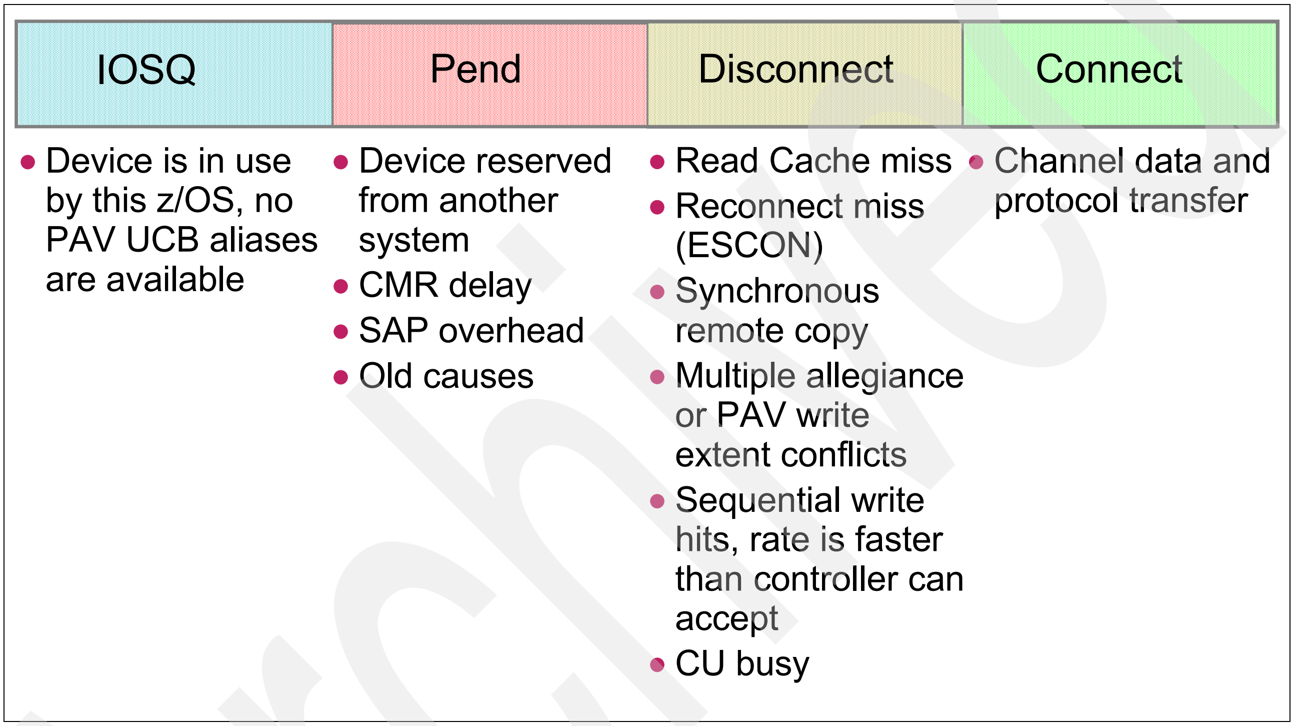
Obviously, you can get excellent transaction response time by reducing I/O wait time (Tw) and I/O service time (Ts). Turning to I/O performance metrics, the I/O Tr is formed by four components:

I/O Tr = IOSQ Time + Pending Time + Connect Time + Disconnect Time

where:

- I/O Tw = IOSQ Time + Pending Time
- I/O Ts = Connect Time + Disconnect Time

Figure 5-5 depicts the four stages of an I/O operation followed by an explanation of each of the fields.



IOSQ

- Device is in use by this z/OS, no PAV UCB aliases are available

Pend

- Device reserved from another system
- CMR delay
- SAP overhead
- Old causes

Disconnect

- Read Cache miss
- Reconnect miss (ESCON)
- Synchronous remote copy
- Multiple allegiance or PAV write extent conflicts
- Sequential write hits, rate is faster than controller can accept
- CU busy

Connect

- Channel data and protocol transfer

Figure 5-5 The four stages of an I/O operation

- IOSQ: Time waiting for the device availability in the z/OS operating system. It is drastically reduced by the implementation of dynamic PAV.
- Pending: Time from the SSCH instruction (issued by z/OS) till the starting of the dialog between the channel and the I/O controller. Previously, the pending time was composed of the following components: all channel path busy time, DPS busy time, CU busy time, DB time and SAP overhead time. DPS, CU, and DB time were presented individually in the RMF DASD Activity report and in the IOQUEUE Activity report. The other two values were derived by subtraction from the total pending time. Due to FICON and new controllers, some of these time components disappeared and some are accounted for as disconnect time. This is because FICON channel is almost always available, FICON switches do not present DPS busy, newer controllers hide the CU busy situation, and Shark multiple allegiance avoids shared device busy. Then, with this technology, the pending time is formed by:
  - Command Reply (CMR) time: This measures how long the channel waits for the controller at the start of the I/O dialog. It is shown in RMF Device activity and IOQUEUE reports. It is an indication of how busy the controller is.

  - Device busy time: This indicates how long the device is busy because a hardware reserve has been issued by another system. The other cases of potential device busy, together with CU busy and DPS busy, were moved to disconnect time. It is shown in RMF Device Activity report and it can be an explanation for Missing Interrupt Handling messages, together with bad performance.
  - SAP overhead:. This indicates the time SAP needs to handle the I/O request. It is not shown in any RMF report. The value should be around 0.1 to 0.3 milliseconds depending on the SAP speed, which depends on the CPC model.
  - Old reasons: This category includes issues such as the existence of ESCON® directors causing DPS busy. These times are not individually portrayed in the DASD activity report but you still can see them in the IOQUEUE Activity report.
- Disconnect: Time that the I/O operation already started, but the channel and I/O controller are not in a dialog. In a FICON protocol there is no disconnect time concept; however, a numerically equivalent figure is accounted for by the hardware component Channel Measurement Facility. The reasons for disconnect time are:
  - Read Cache miss; the I/O block is not in the controller cache.
  - Reconnect miss; the time to dynamically reconnect an ESCON channel.
  - Synchronous remote copy (PPRC); the mirroring is done within the disconnect time.
  - Multiple allegiance or PAV write extent conflicts; these were previously reported as device busy time (in pending time). Modern controllers move it into disconnect time.
  - Sequential write hits faster than controller can accept in the cache.
  - CU busy; this was previously reported in pending time. Modern controllers move it into disconnect time.
- Connect: Time when the channel is transferring data from or to the controller cache or exchanging control information with the controller about one I/O operation. That is, both are in a dialog. Then, connect time includes the real productive time of an I/O operation, meaning the data transfer. However, there is still the overhead of the dialog protocol. For example, in sequential processing, if you decrease the number of SSCHs (by better buffering) even though transferring the same amount of data, your connect time decreases consistently because of less overhead.

There are other I/O metrics, such as:

- I/O Intensity = I/O Tr * I/O Rate

  Measures in ms/sec the activity in the device, including the queue time.
- I/O Traffic = I/O Ts * I/O Rate

  Measures in ms/sec the activity in the device, excluding the queue time. In one device, it is numerically equal to the utilization of such device.
- I/O Density = I/O Rate per DASD space capacity

  Measured in I/Os per sec per Gbyte. This metric depends on type of access, size of your files, and how many gigabytes per device (if not a PAV device).

Techniques to reduce the I/O response time

Generally speaking, you can reduce the average I/O response time (I/O Tr) using software or hardware techniques. The software techniques aim to decrease the number of I/O operations, or bytes transferred, by implementing:

- Buffering, which decreases the number of bytes transferred in a random access, and make the I/Os more efficient for sequential access. Buffering can be implemented through:
  - Virtual address spaces
  - Data spaces
  - Hiperspace™, even though not recommended in z/Architecture™
- Data compression, which decreases the number of bytes transferred in any type of access.
- Data striping, which make the I/Os more efficient for sequential accesses due to parallelism.

Some examples of reducing the I/O response time using hardware techniques are the following:

- Faster channels (such as FICON and FICON Express)
- Faster device paths (adapters) to the controllers
- Larger and more efficient cache controller
- More DASD subsystem concurrency; for example, parallel access volumes (PAV) in Enterprise Storage Server (ESS)
- Faster disks (RPMs), small size, RAID-10
- Implement channel subsystem I/O priority from IRD

Refer to *VSAM Demystified,* SG24-6105, to get more details on techniques to decrease those times.

5.3 z/OS CPU time considerations

An important aspect of performance management is capturing performance data for report generation. z/OS collects CPU time consumed by dispatchable units such as tasks (TCBs) and service requests (SRBs) associated to executed transactions.

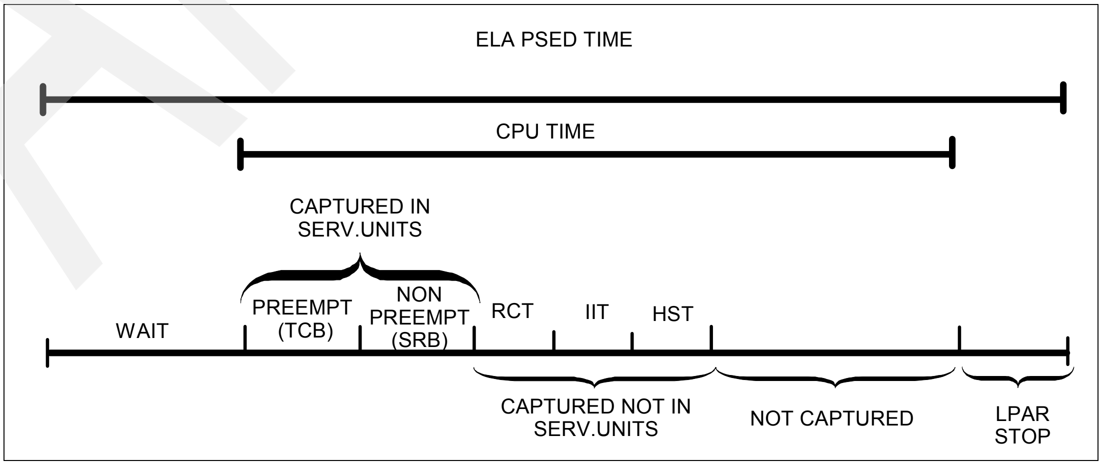


Figure 5-6 z/OS CPU time components

When a TCB or an SRB is dispatched, an **x** value is loaded in the CPU timer (a hardware clock that decreases its contents). When the TCB or SRB is interrupted, the current CPU timer (**x-y**) is stored in memory.

y is then accumulated in the ASCB control block (or in an enclave - ENCB), therefore, in an AS/enclave basis. There are two counters in ASCB, one for each of the following:

- Preemptable dispatchable units, such as TCB, SRB client, SRB enclaves
- Non-preemptable dispatchable units, such as SRB traditional

Refer to “z/OS preemptability” on page 170 to get more information on preemptable and non-preemptable dispatchable units.

The accumulated time (in an AS or enclave basis) is called *captured time* and is converted into CPU service units. Refer to “CPU Service Units - SRM Constant – MSU” on page 162 for details. It typically accounts for 85% to 90% of the total CPU time for systems with a high utilization. The value can be significantly smaller in under-utilized systems, depending on whether or not the Alternate Wait Management is activated.

z/OS (or the LPAR) also accounts for the CP wait time (logical and physical) when the CPU is placed in wait state because there is no work to be done. Then, wait time is the time that the logical CP was in wait state due to the lack of activity in the LP.

There is also a captured time not converted in CPU SUs. It is the captured CPU time not included in ASCB preemptable or non-preemptable counters, and consequently not converted to CPU SUs. There are three components generating such times:

- IIT, the CPU time spent in I/O SLIH routine
- RCT, the CPU time spent in the address space region control task
- HST, the CPU time spent accessing hiperspace

As you can see in Example 5-4 on page 161, these fields are reported into the RMF Workload Activity report. Keep in mind that this RMF report is a global report and those numbers are captured at the sysplex level. The value `16.1%` in field `% APPL` is derived through the formula:

```
% APPL = TCB + SRB + RCT + IIT + HST / RMF Interval * 100
```

% APPL indicates the % of one CPU consumed; a number such as 150% would mean that the reported workload consumed 1.5 CPUs (processors).

There are also other activities executed in the CPU that are not accounted for by z/OS. This time is called non-captured time. It can be numerically derived by subtracting the captured time from the total busy time. Non-captured time includes:

- Overhead such as page faults and swaps.
- Global services, such as dispatcher, WLM sampling, and SRM, where there is not an AS or enclave in charge.
- Services where it is impossible to find the task that requested the service, such as I/O FLIH processing.
- LPAR processing associated with this logical partition.

5.3.1 Capture ratio

The capture ratio represents the percentage of CPU time that can be counted to application work. Capture ratio is the result derived from the formula:

$$\text{Capture Ratio} = \text{Captured\_time} / \text{Total\_CPU\_\_busy\_time}$$

Capture ratio is used in capacity planning and accounting to distribute the non-captured time proportionally to the capture time consumed by address spaces and enclaves. If you divide the address space captured time by the capture ratio, you have an approximation for the time indirectly consumed by such address spaces, including the non-captured time.

You should measure your capture ratio for your installation. Whenever a change of z/OS and its subsystems or new product goes in production, you should re-evaluate this number. Also re-evaluate your capture ratio when you introduce a zAAP processor.

You can use RMF Workload Activity and CPU Activity reports for the capture ratio calculation. To make your task easy, we derive a more adequate formula. When using one of these reports, you can use one of the following formulas with the corresponding fields directly from the RMF reports:

$$\text{LP Capture Ratio} = (\sum \text{SC\%APPL}) / \text{AVG\_Logical\_CPU\_Utilization} \times \text{\#\_of\_Logical\_CPUs}$$

Remember that if you are in sysplex, the Workload Activity report (containing the SUM SC%APPL) presents numbers from all the sysplex. Then, the denominator of the previous formula must be derived from all the images contained in your sysplex.

$$\text{Capture Ratio} = \frac{(\sum \text{Service Class APPL\%}) / \text{Logical CPs}}{\text{LPAR BUSY}}$$

5.3.2 Service definition coefficient (SDC)

In order for an installation to provide a specific weight to the CPU service units consumed, the service definition coefficient (SDC) was introduced in the WLM policy. The final calculation to derive the CPU service units is:

$$\text{CPU Service Units} = \text{Raw CPU Service Units} \times \text{SDC}$$

There is one SDC per TCB time (also called preemptive time) and one SDC for SRB time (also called non-preemptable time). The TCB service units are called CPU service units in RMF reports. The SRB service unit are called SRB service units. Currently, the recommendation is to assign a value of 1.0 to these fields.

In Example 5-4, you can see the TCB time and the CPU service units calculated from the TCB time. In this example, the TCB time consumed by the reported transactions is 271.3 seconds and the corresponding CPU service units is 1680K.

Many installations charge the user’s departments on the basis of SUs. This is a reasonable approximation of relative capacity. But of course, any single-number metric for processor capacity comparison can be misleading, and the Large System Performance Reference (LSPR) numbers based on your own workload topology should be the best model.

5.3.3 z/OS preemptability

Preemptable processes are the dispatchable units capable of being immediately suspended (placed in a ready state) if another dispatchable unit with higher dispatching priority becomes ready for executing in CPU.

In the majority of the cases, z/OS is very responsive to the fact that a high dispatching priority TCB/SRB became ready (just being delayed by CPU), preempting a low dispatching priority TCB/SRB.

Being a preemptable operating system has the advantage of honoring first the high priority transactions (the ones important to business). This is crucial, especially when the CPU utilization is very high. Being a preemptable operating system has the disadvantage of introducing overhead in the system because preempting a TCB/SRB implies saving the status of the preempted TCB/SRB, and restoring its status later when it becomes dispatchable again.

Putting all this together, there are a few cases where z/OS is not preemptable, specifically the following:

- **Traditional SRBs are always non-preemptable**. They were designed to execute a quick code and terminate. Then, it is better to let it go from the start to end without the overhead of saving/restoring its status. However, there might be some “dishonest code” that runs in traditional SRB mode, yet includes large pieces of programs enjoying this lack of preemptability, with the end result that it monopolizes the CPU. Nowadays there are some types of preemptable SRBs, such as enclave SRB and client SRB.
- **Reduced Preemption.** When a task gains the control of the CPU, z/OS limits the use of the processor by setting the CPU timer for a certain amount of time. At the end of this "time slice" z/OS determines if a higher priority task or SRB should be dispatched or if this unit of work may be redispatched for another time slice. There are limited conditions that may cause the unit of work to be preempted to perform higher priority work before the time slice completes but these conditions are not probable.
- **CPU is running disable for interrupts**. This happens rarely in z/OS and it is mainly due to integrity reasons.

5.4 How to measure performance

Now that we have given you a primer on performance management and its metrics, we next consider aspects of the different types of performance monitors.

In this book, we concentrate on software monitors, or to be more specific, on RMF. You know all the different gatherer and reporter functions RMF offers for monitoring your system. We discuss here in detail which functions should be used for what specific tasks.

5.5 Planning system capacity

Capacity planning is the process of planning for sufficient computer resources capacity in a cost-effective manner to meet the service needs for all users and involves asking the following questions:

- How much of your computer resources are being used?
  - CPU
  - Processor storage
  - I/O subsystem
  - Coupling Facilities
  - Channels
  - Network
- Which workloads are consuming the resources (workload distribution)?
- What are the expected growth rates?
- When will the demands on current resources impact service levels?

Benefits of capacity planning

An effective capacity planning process provides:

- A mapping of business objectives (user requirements) into quantifiable information technology (IT) resources.
- Management-oriented reporting of service, resource usage, and cost. In an objective way, this makes clear what is involved in providing users with good performance.
- Input for making business decisions which involve IT.
- A way to avoid surprises.

Performance management versus capacity planning

These two disciplines have much in common. *Performance management* concentrates on allocating existing resources to meet service objectives. *Capacity planning* is a means of predicting resources needed to meet future service objectives.

Similar approaches can be used for both; here are some examples:

- Rules of thumb
- Comparison with other systems
- Parametric methods
  - Transaction profile (10 read calls, 2 update calls, 8 physical I/Os)
  - Cost of function (CICS: 1.5 ms per physical I/O)
- Analytic (queuing theory) models.
- Simulation, using a computer program that has the essential logic of the system.
- Benchmarks, Teleprocessing Network Simulator (TPNS).

In addition, performance and capacity planning work tend to require similar input data from the system. In particular, both require resource consumption by workload data.

5.6 Using RMF

RMF issues reports about performance problems as they occur, so that your installation can take action before the problems become critical. Your installation can use RMF to:

- Determine that your system is running smoothly
- Detect system bottlenecks caused by contention for resources
- Evaluate the service your installation provides to different groups of users
- Identify the workload delayed and the reason for the delay
- Monitor system failures, system stalls, and failures of selected applications

What sampling cycle and reporting interval should you use?

For Monitor III: Use the default sampling cycle of 1 second, with a value of 60 seconds for the reporting interval (MINTIME); 100 seconds is the default value, but you can overwrite it in the ERBRMF04 Parmlib member. Adjust this interval if needed to match a problem occurrence that you are investigating.

For Monitor I: Again, use the default sampling cycle of 1 second. For the reporting interval, a value of 30 minutes is fine to start with. Adjust this if you prefer, or if you need to hone in on or drill down on a problem. (Remember that this value needs to be defined with the SMF parameters because of the SMF synchronization.)

This book discusses key RMF indicators from the different monitors that can be used in a daily report and in problem diagnosis.

5.6.1 Analyzing workload characteristics

Much of performance work is done, not at the individual transaction level, but at a higher, *workload* level. Understanding resource requirements by workload is key to effective performance management.

The reasons you should analyze your workload are:

- To understand your system's behavior
- For setting your SLA
- For tuning
  - Where is the pain?
  - Interactions with other workloads
- For input to the capacity planning process
  - Workload growth projections
  - Processor requirements
  - Storage requirements
  - DASD requirements
  - Coupling Facility requirements
  - Network requirements

Identifying workloads

You will need to identify the different types of work in your system. Usually this is done at a service class level. To assign work to service classes, you must classify the various workloads by their unique characteristics and requirements, that is:

- Response time needs
- Resource consumption (CPU, Storage, I/O)
- Priority
- Anticipated growth

Examples of workload identification at the service class level include:

- Trivial TSO (first period)
- Non-trivial TSO
- Batch
- Production CICS
- Development CICS
- IMS™
- Graphics applications
- WebSphere applications

Ideally, you may want to take workload differentiation one level further, matching workloads to true business functions (for example, claims processing and order fulfilment). This may require more detailed data from SMF records.

RMF reports data about different workloads grouped into categories which you have defined as *workloads* and *service classes* in your service policy. The appropriate grouping of workloads is important.

- If you have different applications that should be managed according to the same goals, you should define the same *service class* for them. Applications with different goals need to be assigned to different service classes.
- If you want to get separate reporting for different applications in the same service class, you can define separate *report classes* for each of them. Reporting for report classes is possible with the same level of detail (report class period) as for service classes.

5.6.2 Measuring resource utilization by workload

Figure 5-7 illustrates one way to report your resource utilization by workload.

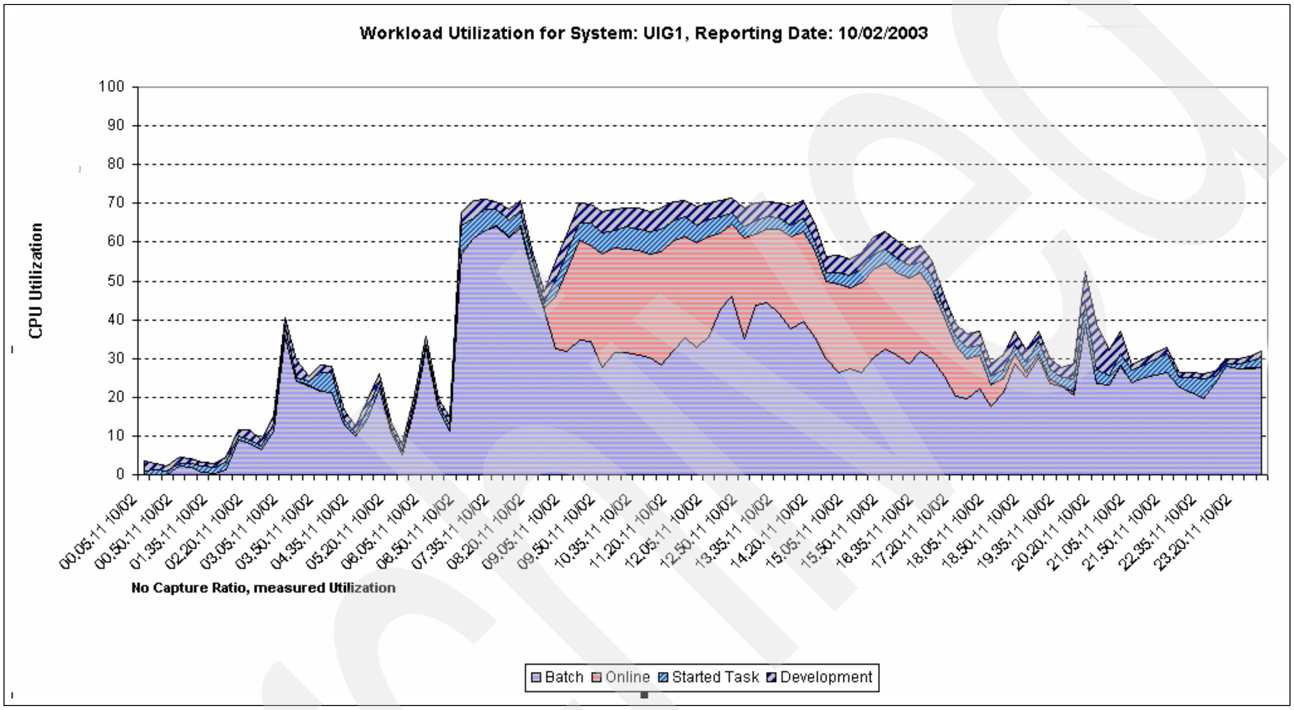


Figure 5-7 Resource utilization by workload

You can use the Spreadsheet Reporter to create spreadsheet files from Postprocessor data and display them graphically. The following sections show you how to use the sample macros for creating the graphics that help you in understanding and analyzing the performance of your system.

In a similar fashion (which could be graphical or numerical), you will need to record such data to:

- Measure resource consumption:
  - CPU utilization
  - I/O rates
  - Central storage usage (In older systems, you can have expanded storage, and you might monitor this, but we will not discuss this item.)
  - Coupling Facility data
- Understand what makes up response time:
  - Waiting for and using the resources
  - Use Monitor III Group Response Time report

- Understand the factors that influence the previous items.

5.7 Workload Management (WLM) highlights

This section includes some basic information about WLM to help you understand some RMF reports.

WLM is a z/OS component in charge of performance management in your sysplex. It *measures* your transaction metrics and *manages* them by changing their priorities in the system queues.

WLM operations are driven by performance goals defined by the installation. The priorities are dynamically calculated and managed in order to pursue the goal. WLM measures Tw (delays) suffered by ASs and enclaves for certain resources (not all of them). When the goal is not being globally (sysplex) achieved, WLM raises the local priority of transactions (receiver) in relation to the resource causing the greatest delay. To provide additional resources to these transactions, WLM needs to find another service class (donor) that can afford to lose some of the priorities.

Every time a transaction arrives in z/OS, the transaction work manager, such as CICS, WebSphere, or TSO, issues the IWMCLSFY (WLM API) passing transaction external properties to WLM. These properties are matched against the installation Classification Rules to derive the service class and optionally the report class that the transaction belongs to. From this point, it is now a matter of enforcing the goals described in the periods of the service class.

To help this task of enforcing the goal, every 10 seconds, each service class period has its performance index (PI) measured. PI is a metric that indicates how far the current values are from the assigned goal, independently of the type of the goal. By definition, when PI is between 0.0 and 1.0 the goal has been achieved (“happy” state), or when greater than 1.0 the goal is not achieved (“unhappy” state), and the higher the value the worse is the scenario.

Figure 5-8 shows the different types of WLM goals. WLM aims to equalize the performance index (PI) of all the service class periods having the same importance. The major action to achieve this objective is to exchange priorities (mainly dispatching and I/O) among donors (happy) and receivers (unhappy).

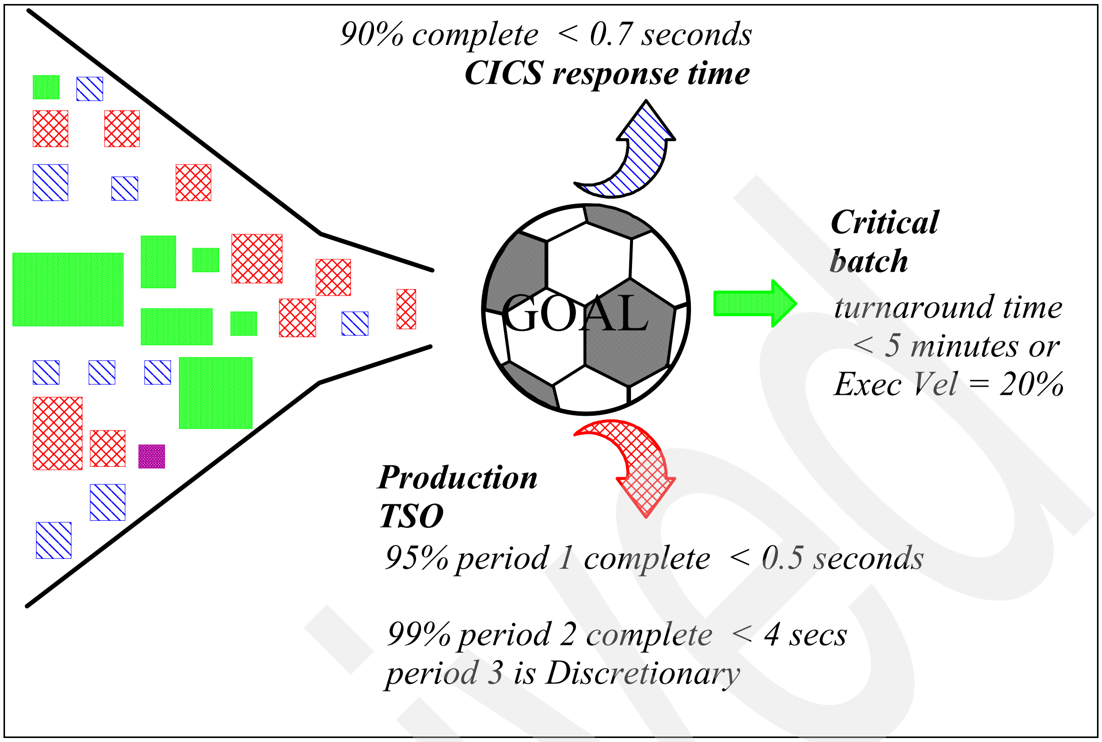

90% complete < 0.7 seconds
CICS response time
GOAL
Critical batch
turnaround time < 5 minutes or Exec Vel = 20%
Production TSO
95% period 1 complete < 0.5 seconds
99% period 2 complete < 4 secs
period 3 is Discretionary

Figure 5-8 Types of goals

Example 5-6 is an example of a service class named TSO4, that for accounting purposes belongs to the TSO Workload. It has three periods. 95% of the transactions ending in the first period (consuming equal or less than 200 service units) should have response time between 0.0 and 0.5 seconds.

99% of the transactions ending in the second period (consuming between 200 and 600 service units) should have response time between 0.0 and 4.0 seconds.

Transactions consuming more than 600 service units should have a discretionary goal.

Example 5-6 Definition of the service class TSO4

```
Service Class Name . . . . . . TSO4      (Required)
 Description  . . . . . . . . .Production TSO
 Workload Name  . . . . . . . .TSO       (name or ?)
 Base Resource Group  . . . . .________  (name or ?)
 CPU Critical . . . . . . . . .NO        (YES or NO)

 Specify BASE GOAL information.  Action Codes: I=Insert new period,
 E=Edit period, D=Delete period.

         ---Period---  ---------------------Goal---------------------
 Action  #  Duration   Imp.  Description

   __
   __     1    200      1    90% complete within 00:00:00.700
   __     2    400      2    90% complete within 00:00:01.500
   __     3             Discretionary
```

Here are some observations on the RMF SYSSUM report, shown in Example 5-7:

- TSO4 first and second period are reaching their goals: their PIs are less than 1.0, that is 0.50 each.

- TSO4 duration values are correctly defined because almost 80% (14.85 / (14.85 + 3.15 + 0.083)) of the transactions are finishing in the first period (trivial), as recommended.
- The TSO4 transaction average response time is 0.200 sec.
- The GRI2 service class is not reaching its velocity goal (Goal: 75%, Actual: 65%). By the way, GRI is a CICS customer (home made), not using WLM API to tell about its transactions. Then, you should not use response time as a goal and also you cannot see values in the last four transaction-related account columns. The same happens with CICS, when you only associate the goal with the CICS ASs (subsystems STC or JES).
- All the CICS ASs, in the Sysplex where the transactions from SC CICS1 run, have their priorities associated with such transaction goal (AVG RT= 0.750 sec). Within the measurement period, we had a rate of 4034 executed transactions/sec of such type.
- All the other CICS ASs have their priorities associated with a velocity goal of 60% in the service class CICS2.
- The Actual Time should be the sum of the Execution Time plus the Wait time, so it should be greater than or equal to the Execution time. However, in some cases for CICS transactions it does not hold true. This is because these two fields report on a different set of transactions.
  - EXECUT time can include transactions which originated on a remote system as well as transactions originating locally.
  - ACTUAL time includes response times for only transactions originating locally.

  If the remote transaction tends to be longer than the local transaction, EXECUT could be greater than ACTUAL, as the case where 0.717 is greater than 0.557.
- The System SC is apparently not doing well, with an execution velocity of 20%. But, address spaces with a Dispatching Priority of 255 usually have a lot of code executed in cross memory mode. For this reason, the reported numbers are not always reflecting the total activity for these service classes.

Example 5-7   Sysplex Summary report

```
             ------- Goals versus Actuals --------             Trans --   Avg. Resp. Time ----
                   Exec Vel   --- Response  Time ---    Perf   Ended   WAIT   EXECUT  ACTUAL
Name         T  I  Goal Act   ---Goal--- --Actual   -- Indx    Rate   Time     Time

CICS         W          N/A                                      4034   0.000   0.717  0.557
CICS1        S  1       N/A   0.750 AVG   0.557 AVG      0.74    4034   0.000   0.717  0.557
CICS2        S  1    60  59                              1.02   0.000
GRI          W           65                                     0.000   0.000   0.000  0.000
GRI2         S  1    75  65                              1.15   0.000   0.000   0.000  0.000
SYSTEM       W           39                                     0.000   0.000   0.000  0.000
SYSSTC       S      N/A  68     N/A                             0.000   0.000   0.000  0.000
SYSTEM       S      N/A  20     N/A                             0.000   0.000   0.000  0.000
TSO          W           66                                     18.08   0.003   0.198  0.200
TSO4         S           66                                     18.08   0.003   0.198  0.200
             1  1        47    0.700 90%        97%      0.50   14.85   0.000   0.096  0.096
             2  2        83    1.500 90%        96%      0.50   3.150   0.015   0.319  0.334
             3  D        78                                     0.083   0.000   13.72  13.72
```

6

Using RMF for performance monitoring and problem diagnosis

Instead of diving in at a low level to investigate perceived performance problems, it is always a good idea first to take a look at the overall performance health of the system. RMF provides a number of facilities to help you do this health check.

In this chapter we how to use the data that RMF provides to perform the following tasks:

- Run a health check
- Use Monitor III for problem analysis
- Investigate Coupling Facility performance

6.1 RMF approaches to performance management

Generally, there are two major RMF approaches to tackle real-life performance problems:

- **Response** - This approach focuses in the response time of a set of key business transactions, hopefully with goals coming from a Service Level Agreement (SLA). It is mainly concerned with Postprocessor Workload Activity reports and several Monitor III reports. One variation of this approach (for another set of transactions) is to replace response time observation by other metrics, such as WLM execution velocity% or RMF workflow%.
- **Throughput** - This approach focuses on the utilization and other system indicators showing stress in key resources, such as CPU, I/O, storage, and coupling facility. It is mainly concerned with Postprocessor reports (which are the base for Spreadsheet Reports graphics) describing these resources. You should create reports for utilization of the key components of your system; then, at a glance, you can see whether everything is running fine.

Our daily work with Performance Management can be divided into two types of tasks: health checks and problem diagnosis.

- Health checks, through daily performance monitoring activities, are the way you can:
  - Recognize any of the pain indicators as being potentially a problem on your sysplex
  - Discover the system in the sysplex where the most important performance problems are occurring
- Problem diagnosis involves a more in-depth analysis of the RMF data to determine which component of the response time holds the most promise for improvement.

6.2 Running a performance health check

First, do not confuse the idea of a performance health check with the software tool available to do health checks, for example for high availability in a Parallel Sysplex. The performance health check mentioned here is just a series of human procedures done in a daily basis to keep track of your system performance. It verifies whether:

- Goals of the most business important transactions are being achieved by WLM. This is the response approach.
- System resource indicators are within normal ranges, meaning at typical daily values or following general recommendations such as rules-of-thumb (ROT). This is the throughput approach.

If a problem is perceived during your health check tasks, you should go through the problem diagnosis in-depth task described in “Running problem diagnosis” on page 194.

The purpose of health checking is to detect performance problems early on, and not just to be waiting for users to call the help desk claiming that “nothing is running.” (Of course, this is normally not true, but it is what the users tend to think.)

How should you do daily performance monitoring (health checks)? Keep in mind that users or customers want to be satisfied with the service that they receive from the system you are responsible for. Your focus should be on observing the SLA. If the goals described there are attained, then you are providing the service that your customers are expecting.

With WLM, you can just define your goals in the service policy and let the system run. With WLM operating in your system, this health check is done frequently and automatically (every

10 seconds), at least for certain types of resources (CPU, I/O, storage and server address spaces) and for their accessing transactions.

6.2.1 Health checking focusing on throughput

A health check for throughput means that you should set up several permanent report procedures based on the data RMF is gathering. This is important because otherwise you may not be able to analyze problems that occurred several days ago, and you would not have any historical data that could be used for creating trend reports. By the way, the Monitor III Trend report has detailed information about the system trends.

Therefore, ensure that the SMF and VSAM data sets for Monitor I and Monitor III are large enough to retain a relevant amount of data.

As mentioned in 5.6.2, “Measuring resource utilization by workload” on page 174, we need to concentrate on the key resources of the system.

All the reports in this chapter have been created using Spreadsheet Reporter; Table 6-1 references the corresponding spreadsheet and chart names.

Table 6-1 Metric and spreadsheet cross reference

Metric	Spreadsheet Name	Chart Name	File Name
Processor			
CPU Utilization	Summary Report	SumChart	Rmfn9sum.xls
CPU Contention	System Overview Report	OneCpuCont	Rmfy9ovw.xls
LPAR Management Time	LPAR Trend Report	RepTrd	Rmfr9lp.xls
Capture Ratio	System Overview Report	OneCpuUtil	Rmfy9ovw.xls
I/O Subsystem			
DASD I/O Intensity	Device Overview Report	Overview	Rmfx9dev.xls
DASD Response Time	Device Overview Report	RepAllDevs	Rmfx9dev.xls
Cache Hit Ratio	Cache Subsystem Report	SSIDOvw	Rmfr9cac.xls
Central Storage			
Paging Activity	System Overview Report	DaysFaults	Rmfy9ovw.xls
Coupling Facilities			
Service Times	CF Trend Report	RepSubChn2	Rmfr9cf.xls

6.2.2 Checking the processor

There are several views that we can use when we want to perform a health check for the processor. We decide to monitor the following performance areas:

CPU utilization Get the average and peak utilization during prime shift for each day.

CPU contention Analyze the In-Ready queue for contention.

LPAR management time Check whether the LPAR configuration works properly.

Capture ratio Verify that the captured time of the processor is in a good range.

CPU utilization

The general guideline for CPU utilization is to compare it with your installation Saturation Design Point (SDP) to see if the 100% mark is being reached during some moments. Ideally, the SDP value should not be exceeded. For further details, refer to “Saturation design point (SDP)” on page 159. In Figure 6-11 you can see the CPU utilization spread across a period of time; this can help you understand the behavior and decide where—in which peak day, shift, or hour—you might need to investigate further.

The following items are some observations about this metric:

- A low CPU utilization value may indicate *latent demand*, although at this time, we do not know yet if this is the case. In other words, the processor could look under-utilized because of bottlenecks in other areas that need to be fixed. Storage contention or I/O problems can be the reason for bad application performance and low processor utilization. When those bottlenecks are removed, the processor utilization may increase.
- A high CPU utilization indicates that there is a queue of dispatchable units (it starts to build up after 35% utilization and at 100% produces a theoretically infinite queue length). What we hope is that the dispatchable units in the queue are not the ones that are critical for the business. We can verify that by inspecting the Monitor III Processor Delay report, where we can see who is delaying whom. The existence of the CPU queue can also be verified by CPU contention, which is discussed in the next section.
- Although z/OS can support the CPU at 100% busy constantly, there is an increased chance of:
  - Deadlocks in resources represented by ENQs and locks.
  - Dispatchable units with low priority holding ENQs and locks, causing temporary contention.

  Consequently, such 100% utilization for a long period of time should be avoided.

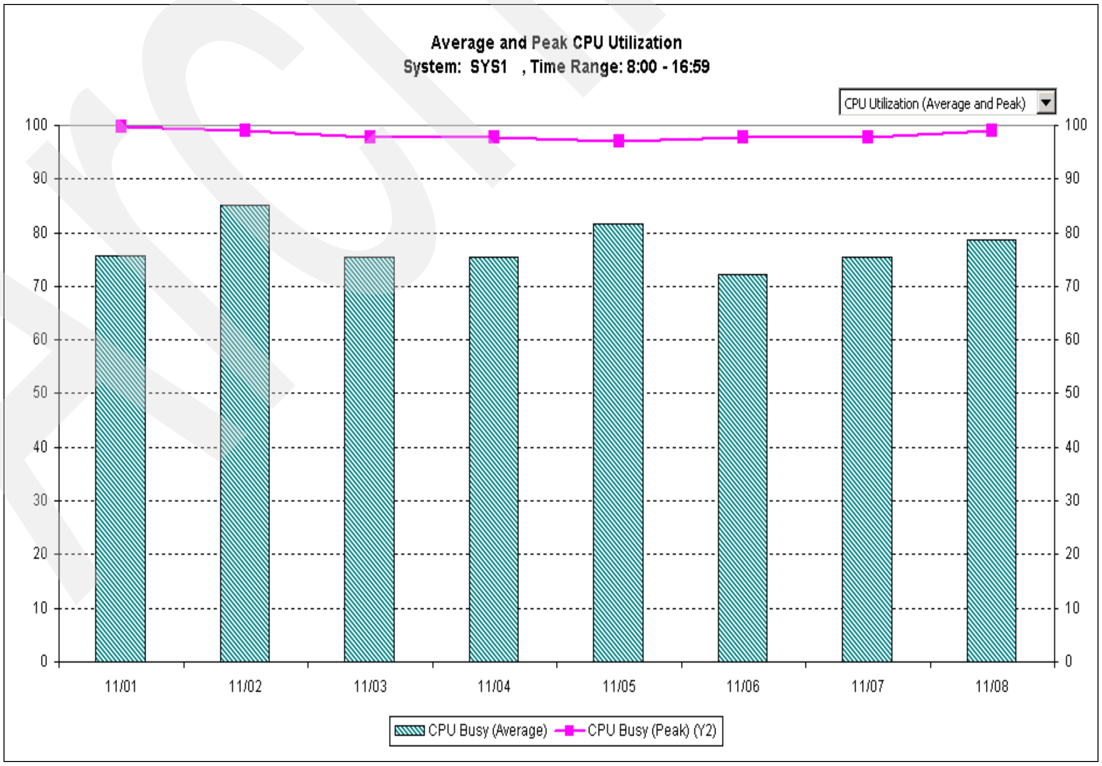


Figure 6-1 CPU utilization

Do not forget to take a look at the *PROC Workflow% field. It is a powerful metric to evaluate the CPU performance.

CPU contention

Figure 6-2 shows the In-Ready queue, sampled by RMF. It includes address spaces and enclaves with dispatchable units ready (delayed by CPU) and active (executing CPU). High CPU utilization causes a high In-Ready queue.

The behavior of the In-Ready queue is taken as an indicator of CPU contention. Consider the following observations:

- The percentage of time when the CPU is utilized and there is at least one address space waiting for the logical processor is already considered contention.
- The contention begins at the point where the In-Ready queue is longer than the number of logical processors. In this case there is one active AS/enclave per each logical CPU.
- The distribution of queues with different lengths can be considered valuable data for our study.

In Figure 6-2, you see that in the range 6:05-10:05 there is heavy contention in the system, with values up to 100%.

This graphic can be used as a confirmation of the average CPU utilization. The key question here is: who is delaying whom, if we have CPU contention. Once again, this type of information can be tracked in the Monitor III Processor Delay report. There are a few reasons why one AS/enclave may be delayed by another:

- The ones with the most difficult and more important goal have higher dispatching priority (DP) and cause delays in the rest. This is a normal situation.
- The ones with the easiest and least important goal have temporarily higher dispatching priority (DP) and cause delays in the rest. This is an apparently abnormal situation. Here, you should trust in WLM (for example, the CPU delay figure can be small and the holder is I/O bound). If you do not want to do this, you can use WLM's CPU Protection facility.
- Someone with a small DP is causing delays for the rest. In this case, when the figures are not at a minimum level, you must blame the non-full preemption functionality of z/OS (mainly from z/OS 1.6).

> **Note:** The difference between MVS BUSY time and LPAR BUSY time is another good indication of latent CPU demand in the observed LP. If significant, and the LP is important to the business, there might be a need to increase the LP Weight (to get CPU from other LP in the same CPC) or to acquire more CPU resource.

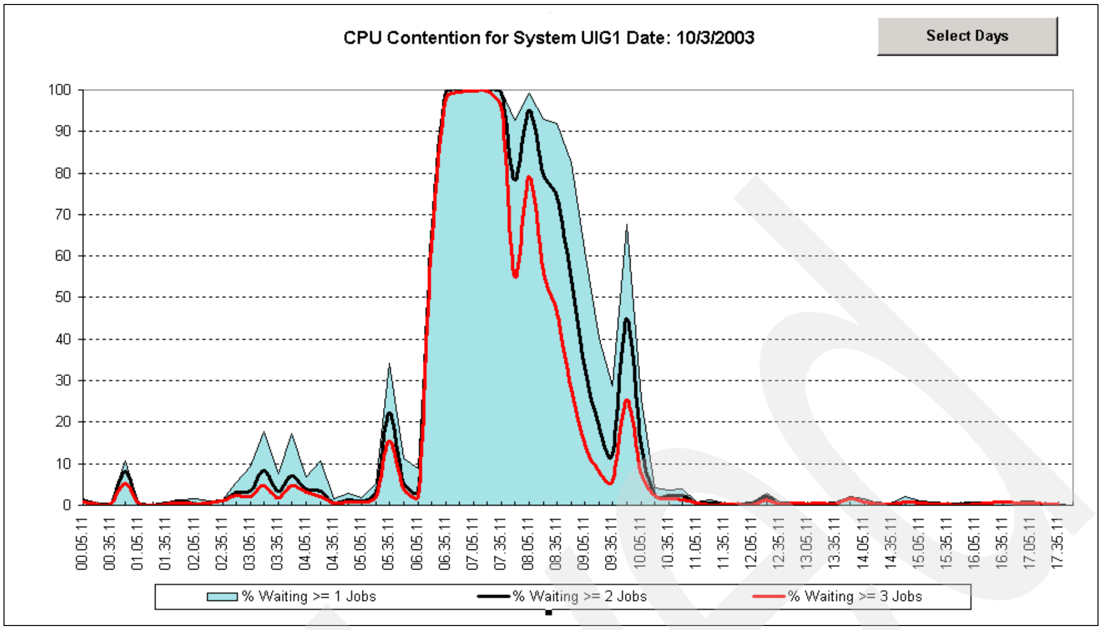


Figure 6-2 CPU contention

LPAR management time

If we are lacking CPU, a good idea is to verify the existence of non-captured CPU consumption time. LPAR management time indicates the amount of processor needed for the LPAR administration of the LPs. See Figure 6-3.

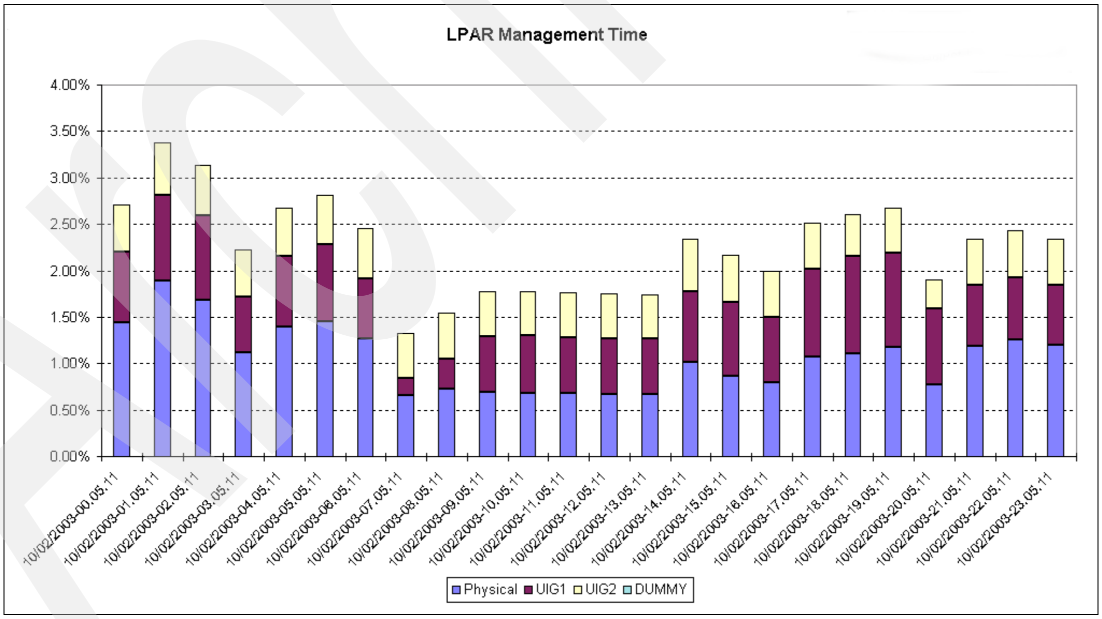


Figure 6-3 LPAR hypervisor management time

In this spreadsheet report, you can see the maximum hypervisor management time was a little lower than 3.5% that is an acceptable figure.

One possible cause for a high figure at such time is the low utilization effect (LUE). LUE increases z/OS CPU time and LPAR time—consequently decreasing the capture ratio, which is discussed in “Capture ratio” on page 186—because the CPUs have low utilization. This increase is explained by the *search for work* activities. Through the OPT keyword CCCAWMT (defaulted to 12000), the installation can require the activation of Alternate Wait Management (AWM). When AWM is activated, SRM and LPAR cooperate to reduce LUE effects.

However, even if the low utilization effect is reduced, the LPAR management time is still somewhat higher on lowly utilized machines than on more highly utilized ones. So, keep an eye on the CPU utilization when justifying the reasons for a high LPAR management time.

Example 6-1 is an RMF Postprocessor Partition Data report. Let’s take a look at the report and the following comments.

Example 6-1   Postprocessor Partition data report

```
                                         P A R T I T I O N   D A T A   R E P O R T
                                                                                                                            PAGE    2
             z/OS V1R4                SYSTEM ID BSBA            DATE 02/09/2004            INTERVAL 30.00.004
                                      RPT VERSION V1R2 RMF      TIME 16.30.00              CYCLE 1.000 SECONDS
-
 MVS PARTITION NAME                  LP0201
 IMAGE CAPACITY                         400
 NUMBER OF CONFIGURED PARTITIONS          6
 NUMBER OF PHYSICAL PROCESSORS           16
                   CP                    16
                   ICF                    0
 WAIT COMPLETION                         NO
 DISPATCH INTERVAL                  DYNAMIC
-
 --------- PARTITION DATA -----------------  -- LOGICAL PARTITION PROCESSOR DATA --   -- AVERAGE PROCESSOR UTILIZATION PERCENTAGES --
                   ----MSU----  -CAPPING--  PROCESSOR-  ----DISPATCH TIME DATA----   LOGICAL PROCESSORS  --- PHYSICAL PROCESSORS ---
 NAME       S   WGT  DEF    ACT  DEF   WLM%  NUM   TYPE   EFFECTIVE      TOTAL      EFFECTIVE    TOTAL  LPAR MGMT  EFFECTIVE  TOTAL
 LP201      A   260  400    281  NO     0.0  12    CP   05.04.07.782  05.05.23.066     84.48     84.83     0.26       63.36  63.62
 LP202      A   290   80     25  NO     0.0   5    CP   00.26.32.475  00.27.09.314     17.69     18.10     0.13        5.53   5.66
 LP203      A   250  180     67  NO     0.0   6    CP   01.12.04.368  01.12.51.197     40.04     40.47     0.16       15.02  15.18
 LP204      A   100   70     55  NO     0.0   6    CP   00.59.19.144  00.59.50.125     32.95     33.24     0.11       12.36  12.47
 LP206      A    20    0      0  NO     0.0   1    CP   00.00.13.207  00.00.16.595      0.73      0.92     0.01        0.05   0.06
 LP207      A    30    0      0  NO     0.0   1    CP   00.00.10.210  00.00.13.276      0.57      0.74     0.01        0.04   0.05
*PHYSICAL*                                                            00.05.42.786                         1.19               1.19
                                                         ------------  ------------                        ------      ------ ------

   TOTAL                                                 07.42.27.188  07.51.26.362                          1.87      96.36  98.23
```

The difference between the total dispatch time and the effective dispatch time for LPs is the LPAR management work in the specific LP. This value for LP201 is 0.26%. Additionally, the total CPU time identified by the name *PHYSICAL* is the LPAR management time that cannot be attributed to any LP. Adding all numbers in the LPAR MGMT column we have the total figure (1.87% in the report).

The use of dedicated processors reduces LPAR management work; however, with the exception of ICFs for production coupling facility LPs, dedicated CPUs are not recommended.

The LPAR management time depends on the number of LPs and the number of logical CPUs, even though it is not a linear function.

Based on that, there is a recent ROT: the sum of all logical shared processors should not be more than *triple* the number of physical processors. Otherwise, the LPAR management time to reassign the PUs to the logical CPUs can increase to an unacceptable level.

However, keep in mind that there is a not clear recommended value for such LPAR management consumed time. Environments have become so complex that rules of thumb can no longer be used. There are moments that the LP need lots of logical CPs (increasing such time) and other moments not. Then, we strongly recommended the activation of the IRD function. IRD dynamically determines the best cost/benefit between the LPAR management time and the required logical CPU capacity and speed.

Finally, we must remember also that the LPAR dispatching on z990 has been improved by:

- The use of affinity bits for optimal dispatching
- The L1 buffers are no longer cleared when switching from one LPAR to another.

However, even though these modifications do not alter the LPAR management time, they make the logical CPU appear faster.

Capture ratio

As explained in 5.3.1, “Capture ratio” on page 169, capture ratio represents the percentage of CPU time which can be counted to application CPU time. Capture ratios have improved over time on z/OS systems. Alternate weight state management has reduced the impact of under-utilized LPARs. A sample of our capture ratio curve is shown in Figure 6-4. If the capture ratio is less than 80%, you need to investigate further. Refer to “Running problem diagnosis” on page 194 for details.

> **Rule of thumb:** The capture ratio should vary between 85 and 93 percent. This value can be significantly smaller in under-utilized systems.

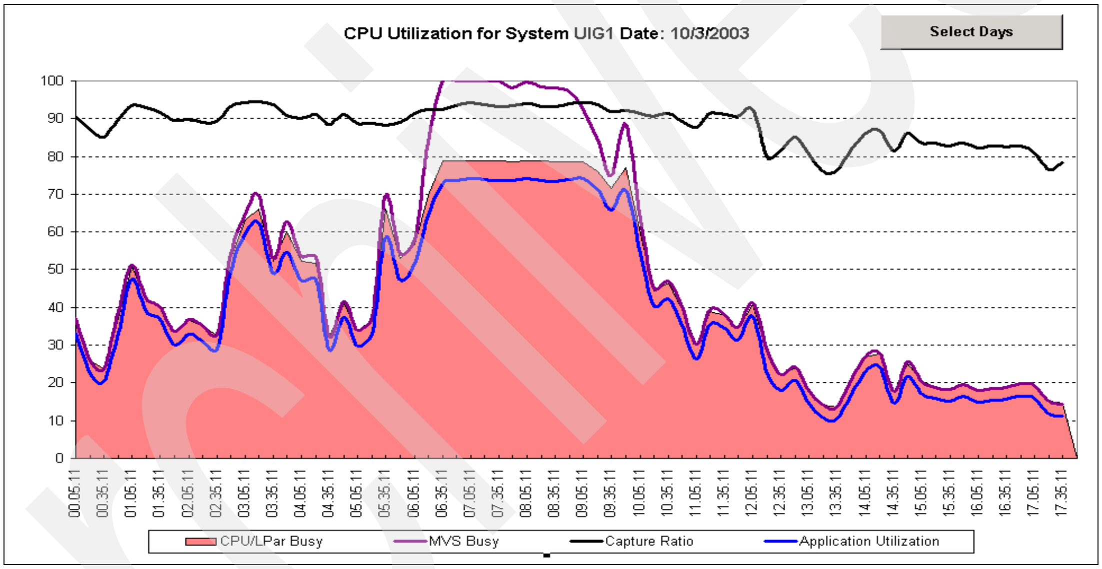


Figure 6-4 Capture ratio

6.2.3 I/O Subsystem

To cover key aspects of the performance of the I/O subsystem, we discuss the following reports:

DASD I/O intensity A combination from utilization and response time that is a good performance indicator.

DASD response time Here we get a one day overview of the response times for selected devices.

Cache hit ratio At a glance, we can see the cache hit ratio for all DASD subsystems.

DASD I/O intensity

Refer to “I/O performance and metrics” on page 165 for more informations about I/O operations metrics. The DASD I/O intensity, illustrated in Figure 6-5, can be taken as an indicator of device utilization and contention:

I/O intensity = Response time × Activity rate

Some comments about Figure 6-5:

- Even the extremely high response time for a single device is not of interest if the activity rate of this device is very low – if these high response times are not affecting key transactions. For this reason, the I/O intensity is a useful measurement to detect overutilized devices that deserves further attention.
- Theoretically, the I/O intensity for a device may be higher than 1000 ms/sec because the IOS queue time is part of the I/O response time.
- DASD performance analysis method based in I/O Intensity:
  - For each DASD device calculate the I/O Intensity
  - Sort them by descending I/O intensity
  - Here you have the devices where you can work (and your work is efficient)!
- For a rough performance evaluation of the entire I/O subsystem, the following guideline can be used: the accumulated I/O intensity for the five most highly utilized devices should not exceed 1500 ms/sec during the prime shift.

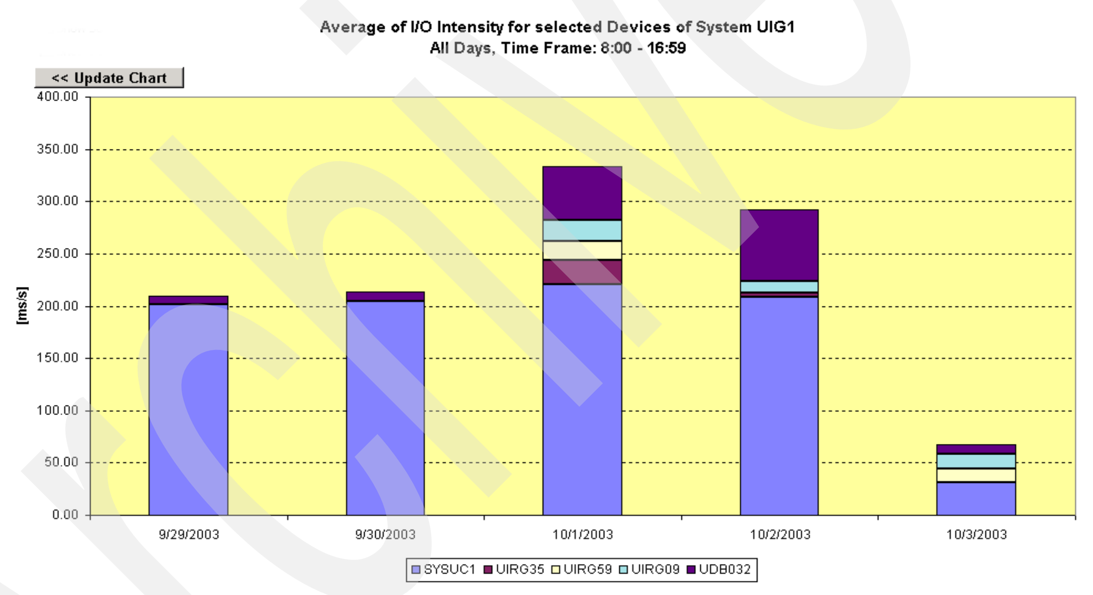


Figure 6-5 DASD I/O intensity

In our report the device SYSUC1 has the highest I/O intensity values during different moments.

DASD response time

The DASD response time can be used as an indicator of device performance mainly for devices selected using the I/O intensity figures, as explained in the previous section. The following formulas are related to I/O response time (I/O Tr):

I/O Tr = IOSQ Time + Pending Time + Connect Time + Disconnect Time

Where:

I/O Tw = IOSQ Time + Pending Time

I/O Ts = Connect Time + Disconnect Time

If the response times exceed the observed average for those devices with high activity rates, you should start further investigation.

Some workloads are prone to higher response times, which may still be quite acceptable because they produce high service times. For example, DB2 sequential prefetch, DB2 castout, and page/swap I/O are I/O operations characterized by a high amount of data transfer per I/O operation.

We have to combine the values from Figure 6-5 on page 187 and Figure 6-6 to understand the numbers.

- Device SYSUC1 has the highest I/O intensity with a value of 202, which is the result of an average response time of 2.1 ms and an average activity rate of 95 I/O/sec.
- Device UDB032 is the device with the longest response time. It has (for most of the time) a response time of about 10 ms. But due to the low activity rate, the I/O intensity is small, as can be seen in Figure 6-5. Therefore, this response time is not a major problem - if no key transactions depend on such I/Os.

Obviously, when there is a device with high I/O intensity and high I/O response time (I/O Tr), we must decrease the:

- I/O rate
- Average I/O response time (I/O Tr)

Refer to “Techniques to reduce the I/O response time” on page 168 to learn how to decrease I/O rate and average I/O response time. Also refer to “Analyzing the stressed volume” on page 228.

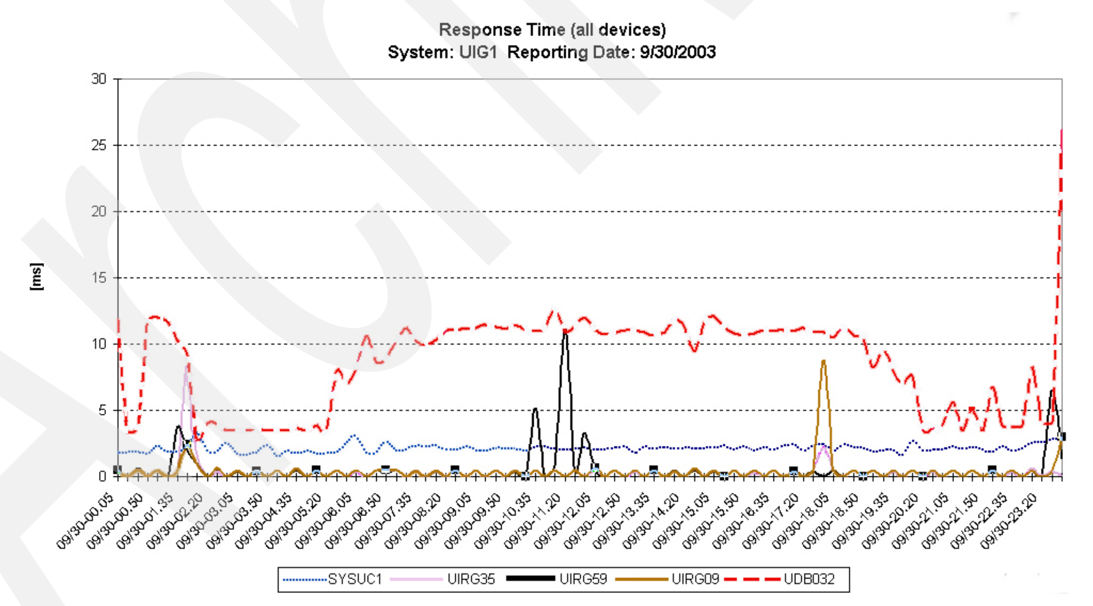


Figure 6-6 DASD response time

Cache hit ratio

The major reason for the existence of cache in DASD controllers is to decrease the I/O disconnect time.The cache hit ratio can be taken as an indicator of cache DASD controller effectiveness. It is the ratio of the hits in cache divided by the number of cacheable I/Os. The following comments concern cache hit ratio:

- Check whether all devices are experiencing a good hit ratio (80% or higher).

- Pay special attention to those devices with high I/O intensity.
- Observe that since the last decade in modern controllers, the huge majority of writes (almost 100%) are hits. So, it is recommended when you derive the Cache Hit Ratio data, to discard the writes.

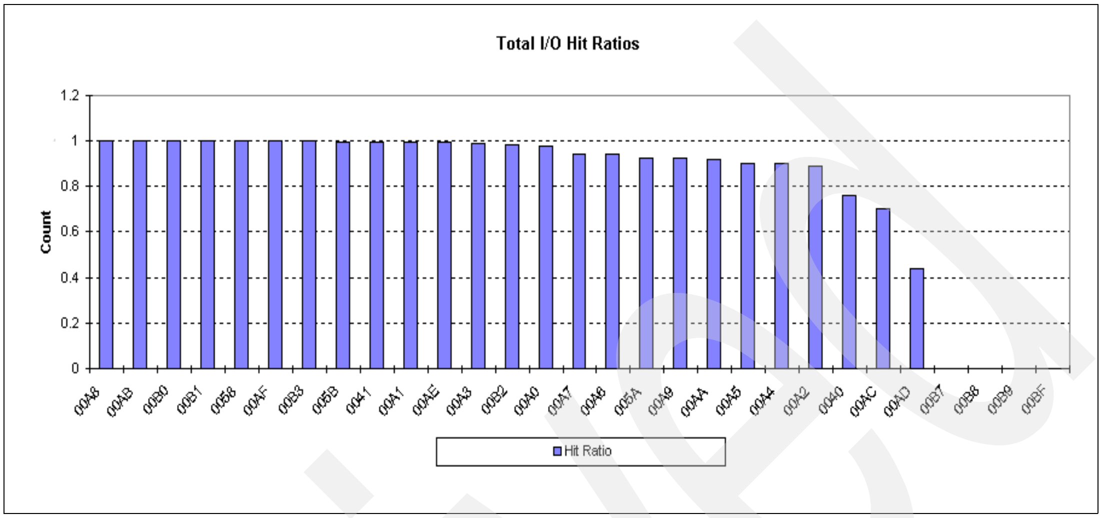


Figure 6-7 Cache hit ratio report

- The occurrence of a low hit ratio for several devices cannot be interpreted as a sure sign of the need for more cache. It may be caused by data sets that do not cache well (said to be cache unfriendly), or the use by SMS software of bypass cache function.

  One example of a cache unfriendly code is DB2. An example scenario follows:

  Anytime you have a read miss in the storage buffer pool, an I/O operation is executed. Data is then copied from disk in the controller cache and in the buffer pool. The time that such data may reside (without references) in DB2 buffer is larger than in the controller cache because of the enormous size of the buffer pool. So in a next miss in the buffer, we will have a miss in the cache again. For this reason, keeping this data in cache can only cause cache pollution, making worse the I/O performance of non-DB2 data.
- There are several suggestion to improve your cache hit ratio in general:
  - Buy more cache or verify how much is the real size of you productive cache. Check how much is being used for cache management.
  - Use data compression.
  - Decrease the load of the controller by moving out data sets.
  - Maybe a recommendation is to declare DB2 files and other cache unfriendly workloads as maybe-cache in SMS storage class.
- As you can see in our graphic, some devices are presenting a cache hit ratio less than 80%.

> **Rule of thumb:** Good: >0.9 – Critical: <0.8

Central storage

To monitor central storage you can use the Paging Activity report, where you can see the average paging rate for several days.

Paging activity

In the past, we had three system components as the major resources that required a lot of attention: CPU – I/O – storage.

Nowadays, storage doesn't need much attention in most installations. This is because CPCs generally have enough central storage to run production without any page fault problems, with one minor exception being certain USS workloads already presenting some non-zero figures. Nevertheless, if you want to get a good feeling about this resource in your system, you can get the page fault rate from RMF, as the report shows in Figure 6-8.

Although a page fault is a rare situation nowadays, there are cases of surges of page-outs towards the slots in local page data sets, such as during a huge memory dump. So, you can be preemptive and define many and large local page data sets to avoid an unplanned lack of auxiliary storage, or you can react to the IRA messages appearing on the z/OS console.

We would suggest that care be taken after a migration to z/Architecture (central storage larger than 2GB per LP) about the page movement within central storage. This figure is presented on the Monitor I Paging Activity report, under the fields:

- Page Movement Within Central Storage (number of pages per second)
- Page Movement Time% (CPU% consumed to perform such movements)

This movement is a type of overhead and is mainly caused by the implementation of recoverable central storage, through the RSU parameter in IEASYSxx. With recoverable central storage active, it is possible to move central storage manually from one LP to another LP. However, due to the huge amount of central storage this might not be necessary any longer. So, you should reconsider this value for your installation and eventually reset to zero the RSU value to decrease the major reason for such page movement.

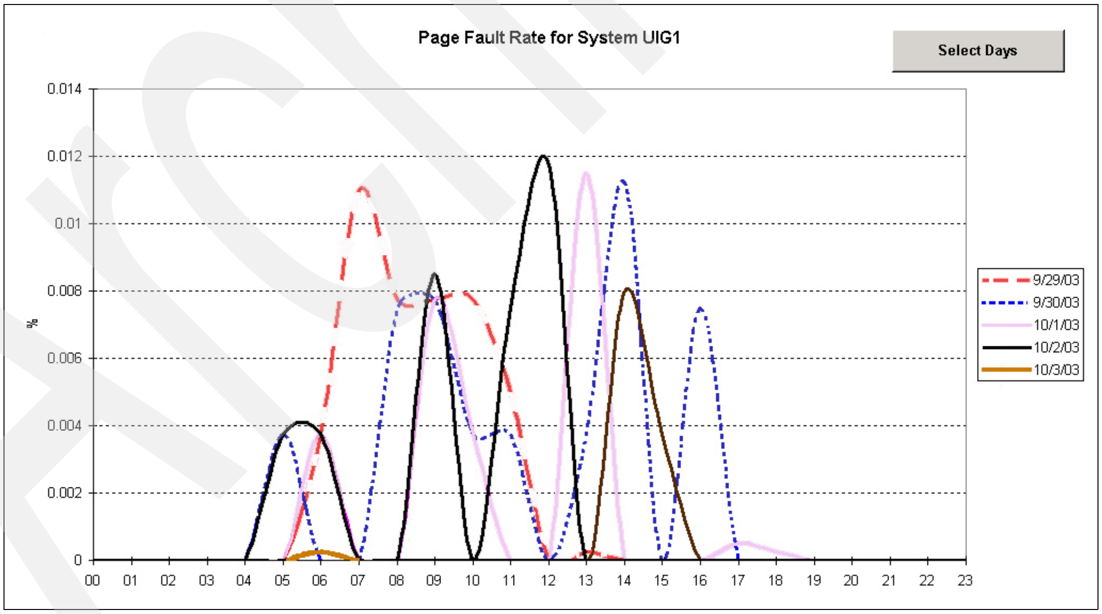


Figure 6-8 Paging activity

In Figure 6-8, it looks like there is significant paging activity in this system, but if you observe the scale, you notice that the paging is only between 0.0 and 0.014 pages/sec. This means that there is no paging activity.

6.2.4 Coupling Facility

To get performance information for the Coupling Facility in the health check inspection, you can use the following metrics:

- Coupling Facility service time
  Service times for synchronous and asynchronous requests.
- Service time and Changed requests
  The number of changed requests (from synchronous to asynchronous) is an indicator of possible performance problems.

Synchronous and asynchronous CF requests

Before we start with the reports, let us review the concept of synchronous and asynchronous CF requests.

- Synchronous requests: The z/OS CPU is placed in a loop by XES (z/OS component in charge of accessing the CF), waiting for the completion of the CF request. The synchronization referred to here is between the requesting task (not placed in wait state) and the execution of the request in the CF. Some observations:
  - It is response-time efficient because the requesting CP spins until the response is received from the CF, saving the overhead of the task being suspended, saving its status, swapping the registers, being dispatched again, and so forth.
  - But, it can burn z/OS CPU cycles if CF takes a long time to respond.

  Therefore, the synchronous requests should be used for short requests.
- Asynchronous requests: Instead of holding the CPU, XES releases it once the CF requests have been sent. Some observations on this approach:
  - It frees up CPU for other work while the request is processed.
  - It causes modifications in the dispatcher component logic because the task must be undispatched, then later re-dispatched though manipulation of HSA bits.

  This should be used for longer requests, when some delays are acceptable.

The CF structure exploiter decides if the request should be synchronous or asynchronous. However, it is up to XES to make the final decision. XES makes use of the CF Sync/Async heuristic algorithm, where it selects the most efficient way of handling each request. The request will be synchronous if the z/OS CPU loop time is less than the z/OS CPU time to suspend and later to resume the requester dispatchable unit. If larger, then it will be asynchronous. The decision depends on two variables:

- The current actual synchronous response times for each CF request type. For duplexed structures, XES keeps the time from the start of the first operation to the end of the last one.
- A threshold, specifically the amount of time needed in the z/OS CPU to implement asynchronous requests, such as saving status, swapping registers, load status, and being dispatched again. This threshold depends on the speed of the CPU that z/OS is running.

This algorithm can modify a synchronous request with the immediate option, as for example, the CF requests for the ISGLOCK GRS structure to asynchronous. This heuristic algorithm explains that if you install a faster z/OS processor, the average CF response time gets slower because the threshold decreases, the opposite if you add more CPUs.

Keep in mind that a synchronous request could be modified by the heuristic algorithm to asynchronous, without being considered as a changed request.

A synchronous request can be changed to asynchronous also when the CF link subchannels are busy. These events are reported in RMF reports as changed.

Service time (Ts), request rate, and changed requests

The CF service time, the request rate, and the number of changed requests can be taken as indicators for the performance of the Coupling Facility. However, due to the heuristic algorithm, the number of changed requests is no longer important.

Before discussing CF service time, let us cover the CF queue time (wait time). CF requests may be queued, thus generating wait time Tw, because of either of the following reasons:

- All subchannels (and their buffers) are busy.
- A shared CF link path is busy on the other LP (should no exceed 10%).

There is a guideline indicating that the sum of queued (delayed) requests should be less than 10% of the total number of requests.

Expected service times depend on so many variables that it is almost impossible to give a single recommendation (ROT). You should instead consider which are the factors that can influence the service time, for example:

- Model of the CPC where the z/OS image resides
- CF links:
  - Type: ISC, ICB, IC
  - Length of CF link (10 additional microseconds for every 1Km length)
- Model of the CPC where the CF image resides
- Total demand in the CF and CF links, which may cause high queue time
- Type of the CF structure access:
  - SYNC/ASYNC
  - List, lock, cache
  - Duplexed or simplex
  - Amount of data being moved per request

The utilization of the CF link (derived by the multiplication of the service time by the rate) should not be higher that 30% to avoid increasing the queue. The same schema applies to subchannel utilization.

In Figure 6-9, you can observe that the CF average service time seems not to be affected by raising the request rate from 14 to 16 requests per second. This fact confirms in general the queueing theory, where a high activity only affects the queue time and not the service time.

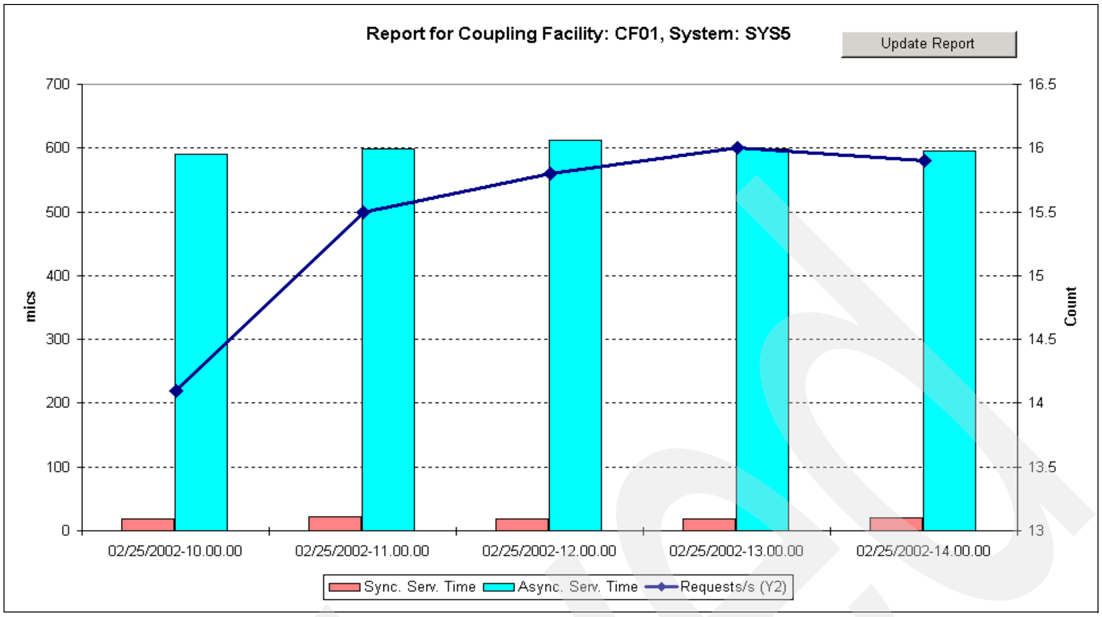


Figure 6-9 Coupling Facility service time

Now, let us take a look at the changed requests graphic in Figure 6-10.

With the presence of the heuristic algorithm, the only cause for changed requests is that a synchronous request is changed to asynchronous because of the lack of available subchannels in the CF link. Then, if we see high figures here, we should confirm that by observing the subchannel queue time and length in the Postprocessor CF Activity report.

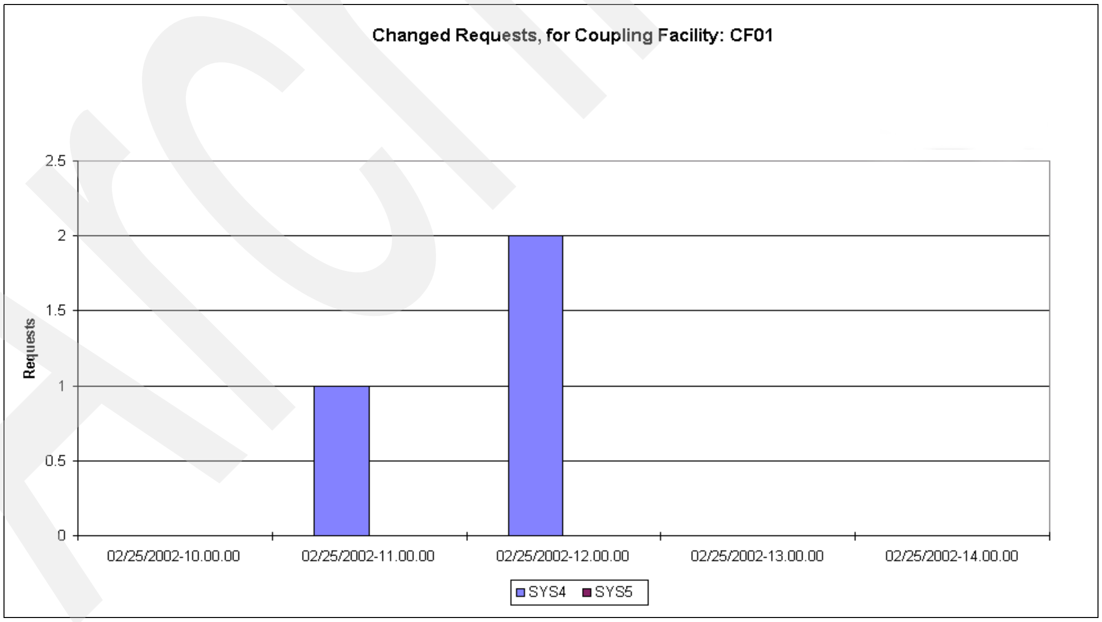


Figure 6-10 Changed requests in the Coupling Facility

6.3 Running problem diagnosis

There are many elements that constitute a performance problem. In this section we describe a set of tasks that you can perform to diagnose them. Most of the time, the main reason for doing diagnostics is that your health check has uncovered something unusual, or because someone complained. Usually, problem diagnosis focusses more on response time issues than on system resource indicators.

Starting with the sysplex view, you get a high-level overview about the performance indicators of your workloads. This leads you to the most relevant systems in the sysplex. There, you can start with the largest cause of delay, decide what you can do to address it, go back and address the next largest cause of delay, and so forth. Continue to do this until the service objectives are met, or until you reach a point of diminishing returns, where further efforts do not pay off.

Note: Here we are assuming that the goals are properly selected and consequently the performance index indicator is a reliable metric.

At this point, you are interested in getting a performance overview at a glance. You have defined a service policy that contains several workloads, each of them with service classes (and report classes) that have goals of different types and importances.

Both Monitor III and Postprocessor offer you some reports that show how your sysplex is running at different levels of details. From the wide range of available reports, we concentrate on the following selections:

- Monitor III Sysplex Summary report (SYSSUM)
- Monitor III System Information (SYSINFO)
- Monitor III Response Time Distribution report
- Monitor III Work Manager Delays report
- Monitor III Common Storage report
- Postprocessor Coupling Facility Activity report

Monitor III offers sysplex reports and local system reports. A good starting point for monitoring the sysplex is the Sysplex Summary report. From this report, you can navigate to other sysplex reports, and you can continue monitoring a single system within the sysplex by using the system reports.

6.3.1 Monitor III Sysplex Summary report

This report shows (based on your option selections) all or only selected workloads, service classes, and report classes with their periods, goals, and actual performance values. The data is shown at the global sysplex level.

You can select what should be part of this report, shown in Example 6-2, by using the report options:

- What types should be reported? These can be everything from workload to service class period, as well as report classes.
- What performance index should be used as the threshold? This allows you to select only the data that has a high enough index that you want to be informed about it.
- What is the importance level of your work? If you want to see only important work, such as high or medium importance, you can select this.

- Are you interested in all the workload groups that you have defined? Then you can display them; otherwise, the report shows only the active workload groups and their details.

Example 6-2 Monitor III Sysplex Summary report

```
                     RMF V1R5   Sysplex Summary - SYSXPLEX        Line 1 of 19
Command ===>                                                    Scroll ===> CSR

WLM Samples: 480     Systems: 3  Date: 11/01/04 Time: 10.14.00  Range: 120 Sec

                      >>>>>>>>XXXXXXXXXXXXXXXXXX<<<<<<<<

Service Definition: WLMDEF                Installed at: 10/30/04, 11.48.01
     Active Policy: WLMPOL                Activated at: 10/30/04, 11.48.10

               ------- Goals versus Actuals --------  Trans --Avg. Resp. Time-
               Exec Vel  --- Response Time ---  Perf  Ended  WAIT EXECUT ACTUAL
Name     T  I  Goal Act  ---Goal--- --Actual--  Indx  Rate   Time   Time   Time

BAT_WKL  W           54                               0.150 1.228  56.48  57.54
BATHI    S  3    31  54                         0.57  0.075 1.296  1.88M  1.90M
BATSUB   S  2    30 9.1                         3.30  0.075 1.161  0.440  1.251
ONL_WKL  W          N/A                               18.75 0.563  1.576  2.139
CICSDEF  S  2       N/A  0.0900 AVG  2.139 AVG  23.8  18.75 0.563  1.576  2.139
STC_WKL  W           96                               0.000 0.000  0.000  0.000
OPSDEF   S  3    40  96                         0.42  0.000

OPSHI    S  1    70 100                         0.70  0.000 0.000  0.000  0.000
```

At the top of the report, you find the performance status line:

```
>>>>>>>>XXXXXXXXXXXXXXXXXX<<<<<<<<
```

As described in 2.4.1, “Using the Monitor III ISPF session” on page 23, you can run Monitor III in GO mode using the **GO** command. The report is refreshed for each time range. This creates a continuous monitoring.

When you run Monitor III in GO mode, you see a performance status line that summarizes the key performance indicators for up to 80 time ranges, showing you the performance history of your sysplex.

```
---XXXX----|||||--|----X--XX-XXXX-XX-XX-
```

It gives a performance indication, by its symbols and colors, for each range when the Monitor III reporter session was in GO mode:

Green (|) All goals have been attained for this range.

Yellow (-) Service class periods with low or medium importance have not attained the goal.

Red (X) Service class periods with high importance have not attained the goal.

The status line is updated with each refresh of the report in GO mode. Assuming a range and refresh value of 120 seconds (or 2 minutes) as in Example 6-2 on page 195, you see the full status line after 160 minutes (for 80 positions in the line) with each interval indicator shifted to the left by one position.

You can use this status line to see at a glance the performance status of your sysplex during the previous minutes or hours.

If green is the dominant color, your sysplex seems to be in good shape, as more yellow or red indicators appear, you might start investigating the sources of possible problems.

Column Perf Indx gives the performance index and shows how well a performance goal could be achieved and is calculated from the goal and actual performance data. A value of 1.0 means that the actual value met the goal exactly, a lesser value indicates that the actual value is better than the goal, a higher value could be seen as an indication of a performance problem.

If the goal could not be attained, the lines for the service class period, service class, and workload are displayed in red (for high importance or 1) and yellow (for medium and low importance or 2 and 3). The same color is also given to the corresponding field in the performance status line.

The red and yellow lines can be indicators of performance problems in the sysplex. If it is a one-time event, you might ignore it. If some lines show red continuously, further investigation is recommended.

In Example 6-2 on page 195, the performance status line contains the characters X as *red* indicators for problems with the performance index. We see for service class CICSDEF:

Importance	2
Response time goal	average response time: 0.0900 sec
Response time actual	average response time: 2.139 sec
Performance index	23.8

A breakdown of the response time is given in the Avg. Resp. Time columns.

These columns provide response time information for all workloads and service classes that deliver transaction information to WLM through WLM APIs. There are three columns, which display the following values:

1. ACTUAL Time
   The average response time for all ended transactions. Note that these response times are for ended transactions only. Thus, if there is a problem where transactions are completely locked out, either while queued or running, you cannot see their influence in the average until they end.
2. EXECUT Time
   For CICS transactions, this includes execution time in AOR and regions invoked by AOR, such as: FOR, DB2, SMSVSAM. For IMS transactions, this includes execution time within the MPR. For Batch, TSO, and so forth, this is the average time that transactions spent in execution. Note: In the Postprocessor Workload Activity report, you see this field as EXECUTION TIME.
3. WAIT Time
   This is time in the queue before execution. It is calculated as the difference between ACTUAL and EXECUT time, as long as ACTUAL time is the bigger value. However, for subsystem data (terminal monitors), it can happen that EXECUT time is larger than the ACTUAL time. This abnormality, for example for CICS, is caused because the ACTUAL time is derived in a local basis and EXECUT time is derived in a sysplex basis.

   For CICS transactions, this includes not only queuing in the TOR and AOR, but also processing time within the TOR. For IMS transactions, this includes not only queuing for the MPR, but also processing time within the CTL region. Otherwise (for batch and APPC), this is the average time that transactions spent waiting on JES or APPC queues. Note that WAIT time may not always be meaningful, depending on how the customer schedules work. For example, if a customer submits jobs in hold status and leaves them until they are ready to be run, all of the held time counts as queued time.

Since the transactions for STC are the address spaces, the Avg. Resp. Time and Trans Ended Rate columns are empty for the STC service classes; the response time data will be available when the address spaces end.

Using cursor-sensitive control for **23.8** (`Perf Indx`) in Example 6-2 on page 195, we navigate to the next report.

6.3.2 Monitor III Response Time Distribution report

You can use this report (Example 6-3) to see how several systems of the sysplex contribute in servicing one specific service class.

Example 6-3 Monitor III Response Time Distribution report

```
                       RMF V1R5   Response Time - SYSXPLEX         Line 1 of 4
Command ===>                                                  Scroll ===> CSR

WLM Samples: 480     Systems: 3  Date: 11/01/04 Time: 10.14.00 Range: 120   Sec

 Class: CICSDEF   %   60|
Period: 1         of    |                                            XXXX
                 TRX    |||||                                        XXXX
Goal:                   |||||                                        XXXX
0.090 sec avg         30|||||                                        XXXX
                        |||||                                        XXXX
                        |||||                                        XXXX
                        |||||                                        XXXX Resp.
                       0|---+//---+---+---+---+---+---+---+---+---+//---- Time
                        <.054 0.063       0.090            0.126  >.135 (sec)

                --Avg. Resp. Time--    Trx     --Subsystem Data--  --Exec Data--
System   Data   WAIT  EXECUT ACTUAL    Rate    Actv  Ready Delay   Ex Vel  Delay
*ALL            0.563  1.576  2.139  18.75        1     2     79
SYS1     all    0.563  1.576  2.139  18.75        1     2     79
SYS2     all
SYS3     all
```

This report enables you to analyze the distribution of a response time to see whether a response time goal was met and, if not, how close it came to failing. The report shows how the response time for a specific service or report class period is distributed. Two levels of detail are shown:

- A character graphic shows the distribution of response time for all systems in a sysplex that have data available in this period. This graphic is shown only for a period with a response time goal.

  It shows the response time (in seconds) with the response time goal in the middle. The middle section of the graph surrounding the goal shows the distribution of transactions that met between 60% and 150% of the goal.

- The detail area (in the report bottom lines) indicates how each system contributed to the overall response time. It shows each system in the sysplex that provides service to the chosen service class. You can use this report to evaluate possible anomalies among the different systems that provide service.

The response time distribution in Example 6-3 on page 197 shows that about 45% of all transactions have a response time of less than 0.054 sec, while all other transactions have a response time of more than 0.135 sec, all together the average response time is 2.139 sec. All transactions of service class CICSDEF are running in system SYS1. The subsystem data is also presented in the “Monitor III Work Manager Delays report” on page 198.

You can navigate from this report to one of the single-system reports. Using cursor-sensitive control, this is the entry point from the sysplex to the following single-system reports:

GROUP	Group Response Time report
SYSINFO	System Information report
STOR	Storage Delays report
DELAY	Delay report
JOB	Job Delays report
WFEX	Workflow/Exceptions report

You can also navigate back to the Sysplex Summary report using F3, and from there call the Work Manager Delays report by using cursor-sensitive control from the service class field.

6.3.3 Monitor III Work Manager Delays report

Here, RMF provides detailed performance data about the transaction running in CICS and IMS subsystems.

The data in this report is generated from performance block (PB) manipulation. PBs are the control blocks used by transaction managers, such as CICS and IMS, to inform WLM about the internal transaction state through a WLM API. A PB is also called a “performance environment.” For each possible internal transaction state (delayed or not) there is information saved in the PB. PBs have the following properties:

- PBs only do reporting. PB reported states are sampled every 250 msecs by WLM, and are later consolidated and then reported to RMF.
- PBs are created and deleted by subsystems per each transaction. However, the same transaction may have several PBs.

PBs occur in two types, one type for each of the following two phases:

- Begin-to-end (BTE) phase, one per transaction for the duration of the transaction. The BTE phase starts when a transaction starts in a TOR, and ends when the TOR receives it back from an AOR. BTE phase duration is roughly the response time minus the TOR queue time.
- Execution (EXE) phase is the time spent either in a CICS AOR or in an IMS MPPR. For example, it is created while entering an AOR and describes the transaction states in AOR; it is deleted when exiting the AOR to return to the TOR.

  The EXEC phase PB has details about the different steps of the transaction out of the TOR. There are two types of EXEC phase:

  – EXEC1, when the transaction is in the execution AS (AOR or IMS/MPR). It is roughly the AOR/MPR transaction service time. It might not be contiguous; it might be interrupted when the database AS is executed. During AOR execution, there are two active and sampled PBs: the BTE (with the delay-for-conversation bit on), and the EXEC1 phase describing the states of the AOR execution.

  – EXEC2, when the execution is done by the supporting database AS, such as DB2, or DL1. It is contiguous in time. During database execution are two active PBs: the BTE with the delay-for-conversation bit on, and the EXEC2 phase describing the status of the database execution.

The Work Manager Delays report also shows some other kinds of internal delays associated with your CICS and IMS transactions (in the begin-to-end or in the execution phase).

Also, you get an overview of how the different CICS address spaces (for example, AOR, TOR, or FOR) provide service to the service classes your transactions belong to. These reports are provided for CICS and IMS classified transactions.

Example 6-4 Monitor III Work Manager Delays report

```
                  RMF V1R5   Work Manager Delays - SYSXPLEX       Line 1 of 3
Command ===>                                                 Scroll ===> CSR

WLM Samples: 480     Systems: 3  Date: 11/01/04 Time: 10.14.00 Range: 120   Sec

Class:  CICSDEF    Period: 1         Avg. Resp. time: 2.139 sec for  2250  TRX.
Goal:   0.090 sec avarage            Avg. Exec. time: 1.576 sec for  1558  TRX.
Actual: 2.139 sec avarage            Abnormally ended:                  0  TRX.

Sub  P  -----------------Response time breakdown (in %)------------ -Switched--
Type     Tot  Act  Rdy Idle -----------------Delayed by------------  Time (%)
                             MISC  I/O CONV DIST SESS TIME PROD LOCK LOC SYS REM

CICS B  79.0  0.5  1.8    0    0  1.3 67.0    0  6.1    0    0  2.3  67   0   0
CICS X  66.4  0.5  0.4    0  0.3 21.6    0    0 40.8    0    0  2.8  22   0   0


------------ Address Spaces Serving this service class CICSDEF  ---------------
Jobname  M ASID System  Serv-Class Service Proc-Usg I/O-Usg  Veloc  Capp  Quies

SYSCCM$1     31  SYS1    OPSDEF      100      .61         4     55     0      0
SYSC1A1A     65  SYS1    OPSDEF      100      3.7         3     74     0      0
SYSC1T1A     64  SYS1    OPSDEF      100      4.2         1     43     0      0
```

Response time breakdown (in % of the `average response time`) provides performance information for the begin-to-end phase (CICS only) and the execution phases of CICS and IMS. This information is the same as the information in the Postprocessor Workload Activity report, except for the difference of the interval length.

- *Act%* - Percentage of the response time transaction that is executing in CPU.
- *Rdy%* - Percentage of the response time being delayed by CPU.
- *Idle%* - User thinking time, for old fashion conversational transactions
- *Delayed by %* - Percentage of the response time being delayed by *internal* CICS reasons.

Some observations on the report:

- 100.0% - 79.0% is the average time in the TOR queue.
- 67.0% (CONV) the transaction was out of TOR (in the AOR)
- 67.0% - 66.4% is the average time in the AOR queue
- The major delay is 40.8% in AOR under the SESS column. It means time spent waiting for a session to be established. Now, you need to ask to your CICS system programmer how to solve this restriction.

`Address Spaces Serving this service class` at the bottom of the report describes which address spaces provide service on different systems in the sysplex. The velocity (`Veloc`) and the processor service (`Proc-Usg`) for each address space can be seen as initial indicators to show how well the address spaces are providing service to the service classes of your transactions. In our case the worst velocity corresponds to the SYSC1T1A address space (43%).

6.3.4 System Information report

The System Information report is a very useful Monitor III report because it allows a broad view of workloads, service classes, and report classes in the sysplex.

Example 6-5 System Information (Sysinfo) report

```
                        RMF V1R5   System Information            Line 1 of 25

 Samples: 100     System: AQTS  Date: 12/22/04  Time: 16.18.20  Range: 100   Sec

 Partition:    AQTS    2064 Model 116         Appl%:      20  Policy: PRIMSHFT
 CPs Online:  11.0     Avg CPU Util%:   22     EAppl%:     20  Date:   09/29/03
 IFAs Online:    -     Avg MVS Util%:   22     Appl% IFA:   -  Time:   15.10.57

 Group     T WFL --Users--  RESP TRANS -AVG USG-   -Average Number Delayed For -
              %   TOT  ACT  Time  /SEC PROC  DEV   PROC  DEV STOR SUBS OPER  ENQ

 *SYSTEM      47  379   15        3.27  2.9  5.6    0.5  1.3  0.0  0.0  1.4  6.2
 *TSO         90  122    1        3.27  0.3  1.2    0.1  0.1  0.0  0.0  0.0  0.0
 *BATCH       31    9    9        0.00  0.9  2.2    0.2  0.8  0.0  0.0  0.0  6.1
 *STC         59  229    4        0.00  0.9  2.2    0.1  0.5  0.0  0.0  1.4  0.2
 *ASCH              0    0        0.00  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 *OMVS        91   19    1        0.00  0.7  0.1    0.1  0.0  0.0  0.0  0.0  0.0
 *ENCLAVE           0  N/A         N/A  0.0  N/A    0.0  N/A  0.0  N/A  N/A  N/A
 PRIMEBAT  W  31    9    9  8.17  0.08  0.9  2.2    0.2  0.8  0.0  0.0  0.0  6.1
 WLMLONG   S  15    6    6  .000  0.00  0.0  1.0    0.0  0.7  0.0  0.0  0.0  5.0
 WLMSHORT  S  62    3    3  8.17  0.08  0.9  1.2    0.2  0.0  0.0  0.0  0.0  1.1
 PRIMETSO  W  90  122    1  .264  3.26  0.3  1.2    0.1  0.1  0.0  0.0  0.0  0.0
 TSOPRIME  S  90  122    1  .264  3.26  0.3  1.2    0.1  0.1  0.0  0.0  0.0  0.0
 PRIMOMVS  W  91   19    1  17.8  0.12  0.7  0.1    0.1  0.0  0.0  0.0  0.0  0.0
 OMVS      S  91   16    1  19.4  0.11  0.7  0.1    0.1  0.0  0.0  0.0  0.0  0.0
 OMVSTASK  S        3    0  .181  0.01  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 SYSTEM    W  54  228    3  .177  0.06  0.7  1.2    0.0  0.2  0.0  0.0  1.4  0.0
```

Let us imagine that the SLA is not happy with the 19.4 secs average response time for the USS transactions running in the service class OMVS. Looking at the report, we may conclude:

- The Workflow% is 91% indicating little delays.
- There are 16 ASs on average, but only one on average has an executing transaction. It means that on average 15 are at user thinking time.
- There is a single transaction with: 0.7 (70%) of its response time in Using Proc (the sampling of CPU service time), 0.1 (10%) of its response time in Using Dev I/O (the sampling of I/O service time), 0.1 (10%) of its response time in Delay Proc (the sampling of CPU wait time). The partials do not add to 1.0 due to rounding. There is the possibility to play with performance goals and importance to reduce 10% of the average transaction time. However, since the transaction seems to be CPU-bound, the easy solution to reduce the CPU service time would be having a faster processor or making this code more efficient.
- Looking at *System, the major delay is ENQ, that is in charge of (6.2 / 15) * 100 of the Monitor III range time. The number of transaction in Using and Delay states does not add to the USERS ACT because Monitor III overlaps Using and Delay counters.

6.3.5 Evaluating virtual storage problems

There are several Monitor III reports available that can be used on a regular basis to determine the cause of running out of SQA or CSA virtual storage, or auxiliary storage slots. In case of a problem, a good approach is to compare actual values with reports taken while your system is running smoothly.

Monitor III provides several useful reports to obtain values on:

- CSA/ECSA occupancy
- SQA/ESQA occupancy
- Storage frames used by address spaces
- Storage not released at the end of a task

The first one is the Common Storage report, obtained with the **STORC** command.

Note the remarks about CSA monitoring in “Ensure common storage tracking” on page 115.

The upper part of the report in Example 6-6 provides information about the common storage CSA and SQA below and above the line. It shows also an overflow of the ESQA in the ECSA. This means that either the ESQA size definition needs to be reviewed or that an address space uses more ESQA than usual or expected. However, there are installations that purposely let the SQA overflows into CSA. Scrolling down to the lower part of this report you can check the amount of virtual storage used by address spaces.

A key observation is that the lack of SQA pages below the line without space in CSA for overflow is not a performance problem, but it is an availability concern since this condition can cause an unplanned outage. You might receive some IRA warning messages, but this might not give you enough reaction time to avoid the unwanted IPL.

Example 6-6 Common Storage

```
                        RMF V1R5   Common Storage                Line 1 of 75
Command ===>                                                 Scroll ===> CSR

Samples: 60      System: SYS1  Date: 10/29/04  Time: 20.55.00  Range: 60    Sec

                                    ---- Percent ----   ------- Amount --------
System Information                  CSA ECSA SQA ESQA     CSA  ECSA   SQA  ESQA
 IPL Definitions                                        3212K   50M 2044K   15M
 Peak Allocation Values               9   77  59  120    300K   39M 1209K   18M
 Average CSA to SQA Conversion        0    9                0 4436K
 Average Use Summary                  9   77  59  120    300K   39M 1209K   18M
 Available at End of Range           91   23  41    9   2912K   12M  835K 1355K

               Service        ELAP  -- Percent Used -   ----- Amount Used -----
Jobname  Act C Class    ASID  Time  CSA ECSA SQA ESQA     CSA  ECSA   SQA  ESQA
%MVS                                  3   22  55   62   90472   11M 1122K 9476K
%REMAIN                               0    0   0    0     560  254K   128  2464
RMF          S SYSSTC     33   2.0D   0    0   0   27       0   576     0 4071K
NET          S SYSSTC     25   2.0D   1   12   0    0   23936 6119K     0  5432
TCPIP        S SYSSTC    130   2.0D   0    9   0    0     136 4596K     0  5548
*MASTER*     S SYSTEM      1   2.0D   2    7   3    8   78216 3357K 62368 1283K
SYSC1T1A     S OPSDEF     64   2.0D   0    0   0    8     136 67264    64 1214K
SYSCCC$1     S OPSDEF     34   2.0D   0    6   0    0    1000 2911K     0   640
RMFGAT       S SYSSTC     61   2.0D   0    0   0    5       0   576    64  727K
```

In the upper part of this report, the heavier user of the virtual storage common area below the line is Master Scheduler.

Another consideration is that the total available virtual storage is not the most important indicator of how healthy your system is. The most important information is the size of the largest free available block of virtual storage, and this information is only available in the Postprocessor Virtual Storage Activity report.

When terminating, some address spaces do not release (FREEMAIN) common storage that they requested earlier (GETMAIN). The Common Storage Remaining report shown in Example 6-7 helps us monitor this situation. We obtain this report using the **STORCR** command.

The purpose of this report is to investigate if this situation is normal or not. It is not definitively bad behavior that an address space **getmains** for common virtual storage is not followed up by **freemain** when the address space ends. So, for each address space listed in this report, an inquiry must be made to the software vendor. If the behavior *is* an error, a correction must be applied in order to release the storage. You should not force the freemain of these storage areas outside of their address space. This action can be dangerous and cause integrity problems.

Example 6-7 Common Storage Remaining

```
                       RMF V1R5   Common Storage Remaining        Line 1 of 12
Command ===>                                                  Scroll ===> CSR

Samples: 60      System: SYS1  Date: 10/29/04  Time: 20.55.00  Range: 60    Sec

                                        Amount of Common Storage
                       Job Ended       Not Released at End of Job
Jobname     ID       Date     Time        CSA  ECSA   SQA  ESQA

%REMAIN                                   560  254K   128  2464
DYNPARS  STC02170  10/27/04  19.30.18       0 68032     0     0
DB2GJ1   JOB02483  10/28/04  08.00.45       0 65536     0     0
DB2GJ1   JOB02453  10/27/04  22.39.29       0 65536     0     0
RRS                10/27/04  19.30.45     272 55920     0   120
CATALOG            10/27/04  19.29.33       0     0   128  2344
EZAZSSI            10/27/04  19.29.37     288     0     0     0
DB2GJ1   JOB02434  10/27/04  21.01.34       0  4288     0     0
IEECMDPF           10/27/04  19.30.03       0   384     0     0
DB2GJ4   JOB02437  10/27/04  21.30.05       0   144     0     0
DB2GJ1   JOB02473  10/28/04  00.51.46       0    48     0     0
IEEGSYS            10/27/04  19.30.03       0    72     0     0
```

6.4 Diagnosing Coupling Facility performance

This section describes the RMF reports that can help you review the performance of the components of your sysplex, do problem diagnosis and make some performance projections.

The Coupling Facility (CF) is the focal point of the information in the sysplex. We are very interested in its performance because many critical system components use it heavily.

You can get reports for the CF from Monitor III, as well as from the Postprocessor.

Monitor III provides three CF-related reports:

CFOVER or CO	CF Overview report
CFACT or CA	CF Activity report
CFSYS or CS	CF Systems report

The Postprocessor offers one report, which contains several sections:

Usage Summary section

Structure Activity section

Subchannel Activity section

CF to CF Activity section

You create the Postprocessor report with all the sections by specifying:

```
SYSRPTS(CF)
```

6.4.1 Coupling Facility response time and utilization

When checking the performance of the Coupling Facility, the first questions are:

- What is the response time for a synchronous request? What is a good value?
- What is the processor (CF CPU) utilization? What is a good value?

The recommendations are:

- Synchronous request service in the CF CPU should be in line with your z/OS CPU speed. This is because during synchronous request processing, the CP in your CPC is spinning and you do not want to waste CP processing cycles. A few tens of microseconds are an acceptable value. The faster your CPC is, the faster your CF should be. Keep the manufacturing level (and the speed) of the CF the same as the manufacturing level (and the speed) of the z/OS CPU. The major differences between synchronous and asynchronous, as the CPU account is concerned, are:
  - Synchronous requests are designed to take less time than asynchronous ones. They are included in the application response time and in the CPU captured time. However, the synchronous request keeps your z/OS CPU busy while processing is done in the CF. Also, remember that the dispatchable unit requesting to the CF keeps in hold the resources (as locks) it is already owning.
  - The asynchronous requests last longer, being included in the application response time only. Also, the dispatchable unit requesting to the CF keeps in hold the resources (as locks) it is already owning.
- CF CPU utilization should be below 50% since high processor utilization increases the queue time of the requests, making the response time of your request almost double. However, if your CF has several CPUs, then a load slightly above 50% could be acceptable. It is not recommended at this point to define many CF CPUs. The below 50% recommendation also applies for availability reasons since your installation must be able to allocate enough resources to support the structure rebuild process of a failing CF over another CF.
- Do not allow a production CF to share CPs or ICFs with other LPs. This creates long service times to synchronous requests, causing severe performance problems.

If utilization and synchronous request time figures are acceptable to you, then you do not have to check the performance of the CF any further.

Now, let us follow an example where we calculate the z/OS CPU (2094 - 303) average utilization with a long service time for synchronous requests:

9000	number of synchronous requests per second
0.060	60 microsecond average service time in milliseconds
3	number of CPs

$$\text{Utilization} = \frac{9000 \times 0.060}{3} = 180\text{ ms/s} = 18\%$$

This means 18% CPU is used on the z/OS LP. This is the cost of having such huge activity against the CF. Also, the CF Sync/Async heuristic algorithm may avoid having the synchronous service time be in the 60 microseconds range.

Monitor III gives us the most important information about CF, easily and in a convenient format for us to use. Let’s have a look at the Coupling Facility Overview report using the **CO** command.

Example 6-8 shows that our CFs’ CPUs are less than 20% loaded, which is very much acceptable.

The column `Processor (Effect)`, when not equal to `Processor (Defined)` indicates that the CF logical CPUs are shared (not recommended for production CFs). It shows how much of the defined logical CPUs is really used. There is also CF request rate and storage information.

Example 6-8 Monitor III Coupling Facility Overview report

```
                   RMF V1R5   CF Overview      - SYSXPLEX         Line 1 of 3
Command ===>                                                 Scroll ===> CSR

Samples: 60      Systems: 3    Date: 10/11/04  Time: 10.10.00  Range: 60    Sec

---- Coupling Facility -----    ---- Processor -----   Request    -- Storage --
Name      Type   Model Level    Util% Defined Effect   Rate       Size    Avail

FACIL03   2084     B16    13    17.0       1     0.0      125.2      959M    871M
FACIL04   2084     B16    13    11.6       1     0.1      509.7      959M    889M
FACIL07   2084     B16    13     0.2       1     1.0                 959M    957M
```

We now take a look at the CF Systems report, with the **CS** command.

Example 6-9 shows that synchronous service time for our system (SYS1) is 47 microseconds at CF FACIL04, and 253 microseconds at the other CF FACIL03. These values are high. We have to investigate our system more closely. In this case, the long service time is not cause by a high processor utilization. Comparing Processor(Defined) with Processor(Effect), we realize that the CF’s logical CPUs are defined to use *shared* processors. This creates long service times. Keep in mind the recommendation to use *dedicated* processors.

Example 6-9 Monitor III Coupling Facility Systems report

```
                   RMF V1R5   CF Systems       - SYSXPLEX         Line 1 of 9
Command ===>                                                 Scroll ===> CSR

Samples: 60      Systems: 3    Date: 10/11/04  Time: 10.10.00  Range: 60    Sec

CF Name   System       Subch    -- Paths --   -- Sync ---    ------- Async -------
                       Delay    Avail Delay    Rate  Avg      Rate  Avg  Chng  Del
                         %             %             Serv           Serv   %     %

FACIL03   SYS1           0.0       4     0.0     7.8   253     69.1   875  0.0  0.0
          SYS2           0.0       4     0.0    <0.1  1169     <0.1  1127  0.0  0.0
          SYS3           0.0       4     0.0     0.0     0     17.6  1245  0.0  0.0
FACIL04   SYS1           0.0       2     0.0   128.5    47     96.3   596  0.0  0.0
          SYS2           0.0       2     0.0    90.8    28     34.3   727  0.0  0.0
          SYS3           0.0       2     0.0   120.2    22     23.3   914  0.0  0.0
FACIL07   SYS1                     2
          SYS2                     2
          SYS3                     2
```

We next calculate CPU utilization in our z/OS CPU with one logical CP used due to synchronous requests to CF FACIL04.

128.5	number of synchronous requests per second
0.047	47 microsecond average service time in milliseconds
1	number of CPUs

$$\text{Utilization} = \frac{128 \times 0.047}{1} = 6 \text{ msec/sec} = 0.6\%$$

We spend less than 1% on synchronous requests in our z/OS logical partition. Is it too much? It is very small in this case.

Just for comparison, let us look at the CPU utilization in our partition (A04) and in our CPC. Using the **CPC** command, we get the CPC Capacity report that provides information about all partitions running on the CPC. The utilization of the partition that we are running in can also be obtained from the System Information report with the command **SI**.

Example 6-10 shows that our partition A04 has just one logical CPU. It is using 0.8% of the CPC; this is very low. Also the CPC logical utilization is low, only 15.2%. So, we do not need to investigate further. We can now continue and have a look at the CF structures.

Example 6-10 Monitor III CPC Capacity Report

```
                        RMF V1R5   CPC Capacity                 Line 1 of 33
Command ===>                                                Scroll ===> CSR

Samples: 60      System: SYS1  Date: 10/11/04  Time: 10.10.00  Range: 60    Sec

Partition:   A04        2084 Model 318
CPC Capacity:     837   Weight % of Max: ****      4h MSU Average:     5
Image Capacity:    93   WLM Capping %:   ****      4h MSU Maximum:     6

Partition  --- MSU ---  Cap  Proc    Logical Util %   - Physical Util % -
            Def   Act  Def   Num    Effect   Total   LPAR  Effect  Total

*CP                                                   7.8     7.4   16.2
A0A           0     3  NO    3.0       2.2     2.4    0.0     0.4    0.4
A0B           0     2  NO    2.0       2.0     2.0    0.0     0.2    0.2
A0C           0     4  NO    2.0       4.5     4.8    0.0     0.5    0.5
A01           0     4  NO    2.0      11.4    12.3    0.0     1.4    1.5
A02           0     4  NO    2.0       3.7     4.0    0.0     0.4    0.4
A03           0     3  NO    2.0       3.3     3.6    0.0     0.4    0.4
A04           0     7  NO    1.0      14.7    15.2    0.0     0.8    0.8
A05           0     5  NO    1.0       9.4     9.9    0.0     0.5    0.6
A06           0     5  NO    1.0       9.6    10.1    0.0     0.5    0.6
A07           0     4  NO    2.0       4.3     4.6    0.0     0.5    0.5
A08           0     4  NO    2.0       4.4     4.7    0.0     0.5    0.5
A09           0     4  NO    2.0       4.5     4.8    0.0     0.5    0.5
A1A           0     0  NO    2.0       0.0     0.0    0.0     0.0    0.0
A1B           0     2  NO    2.0       2.5     2.7    0.0     0.3    0.3
A11           0     4  NO    2.0       4.1     4.4    0.0     0.5    0.5
A12           0     4  NO    2.0       4.5     4.8    0.0     0.5    0.5
A13           0     3  NO    2.0       3.0     3.3    0.0     0.3    0.4
A14           0     1  NO    2.0       1.3     1.3    0.0     0.1    0.1
A17           0     0  NO    2.0       0.4     0.5    0.0     0.0    0.1
A18           0     0  NO    1.0       0.0     0.0    0.0     0.0    0.0
A19          10     0  NO    2.0       0.3     0.4    0.0     0.0    0.0
PHYSICAL                                              7.3            7.3
```

6.4.2 Coupling Facility structures

All data in the CF is stored in structures, so the requests to the CF are directed to structures. There are three types of structures: lock, list, and cache. The internal data formatting within a structure depends on the type of the structure.

- Cache structure has data elements and directory entries (pointing to data elements in the structure and to buffers in local buffers). Two factors may cause bad performance in using a cache structure:
  - Directory reclaim, caused by a full directory entries condition forcing the deletion of data elements, in order to reclaim directory entries to fulfill new incoming data.
  - Directory reclaim cross invalidation (XI), caused by a full directory entries condition forcing the cross invalidation in local buffers, in order to reclaim directory entries to fulfill new incoming data. Cross invalidation means, for example, that DB2 pages located in local buffer pools are no longer valid. In this case the invalidation is caused by the inability of the cache directory to keep track of them, not because of an update of the page contents done in the local buffer pool.
- List structure has data elements and list entries (pointing to queues of data elements).
- Lock structure has lock entries and list entries (pointing to lock entries).

The size relation between the two areas in a structure is decided at structure creation by the first connecting user. Sometimes, we have a performance or an availability problem when one of the areas is exhausted, even if the total structure is not. However, in the majority of cases, the new exploiters are able to start an automatic structure rebuild to solve this kind of problem.

We can obtain information about the CF structures with the Coupling Facility Activity report obtained using the **CA** command.

Example 6-11 shows that our system uses the IBMDG list structure with a reasonable synchronous service time of 23 microseconds. But for the lock structure ISGLOCK (GRS structure), the synchronous service time is 385 microseconds, which is very high. However, the Sync/Async heuristic algorithm is partially relieving the problem because 11.9 requests per second are asynchronous in this structure. Refer to “Coupling Facility” on page 191. GRS requests are always requested as synchronous; however, in this case, the vast majority of them—11.9 reqs/sec are asynchronous in a total of (11.9 + 1.3) reqs/sec—are converted by the algorithm to asynchronous. This conversion does not make them faster, but just releases the z/OS CPU from the loop.

Example 6-11 Monitor III CF Activity Report

```
                    RMF V1R5   CF Activity      - SYSXPLEX     Line 41 of 104
Command ===>                                                  Scroll ===> CSR

Samples: 60      Systems: 3    Date: 10/11/04  Time: 10.10.00  Range: 60    Sec

CF: ALL          Type  ST System      --- Sync ---     --------- Async --------
                                       Rate   Avg       Rate   Avg   Chng   Del
Structure Name                                Serv             Serv   %      %

DFHCFLS_SYSCFDT1 LIST     *ALL          7.8    252      42.3    855   0.0   0.0
                          SYS1          7.8    252      42.3    855   0.0   0.0
                          SYS2          0.0      0       0.0      0   0.0   0.0
                          SYS3          0.0      0       0.0      0   0.0   0.0
IBMBDG           LIST     *ALL        302.7     23      10.4    377   0.0   0.0
                          SYS1         92.8     23       1.5    401   0.0   0.0
                          SYS2         89.7     24       4.4    380   0.0   0.0
```

```
                                SYS3        120.2      21        4.5     367   0.0    0.0
ISGLOCK          LOCK           *ALL          1.7     371       16.5     858   0.0    0.0
                                SYS1          1.3     385       11.9     842   0.0    0.0
                                SYS2          0.3     147        2.5     946   0.0    0.0
                                SYS3         <0.1    3289        2.1     844   0.0    0.0
```

Usually, a well performing lock CF structure should have a high majority of its lock requests as synchronous. If they turned into asynchronous, it is because the SYNC service times are too long and the request types are being changed to ASYNCs. You can see this happening in Example 6-11, where the majority of the requests are ASYNC and not SYNC to the ISGLOCK structure.

Besides utilization and response time, you may have other performance problems with your CF structures. We can consider them by looking into the Postprocessor Coupling Facility Activity report, with the control statement: `SYSRPTS(CF)`.

Example 6-12 shows part of the CF usage summary section of the CF activity report. You could check the utilization of each structure space wise. No problems, with the exception of the number of data elements for the cache structure:

- IXC_LST6 has 4210 list entries; 2 of them are used.
- IXC_LST6 has 4242 data elements; 32 of them are used.
- DB2P_LOCK1 has 290K list entries; 5964 of them are used.
- DB2P_LOCK1 has 17M lock entries; 86K of them are used.
- DB2P_GBP1 has 1299K directory entries; 320K of them are used.
- DB2P_GBP1 has 144K data elements; 144K of them are used. The data element area is completely full. Does this structure deserve more memory to improve the overall performance or it is worthless? In order to answer this question, we need to develop a metric indicating the yield of the cache structure. This metric is based on the numbers presented and discussed in the Coupling Facility structure activity section of the CF report (Example 6-12) and is related with the Data Access fields: Reads, Writes, Castouts, and XIs. Another important metric is to relate the size of the cache area containing the data elements with the number of DASD I/O reads executed per DB2. This curve indicates the right size of such area.
- DB2P_GBP1 has no directory reclaims, as it should be. There are enough directory entries.

Example 6-12 Postprocessor CF activity report, Usage Summary section

```
                                           C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                                                              PAGE   1
          z/OS V1R4           SYSPLEX PLEXABC            DATE 11/18/2004          INTERVAL 015.00.000
                              RPT VERSION V1R2 RMF       TIME 10.00.00            CYCLE 01.000 SECONDS

    -----------------------------------------------------------------------------------------------------------------
     COUPLING FACILITY NAME = CFAA
     TOTAL SAMPLES(AVG) =   899  (MAX) =   899  (MIN) =   899
    -----------------------------------------------------------------------------------------------------------------
                                                  COUPLING  FACILITY  USAGE  SUMMARY
    -----------------------------------------------------------------------------------------------------------------
     STRUCTURE SUMMARY
    -----------------------------------------------------------------------------------------------------------------
.........
                                                    % OF              % OF     AVG    LST/DIR  DATA      LOCK     DIR REC/
            STRUCTURE                      ALLOC    CF            #   ALL      REQ/   ENTRIES  ELEMENTS  ENTRIES  DIR REC
     TYPE   NAME             STATUS CHG    SIZE     STORAGE     REQ   REQ      SEC    TOT/CUR  TOT/CUR   TOT/CUR  XI'S
........
     LIST   IXC_LST6         ACTIVE         22M         0.2   2907K   20.4   3230.3     4210      4242       N/A       N/A
                                                                                           2        32       N/A       N/A
........
     LOCK   DB2P_LOCK1       ACTIVE        131M         1.4   5706K   40.1   6340.0     290K         0       17M       N/A
                                                                                        5964         0       86K       N/A
```

```
........
    CACHE  DB2P_GBP1         ACTIVE          1G         10.9   632906      4.4     703.23    1299K       144K      N/A         0
                                                                                              320K       144K      N/A         0
.......
```

Now we take a look at another section of the report.

Example 6-13 reports the sysplex Coupling Facility activity, where you can see the following:

- Structures should have acceptable service times and no delayed requests. Structure IXC_LST6 has 61 delayed requests due to subchannel busy (a CF link like ICB-3, ISC-3 and IC-3 defines 7 subchannels each). This type of delay is due to heavy activity. The other reason for a request delay is when the CF link (path) itself is busy because it is shared with another LP. On the other hand, this is only 61 out of 2907K requests. Each one of these 61 requests was delayed for subchannel for 73.3 microsecs. However, if we derive the average queue time for all the requests (2907K), in /ALL field, the number coming out is 0.0. The fields PR WT and PR CMP have to do with duplexing delays.
- DB2P_LOCK1 has no delayed requests due to subchannel busy, indicating that there is not a CF link contention. The average synchronous request time is an acceptable 16.2 microseconds. However, due to DB2 serialization, 377K times DB2 lock requests are initially denied because the lock entry is busy (lock contention). That is, the lock is owned by some other user. This is about 5% (ROT is less than 2-5%) of the 7131K requests. When this number is high (discarding the false contention figures), the application specialist and the DBA need to be informed because the contention might be alleviated by application modifications.

  This number includes false contention (seen next), that is, 107K delayed requests. False contention is caused by having more allowed locks than the number of lock entries available in the lock structure (lock table). It occurs when two different locks point to the same entry. It is solved by IRLMs conversation across XCF paths, causing traffic in XCF paths and delays in the transaction execution. Our number (107K) corresponds to 2% of the total requests, 7131K. If the values are too high, we should increase the structure size or decrease the size of each entry by decreasing the maximum number of systems in the sysplex, as declared when formatting the Sysplex Couple data set. Generally speaking, this value should be less than 1-1.5%.
- Next we analyze DB2_GBP1 Data Access fields. Just comparing Writes with Castouts (164064: 81708), we can see that almost 50% of the writes do not cause a DASD I/O operation, meaning that the cache saved real DASD I/O operations. Along the Castout, DB2 pages are written during the 900 second RMF period (91 page/s) from this structure to DASD subsystem. The number of Reads must be compared with the number of DASD I/O reads in order to establish a cache hit ratio metric, associated with the cache size performance.
- DB2_GBP1 has number 12407 in the cross invalidation (XI) field. Any time there are XIs, you should check first any XI directory reclaims and if so, correct that by at least increasing the size of the structure. Another way to decrease XIs is by decreasing the size of the DB2 local buffer pools.

Example 6-13 Postprocessor CF activity report, Structure activity section

```
                                            C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                                                                PAGE  20
         z/OS V1R4              SYSPLEX PLEXABC             DATE 11/18/2004           INTERVAL 015.00.000
                                RPT VERSION V1R2 RMF        TIME 10.00.00             CYCLE 01.000 SECONDS

    ------------------------------------------------------------------------------------------------------------------------------
     COUPLING FACILITY NAME = CFAA
    ------------------------------------------------------------------------------------------------------------------------------
                                                 COUPLING  FACILITY  STRUCTURE  ACTIVITY
    ------------------------------------------------------------------------------------------------------------------------------
.........
     STRUCTURE NAME = IXC_LST6          TYPE = LIST   STATUS = ACTIVE
               # REQ     -------------- REQUESTS -------------   -------------- DELAYED REQUESTS -------------
     SYSTEM    TOTAL            #     % OF  -SERV TIME(MIC)-     REASON    #    % OF  ---- AVG TIME(MIC) -----
     NAME      AVG/SEC          REQ    ALL     AVG   STD_DEV              REQ    REQ   /DEL    STD_DEV   /ALL
........
     TOTAL      2907K  SYNC      0     0.0     0.0      0.0      NO SCH   61     0.0   73.3      101.1    0.0
                3230   ASYNC  2907K    100    89.8    529.5      PR WT     0     0.0    0.0        0.0    0.0
                       CHNGD     0     0.0                       PR CMP    0     0.0    0.0        0.0    0.0
                                                                 DUMP      0     0.0    0.0        0.0    0.0
........
     STRUCTURE NAME = DB2P_LOCK1        TYPE = LOCK   STATUS = ACTIVE
               # REQ     -------------- REQUESTS -------------   -------------- DELAYED REQUESTS -------------
     SYSTEM    TOTAL            #     % OF  -SERV TIME(MIC)-     REASON    #    % OF  ---- AVG TIME(MIC) -----  EXTERNAL REQUEST
     NAME      AVG/SEC          REQ    ALL     AVG   STD_DEV              REQ    REQ   /DEL    STD_DEV   /ALL   CONTENTIONS
........
     TOTAL      5706K  SYNC   5503K   96.4    16.2     10.2      NO SCH    0     0.0    0.0        0.0    0.0   REQ TOTAL     7131K
                6340   ASYNC   203K    3.6   102.6    131.3      PR WT     0     0.0    0.0        0.0    0.0   REQ DEFERRED   377K
                       CHNGD     0     0.0                       PR CMP    0     0.0    0.0        0.0    0.0   -CONT          377K
                                                                                                              -FALSE CONT    107K
........
     STRUCTURE NAME = DB2P_GBP1         TYPE = CACHE  STATUS = ACTIVE
               # REQ     -------------- REQUESTS -------------   -------------- DELAYED REQUESTS -------------
     SYSTEM    TOTAL            #     % OF  -SERV TIME(MIC)-     REASON    #    % OF  ---- AVG TIME(MIC) -----
     NAME      AVG/SEC          REQ    ALL     AVG   STD_DEV              REQ    REQ   /DEL    STD_DEV   /ALL
........
     TOTAL       633K  SYNC    442K   69.8    21.9     14.0      NO SCH    0     0.0    0.0        0.0    0.0    -- DATA ACCESS ---
               703.2   ASYNC   191K   30.2   100.3     78.7      PR WT     0     0.0    0.0        0.0    0.0    READS        52010
                       CHNGD     0     0.0                       PR CMP    0     0.0    0.0        0.0    0.0    WRITES      164064
                                                                 DUMP      0     0.0    0.0        0.0    0.0    CASTOUTS     81708
                                                                                                               XI'S         12407
```

Now, let us go through a more detailed study of the CF links performance.

6.4.3 Coupling facility links

Coupling facility links are fast paths between z/OS LPs and CF LPs. In the z/Series machines CF links can also be used between CFs' LPs for structure duplexing purposes. We have CF links performance information in the subchannel activity section of the Postprocessor Coupling Facility Activity report.

Example 6-14 shows that we have 69537 requests (115.9 per sec) from our system SYS1 to the CF (FACIL04).

Example 6-14 Postprocessor CF activity report, Subchannel activity section

```
                                                                                                            PAGE  35
     z/OS V1R5               SYSPLEX SYSXPLEX          DATE 10/11/2004          INTERVAL 010.00.000
                        CONVERTED TO z/OS V1R5 RMF     TIME 10.00.00            CYCLE 01.000 SECONDS

-------------------------------------------------------------------------------------------------------------------------------
 COUPLING FACILITY NAME = FACIL04
-------------------------------------------------------------------------------------------------------------------------------
                                                      SUBCHANNEL  ACTIVITY
-------------------------------------------------------------------------------------------------------------------------------
          # REQ                                   ----------- REQUESTS -----------   ------------------ DELAYED REQUESTS --------------
 SYSTEM   TOTAL   -- CF LINKS --  PTH                 #   -SERVICE TIME(MIC)-                #     % OF  ------ AVG TIME(MIC) ------
 NAME     AVG/SEC TYPE  GEN  USE  BUSY               REQ     AVG    STD_DEV                REQ    REQ    /DEL     STD_DEV     /ALL

 SYS1      69537  ICP      2    2     0  SYNC      18770   109.3     403.7   LIST/CACHE     0    0.0    0.0         0.0       0.0
           115.9  SUBCH   14   14        ASYNC     45098   561.2      1432   LOCK           0    0.0    0.0         0.0       0.0
                                         CHANGED       0  INCLUDED IN ASYNC  TOTAL          0    0.0
                                         UNSUCC        0      0.0      0.0
 SYS2       24378  ICP     2    2     0  SYNC       5282    47.6     276.9  LIST/CACHE    0   0.0    0.0         0.0       0.0
             40.6  SUBCH  14   14        ASYNC     14653   954.5      1216  LOCK          0   0.0    0.0         0.0       0.0
                                         CHANGED       0  INCLUDED IN ASYNC  TOTAL        0   0.0
UNSUCC        0     0.0       0.0
 SYS3       21995  ICP     2    2     0  SYNC       6959    28.0     162.1  LIST/CACHE    0   0.0    0.0         0.0       0.0
             36.7  SUBCH  14   14        ASYNC     10494  1120.5      1331  LOCK          0   0.0    0.0         0.0       0.0
                                         CHANGED       0  INCLUDED IN ASYNC  TOTAL        0   0.0
                                         UNSUCC        0     0.0       0.0
```

We are using 2 IC links with 14 subchannels. A subchannel is like a device accessed by the CF link associated with a service buffer where the request is stored to be processed. Do we have enough subchannels? We will check that through the calculation of the number of average active requests to the CF as follows:

$$\text{Total Service Time} = \frac{18770 \times 109.3 + 45098 \times 561.2}{1000} = 27360\text{ ms} = 27.36\text{ sec}$$

18770	Number of synchronous requests
109.3	Synchronous request service time in microseconds
45098	Number of asynchronous requests
561.2	Asynchronous request service time in microseconds

This is our total service time in a 10 minute interval (600 sec). That leads to a utilization and a number of active subchannels:

$$\text{Number of active subchannels} = \frac{27.36}{600} = 0.045$$

So 14 subchannels are more than enough. We have activity for 0.045 subchannels. Field PTH BUSY shows the number of times that our request has been delayed because the path (CF link) is shared with other LPs and busy because of activities in other LPs. As you can see, we are not been delayed in this sample.

We look at the number of delayed requests and they are zeros, which is obviously good. We have enough links and subchannels.

6.4.4 Links between Coupling Facilities

One point to check is the traffic and service times between Coupling Facilities, for duplexing purposes. This information is in the CF to CF activity section.

Example 6-15 Postprocessor CF to CF Activity section

```
                                                                                                                    PAGE  36
     z/OS V1R5              SYSPLEX SYSXPLEX           DATE 10/11/2004          INTERVAL 010.00.000
                       CONVERTED TO z/OS V1R5 RMF      TIME 10.00.00            CYCLE 01.000 SECONDS

------------------------------------------------------------------------------------------------------------------------
 COUPLING FACILITY NAME = FACIL04
------------------------------------------------------------------------------------------------------------------------
                                                   CF TO CF ACTIVITY
------------------------------------------------------------------------------------------------------------------------
           # REQ                          ----------- REQUESTS -----------   -------------- DELAYED REQUESTS --------------
 PEER      TOTAL      -- CF LINKS --               #   -SERVICE TIME(MIC)-         #     % OF  ------ AVG TIME(MIC) ------
 CF        AVG/SEC    TYPE       USE              REQ     AVG    STD_DEV          REQ    REQ   /DEL    STD_DEV     /ALL

 FACIL03   10756      ICP          2      SYNC   10756     0.4       0.0     SYNC     0    0.0    0.0        0.0       0.0
            17.9
```

Example 6-15 shows the activity from FACIL04 to FACIL03. There are no link problems since the number of delayed requests is zero. The other CF is responding fast since the service time is only 0.4 microsecond. There are only 17.9 requests per second. There are two CF-links of type ICP, which is a CF memory bus.

6.4.5 XCF signalling resources

XCF is a z/OS component in charge of passing messages between software members in a Sysplex environment. XCF is then used for signalling, heartbeating (verify if the member is alive, informing other members if it is not) and group support (a group is a set of members of the same nature).

Signalling paths is the set of outbound paths (CTC or list structure) and outbound message buffers in one XCF. A transport class (TC) defines a unique set of signaling paths for messages belonging to this TC.

If a message is bigger than the buffer assigned to its transport class's buffer pool, the message needs to be broken, causing some XCF CPU overhead. However, if this resizing happens several times, XCF reformats the buffer pool to avoid such overhead. You may also define additional TC and segregate messages based in message size using GROUP(UNDESIG). XCF transport classes are defined in the COUPLE parmlib member. As shown in Example 6-16, messages coming from members of the DFHIR000 group are preferentially sent through the CF list structure IXC_CICS. Also, messages from other groups use buffers in two different buffer pools based in their size.

Example 6-16 Sample COUPLE member

```
CLASSDEF CLASS(CICS) GROUP(DFHIR000)
PATHOUT STRNAME(IXC_CICS) CLASS(CICS)
CLASSDEF CLASS(DEFAULT) CLASSLEN(956) GROUP(UNDESIG)
CLASSDEF CLASS(DEF18K) CLASSLEN(16316) GROUP(UNDESIG)
```

For more information about XCF, refer to the WSC Flash in *Setting Up a Sysplex*, SA22-7625, Chapter 5, where you can find a detailed discussion about XCF traffic.

For an evaluation of your XCF environment, you can download and run the z/OS Sysplex Health Checker program at:

http://www.ibm.com/servers/eserver/zseries/zos/downloads/

We first look at our system XCF traffic using the Monitor III **XCF** command.

Example 6-17 Monitor III XCF Delays report

```
                         RMF V1R5   XCF Delays
Command ===>                                                  Scroll ===> CSR
Samples: 60      System: SYS1  Date: 10/12/04  Time: 14.35.00  Range: 60    Sec
             Service    DLY     ------------ Main Delay Path(s) -----------
Jobname   C  Class       %        %  Path    %  Path    %  Path    %  Path
WLM       S  SYSTEM       2       2  -CF-
```

XCF delay is one of the delays tracked by Monitor III. In the Delay Activity report, this delay is presented together with JES and HSM delay under the keyword Subsystems.

Example 6-17 shows that WLM address space is delayed by 2% due to XCF signalling caused by delays accessing the list structure in the CF.

Next, we produce the Postprocessor XCF Activity and CF Activity reports from another system which has more activity, using the following control statements:

```
REPORTS(XCF)
SYSRPTS(CF)
```

Let's start with the XCF Activity report and the XCF Usage by System section shown in Example 6-18.

Example 6-18 Postprocessor XCF Activity report, Usage by System section

```
                                                     X C F   A C T I V I T Y

            z/OS V1R4                SYSTEM ID XYZS             DATE 11/17/2004            INTERVAL 15.00.000
                                     RPT VERSION V1R2 RMF       TIME 10.00.00              CYCLE 1.000 SECONDS

                                                        XCF USAGE BY SYSTEM
---------------------------------------------------------------------------------------------------------     -----------------
                          OUTBOUND FROM XYZS                                                  INBOUND TO XYZS            XYZS
---------------------------------------------------------------------------   ---------------------------     -----------------
                                        ----- BUFFER -----      ALL
TO        TRANSPORT  BUFFER        REQ    %    %    %    %    PATHS     REQ   FROM            REQ     REQ     TRANSPORT     REQ
SYSTEM    CLASS      LENGTH        OUT  SML  FIT  BIG  OVR  UNAVAIL  REJECT   SYSTEM           IN  REJECT     CLASS      REJECT

XYZA      DEFAULT    28,604      3,982  100    0    0    0        0       0   XYZA        274,764       0     DEFAULT         0
          DEFMED      8,124     88,921    1   99    0    0        0       0                                   DEFMED          0
          DEFSMALL      956    167,093    0  100    0    0        0       0                                   DEFSMALL        0
          CLS20      20,412          2  100    0    0    0        2       0                                   CLS20           0

XYZR      DEFAULT    28,604        283   92    7   <1  100        0       0   XYZR         82,862       0
          DEFMED      8,124      6,867   16   84    0    0        0       0
          DEFSMALL      956     80,532    0  100    0    0        0       0
          CLS20      20,412        111  100    0    0    0      111       0
```

Here is an explanation of some of the fields in the RMF XCF Activity reports, directly affecting the XCF performance:

- RMF ALL PATHS UNAVAILABLE: Number of messages migrated to other TC because no operational signaling path, usually normal in a TC without a pathout. The TC without defined pathouts has its messages migrated to the default TC using its pathouts.
- REQ REJECTED: Number of requests for a message buffer that could not be satisfied due to constraints on the amount of message space buffer, that is, for example, when the path out buffer pool is full.
- BUSY: Number of times XCF selected a signalling path, while a message was already in the process of being transferred.
- RETRY: Number of times XCF initialized the signalling path.
- AVAIL: Number of times the signalling path was selected and it was immediately available to transfer a message.

Example 6-18 describes the traffic of system XYZS to the other systems. In this example, only two systems are shown and here is what we can see from the report:

- To system XYZA using transport class DEFMED, we have 88,921 messages (for a 15 min period). Of that number, 99% were suitable in the buffer length (8124 bytes) and 1% were small. This just wastes storage; there is no CPU overhead involved.
- To system XYZR using transport class CLS20, we have 111 messages. Since 100% were smaller than 20412 bytes, it is suitable for a smaller class. For 111 requests we have no path available, so traffic goes via some other path, which means some little extra processing. To avoid this, this transport class should be removed.
- To system XYZR using transport class DEFAULT, we have 283 messages. Less than 1% are big and for all of them (100%), XCF has the overhead of breaking the message in pieces. XCF did not reformat the buffer pool yet.

- No messages are rejected, indicating that our buffer areas have enough size for our need. The MAXMSG definitions are large enough.

Now, let’s have a look at the XCF activity report, XCF Path Statistics section.

Example 6-19 describes the traffic of the system XYZS to the other systems, per path. This report shows us:

- To system XYZA using CF structure IXC_LSTB and transport class DEFMED:
  - We have 22,834 requests.
  - We have no queueing, queue length is 0.00 requests, and structure busy shows 0 requests.
  - Path IXC_LSTB has been always available, for 22,834 requests.
  - We have no hardware problems, retry shows 0 requests.
- To system XYZA using CF structure IXC_LST6 and transport class DEFSMALL. Transport class DEFSMALL is using the structures IXC_LST1, -2, -5 and -6.
  - We have 58,014 requests.
  - Queuing is minor; queue length is 0.01. With queue length of higher than 0.5 we could have problems. Possible problem creators could be:
    - Number of paths (links) and subchannels to CF is insufficient.
    - CF processors are overloaded.
    - Not enough structures (IXC_LST1 -2 -5 -6) for this class.

Example 6-19 Postprocessor XCF Activity report, Path Statistics section

```
------------------------------------------------------------------------------------------------------------------------------------
TOTAL SAMPLES = 899                                     XCF PATH STATISTICS
-------------------------------------------------------------------------------------------------------------------------------------
                             OUTBOUND FROM XYZS                                                    INBOUND TO XYZS
----------------------------------------------------------------------------------        -------------------------------------------
         T FROM/TO                                                                                 T FROM/TO
 TO      Y DEVICE, OR      TRANSPORT        REQ    AVG Q                                 FROM     Y DEVICE, OR           REQ BUFFERS
 SYSTEM  P STRUCTURE       CLASS            OUT    LNGTH     AVAIL   BUSY   RETRY        SYSTEM   P STRUCTURE             IN UNAVAIL

 XYZA    S IXC_LSTA        DEFAULT           524    0.00        524      0       0        XYZA     S IXC_LSTA           4,894       0
         S IXC_LSTB        DEFMED         22,834    0.00     22,834      0       0                 S IXC_LSTB          33,747       0
         S IXC_LSTC        DEFMED         29,317    0.01     29,317      0       0                 S IXC_LSTC          27,660       0
         S IXC_LST1        DEFSMALL       44,660    0.01     44,660      0       0                 S IXC_LST1          14,207       0
         S IXC_LST2        DEFSMALL       51,667    0.00     51,667      0       0                 S IXC_LST2          70,188       0
         S IXC_LST3        DEFMED         23,515    0.01     23,515      0       0                 S IXC_LST3          18,819       0
         S IXC_LST4        DEFAULT           713    0.00        713      0       0                 S IXC_LST4           8,618       0
         S IXC_LST5        DEFSMALL       12,935    0.00     12,935      0       0                 S IXC_LST5          31,739       0
         S IXC_LST6        DEFSMALL       58,014    0.01     58,014      0       0                 S IXC_LST6          16,944       0
         S IXC_LST7        DEFMED         13,257    0.00     13,257      0       0                 S IXC_LST7          39,923       0
         S IXC_LST8        DEFAULT         1,425    0.00      1,425      0       0                 S IXC_LST8           8,635       0
         S IXC_LST9        DEFAULT         1,320    0.00      1,320      0       0                 S IXC_LST9           8,300       0

 XYZR    S IXC_LSTA        DEFAULT           143    0.00        143      0       0        XYZR     S IXC_LSTA           2,918       0
         S IXC_LSTB        DEFMED            719    0.00        719      0       0                 S IXC_LSTB           2,223       0
         S IXC_LSTC        DEFMED          2,248    0.00      2,248      0       0                 S IXC_LSTC           2,096       0
         S IXC_LST1        DEFSMALL       22,846    0.00     22,846      0       0                 S IXC_LST1           2,250       0
         S IXC_LST2        DEFSMALL        8,071    0.00      8,071      0       0                 S IXC_LST2          25,543       0
         S IXC_LST3        DEFMED          1,889    0.00      1,889      0       0                 S IXC_LST3           1,567       0
         S IXC_LST4        DEFAULT            44    0.00         44      0       0                 S IXC_LST4           4,223       0
         S IXC_LST5        DEFSMALL       23,088    0.00     23,088      0       0                 S IXC_LST5          13,424       0
         S IXC_LST6        DEFSMALL       26,820    0.00     26,820      0       0                 S IXC_LST6          24,638       0
         S IXC_LST7        DEFMED          2,012    0.00      2,012      0       0                 S IXC_LST7           1,984       0
         S IXC_LST8        DEFAULT            54    0.00         54      0       0                 S IXC_LST8           3,680       0
         S IXC_LST9        DEFAULT            42    0.00         42      0       0                 S IXC_LST9           2,756       0
```

Let’s look at the CF and the structures, and especially one of the structures, IXC_LST6 for class DEFSMALL in the CF Activity report, to see how busy it is. There, we can find the information about the list structures IXC_LST6, -1, -2, and -5 in the Structure Activity section of the CF Activity report.

With the numbers given in Example 6-20, we do a small calculation to see how busy the busiest structure IXC_LST6 is:

```
IXC_LST6 busy = 3557 requests/s * 94.9 microseconds/request = 337 ms/s = 33.7%
```

The utilization of the other structures is less than 30%. Our structure is like a server in the multiserver pool for the transport class DFSMALL, when it is receiving a message. But we are not willing to load it too much; perhaps 70% is enough, 100% is too much. We should consider adding more list structures if the load increases significantly, let us say, when it is doubled.

Example 6-20 Postprocessor CF activity report, Structure activity section

```
                                         C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                                                                        PAGE   9
         z/OS V1R4               SYSPLEX PLEXABC             DATE 11/17/2004           INTERVAL 015.00.000
                                 RPT VERSION V1R2 RMF        TIME 10.00.00             CYCLE 01.000 SECONDS

    ------------------------------------------------------------------------------------------------------------------------------
     COUPLING FACILITY NAME = CFAA
    ------------------------------------------------------------------------------------------------------------------------------
                                               COUPLING  FACILITY  STRUCTURE  ACTIVITY
    ------------------------------------------------------------------------------------------------------------------------------

     STRUCTURE NAME = IXC_LST6          TYPE = LIST   STATUS = ACTIVE
   -----
               # REQ    -------------- REQUESTS -------------   -------------- DELAYED REQUESTS -------------
     SYSTEM    TOTAL             #     % OF  -SERV TIME(MIC)-    REASON    #   % OF  ---- AVG TIME(MIC) -----
     NAME      AVG/SEC           REQ    ALL     AVG   STD_DEV            REQ    REQ   /DEL    STD_DEV   /ALL

                3201K   SYNC      0     0.0     0.0      0.0     NO SCH    0    0.0    0.0        0.0    0.0
                3557    ASYNC  3201K    100    94.9    244.9     PR WT     0    0.0    0.0        0.0    0.0
                        CHNGD     0     0.0  INCLUDED IN ASYNC   PR CMP    0    0.0    0.0        0.0    0.0
                                                                 DUMP      0    0.0    0.0        0.0    0.0
  -----
STRUCTURE NAME = IXC_LST1          TYPE = LIST   STATUS = ACTIVE
                2663    ASYNC  2396K    100   100.5    185.0     PR WT     0    0.0    0.0        0.0    0.0
  -----
     STRUCTURE NAME = IXC_LST2          TYPE = LIST   STATUS = ACTIVE
                3396    ASYNC  3056K    100    92.9    238.7     PR WT     0    0.0    0.0        0.0    0.0
  -----
     STRUCTURE NAME = IXC_LST5          TYPE = LIST   STATUS = ACTIVE
                2105    ASYNC  1895K    100   101.1    174.3     PR WT     0    0.0    0.0        0.0    0.0
  -----
```

6.4.6 Coupling Facility study with the Spreadsheet Reporter

We investigate a time frame when DB2 database activity starts. Databases have mainly read activity, so the group buffer pools (GBP) should only be slightly used; by a default DB2 option, only the written pages are sent to the GBP CF structure. Spreadsheet Reporter is very suitable for this kind of analysis. Using the Coupling Facility Trend Report macro, we display two of its reports.

The chart shown in Figure 6-11 on page 215 shows how DB2 begins to use CF structure. In this case only the updated pages are taken to the GBP structure. The directory elements are used both for the GBP pages in the CF and the virtual pages in the local buffer pools. As soon as the applications start activity, the CF at 16.40 has most of the directory elements marked in use (98% of the elements). The directory elements are pointing to pages in local buffer pools. The applications use relatively few data elements in the structure due to the low update activity. Using the directory elements in a CF structure may cause the appearance of forced reclaims. Obviously, if the pages in local buffers are heavily in use, as a consequence the directory elements in the CF are also heavily used.

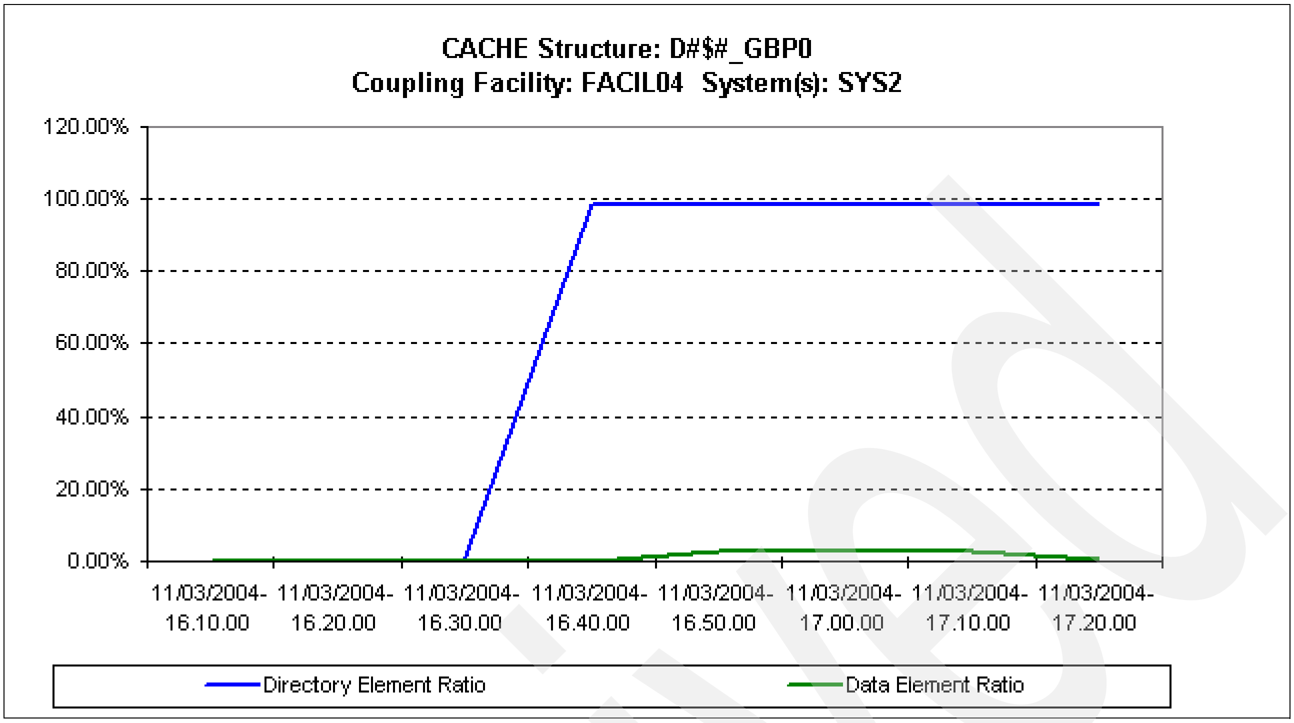


Figure 6-11 Spreadsheet Reporter – CF cache structure, directory/data elements

Reclaim happens when we are running out of directory entries in the cache (even when we have enough data entries). Then the contents of entry LRU is replaced by a new contents, pointing to other page. The replaced page vanishes from where it is. Figure 6-12 on page 216 shows how reclaiming begins at the 16.30 period. We have 10 minute periods where we have about 700 reclaims per second, and for sure this is an indication of a cache performance problem.

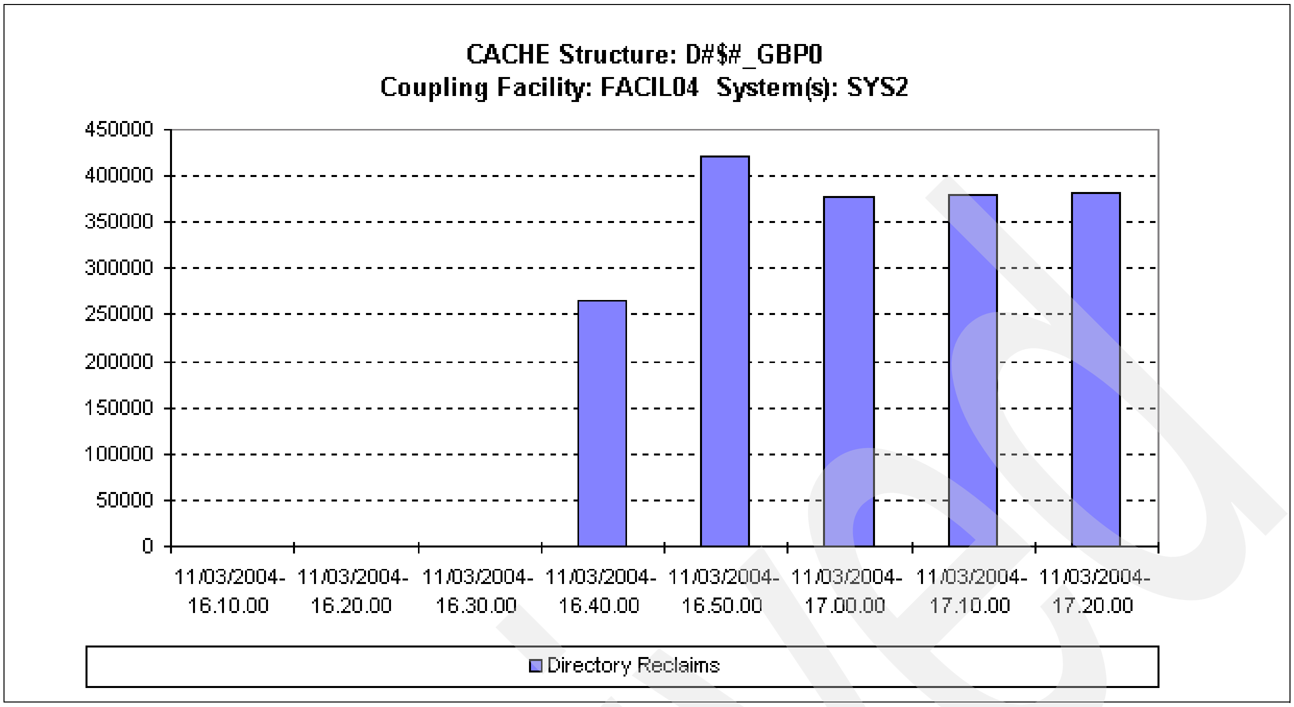


Figure 6-12 Spreadsheet Reporter – CF cache structure, directory reclaims

What should we do if we do not have enough directory elements in the structure for our GBP? We have two possibilities, either we increase the size of the GBP, or we hope that the directory/data size ratio can be correct by the exploiters through a CF structure rebuild.

7

Monitoring batch workloads using RMF

This chapter describes the RMF functions that can help you investigate a perceived batch performance problem.

It describes the following:

- Investigating CPU usage and delays
- Investigating I/O response times, utilization, and delays
- Understanding batch window considerations
- Exploring the technologies for improving batch performance

7.1 Introduction

What is a batch performance problem?

- The unusual or unacceptable duration (elapsed time) of a job step
- The high resource consumption of a program
- The total duration of the jobs does not fit in the batch window
- The impact of running batch concurrently with the online workload

First case study

We are going to analyze a basic batch performance problem (first case) to demonstrate the use of RMF to accomplish this analysis. Many of the commonly used reports or panels (Postprocessor, Monitor II, or III) are shown and discussed.

We consider this example: Users have complained that, since 09:10 a.m. their jobs running in the service class (BATMED) were taking a very long time to complete.

7.1.1 Summary report as a good starting point

A good point to start investigation of a batch performance problem is the Postprocessor Summary report. It gives an overview (one line per interval) of the system activity and resources used (CPU, DASD, and so on). The Postprocessor Summary report (a local report) gives the profile of a batch window on a single page.

Control statement: `SUMMARY(INT)`

Example 7-1 Postprocessor Summary report

```
                                                  R M F   S U M M A R Y   R E P O R T
                                                                                                                      PAGE 001
          z/OS  V1R5                   SYSTEM ID SYS1             START 10/18/2004-08.40.00 INTERVAL 00.09.03
                                       RPT VERSION V1R5 RMF       END   10/18/2004-10.10.00  CYCLE 1.000 SECONDS

NUMBER OF INTERVALS 32            TOTAL LENGTH OF INTERVALS 04.49.44
DATE   TIME     INT   CPU   DASD  DASD   JOB   JOB   TSO   TSO   STC   STC  ASCH  ASCH  OMVS  OMVS SWAP DEMAND
MM/DD HH.MM.SS MM.SS  BUSY  RESP  RATE   MAX   AVE   MAX   AVE   MAX   AVE   MAX   AVE   MAX   AVE RATE PAGING
10/18 08.40.00 10.00  55.1   1.6  3447    12     8     2     2   108   105     0     1     1     1 0.00   0.08
10/18 08.50.00 10.00  83.1   2.2  2161    15     8     2     2   109   105     0     1     1     1 0.00   0.55
10/18 09.00.00 09.59  98.6   2.1  1021    12     8     2     2   109   105     0     1     1     1 0.00   0.30
10/18 09.10.00 10.00  99.7   1.2  1029    12    12     2     2   102   101     0     1     1     1 0.00   0.11
10/18 09.20.00 10.00  99.6   1.3  1003    13    12     2     2   102   101     0     1     1     1 0.00   0.14
10/18 09.30.00 09.59  99.6   1.2  1026    12    12     2     2   102   101     0     1     1     1 0.00   0.10
10/18 09.40.00 09.59  99.6   1.2  1008    13    12     2     2   102   101     0     1     1     1 0.00   0.08
10/18 09.50.00 10.00  96.8   2.2  1934    13    12     2     2   101   100     0     1     1     1 0.00   1.86
10/18 10.00.00 09.59  95.7   2.3  2009    13    12     2     2   100    99     0     1     1     1 0.00   0.13
10/18 10.10.00 10.00  95.9   2.3  2026    13    12     2     2   100    99     0     1     1     1 0.00   0.08
10/18 10.20.00 10.00  95.7   2.3  2068    12    12     2     2   100    99     0     1     1     1 0.00   0.13
```

Report analysis

- The first column to check is `CPU BUSY`. For systems running in LPAR mode, as in this example, this means LPAR BUSY percentage. We are interested in the interval starting at 09.10.00; during that time, CPU BUSY was 99.7%.
- The column `DASD RESP` indicates the average number of milliseconds required to complete an I/O request on direct access device storage was 1.2 ms.
- The column `DASD RATE` gives the number of I/Os for DASD per second: 1029.
- The columns `JOB MAX` and `JOB AVG` give the maximum and average number of batch jobs that were running during the interval: 12.

- The TSO, STC, and OMVS columns give maximum and average numbers for these address space types. Though we are not yet interested in these numbers, we can check whether there is a deviation from the previous or next interval values.
- No paging activity (DEMAND PAGING) is observed.

Conclusions that suggest further investigation:

- The CPU BUSY value is very high; this requires an additional look at the overall CPU resources by using CPU Activity report.
- DASD I/O rate suffers a decrease from a 3000 level to 1000 level in 30 minutes.
- JOB AVE (or the number of concurrent running Jobs) has an increase, which indicates the potential existence of a bottleneck.

7.2 CPU delay investigation

Knowledge of the environment may let the performance expert start directly with a specific report or panel, but in this study, we follow a step-by-step approach.

7.2.1 Using the CPU Activity report

This report is divided in two sections: the first one is called CPU Activity report, the second one is the Partition Data report. There we begin with control statement REPORTS(CPU).

Example 7-2 Postprocessor CPU Activity Report (Partition Data)

```
                                              P A R T I T I O N   D A T A   R E P O R T
                                                                                                                            PAGE    2
             z/OS V1R5                        SYSTEM ID SYS1           START 10/18/2004-09.10.00  INTERVAL 000.10.00
                                              RPT VERSION V1R5 RMF     END   10/18/2004-09.20.00  CYCLE 1.000 SECONDS

 MVS PARTITION NAME                       A04
 IMAGE CAPACITY                            93
 NUMBER OF CONFIGURED PARTITIONS           29
 NUMBER OF PHYSICAL PROCESSORS             24
                    CP                     18
                    ICF                     6
 WAIT COMPLETION                           NO
 DISPATCH INTERVAL                    DYNAMIC

--------- PARTITION DATA -----------------  -- LOGICAL PARTITION PROCESSOR DATA --   -- AVERAGE PROCESSOR UTILIZATION PERCENTAGES --
                    ----MSU----  -CAPPING--  PROCESSOR-  ----DISPATCH TIME DATA----   LOGICAL PROCESSORS  --- PHYSICAL PROCESSORS ---
 NAME       S   WGT  DEF    ACT  DEF   WLM%  NUM   TYPE   EFFECTIVE       TOTAL      EFFECTIVE    TOTAL  LPAR MGMT  EFFECTIVE  TOTAL
 A04        A    10    0     46  NO     0.0    1   CP    00.09.57.950  00.09.57.950        99.66    99.66      0.00       5.54   5.54
 A0A        A   185    0      4  NO     0.0    2   CP    00.00.46.761  00.00.50.044         3.90     4.17      0.03       0.43   0.46
 A0B        A    10    0      2  NO     0.0    2   CP    00.00.25.633  00.00.26.687         2.14     2.22      0.01       0.24   0.25
 A0C        A    10    0      4  NO     0.0    2   CP    00.00.51.366  00.00.54.774         4.28     4.56      0.03       0.48   0.51
        .................................................
 A17        A    10    0      0  NO     0.0    2   CP    00.00.04.555  00.00.06.037         0.38     0.50      0.01       0.04   0.06
 A18        A    10    0      0  NO     0.0    1   CP    00.00.00.233  00.00.00.235         0.04     0.04      0.00       0.00   0.00
 A19        A   185   10      1  NO     0.0    2   CP    00.00.05.530  00.00.07.362         0.46     0.61      0.02       0.05   0.07
 *PHYSICAL*                                                            00.11.21.882                             6.31              6.31
                                                         ------------  ------------                           ------     ------ ------
   TOTAL                                                 00.22.35.096  00.34.48.819                             6.79      12.55  19.34
```

Report analysis

- First of all, you can verify on the Monitor III CPC Capacity report the current configuration: in our case, we have a 2084-318 processor with 18 CPs and a CPC capacity value of 837. The logical partition that we are interested in, A04, is defined with two reserved processors, but only one CP is online; therefore, the actual consumption is limited to 46.
- Example 7-2 shows measurement data for all configured partitions. From this report, we focus on our LP (A04).
  - PARTITION NAME: A04

- AVERAGE LOGICAL PROCESSOR UTILIZATION: 99.66%
- AVERAGE PHYSICAL PROCESSOR UTILIZATION: 5.54%
- Number of logical processors online in LP A04: 1

- The system is running in LPAR A04, which is defined with a weight (WGT) of 10. The total weight of all active LPARs is 910 (we removed some LPs from the example). This means that A04 is guaranteed 1.1% of the total CPC capacity provided by the 18 CPUs. It uses 5.54% which is five times its weight. This is only possible because other LPS are not exploiting their guarantee. This LP cannot go beyond 5.55% because it has currently assigned only one logical CP, that is 1/18 = 0.555. Then, we can assume that there is a latent demand.
- The CPC is not overloaded: 19.34% (TOTAL at the end of report). In such an environment—with idle processor time and latent demand, at least in A04—it will be worthwhile to implement the IRD feature WLM Vary CPU Management.
- CAPPING DEF=NO indicates that the partition is not LPAR capped.
- MSU DEF=0 indicates that defined capacity (also called soft cap) is not used.
- MSU ACT=46 shows actual consumption is 46 MSU/h. This value corresponds to 5.54% of the CPC, which has 837 total MSU/h (46/837 = 0.55). The LP image capacity is 93 MSU/h. If an LP has no defined capacity limit (no soft capping), this value is calculated based on the partition's logical CP configuration, with the following formula:

$$\text{Image capacity} = \frac{\text{CPC capacity}}{\text{\# Phys. Processors}} \times \text{\# Logical Processors (Initial+Reserved)} = \frac{837}{18} \times 2 = 93$$

- Logical partition A0A has a weight value of 185, which is 20% of the total weight. Due to the fact that only 2 CPs are defined (11% of 18 CPs), A0A cannot get the resources as defined with the weight value; four CPs would be required to reach the upper limit. This is another reason to implement the IRD feature WLM Vary CPU Management.

With a value of LPAR BUSY = 99.66%, we *have to* investigate the MVS view of the CPU utilization in the first section of the CPU Activity report shown in Example 7-3.

Example 7-3 Postprocessor CPU Activity report

```
                                                      C P U  A C T I V I T Y
                                                                                                          PAGE    1
            z/OS V1R5                SYSTEM ID SYS1            START 10/18/2004-09.10.00  INTERVAL 000.10.00
                                     RPT VERSION V1R5 RMF      END   10/18/2004-09.20.00  CYCLE 1.000 SECONDS

CPU  2084   MODEL  318

CPU     ONLINE TIME   LPAR BUSY       MVS BUSY      CPU SERIAL  I/O TOTAL         % I/O INTERRUPTS
NUMBER  PERCENTAGE    TIME PERC       TIME PERC     NUMBER      INTERRUPT RATE     HANDLED VIA TPI
 0      100.00        99.66           99.96         046A3A       1038               1.59
TOTAL/AVERAGE         99.66           99.96                      1038               1.59

SYSTEM ADDRESS SPACE ANALYSIS         SAMPLES =    600
            NUMBER OF ASIDS                                    DISTRIBUTION OF QUEUE LENGTHS  (%)
TYPE       MIN    MAX       AVG      0      1      2      3      4      5      6    7-8   9-10  11-12  13-14    14+
----    ------ ------  --------  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----
IN
READY        2     23      10.5    0.0    0.0    0.1    0.0    0.5    1.3    2.0    9.0   26.0   56.3    3.1    1.5

                                     0    1-2    3-4    5-6    7-8   9-10  11-15  16-20  21-25  26-30  31-35    35+
                                 -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----
IN          71     73      71.4    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0  100.0
OUT
READY        0      0       1.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0
OUT
WAIT         0      0       1.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0
LOGICAL
OUT RDY      0      1       1.0    2.6    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0
LOGICAL
OUT WAIT    43     45      44.5    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0  100.0
BATCH       11     12      12.0    0.0    0.0    0.0    0.0    0.0  100.0    0.0    0.0    0.0    0.0    0.0    0.0
STC        101    102     101.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0  100.0
```

```
TSO          2     2     2.0  100.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
ASCH         0     0     1.0    0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
OMVS         1     1     1.0  100.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
```

Report analysis

First, we have to explain the difference between LPAR BUSY and MVS BUSY:

- LPAR BUSY TIME PERC
  The LPAR BUSY time is the dispatch time of the processors that are assigned to the logical CPUs in the partition and the percentage is based on their online time. This definition applies when Wait Completion = NO, which means that logical CPUs in wait do not hold the physical CP (this is a recommended option).
- MVS BUSY TIME PERC
  The MVS BUSY TIME for one processor is the difference between the Online Time and the Wait Time. It includes the LPAR BUSY TIME plus the time that the logical CPU is ready to work, but there are no physical CPs available. The MVS BUSY TIME can be higher than the LPAR BUSY TIME, and the difference between the two values is an indicator of CPU latent demand. This conclusion is true when we have enough logical CPUs in the LP; in this case, if the LP is key to the business, the solution is to raise the LP weight.
- In the report:
  - `LPAR BUSY TIME PERC` = 99.66% (as seen in the partition report)
  - `MVS BUSY TIME PERC` = 99.96%

Comments

- LPAR BUSY is very high. It could indicate that there are not enough processor resources.
- If we check the IN READY QUEUE (from an RMF point of view, this is the count with address spaces with active and ready dispatchable units) for contention, we see high numbers, indicating significant contention:
  - MVS BUSY is also very high. However, in this case the difference is zero because the real latent demand is hidden by the fact that there is just one logical CPU.
  - During 100% of the time, there was more than one task ready, waiting to be dispatched.
  - Almost 80% of the time (26.0% + 56.3%), there were 9 to 12 tasks ready and active.

This activity has to be handled by *one* CP only, which leads to a large amount of pending work. But to know which workload is affected by those delays, we have to pursue our analysis with Monitor III because it provides a good navigation capability among the different reports that we want to analyze.

7.2.2 Using Monitor III

Using Monitor III, we continue our analysis with the Sysplex Summary report that we can call with the command **SYSSUM**.

Example 7-4 Sysplex Summary Report

```
                     RMF V1R5   Sysplex Summary - SYSXPLEX       Line 1 of 19
Command ===>                                                    Scroll ===> CSR

WLM Samples: 1200   Systems: 3  Date: 11/01/04 Time: 09.12.00    Range: 300 Sec

                      >>>>>>>>XXXXXXXXXXXXXXXXXX<<<<<<<<
Service Definition: WLMDEF                Installed at: 10/18/04, 08.30.52
     Active Policy: WLMPOL                Activated at: 10/18/04, 08.31.03

               ------- Goals versus Actuals --------  Trans --Avg. Resp. Time-
               Exec Vel  --- Response Time ---  Perf  Ended  WAIT EXECUT ACTUAL
Name     T  I  Goal Act  ---Goal--- --Actual--  Indx  Rate   Time   Time   Time

BAT_WKL  W           15                               0.027 1.230  3.78M  3.79M
BATHI    S  3    30  46                         0.65  0.020 1.198  2.66M  2.68M
BATLO    S  D       0.3                               0.000 0.000  0.000  0.000
BATMED   S  5    25 2.6                         9.57  0.000 0.000  0.000  0.000
BATPIER  S  4    30  13                         2.32  0.007 1.324  7.11M  7.12M
```

Using the report option (**RO**) command, the report as shown in Example 7-4 has been filtered to display only the BAT_WKL workload and associated service classes. The report covers a five minute interval.

Report analysis

- BAT_WKL is a workload, as indicated by W in the T column.
- Regarding service class BATHI:
  - We get the characteristics of the goal of the service class:
    - Importance 3
    - Velocity 30
  - It achieves its goal. This is indicated in Perf Indx = 0.65 which, for an execution velocity type of goal, is the ratio between goal and actual value (30/46).
  - 6 jobs ended during the interval: 0.02 * 300 = 6, with an average response time of 2.68 minutes.
- For service class BATLO
  - Its goal is discretionary (D).
  - Execution velocity is very low (0.3), and this might not be acceptable even for a discretionary goal.
- Considering service class BATMED:
  - For execution velocity, this service class has a goal of 25 while the actual value is 2.6; this results in a very bad performance index of 9.57.
  - No jobs ended during the interval, which is the very problem reported.

With the **DELAY** command, we obtain the Delay report shown in Example 7-5. This report is not a sysplex-wide report, but a local system report. We have to choose (in the `System` field) the system to be analyzed. By default, it is the system where we are logged on (SYS1, which corresponds to the A04 LP).

Example 7-5 Monitor III Delay report

```
                      RMF V1R5   Delay Report                   Line 1 of 20
Command ===>                                                Scroll ===> CSR

Samples: 300     System: SYS1  Date: 10/18/04  Time: 09.12.00  Range: 300   Sec

            Service     WFL USG DLY IDL UKN ---- % Delayed for ---- Primary
Name     CX Class    Cr  %   %   %   %   %  PRC DEV STR SUB OPR ENQ Reason

JESJOP4  B  BATLO         0   0 100   0   0 100   0   0   0   0   0 JESJOP8
JESJOP6  B  BATLO         0   0 100   0   0 100   0   0   0   0   0 JESJOP8
JESJOA3  B  BATMED        2   2  96   0   3  95   1   0   0   0   0 JESJOP8
JESJOA1  B  BATMED        3   3  97   0   2  96   2   0   0   0   0 JESJOP8
JESJOA2  B  BATMED        3   3  96   0   2  96   1   0   0   0   0 JESJOP8
JESJOP1  B  BATPIER       6   4  63   0   5  63   0   0   0   0   0 JESJOP9
JESJOP9  B  BATHI        10   0   3   0   1   3   0   0   0   0   0 JESJOP8
JESJOP3  B  BATPIER      11  10  84   0   6  84   0   0   0   0   0 JESJOP8
JESJOP8  B  BATHI        19   1   4   0   4   4   0   0   0   0   0 JESJOP7
```

From a Delay report, we obtain significant information about delays:

- The address spaces that are being delayed
- The type of the delay (and the main cause originator)
- The amount of the delay

The delays that are monitored by Monitor III are the following:

PRC Processor delay: Job has ready dispatchable units but it is not dispatched.

DEV Device delay: Job is delayed for DASD or tape. I/Os have been queued.

STR Paging or swapping: Job is waiting for a page fault or is swapped out in the out ready queue.

SUB Subsystem delay: Job is delayed by JES, HSM, or XCF request.

OPR Operator delay: Job is delayed by a pending WTOR, a mount, or a quiesce.

ENQ Enq delay: Job is waiting for an enqueued resource.

As you can see, there are several sources for the delay of a job. In this chapter, we discuss only delays coming from the most important resources in a system, the processors and the I/O subsystem. In current installations, storage should not be a problem. If you observe storage delays or high paging activity in your environment, you can analyze the problem in a similar way to the analysis we are performing here with the processor and device delays.

For a complete description, refer to *RMF Report Analysis*, SC33-7991.

Delay report analysis

- Processor delay is the major delay which affects the job running in the service class BATMED.

A detailed view of the workload activity is now necessary, and we will use the Postprocessor to get a Workload Activity report.

7.2.3 Using the Workload Activity report

The report has a sysplex scope. The control statement is `SYSRPTS(WLMGL(SCPER))`.

Example 7-6 shows the upper portion of the Workload Activity report and Example 7-7 shows the policy portion from the same report, which you can request with the Postprocessor control statement SYSRPTS(WLMGL(POLICY)).

Example 7-6 Workload Activity report (service class)

```
                                             W O R K L O A D   A C T I V I T Y
                                                                                                                     PAGE   2
      z/OS V1R5               SYSPLEX SYS#PLEX           START 10/18/2004-09.10.00 INTERVAL 000.10.00   MODE = GOAL
                         CONVERTED TO z/OS V1R5 RMF      END   10/18/2004-09.20.00

REPORT BY: POLICY=WLMPOL     WORKLOAD=BAT_WKL    SERVICE CLASS=BATMED     RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=5
                                                 CRITICAL     =NONE

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--    ---SERVICE----   --SERVICE RATES--   PAGE-IN RATES    ----STORAGE----
AVG      3.00   ACTUAL                   0   SSCHRT  15.6   IOC      4686    ABSRPTN      1020   SINGLE      0.0   AVG     3882.74
MPL      3.00   EXECUTION                0   RESP     7.8   CPU    190642    TRX SERV     1020   BLOCK       0.0   TOTAL   11648.3
ENDED       0   QUEUED                   0   CONN     4.1   MSO      1639K   TCB           8.7   SHARED      0.0   CENTRAL 11648.3
END/S    0.00   R/S AFFINITY             0   DISC     0.0   SRB      2171    SRB           0.1   HSP         0.0   EXPAND     0.00
#SWAPS      0   INELIGIBLE               0   Q+PEND   0.3   TOT      1837K   RCT           0.0   HSP MISS    0.0
EXCTD       0   CONVERSION               0   IOSQ     3.4   /SEC     3061    IIT           0.0   EXP SNGL    0.0   SHARED     0.00
AVG ENC  0.00   STD DEV                  0                                   HST           0.0   EXP BLK     0.0
REM ENC  0.00                                                                APPL %        1.5   EXP SHR     0.0
MS ENC   0.00

VELOCITY MIGRATION:   I/O MGMT   2.7%    INIT MGMT  2.7%

        ---RESPONSE TIME---  EX   PERF  AVG   --USING%- ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--    %
        HH.MM.SS.TTT         VEL  INDX ADRSP   CPU  I/O  TOT  CPU  I/O                          UNKN IDLE  USG  DLY  USG  DLY QUIE
GOAL                        25.0%
ACTUALS
SYS1                         2.7%  9.4   1.0   0.5  2.1 95.0 93.1  1.9                           2.4  0.0  0.0  0.0  0.0  0.0  0.0
```

Example 7-7 Workload Activity report (policy)

```
SERVICE DEFINITION: WLMDEF    Sample WLM Service Definition.   -SERVICE DEFINITION COEFFICIENTS-
 INSTALL DATE: 10/27/2004 20.31.14  INSTALLED BY: CASSIER      IOC    CPU     SRB     MSO
 POLICY: WLMPOL    Sample WLM policy
 I/O PRIORITY MANAGEMENT:  YES                                  0.1    1.0     1.0  0.1000
 DYNAMIC ALIAS MANAGEMENT: YES

SYSTEMS
 ---ID--- OPT SU/SEC  --TIME-- INTERVAL
 SYS1      00 20752.3 07.30.00 00.09.59
```

Report analysis

This report shows service class BATMED.

- No jobs have ended during the interval; therefore, an average elapsed time cannot be calculated.
- CPU consumption for this service class is 1.5% (APPL%) of one CPU.

  The value is measured by z/OS and accounted to the following elements: TCB, SRB, RCT (region control task), IIT (I/O FLIH), and HST (hiperspace processing). From TCB time (8.7 secs), the number of CPU service units (190642 SUs) is calculated by multiplying it by the SRM constant for this LP (SRM constant = 20752 SUs/sec) and by the CPU SDC (1.0). The SRM constant is listed in the field SU/SEC in the Policy section of the Workload Activity report (Example 7-7).

  APPL% shows CPU utilization based on a single processor capacity. This means that the value can exceed 100% in a sysplex with more than one processor. To get the system utilization, this value has to be divided by the number of processors.

What is the problem?

The problem is correctly identified as not having enough processor resources. As we saw in the CPU Activity report, there is a significant lack of CPU resource. And during the previous

RMF interval (9:00), the CPU busy percentage was just a little bit lower. We have to look at the same report for the previous interval.

Example 7-8 Workload Activity Report (interval 09.00.00)

```
                                                  W O R K L O A D   A C T I V I T Y
                                                                                                                     PAGE   2
      z/OS V1R5               SYSPLEX SYS#PLEX          START 10/18/2004-09.00.00 INTERVAL 000.10.00   MODE = GOAL
                         CONVERTED TO z/OS V1R5 RMF      END   10/18/2004-09.10.00

REPORT BY: POLICY=WLMPOL     WORKLOAD=BAT_WKL    SERVICE CLASS=BATMED     RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=5
                                                 CRITICAL     =NONE

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--   ---SERVICE----   --SERVICE RATES--   PAGE-IN RATES    ----STORAGE----
AVG      2.02   ACTUAL             6.29.191   SSCHRT 167.0   IOC      49738    ABSRPTN     15579   SINGLE     0.0   AVG     3909.03
MPL      2.02   EXECUTION          6.28.260   RESP     6.5   CPU      1954K   TRX SERV    15579   BLOCK      0.0   TOTAL   7911.91
ENDED       2   QUEUED                  930   CONN     4.1   MSO     16889K   TCB          89.4   SHARED     0.0   CENTRAL 7911.91
END/S    0.00   R/S AFFINITY              0   DISC     0.0   SRB      26704    SRB           1.2   HSP        0.0   EXPAND     0.00
#SWAPS      0   INELIGIBLE                0   Q+PEND   0.2   TOT     18919K   RCT           0.0   HSP MISS   0.0
EXCTD       0   CONVERSION                0   IOSQ     2.2   /SEC     31534    IIT           0.5   EXP SNGL   0.0   SHARED     0.00
AVG ENC  0.00   STD DEV            1.01.079                                    HST           0.0   EXP BLK    0.0
REM ENC  0.00                                                                  APPL %       15.2   EXP SHR    0.0
MS ENC   0.00

VELOCITY MIGRATION:   I/O MGMT  42.9%    INIT MGMT 42.9%

        ---RESPONSE TIME---  EX   PERF  AVG   --USING%- ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--    %
        HH.MM.SS.TTT         VEL  INDX ADRSP   CPU  I/O  TOT  CPU  I/O CAPP                     UNKN IDLE  USG  DLY  USG  DLY QUIE
GOAL                        25.0%
ACTUALS
SYS1                        42.9%  0.6   0.7   6.1 28.3 45.7 29.2 16.4  0.2                     19.8  0.0  0.0  0.0  0.0  0.0  0.0
```

With this report, we are able to compare some fields of the BATMED service class:

- APPL% - The utilization CPU decreased from 15.2% to 1.5%.
- MPL - The number of concurrent jobs increased from 2 to 3.

If we compare the other service classes, we obtain the values for APPL% and MPL for each service class, as shown in Table 7-1.

Table 7-1 APPL% and MPL for each service class

Interval	BATHI	BATLO	BATMED	BATPIER	TOTAL
09.00.00	**33.8** / 2	3.2 / 2	**15.2** / 2	33.7 / 2	**85.9** / 8
09.10.00	**64.5** / 3	0.4 / 3	**1.5** / 3	20.2 / 3	**86.6** / 12

So, even if four additional jobs are running, the total CPU consumption of the batch workload does not increase significantly (from 85.9% to 86.6%), but service class BATHI used 75% of the CPU resources used for batch.

What can we do?

There are several ways to react to this. Which of the following solution is appropriate depends on the requirements of all the workloads in the system:

- Do nothing. WLM is driving toward business goals and BATHI is the more important work, so unless a change happens in your SLAs, nothing should be done by an analyst. BATMED is less important and so WLM is doing exactly what it has been instructed to do. In reality, WLM is in fact doing something more: it is applying discretionary goal management because you can see that BATHI has a PI <0.7 and WLM is capping the work to donate to BATLO. It is not helping BATMED, which is the problem. So maybe you should have a look at the performance goal for BATLO.
- Adjust the performance goals.

- Decrease the MPL of BATHI (for example, by reducing the number of initiators serving a dedicated JES class for those jobs).
- Increase the CPU capacity by adding one or more logical processors to the LP.
- Increasing the weight does not help because the problem is the number of logical CPUs.

7.3 I/O delay investigation

To continue with our example, a second processor has been varied online.

Second case study

In this case, we investigate an I/O delay situation and we use the RMF report to show how you can collect the information needed to help you diagnose the problem.

7.3.1 Investigating delays

We check the effect of increasing the CPU capacity with a new Sysplex Summary report for the interval at 10.02.00.

Example 7-9 Monitor III Sysplex Summary Report

```
                     RMF V1R5   Sysplex Summary - SYS#PLEX         Line 1 of 5
Command ===>                                                  Scroll ===> CSR

WLM Samples: 1200    Systems: 2  Date: 10/18/04 Time: 10.02.00 Range: 300   Sec

                      >>>>>>>>XXXXXXXXXXXXXXXXXX<<<<<<<<

Service Definition: WLMDEF                Installed at: 10/18/04, 08.30.52
     Active Policy: WLMPOL                Activated at: 10/18/04, 08.31.03

               ------- Goals versus Actuals --------  Trans --Avg. Resp. Time-
               Exec Vel  --- Response Time ---   Perf  Ended   WAIT EXECUT ACTUAL
Name     T  I  Goal Act  ---Goal--- --Actual--  Indx  Rate    Time   Time   Time

BAT_WKL  W           36                               0.053 1.172  4.48M  4.50M
BATHI    S  3    30  44                         0.68  0.020 1.165  2.58M  2.59M
BATLO    S  D        19                               0.007 1.286  6.27M  6.29M
BATMED   S  5    25  38                         0.66  0.010 0.835  9.12M  9.14M
BATPIER  S  4    30  46                         0.65  0.017 1.338  3.27M  3.29M
```

This time, the Sysplex Summary report does not show any critical indicator. We then check the Delay report.

Example 7-10 Monitor III Delay report

```
                      RMF V1R5   Delay Report                     Line 1 of 2
Command ===>                                                  Scroll ===> CSR

Samples: 300     System: SYS1  Date: 10/18/04  Time: 10.02.00  Range: 300   S

            Service     WFL USG DLY IDL UKN ---- % Delayed for ---- Primary
Name     CX Class    Cr  %   %   %   %   %  PRC DEV STR SUB OPR ENQ Reason

JESJOP4  B  BATLO         9   7  64   0   9  63   0   0   0   0   0 JESJOP3
JESJOP8  B  BATHI        13   0   2   0   0   1   0   0   0   0   0 JESJOP9
```

```
JESJOP9  B  BATHI       35   7  12   0   7   5   0   0   0   0   0 RG-Cap
JESJOA3  B  BATMED      37  15  25   0   3   2  24   0   0   0   0 NW353F
JESJOP3  B  BATPIER     37   8  13   0  12   9   0   0   0   0   0 JESJOP1
JESJOA3  B  BATMED      38  20  32   0   7   4  30   0   0   0   0 NW353F
JESJOA1  B  BATMED      40  35  52   0   7   4  50   0   0   0   0 NW353F
JESJOP8  B  BATHI       40  18  25   0  13   8   1   0   0   0   0 RG-Cap
JESJOA1  B  BATMED      42   3   5   0   1   1   4   0   0   0   0 NW353F
```

Analyzing the delayed job

Device delay is now the major delay for the service class BATMED. With cursor-sensitive control for field **50** (device delay for job JESJOA1), we get the Job Delays report.

Example 7-11 JOB Delays report

```
                        RMF V1R5   Job Delays                      Line 1 of 2
Command ===>                                                   Scroll ===> CSR

Samples: 300     System: SYS1  Date: 10/18/04  Time: 10.02.00  Range: 300   Sec

Job: JESJOA1      Primary delay: Excessive connect time on volume NW353F.

Probable causes: 1) Unnecessary I/O.
                 2) Excessive PDS directory searches or VTOC activity.
                 3) Too many active datasets on volume.

-------------------------- Volume NW353F Device Data --------------------------
Number:    353F        Active:      99%        Pending:    6%     Average Users
Device:    33903       Connect:     92%        Delay DB:   0%       Delayed
Shared:    Yes         Disconnect:   1%        Delay CM:   0%         1.5

--------------------------- Job Performance Summary ---------------------------
        Service       WFL -Using%- DLY IDL UKN ---- % Delayed for ---- Primary
CX ASID Class    P Cr  %   PRC DEV  %   %   %  PRC DEV STR SUB OPR ENQ Reason
B  0048 BATMED   1     40    8  28  52   0   7   4  50   0   0   0   0 NW353F
B  0030 BATMED   1     42    2   2   5   0   1   1   4   0   0   0   0 NW353F
```

Example 7-11 provides a detailed description of the delays job JESJOA1 is experiencing. The main reason for the delays seems to be the I/O connect time (92%) on the volume NW353F. The report does not show which address space/enclave is causing this high connect time in such device. This may or may not be a performance problem, so we go on in our analysis to get more details. Refer to “I/O performance and metrics” on page 165 for more information about the components of the I/O response time.

Using the cursor-sensitive control for field **28**, which is `USING% DEV`, we get the Device Delays report shown in Example 7-12.

Example 7-12 Device Delays report

```
                      RMF V1R5   Device Delays                    Line 1 of 15
Command ===>                                                   Scroll ===> CSR

Samples: 300     System: SYS1  Date: 10/18/04  Time: 10.02.00  Range: 300   Sec

         Service  DLY USG CON  ------------ Main Delay Volume(s) ------------
Jobname  C Class    %   %   %    %  VOLSER   %  VOLSER   %  VOLSER   %  VOLSER

JESJOA1  B BATMED   50  28  28   50 NW353F
JESJOA2  B BATMED   35  29  23   35 NW353F
JESJOA3  B BATMED   30  17  18   30 NW353F
```

```
JESJOA2  B BATMED    12   8   8   12 NW353F
JESJOA1  B BATMED     4   2   3    4 NW353F
JESJOP7  B BATHI      2  11   9    1 PAVTS1    0 SYS#J1
JESJOP8  B BATHI      1  10   8    1 PAVTS1    0 SYS#J1
JESJOP1  B BATPIER    1   4   6    1 SYS#J1
JESJOP3  B BATPIER    1   6   8    1 PAVTS1
JESJOP9  B BATHI      1   4   4    1 PAVTS1
JESJOP5  B BATLO      1   2   7    1 SYS#J1
```

Analyzing the stressed volume

This report shows jobs being delayed by a device. We can focus on the specific device by placing the cursor under the VOLSER NW353F. We get the Device Resource Delays report shown in Example 7-13.

Example 7-13 Device Resource Delays report

```
                     RMF V1R5   Device Resource Delays          Line 1 of 6
Command ===>                                                Scroll ===> CSR

Samples: 300     System: SYS1  Date: 10/18/04  Time: 10.02.00  Range: 300   Sec

Volume S/   Act  Resp  ACT CON DSC PND %,  DEV/CU            Service  USG DLY
  /Num PAV  Rate Time   %   %   %  Reasons Type     Jobname  C Class     %   %

NW353F S     228 .011   99  92   1 PND   6 33903    JESJOA1  B BATMED    28  50
  353F                                     3990-3   JESJOA2  B BATMED    29  35
                                                    JESJOA3  B BATMED    17  30
                                                    JESJOA3  B BATMED    13  24
                                                    JESJOA2  B BATMED     8  12
                                                    JESJOA1  B BATMED     2   4
```

You can see a list of jobs being delayed by this specific device and some performance-related data. Refer to “I/O performance and metrics” on page 165 for more details on I/O response time elements. Before we start our analysis, let us make some observations on the data presented by this report. The response time is in seconds and its components (although IOSQ% is not listed) are in percentage scale. As in the previous report, we can see that the connect time is the largest component of the response time. We convert the numbers into some metrics for the 300 seconds period:

- 228 I/Os per sec with an average of .011 seconds yield an accumulated IO time of: 228 * 300 * .011 = 752 seconds.
- At 99% of connect time, the accumulated connect time is: 0.99 * 300 = 297 seconds.
- For 1% of disconnect time, the accumulated disconnect time is: 0.01 * 300 = 3 seconds.
- 6% of pending time yields an accumulated pending time of: 0.06 * 300 = 18 seconds.
- 752 - (297 + 3 + 18) = 434 seconds of IOSQ time.

So, definitively, the IOSQ time is the major component of the I/O response time.

Now, let’s see what other information is provided in the report:

- `S` indicates that the device was z/OS-generated as a shared device.
- `PAV` gives the number of parallel access UCBs (base and alias) available at the end of the interval. Here no PAV has been defined for this device so there is no concurrency.
- `Act Rate` is the number of I/O operations completed per second, 228.
- `Resp Time` is the average response time in seconds, which is 11 ms for this device.

Response time = IOSQ time + Pending time + Disconnect time + Connect time

- `ACT` is the percentage of elapsed time in which the device was active, that is, with one SSCH instruction outstanding (no I/O interrupt occurred yet for that SSCH). It follows the formula:

  Active time = Pending + Connect + Disconnect time
  Then: Response time - Active time = IOSQ time
- `CON` is the percentage of the elapsed time that the device is executing one I/O operation in connect mode.
- `DSC` is the percentage of the elapsed time that the device is executing one I/O operation in disconnect mode. The disconnect time for NW353F volume is mainly 0 or close to 0, providing an indication that the cache hit percent is very high. We can verify this later in 7.3.2, "Investigating cache activity" on page 230.
- `PND` is the percentage of the elapsed time that the I/O requests to the device are in the channel subsystem (SAP) queues. Another equivalent definition is the time from the SSCH instruction (issued by z/OS) till the starting of the dialog between channel and I/O controller. Besides the value of the pending time, the report shows its partition time caused by controller command reply (CMR) delay and device busy delays (only caused by reserve). In our case the pending time is 6% and there is no indication of CMR or device busy delays. So, one of the following can be the reason for the PND time: all ESCON channel busy, ESCON DPS busy, CUB, DB due to ongoing I/O. The I/O Queuing Activity report in Example 7-19 on page 233 indicates some % DPS delay (0.7%), so the remaining 6% is due to all ESCON channels being busy and to the SAP overhead.

In this case, we found that the reason of our delay is the NW353F volume, where we see a high I/O rate for a non-PAV device and a high IOSQ time. Here are some considerations on how to decrease the IOSQ time:

- Increase the parallelism as a way to decrease queue times:
  - If your controller has support for dynamic parallel access volume (PAV), as in the ESS controller, the most effective action to decrease your high IOSQ time is to implement this hardware/software feature. With PAV you may have parallel I/Os against the same logical 3390/3380 device (two writes or one write and several reads in the same extent are not allowed), consequently decreasing the IOSQ time. However, in order to have *n* parallel access on a PAV device, you need to have *n* aliases UCBs, *n* UCWs and *n* I/O addresses in the controller. If dynamic PAV is activated, WLM manages the number of UCBs and UCWs per device, increasing the parallelism in certain devices (and decreasing it in others) in order to fulfill the goals of the most important service classes. Refer to "ESS performance features" on page 234 for more on PAV.
  - Use the great level of parallelism implemented through FICON channels.
- Buy a faster device/channel to decrease the I/O service time (connect plus disconnect) and consequently the utilization (for the same I/O load) and the queue time.
- Decrease the I/O service time by tuning.
- Change the service class goal containing the transaction which is generating the I/Os causing the IOSQ time delay. Increase the importance of the goal or change its numerical value to make it more difficult to be obtained. As a consequence, the I/O priority will be raised by WLM because the transaction is not reaching its goal and the major delay is the I/O delay. Keep in mind that, in this case, you are not improving the I/O in general, but just improving the response time of your favored transactions.
- Decrease the I/O rate against the device by avoiding placement of several active data sets on the same volume, for example index and data sections from to the same VSAM cluster. If this happens, verify that your ACS routines are forcing the allocation to a specific volume by using guaranteed space. Use different storage groups or the DFSMS Data Set

Separation function that allows you to separate data sets from each other in different physical DASD controllers.

- Determine whether the jobs are issuing I/Os against the same data set. SMF record type 42 can help in this diagnosis. If so, you can see if it is possible to run these job sequentially in order to reduce the contention.

> **Additional information:** Although this report indicates that IOSQ time is the largest component of the I/O response time, consider the following recommendations that you can use to address a situation where you need to decrease the connect time for both direct and sequential accesses.
>
> - For sequential I/O accesses, decrease the number of SSCHs instructions (I/O operations). For every I/O operation starting, there is a standard conversation between the channel and the controller. If you decrease the number of SSCH instructions, but transfer the same amount of data, you save in total connect time. In other words, your average connect time (per I/O operation) increases, but your total connect time decreases. To accomplish that, you can define many data buffers, thus causing the access method to chain I/O blocks in just one channel program.
> - Use FICON native or FICON Express channels for any type of access. For that, your controllers must have a host adapter able to understand such protocol.
> - Use data compression in CPU.
> - For VSAM data sets, decrease the free space in the CIs.

7.3.2 Investigating cache activity

We know that caching is not posing any problem in the performance of our volume because of the very low disconnect time, but we would still like to discuss what kind of information RMF can provide. Monitor III offers two reports that assist you in monitoring the performance of your cache subsystem. Both are global reports. They present all the I/O activity coming from any z/OS system (in the same or in different sysplex) from/to the controller.

We start with the Cache Summary report shown in Example 7-14. The cache investigation should be a required task when your disconnect time is a large value compared with the other components of the I/O response time.

Example 7-14 Monitor III Cache Summary Report

```
                     RMF V1R5   Cache Summary    - SYS#PLEX       Line 1 of 31
Command ===>                                                Scroll ===> CSR

Samples: 200     Systems: 3    Date: 10/20/04  Time: 17.43.00  Range: 60    Sec
                              CDate: 10/20/04 CTime: 17.41.35 CRange: 200   Sec

SSID  CUID Type-Mod  Size  I/O   Hit  Hit   -- Miss ---  Read  Seq  Async  Off
                           Rate   %   Rate  Total Stage    %   Rate Rate   Rate
00FB  3B00 9393-002 2048M   3.4 98.7   3.4    0.0   0.0  58.5   0.0   0.3   0.0
8802  37AF 9393-002 2048M 124.9  100 124.9    0.0   0.0  56.3   0.0   1.6   0.0
8803  37C0 9393-002 2048M   7.4  100   7.4    0.0   0.0  93.0   0.0   0.1   0.0
8820  3520 9393-002 2048M 229.0  100 229.0    0.0   0.0   100   0.0   0.0   0.0
8821  3560 9393-002 2048M   1.4 93.6   1.3    0.1   0.1  72.7   0.0   0.2   0.0
```

We know that volume NW353F is connected to control units 3500 and 3501, but we do not know these CUIDs in this report. The explanation for this is that the report does not really display the physical control unit number of the caching subsystem, but the lowest device number, or the device that has been turned online first, respectively.

This leads us to the information for NW353F (device number 353F):

- SSID: 8820
- All I/Os to this controller are reads and the cache hit ratio is 100%

Cursor-sensitive control for field 8820 shows the Cache SSID Information report, which offers some more details for this SSID, as shown in Example 7-15.

Example 7-15 Monitor III Cache SSID Information Report

```
                        RMF Cache SSID Information

 The following details are available for SSID 8820
 Press Enter to return to the Report panel.

 CUID    : 3520     Cache : Active           NVS   : Active
 Type-Mod: 9393-002 Config: 2048M            Config: 8192K
                    Avail : 1975M            Pinned:     0
                    Offl  :     0
                    Pinned:     0

        ------ Read ------   --------- Write ---------   Read   Tracks
         Rate    Hit  Hit%    Rate   Fast    Hit  Hit%     %
 Norm     1.2    1.2   100     0.1    0.1    0.1   100   92.2      0.0
 Seq    227.7  227.7   100     0.0    0.0    0.0   100    100      0.0
 CFW      0.0    0.0   0.0     0.0    0.0    0.0   0.0    0.0
 Total  228.9  228.9   100     0.1    0.1    0.1   100    100
```

Using cursor-sensitive control for one of the data fields in the Cache Summary report (Example 7-14 on page 230) navigates to the Cache Detail report.

Example 7-16 Monitor III Cache Detail Report

```
                     RMF V1R5   Cache Detail     - SYS#PLEX       Line 1 of 67
Command ===>                                                   Scroll ===> CSR

Samples: 200     Systems: 3    Date: 10/20/04  Time: 17.43.00  Range: 60    Sec
                              CDate: 10/20/04 CTime: 17.41.35 CRange: 200   Sec

Volume /Num SSID   I/O  I/O    Hit - Cache Hit Rate - - DASD I/O -  Seq   Async
                     %   Rate    %    Read   DFW   CFW  Total Stage  Rate  Rate

*ALL               100 229.0   100 228.9   0.1   0.0    0.0   0.0    0.0    0.0
*NOCAC             0.0   0.0   0.0   0.0   0.0   0.0    0.0   0.0    0.0    0.0
*CACHE             100 229.0   100 228.9   0.1   0.0    0.0   0.0    0.0    0.0
NW353F 353F 8820  99.5 227.8   100 227.8   0.0   0.0    0.0   0.0    0.0    0.0
TOTSS1 3520 8820   0.2   0.6   100   0.5   0.0   0.0    0.0   0.0    0.0    0.0
T5CI22 3518 8820   0.1   0.2   100   0.2   0.0   0.0    0.0   0.0    0.0    0.0
TOTSS6 352F 8820   0.0   0.1   100   0.1   0.0   0.0    0.0   0.0    0.0    0.0
MVS009 350C 8820   0.0   0.1   100   0.0   0.0   0.0    0.0   0.0    0.0    0.0
```

The report provides details for all volumes belonging to cache subsystem 8820, and we see that our favorite NM353F is the most heavy user. Again, we can get more information about the volume by using cursor-sensitive control for field NM353F. It now shows the Cache Volume Detail report as in Example 7-17.

Example 7-17 Monitor III Cache Volume Detail Report

```
                          RMF Cache Volume Detail

 The following details are available for Volume NW353F on SSID 8820
 Press Enter to return to the Report panel.

 Cache: Active       DFW: Active       Pinned: None

        ------ Read ------   --------- Write ---------   Read   Tracks
         Rate    Hit  Hit%    Rate   Fast    Hit  Hit%     %
 Norm     0.1    0.1   100     0.0    0.0    0.0   0.0   100      0.0
 Seq    227.7  227.7   100     0.0    0.0    0.0   0.0   100      0.0
 CFW      0.0    0.0   0.0     0.0    0.0    0.0   0.0   0.0
 Total  227.8  227.8   100     0.0    0.0    0.0   0.0   100

 ------ Misc ------  - Non-Cache -  --- CKD ----  - Record Caching -
 DFW Bypass :   0.0  ICL   :   0.0  Write:   0.0  Read Miss :   0.0
 CFW Bypass :   0.0  Bypass:   0.0  Hits :   0.0  Write Prom:   0.0
 DFW Inhibit:   0.0
```

This really confirms that all read activity for this volume is done in the cache. By the way, all these reads are sequential.

In summary, we consider that volume NW353F has a high activity rate, good cache characteristics, but long response times caused by long IOS queue times. This is an indicator of high access rate to this volume, generated in our z/OS system. This problem can be solved by the performance features of the Enterprise Subsystem Server (ESS) which are described in 7.3.4, “ESS performance features” on page 234.

7.3.3 Investigating channel activity

In this case, we assume you are familiar with your configuration. Nevertheless, we would like to show you additional information in the DEVV report of Monitor II, shown in Example 7-18, where you can check, for example, for the channel and SSID configuration.

Example 7-18 Monitor II DEVV command

```
                        RMF - DEVV Device Activity                Line 1 of 5
Command ===>                                                 Scroll ===> PAGE

                        CPU=100/ 99 UIC=2540 PR=   0         System= SYS1 Total

        I=66%  DEV             ACTV RESP IOSQ -DELAY- PEND DISC CONN %D %D
  TIME   VOLSER NUM  PAV  LCU   RATE TIME TIME CMR DB  TIME TIME TIME UT RV

17:36:38 NW353F 353F      0023 132.1 15.3 10.9 0.0 0.0  0.3  0.0  4.0 53  0
17:36:39 NW353F 353F      0023 132.0 15.3 10.9 0.0 0.0  0.3  0.0  4.0 53  0
17:36:39 NW353F 353F      0023 132.1 15.3 10.9 0.0 0.0  0.3  0.0  4.0 53  0
17:36:39 NW353F 353F      0023 131.8 15.3 10.9 0.0 0.0  0.3  0.0  4.0 53  0
17:36:40 NW353F 353F      0023 131.9 15.3 10.9 0.0 0.0  0.3  0.0  4.0 53  0
```

As we also observed in the previous report, from here you can see that the largest component of response time is the IOSQ time. We can get more information about the I/O configuration using RMF reports. For example, we can find out which channel and SSID controller are being used by this volume. To obtain the list of channels used to access a volume, we get the logical control unit (LCU) number (0023). The logical control unit definition is a logical piece of a physical controller in charge of up to 256 devices.

Example 7-19 is a sample of the IOQUEUE report, which can be created by either Monitor II or Monitor III. We used Monitor III here because of its great navigational capabilities.

Example 7-19 Monitor III IOQUEUE

```
                      RMF V1R5   I/O Queuing Activity           Line 8 of 137
Command ===>                                                 Scroll ===> CSR

Samples: 60      System: SYS1  Date: 10/22/04  Time: 17.42.00  Range: 60    Sec

                   DCM Group       Cont  Del Q  AVG   CHPID  %DP  %CU   AVG AVG
Path DCM CTL Units MN MX DEF LCU   Rate  Lngth  CSS   Taken  Busy Busy  CUB CMR

7D       3500                0023                     66.90   0.7  0.0  0.0 0.0
7F       3501                0023                     75.75   0.7  0.0  0.0 0.0
                             0023   0.0   0.0   0.2  142.65   0.7  0.0  0.0 0.0
7D       3540                0024                      0.00   0.0  0.0  --- ---
7F       3541                0024                      0.03  81.8  0.0  0.0 0.0
                             0024   0.0   0.0    11    0.03  81.8  0.0  0.0 0.0
7D       3580                0025                      0.03  33.3  0.0  0.0 0.0
```

For LCU 0023, we get the channels 7D and 7F and control units 3500 and 3501. Now we can have a look at the channel activity by using the Channel Path Activity report shown in Example 7-20.

Example 7-20 Channel Path Activity Report

```
                      RMF V1R5   Channel Path Activity          Line 57 of 96
Command ===>                                                 Scroll ===> CSR

Samples: 60      System: SYS1  Date: 10/20/04  Time: 17.58.00  Range: 60    Sec

 Channel Path       Utilization(%)   Read(B/s) Write(B/s)    MSG  MSG Send Recv
ID No  G  Type  S   Part  Tot  Bus   Part  Tot  Part  Tot   Rate Size Fail Fail

79         CNC_S Y    0.0  3.1
7A         CNC_S Y    8.2 16.2
7D         CNC_S Y   49.0 58.6
7F         CNC_S Y   52.0 59.6
80      2  FC_S  Y    0.0  3.9  5.8     0  31K     0    0
81      2  FC_S  Y   10.3 11.8  9.8    6M   6M     0   3K
82      2  FC_S  Y    8.0  8.1  9.5    6M   6M     0  512
83      2  FC_S  Y    7.5  7.6  9.3    6M   6M     0  366
84      2  FC_S  Y    0.0  6.4  6.0     0  47K     0    0
85      2  FC_S  Y   11.2 13.6  9.9    6M   6M     0   5K
```

Major fields shown in this report are:

- The type of channel. In this example, two types are shown:
  - CNC_S = Switched ESCON Channel (accessing an ESCON director)
  - FC_S = Switched FICON Channel (accessing a FICON director)
- Indication whether this channel is shared with other LPs (Y or N)
- Percentage utilization by the LP (Part) and by the CPC (Tot)

The following considerations apply to FICON only:

- PART denotes the FICON processor utilization due to this LP. The FICON utilization is not a real number because these channels always respond available until 32 Open

Exchanges (concurrent I/O operations) are reached. It is calculated based in the I/O rate and its theoretical maximum capacity.

- TOTAL denotes the FICON processor utilization for the sum of all the LPs.
- BUS denotes the FICON internal bus utilization for the sum of all the LPs.
- FICON channels provide bandwidth information (MB/SEC) not available for ESCON channels. This is provided separately for READs and WRITEs since the fibre channel link is full duplex at both the logical partition level (PART) and the entire system level (TOTAL).

  The FICON processor works during channel program processing, which includes the processing of each individual channel command word (CCW) in the channel program and some setup activity at the beginning of the channel program and cleanup at the end.

  The FICON bus is busy for the actual transfer of command and data frames from the FICON channel to the adapter card, which is connected via the FICON link to the director or control unit. The FICON bus is also busy when the FICON processor is polling (mainly at low utilization) for work to be done. This is why we can see anywhere from 5% to 12% FICON bus utilization even if there are no I/Os active on that channel.

Example 7-20 shows high utilization values for channels 7D and 7F, respectively 58.6% and 59.6%.

7.3.4 ESS performance features

This section covers the performance features of the ESS including PAV, multiple allegiance, and I/O priority queuing.

Multiple allegiance

I/O device z/Architecture has defined that a state of implicit allegiance exists between a device and the channel path group that is accessing it. This allegiance is created in the control unit between the device and a channel path group (set of channels from the same z/OS system) when an I/O operation is accepted by the device. The allegiance causes the control unit to guarantee access to the device for the remainder of the channel program over the set of paths associated with the allegiance. This includes also the implementation of ESCON dynamic reconnect, stating that only the channels of the channel group are candidates to be reconnected.

This concept has been expanded to support the ESS with a concept of *multiple allegiance*. ESS's concurrent operations capability supports concurrent accesses to or from the same volume from multiple channel path groups and system images. The ESS's multiple allegiance support allows different hosts to have concurrent implicit allegiances provided that there is no possibility that any of the channel programs can alter any data that another channel program might read or write. This provision is done by not permitting several writes or one write and several reads in the same extent.

Multiple allegiance requires no additional software or host support, other than to support the ESS. It is not externalized to the operating system or operator. Multiple allegiance reduces contention reported as PEND time. Resources that benefit most from multiple allegiance are:

- Volumes that:
  - Have many concurrent read operations
  - Have a high read to write ratio
- Data sets that:
  - Have a high read to write ratio
  - Have multiple extents on one volume
  - Are concurrently shared by many users

Parallel Access Volumes

z/OS systems queue I/O activity on a unit control block (UCB) that represents the physical device. Traditionally this I/O device does not support concurrency, being treated as a single resource, serially reused. Then, high I/O activity towards the same device can adversely affect performance. This contention is worst for large volumes with numerous small data sets. The symptom displayed is extended IOSQ time. The operating system cannot attempt to start more than one I/O operation at a time to the device.

The ESS's concurrent operations capabilities now support concurrent data transfer operations to or from the same device from the same system. A device (volume) accessed in this way is called a Parallel Access Volume (PAV).

Figure 7-1 illustrates multiple allegiance and PAV allowing concurrent I/O processing.

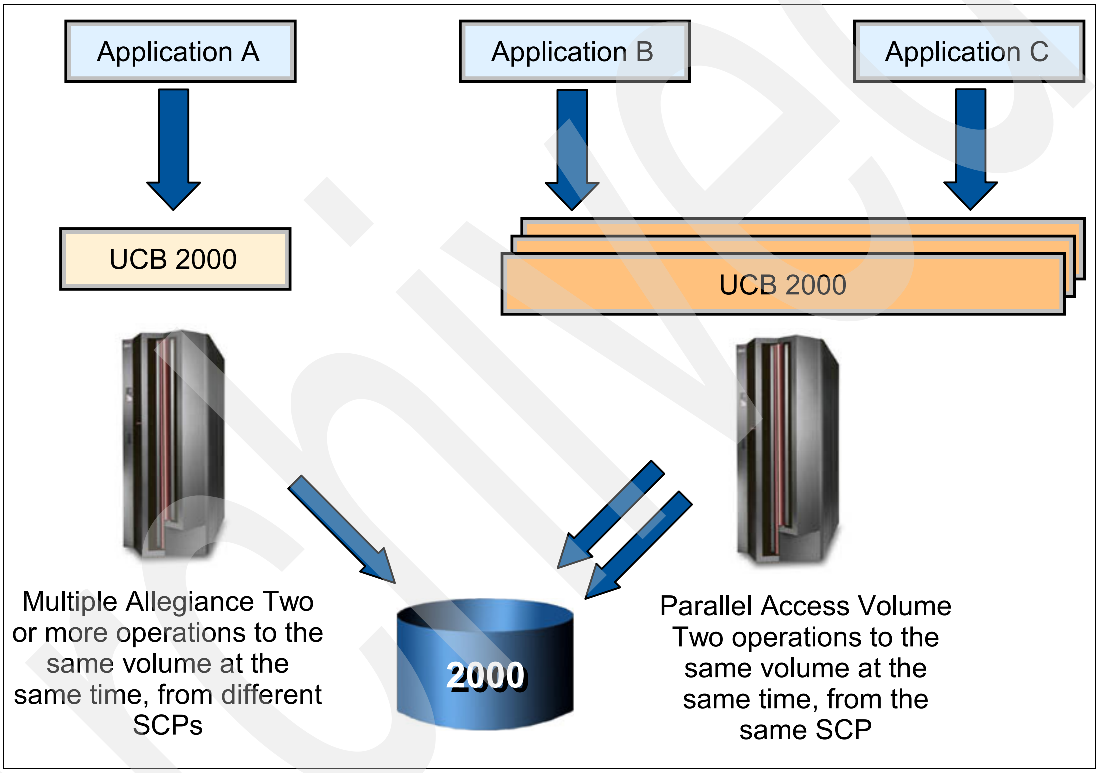

Application A
Application B
Application C
UCB 2000
UCB 2000
2000
Multiple Allegiance Two or more operations to the same volume at the same time, from different SCPs
Parallel Access Volume Two operations to the same volume at the same time, from the same SCP

Figure 7-1 Device queuing in a Parallel Access Volume environment

PAV exploitation requires both software enablement and an optional feature on your ESS. PAV support must be installed on each ESS. It enables the issuing of multiple channel programs to a volume from a single system, and allows simultaneous access to the logical volume by multiple users or jobs. Reads, as well as writes to different extents, can be satisfied simultaneously. The domain of an I/O consists of the specified extents to which the I/O operation applies, which corresponds to the extents of the same data set. Writes to the same domain still have to be serialized to maintain data integrity, and so it is for reads and writes.

The implementation of *n* parallel I/Os to the same 3390/3380 device consumes *n* addresses in the logical controller, thus decreasing the number of possible real devices. Also, UCBs are not prepared to allow multiples I/Os due to software product compatibility issues. Support is then implemented by defining multiple UCBs for the same device. The UCBs are of two types:

- Base address:

  This is the actual unit address of the volume. There is only one base address for any volume.

- Alias address:

  Alias addresses are mapped back to a base device address. I/O scheduled for an alias is physically performed against the base by the ESS. No physical disk space is associated with an alias address; however, they do occupy storage within z/OS. Alias UCBs are stored above the 16 MB line.

The workloads that are most likely to benefit from the PAV function being available include:

- Volumes that have many concurrently open data sets, for example, volumes in a work pool
- Volumes that have high read to write ratio per extent
- Volumes reporting high IOSQ times

Candidate data types are:

- High read to write ratio
- Many extents on one volume
- Concurrently shared by many readers
- Accessed using media manager or allocated as VSAM-extended format

PAVs can be assigned to base UCBs either manually by the installation or automatically by WLM. PAVs assigned manually are called static, while those movable by WLM are called dynamic. When WLM manages I/O priorities, WLM can also perform dynamic PAV management. This means that WLM can move one alias UCB from one base UCB to another base UCB in order to:

- Balance device utilizations
- Honor the goal of transactions suffering I/O delays because long IOSQ time

When in a Sysplex, all WLMs must agree with the movement of the alias’ UCBs.

I/O Priority Queuing

If I/Os cannot run in parallel, the ESS will internally queue I/Os. This reduces operating system overheads incurred by having to post “device busy” and redriving channel programs.

You also have the option to enable priority queuing of I/Os to the ESS. Priority queuing is within a sysplex and queuing is at a volume level. The ESS queues I/O requests in the order specified by WLM. I/O can be queued in the following situations:

- An extent conflict exists for a write operation.
- To allow servicing of a cache miss. Device will reconnect when data has been staged to cache.
- A reserve request is issued and other accesses are current with a different path group ID.
- Control unit is busy.

ESS queues I/O requests in the order in which they are received (FIFO). This helps to reduce problems that occur when one processor can respond to interrupts faster than a sharing one with the consequence of monopolizing a device.

Third case study

To illustrate ESS performance and, more generally, to show how improving hardware may impact significantly batch processing with heavy I/O load, we have an example running three jobs in each of the three partitions of our sysplex.

Each job reads the same data set to force IOSQ time and pending time.

- First run: The data set resides on an RVA device connected with ESCON channels.
- Second run: The data set is placed on an ESS device connected with FICON channels.

To get this information, we need to look at the Workload Activity report (Example 7-21).

Example 7-21 Workload Activity report – First run (RVA)

```
                                             W O R K L O A D   A C T I V I T Y
                                                                                                                     PAGE   1
      z/OS V1R5               SYSPLEX SYS#PLEX           START 10/26/2004-19.10.00 INTERVAL 000.20.00   MODE = GOAL
                         CONVERTED TO z/OS V1R5 RMF      END   10/26/2004-19.30.00

                                        POLICY ACTIVATION DATE/TIME 10/26/2004 19.08.07

------------------------------------------------------------------------------------------------------------ service class PERIODS

REPORT BY: POLICY=WLMPOL     WORKLOAD=BAT_WKL    SERVICE CLASS=BATHI      RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=3
                                                 CRITICAL     =NONE

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--    ---SERVICE----   --SERVICE RATES--   PAGE-IN RATES    ----STORAGE----
AVG      9.05   ACTUAL           8.54.018   SSCHRT 209.8   IOC    124926    ABSRPTN      3264   SINGLE     0.0   AVG     3614.33
MPL      9.05   EXECUTION        8.53.902   RESP    40.9   CPU      3865K   TRX SERV     3264   BLOCK      0.0   TOTAL   32703.2
ENDED      10   QUEUED                116   CONN     4.1   MSO     31353K   TCB         176.8   SHARED     0.0   CENTRAL 32703.2
END/S    0.01   R/S AFFINITY            0   DISC     0.0   SRB    100732    SRB           4.6   HSP        0.0   EXPAND     0.00
#SWAPS      0   INELIGIBLE              0   Q+PEND   9.7   TOT     35444K   RCT           0.0   HSP MISS   0.0
EXCTD       0   CONVERSION              0   IOSQ    27.1   /SEC    29536    IIT           2.7   EXP SNGL   0.0   SHARED     0.00
AVG ENC  0.00   STD DEV          2.18.152                                   HST           0.0   EXP BLK    0.0
REM ENC  0.00                                                               APPL %       15.3   EXP SHR    0.0
MS ENC   0.00

VELOCITY MIGRATION:   I/O MGMT  11.4%    INIT MGMT 11.4%

        ---RESPONSE TIME---  EX   PERF  AVG   --USING%- ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--    %
        HH.MM.SS.TTT         VEL  INDX ADRSP   CPU  I/O  TOT  I/O  CPU                          UNKN IDLE  USG  DLY  USG  DLY QUIE
GOAL                        31.0%
ACTUALS
*ALL                        11.4%  2.7   9.1   1.3  7.4 67.5 66.6  0.9                          23.5  0.4  0.0  0.0  0.0  0.0  0.0
SYS1                        12.0%  2.6   3.1   1.9  7.2 66.6 65.4  1.2                          23.3  1.1  0.0  0.0  0.0  0.0  0.0
SYS2                        11.2%  2.8   3.0   1.0  7.6 67.9 67.2  0.7                          23.5  0.0  0.0  0.0  0.0  0.0  0.0
SYS3                        11.2%  2.8   3.0   1.0  7.5 67.9 67.1  0.8                          23.6  0.0  0.0  0.0  0.0  0.0  0.0
```

From the report, we obtain the following data:

- APPL% = 15.3%
- DASD Rate = 209.8 I/O per sec
- DASD I/O Resp. Time = 40.9 ms
  - IOSQ time = 27.1
  - PEND time = 9.7
- Actual execution velocity: 11.4% (compared to goal of 31.0%)

Using the Shared DASD Activity report (Figure 7-22), we get the activity on volume NW353F in all three partitions, and we see:

- It is not a PAV volume.
- Total I/O activity rate (in the sysplex): 219 I/O per sec with a response time of 39.7 ms.

The slight differences in the reports for the I/O data can be explained with different measurement techniques for both reports.

Example 7-22 Shared DASD Activity report

```
                             S H A R E D   D I R E C T   A C C E S S   D E V I C E   A C T I V I T Y
                                                                                                                       PAGE  36
       z/OS V1R5               SYSPLEX SYS#PLEX     START 10/26/2004 - 19.10.00          INTERVAL 000.20.00
                               RPT VERSION V1R5 RMF END   10/26/2004 - 19.30.00          CYCLE  1.000  SECONDS


   TOTAL SAMPLES(AVG) = 1200.0  (MAX) = 1200.0  (MIN) = 1200.0


                          SMF            DEVICE    AVG  AVG    AVG  AVG  AVG  AVG  AVG    %      %     %     AVG
DEV  DEVICE   VOLUME PAV  SYS  IODF LCU  ACTIVITY  RESP IOSQ   CMR  DB   PEND DISC CONN   DEV    DEV   DEV   NUMBER
NUM  TYPE     SERIAL      ID   SUFF      RATE      TIME TIME   DLY  DLY  TIME TIME TIME   CONN   UTIL  RESV  ALLOC


353F 33903    NW353F      *ALL            219.196  39.7 26.1   0.0  9.2   9.5  0.0  4.0  87.16  87.36   0.0   9.0
                          SYS1  29  0023   72.423  40.0 26.3   0.0  9.4   9.7  0.0  4.0  28.74  28.93   0.0   3.0
                          SYS2  29  0023   73.264  39.5 26.0   0.0  9.2   9.5  0.0  4.0  29.26  29.27   0.0   3.0
                          SYS3  29  0023   73.509  39.5 26.1   0.0  9.2   9.5  0.0  4.0  29.16  29.16   0.0   3.0
```

Now we check the second run using a PAV volume.

Example 7-23 Workload activity report – Second run (ESS)

```
                                               W O R K L O A D   A C T I V I T Y
                                                                                                                       PAGE   1
        z/OS V1R5               SYSPLEX SYS#PLEX           START 10/26/2004-20.00.00 INTERVAL 000.19.59   MODE = GOAL
                           CONVERTED TO z/OS V1R5 RMF      END   10/26/2004-20.20.00

                                          POLICY ACTIVATION DATE/TIME 10/26/2004 19.08.07

  ------------------------------------------------------------------------------------------------------------ service class PERIODS

  REPORT BY: POLICY=WLMPOL     WORKLOAD=BAT_WKL    SERVICE CLASS=BATHI      RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=3
                                                   CRITICAL     =NONE

  TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--   ---SERVICE----   --SERVICE RATES--   PAGE-IN RATES    ----STORAGE----
  AVG      8.92   ACTUAL            1.33.022   SSCHRT  3860   IOC       2304K   ABSRPTN     51521   SINGLE     0.0   AVG     3611.22
  MPL      8.92   EXECUTION         1.31.896   RESP     1.1   CPU      60514K   TRX SERV    51521   BLOCK      0.0   TOTAL   32227.6
  ENDED     118   QUEUED                 939   CONN     0.7   MSO     487708K   TCB        2916.0   SHARED     0.0   CENTRAL 32227.6
  END/S    0.10   R/S AFFINITY             0   DISC     0.0   SRB       1196K   SRB          57.6   HSP        0.0   EXPAND     0.00
  #SWAPS      0   INELIGIBLE             186   Q+PEND   0.4   TOT     551722K   RCT           0.0   HSP MISS   0.0
  EXCTD       0   CONVERSION             168   IOSQ     0.0   /SEC     459790   IIT          20.6   EXP SNGL   0.0   SHARED     0.00
  AVG ENC  0.00   STD DEV                  0                                   HST           0.0   EXP BLK    0.0
  REM ENC  0.00                                                                APPL %      249.5   EXP SHR    0.0
  MS ENC   0.00

  VELOCITY MIGRATION:   I/O MGMT  67.2%    INIT MGMT 66.7%

          ---RESPONSE TIME---  EX   PERF  AVG   --USING%-  ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--    %
          HH.MM.SS.TTT         VEL  INDX ADRSP   CPU  I/O  TOT  I/O  CPU                          UNKN IDLE  USG  DLY  USG  DLY QUIE
  GOAL                        31.0%
  ACTUALS
  *ALL                        67.2%  0.5   9.0  21.0 33.9 26.7 13.6 13.1                          18.1  0.2  0.0  0.0  0.0  0.0  0.0
  SYS1                        69.2%  0.4   3.0  26.3 29.6 24.9 10.9 14.0                          18.4  0.7  0.0  0.0  0.0  0.0  0.0
  SYS2                        67.1%  0.5   3.0  18.4 36.3 26.8 14.3 12.6                          18.4  0.0  0.0  0.0  0.0  0.0  0.0
  SYS3                        65.6%  0.5   3.0  18.4 35.8 28.4 15.6 12.8                          17.5  0.0  0.0  0.0  0.0  0.0  0.0
```

Example 7-24 Shared DASD Activity report

```
                             S H A R E D   D I R E C T   A C C E S S   D E V I C E   A C T I V I T Y
                                                                                                                       PAGE 336
       z/OS V1R5               SYSPLEX SYS#PLEX     START 10/26/2004 - 20.00.00          INTERVAL 000.19.59
                               RPT VERSION V1R5 RMF END   10/26/2004 - 20.20.00          CYCLE  1.000  SECONDS


   TOTAL SAMPLES(AVG) = 1200.0  (MAX) = 1200.0  (MIN) = 1200.0


                          SMF            DEVICE    AVG  AVG    AVG  AVG  AVG  AVG  AVG    %      %     %     AVG
DEV  DEVICE   VOLUME PAV  SYS  IODF LCU  ACTIVITY  RESP IOSQ   CMR  DB   PEND DISC CONN   DEV    DEV   DEV   NUMBER
NUM  TYPE     SERIAL      ID   SUFF      RATE      TIME TIME   DLY  DLY  TIME TIME TIME   CONN   UTIL  RESV  ALLOC


8004 33903    PAVTS1      *ALL           4043.964   1.0  0.0   0.2  0.0   0.4  0.0  0.7  33.03  33.03   0.0   8.9
                       8  SYS1  29  00A9 1139.045   1.0  0.0   0.2  0.0   0.4  0.0  0.7   9.33   9.33   0.0   3.0
                       8  SYS2  29  00A9 1466.329   1.0  0.0   0.2  0.0   0.4  0.0  0.7  11.96  11.96   0.0   3.0
                       8  SYS3  29  00A9 1438.591   1.0  0.0   0.2  0.0   0.4  0.0  0.7  11.74  11.74   0.0   2.9
```

The results seen for the second run, shown in Example 7-23 and Example 7-24, have improved significantly without any tuning action:

- APPL% = 249.5%
- DASD Rate = 3860 I/O per sec
- DASD I/O Resp. Time = 1.1 ms
  - IOSQ time = 0.4
  - PEND time = 0
- Actual execution velocity: 67.2% (compared to goal of 31.0%)
- Number of aliases = 8

The important messages from this example are the drastically reduced DASD response times because of the new technology, which resulted in:

- Higher DASD rate: 3860 versus 210 – a ratio of 18, mainly attributable to multiple allegiance and a faster channel
- Higher APPL%: 249.5 versus 25.6 – a ratio of 10

It has to be mentioned that the reports show results of two test measurements running the same set of batch jobs in both periods. Therefore, these numbers can only be used as examples for possible improvements.

But one other value is important as a general indicator for the improvement – the execution velocity:

- Higher execution velocity: 67.2% versus 11.4% – a ratio of 5.9

This is really a significant improvement for batch processing.

7.4 Other batch-oriented problems

This section describes other problems that can be seen in the system during batch processing.

7.4.1 Duration of performance periods

Trivial transactions are the ones that consume just 20% of the resources and represent 80% of the total number of transactions (80/20 rule). Performance wise, it is recommended to protect trivial transactions (if they have the same business importance as the non-trivial ones). The rational is that you are making 80% of your users happy, with little consumption cost in your system. However, for certain workloads, you cannot predict if the new coming transaction will be trivial or not. To address this concern, service class periods are available for these types of transactions. Service class periods allow goals to be changed to less important and less challenging as the transaction uses more resources. The change of the goal is done through period migration. This migration is controlled by a service units duration value. All transactions start in the first period, where they enjoy the more important and challenging goal. The trivial transactions should complete in this period, so they should consume less than the duration of such period. The non-trivial migrate to other periods where they will be running with a less important and an easier goal. The value of the duration must be set for allowing almost 80% of the transactions to end in the first period.

Usually, in a batch environment and in environments other than CICS and IMS, it is recommended to make the short jobs run faster. You specify a goal, an importance, and a duration for a performance period. The service units that account for the duration period include CPU, SRB, I/Os, and storage. The recommended SDC values in order to implement service class periods (duration) is to make the MSO (storage coefficient) equal to zero. This makes the duration service units metric more predictable and repetitive.

A quick way to verify the correctness of duration is to get the percentage of jobs ended during the first or second period in the WGPER section of the Workload Activity report using the following statement:

```
SYSRPTS(WLMGL(WGPER))
```

Example 7-25   Workload Activity report (service class period)

```
REPORT BY: POLICY=WLMPOL     WORKLOAD=BAT_WKL    SERVICE CLASS=BATPIER    RESOURCE GROUP=*NONE
                                                    CRITICAL     =NONE
                                                    DESCRIPTION  =test   priority batch

   TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--   ---SERVICE----   --SERVICE RATES--   PAGE-IN RATES    ----STORAGE----
   AVG      1.99   ACTUAL             4.51.217   SSCHRT 274.9   IOC     81950    ABSRPTN     25724   SINGLE     0.0   AVG     3990.80
   MPL      1.99   EXECUTION          2.25.229   RESP     0.9   CPU      3133K   TRX SERV    25724   BLOCK      0.0   TOTAL   7954.70
   ENDED       8   QUEUED                  891   CONN     0.6   MSO     27509K   TCB         151.0   SHARED     0.0   CENTRAL 7954.70
   END/S    0.01   R/S AFFINITY              0   DISC     0.0   SRB     40044    SRB           1.9   HSP        0.0   EXPAND     0.00
   #SWAPS      0   INELIGIBLE         2.25.096   Q+PEND   0.3   TOT     30765K   RCT           0.0   HSP MISS   0.0
   EXCTD       0   CONVERSION              131   IOSQ     0.0   /SEC    51274    IIT           0.6   EXP SNGL   0.0   SHARED     0.00
   AVG ENC  0.00   STD DEV                   0                                   HST           0.0   EXP BLK    0.0
   REM ENC  0.00                                                                 APPL %       25.6   EXP SHR    0.0
   MS ENC   0.00

   PER IMPORTANCE  PERF     --TRANSACTIONS--     -------------RESPONSE TIME--------------   -EX VEL%-   -CPU--  -EXE--
                   INDX     -NUMBER-     -%-     ------GOAL------  ---ACTUAL---     TOTAL   GOAL  ACT   USING%  DELAY%
   1   4            1.5            4      50                                                  30 20.0    10.5    42.1
   2   DISC                        4      50      DISCRETIONARY                                  26.0    11.3    66.4
 TOTAL                             8     100
```

In Example 7-25, we can see the number and the percentage of transactions ended during the period. So, if the ratio is not what you expect, you should revisit the definition of the period of the service class. In this case the ratio is 56%, so the duration should be increased.

> **Attention:** Batch jobs updating databases (DB2, IMS) may not be good candidates for using service classes with periods. If the last period is discretionary or of very low importance, a job maintaining locks may cause a global contention in the batch workload.

7.4.2 Capping delays

Capping is used to limit, artificially, the CPU consumption rate of a specific set of dispatchable units, usually associated with user transactions.

Among the reasons for capping are the following:

- In a service bureau, where a company customer should not consume more than what they are paying for
- To limit I/O access from a less important workload to a common I/O controller
- To save software fees in a WLC software contract
- To protect the most trivial transactions against z/OS non-full preemptability
- To emulate a Quiesce action

There are different types of capping, as follows:

- LPAR capping defined through weights or defining in an LP with less logical CPUs than the number of the physical CPUs.
- WLM Resource Group capping.

- Soft capping defined with LPAR and WLM for WLC.
- Discretionary capping (a hidden capping), to cap happy service classes (with PI less than 0.81and velocity goal less than 30%). The purpose is to allow discretionary goal transactions to get some CPU in a heavy CPU loaded sysplex.

In this section, we cover the WLM Resource Group capping that might be the potential cause of capping delays for a service class.

Definition of resource groups

You can use a resource group WLM construct to limit the amount of processing capacity across the sysplex available to one or more service classes, or to set a minimum protection processing capacity for one or more service classes in the event that the work is not achieving its goal. This amount of capacity is specified in unweighted CPU service units not multiplied by SDC.

The policy page of the Workload Activity report contains a section named Resource Groups with the list for each group:

- Associated service class in the resource group
- Amount of CPU service units actually consumed during the interval; the number of these service units is rounded up or down to the nearest 1K
- Capacity defined in the resource group (min and max)

Example 7-26 Workload Activity report (Resource groups)

```
RESOURCE GROUPS
    --NAME--    ----------DESCRIPTION-----------     -SERVICE-      ACTUAL ---CAPACITY---
                                                       CLASS      CONSUMED  MIN      MAX
    RGPIER                                                              2K          2000
                                                     BATPIER            2K
```

Example 7-26 shows that service class BATPIER belongs to resource group report RGPIER. Let’s have a look at the service class report for this capped class.

Example 7-27 Workload Activity report (Capped service class)

```
  REPORT BY: POLICY=WLMPOL     WORKLOAD=BAT_WKL    SERVICE CLASS=BATPIER    RESOURCE GROUP=RGPIER     PERIOD=1 IMPORTANCE=4
                                                   CRITICAL     =NONE

  TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--   ---SERVICE----   --SERVICE RATES--   PAGE-IN RATES    ----STORAGE----
  AVG      8.41   ACTUAL          21.11.012   SSCHRT  81.1   IOC      24100   ABSRPTN      2059   SINGLE     0.0   AVG     3901.37
  MPL      8.41   EXECUTION        9.55.970   RESP     1.0   CPU      1056K   TRX SERV     2059   BLOCK      0.0   TOTAL   32795.2
  ENDED       6   QUEUED              3.078   CONN     0.6   MSO      9287K   TCB          48.3   SHARED     0.0   CENTRAL 32795.2
  END/S    0.01   R/S AFFINITY            0   DISC     0.0   SRB      18827   SRB           0.9   HSP        0.0   EXPAND     0.00
  #SWAPS      0   INELIGIBLE      11.11.962   Q+PEND   0.3   TOT     10386K   RCT           0.0   HSP MISS   0.0
  EXCTD       0   CONVERSION            349   IOSQ     0.1   /SEC     17310   IIT           0.4   EXP SNGL   0.0   SHARED     0.00
  AVG ENC  0.00   STD DEV                 0                                   HST           0.0   EXP BLK    0.0
  REM ENC  0.00                                                               APPL %        8.3   EXP SHR    0.0
  MS ENC   0.00

  VELOCITY MIGRATION:   I/O MGMT   2.1%    INIT MGMT  2.1%

         ---RESPONSE TIME---  EX   PERF  AVG   --USING%- ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--    %
         HH.MM.SS.TTT         VEL  INDX ADRSP   CPU  I/O  TOT CAPP  CPU  I/O                     UNKN IDLE  USG  DLY  USG  DLY QUIE
  GOAL                       30.0%
  ACTUALS
  SYS1                        2.1% 14.3   2.8   1.0  1.0 91.1 75.8 14.8  0.4                      6.9  0.0  0.0  0.0  0.0  0.0  0.0
```

In Example 7-27, service class BATPIER gets a very bad actual velocity of 2.1% due to capping: EXECUTION DELAYS CAPP = 75.8%.

Using the Monitor III **RG** command, you can also get information about resource groups. (The report is a modification of the Sysplex Summary report, so both reports have the same title line.)

Example 7-28 Resource Group report

```
                     RMF V1R5   Sysplex Summary - SYS#PLEX         Line 1 of 4
Command ===>                                                  Scroll ===> CSR

WLM Samples: 293     Systems: 3  Date: 10/22/04 Time: 11.32.00 Range: 60    Sec

                      >>>>>>>>------------------<<<<<<<<
Service Definition: WLMDEF                Installed at: 10/22/04, 11.16.39
     Active Policy: WLMPOL                Activated at: 10/22/04, 11.16.47

               Exec Vel  Resp. Time            Resource  ----- Capacity -----
Name     T  I    Goal       Goal     Duration    Group     Min    Max    Act

BAT_WKL  W
BATHI    S  3     31                                        N/A    N/A  18632
BATLO    S  D                                               N/A    N/A 8265.9
BATPIER  S  4     30                           RGPIER       N/A   2000 1256.6
```

If you detect some CAPP delay in the Execution Delays of a service class, you should also check for resource group utilization. Remember, it is a sysplex-wide counter.

Discretionary capping (overachieving goals)

There is another possibility for suffering capping delays even when there is not a defined resource group.

Example 7-29 Workload Activity report – Capping delays

```
                                              W O R K L O A D   A C T I V I T Y
                                                                                                                     PAGE   2
      z/OS V1R5               SYSPLEX SYS#PLEX           START 10/18/2004-10.00.00 INTERVAL 000.09.59   MODE = GOAL
                         CONVERTED TO z/OS V1R5 RMF      END   10/18/2004-10.09.59

                                        POLICY ACTIVATION DATE/TIME 10/18/2004 08.31.03

REPORT BY: POLICY=WLMPOL     WORKLOAD=BAT_WKL    SERVICE CLASS=BATHI      RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=3
                                                 CRITICAL     =NONE

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--   ---SERVICE----   --SERVICE RATES--   PAGE-IN RATES    ----STORAGE----
AVG      3.10   ACTUAL            2.40.822   SSCHRT 729.6   IOC    217479    ABSRPTN     44442   SINGLE     0.0   AVG     3988.64
MPL      3.10   EXECUTION         2.40.067   RESP     0.8   CPU      8416K   TRX SERV    44442   BLOCK      0.0   TOTAL   12382.4
ENDED      12   QUEUED                 667   CONN     0.6   MSO     74033K   TCB         405.6   SHARED     0.0   CENTRAL 12382.4
END/S    0.02   R/S AFFINITY             0   DISC     0.0   SRB    112054    SRB           5.4   HSP        0.0   EXPAND     0.00
#SWAPS      0   INELIGIBLE              87   Q+PEND   0.3   TOT     82779K   RCT           0.0   HSP MISS   0.0
EXCTD       0   CONVERSION             174   IOSQ     0.0   /SEC   137978    IIT           1.8   EXP SNGL   0.0   SHARED     0.00
AVG ENC  0.00   STD DEV             16.589                                   HST           0.0   EXP BLK    0.0
REM ENC  0.00                                                                APPL %       68.8   EXP SHR    0.0
MS ENC   0.00

VELOCITY MIGRATION:   I/O MGMT  42.4%    INIT MGMT 42.2%

        ---RESPONSE TIME---  EX   PERF  AVG   --USING%- ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--     %
        HH.MM.SS.TTT         VEL  INDX ADRSP   CPU  I/O  TOT CAPP  CPU  I/O                     UNKN  IDLE USG  DLY  USG  DLY QUIE
GOAL                        30.0%
ACTUALS
SYS1                        42.4%  0.7   1.0  15.4 17.0 44.1 28.1 11.3  4.8                     22.2  1.3  0.0  0.0  0.0  0.0  0.0
```

In Example 7-29, service class BATHI has a capping delay of 28.1%.

- The service class is not defined with a resource group: `RESOURCE GROUP =*NONE`.

- The explanation is in the definition of the goal: a velocity goal of 30% has been defined.

  If this service class is overachieving its goal (happy with PI less than 0.81), WLM caps jobs running in this service class in order to give CPU resources to jobs running as discretionary service classes.

You have to consider whether you want this effect for this service class. If you want to avoid such capping, make the batch service class point to a Resource Group without minimum and maximum values.

7.5 Batch performance

This section provides information about general performance considerations for batch processing. These topics are not related directly to the monitoring functions of RMF; they are discussed here to give you a wider perspective of performance-related issues.

7.5.1 Batch window

When talking about batch performance, there are two areas that we need to discuss:

- The elapsed time of a single job
- The time which is required to process all batch jobs belonging to the same production workload

In the first part of this chapter, we discussed problems that batch jobs can have with their elapsed time. Various system components, resources, and definitions are contributing factors. We have shown, with the help of some examples, how we can analyze the reasons for some problems and define possible solutions.

But performance is not always measured in seconds and minutes, it can also be a subject of hours. As an example for a typical application of this scope, we discuss the workload of a banking company.

During the day, online banking is the key application. Thousands or millions of transactions need to be processed, with an optimal performance, of course. Typically, these transactions are not running 24 hours a day, but only during a time window defined by the bank, for example, from 7:00 a.m. to 8:00 p.m. Of course, this window is enlarged in times of globalization, with a target of 24 hours, but since this just complicates the discussion we do not consider it here.

All or at least most online transactions that have been performed during the day need some further processing later on by batch jobs. It is required that this batch processing be completed by the next morning so that online processing can be started again with the most current data. That means that there is a time range from 8:00 p.m. until 7:00 a.m. available for all jobs. This is called a *batch window*.

Therefore, it is not only sufficient that each job in this more or less complicated net of jobs be processed as fast as possible, but also the entire net of jobs needs to be completed at a well-defined point in time. On the other hand, maybe not all jobs need to be finished within the batch window. For example, jobs that print listings for all bank accounts may run longer because they have no impact on the online processing. This has to be analyzed very carefully when setting up the batch net.

Optimizing the net is not a one-shot task: typically, applications are growing, requiring more and more resources to be produced. Each bank is interested in getting more customers, creating more transactions/products, and they are interested in providing better service by extending the online times. This results in the task for the IT shop to process a bigger load in a shorter batch window – really a challenge.

What solutions are possible to solve the problem, assuming that the net has been squeezed and optimized and does not allow further reduction of the run time of all jobs? Hopefully the discussion about this topic takes place before it is not too late.

One solution to get everything done faster is to install faster processors. But does this help, and how much?

We want to assume that considerations are going on in the IT shop to install a processor with a doubled speed compared to the current system. Based on the descriptions in 5.2.3, “Processor performance metrics” on page 160, this means there is an ITR ratio of 2 between both processors for our bank workload. This brings us to the question: “Is it possible to process twice as many transactions or to reduce the elapsed time of batch jobs by 50% with the new processor?”

The answer is: “It depends, probably not.”

To get the correct answer for that question, we have to consider that the elapsed time for the entire net depends mainly on the following aspects:

- Elapsed time of each single job.
- Dependencies within the net and possible wait times (for example, JOBB can run only when JOBA has finished).
- Classification of the batch Jobs in one or more service classes. Critical path Jobs should be classified in a service class with a higher importance.

The single job

First, we have a look at a single job. Its elapsed time is given primarily by the CPU time and the I/O time (if we neglect CPU and I/O wait times, which might be reduced by the adequate choice of a service class with an important and difficult goal).

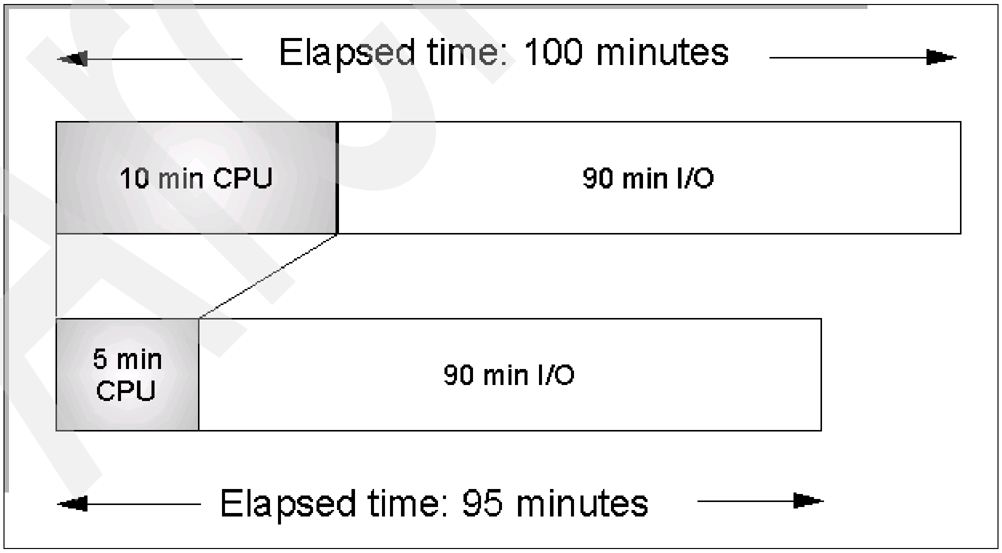


Figure 7-2 Reduction of job elapsed time

Figure 7-2 shows a job that needs 10 minutes CPU time and 90 minutes I/O time, resulting in 100 minutes elapsed time. When the job can run on the faster processor, we can expect that the CPU time is reduced by about 50%, resulting in 5 minutes CPU time with an overall elapsed time of 95 minutes.

- The result of the double-speed processor is an improvement for this one job of 5%.

Of course, this does not mean that the new processor has little advantage for the IT shop, but it has only limited advantage for this one specific job. It offers a lot more capacity for running more jobs in parallel; for installations with capacity constraints this helps significantly in decreasing the CPU wait time and increasing the ended jobs per second rate.

This means that other actions have to be performed to get the job running faster. In 7.3.4, “ESS performance features” on page 234, we have seen the potential for improvements of the I/O time and its impact on the job elapsed time. You can reduce it significantly by exploiting the new DASD technology; nevertheless, it takes time to process all I/O activities in the application.

These considerations apply to jobs with a high percentage of I/O time. There are other types of jobs, for example, DB2 batch jobs (with many hits in the DB2 buffer pool) or number-crunching applications, for which the elapsed time is very close to the CPU time (in an unconstrained environment). These jobs experience a significant reduction of their elapsed time when running on a faster processor.

The fastest I/O is the zero I/O. Each I/O that you can eliminate reduces the run time of the job. Therefore, we discuss this topic in 7.5.2, “Exploiting Data-in-Memory” on page 246.

The net of jobs

The other task that needs to be done is a detailed analysis of the net and the relationship among all jobs in the net. You have to define the job flow in a way that as many jobs as possible can run in parallel, depending on the CPU capacity of your system. With respect to the prior discussion about installing double-speed processors, this duplicates your CPU capacity whether or not your system is experiencing latent demand.

It is obvious that not all jobs can run in parallel because of dependencies among them. For instance, if one job writes data and another job has to read this data, the second job has to wait until the first job has completed. But it depends: it might be possible for both jobs to run nearly in parallel by using the capabilities of batch pipes. We discuss this in 7.5.3, “Improving performance using BatchPipes®” on page 247. Another important approach to consider is to implement VSAM data sharing (RLS) between jobs, even within the same z/OS system, allowing several jobs to read and write in the same cluster with total integrity and parallelism.

Knowledge of the application and an analysis of the net are required to fully understand the dependencies among all jobs. This should lead you to the result of being able to answer, for each job, the two questions:

- Which jobs need to be completed before this job can run?
- Which jobs can run when this job has completed?

With this analysis, you can define the *critical path* of your net.

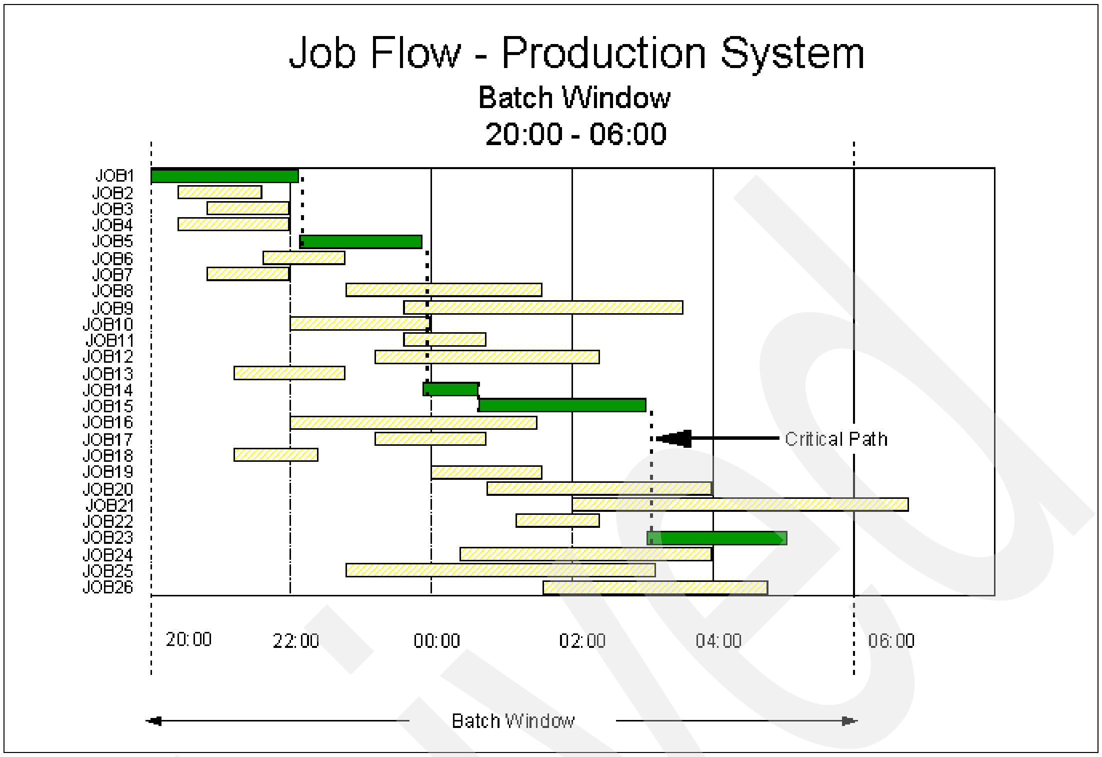


Figure 7-3 Batch Window – Critical Path

Figure 7-3 shows the job flow of a production environment with 26 jobs running during the time frame of the batch window. The analysis has resulted in the following jobs defining the critical path:

JOB1 → JOB5 → JOB14 → JOB15 → JOB23

Now you know the jobs that need the highest attention.

There is no improvement for the batch window from techniques like reducing the elapsed time of a long-running print job, which has no dependencies for other jobs in the system. You must concentrate on the jobs in the critical path, answering the following questions about them:

- Are there any possibilities to reduce the elapsed time of each of the jobs?
- Are there any possibilities to run some of these jobs in parallel?

Having some positive answers to these questions helps you improve the run time of the net – providing new capacity in the batch window for the future growth of your application. The following two sections explain some technologies which can be useful in this context.

7.5.2 Exploiting Data-in-Memory

Data-In-Memory (DIM) techniques enable individual jobs (or a group of chained jobs) to run faster by reducing the I/O component of the elapsed time for the job. The reduction is achieved by reading the data from (or in some cases writing to) a buffer in processor storage instead of tape or DASD. There are several benefits from using these techniques:

- The number of I/Os is reduced, and therefore the I/O component of elapsed time is reduced proportionately.

- The remaining I/Os are made to a less busy I/O subsystem, which gives the remaining I/Os a faster response. The CPU is no longer used for driving I/Os.
- The CPU is no longer used for pre-processing CLISTs or LOAD modules.
- Some CPU cycles are saved as a result of less contention and a shorter transaction residency time.

The following sections briefly describe a few of the functions or products that implement the DIM techniques in a batch environment. You can find a more detailed description in the *RMF Performance Management Guide*, SC33-7992.

Batch LSR subsystem

The Batch LSR subsystem (BLSR) extends the benefits of VSAM Local Shared Resources (LSR) to applications written to use VSAM Non-Shared Resources (NSR), typically programs written in high-level languages. VSAM LSR allows a program processing a VSAM data set to buffer frequently used data in processor storage, eliminating read I/Os and reducing the elapsed time of an individual job. VSAM NSR does not provide this kind of buffering.

System management buffering (SMB)

Today, the recommendation is to use the VSAM standard function system management buffering (SMB) introduced in DFSMS V1R4 as an alternative to BLSR. SMB is available for NSR processing. SMB enables VSAM (at Open time) to determine the optimum number of index and data buffers, as well as the type of buffer management: LSR or LRU algorithm or NSR or sequential algorithm.

DFSORT™ hipersorting

Many large sorts (particularly the larger ones in the batch window) require work space to hold intermediate data. Typically this is held on DASD (although tape and VIO are also supported). As much as 20% of a large sort’s elapsed time might be spent performing I/O to work data sets. DFSORT hipersorting allows part or all of the sort work space to be allocated in a hiperspace. This approach leads to a reduction or elimination of sort work I/O and a consequent reduction in elapsed time for the sort.

Hiperbatch™

Frequently, some data sets are read sequentially several times during a batch window. These data sets may be written during the batch window and then subsequently read. Typically, installations have found that these multiply-accessed data sets produce performance bottlenecks because of DASD contention. Traditional methods of reducing this effect have been to:

- Duplicate the data set over several DASD volumes or even controllers.
- Schedule each of the jobs and job steps so that they access the data at different times.
- Copy the data set (if it is small) to a VIO data set (using paging I/O instead of traditional I/O).

Hiperbatch provides a set of services that can provide greater benefits than these techniques and without many of their associated problems.

7.5.3 Improving performance using BatchPipes®

BatchPipes for OS/390 is a z/OS software product that began life as BatchPipes/MVS, announced April 1994 (Product ID 5655-A17). After a second release in September 1995, it was replaced in March 1997 by SmartBatch for OS/390, which includes BatchPipes functionality plus some BMC software. Then, in April 2000, SmartBatch was withdrawn, and a few days later, Version 2 of BatchPipes for OS/390 was announced. Currently, BatchPipes uses the Coupling Facility to pipe data between jobs running on different systems in a Parallel Sysplex.

BatchPipes for OS/390 offers a way to connect jobs so that data from one job can move through central storage to another job without going to DASD or tape. It addresses a growing problem that faces installations with batch workloads: insufficient time to complete that work. Given a batch job stream with a data flow of certain characteristics, BatchPipes can dramatically shorten the elapsed time of the job stream. The jobs can run faster and process larger volumes of data in the available batch window; your processors are freed to run more work—perhaps extending the period of time that interactive applications are available or supporting your company's work in another time zone. In short, you might get more work from your processors.

BatchPipes, running on z/OS:

- Allows two or more jobs that formerly ran serially to run concurrently.
- Reduces the number of physical I/O operations by transferring data through processor storage rather than transferring data to and from DASD or tape.
- May reduce tape mounts and use of DASD.

What difference does parallel processing make to my batch workload? Large, tightly-coupled processing systems running with z/OS can have many jobs in various stages of processing at one time. Batch applications have not traditionally taken advantage of this multiple processing capability. Rather, they often consist of multiple jobs that run one after another. With BatchPipes, the processor can run two or more jobs in a job stream at one time. Jobs that once ran sequentially can now run in parallel because data records or blocks are available to the next job immediately after they are written. That is, the whole sequential file does not have to be written and then closed before the next job can access the file.

What difference does keeping data in the processor storage make to my batch workload? When data moves from one job to another through processor storage, the transfers take place in microseconds as compared to the milliseconds required to transfer data to and from tape or DASD. Keeping data in processor storage reduces the number of physical I/O operations, causes less I/O contention, and frees the device that holds the intermediate data set. An additional benefit of transferring data through processor storage is a reduction in the mounting and managing of tapes for applications using BatchPipes.

7.5.4 VSAM Record Level Sharing (RLS)

VSAM RLS is another mode of managing buffer pools, which allows any number of users within your parallel sysplex to share your existing VSAM spheres. It provides full data integrity (read and write). The serialization is at record level. However, to implement recoverable spheres, the user must have its own backout logging mechanism. VSAM RLS does not introduce new types of VSAM clusters; rather, it introduces a new way of accessing existing data sets. Apart from the need to open data sets in RLS mode, the same VSAM record management interfaces (get, put, point, erase) are used.

RLS allows concurrent access, in a sysplex, to your VSAM data sets at record level, while maintaining data integrity through the use of coupling facility structures.

You can access VSAM sphere in RLS mode by having users run also in one z/OS image. However, even in such an environment (monoplex) a coupling facility is required to exploit the serialization mechanism provided by RLS. For example, in a non-RLS mode, if two jobs are trying to access for update a cluster with total read and write integrity, that is, SHAREOPTIONS (1,3), just one can open the cluster. The other needs to wait for the first to close the cluster. The granularity of the serialization is at cluster level. If the two jobs access the cluster in RLS mode, then both the OPENs succeed, and both in parallel can update the cluster.

7.5.5 WLM batch initiator

With this technique WLM reduces batch delay time for initiators. This aim is reached by allowing WLM to start and stop initiators. The decision is taken based on the batch service class goal being not obtained and the major reason being the delay by initiators.

8

Monitoring transactional workloads using RMF

In this chapter we describe how RMF can be used to monitor transactional workloads, using CICS as our example. We explain the relationship between CICS and WLM, and we discuss how to make the best use of the RMF facilities to investigate a perceived performance problem affecting a transactional workload.

8.1 Introduction

With the introduction of the Workload Manager, a z/OS system can manage workloads according to user-specified goals. This includes goals not only for traditional workloads such as TSO, batch, and started tasks, but also goals for CICS and IMS transactions. Usage of specific Workload Management API services by CICS or IMS, such as Execution Delays Monitoring services, allows WLM to know how well transactions are executing, and where any delays are occurring. This provides measurements that can be reported by RMF and lets us determine whether goals for transactions are being met, both at a single system level and at a sysplex level.

Since there is no difference, as far as performance monitoring with RMF is concerned, between these two subsystems, we concentrate on CICS as a sample throughout this chapter.

8.2 Monitoring CICS workload

Before we can start monitoring CICS activities with RMF, it is important to understand the relationship between CICS and WLM.

8.2.1 CICS and WLM

Starting with CICS/ESA® (Version 4.1) and assuming WLM is running in goal mode, a CICS address space (commonly called a region) can be managed by WLM in two ways:

- Transaction mode: The region is managed by the performance goal of the transactions the region is processing. A response time goal is assigned to the transactions using CICS classification rules. This mode has the following properties:
  - The priorities (I/O and CPU) of the CICS address spaces are determined by the needs of the most important, most unhappy (PI > 1.0) transactions running in those address spaces. In this environment, there are changes of free rides, where less important happy transactions running in the same address spaces may have a lift in their performance.
  - Only average response time and percentile response time can be specified as a goal.
  - CICS subsystem is used in classification rules to associate with a service class.
  - Periods or resource group capping are not allowed.
  - Subsystem work manager delays are reported by RMF. Refer to “Workload Activity report: Transaction view” on page 257.
  - By defining special report classes, you can get RMF reports that cover transaction response time and transaction rates.
- Region mode: The region is managed by the performance goal of the service class assigned to this region, which is a velocity goal. That is, it is not managed as a transaction server. This mode has the following properties:
  - The priorities (I/O and CPU) of the CICS address spaces are determined by the PI of those address spaces. These address spaces are managed as normal STCs. Installation must manually cluster transactions by business importance in the same CICS regions.
  - Only execution velocity goals are possible.
  - Good RMF reports cover transaction response time and transaction rates using special report classes.

  - STC (or JES if a batch Job) subsystem is used in classification rules to associate with a service class.
  - Periods or resource group capping are not allowed.
  - Subsystem work manager delays are reported by RMF. Refer to “Report class enhancement” on page 254 for details.

It is possible to mix these two modes: region mode for the CICS address spaces and transaction mode for the transactions running in those address spaces. However, the region mode goal is used during address space initialization, until the appearance of the first transaction in such address space. From this moment on, the transaction mode goal rules the address space priorities.

Transaction mode versus region mode

There are several differences between these modes with respect to performance management and performance monitoring:

- You get better management of CICS regions in a constrained environment with transaction mode.
  - As the goal of a service class is expressed in response time, the continuous monitoring by WLM of this response time makes it easier to detect a degradation. In region mode, WLM is not aware of the response time degradation, but only of a drop in the execution velocity, which may not reflect the level of this degradation.
  - In a CICSplex environment, WLM is able to detect exactly which regions need resources and to make the adjustments.
- A velocity goal value (for regions in region mode) is very difficult to determine and may vary with the speed or the number of processors (it should be revisited after a hardware upgrade) and the same service class can be used in a sysplex environment on different systems.
- When using RMF, for regions with a response time goal you get meaningful measurement data such as response time averages and distributions. In a single report, you can obtain global CPU utilization, resource consumption of CICS regions, response times, and number of transactions for the same interval. This may be very valuable data for capacity planning.
- Some specific environments still need to be managed as regions. Typically, test regions may not benefit from management in transaction mode. If the rate of transactions is too low, WLM cannot work efficiently, and with some development tools (such as step-by-step execution and traces), response time is not meaningful.

A general consideration: If there are too many service classes for CICS or IMS with response time goals, the management of these transactions can become a little unpredictable. This is because WLM does not manage the transactions, but rather it manages the regions based on the mix of transactions that run in these regions. If this mix is too diverse, or even if there are just a few service classes with response time goals, but some of them with little activity, the mix can result in management which can't be understood from just looking through monitors. This is one of the things you might want to consider if response time goals for CICS and IMS are the best choice for your installation. Execution velocity goals may have their downside, but they are much easier to correlate to what is running on the system.

WLM does not control directly each classified transaction, but only the regions for the resources managed, such as CPU, processor storage i/o priority, and alias; too much granularity for transaction service classes can be counterproductive. The best results can usually be obtained when similar response time transactions are mapped to the certain regions, but in some installations this might be hard to achieve.

Report class enhancement

With the full reporting scope provided by the report class enhancements in z/OS V1.2, you can now gather response time distribution data and subsystem work manager delay data for CICS and IMS transactions while the region is still managed towards velocity goals.

This is very useful, for example, in a migration or setup scenario when you want to go from execution velocity goals for the CICS environments to response time goals: in this case, this allows your installation to set it up and collect data while maintaining the current existing management definitions.

Nevertheless, to produce a response time distribution, a response time goal is needed. To solve that problem an artificial response time goal specified in the transaction's service class is used.

For example, you might have the service class CICSTEST defined in the subsystem STC with a Velocity goal applied to the CICS regions. On top of that, in the subsystem CICS, one service class SCATM (with a fake goal) and a report class RCATM are defined. For these CICS regions, SCATM is the transaction service class that is just used as a reporting vehicle for subsystem work manager delay data and as the master for the report class RCATM from which to show transaction data. RMF reports resource consumption data and general execution delay data for the region service class, that is, CICSTEST and response time data as well as subsystem work manager delay data for the report class RCATM. With z/OS V1.2, response time distribution information is reported also, with RCATM (provided such a report class is homogeneous within RMF's reporting interval). SCATM is not reported by RMF.

How does the transaction management work

The objective of the transaction server management facility is to assign goals to the transactions and let the system determine which regions need the resources (and how much) to meet these goals.

During WLM Service Definition setup, you have to define:

- Service classes for the transactions with a goal response time (typically average response time with percentile).

  Example: service class CICSTRN

  – Importance: 2
  – Percentile: 85%
  – Average response time: 0.250 sec (85% of the transactions should complete with a response time between 0.000 and 0.250 secs)

- A service class for the CICS region with a velocity goal.

  Example: service class CICSRGN

  – Importance: 2
  – Velocity: 60

- CICS rules to assign transactions to service classes. Since the classification is done in the terminal owing region (TOR), if the APPLID is used as qualifier to classify the transaction, you must specify the APPLID of the TOR. When a transaction is attached, CICS asks WLM to classify the transaction using the CICS subsystem classification rules.
- Either transaction or region in the WLM application in the STC or JES classification rules (under the field “managed regions to goals of”). The default is transaction; you should specify REGION if you do not want transaction management, but only the reporting capability, and you have the CICS rules defined.

When in transaction mode, WLM builds a topology of the transaction, where a transaction begins in the TOR, executes in an application-owning region (AOR), may be in a file-owning region (FOR), and terminates in the TOR.

Each minute, WLM enters a *topology* routine to identify which regions are serving which transaction service classes and to group similar regions into dynamic *internal service classes*. These service classes are used to manage the address spaces priorities. However, with SDSF, we can still see the *external service class* in front of the STCNAME or JOBNAME and for each address space, we observe its current status:

Server = YES	Transactions are being processed
Server = NO	No transactions are being processed
Server = N/A	This address space is region-managed

Tip: If you have not installed SDSF, you can get this information by using the D A,jobname command.

Example 8-1 SDSF display

```
SDSF DA ZO    JBO      PAG    0 SIO     0 CPU 100/ 61  LINE 10-28 (121)
COMMAND INPUT ===>                                            SCROLL ===> CSR
NP   JOBNAME  SrvClass DP Server   SysName  SP ResGroup Quiesce  ECPU-Time  ECP
     CICSCTBA CICSRGN  F7 NO       JBO       1                        4.34   0.
     CICSCWBA CICSRGN  F1 YES      JBO       1                       72.04   0.
     CICSCWBB CICSRGN  F7 NO       JBO       1                       63.97   0.
     CICSCWBC CICSRGN  F7 NO       JBO       1                       73.25   0.
     CICS1ABA CICSRGN  FD YES      JBO       1                       85.47   0.
     CICS2ABA CICSRGN  FD N/A      JBO       1                      200.66   0.
     CICS2ABB CICSRGN  FD YES      JBO       1                      195.69   0.
     CICS2ABC CICSRGN  FD YES      JBO       1                      206.14   0.
     CICS2FBA CICSRGN  F1 YES      JBO       1                     2029.60   2.
```

In Example 8-1, you find the server status NO for three address spaces. You can also notice that there are different dispatching priorities (DP) for the same service class address spaces. This looks contrary to the facts that you might have learned in your last WLM class. You have to consider that an address space with a server status YES is managed by an internal service class, and different DP values are the only externalization of this internal service class.

Important: In transaction mode, the goal of a service class for the regions as specified in the STC and JES classification rules is used only during the starting phase of the region, the ending phase of the region, or when the region is idle for more than one minute. Service and delays are reported for this service class.

When we monitor a CICS workload, we have two views to consider:

- Monitoring transactions (transaction view)

  Each service class is associated with a group of transactions through the CICS classification rules, with a response time goal. In the reports, we get the number of transactions executed, the response time, and the distribution of this response time. In fact, several service classes are necessary to classify transactions.

- Monitoring address spaces (server view)

  Another service class, associated with one or more address spaces. The reports show the service consumed and some performance data (performance index, delays).

Of course, there are several reports required and available in RMF when we want to monitor a CICS workload. But many of them, for example, the CPU Activity report, have been discussed in previous chapters and are not repeated in this chapter. We concentrate here on the Workload Activity report and some Monitor III reports.

8.2.2 Using the Workload Activity report

We start with the *server view*, that is, the report of the service class CICSRGN that is assigned to all CICS regions.

Example 8-2 Workload Activity report (server view)

```
REPORT BY: POLICY=WLMPOL01   WORKLOAD=CICS       SERVICE CLASS=CICSRGN    RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=2
                                                     CRITICAL     =NONE

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--   ---SERVICE----   --SERVICE TIMES--   PAGE-IN RATES    ----STORAGE----
AVG     78.99   ACTUAL        13.41.21.297   SSCHRT  8746   IOC    113899K   TCB        4234.4   SINGLE     0.0   AVG     17763.3
MPL     78.99   EXECUTION     13.41.21.297   RESP     0.9   CPU    616172K   SRB        2311.8   BLOCK      0.0   TOTAL   1403209
ENDED       2   QUEUED                   0   CONN     0.5   MSO      3581M   RCT           0.0   SHARED     0.0   CENTRAL 1403209
END/S    0.00   R/S AFFINITY             0   DISC     0.0   SRB    340549K   IIT         135.6   HSP        0.0   EXPAND     0.00
#SWAPS      2   INELIGIBLE               0   Q+PEND   0.4   TOT      4652M   HST           0.0   HSP MISS   0.0
EXCTD       0   CONVERSION               0   IOSQ     0.0   /SEC     2584K   IFA         120.2   EXP SNGL   0.0   SHARED  2840.76
AVG ENC  0.00   STD DEV       18.54.01.215                                   APPL% CP    364.5   EXP BLK    0.0
REM ENC  0.00                                               ABSRPTN    33K   APPL% IFACP   2.9   EXP SHR    0.0
MS ENC   0.00                                               TRX SERV   33K   APPL% IFA     6.7

GOAL: EXECUTION VELOCITY 60.0%     VELOCITY MIGRATION:   I/O MGMT  66.5%     INIT MGMT 66.5%

         RESPONSE TIME EX   PERF  AVG   --- USING% --- ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--    %
SYSTEM                 VEL% INDX ADRSP   CPU  IFA  I/O  TOT  I/O  CPU                          UNKN IDLE  USG  DLY  USG  DLY QUIE

*ALL        --N/A--    66.5  0.9  79.2   4.3  0.1  9.6  7.1  3.7  3.3                           0.2 78.8  0.5  0.0  0.0  0.0  0.0
JA0                    81.0  0.7  12.0   7.9  0.0  1.5  2.2  0.6  1.6                           0.0 88.4  0.1  0.0  0.0  0.0  0.0
JB0                    71.3  0.8  18.1   4.3  0.1 27.4 12.8 10.8  2.0                           0.3 55.2  0.4  0.0  0.0  0.0  0.0
JC0                    64.8  0.9  15.1   5.4  0.3  1.3  3.8  0.4  3.4                           0.4 88.9  1.9  0.0  0.0  0.0  0.0
JE0                    66.3  0.9   8.0   5.9  0.0  3.4  4.7  1.1  3.6                           0.4 85.6  0.0  0.0  0.0  0.0  0.0
JF0                    25.1  2.4   7.0   1.6  N/A  0.2  5.2  0.0  5.1                           0.1 93.0  0.0  0.0  0.0  0.0  0.0
J90                    28.3  2.1  12.0   2.0  N/A  1.5  9.0  0.5  8.4                           0.1 87.4  0.0  0.0  0.0  0.0  0.0
TPN                    79.6  0.8   2.0   0.3  N/A  0.0  0.1  0.0  0.1                           0.0  100  0.0  0.0  0.0  0.0  0.0
Z0                     34.3  1.8   5.0   0.4  N/A  0.0  0.7  0.0  0.7                           0.0 98.9  0.0  0.0  0.0  0.0  0.0

-----------------------------------------------service classES BEING SERVED-------------------------------------------------------
CICSTRN    CICSCONV   CICSDEFA   CICSLONG   CICSMISC
```

Example 8-2 is the server view of a CICS workload. This report is divided in two parts: SRM data and WLM data.

SRM data

Some interesting values:

Number of regions	79 (2 regions were stopped during this interval)
DASD rate	8746 I/O per sec
DASD resp. time	0.9 ms
CPU usage	364.5%, that is, 3.65 CPUs consumed
Paging activity	0.0
Storage used	5.5 GB (1403209 4K pages)

Tip: To monitor a specific region or a group of regions, we recommend that you use report classes. To report them, use the statement: `SYSRPTS(WLMGL(RCPER))`.

WLM data

This part of the report shows, at a sysplex level and for each member of the sysplex, all the WLM indicators:

- Execution velocity
- Performance index

- Major delays

Some additional observations regarding this sample report follow.

- The execution velocities are very different among the systems.
  - As mentioned at the beginning of the chapter, the goal of this service class (region mode) is not used to manage regions based in the performance of the transactions being executed on those address spaces.
  - This sysplex is running on three different CPCs (2 x 2084 and 1 x 2064), and the number of logical processors is different for each LP.
- For system JB0, we notice an execution I/O delay of almost 11%.

If you notice any specific delays in one member of the sysplex (for example, the mentioned I/O delay for JB0), you might investigate that system in more detail.

For example, if paging occurs, an average rate for the entire sysplex has no meaning. Paging has local explanations, among them:

- Not enough real storage installed in a LP
- Number of regions started in this image of the sysplex
- Activity of this image

The last part of the report (not shown here) gives the list of the transaction service classes *served* by this region:

- CICSTRN
- CICSCONV
- CICSDEFA
- CICSLONG
- CICSMISC

Note: We get this list only with a service class report: `SYSRPTS(WLMGL(SCLASS))`.

Now we have information about resource usage of the CICS workload in terms of CPU, I/O, and storage, but we also require data about the transactions within CICS. So we look at the second part of the Workload Activity report, the *transaction view*.

Workload Activity report: Transaction view

The report shown in Example 8-3 looks a little complex, but is very important to understand because it shows in detail where a transaction spends its time.

Subsystem work manager delays

To understand the data in the report, you first need to get the idea of subsystem work manager delays.

A Performance Block (PB) is a control block used by transaction managers such as CICS and IMS to inform WLM about the internal transaction state through the use of a fast API. A PB is also called a *performance environment*. For additional information, refer to “Monitor III Work Manager Delays report” on page 198.

Generally, CICS transactions have more than one phase:

- The begin-to-end phase (CICS BTE) takes place in the first CICS region to begin processing a transaction. Usually this is the terminal owning region (TOR). The TOR is responsible for starting and ending the transaction.

In our example:

– The ENDED field shows that 374881 transactions completed.

– The ACTUAL time shows that the 374881 transactions completed in an average response time of 0.030 seconds.

- The execution phase (CICS EXE) can take place in one or more application owning regions (AOR) and in a file owning region (FOR). However, there will be only one EXE row, even if transactions execute through multiple regions.

  In this example, only 225309 transactions were routed by the TOR to the AOR.

  – The EXCTD field shows that the AORs completed 225309 transactions in the interval.

  – The EXECUTION time shows that on average it took 0.029 seconds for the AORs to execute the transactions.

The EXECUTION time applies only to the EXCTD transactions.

Example 8-3 Workload Activity Report (transaction view)

```
REPORT BY: POLICY=WLMPOL01   WORKLOAD=CICS       SERVICE CLASS=CICSTRN          RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=2
                                                    CRITICAL     =NONE

   TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT
   AVG      0.00   ACTUAL                  30
   MPL      0.00   EXECUTION               29
   ENDED  374881   QUEUED                   0
   END/S  208.26   R/S AFFINITY             0
   #SWAPS      0   INELIGIBLE               0
   EXCTD  225309   CONVERSION               0
   AVG ENC  0.00   STD DEV                120
   REM ENC  0.00
   MS ENC   0.00

              RESP  -------------------------------- STATE SAMPLES BREAKDOWN (%) -------------------------------  ------STATE------
   SUB    P   TIME --ACTIVE--  READY IDLE -----------------------------WAITING FOR-----------------------------  SWITCHED SAMPL(%)
   TYPE        (%)   SUB  APPL              I/O CONV LOCK PROD MISC                                               LOCAL SYSPL REMOT
   CICS  BTE  75.8   3.9   0.0   0.1  0.0   1.4 94.4  0.0  0.0  0.1                                               84.3  10.3   0.0
   CICS  EXE  28.5  26.3   0.0   8.8  0.0  63.1  0.0  0.3  0.0  1.5                                               15.1   111   0.0
   DB2   BTE   0.0   0.0   0.0   0.0  0.0   0.0  0.0  0.0  0.0  0.0                                                0.0   0.0   0.0
   DB2   EXE   2.6  42.3   0.0   0.0  0.0   7.6  0.0 48.7  0.2  1.3                                                0.0   0.0   0.0
   IMS   BTE   0.5   100   0.0   0.0  0.0   0.0  0.0  0.0  0.0  0.0                                                0.0   0.0   0.0
   IMS   EXE   2.4  91.6   0.0   0.0  0.0   0.0  0.0  0.0  8.4  0.0                                                0.0   0.0   0.0
   SMS   BTE   0.0   0.0   0.0   0.0  0.0   0.0  0.0  0.0  0.0  0.0                                                0.0   0.0   0.0
   SMS   EXE  25.5  12.0   0.0   0.0  0.0  87.7  0.0  0.0  0.3  0.0                                                0.0   0.0   0.0
```

Attention: Since z/OS 1.4, the breakdown of delay is expressed in percentage of *samples* and no longer in percentage of *response time*. The response time is calculated when a transaction completes. This could result in values greater than 100% when samples for long-running transactions were included, which did not complete in the RMF interval.

Meaning of the fields

SUBTYPE Can be CICS, DB2, IMS, or SMS (for VSAM RLS).

Refer to the documentation of each subsystem to get more information about the meaning of each delay for the subsystem.

P BTE (begin-to-end phase) or EXE (execution phase).

RESP TIME(%) Transaction response time percentage in either BTE or EXE phase.

STATE SAMPLES BREAKDOWN(%):

ACTIVE From the CICS point of view, the program is executing (it does not mean it is executing from an MVS view). This state has been recently divided into two parts:

ACTIVE SUB Work has been requested by the subsystem itself.

ACTIVE APPL Work has been requested by an application invoked by the subsystem.

READY This indicates that a program was ready to execute on behalf of a work request, but the work manager has given priority to another work request.

IDLE Idle due to thinking time in a conversational transaction.

STATE SAMPLES BREAKDOWN (%) – WAITING FOR WORK:

LOCK Waiting for a lock. For example, waiting for:

- A lock on a CICS resource
- A record lock on a recoverable VSAM file
- An application resource that has been locked by an EXEC CICS ENQ command.

I/O Waiting for an I/O request or another function related to the I/O request:

- File control, transient data, temporary storage, or journal I/O
- Waiting for I/O buffers or VSAM strings

CONV Waiting for a conversation between work manager subsystems. For example, TOR waiting for a response back from an AOR.

LOCL Waiting for a session to establish with another CICS locally.

SYSP Waiting for a session to establish with another CICS in the sysplex.

REMT Waiting for a session to establish with another CICS in the network.

TIME Waiting for a timer event of an interval control event to complete.

LTCH Waiting for a DB2 latch.

PROD Waiting for another product to complete its function. For example, when the work requests have been passed to a DBCTL subsystem.

SSTL Waiting for an SSL thread.

REGT Waiting for a regular thread.

WORK Waiting for registration to a work table.

MISC Waiting for an unidentified resource.

STATE SWITCHED SAMPL(%):

LOCAL State representing transactions for which there are logical continuations on this z/OS image. Subsystems set this state when they function-ship a transaction to another component in the same z/OS image.

SYSPL Same as local, but for another component in the sysplex.

REMOT Same as local, but for another component in the network.

How to interpret this report

- The average response time (`ACTUAL`) for all CICS transactions in service class CICSTRN is 30 msec (0.030 sec).

- In the first row, CICS BTE is 75.8% of the response time, which is 22.7 msec. Then 30.0 - 22.7 msecs is the average time in the TOR queue, equivalent to 24.2% of the response time. Most of this time was spent outside the TOR; that is, in AOR: CONV = 94.4%. This information is given in detail in STATE SWITCHED SAMPL: 84% in another AOR region in the same system and 10% in another AOR region in another member of the sysplex.

The response times given for all other states are part of this response time.

- In the last row, for SMS EXE, 25.5% of the actual time was spent in the SMS subsystem, which is 7.6 msec. The delay I/O value is 87.7% of this time, which results in 6.6 msec for the I/O time.
- The following value needs some careful interpretation: the 48.7% of LOCK delay for the row DB2 EXE seems to be high, but it is only 48.7% of 2.6% of the response time, or 0.37 msec.

> **Note:** You can get this information also in the Monitor III Work Manager Delays report (see Example 8-7 on page 263), were you can select between the response time values and the response time percentages.

These values are a good starting point for further application investigation.

Workload Activity report: Response time distribution

Now we are looking at the third part of this report, the response time distribution.

Example 8-4 Workload Activity Report (distribution report)

```
REPORT BY: POLICY=WLMPOL01   WORKLOAD=CICS       SERVICE CLASS=CICSTRN    RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=2
                                                    CRITICAL     =NONE

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT
   AVG      0.00   ACTUAL                  30
   MPL      0.00   EXECUTION               29
   ENDED  374881   QUEUED                   0
   END/S  208.26   R/S AFFINITY             0
   #SWAPS      0   INELIGIBLE               0
   EXCTD  225309   CONVERSION               0
   AVG ENC  0.00   STD DEV                120
   REM ENC  0.00
   MS ENC   0.00

   GOAL: RESPONSE TIME 000.00.00.600 FOR  80%

            RESPONSE TIME EX   PERF
   SYSTEM      ACTUAL%    VEL% INDX

   *ALL          99.4      N/A  0.5
   JA0            100      N/A  0.5
   JC0            100      N/A  0.5
   JE0            100      N/A  0.5
   JF0           98.7      N/A  0.5
   J90           98.8      N/A  0.5
   Z0             0.0      N/A  N/A

                                          ----------RESPONSE TIME DISTRIBUTION----------
      ----TIME----    --NUMBER OF TRANSACTIONS--    -------PERCENT-------  0    10   20   30   40   50   60   70   80   90   100
      HH.MM.SS.TTT    CUM TOTAL        IN BUCKET    CUM TOTAL   IN BUCKET  |....|....|....|....|....|....|....|....|....|....|
   <  00.00.00.300          369K            369K         98.5        98.5  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
   <= 00.00.00.360          370K            1186         98.8         0.3  >
   <= 00.00.00.420          371K             868         99.0         0.2  >
   <= 00.00.00.480          372K             658         99.2         0.2  >
   <= 00.00.00.540          372K             388         99.3         0.1  >
   <= 00.00.00.600          373K             345         99.4         0.1  >
   <= 00.00.00.660          373K             267         99.5         0.1  >
   <= 00.00.00.720          373K             220          100         0.1  >
   <= 00.00.00.780          373K             185          100         0.0  >
   <= 00.00.00.840          374K             182          100         0.0  >
   <= 00.00.00.900          374K             114          100         0.0  >
   <= 00.00.01.200          374K             463          100         0.1  >
   <= 00.00.02.400          375K             618          100         0.2  >
   >  00.00.02.400          375K             131          100         0.0  >
```

The report shown in Example 8-4 shows the following:

- Goal of the service class
  - Importance = 2 (you see this value on the top of the report)
  - Response time with percentile = From 0.0 secs to 0.6 secs for 80%
- Number of transactions ended during the interval: ENDED = 374881
- Average response time: ACTUAL = 0.030 sec
  This response time is 20 times below the maximum value for 80% of the transactions.
- Standard deviation: STD DEV = 0.120 sec
  A low response time standard deviation value indicates that the response times are fairly consistent. A high response time standard deviation value would indicate inconsistent response times.
- Percentile of goal met: 99.4%. This means the goal is reached because 99.4 is greater than 80%.
- Average performance index: 0.5

We now have all the key indicators to monitor our workload for CICS:

- Number of transactions
- Average response time
- Resources utilization (CPU, I/O, and storage)
- Performance index

Using different report classes for the servers, we can obtain resource utilization by address spaces or by groups of address spaces. And we can do the same for the transaction service classes to obtain a number of transactions and response time for a group of transactions, or for a group of users.

The performance index should be a real indicator. We may ask why is the response time so good (what a strange question for a performance analyst)?

- Is the system temporarily underutilized?
- Is the goal not correctly defined?

If the goal is correctly defined (connected to the business needs), you can use this performance index as a very precise indicator of your response time. But for a correct interpretation, you should know how it is calculated for a percentile response time type of goal. An example follows.

Example 8-5 Calculation of Performance Index

```
        ---RESPONSE TIME---  EX   PERF
        HH.MM.SS.TTT         VEL  INDX
GOAL    00.00.00.300  90.0%
ACTUALS
JA00                  87.1%  N/A   1.4
                                       ----------RESPONSE TIME DISTRIBUTION----------
   ----TIME----    --NUMBER OF TRANSACTIONS--    -------PERCENT-------  0    10   20   30   40   50   60   70   80   90   100
   HH.MM.SS.TTT    CUM TOTAL        IN BUCKET    CUM TOTAL   IN BUCKET  |....|....|....|....|....|....|....|....|....|....|
<  00.00.00.150         3686             3686         79.5        79.5  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
<= 00.00.00.180         3778               92         81.5         2.0  >>
<= 00.00.00.210         3875               97         83.6         2.1  >>
<= 00.00.00.240         3942               67         85.0         1.4  >
<= 00.00.00.270         3990               48         86.1         1.0  >
<= 00.00.00.300         4036               46         87.1         1.0  >
<= 00.00.00.330         4080               44         88.0         0.9  >
<= 00.00.00.360         4121               41         88.9         0.9  >
<= 00.00.00.390         4154               33         89.6         0.7  >
<= 00.00.00.420         4178               24         90.1         0.5  >
<= 00.00.00.450         4178                0         90.1         0.0  >
<= 00.00.00.600         4293              115         92.6         2.5  >>
<= 00.00.01.200         4544              251         98.0         5.4  >>>
>  00.00.01.200         4635               91          100         2.0  >>
```

To obtain the performance index, we work with Example 8-5. There, we have the response time distribution with 14 buckets, giving the number of transactions that belong to each bucket. In this report, we have a goal of 0.030 sec or less for 90% and a performance index of 1.4.

- We check column `PERCENT CUM TOTAL`, which contains the cumulative percentage values. The first value above 90% is in the line with 90.1. The corresponding response time is 0.420. This means that 90.1% of all transactions have a response time of 0.420 sec or less.
- Now we divide this response time by the goal response time and we obtain: 0.420/0.300 = 1.4.

Make sure that you carefully interpret this performance index of 1.4, which means that there is a degradation of 40%. In this case, fewer than 3% of all transactions led to this degradation, and 87.1% of them met the objective.

Tip: For RMF to provide meaningful Workload Activity Report data, we suggest that you use the following guidelines when defining the service classes for CICS transactions. Do not mix together in the same service class:

- CICS-supplied transactions with user transactions
- Routed with non-routed transactions
- Conversational with pseudo-conversational transactions
- Long-running and short-running transactions

8.2.3 Using Monitor III

We start with the Sysplex Summary report. Using the report options (**RO** command), we select only the Workload CICS.

Tip: During setup of WLM Service Definition, it is useful to define the transactions service classes and the server service classes in the same workload.

Example 8-6 Sysplex Summary report

```
                     RMF V1R5   Sysplex Summary - UTCPLXJ8       Line 1 of 106
Command ===>                                                  Scroll ===> CSR

WLM Samples: 239     Systems: 12 Date: 11/09/04 Time: 19.38.00 Range: 60    Sec

                      >>>>>>>>XXXXXXXXXXXXXXXXXX<<<<<<<<

Service Definition: WLMDEF01              Installed at: 11/09/04, 10.27.25
     Active Policy: WLMPOL01              Activated at: 11/09/04, 10.27.50

               ------- Goals versus Actuals --------  Trans --Avg. Resp. Time-
               Exec Vel  --- Response Time ---  Perf  Ended  WAIT EXECUT ACTUAL
Name     T  I  Goal Act  ---Goal--- --Actual--  Indx  Rate   Time   Time   Time

BATCH    W           80                               0.017 1.304  3.105  4.408
DISCR    S  D        93                               0.017 1.304  3.105  4.408
WLMBTCHH S  2    50  80                         0.62  0.000
CICS     W          N/A                                1455 0.000  0.027  0.092
CICSTRN  S  2       N/A  0.600 80%        100%  0.50   1254 0.000  0.027  0.018
CICSCONV S  3       N/A  10.00 50%         36%  1.50  8.100 0.000  13.77  13.77
CICSDEFA S  3       N/A  1.000 90%        100%  0.50  9.983 0.000  0.073  0.028
```

```
CICSMISC S  3       N/A  1.000 90%        100%  0.50  183.1 0.000  0.002  0.002
CICSRGN  S  2    60  77                         0.78  0.000
ICSS     W           46                               0.000 0.000  0.000  0.000
```

With this first report (Example 8-6), we get an overview of the CICS workload:

- Transactions rate global and detailed by service classes
- Average response time (global and detailed)
- Performance index of each service class
- Velocity and performance index of the servers

Using cursor-sensitive control, we can obtain a Work Manager Delays report by pointing under a service class (CICSTRN in this case). This report is based in the sampling of the PBs.

Example 8-7 Work Manager Delays report

```
                  RMF V1R5   Work Manager Delays - UTCPLXJ8       Line 1 of 39
Command ===>                                                  Scroll ===> CSR

WLM Samples: 239     Systems: 12 Date: 11/09/04 Time: 19.38.00 Range: 60    Sec

Class:  CICSTRN    Period: 1         Avg. Resp. time: 0.018 sec for 75243  TRX.
Goal:   0.600 sec for  80%           Avg. Exec. time: 0.027 sec for 47330  TRX.
Actual: 0.600 sec for 100%           Abnormally ended:                  0  TRX.

Sub  P  -----------------Response time breakdown (in %)------------ -Switched--
Type     Tot  Act  Rdy Idle -----------------Delayed by------------  Time (%)
                            CONV  I/O MISC LOCK PROD TIME DIST SESS LOC SYS REM

CICS B  75.8  2.7  0.1  0.0 71.7  1.4    0    0    0    0    0    0  69 2.6   0
CICS X  30.2  8.7  1.5    0    0 19.9  0.1    0  0.0    0    0    0 4.9  27   0
SMS  X  29.3  2.6    0    0    0 26.7    0  0.1    0    0    0    0   0   0   0
DB2  X   5.1  2.7    0    0    0  0.1  2.1  0.0    0    0    0    0   0   0   0
------------ Address Spaces Serving this service class CICS     ---------------
Jobname  M ASID System  Serv-Class Service Proc-Usg I/O-Usg  Veloc  Capp  Quies

CICS2ABA Y  805  JBO     CICSRGN     100       12     8.3      86     0       0
CICS2ABB Y  553  JBO     CICSRGN     100      8.3     3.3      64     0       0
CICS2ABC Y  813  JBO     CICSRGN     100       12     8.3      92     0       0
CICS2FBA Y  575  JBO     CICSRGN      13       53     347      73     0       0
```

This report (Example 8-7) is a different presentation of the values in the Workload Activity report already shown.

- The first section is very similar, but gives new information: the number of abended transactions.
- The average response time is lower than the execution time. One possible explanation is the presence of non-routed transactions in this service class. We see that there is a noticeably large difference between the two numbers. Non-routed transactions are generally fast and can lower the response time average.
- The second section (Response time breakdown) contains the same fields as the batch report, but the content is either a percentage of the response time or the time itself, in seconds. To toggle between percentage and seconds, use cursor-sensitive control anywhere in the middle section of the report.

  If the report is displayed in seconds and a value does not fit, *** is shown in that field. Switching to percentage provides a better representation of the numbers.

These transactions are started in TOR, where the beginning-to-end phase starts and ends. After that, they are routed to an AOR exec phase, and while in that phase, the VSAM (SMS) and DB2 exec phases were invoked.

- For consistency, both begin-to-end phase (Phase = B) and execution phase rows (Phase = X) are in relation to the average response time (Avg. Resp. Time).

  For example, if we take the row SMS, I/O delay = 26.7% means 26.7*18/100=4.8 msec.
- The last section is a scrollable area which gives the using data and the velocity for all address spaces serving this service class.

Example 8-8 Workload Manager report (toggled in seconds)

```
           .............
Sub  P  -----------------Response time breakdown (in seconds)------ -Switched--
Type     Tot  Act  Rdy Idle -----------------Delayed by------------  Time (%)
                            CONV  I/O MISC LOCK PROD TIME DIST SESS LOC SYS REM

CICS B  .014 .000 .000 .000 .013 .000    0    0    0    0    0    0  69 2.6   0
CICS X  .005 .002 .000    0    0 .004 .000    0 .000    0    0    0 4.9  27   0
SMS  X  .005 .000    0    0    0 .005    0 .000    0    0    0    0   0   0   0
DB2  X  .001 .000    0    0    0 .000 .000 .000    0    0    0    0   0   0   0
------------ Address Spaces Serving this service class CICS     ---------------
           ..............
```

If we toggle to seconds, the section Response time breakdown will appear as in Example 8-8.

Placing the cursor under the service class name CICSRGN, we get the Group Response Time report.

Example 8-9 Group Response Time report

```
                        RMF V1R5   Group Response Time
Command ===>
Switched to option set WLMPOL01 on JB0.
Samples: 60      System: JB0   Date: 11/09/04  Time: 19.38.00  Range: 60    Sec

Class: CICSRGN      Period: 1    Description: All CICS regions
Primary Response Time Component: Using the processor

                                                 TRANS    --- Response Time ----
WFL    Users    Frames   Vector   EXCP    PGIN   Ended    -- Ended TRANS-(Sec) -
 %    TOT  ACT   %ACT     UTIL    Rate    Rate   Rate       WAIT  EXECUT  ACTUAL
 57    15    3      5        0   4,633    0.0   0.000     0.000   0.000   0.000

                         -AVG USG-   -------------Average Delay--------------
                  Total  PROC  DEV   PROC   DEV  STOR  SUBS  OPER   ENQ OTHER
Average Users     2.666  1.08 0.86   0.40  0.13  0.00  0.06  0.85  0.00  0.00
Response Time ACT 0.000  0.00 0.00   0.00  0.00  0.00  0.00  0.00  0.00  0.00

                    ---STOR Delay---  ---OUTR Swap Reason---  ---SUBS Delay---
                    Page  Swap  OUTR    TI    TO    LW    XS   XCF   JES   HSM
Average Users       0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.06  0.00  0.00
Response Time ACT   0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
```

Example 8-9 shows delays occurring in all address spaces of the service class CICSRGN.

Although they are not important in our example, we investigate them here to demonstrate the navigational capabilities of Monitor III.

Placing the cursor under the processor delay value (0.40), we get the Processor Delays report.

Example 8-10 Processor Delays report

```
                      RMF V1R5   Processor Delays                  Line 1 of 8
Command ===>                                                  Scroll ===> CSR
Report is for service class CICSRGN only.
Samples: 60      System: JB0   Date: 11/09/04  Time: 19.38.00  Range: 60    Sec

            Service  DLY USG  Appl EAppl ----------- Holding Job(s) -----------
Jobname  CX Class     %   %      %     %  %  Name      %  Name      %  Name


CICS2FBA SO CICSRGN   17  42  23.5  23.5   5 U0210194   3 VTAM44     3 CICS2ABC
CICS2TBA SO CICSRGN   10  22  21.0  21.0   7 VTAM44     5 CICS2TBA   3 CICS2ABA
CICS2ABB SO CICSRGN    7   8  10.9  10.9   2 U0210254   2 U0210194   2 CICS2ABC
CICS2ABA SO CICSRGN    3  10  12.8  12.8   2 *MASTER*   2 VTAM44     2 CSQBCHIN
CICS3ABA SO CICSRGN    2  13  18.0  17.2   2 *MASTER*   2 VTAM44     2 CSQBCHIN
CICS2ABC SO CICSRGN    2  12  12.2  12.2   2 U0210174   2 CICS2TBA
CICS3TBA SO CICSRGN    0   2   4.8   4.8
CPSMCAS  S  CICSRGN    0   0   0.3   0.3
```

If we want to investigate the I/O delays, we can navigate from the Group Response Time report by placing the cursor under the device delay field to the Device Delays report.

Example 8-11 Device Delays report

```
                      RMF V1R5   Device Delays                     Line 1 of 1
Command ===>                                                  Scroll ===> CSR
Report is for service class CICSRGN only.
Samples: 60      System: JB0   Date: 11/09/04  Time: 19.38.00  Range: 60    Sec

           Service  DLY USG CON  ------------ Main Delay Volume(s) ------------
Jobname  C Class     %   %   %    %  VOLSER   %  VOLSER   %  VOLSER   %  VOLSER


CICS2FBA S CICSRGN   13  85 217   10 NRLS05    3 NRLS0D    3 NRLS06    3 NRLS09
```

This report (Example 8-11) brings us to the Device Resource Delay report by placing the cursor under NRLS05.

Example 8-12 Device Resource Delay

```
                      RMF V1R5   Device Resource Delays            Line 1 of 3
Command ===>                                                  Scroll ===> CSR

Samples: 60      System: JB0   Date: 11/09/04  Time: 19.38.00  Range: 60    Sec

Volume S/   Act  Resp  ACT CON DSC PND %,  DEV/CU            Service  USG DLY
  /Num PAV  Rate Time   %   %   %  Reasons Type     Jobname  C Class     %   %


NRLS05 S    2824 .001   31  17   2 PND  12 33903    CICS2FBA S CICSRGN   78  10
  2B35  10                         CMR   7 2105
                                   DB    1
```

Example 8-12 shows that there is very high activity on this volume, with a very good response time. The explanation is the use of 10 parallel access volumes.

From here, we navigate to the Data Set Delays report.

Example 8-13 Data Set Delays report

```
                      RMF V1R5   Data Set Delays - Volume         Line 1 of 2
Command ===>                                                  Scroll ===> CSR

Samples: 60      System: JB0   Date: 11/09/04  Time: 19.38.00  Range: 60    Sec

-------------------------- Volume NRLS05 Device Data --------------------------
Number:    2B35        Active:      31%       Pending:   12%     Average Users
Device:    33903       Connect:     17%       Delay DB:   1%        Delayed
Shared:    Yes         Disconnect:   2%       Delay CM:   7%          0.1
PAV:       10
-------------- Data Set Name ---------------   Jobname   ASID  DUSG% DDLY%
RLSADSW.KILLER.GPVSAM5.INDEX                   CICS2FBA  0575     58     8
RLSADSW.KILLER.GPVSAM5.DATA                    CICS2FBA  0575     40     3
```

Finally, with this report (Example 8-13), we get the data set name of these busy data sets.

Another delay to be analyzed is the Subsystem delay in Example 8-9 on page 264. There we see that it is an XCF delay, and pointing under the value, we get the XCF Delays report.

Example 8-14 XCF Delays report

```
                      RMF V1R5   XCF Delays                       Line 1 of 1
Command ===>                                                  Scroll ===> CSR
Report is for service class CICSRGN only.
Samples: 60      System: JB0   Date: 11/09/04  Time: 19.38.00  Range: 60    Sec

             Service    DLY     ------------ Main Delay Path(s) -----------
Jobname   C  Class       %         %  Path    %  Path    %  Path    %  Path


CICS2FBA  S  CICSRGN      7        7  -CF-
```

In this case (Example 8-14), XCF is delayed by an access to the signaling structure. XCF and CICS delays are different than other uses of XCF services. In fact what is being measured here is the intensity of the use of XCF services, and individual CICS transactions are not necessarily being delayed. More often, when RMF is sampling the CICS address space, it sees a high usage of XCF and comes to a conclusion that the CICS address space (as a single entity) is hung up on XCF. This is most often not true and might make it difficult to use RMF for this environment to do your problem determination when a problem arises.

And the last delay to be analyzed is uncommon, but potentially critical for an online environment. If we point the cursor under the field OPER in Example 8-9 on page 264, we get the Delay report.

Example 8-15 Delay report

```
                      RMF V1R5   Delay Report                    Line 1 of 16
Command ===>                                                  Scroll ===> CSR
Report is for service class CICSRGN only.
Samples: 60      System: JB0   Date: 11/09/04  Time: 19.38.00  Range: 60    Sec

            Service     WFL USG DLY IDL UKN ---- % Delayed for ---- Primary
Name     CX Class    Cr  %   %   %   %   %  PRC DEV STR SUB OPR ENQ Reason


*CICSRGN                57  11   9   0  82   3   1   0   0   6   0
CICS1ABC SO CICSRGN      0   0  85   0  15   0   0   0   0  85   0 Message
CICS2ABB SO CICSRGN     56   8   7   0  85   7   0   0   0   0   0 U0210254
CICS2TBA SO CICSRGN     70  23  10   0  72  10   0   0   0   0   0 VTAM44
CICS2ABA SO CICSRGN     75  10   3   0  88   3   0   0   0   0   0 *MASTER*
```

```
CICS2FBA SO CICSRGN      78  92  35   0   3  17  13   0   7   0   0 U0210194
CICS2ABC SO CICSRGN      88  12   2   0  87   2   0   0   0   0   0 U0210174
CICS3ABA SO CICSRGN      89  13   2   0  85   2   0   0   0   0   0 *MASTER*
CICS3TBA SO CICSRGN     100   2   0   0  98   0   0   0   0   0   0
CICSCWBC SO CICSRGN           0   0   0 100   0   0   0   0   0   0
```

Example 8-15 gives us the beginning of the explanation to the operator intervention for CICS1ABC: it is a WTOR. This really should not happen in an online environment. By placing the cursor on the primary reason, we get the report shown in Example 8-16, which indicates the reply id of this WTOR.

Example 8-16 Job Delays report

```
                        RMF V1R5   Job Delays                      Line 1 of 1
Command ===>                                                  Scroll ===> CSR

Samples: 60      System: JBO   Date: 11/09/04  Time: 19.38.00  Range: 60    Sec

Job: CICS1ABC     Primary delay: Awaiting reply to operator request 3434.

--------------------------- Job Performance Summary ---------------------------
        Service       WFL -Using%- DLY IDL UKN ---- % Delayed for ---- Primary
CX ASID Class    P Cr  %   PRC DEV  %   %   %  PRC DEV STR SUB OPR ENQ Reason
SO 0034 CICSRGN  1      0    0   0  85   0  15   0   0   0   0  85   0 Message
```

Finally, we discover the following message in the Syslog:

Example 8-17 Pending WTO in Syslog

```
@3434 DFS690A CTL IMSB NOT ACTIVE, REPLY 'WAIT' OR 'CANCEL' OR 'alt-id'
```

The solution for this problem is simple: start the missing subsystem.

Conclusion

In this chapter, we have shown you many reports from the Postprocessor and Monitor III that can help you monitor your CICS subsystem and analyze performance problems if they appear. Of course, there are other reports, for example, those shown in Chapter 7, “Monitoring batch workloads using RMF” on page 217, that provide a wider view of your system, and that are not specific for a transactional workload.

9

Understanding the new RMF reporting capabilities

New RMF reporting capabilities have been introduced to reflect enhancements to zSeries products and offerings. This chapter describes these new capabilities, specifically the following:

- Support for the new concept of Workload Licensed Charges
- Information about Intelligent Resource Director (IRD)
- Postprocessor details regarding cryptographic processing
- zSeries Application Assist Processor (zAAP) data in some reports
- UNIX System Services reporting
- WebSphere Application Server information

9.1 Workload Licensing Charges considerations

Prior to z/OS, IBM software products running on OS/390 were typically priced based on the computing capacity of the CPC on which the software was running. With z/OS running on a zSeries, the new concept of Workload Licensed Charges (WLC) is used to manage this software pricing system. Now the cost is based on the consumption of the LP defined for the product, which is typically less than the total CPC capacity.

z/OS measures the consumption in LPs based on a four hour millions of CPU service units (MSU) rolling average. This value is derived every 5 minutes. However, the installation is charged by the peak of this 4-hour MSUs/hour during the month. So, there is the possibility to implement a defined capacity (soft capping), to enforce a limit on LP consumption in 4-hour MSUs/hour. This type of capping is controlled by WLM and executed by LPAR hypervisor code. However, you may see that for a certain amount of time (without being soft capped) your LP consumption can be larger than this defined capacity, as long as the average workload during the four hour period does not exceed it.

RMF provides ways to monitor both of following:

- Current capacity
- 4h MSU average for partitions having Defined Capacity enabled

9.1.1 Defined Capacity

Example 9-1 shows a Postprocessor partition report for an installation with Defined Capacity enabled.

Example 9-1 Partition report (defined capacity)

```
MVS PARTITION NAME                     A04
IMAGE CAPACITY                          50
NUMBER OF CONFIGURED PARTITIONS         29
NUMBER OF PHYSICAL PROCESSORS           24
                   CP                   18
                   ICF                   6
WAIT COMPLETION                         NO
DISPATCH INTERVAL                  DYNAMIC

--------- PARTITION DATA -----------------  -- LOGICAL PARTITION PROCESSOR DATA --   -- AVERAGE PROCESSOR UTILIZATION PERCENTAGES --
                   ----MSU----  -CAPPING--  PROCESSOR-  ----DISPATCH TIME DATA----   LOGICAL PROCESSORS  --- PHYSICAL PROCESSORS ---
NAME       S   WGT  DEF    ACT  DEF   WLM%  NUM   TYPE   EFFECTIVE       TOTAL       EFFECTIVE    TOTAL  LPAR MGMT  EFFECTIVE  TOTAL
A04        A    10   50     40  NO    10.6    1   CP    00.08.39.484  00.08.39.484       86.58    86.58      0.00        4.81  4.81
A0A        A   185    0      4  NO     0.0    2   CP    00.00.45.529  00.00.48.744        3.79     4.06      0.03        0.42  0.45
```

In this report we see field `MSU DEF` where defined capacity has been set to 50 MSU/h for the A04 LP. This value is reflected in the top of the report in the field `IMAGE CAPACITY`.

If defined capacity is not used, MSU DEF contains 0 and IMAGE CAPACITY contains the capacity derived from the LP definition with the following the formula:

$$\text{Image capacity} = \frac{\text{CPC capacity}}{\text{\# Phys. Processors}} \times \text{\# Logical Processors (Initial+Reserved)}$$

where Reserved are the logical CPUs that are defined in the HMC reset profile of this LPAR.

Field `MSU ACT` in this report always shows the actual consumption in MSU/hour value, even if Defined Capacity is not enabled.

- The value is calculated based on the given capacity of the CPC and the percentage of processors used. In this case, the capacity of this CPC is 837 (we can see this on the CPC report in Example 9-2 on page 271).

MSU ACT = 837 * 4.81% = 40 MSU/h

- The A04 LP is soft capped at 50 MSU/h, to avoid high fees in software products charged through the WLC plan.
- Do not correlate this value to the amount of CPU service units you can retrieve in the Workload Activity report because this value is calculated with another algorithm.
- The field Capping (DEF) equal to NO means that the installation does not require LPAR hardware capping, where you cap the LP to the limit of its relative weight.

9.1.2 Monitoring Defined Capacity

Using Monitor III, we get the CPC Capacity report with the **CPC** command. Example 9-2 shows an example of the report fro the A04 logical partition.

Example 9-2 CPC Capacity report

```
                        RMF V1R5   CPC Capacity                    Line 1 of
Command ===>                                                   Scroll ===> CS

Samples: 60      System: SYS1  Date: 10/25/04  Time: 14.38.00  Range: 60

Partition:   A04        2084 Model 318
CPC Capacity:     837   Weight % of Max: 74.2       4h MSU Average:    42
Image Capacity:    36   WLM Capping %:    100       4h MSU Maximum:    46

Partition  --- MSU ---  Cap  Proc    Logical Util %   - Physical Util % -
             Def   Act  Def   Num    Effect   Total   LPAR  Effect  Total

*CP                                                    6.9    12.2   19.1
A03            0     4  NO    2.0       4.5     4.8    0.0     0.5    0.5
A04           36    37  NO    1.0      80.1    80.1    0.0     4.4    4.4
A05            0     3  NO    1.0       7.0     7.5    0.0     0.4    0.4
```

This report contains the following information:

- `Partition`: The name of the LP is A04.
- `CPC Capacity` is the total capacity of the CPC: 837 MSU/h
- `Weight % of Max` is the average weighting factor in relation to the maximum defined weighting factor for this partition.
- `Image Capacity` is either the defined capacity (soft capping) or the default capacity of the LP. Here we have defined a capacity of 36 MSU/h as you can see in field `MSU Def`.
- `4h MSU Average` is the average value of consumed MSU/h during the last four hours: 42
- `4h MSU Maximum` is the maximum value, based on an interval of five minutes within the last four hours: 46. There is not much difference between the maximum and the average in this case.
- `WLM Capping %` is the percentage of time when WLM instructed LPAR hypervisor to cap the LP. If this is not done, the four hour average MSU value would have exceed the defined capacity limit. The value of 100% indicates that the partition was capped for the entire range of 60 seconds, so the demand is larger than 36 MSU/h. Soft capping is not as drastic as LPAR capping; as you can see the A04 LP consumed 37 MSU/h, when the defined capacity is 36 MSU/h. In this case the installation will be charged for 36MSU/h and not 37MSU/h.

How does it work

As soon as the 4h MSU/h average is higher than the defined capacity, the partition has to be capped. WLM then turns on LPAR hardware capping during the time necessary to obtain an actual capacity close to the defined capacity.

LPAR hardware capping uses the specified weight. So, we have to compare the specified weight value to the `Def MSU` value. To do this comparison, we obtain an MSU value from the specified weight value using the formula:

$$\text{MSU (based on LPAR weight)} = \frac{\text{Total defined capacity} \times \text{LPAR weight}}{\text{Total CPC weight}} = \frac{837 \times 10}{910} = 9$$

Without the soft capping capability, LP A04 would have guaranteed 9 MSU/h. This value is used when LPAR capping is active through the HMC setting.

In our case, this figure is lower than the defined capacity. WLM turns on LPAR hardware capping for the time necessary to obtain the defined average, based on a five minute interval. If the CPC is not constrained and if the LP is highly loaded, the partition will be given more than 9 MSU/h during the rest of the time interval, and will be capped for the time necessary.

If the weight guaranteed value is higher than the defined capacity, WLM just turning on LPAR capping has no effect. So, WLM defines a phantom LP with a suitable weight to artificially lower the share of the total CPC to the level of the defined capacity. So, if the LPAR is 100% busy, it will be capped 100% of the time.

Example 9-3 Defined capacity (low weight)

```
                        RMF V1R5   CPC Capacity                  Line 1 of 33
Command ===>                                                 Scroll ===> CSR

Samples: 50      System: SYS1  Date: 10/25/04  Time: 07.34.00  Range: 60    Sec

Partition:   A04        2084 Model 318
CPC Capacity:     837   Weight % of Max: ****       4h MSU Average:    36
Image Capacity:    36   WLM Capping %:    100       4h MSU Maximum:    41

Partition  --- MSU ---  Cap  Proc    Logical Util %   - Physical Util % -
             Def   Act  Def   Num    Effect   Total   LPAR  Effect  Total

*CP                                                    6.8     7.7   14.5
A02            0     4  NO    2.0       3.8     4.1    0.0     0.4    0.5
A03            0     3  NO    2.0       3.3     3.6    0.0     0.4    0.4
A04           36    10  NO    1.0      20.7    20.7    0.0     1.2    1.2
```

As shown in Example 9-3, partition A04 was capped during the entire interval (WLM Capping% = 100) to a capacity of 10. In this case, you should verify whether the weight value of the partition has been defined correctly.

Monitoring can also be done with RMF PM, which adds very useful information about the remaining time before capping.

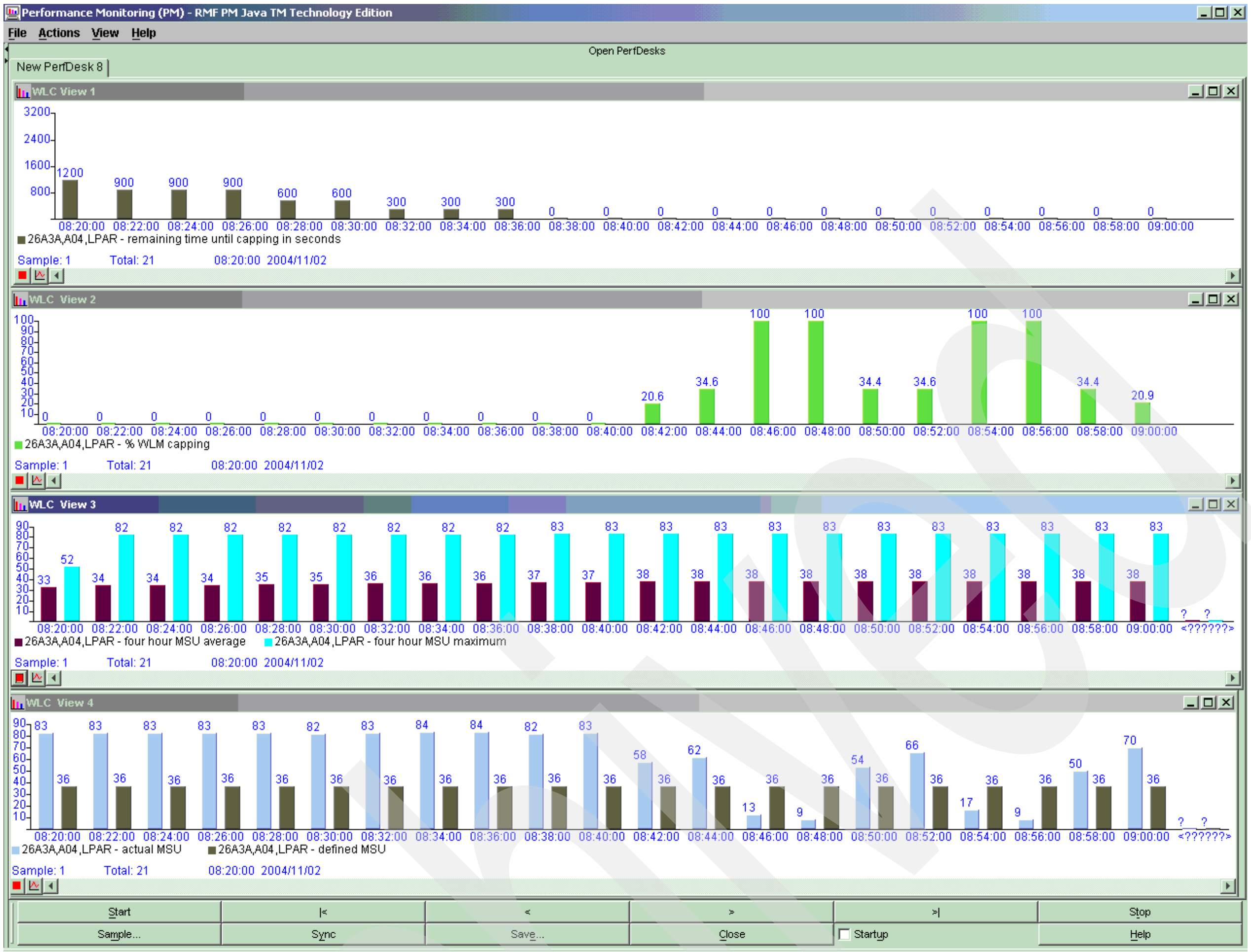


Figure 9-1 RMF PM Perfdesk

Figure 9-1 shows an RMF PM PerfDesk with 4 DataViews that have been built using the CPC resources. Refer to 2.7, “RMF Performance Monitoring” on page 74 for details.

Using DataViews

- `LPAR - remaining time until soft capping starts`

  Based on the constant activity of the LP, this DataView displays a single metric, the remaining time before soft capping takes effect.

- `LPAR - % WLM capping`

  This DataView is also a single metric showing the percentage of time the LP is soft capped during the interval.

- `LPAR - four hour MSU average / maximum`

  This DataView shows the four hour MSU average and the four hour MSU maximum.

- `LPAR - actual / defined MSU`

  This DataView shows the actual and the defined MSU values.

This example illustrates the capping problem generated by a very low weight value in an LP. We can observe that, as soon as the 4h average is higher than the defined capacity (36), the capping begins, but we observe too that during large intervals (two minutes), actual capacity is very low, and this may generate dramatic effects on an online workload.

Note: We recommend to always try to set the defined capacity close to the weight - or vice versa.

9.2 Intelligent Resource Director

In this section, we discuss the Intelligent Resource Director (IRD), and we track how RMF reports the resource usage when IRD is in action. For detailed information about IRD, refer to redbook *z/OS Intelligent Resource Director*, SG24-5952.

IRD was announced on October 3, 2000, as one of the new capabilities available on the IBM zSeries family of processors and delivered as part of z/OS. IRD might be viewed as “stage two” of Parallel Sysplex. Stage one provided facilities to let you share your data, programs, and workload across multiple system images. As a result, applications that supported data sharing could potentially run on any system in the sysplex, thus allowing you to move your workload to where the processing resources are available.

However, not all applications support data sharing. For these applications, IBM has provided IRD, which basically gives you the ability to move the resource to where the workload is.

IRD is not actually a product or a system component; instead, it is three separate but mutually supportive functions:

- WLM LPAR CPU management
  - WLM Vary logical CPU management
  - WLM Weight management
- Dynamic Channel-path management (DCM) for ESCON channels
- Channel subsystem I/O priority queuing (CSS IOPQ)

IRD is implemented by new functions in:

- z/OS (in z/Architecture mode)
- Workload Manager (WLM)
- IBM zSeries

IRD uses the concept of an LPAR cluster, which consists of the subset of z/OS systems (we also may have LPs with Linux with or without z/VM) that are running as LPs on the same zSeries server. The z/OS images must be in the same Parallel Sysplex. IRD uses facilities in z/OS Workload Manager (WLM), Parallel Sysplex, SAP (the PU in charge of starting an I/O operation) and PR/SM to help you derive greater value from your z/Series investment. Compared to other platforms, z/OS with WLM already provides benefits from the ability to drive a processor at 100% while still providing acceptable response times for your critical applications. IRD amplifies this advantage by helping you make sure that all those resources are being utilized by the right workloads, even if the workloads exist in different LPs.

So, IRD is active in our system and we are going to use and report the WLM weight management function.

9.2.1 IRD in action and RMF reports

In this section, we concentrate on using WLM LPAR CPU Management. We see the Monitor I Partition Data Report, and we also see the Monitor III CPC Report, and the Sysplex Summary

Report. We track how IRD changes the weights of the partitions in our CPC in our sysplex, when important work begins on one of the partitions, and the goals are not achieved.

The starting point

We have a sysplex with seven partitions in one CPC forming one LPAR cluster. However, IRD (WLM Weight CPU Management) is activated through HMC on only two of them, in the partition A0A with SC69 z/OS system and partition A01 with SC55 z/OS system. We recommend that the LPname be different from the z/OS image name running on the LPAR. We monitored with RMF reports only these two partitions, although we started our scenario by having IRD not in use at the beginning. We created a set of sample synchronized reports that we can then compare as soon as we activate IRD. The loads of the partitions are reported in the following reports.

Example 9-4 Partitions.1, Monitor III CPC Report

```
                           RMF V1R5   CPC Capacity                  Line 1 of 34
Command ===>                                                   Scroll ===> CSR

Samples: 573     System: SC69  Date: 11/07/04  Time: 16.00.00  Range: 600   Sec

 Partition:   A0A        2084 Model 318
 CPC Capacity:     837   Weight % of Max: ****       4h MSU Average:    77
 Image Capacity:   744   WLM Capping %:   ****       4h MSU Maximum:    92

 Partition  --- MSU ---  Cap  Proc    Logical Util %   - Physical Util % -
             Def   Act  Def   Num    Effect   Total   LPAR  Effect  Total

 *CP                                                    2.6    93.1   95.6
 A0A            0    87  NO    2.0      93.1    93.2    0.0    10.3   10.4
 A0B            0     3  NO    2.0       3.2     3.3    0.0     0.4    0.4
 A0C            0     4  NO    3.0       3.0     3.2    0.0     0.5    0.5
 A01            0    87  NO    2.0      93.1    93.2    0.0    10.3   10.4
 A02            0    87  NO    2.0      93.1    93.2    0.0    10.3   10.4
 A03            0    93  NO    2.0      99.9    99.9    0.0    11.1   11.1
 A04            0    91  NO    2.0      97.4    97.4    0.0    10.8   10.8
 A05            0    18  NO    2.0      19.4    19.7    0.0     2.2    2.2
 A06            0    70  NO    2.0      74.8    75.0    0.0     8.3    8.3
 A07            0    87  NO    2.0      93.1    93.2    0.0    10.3   10.4
 A08            0    87  NO    2.0      93.1    93.2    0.0    10.3   10.4
 A09            0    46  NO    1.0      99.6    99.6    0.0     5.5    5.5
 A1A            0     0  NO    2.0       0.4     0.4    0.0     0.0    0.0
 A1B            0     7  NO    2.0       7.4     7.8    0.0     0.8    0.9
 A11            0     4  NO    2.0       4.2     4.4    0.0     0.5    0.5
 A12            0     5  NO    2.0       5.1     5.4    0.0     0.6    0.6
 A13            0     4  NO    2.0       3.5     3.8    0.0     0.4    0.4
 A14            0     2  NO    2.0       1.6     1.7    0.0     0.2    0.2
 A17            0     0  NO    2.0       0.1     0.1    0.0     0.0    0.0
 A18            0     0  NO    1.0       0.1     0.1    0.0     0.0    0.0
 A19           10     1  NO    2.0       0.5     0.6    0.0     0.1    0.1
 PHYSICAL                                               2.3            2.3
```

Example 9-5 shows the Monitor I partition data report: The total load is 95.64%. The weight values and the number of logical CPUs do not vary for all the partitions.

Example 9-5   LPAR.1, Monitor I Partition Data Report

```
                                               P A R T I T I O N   D A T A   R E P O R T
                                                                                                                            PAGE    2
             z/OS V1R5                SYSTEM ID SC69           DATE 11/07/2004           INTERVAL 14.59.959
                                      RPT VERSION V1R5 RMF     TIME 16.00.00             CYCLE 1.000 SECONDS

 MVS PARTITION NAME                       A0A                  NUMBER OF PHYSICAL PROCESSORS          24
 IMAGE CAPACITY                           744                                     CP                  18
 NUMBER OF CONFIGURED PARTITIONS           29                                     ICF                  6
 WAIT COMPLETION                           NO
 DISPATCH INTERVAL                    DYNAMIC
---------- PARTITION DATA -----------------  -- LOGICAL PARTITION PROCESSOR DATA --   -- AVERAGE PROCESSOR UTILIZATION PERCENTAGES --
                    ----MSU----  -CAPPING--  PROCESSOR-  ----DISPATCH TIME DATA----   LOGICAL PROCESSORS  --- PHYSICAL PROCESSORS ---
 NAME       S    WGT DEF     ACT DEF    WLM% NUM  TYPE    EFFECTIVE        TOTAL      EFFECTIVE     TOTAL LPAR MGMT  EFFECTIVE  TOTAL

 A0A        A     10   0      87 NO      0.0   2   CP    00.27.55.385  00.27.57.235        93.08    93.18      0.01      10.34  10.35
 A0B        A     10   0       3 NO      0.0   2   CP    00.00.57.700  00.00.59.484         3.21     3.30      0.01       0.36   0.37
 A0C        A     10   0       4 NO      0.0   3   CP    00.01.21.206  00.01.26.893         3.01     3.22      0.04       0.50   0.54
 A01        A     10   0      87 NO      0.0   2   CP    00.27.55.898  00.27.57.731        93.11    93.21      0.01      10.35  10.36
 A02        A     10   0      87 NO      0.0   2   CP    00.27.56.104  00.27.57.944        93.12    93.22      0.01      10.35  10.36
 A03        A    185   0      93 NO      0.0   2   CP    00.29.58.891  00.29.58.891        99.94    99.94      0.00      11.10  11.10
 A04        A     50   0      90 NO      0.0   2   CP    00.29.10.886  00.29.11.018        97.28    97.28      0.00      10.81  10.81
 A05        A     40   0      18 NO      0.0   2   CP    00.05.49.350  00.05.54.617        19.41    19.70      0.03       2.16   2.19
 A06        A     50   0      70 NO      0.0   2   CP    00.22.27.404  00.22.29.324        74.86    74.97      0.01       8.32   8.33
 A07        A     10   0      87 NO      0.0   2   CP    00.27.55.881  00.27.57.725        93.11    93.21      0.01      10.35  10.36
 A08        A     10   0      87 NO      0.0   2   CP    00.27.55.866  00.27.57.708        93.11    93.21      0.01      10.35  10.36
 A09        A    185   0      46 NO      0.0   1   CP    00.14.56.834  00.14.56.839        99.65    99.65      0.00       5.54   5.54
 A1A        A     10   0       0 NO      0.0   2   CP    00.00.07.586  00.00.07.686         0.42     0.43      0.00       0.05   0.05
 A1B        A     10   0       7 NO      0.0   2   CP    00.02.13.522  00.02.20.504         7.42     7.81      0.04       0.82   0.87
 A11        A     10   0       4 NO      0.0   2   CP    00.01.15.293  00.01.20.301         4.18     4.46      0.03       0.46   0.50
 A12        A     10   0       5 NO      0.0   2   CP    00.01.32.920  00.01.38.127         5.16     5.45      0.03       0.57   0.61
 A13        A     10   0       4 NO      0.0   2   CP    00.01.03.737  00.01.08.928         3.54     3.83      0.03       0.39   0.43
 A14        A     10   0       2 NO      0.0   2   CP    00.00.29.504  00.00.31.271         1.64     1.74      0.01       0.18   0.19
 A17        A     10   0       0 NO      0.0   2   CP    00.00.00.993  00.00.01.020         0.06     0.06      0.00       0.01   0.01
 A18        A     10   0       0 NO      0.0   1   CP    00.00.00.672  00.00.00.674         0.07     0.07      0.00       0.00   0.00
 A19        A    185  10       1 NO      0.0   2   CP    00.00.08.024  00.00.10.605         0.45     0.59      0.02       0.05   0.07
 *PHYSICAL*                                                            00.06.08.873                             2.28              2.28
                                                         ------------  ------------                           ------      ------ ------
   TOTAL                                                 04.11.13.667  04.18.13.407                             2.59      93.05  95.64
```

Our LPAR cluster includes the partitions: A0A, A01, A02, A03, A07, A08, and A09. Their weight values (WGT) are, respectively: 10, 10, 10, 185, 10, 10, and 185.

Example 9-6 shows the Summary report with the execution velocities of the service classes, the desired goals, and the current values.

Example 9-6   Goals.1, Monitor III Sysplex Summary Report

```
                      RMF V1R5   Sysplex Summary - WTSCPLX1       Line 1 of 24
Command ===>                                                   Scroll ===> CSR

WLM Samples: 387     Systems: 15 Date: 11/07/04 Time: 16.00.00 Range: 60    Sec

                      >>>>>>>>XXXXXXXXXXXXXXXXXX<<<<<<<<

Service Definition: SAP414               Installed at: 10/25/04, 17.39.00
     Active Policy: SPSTPC               Activated at: 10/25/04, 17.39.38
               ------- Goals versus Actuals --------  Trans --Avg. Resp. Time-
               Exec Vel  --- Response Time ---  Perf  Ended  WAIT EXECUT ACTUAL
Name     T  I  Goal Act  ---Goal--- --Actual--  Indx  Rate   Time   Time   Time

BATCH    W            14                               0.000 0.000  0.000  0.000
BATCHLOW S  5    25  14                         1.74  0.000 0.000  0.000  0.000
CICS     W           N/A                               0.150 0.000  0.001  0.001
CICSDFLT S  2        N/A  0.015 99%        100%  0.60  0.150 0.000  0.001  0.001
DDF      W            40                               0.330 0.000  0.190  0.190
SAPHIGH  S  2    55 0.0                          N/A  0.120 0.000  0.042  0.043
SAPLOW   S  3    25  40                         0.63  0.210 0.000  0.275  0.275
OTHER    W            62                               0.050 0.090  10.72  10.81
```

```
TSO      S          0.0                               0.020 0.000  0.003  0.003
         1  2    80 0.0                          N/A  0.020 0.000  0.003  0.003
VEL70    S  1    70 5.7                         12.3  0.020 0.002  1.366  1.367
SYSTEM   W          N/A                               0.817 0.022  0.150  0.172
SYSSTC   S      N/A N/A    N/A                        0.817
SYSTEM   S      N/A  17    N/A                        0.000 0.000  0.000  0.000
WAS      W          N/A                               0.060 0.000  0.049  0.050
WASDF    S  1       N/A  0.350 90%        100%  0.50  0.060 0.000  0.049  0.050
WBIFN    W          5.7                               0.000 0.000  0.000  0.000
VELRRS   S  1    95  18                         5.32  0.000 0.000  0.000  0.000
VEL80    S  1    80 1.3                         61.6  0.000 0.000  0.000  0.000
```

Recall that the Sysplex Summary Report is an RMF report with global sysplex WLM information. Now that we have created a model for our test case, we can activate IRD and see how things change in our cluster.

IRD and SC69/A0A with important work

At this point, we activate IRD (WLM Weight Management) for the partitions A0A and A01. The weight values declared in HMC are: 10 (initial), 1(minimum), and 999 (maximum) for both partitions. And we start running our more important work on partition A0A, in service class VEL50.

As a consequence, at 16.30, Example 9-7 shows that the CPC is now 95.7% utilized.

Example 9-7 Partitions.2, Monitor III CPC Report

```
                          RMF V1R5   CPC Capacity                  Line 1 of 34
Command ===>                                                   Scroll ===> CSR
Samples: 574     System: SC69  Date: 11/07/04  Time: 16.30.00  Range: 600   Sec
 Partition:   A0A        2084 Model 318
 CPC Capacity:     837   Weight % of Max:  1.9       4h MSU Average:    86
 Image Capacity:   744   WLM Capping %:   ****       4h MSU Maximum:    92

 Partition  --- MSU ---  Cap  Proc    Logical Util %   - Physical Util % -
              Def   Act  Def   Num    Effect   Total   LPAR  Effect   Total
 *CP                                                    2.7    92.9    95.7
 A0A            0    92  NO    2.0      99.3    99.3    0.0    11.0    11.0
 A0B            0     3  NO    2.0       3.2     3.3    0.0     0.4     0.4
 A0C            0     4  NO    3.0       2.9     3.1    0.0     0.5     0.5
 A01            0    75  NO    2.0      80.2    80.4    0.0     8.9     8.9
 A02            0    88  NO    2.0      94.9    95.0    0.0    10.5    10.6
 A03            0    93  NO    2.0      99.9    99.9    0.0    11.1    11.1
 A04            0    90  NO    2.0      97.2    97.2    0.0    10.8    10.8
 A05            0    18  NO    2.0      19.5    19.8    0.0     2.2     2.2
 A06            0    70  NO    2.0      75.0    75.1    0.0     8.3     8.3
 A07            0    88  NO    2.0      94.9    95.0    0.0    10.5    10.6
 A08            0    88  NO    2.0      94.9    95.0    0.0    10.5    10.6
 A09            0    46  NO    1.0      99.6    99.6    0.0     5.5     5.5
 A1A            0     0  NO    2.0       0.4     0.4    0.0     0.0     0.0
 A1B            0     7  NO    2.0       7.5     7.9    0.0     0.8     0.9
 A11            0     4  NO    2.0       4.2     4.4    0.0     0.5     0.5
 A12            0     5  NO    2.0       5.2     5.5    0.0     0.6     0.6
 A13            0     4  NO    2.0       3.6     3.9    0.0     0.4     0.4
 A14            0     2  NO    2.0       1.6     1.7    0.0     0.2     0.2
 A17            0     0  NO    2.0       0.0     0.1    0.0     0.0     0.0
 A18            0     0  NO    1.0       0.1     0.1    0.0     0.0     0.0
 A19           10     1  NO    2.0       0.4     0.6    0.0     0.0     0.1
 PHYSICAL                                               2.4             2.4
```

Example 9-8 is a Partition Data report and at 16.30 shows, in the upper part, the CPC is 95.64% loaded. The numbers in this report just confirm the data reported in the Monitor III CPC report. Partition A01, which is running the less important load, is less loaded with 8.9% of the CPC (logical CPU utilization of 80.4%). It is getting less CPU. Partition A0A, which is now running the more important load, is fully loaded, with 11.0% of the CPC (logical CPU utilization of 99.3%). It is getting more CPU now, but from now on, no more is available.

Example 9-8 LPAR.2 Monitor I Partition Data Report and LPAR Cluster Report

```
                                               P A R T I T I O N   D A T A   R E P O R T
                                                                                                                            PAGE    2
             z/OS V1R5                SYSTEM ID SC69             DATE 11/07/2004            INTERVAL 14.59.995
                                      RPT VERSION V1R5 RMF       TIME 16.30.00              CYCLE 1.000 SECONDS

 MVS PARTITION NAME                     A0A                      NUMBER OF PHYSICAL PROCESSORS           24
 IMAGE CAPACITY                         744                                         CP                   18
 NUMBER OF CONFIGURED PARTITIONS         29                                         ICF                   6
 WAIT COMPLETION                         NO
 DISPATCH INTERVAL                  DYNAMIC
---------- PARTITION DATA -----------------  -- LOGICAL PARTITION PROCESSOR DATA --   -- AVERAGE PROCESSOR UTILIZATION PERCENTAGES --
                    ----MSU----  -CAPPING--  PROCESSOR-  ----DISPATCH TIME DATA----   LOGICAL PROCESSORS  --- PHYSICAL PROCESSORS ---
 NAME       S   WGT  DEF    ACT  DEF   WLM%  NUM   TYPE   EFFECTIVE       TOTAL       EFFECTIVE    TOTAL  LPAR MGMT  EFFECTIVE  TOTAL

 A0A        A    19    0     92  NO     0.0  2.0   CP    00.29.46.940  00.29.47.310       99.28    99.30      0.00      11.03  11.03
 A0B        A    10    0      3  NO     0.0    2   CP    00.00.58.358  00.01.00.154        3.24     3.34      0.01       0.36   0.37
 A0C        A    10    0      4  NO     0.0    3   CP    00.01.19.241  00.01.24.939        2.93     3.15      0.04       0.49   0.52
 A01        A     1    0     75  NO     0.0  2.0   CP    00.24.04.345  00.24.08.654       80.24    80.48      0.03       8.92   8.94
 A02        A    10    0     88  NO     0.0    2   CP    00.28.29.039  00.28.30.947       94.95    95.05      0.01      10.55  10.56
 A03        A   185    0     93  NO     0.0    2   CP    00.29.58.444  00.29.58.444       99.91    99.91      0.00      11.10  11.10
 A04        A    50    0     90  NO     0.0    2   CP    00.29.05.148  00.29.05.407       96.95    96.97      0.00      10.77  10.77
 A05        A    40    0     18  NO     0.0    2   CP    00.05.52.154  00.05.57.473       19.56    19.86      0.03       2.17   2.21
 A06        A    50    0     70  NO     0.0    2   CP    00.22.28.968  00.22.30.927       74.94    75.05      0.01       8.33   8.34
 A07        A    10    0     88  NO     0.0    2   CP    00.28.28.987  00.28.30.896       94.94    95.05      0.01      10.55  10.56
 A08        A    10    0     88  NO     0.0    2   CP    00.28.29.032  00.28.30.946       94.95    95.05      0.01      10.55  10.56
 A09        A   185    0     46  NO     0.0    1   CP    00.14.56.627  00.14.56.627       99.63    99.63      0.00       5.53   5.53
 A1A        A    10    0      0  NO     0.0    2   CP    00.00.07.583  00.00.07.686        0.42     0.43      0.00       0.05   0.05
 A1B        A    10    0      7  NO     0.0    2   CP    00.02.13.613  00.02.20.772        7.42     7.82      0.04       0.82   0.87
 A11        A    10    0      4  NO     0.0    2   CP    00.01.14.645  00.01.19.703        4.15     4.43      0.03       0.46   0.49
 A12        A    10    0      5  NO     0.0    2   CP    00.01.34.176  00.01.39.381        5.23     5.52      0.03       0.58   0.61
 A13        A    10    0      4  NO     0.0    2   CP    00.01.04.507  00.01.09.675        3.58     3.87      0.03       0.40   0.43
 A14        A    10    0      2  NO     0.0    2   CP    00.00.29.472  00.00.31.247        1.64     1.74      0.01       0.18   0.19
 A17        A    10    0      0  NO     0.0    2   CP    00.00.00.870  00.00.00.895        0.05     0.05      0.00       0.01   0.01
 A18        A    10    0      0  NO     0.0    1   CP    00.00.00.672  00.00.00.674        0.07     0.07      0.00       0.00   0.00
 A19        A   185   10      1  NO     0.0    2   CP    00.00.07.904  00.00.10.555        0.44     0.59      0.02       0.05   0.07
 *PHYSICAL*                                                            00.06.30.817                           2.41              2.41
                                                         ------------  ------------                         ------     ------ ------
   TOTAL                                                 04.10.50.736  04.18.14.137                           2.74      92.91  95.64
....
                                                 L P A R   C L U S T E R   R E P O R T
                                                                                                                            PAGE    4
             z/OS V1R5                SYSTEM ID SC69             DATE 11/07/2004            INTERVAL 14.59.995
                                      RPT VERSION V1R5 RMF       TIME 16.30.00              CYCLE 1.000 SECONDS

                               ------ WEIGHTING STATISTICS ------   ---- PROCESSOR STATISTICS ----   ------ STORAGE STATISTICS -----
                               --- DEFINED ---  ---- ACTUAL -----   ---- NUMBER ---   -- TOTAL% --   --- CENTRAL ---   -- EXPANDED --
 CLUSTER   PARTITION  SYSTEM    INIT  MIN  MAX  AVG  MIN %  MAX %   DEFINED  ACTUAL   LBUSY  PBUSY
....
 WTSCPLX1  A0A        SC69        10    1  999   19    0.0    0.0        16     2.0   99.30  11.03          4096          N/A
           A01        SC55        10    1  999    1    100    0.0         2     2.0   80.48   8.94          2048          N/A
           A02        SC54        10             10                       2       2   95.05  10.56          2048          N/A
           A03        SC49       185            185                       2       2   99.91  11.10          4096          N/A
           A07        SC52        10             10                       2       2   95.05  10.56          2048          N/A
           A08        SC48        10             10                       2       2   95.05  10.56          2048          N/A
           A09        SC47       185            185                       2       1   99.63   5.53          4096          N/A
 ---------------------------   ----------------------------------   ------------------------------   -------------------------------
                       TOTAL     420                                     28           664.5  68.30         20480          N/A
```

However, in the bottom part of the report named LPAR Cluster report, the weight (WGT) values for A0A and A01 are changed. Their actual weight values are changed from the initial values. For partition A01, it is equal to the minimum 1. For partition A0A is 19. The amount of weight lost by the donor (A01) is equal to the amount received by the receiver (A0A).

Example 9-9 shows that the velocity goal for the service class VEL50 is not achieved; the actual velocity is only 17%, but the IRD is helping it as much as it can.

The reason IRD raised the weight of A0A is that the VEL50 service class running in A0A was unhappy (PI > 1.0), mainly because of a delay for CPU. The first attempt was to increase the dispatching priority, but this did not produce much relief in the CPU congestion on A0A. Then, the next decision was to increase the weight of A0A, taking it from A01 where there are less important service class that will suffer.

As you can see, in this interval, WLM decides not to activate the IRD function WLM Vary CPU Management. You can see this because the field `Number - Actual` in the Processor Statistics section of the LPAR Cluster Report has integer numbers.

Example 9-9 Goals.2, Monitor III Sysplex Summary Report

```
                      RMF V1R5   Sysplex Summary - WTSCPLX1       Line 1 of 24
Command ===>                                                   Scroll ===> CSR

WLM Samples: 388     Systems: 15 Date: 11/07/04 Time: 16.30.00 Range: 60    Sec

                      >>>>>>>>XXXXXXXXXXXXXXXXXX<<<<<<<<

Service Definition: SAP414                Installed at: 10/25/04, 17.39.00
     Active Policy: SPSTPC                Activated at: 10/25/04, 17.39.38
               ------- Goals versus Actuals --------  Trans --Avg. Resp. Time-
               Exec Vel  --- Response Time ---  Perf  Ended  WAIT EXECUT ACTUAL
Name     T  I  Goal Act  ---Goal--- --Actual--  Indx  Rate   Time   Time   Time

BATCH    W           17                               0.000 0.000  0.000  0.000
BATCHLOW S  5    25  17                         1.51  0.000 0.000  0.000  0.000
CICS     W          N/A                               0.150 0.000  0.001  0.001
CICSDFLT S  2       N/A  0.015 99%        100%  0.60  0.150 0.000  0.001  0.001
DDF      W          0.0                               0.250 0.000  0.051  0.051
SAPHIGH  S  2    55 0.0                          N/A  0.130 0.000  0.019  0.019
SAPLOW   S  3    25 0.0                          N/A  0.120 0.000  0.086  0.086
OTHER    W           17                               0.040 0.001  0.690  0.691
TSO      S          0.0                               0.020 0.000  0.003  0.003
         1  2    80 0.0                          N/A  0.020 0.000  0.003  0.003
VEL50    S  1    50  17                         2.95  0.000
VEL70    S  1    70  15                         4.62  0.020 0.002  1.378  1.379
SYSTEM   W          N/A                               1.093 0.004  0.074  0.079
SYSSTC   S      N/A N/A    N/A                        1.093
SYSTEM   S      N/A  16    N/A                        0.000 0.000  0.000  0.000
WAS      W          N/A                               0.120 0.000  0.102  0.102
WASDF    S  1       N/A  0.350 90%         92%  0.80  0.120 0.000  0.102  0.102
WBIFN    W          6.4                               0.000 0.000  0.000  0.000
VELRRS   S  1    95  19                         4.99  0.000 0.000  0.000  0.000
VEL80    S  1    80 2.7                         29.2  0.000 0.000  0.000  0.000
```

9.2.2 IRD and capping

IRD WLM CPU management allows WLC's soft capping or WLM resource group capping, but prohibits traditional physical LPAR capping.

9.3 Analyzing crypto performance

Cryptographic processing in the PCICC/PCIXCC and PCICA cards on the CPC, and reporting with RMF are new areas. There are several updates required for the CPC hardware and software in order to get all the information on the RMF reports. This section describes a sample case on crypto.

9.3.1 Crypto and workload reports

In Example 9-10, we have a CPC with heavy use of crypto hardware. This is a z990 with two PCIXCC-cards heavily loaded, with utilizations of 97.0% and 96.8%. Processing times are 2.5 ms (milliseconds) and the processing rates are 392.6 and 391.0 requests per second. Obviously, these cards are fully loaded.

The cards are serving DES-encryption requests via the ICSF software from the application, as you can see in the final part of the report. The application is generating 783.6 requests per second, using 8000 byte data blocks. With two PCIXCC-cards, we should be able to handle about 780 requests per second.

Example 9-10 Postprocessor Crypto Hardware Activity reports

```
                                            C R Y P T O   H A R D W A R E   A C T I V I T Y
                                                                                                                           PAGE    1
             z/OS V1R5                SYSTEM ID SC69             DATE 10/30/2004            INTERVAL 14.59.996
                                      RPT VERSION V1R5 RMF       TIME 10.00.00              CYCLE 1.000 SECONDS
------- CRYPTOGRAPHIC COPROCESSOR --------
          -------- TOTAL --------  KEY-GEN
TYPE   ID   RATE  EXEC TIME  UTIL%    RATE
PCIXCC  2  392.6      2.5    97.0     0.00
        3  391.0      2.5    96.8     0.00

 -------- CRYPTOGRAPHIC ACCELERATOR ---------------------------------------------------------------------------------------------
            -------- TOTAL -------  ------- ME(1024) -----  ----- ME(2048) ------  ------ CRT(1024) -----  ----- CRT(2048) ------
 TYPE   ID   RATE  EXEC TIME UTIL%   RATE  EXEC TIME UTIL%   RATE EXEC TIME UTIL%   RATE  EXEC TIME UTIL%   RATE  EXEC TIME UTIL%
 PCICA   0   0.00      0.0     0.0   0.00      0.0     0.0   0.00     0.0     0.0   0.00      0.0     0.0   0.00      0.0     0.0
         1   0.00      0.0     0.0   0.00      0.0     0.0   0.00     0.0     0.0   0.00      0.0     0.0   0.00      0.0     0.0

 -------- ICSF SERVICES EXECUTED ON PCIXCC ----------------------------------------------------
         DES ENCRYPTION    DES DECRYPTION    ----- MAC ------    - HASH -    ------ PIN -------
         SINGLE  TRIPLE    SINGLE  TRIPLE    GENERATE  VERIFY                TRANSLATE   VERIFY
 RATE     783.6    0.00      0.00    0.00        0.00    0.00       0.00          0.00     0.00
 SIZE      8000    0.00      0.00    0.00        0.00    0.00       0.00
```

Now let’s take a look at the Postprocessor Workload Activity report in Example 9-11, where WLM does some sampling of address spaces/enclaves in relation to crypto hardware. These samples are consolidated in a service class period basis.

We have four programs in four jobs simultaneously creating DES requests via ICSF to the PCIXCC-cards, so the multiprogramming level (MPL) is 4.00.

They are simultaneously using the crypto facilities (PCIXCC through ICSF) about half of their RMF interval time, 49.9% (field CRYPTO% USG).

- WLM Using Crypto applies to using the processor asynchronously.
- WLM Delay Crypto applies to being in the queue for the asynchronous crypto processors.

All WLM Using and Delay counters shown in this report should add to roughly 100%. Any deviation is because of the possibility of overlap between Using and Delay states. However, Crypto (Using and Delay) are outside of this rule. When a dispatchable unit is using or delayed because of asynchronous crypto hardware, WLM considers it an Unknown state and includes it in such number. However, to help an installation monitor crypto performance, WLM started to sample the Using and Delay states for Crypto, while still considering these states as Unknown.

We see that the application programs are mostly in an unknown state, 98.1% (field DLY% UNKN).

If you suspect crypto performance problems in your installation, you can use the following procedure to do some investigation.

Start the application and monitor with the WLMGL report:

- If the goals are met, everything looks fine.
- If the goals are not achieved, you can check `Crypto Using and Delay` to verify whether crypto might be the cause of the delay.
- In case of `Crypto Delay` or high `Crypto Using`, check the Crypto Activity report.

  PCI-Crypto Measurement Reporting: If there is high utilization on one or more crypto cards, try to shift work to the less utilized crypto cards or order additional hardware.

Example 9-11 Postprocessor Workload Activity Report with Crypto-fields

```
                                               W O R K L O A D   A C T I V I T Y
                                                                                                                     PAGE   1
      z/OS V1R4               SYSPLEX WTSCPLX1           DATE 10/30/2004           INTERVAL 15.00.296   MODE = GOAL
                          CONVERTED TO z/OS V1R5 RMF     TIME 10.00.00

                                        POLICY ACTIVATION DATE/TIME 10/25/2004 17.39.38

------------------------------------------------------------------------------------------------------------ service class PERIODS

REPORT BY: POLICY=SPSTPC     WORKLOAD=BATCH      SERVICE CLASS=BATCRYPT   RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=3
                                                 CRITICAL     =NONE

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--   ---SERVICE----   --SERVICE TIMES--   PAGE-IN RATES    ----STORAGE----
AVG      4.00   ACTUAL                   0   SSCHRT   0.0   IOC         0    TCB          46.7   SINGLE     0.0   AVG      290.75
MPL      4.00   EXECUTION                0   RESP     0.0   CPU      9688K   SRB           0.0   BLOCK      0.0   TOTAL   1162.63
ENDED       0   QUEUED                   0   CONN     0.0   MSO      5638K   RCT           0.0   SHARED     0.0   CENTRAL 1162.63
END/S    0.00   R/S AFFINITY             0   DISC     0.0   SRB       345    IIT           0.0   HSP        0.0   EXPAND     0.00
#SWAPS      0   INELIGIBLE               0   Q+PEND   0.0   TOT     15327K   HST           0.0   HSP MISS   0.0
EXCTD       0   CONVERSION               0   IOSQ     0.0   /SEC    17030    IFA           N/A   EXP SNGL   0.0   SHARED     0.00
AVG ENC  0.00   STD DEV                  0                                   APPL% CP      5.2   EXP BLK    0.0
REM ENC  0.00                                               ABSRPTN  4257    APPL% IFACP   0.0   EXP SHR    0.0
MS ENC   0.00                                               TRX SERV 4257    APPL% IFA     N/A

GOAL: EXECUTION VELOCITY 60.0%     VELOCITY MIGRATION:   I/O MGMT  67.4%     INIT MGMT 67.4%

         RESPONSE TIME EX   PERF  AVG   --- USING% --- ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--    %
SYSTEM                 VEL% INDX ADRSP   CPU  IFA  I/O  TOT  CPU                               UNKN IDLE  USG  DLY  USG  DLY QUIE

SC69        --N/A--    67.4  0.9   0.3   1.3  N/A  0.0  0.6  0.6                               98.1  0.0 49.9  0.0  0.0  0.0  0.0
```

9.4 zAAP on RMF reports

zSeries Application Assist Processor (zAAP) is a new type of characterizable processor (PU) that allows the running of an specific type of work (Java Virtual machine - JVM) under control of the z/OS 1.6 operating system. zAAP is also referred to as Integrated Facility for Application (IFA). With the introduction of zAAP, z/OS now recognizes two types of logical processors, one for standard CPUs and one for zAAPs.

It allows saving of software fees in WLC delivering additional Java capacity without affecting the total MSU/hour of the CEC, and there is no need to rewrite Java code to benefit this processor. zAAP CP usually does not run z/OS code and cannot be used in IPLs.

The JVM code running in a task has been enhanced to signal the dispatcher that zAAP-eligible work is now starting to execute. From this moment on, this JVM task is placed in an specific zAAP dispatcher queue. This task is then called JVM task, or zAAP task, or zAAP affinity task. When a zAAP CPU becomes available, the dispatcher running on it selects a JVM task from the zAAP dispatcher queue and dispatches it. This JVM task is also eligible to be processed on standard CPUs, depending on the installation options that can be set through Parmlib IEAOPT parameters. However, the zAAP CPU should not execute a non-JVM dispatchable unit.

Here are the IEAOPT parmlib options to control the zAAP processing:

IFACrossover = Yes (default): Eligible JVM code can run both in zAAP logical CPUs or in standard logical CPUs. This option improves Java performance, but pays more in LP subcapacity (WLC).

IFACrossover = No: Eligible JVM code only runs in zAAP logical CPUs in the LP, unless the zAAP CPs are not operational.

IFAHonorPriority = Yes (default): This keyword only applies if IFACrossover is equal to Yes. WLM instructs the z/OS dispatcher (when dispatching zAAP affinity task in standard logical CPs) to follow the exact dispatching priority of all dispatchable units—with or without zAAP affinity.

IFAHonorPriority = No: This keyword only applies if IFACrossover is equal to Yes. WLM instructs the z/OS dispatcher (when dispatching zAAP affinity tasks in standard logical CPs) to place dispatchable units with zAAP affinity with the lowest dispatching priority. However, the dispatching priority of the zAAP affinity dispatchable units is respected when dispatching on the zAAP processor.

9.4.1 A sample scenario

In this section, we describe a scenario with zAAP-eligible work running in a system and we show several reports; then we discuss a planning tool. For more information about zAAP, refer to redbook *zSeries Application Assist Processor (zAAP) Implementation,* SG24-6386.

Following is a theoretical discussion showing an LP with more zAAP-eligible work than just one zAAP can handle. We can see how the IEAOPT parmlib parameters might affect the loads and how RMF reports these loads.

We are running in an LPAR with one standard CPU and one zAAP. During a 10 minute (600 seconds) period, we generate a total of 1200 seconds of work, specifically:

- 840 seconds of zAAP eligible work, corresponding to 140% utilization.
- 360 seconds of non-zAAP eligible work, corresponding to 60% utilization.

Table 9-1 shows where the work could be processed: on zAAP or on standard CP. Keep in mind that this table shows purely theoretical numbers.

Table 9-1 Work assignment: zAAP, standard CP or queue

IFA Cross Over	**IFA Honor Priority**	**Eligible to zAAP**	**zAAP CP load**	**zAAP CP queued**	**Move to standard CP - Same priority**	**Move to standard CP - Lower priority**	**Non-eligi ble On standard CP**	**Total load on standard CP**
Yes	Yes	140%	100%	0	40%	0	60%	100%
Yes	No	140%	100%	0	0	40%	60%	100%
No	No	140%	100%	40%	0	0	60%	60%
When zAAP is offline								
Yes	Yes	140%	N/A		140% but some queuing	0	60% but some queuing	100% plus 100% queuing
Yes/No	No	140%	N/A		140% but 100% queued	0	60%	100% plus 100% queued

9.4.2 Using and Delay samples

The JVM tasks which are eligible to run on a zAAP processor can also be executed on a standard CPU. Therefore, three new CPU Using and Delay states are introduced in addition to the current AS/enclaves CPU Using and Delay states. The CPU states are:

IFA using JVM dispatchable unit is found executing on an zAAP

IFA on CP using JVM dispatchable unit is executed on a standard CPU

IFA delay JVM dispatchable unit is delayed for an zAAP

CPU using Dispatchable unit is found executing on a standard CPU (existing)

CPU delay Dispatchable unit is delayed for a standard CPU (existing)

Figure 9-2 illustrates a numeric example with zAAP-eligible work, on the left, arriving to the zAAP processor and non-eligible work, on the right, arriving to the standard CPU. The zAAP processor is on the left and its queue is below it. The standard CPU is on the right and its queue is below it.

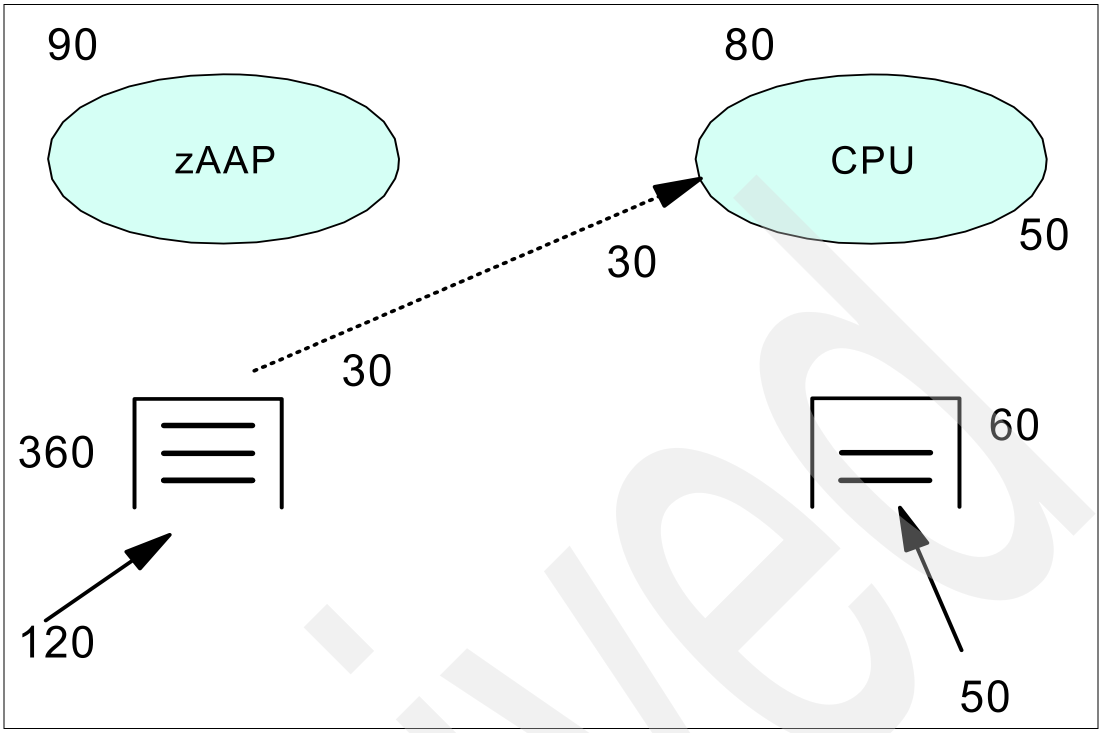


Figure 9-2 zAAP processor and standard CP

Considering these numbers from the bottom up, the meanings of the values are as follows:

120 The rate of zAAP eligible work arriving to the zAAP queue.

50 The rate of non-eligible work arriving to the standard CPU queue.

360 The amount of work in the queue to the zAAP processor, zAAP delay samples.

60 The amount of work in the queue to the standard CPU, CPU delay samples.

90 The amount of zAAP eligible work on the zAAP processor, zAAP using samples, load (utilization) is 90%.

80 The amount of work in the standard processor, CPU using samples, load is 70%. This is the sum of 50 plus 20.

50 The amount of non-eligible work in the standard processor.

30 The amount of zAAP eligible work in the standard processor, zAAP on CPU using samples.

30 The amount of zAAP eligible work taken by dispatcher to the standard CPU.

The work which is eligible to run on zAAPs, but which is executed on standard CPs, “IFA on CP using,” must be regarded with respect as to whether the dispatcher honors priorities for selecting work from the zAAP work unit queue or not.

- When the dispatcher does not honor priority, check why the installation has it turned off.

  “IFA on CP using” samples are added to the “IFA using” samples.

- If the dispatcher honors the priority:
  - “IFA on CP using” samples are also contained in the “CPU using” samples.
  - The amount of “IFA delay” samples is reduced (proportionally) by the amount of “IFA on CP using” samples.

So, the calculations are done as follows:

CPU using = CPU using + IFA_on_CP using

All IFA using = MAX(1, IFA using + IFA_on_CP using)

Conv IFA delays = IFA delay x IFA_on_CP using / All IFA using

CPU delay = CPU delay + Conv IFA delays

IFA delay = IFA delay - Conv IFA delays

- Thus, “IFA on CP using” samples are always contained in either “CPU using” samples or “IFA using” samples.

Execution velocity

The “IFA using” and “IFA delay” samples are included in the WLM execution velocity goal.

9.4.3 RMF reports

We now take a look at some RMF reports. Our system has the default IEAOPT parmlib parameter settings, IFACrossOver=Yes and IFAHonorPriority=Yes. Our LPAR A13 has one zAAP processor and two standard processors. We have two zAAP-eligible, processor-intensive batch jobs. Additionally, we have one mostly idle zAAP-eligible batch job. All of them are running in a service class OMVSLOW.

zAAP on the Monitor III CPC Capacity report

First we see the load and other values in the Monitor III CPC report in Example 9-12. This report splits each LP in two sets of data:

- CP PU pool: formed by standard CPUs, in the *CP portion of the report. All the MSU/hour consumed by this pool is used in WLC account.
- ICF PU pool: formed by: ICF, IFL, zAAP, in the *ICF portion of the report. There is no MSU/hour calculation for this pool.

Example 9-12 Monitor III CPC Capacity

```
                           RMF V1R5   CPC Capacity                  Line 1 of 34
Command ===>                                                  Scroll ===> PAGE

Samples: 60      System: SC70  Date: 11/08/04  Time: 20.30.00  Range: 60    Sec
 Partition:   A13        2084 Model 318
 CPC Capacity:     837   Weight % of Max: ****       4h MSU Average:    19
 Image Capacity:   837   WLM Capping %:   ****       4h MSU Maximum:    91

 Partition  --- MSU ---  Cap  Proc    Logical Util %   - Physical Util % -
              Def   Act  Def   Num    Effect   Total   LPAR  Effect  Total
 *CP                                                    4.9    36.7   41.6
 AOA            0    11  NO    2.0      11.9    12.2    0.0     1.3    1.4
...
 A13            0    49  NO    2.0      52.6    52.7    0.0     5.8    5.9
...
 A19           10     0  NO    2.0       0.4     0.5    0.0     0.0    0.1
 PHYSICAL                                               4.4            4.4

 *ICF                                                   1.4    46.6   48.1
...
 A1F                     NO    1.0       3.6     3.7    0.0     0.6    0.6
 A13                     NO    1.0      97.1    97.2    0.0    16.2   16.2
 PHYSICAL                                               1.4            1.4
```

In Example 9-12, we see:

- The total physical CP pool (standard CPUs only) utilization in our CPC is 41.6%.
- Our LP A13 has two logical standard CPUs; its physical utilization is 5.9% of the CP pool (logical utilization is 52.7%).
- The total physical ICF pool (ICF, IFL and zAAP) utilization in our CPC is 48.1%.
- Our LP A13 has one logical zAAP, its physical utilization is 16.2% of the ICF pool (logical utilization is 97.2%). This 16.2% is not really important because it does not apply to zAAPs only, but to the total consumption of IFLs, ICFs and zAAPs in the CPC. However, the 97.2% figure is indicating that this logical zAAP processor is very busy.

zAAP on the Monitor III System Information report

Now, we look at the Monitor III System Information report. Remember that this report is local.

Example 9-13 Monitor III System Information

```
                           RMF V1R5   System Information            Line 1 of 18
Command ===>                                                  Scroll ===> PAGE

Samples: 60      System: SC70  Date: 11/08/04  Time: 20.30.00  Range: 60    Sec
 Partition:    A13     2084 Model 318         Appl%:      50  Policy: WLMPOL
 CPs Online:   2.0     Avg CPU Util%:   53     EAppl%:     51  Date:   11/08/04
 IFAs Online:  1.0     Avg MVS Util%:   53     Appl% IFA:  97  Time:   18.37.41

 Group     T WFL --Users--  RESP TRANS -AVG USG-   -Average Number Delayed For -
              %  TOT  ACT  Time  /SEC PROC  DEV   PROC  DEV STOR SUBS OPER  ENQ

 *SYSTEM      94   97    2        0.02  2.1  0.2    0.1  0.1  0.0  0.0  0.0  0.0
 *TSO              3    0        0.02  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 *BATCH            1    0        0.00  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 *STC         76   84    0        0.00  0.0  0.2    0.0  0.1  0.0  0.0  0.0  0.0
 *ASCH             0    0        0.00  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 *OMVS        96    9    2        0.00  2.1  0.0    0.1  0.0  0.0  0.0  0.0  0.0
 *ENCLAVE          0  N/A         N/A  0.0  N/A    0.0  N/A  0.0  N/A  N/A  N/A
 BATCH     W  96    6    2  .000  0.00  2.1  0.0    0.1  0.0  0.0  0.0  0.0  0.0
 BATCH     S        1    0  .000  0.00  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 OMVSLOW   S  96    5    2  .000  0.00  2.1  0.0    0.1  0.0  0.0  0.0  0.0  0.0
 STCTASKS  W 100   15    0  3.10  0.02  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 OMVS      S        4    0  .000  0.00  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 STC       S 100   11    0  3.10  0.02  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 SYSTEM    W  73   73    0  .000  0.00  0.0  0.2    0.0  0.1  0.0  0.0  0.0  0.0
 SYSSTC    S  75   52    0  .000  0.00  0.0  0.1    0.0  0.0  0.0  0.0  0.0  0.0
 SYSTEM    S  71   21    0  .000  0.00  0.0  0.1    0.0  0.0  0.0  0.0  0.0  0.0
 TSO       W        3    0  .050  0.02  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
 TSO       S        3    0  .050  0.02  0.0  0.0    0.0  0.0  0.0  0.0  0.0  0.0
```

In Example 9-13, we see:

- The system name SC70 runs in the LP A13.
- We have 2.0 standard CPU processors online.
- We have 1.0 zAAP processor online.
- The average standard CPU utilization is 53% on the standard processors (CP pool).
- The average application CPU utilization is 50% on the standard CPU processors. It is not included in the zAAP processing.
- The average zAAP processor application utilization is 97%.

- OMVSLOW service class:
  - Has about two non-idle address spaces.
  - On average, 2.1 address spaces are in the Using state, including using zAAP.
  - On average, 0.1 address spaces are delayed by CPU, including delay per zAAP.

zAAP on the Monitor III Processor Delays report

We next consider what the Monitor III Processor Delays report can show us. It has local data captured by Monitor III.

Example 9-14 Monitor III Processor Delays

```
                           RMF V1R5   Processor Delays             Line 1 of 16
Command ===>                                                     Scroll ===> PAGE

Samples: 60      System: SC70  Date: 11/08/04  Time: 20.30.00  Range: 60    Sec
             Service  DLY USG  Appl EAppl ----------- Holding Job(s) -----------
 Jobname  CX Class     %   %      %     %  %  Name      %  Name      %  Name

 CASSIER9 O  OMVSLOW    3  97  96.5  96.5   3 RAIMO9     3 CASSIER3
 RAIMO9   O  OMVSLOW    3  97  96.4  96.4   3 CASSIER3   3 CASSIER9
 CASSIER3 O  OMVSLOW    2  13   0.4   0.4   2 RAIMO9     2 CASSIER9
 XCFAS    S  SYSTEM     0   2   0.9   0.9
 *MASTER* S  SYSTEM     0   0   0.3   0.3
 GRS      S  SYSTEM     0   0   0.1   0.1
 SMSVSAM  S  SYSTEM     0   0   0.1   0.1
 WLM      S  SYSTEM     0   0   0.2   0.2
 SMS      S  SYSSTC     0   0   0.1   0.1
 RMF      S  SYSSTC     0   0   0.5   0.5
 VTAM44   S  SYSSTC     0   0   0.5   0.5
 RRS      S  STC        0   0   0.1   0.1
 OPTSO    S  STC        0   0   0.1   0.1
 CATALOG  S  SYSTEM     0   0   0.1   0.1
 RMFGAT   SO SYSSTC     0   0   1.7   1.7
 SMFCLEAR S  ********   0   0   0.1   0.1
```

In Example 9-14, we see three address spaces in the OMVSLOW service class. The DLY% and USG% figures include delay and using the zAAP PUs.

Observe that Appl% in this report includes zAAP processing, while on the System Information report Appl% does not include zAAP consumption.

The three address spaces consumed nearly 200% (96.5% + 96.5% + 0.4%) of the processors, both zAAP and standard processors. This was possible because IFACrossover is equal to Yes.

zAAP on the Monitor III Sysplex Summary report

Let’s see now what Monitor III Sysplex Summary report can show. This is a sysplex global report.

Example 9-15 Monitor III Sysplex Summary

```
                         RMF V1R5   Sysplex Summary - SANDBOX        Line 1 of 1
Command ===>                                                  Scroll ===> PAGE

WLM Samples: 240     Systems: 4  Date: 11/08/04 Time: 20.30.00 Range: 60    Sec
 Service Definition: Sampdef               Installed at: 11/08/04, 18.37.33
      Active Policy: WLMPOL                Activated at: 11/08/04, 18.37.41

                ------- Goals versus Actuals --------  Trans --Avg. Resp. Time-
                Exec Vel  --- Response Time ---  Perf  Ended  WAIT EXECUT ACTUAL
 Name     T  I  Goal Act  ---Goal--- --Actual--  Indx  Rate   Time   Time   Time

 BATCH    W           94                               0.000 0.000  0.000  0.000
 OMVSLOW  S  5    20  94                         0.21  0.000 0.000  0.000  0.000
 STCTASKS W           62                               0.467 0.004  0.256  0.261
 OMVS     S           67                               0.450 0.001  0.154  0.156
          1  2        67  1.000 80%        100%  0.50  0.417 0.001  0.143  0.144
          2  4    30 0.0                          N/A  0.033 0.002  0.295  0.296
 STC      S  3    30  62                         0.49  0.017
 SYSTEM   W           53                               0.000 0.000  0.000  0.000
 SYSSTC   S      N/A  59    N/A                        0.000 0.000  0.000  0.000
 SYSTEM   S      N/A  47    N/A                        0.000 0.000  0.000  0.000
 TSO      W           50                               0.733 0.000  0.022  0.022
 TSO      S           50                               0.733 0.000  0.022  0.022
          1  2       0.0  1.000 90%        100%  0.50  0.717 0.000  0.007  0.007
          3  4    20 100                         0.20  0.017 0.000  0.637  0.637
```

In Example 9-15, we see that OMVSLOW service class has nearly no delays in the use of the processors, because actual execution velocity is 94%.

Apparently, the 94% can look a little unexpected because the zAAP processor was very busy (97.2) due to the processing of the two Jobs. The reason it is possible to achieve this percentage is that we are not only using one logical ZAAP processor, but three logical processors (one zAAP and two standard). This is because IFACrossover is equal to Yes in our runs.

zAAP on the Monitor I CPU reports

We next consider what the Monitor I CPU reports, CPU Activity and Partition Data sections show. This report has local data in the first half and CPC global data in the second half.

Example 9-16 Monitor I CPU Activity

```
                                                        C P U  A C T I V I T Y
                                                                                                                            PAGE    1
             z/OS V1R6                SYSTEM ID SC70             DATE 11/08/2004            INTERVAL 09.59.939
                                      RPT VERSION V1R5 RMF       TIME 20.30.00              CYCLE 1.000 SECONDS
 CPU  2084   MODEL  318
 ---CPU---  ONLINE TIME   LPAR BUSY       MVS BUSY      CPU SERIAL  I/O TOTAL          % I/O INTERRUPTS
 NUM  TYPE  PERCENTAGE    TIME PERC       TIME PERC     NUMBER      INTERRUPT RATE     HANDLED VIA TPI
  0   CP    100.00        52.31           52.81         136A3A      16.62               0.41
  1   CP    100.00        52.31           52.82         136A3A      16.25               0.41
 CP   TOTAL/AVERAGE       52.31           52.82                     32.87               0.41
  2   IFA   100.00        98.24           99.37         136A3A
 IFA  AVERAGE             98.24           99.37
...
...                                                 P A R T I T I O N  D A T A  R E P O R T
                                                                                                                            PAGE    2
             z/OS V1R6                SYSTEM ID SC70             DATE 11/08/2004            INTERVAL 09.59.939
                                      RPT VERSION V1R5 RMF       TIME 20.30.00              CYCLE 1.000 SECONDS

 MVS PARTITION NAME                     A13                      NUMBER OF PHYSICAL PROCESSORS           24
 IMAGE CAPACITY                         837                                         CP                   18
 NUMBER OF CONFIGURED PARTITIONS         29                                         ICF                   6
 WAIT COMPLETION                         NO
 DISPATCH INTERVAL                  DYNAMIC
```

```
--------- PARTITION DATA -----------------  -- LOGICAL PARTITION PROCESSOR DATA --   -- AVERAGE PROCESSOR UTILIZATION PERCENTAGES --
                 ----MSU----  -CAPPING--  PROCESSOR-  ----DISPATCH TIME DATA----   LOGICAL PROCESSORS  --- PHYSICAL PROCESSORS ---
NAME       S   WGT  DEF    ACT  DEF   WLM%  NUM   TYPE   EFFECTIVE       TOTAL       EFFECTIVE     TOTAL  LPAR MGMT  EFFECTIVE  TOTAL

A13        A    10    0     49  NO     0.0    2   CP     00.10.26.921  00.10.27.622      52.25     52.31       0.01       5.81   5.81
A0A        A   185    0      6  NO     0.0    2   CP     00.01.19.541  00.01.22.680       6.63      6.89       0.03       0.74   0.77
....
....
A19        A   185   10      0  NO     0.0    2   CP     00.00.04.695  00.00.06.085       0.39      0.51       0.01       0.04   0.06
*PHYSICAL*                                                             00.08.25.686                            4.68              4.68
                                                         ------------  ------------                          ------     ------ ------
  TOTAL                                                  01.01.06.561  01.10.20.879                            5.13      33.95  39.09

A13        A    10                              1   ICF    00.09.49.303  00.09.49.400      98.23     98.24       0.00      16.37  16.37
....
....
A1F        A    10                              1   ICF    00.00.19.327  00.00.19.818       3.22      3.30       0.01       0.54   0.55
*PHYSICAL*                                                             00.00.53.141                            1.48              1.48
                                                         ------------  ------------                          ------     ------ ------
  TOTAL                                                  00.22.03.845  00.22.59.739                            1.55      36.78  38.33
```

In Example 9-16, on the CPU Activity report, we see that the zAAPs appear as ICF processors because the hardware manages ICF, IFL, and zAAPs as one single pool of resources.

- We have two standard logical CPUs, 100.0% of the time being online.
- We have one zAAP logical processor, 100.0% of the time being online.
- The standard logical CPU busy is 52.31%.
- The zAAP logical processor busy is 98.24%.

In the Partition Data section of the report, we see:

- Partition A13 has 2 standard CPUs, which have a logical utilization of 52.31% and physical of 5.81%, that is 5.81% of 24 CPUs from the CP pool.
- The same partition A13 has one zAAP processor (ICF in Type means belonging to the ICF pool), which has a logical utilization of 98.24% and physical utilization of 16.37%, that is 16.37% of 6 PUs from the ICF pool.

zAAP on the Postprocessor Workload Activity report

We now look at what the Postprocessor Workload Activity report shows to us with global Sysplex report data.

Example 9-17 Postprocessor Workload Activity

```
                                            W O R K L O A D   A C T I V I T Y
                                                                                                                      PAGE   1
     z/OS V1R5             SYSPLEX SANDBOX            DATE 11/08/2004          INTERVAL 10.00.000   MODE = GOAL
                       CONVERTED TO z/OS V1R5 RMF     TIME 20.30.00

                                              POLICY ACTIVATION DATE/TIME 11/08/2004 18.37.41
------------------------------------------------------------------------------------------------------------ service class PERIODS
REPORT BY: POLICY=WLMPOL     WORKLOAD=BATCH      SERVICE CLASS=OMVSLOW    RESOURCE GROUP=*NONE      PERIOD=1 IMPORTANCE=5
                                                 CRITICAL     =NONE

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--   ---SERVICE----   --SERVICE TIMES--   PAGE-IN RATES    ----STORAGE----
AVG      3.00   ACTUAL            7.20.315   SSCHRT   0.0   IOC         0    TCB        1168.8   SINGLE    0.0    AVG     3230.98
MPL      3.00   EXECUTION         7.20.314   RESP     0.0   CPU     24256K   SRB           0.3   BLOCK     0.0    TOTAL   9692.42
ENDED       1   QUEUED                   1   CONN     0.0   MSO     19472K   RCT           0.0   SHARED    0.0    CENTRAL 9692.42
END/S    0.00   R/S AFFINITY             0   DISC     0.0   SRB       5490   IIT           0.0   HSP       0.0    EXPAND     0.00
#SWAPS      0   INELIGIBLE               0   Q+PEND   0.0   TOT     43734K   HST           0.0   HSP MISS  0.0
EXCTD       0   CONVERSION               0   IOSQ     0.0   /SEC     72897   IFA         587.5   EXP SNGL  0.0    SHARED   182.99
AVG ENC  0.00   STD DEV                  0                                   APPL% CP     96.9   EXP BLK   0.0
REM ENC  0.00                                               ABSRPTN    24K   APPL% IFACP  96.9   EXP SHR   0.0
MS ENC   0.00                                               TRX SERV   24K   APPL%IFA     97.9

GOAL: EXECUTION VELOCITY 20.0%     VELOCITY MIGRATION:   I/O MGMT  93.4%     INIT MGMT 93.4%

         RESPONSE TIME EX   PERF  AVG    --- USING% --- ---------- EXECUTION DELAYS % --------- ---DLY%-- -CRYPTO%- ---CNT%--    %
SYSTEM                 VEL% INDX ADRSP   CPU  IFA  I/O  TOT  IFA  CPU                           UNKN IDLE  USG  DLY  USG  DLY QUIE
SC70        --N/A--    93.4  0.2   1.0  23.9 24.1  0.0  3.4  2.3  1.0                           21.0 27.6  0.0  0.0  0.0  0.0  0.0
```

There are three new fields related to zAAP processing (APPL% IFACP, APPL% IFA, and IFA) and an old one with a changed name (from APPL% to APPL% CP).

- APPL% CP meaning does not change; it is the percentage of standard CPU time used by transactions in the service report class period (excluding zAAP). It includes the JVM tasks running in standard CPUs. The calculation is:

  APPL% CP = (TCB + SRB + RCT + IIT + HST - IFA) * 100 / Interval length

- IFA is the zAAP service time in seconds used by transactions in the service report class period. It is included in TCB or SRB.
- APPL% IFA is the percentage of processing time used by transactions executed on zAAP processors in the service report class period. The calculation is:

  APPL% IFA = (IFA * 100) / Interval length

- APPL% IFACP is the percentage of CPU time used by zAAP-eligible transactions running on standard CPs. This is a subset of APPL% CP.

In Example 9-17, we can see the following:

- There are three address spaces (batch Jobs) in this service class. MPL is 3.00, with two heavy processor users and one mostly idle.
- The zAAP processor service time is 587.5 seconds of the time, of the period of 600 seconds. That gives 97.9%; then as expected APPL% IFA is 97.9%.
- The zAAP-eligible application processing on the standard CPUs is 96.9% (APPL% IFACP). This is the Cross Over amount from the zAAP processor to the standard CP. Then in seconds we have:

  0.969 * 600 = 581 seconds

  These 581 seconds are included in the TCB time, consequently 1168.8 - 581.0 = 587.8 refers to JVM code running in a zAAP.

In Example 9-17, we also see values from the WLM Using and Delay sampling:

- The average number of address spaces is 1.0.
- CPU using samples 23.9%. Here it is not included the zAAP Using samples.
- IFA using samples 24.1%.
- IFA delay samples 2.3%.
- CPU delay samples 1.0%. Here it does not include the zAAP Delay samples.
- Unknown delay samples 21.0%.
- Idle delay samples 27.5%

9.4.4 zAAP Projection Tool

The zAAP Projection Tool for Java 2 Technology Edition allows customers who are considering using zAAPs to learn the potential for Java execution on zAAPs for their existing applications. This tool gathers usage information about how much CPU time is spent executing Java code that could potentially execute on zAAPs.

For further details on the Projection Tool, you can visit the home page at:

http://www.ibm.com/servers/eserver/zseries/zaap/gettingstarted/

Usage of APPL%IFACP as a projection tool

Without zAAPs configured, all IFA-related values show with N/A. This also applies to APPL% IFACP which is the CPU time on standard CPs spent by zAAP eligible work. However, RMF APAR OA07950 introduces the following exception: If no zAAPs are configured but the -Xifa:force JVM runtime option is set, APPL% IFACP is displayed. This will help your installation to assess the amount of IFA-eligible work and the number of zAAP processors needed.

For example, APPL% IFACP = 195 would indicate that almost two (1.95) standard CPs are busy with zAAP work. In addition, on certain z890 processors models, zAAPs (also known as IFA) might run at a faster speed than the standard CPs. This is where NORMALIZATION of IFA times comes into play. Certain fields in the WLMGL report are affected:

- TCB TIME reflects time spent on standard CPs as well as zAAPs. The zAAP portion of time is a normalized time, that is, the equivalent time as if the zAAP work was running on a standard CP. IFA TIME reflects time on zAAP processors. The value shown is not normalized even if zAAPs run at a different speed. Thus, if your zAAPs run twice as fast as your standard CPs and the WLMGL report displays TCB = 20.0 and IFA = 5.0, the TCB time includes 10.0 seconds of normalized zAAP time.
- APPL% CP is the percentage of CPU time spent on standard CPs. Without zAAPs the calculation is: APPL% CP = (TCB+SRB+RCT+IIT+HST) x 100 / interval. With zAAPs configured, the calculation is changed because TCB time includes time on zAAP processors: APPL% CP = (TCB+SRB+RCT+IIT+HST-IFA) x 100 / interval. The zAAP time subtracted is the normalized zAAP time.

9.5 Monitoring UNIX System Services (USS) applications

This section covers some performance aspects of USS applications running on a z/OS system. USS allows UNIX transactions to run under z/OS with total functionality rather than emulation (data, Syscall API, security, shell operator commands and so forth) of a Posix/XPG 4.2 certified UNIX operating system.

Before having a look at the RMF reports, here are some considerations regarding UNIX workload on z/OS:

- Each z/OS UNIX user consumes up to twice the system resource of a TSO/E user. Depending on the USS configuration, a user might require concurrent active address spaces. These address spaces are needed because the logic of the Fork (UNIX Attach) processing. When a user sends a transaction in OS/390 UNIX, this new process could be executed in a new address space. To minimize such overhead, WLM pre-creates a set of address spaces to be used at Fork moment. Also, the creation of an address space consumes central storage, to save this central storage, the z/Architecture concept of shared pages was introduced.
- Every time a user tries to invoke the z/OS UNIX shell, RACF has to deliver the security information out of the RACF database. To improve it, you should add the Virtual Lookaside Facility (VLF) classes that control caching of the UID and GID information.
- A UNIX system has additional definitions to define limits for workloads and resources in the parmlib member BPXPRMxx. The initial recommendation is to adhere to the default values and later, after inspecting the RMF reports, to change them to some more appropriate values for your installation.
- Hierarchical File Systems is the UNIX organization of data. IBM created the z File System (zFS), another file organization totally compatible at application level with HFS, but with better performance. A strong USS performance recommendation is to convert your USS HFS files to zFS.
- The installation should follow these recommendations to define the USS service class goals in the WLM policy:
  - The address spaces OMVS and BPXOINIT are automatically placed in the System service class. Do not modify them.

- Classify SYSBMAS (DFSMS data space buffer manager owner) in SYSSTC service class. SYSBMAS does not get classified automatically.
- The Daemon processes (as STC z/OS address spaces), which are long-lived processes running to perform continuous or periodic system-wide functions, should be placed in a service class with importance one and execution velocity close to 80%.
- The USS forked processes:
  - USS processes forked by BPXOINIT. These processes run under the user ID of OMVSKERN and they should be given an execution velocity goal and be more important than the other forked children. In the WLM OMVS subsystem rules classify UI of OMVSKERN to a service class with an importance of 1 and velocity goal of 50 or 60.
  - Place other forked children in service classes with several periods and response time goal but with a less importance.
- The USS services: setpriority(), chpriority(),and nice() should not be defined in the application.
- If possible, the applications should use Spawn (Fork plus Exec without address space creation) instead of Fork, or the _BPX_SHAREAS environment variable should be set to YES or REUSE to avoid the address space creation when the fork API is invoked. In this case only a new task is created. To use Spawn for batch, set the environment variable _BPX_BATCH_SPAWN to YES.

The Postprocessor offers the OMVS Kernel Activity report with the following sections:

- OMVS System Call Activity
- OMVS Process Activity
- OMVS Inter-Process Communication

Note: The EXCP count (or I/O block count) is used for a number of things under z/OS. This value is used to understand I/O activity from a performance perspective. It's also used to compute service units by WLM and many customers use the value for accounting, to charge back users on their systems. You can see the EXCP count in Monitor III Data Set Delay by Job report (DSNJ) and in the Group Response Time report. It is also reported in the Monitor II ARD report .

With the introduction of HFS, some difficulties arose because now there is a file system which can be shared by many users and which caches file activity and often avoids I/O. You can see that, for example, even when you are running the same application reading/writing the same exact files, depending on the activity of other users as well as your application, the final amount of I/O activity can be completely different. In fact, one run might show virtually no I/O at all, while another might show large amounts of I/O.

From a performance monitoring perspective, this could be a good thing. However, from the perspective of accounting and WLM, this might be a problem. A user who runs the same job two days in a row should be charged the same both days. Also, the administrator who sets a reasonable duration for period 1 of a service class does not want applications to fall into period 2 prematurely because there is more HFS contention.

The solution can be to use a model more or less like a cached control unit. You can charge for all the I/O your application would do if the physical file system did not cache the data for you. For each block (4K) you read or write, you are charged an EXCP count. You are also charged for directory lookups in the HFS.

While this approach produces repeatable results, the HFS does an excellent job of avoiding I/O, so your EXCP counts are much higher than the number of actual I/Os required.

9.5.1 OMVS System Call activity

Example 9-18 shows a System Call Activity report.

Example 9-18 OMVS System Call Activity report

```
                                     O M V S   K E R N E L   A C T I V I T Y
                                                                                                                              PAGE 1

            z/OS V1R6           SYSTEM ID SYS1                 DATE 10/15/2004            INTERVAL 30.00.000
                                 RPT VERSION V1R5 RMF           TIME 13:00:00              CYCLE 1.000 SECONDS

TOTAL SAMPLES = 1,800

                                         OMVS SYSTEM CALL ACTIVITY
 --------------------------------------------------------------------------------------------------------------------------------
                    MINIMUM AVERAGE MAXIMUM
 --------------------------------------------------------------------------------------------------------------------------------
 SYSCALLS (N/S)       102.0 1261.9*   7129
 CPU TIME (H/S)          16     47*     88
```

This section of the report is about syscalls and the CPU time they consume. Syscall is an invocation of USS (OMVS kernel) done by an application.

- SYSCALLS (N/S): The number of system calls per second processed by the OMVS kernel address space in this interval. There is no breakout of the system calls by each type.
- CPU TIME (H/S): Time spent to process system calls in hundredths of seconds per second. There is no breakout of the CPU by each type. In z/OS point of view, this time is classified as captured and is added to the address space ASCB.

9.5.2 OMVS Process activity

This section of the report is about USS processes and users. A USS process has the following characteristics:

- It is created by other parent processes, such as: Kernel (OMVS and BPXONIT), Daemons and Users. The parents are identified in the child by PPID. The child has its own PID. The syscall is a FORK() request.
- Depending on the system configuration, it requires a new address space.
- It can be a user process or a system process.

Do not confuse a USS *process* with a USS *thread*. Threads are separate dispatchable units of work within a process, and are a way for an application to execute different independent units of work concurrently. The benefits of this are that all the threads in a process can share all the resources (as virtual addresses) owned by the process. Making an analogy with z/OS, a USS process looks like a mix of a step task plus the address space concept. The USS threads are similar to subtasks. Example 9-19 shows a sample of the OMVS Process Activity section.

Example 9-19 OMVS Process Activity report

```
                             O M V S   K E R N E L   A C T I V I T Y
                                                                                                                           PAGE    1
            z/OS V1R5                SYSTEM ID SYSF             DATE 02/03/2005            INTERVAL 15.00.000
                                     RPT VERSION V1R5 RMF       TIME 10.45.00              CYCLE 1.000 SECONDS
TOTAL SAMPLES =   898
                                        OMVS PROCESS ACTIVITY
------------------------------------------------------------------------------------------------------------------------------------
                          PROCESSES                        USERS                    PROCESSES PER USERS
 MAXIMUM  (TOT)               4096                              200                        1000
------------------------------------------------------------------------------------------------------------------------------------
                  MINIMUM  AVERAGE  MAXIMUM      MINIMUM  AVERAGE  MAXIMUM      MINIMUM  AVERAGE  MAXIMUM
------------------------------------------------------------------------------------------------------------------------------------
 CURRENT  (TOT)       33     33.00        33            1     1.000        1
 OVERRUNS (N/S)   0.000    0.000    0.000        0.000    0.000    0.000        0.000    0.000    0.000
```

In the top section of the report, the BPXPRMxx parmlib definitions for the maximum number allowed of OMVS processes, Users, and OMVS processes per user are reported. Some comments about these values follow:

- About processes: The recommendation (maxprocsys) is to start with the default and raise the value as the number of processes approaches the limit, or initially set it to 4 times maxuids.
- The max number of concurrent UNIX users (maxuid): If the limit is reached, OMVS log ins are denied. As a general rule: 1 UNIX user = 3 TSO users (resources wise). Default = 200. The recommendation is to specify a realistic value. A user with UID=0 (superuser) is not limited by MAXUIDS.
- About process per user: The default is 25 and the recommendation is to start with the default and monitor through RMF reports.

Definitions of the fields are as follows:

- CURRENT PROCESSES: Number of OMVS processes controlled by OMVS during this interval.
- CURRENT USERS: Number of OMVS users controlled by OMVS during this interval.
- PROCESSES PER USER: Number of processes per each user during this interval.
- OVERRUNS PROCESSES: Rate of processes that could not be created by OMVS because the maximum number of processes would have been exceeded. Here you can see if the defined maximums are limiting the demand.
- OVERRUNS USERS: Rate of OMVS users that could not be created by OMVS because the maximum number of users would have been exceeded.
- OVERRUNS PROCESSES PER USER: Rate of processes per user that could not be created by OMVS because the maximum number of processes per user would have been exceeded.

9.5.3 OMVS Inter-process communication

This section of the report is about the different ways processes communicate. When processes need to communicate (passing control or exchanging data), they can do so in a number of different ways. From a networking client/server perspective this can be accomplished with TCP/IP sockets. However, processes on the same UNIX system can also communicate using:

- Shared memory for communicating structures in storage.
- Message queues for communicating lists and queues.
- Semaphores for locking and serialization.

Example 9-20 shows an extract of the Postprocessor OMVS Inter-Process Communication report.

Example 9-20 OMVS Inter-Process Communication

```
OMVS INTER-PROCESS COMMUNICATION
------------------------------------------------------------------------------------------------------------------------------------
                     MESSAGE QUEUE IDS              SEMAPHORE IDS                  SHARED MEMORY IDS            SHARED MEMORY PAGES
 MAXIMUM  (TOT)               500                         4096                         4096                         262K
------------------------------------------------------------------------------------------------------------------------------------
                  MINIMUM  AVERAGE  MAXIMUM    MINIMUM  AVERAGE  MAXIMUM    MINIMUM  AVERAGE  MAXIMUM    MINIMUM  AVERAGE  MAXIMUM
------------------------------------------------------------------------------------------------------------------------------------
 CURRENT  (TOT)       11    11.00       11          0    0.000        0          0    0.000        0          0    0.000        0
 OVERRUNS (N/S)    0.000    0.000    0.000      0.000    0.000    0.000      0.000    0.000    0.000      0.000    0.000    0.000
```

```
                                      OMVS MEMORY MAP                              SHARED LIB REGION             QUEUED SIGNALS
---------------   --------------------------------------------------------     --------------------------   --------------------------
                  MEMORY MAP STORAGE PAGES         SHARED STORAGE PAGES        MAX SHARED LIBRARY REGION     MAXIMUM QUEUED SIGNALS
 MAXIMUM  (TOT)             65536                          131K                             64                          1000
---------------   --------------------------------------------------------     --------------------------   --------------------------
                  MINIMUM  AVERAGE  MAXIMUM    MINIMUM  AVERAGE  MAXIMUM       MINIMUM  AVERAGE  MAXIMUM    MINIMUM  AVERAGE  MAXIMUM
---------------   --------------------------------------------------------     --------------------------   --------------------------
 CURRENT  (TOT)        0    0.000         0     532         532.0       532            0     0.000        0
 OVERRUNS (N/S)    0.000    0.000    0.000      0.000    0.000    0.000         0.000    0.000    0.000      0.000    0.000    0.000

 Units:   (TOT) = Total Value, (N/S) = Number per Second, (H/S) = Hundredth of seconds per Second
```

At the top of the report, the BPXPRMxx parmlib definitions for the maximum message queue IDs (IPCMSGNIDS), semaphore IDs (IPCSEMNIDS), unique shared memory IDs (IPCSHMNIDS) and shared memory pages (IPCSHMSPAGES) related to OMVS inter-process communication, are reported. (For further details, refer to *z/OS MVS Initialization and Tuning Reference,* SA22-7592.)

The BPXPRM parmlib definitions are:

- IPCSMSGNIDS: The maximum number of unique system-wide message queues. The default is 500. The current and the overrun figures are also presented.
- IPCSEMNIDS: The maximum number of unique system-wide semaphore sets. The default is 500. The current and the overrun figures are also presented.
- IPCSHMNIDS: The maximum number of unique shared memory segments (for one request). The default is 500. The current and the overrun figures are also presented.
- IPCSHMSPAGES: The maximum number of shared memory segments, created by calls to the fork and shmat functions. The default is 262144. The current and the overrun figures are also presented.
- MAXMMAPAREA: The maximum amount of data space storage (in pages) that can be allocated for memory mappings of HFS files. Storage is not allocated until the memory mapping is active. Using memory map services causes a large amount of system memory to be consumed. For each page (4KB) that is memory-mapped, 96 bytes of ESQA are consumed when a file is not shared with any other users. When a file is shared by multiple users, each user after the first causes 32 bytes of ESQA to be consumed for each shared page. Assuming that the default of 4096 pages is taken, and assuming that no sharing is done by mmap() users, a maximum of 384KB (4096 * 96) of ESQA could be consumed. The ESQA storage is consumed when the mmap() function is invoked rather than when the page is accessed by the memory mapping application program. Recommendation is to start with the default and monitor through RMF reports.
- MAXSHAREPAGES: The maximum number of pages for shared storage (allocated in ESQA). These pages are used, for example, by fork() (copy-on-write forks), mmap() (memory map files), and shmat() (shared memory). The default is 131072.

  Estimate of the ESQA storage used is equal to 32 * maxsharepages.

  Recommendation is to start with the default and monitor set high enough to prevent OVERRUNS, since fixed storage is used to make sure other work is not impacted.

If all these values are set too low, you may limit the number of shared storage pages used by UNIX processes, consequently affecting performance. Setting these values too high could cause shortages of ESQA. Remember that all ESQA pages are fixed pages, which implies that real storage as well as virtual storage is allocated. Then, together with these data, you should keep an eye in the Postprocessor Virtual Storage Activity report to track the Larger Free Block field in ESQA to see how much room you still have to exploit USS shared memory communication features.

In the second part of the report, memory usage is reported:

- CURRENT MEMORY MAP STORAGE PAGES: Number of memory map storage pages during this interval.
- CURRENT SHARED STORAGE PAGES: Number of shared storage pages during this interval.
- OVERRUNS MEMORY MAP STORAGE PAGES: Rate of memory map storage pages that could not be created by OMVS because the maximum number of memory map storage pages would have been exceeded.
- OVERRUNS SHARED STORAGE PAGES: Rate of shared storage pages that could not be created by OMVS because the maximum number of shared storage pages would have been exceeded.
- SHARED LIBRARY REGION: The following fields cover shared library region. The values are provided in units of MB.
  - MAX SHARED LIBRARY REGION: Maximum amount of storage available for shared library region as specified by parmlib statement SHRLIBRGNSIZE.
  - CURRENT SHARED LIBRARY REGION: The current amount of storage in MB available for shared library region.
  - OVERRUNS SHARED LIBRARY REGION: Rate of attempts to exceed the maximum storage amount for shared library region.
- MAX QUEUED SIGNALS: The following fields cover queued signals.
  - MAXQUEUEDSIGS: Maximum amount of queued signals allowed per process as specified by parmlib statement.
  - OVERRUNS QUEUED SIGNALS: Rate of attempts to exceed the maximum number of queued signals.

For further details, refer to *UNIX System Services z/OS Version 1 Release 4 Implementation*, SG24-7035.

9.5.4 HFS Global Statistics report

This Postprocessor report is one of the two reports belonging to the HFS Statistics report and provides overall data about I/O activities of the HFS file and gives statistics about the various buffer pools which have been defined.

Example 9-21 HFS Global Statistics report

```
                                       H F S   G L O B A L   S T A T I S T I C S
                                                                                                                           PAGE    1
            z/OS V1R5                SYSTEM ID TTR1             DATE 10/10/2004            INTERVAL 15.00.000
                                     RPT VERSION V1R5 RMF       TIME 10.45.00              CYCLE 1.000 SECONDS

        --- STORAGE LIMITS (MB) --                      ----- FILE I/O ----  --- METADATA I/O --
                                                             COUNT     RATE       COUNT     RATE
        VIRTUAL        MAX       499
                      USE      5.382       CACHE                0    0.000         245    0.272
        FIXED        MIN     0.000      DASD                 0    0.000           0    0.000
                     USE     0.000      HIT RATIO         0.00              100.00
------------------------------------------ BUFFER POOL STATISTICS -------------------------------------------
  POOL       NUMBER    BUFFER   -------- POOL SIZE --------      DATA      ---------- I/O ACTIVITY ---------
NUMBER      BUFFERS      SIZE     PAGES     BYTES    %FIXED    SPACES        TOTAL        FIXED       %FIXED

     1          366         1       366     1464K         0         1          220            0            0
     2            5         4        20       80K         0         1            0            0            0
     3           10        16       160      640K         0         1            0            0            0
     4          13       64       832    3328K        0         1           0           0           0
```

Storage limits

All fields are given in megabytes and show the values at interval end:

- VIRTUAL MAX: Maximum amount of virtual storage that HFS data and metadata buffers should use. If you do not set a value for max, the system assigns a default value that is equal to half the amount of real storage available to the system at HFS initialization. The HFS buffer pools are usually defined in data spaces. Defined by VIRTUAL(MAX) in BPXPRMxx.
- VIRTUAL USE: Current total amount of virtual storage assigned to I/O buffers.
- FIXED MIN: Amount of virtual storage that is fixed at HFS initialization time and remains fixed even if HFS activity drops to zero. Min must be less than or equal to VIRTUAL(max). Min cannot exceed 50% of real storage available to the system. If the allowed amount of storage is exceeded, an informational message is issued and min is set to 50% of real storage. The minimum limit can be changed dynamically by invoking the **confighfs** shell command. Defined by FIXED(MIN) in BPXPRMxx.
- FIXED USE: Current total amount of permanently fixed storage assigned to I/O buffers. Remember that during an I/O operation, the I/O buffer must be fixed in central storage: if the buffer is already fixed you may save some CPU cycles. This number is included in the VIRTUAL USE field.

File I/O

The fields are given as COUNT and RATE (count per second):

- CACHE: First page of a data file is requested and found in virtual storage buffer pool (cache). Remember that in UNIX the buffer pool is called cache.
- DASD: First page of a data file is requested and not found in virtual storage buffer pool and a DASD I/O is necessary.
- HIT RATIO: Percentage of cache-found requests based on total number of requests.

  The generic rule-of-thumb for this value is at least 80%. To improve this value, you can raise the buffer pool size. However, you should consider also the possibility that the workload running is cache unfriendly. We also recommend that you take a look at the “HFS File System Statistics report” on page 298, to get information about each file hit ratio.

Metadata I/O

The fields are given as COUNT and RATE (count per second). The attribute, name, and name directory information that is part of the HFS structure is also referred to as metadata.

- CACHE: Metadata for a file was found in virtual storage during file lookup.
- DASD: Metadata for a file was not found in virtual storage during file lookup, and an index call was necessary which may result in an I/O.
- HIT RATIO: Percentage of cache-found requests based on total number of requests.

Performance wise it is very important to get close to 100% value in the hit ratio of metadata.

Buffer pool statistics

HFS defines up to 4 buffer pools, each one with a different buffer size.

- POOL NUMBER: This number is used to refer to one of these buffer pools.
- NUMBER BUFFERS: Number of buffers in this buffer pool.
- BUFFER SIZE: Size of each buffer in this pool (in pages).

- POOL SIZE: Gives more details about each buffer pool.
  - PAGES: Size of this buffer pool in pages, it is the product of multiplying of Buffer_size by Number_buffers.
  - BYTES: Size of this buffer pool in bytes.
  - %FIXED: Percentage of the size of the buffers which are permanently fixed.
- DATA
  - SPACES: Number of data spaces comprising this buffer pool.
- I/O ACTIVITY
  - TOTAL: Number of buffers in this buffer pool for which I/Os was issued. Remember that multiple buffers can be written in a single I/O request.

    If the number for TOTAL is consistently smaller than NUMBER BUFFERS, your buffered size might be overestimated.
  - FIXED: Number of times a buffer was already fixed prior to an I/O request in this buffer pool. In this case, there is not the overhead of fixing.
  - %FIXED: Percentage view of the FIXED number.

9.5.5 HFS File System Statistics report

The second part of the HFS Global Statistics report is based on data gathering for specific file systems. You get data about I/O activities and about the internal structure (index) of the HFS files. You need to specify the HFS you want to monitor to the Monitor I, for example in order to monitor the HFS file *ZOSR04.OMVS.ROOT,* you need to specify the following parameter:

```
HFSNAME(ADD(ZOSR06.OMVS.ROOT))
```

The fields are almost the same portrayed globally with a few exceptions, such as:

- Mount date and time.
- Type of I/O request: sequential or random.
- Index events, such as: new level in the index, number of joins, number of splits. These numbers give you an idea about the level of insertions and deletions. Also an indication about when there is a need to reorganize the file system using the **copytree** utility.

Example 9-22 HFS file system statistics

```
                                        H F S   F I L E   S Y S T E M   S T A T I S T I C S                           PAGE    2
            z/OS V1R5                SYSTEM ID SYSE             DATE 10/10/2004            INTERVAL 15.00.000
                                     RPT VERSION V1R5 RMF       TIME 10.45.00              CYCLE 1.000 SECONDS

--- ALLOCATION (MB) --                   ----- FILE I/O ----  --- METADATA I/O --  ---- INDEX I/O ----     ---- INDEX EVENTS ---
                  SIZE                        COUNT     RATE       COUNT     RATE       COUNT     RATE                     COUNT

FILE SYSTEM NAME: OMVS.SYSE.LOCAL.HFS
MOUNT DATE: 01/26/2005   TIME: 21:41:52

SYSTEM              48     CACHE                  0    0.000          25    0.028          25    0.028     NEW LEVEL           0
DATA                45     DASD                   0    0.000           0    0.000           0    0.000     SPLITS              0
ATTR. DIR        0.199     HIT RATIO           0.00                 100.00                100.00                JOINS          0
                           SEQUENTIAL              0    0.000
CACHED           0.000     RANDOM                  0    0.000

FILE SYSTEM NAME: OMVS.SYSE.LOCAL.JAVA130
MOUNT DATE: 01/26/2005   TIME: 21:41:53

SYSTEM             259     CACHE                  0    0.000           0    0.000           0    0.000     NEW LEVEL           0
DATA               249     DASD                   0    0.000           0    0.000           0    0.000     SPLITS              0
ATTR. DIR        0.464    HIT RATIO           0.00                0.00                 0.00              JOINS              0
                            SEQUENTIAL             0    0.000
CACHED           0.000    RANDOM                 0    0.000
```

Following is a description of the fields:

- FILE SYSTEM NAME: Name of the HFS file system which has been selected for reporting.
- MOUNT DATE and TIME: Time when the selected file system was mounted.

Allocation

All fields are provided in megabytes:

- SYSTEM: Amount of storage allocated to this file system
- DATA: Amount of storage internally used within HFS for data files, directories, and HFS internal structures like the attribute directory (AD).
- ATTR. DIR: Amount of storage used for the attribute directory (AD). This number is included in the DATA field. The attribute directory is the internal HFS structure (index) that contains attribute information about individual file system objects as well as attributes of the file system itself.

File I/O

The fields are given as COUNT and RATE (count per second):

- CACHE: First page of a data file was requested and found in virtual storage (cache). This counter is repeated for metadata and index I/O.
- DASD: First page of a data file was requested but was not found in virtual storage (cache) and an I/O was necessary. This counter is repeated for metadata and index I/O.
- HIT RATIO: Percentage of cache-found requests based on total number of requests. This counter is repeated for metadata and index I/O.
- SEQUENTIAL: Number of sequential file data I/O requests. A sequential I/O is an I/O where the sequencing of reads or writes is pre-determined by a logical key or the physical location of the data. Usually the file is read or written from the beginning to the end. The system knows the next data to be read and may do an anticipatory buffering to improve the sequential performance.
- RANDOM: Number of random file data I/O requests. In a random I/O there is not a previous indication of the next data to be read or written.

Index Events

- NEW LEVEL: Number of how often HFS added a new level to its index structure. The index statistics are relative to all of the indices in the HFS data set. The attribute directory (AD) is one index (the largest), but each directory (including the root) is also an index.
- SPLITS: Number of times an index page was split into two pages because new records were inserted. This gives an idea of how much insertion activity there has been for the index structure.
- JOINS: Number of times HFS was able to combine two index pages into one, because enough index records had been deleted in the two pages.

You can find further information in *Hierarchical File System Usage Guide*, SG24-5482.

9.5.6 OMVS process data report

This Monitor III report provides information about UNIX System Services address spaces and server processes and it can be used to address the following questions:

- What are the delayed processes?

- What command is associated with the process?
- What is the status of each of the processes?
- Which processes are high CPU consumers?

Example 9-23 OMVS Process Data

```
                     RMF V1R5 OMVS Process Data                  Line 1 of 24
Command ===>                                                   Scroll ===> HALF

Samples: 18    System: SYS4    Date: 07/28/04    Time: 15.50.41   Range: 19 Sec

Kernel Procedure: OMVS        Kernel ASID: 0014      Option: PID       ALL
BPXPRM: OMVS=(71,04)

-------------------------------------------------------------------------------
Jobname   User      ASID       PID       PPID  LW  State  Appl%   Total Server

BPXOINIT  OMVSKERN  0030         1          0        MF    0.0  0.234   FILE
INETD8    OMVSKERN  0047         5          1        1FI   0.0  0.052   N/A
MVSNFSC   MVSNFS    5001         7          1        1A    0.0  0.229   N/A
MVSNFSC   MVSNFS    5001         8          1        1A    0.0  0.229   N/A
MVSNFSC   MVSNFS    5001         9          1        1A    0.0  0.229   N/A
MVSNFSC   MVSNFS    5001        10          1        1A    0.0  0.229   N/A
MVSNFSC   MVSNFS    5001        11          1        1A    0.0  0.229   N/A
MVSNFSC   MVSNFS    5001        12          1        1A    0.0  0.229   N/A
MVSNFSC   MVSNFS    5001        13          1        1A    0.0  0.229   N/A
MVSNFSC   MVSNFS    5001        14          1        1A    0.0  0.229   N/A
TCPIP     TCPIP     0044        15          1        MR    0.0  43.59   N/A
TCPIP     TCPIP     0044        16          1        1R    0.0  43.59   N/A
TCPIP     TCPIP     0044        17          1        1R    0.0  43.59   N/A
```

The fields have the following meanings:

- Jobname: The Jobname in the case of a batch address space; otherwise, the name of the JCL procedure that started the process.
- User: User name (also called user ID) associated with the process.
- ASID: Decimal ID of the address space associated with the process.
- PID: Process ID.
- PPID: Parent process ID.
- LW: If the reported process is waiting for the process latch (a sort of lock) of another process, 'Y' is shown; otherwise, blank.
- State: Cumulative state information of the address space and process. Some of these states are the following:

M:	Multiple threads, no pthread_create used
F:	File system kernel wait
I:	Swapped out
1:	Single thread
X:	Creating new process
R:	Running
F:	File system kernel wait
H:	Multiple threads, pthread_create used
K:	Other kernel wait

- Appl%: Percentage of TCB and local/global SRB time that the process consumed during the report range.

- Total: Total computing time since the process was started.
- Server: If the process represents a server, the server type is shown; otherwise, N/A.

This report has cursor sensitivity capability. When the cursor is placed on certain fields, and Enter is pressed, an additional report is invoked, as follows:

- On the Jobname field, the Job report is invoked for the jobname the cursor was on.
- On the Parent Process ID (PPID) field, the OMVS Process Data report is invoked again with internal search for the parent process.
- On any other column, the OMVS Process Data Details pop-up panel is invoked, showing details of the process in the row the cursor is on.

Example 9-24 OMVS Process Data - Details

```
               RMF OMVS Process Data - Details

Press Enter to return to the Report panel.

Start Time/Date : 12.08.57 03/09/2005
Command         : GFSAMAIN
Process-ID :       25    Parent Process-ID : 1
Jobname    :  MVSNFSS    User Name         : MVSNFS
ASID       :     0049    Hexadecimal ASID  : 0031

Appl% : 0.0     Total CT     : 0.485   LW-PID     :     0

Server Information:
  Name   : MVSNFS
  Type   : FILE  Active Files      0   Max. Files :  200K

Process State : MF
M: Multiple threads, no pthread_create used
F: File system kernel wait
```

9.6 Monitoring WebSphere Application Server workload

This section covers some aspects of WebSphere applications running on a z/OS system. WebSphere Application Server is a transactional workload, and all performance information given in this book applies to it in the same way as for other applications. Therefore, you can use the methodologies for monitoring and analyzing the performance as described in this and other chapters.

Nevertheless, some explanation of the unique terms used in context with WebSphere is in order.

9.6.1 WebSphere Application Server configuration

A Server Instance, as shown in Figure 9-3, is a set of WebSphere Application Server address spaces formed by:

- Control Region (an AS like TOR in CICS) with the functions:
  - Trusted/Authorized/Integrity
  - Communications Endpoint
  - Recoverable Resources
  - WLM Classification and Routing

- Scalable Transaction Recovery/Restart
- Choice of Scheduling Policies
- No Application Code, only IBM code

- Several Server Regions—optional—(an AS like AOR in CICS) with the functions:
  - Application code
  - Transaction/User Isolation
  - Back-end Data Attachments
  - Started and Managed by WLM
  - No Recoverable Resources
  - Created and deleted dynamically by WLM through the implementation of dynamic application environment.

Control Region and Server Regions code is written in Java, thus it includes the JVM component.

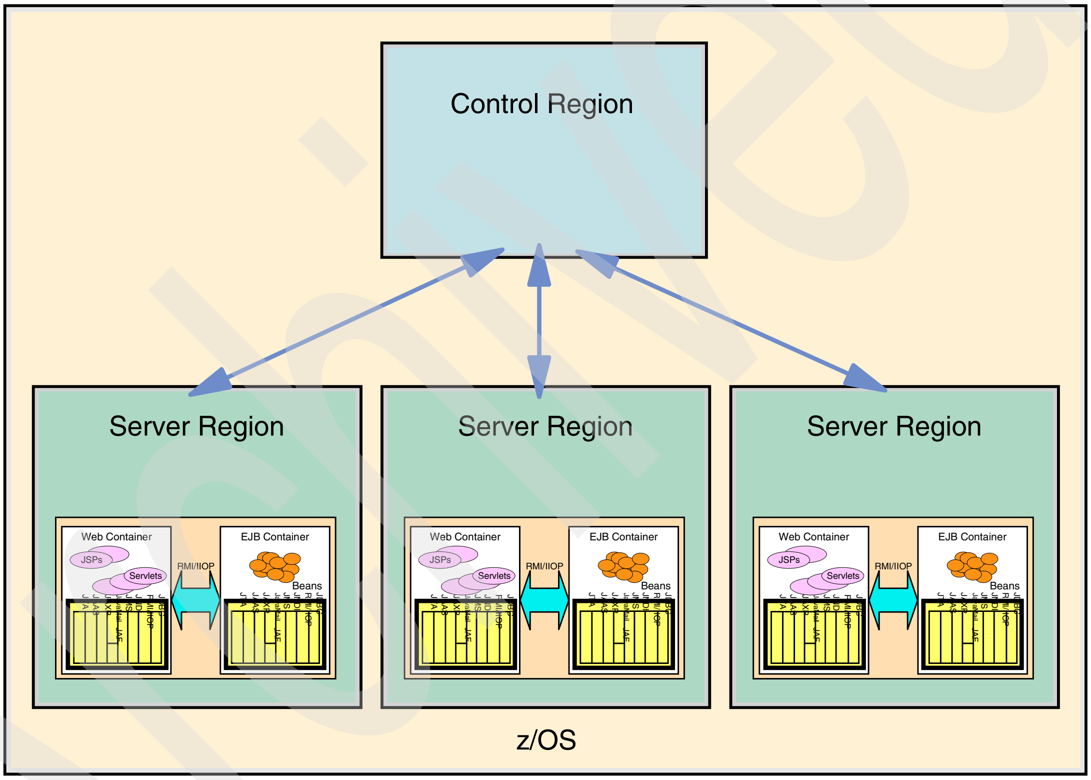


Figure 9-3 WebSphere Application Server instance

9.6.2 Some specific definitions for WebSphere Application Server

This section presents some WebSphere Application Server specific performance definitions.

Response time

Response time definition, including processing in the PC for a modern transaction such as WebSphere Application Server, is discussed in “Average transaction response time” on page 155.

RMF reports for WebSphere on z/OS measure the time between work being queued by a Control Region and that piece of work being completed in the Server Region. We use this definition of response time in this section.

External throughput rate (ETR)

This is a measure of the amount of work going through a system in a given time. Typically this may be measured as the number of transactions per second. As with response time, our measurement of ETR with WebSphere on z/OS is based on the work completed by the Server Region. Refer to "External throughput rate (ETR)" on page 157.

Transaction

A strict definition of transaction is "logical unit of work." When one transfers money from one account to another, it is to be removed from the first account and then added to the second. The transaction includes both these processes, and they must both complete successfully for the transaction to be considered complete. A mechanism is also required to ensure that if one of the processes fails, the other is either not attempted or is also undone.

A customer at a browser may consider the complete process of selecting a book, entering payment and delivery details, and then finalizing the purchase as a single transaction.

WebSphere considers each incoming request as a transaction. Each of the requests to WebSphere generated by the customer as they go through the process of buying a book will be treated as a separate transaction. Our discussion in this chapter uses the term *transaction* as viewed by z/OS Workload Manager and reported by RMF. All the WebSphere transactions run under an enclave.

Hit rate

Often used as a measure of activity on a Web site. A *hit* is the retrieval of any single item from a Web server. Hence, a Web page with four graphical items actually count as five hits: one for the html page and one for each of the graphical items. Hit rate is the number of hits in a given time. While this does measure all the interactions between user and browser, it tends to hide the more valuable measure of the number of pages being accessed.

Page view rate

Page view rate is a more valuable measure than hit rate. This counts complete pages retrieved in a given time rather than all the individual elements.

Important: These definitions should be understood when interpreting WebSphere transaction rate from a Workload Activity report. For WebSphere on z/OS, RMF views each request as a transaction, whether it is a call to a J2EE application or a request to a static page element.

If WebSphere on z/OS is serving static pages, the transaction rate reported by RMF will in fact be closer to a measure of the hit rate.

If static content is served from another source, for example a WebSphere Edge server front end, and requests issued to the back end application server are mostly for J2EE applications, then the value reported by RMF will be closer to the resulting Page View rate.

Number of clients and think time

The number of clients is the number of users connected to the Web site. However, as opposed to legacy applications, there is no direct relation between the number of clients connected and the load on the Web server. This is due to the heterogeneous nature of Web applications. In a traditional CICS or IMS application, users tend to be logged on working almost continuously. In a Web application, for example when buying a book, there tends to be more browsing while users evaluate the information that has been returned. This is *think time*. While the users are thinking, they are still effectively connected to the site but they are not driving work in WebSphere (although there may still be a session object from their previous interaction). Thus, in an application that tends towards long think time, there may be a large number of concurrent users, also called clients, but a low transaction rate in the WebSphere application server.

Resource

This is any item that can be used in the execution of a transaction. This can be a physical resource (for example, CPU or memory) or a logical resource (for example, JDBC connection, a queue in WLM, and so forth). When a WebSphere transaction accesses data in DB2 or CICS, it may also be convenient to refer to DB2 or CICS as a resource.

For a transaction to complete, it must be able to access all the resources it requires. For a transaction to perform well, there should be enough of these resources available and they need to be available quickly enough. How much is enough? How quick is quick? There is no straight answer.

9.6.3 WLM Delay Monitoring

WebSphere Application Server for z/OS Version 5 uses Workload Manager (WLM) services to report transaction begin-to-end response times and execution delay times. The WLM data collected by Resource Measurement Facility (RMF) is captured in two phases of the RMF report:

- BTE - the begin-to-end phase applies to requests handled by the controller
- EXE - the execution phase applies to requests handled by the servant

You can use this status information to determine where possible performance bottlenecks are occurring. This feature is available on z/OS V1R2 and above with WLM APAR OW51848 and RMF APAR OW52227.

When a new transaction enters the system, the WebSphere Application Server for z/OS application control region (ACR) starts the classify service. Delays associated with the WebSphere Application Server for z/OS ACR service class are counted separately for the BTE phase and the EXE phase. This support allows WLM to associate a performance block (PB) with an enclave to record delays that occur in the flow of a transaction. The state samples are collected on an ongoing basis and reported as a percentage of average transaction response time. Table 9-2 shows the states, their codes, the section of the RMF report where each is reported, the meaning, and suggested response. You can use this information in the RMF report to determine where some of your system's performance problems may be occurring.

Table 9-2 WLM delay monitoring states

State	Code	Report	Meaning	Response
ACTIVE	ACTIVE SUB	Both BTE and EXE	WebSphere is actively processing request	
ACTIVE_APPLIC	ACTIVE APPL	Both BTE and EXE	Application is running	Use application monitoring tool to determine the cause of the delay
WAITING TYPE1	TYP1	EXE	EJB collaborator delay	
WAITING TYPE2	TYP2	EXE	Resource manager delay	Called a J2C connector to perhaps DB2, CICS, IMS. Investigate other resource manager using their monitoring tools

State	Code	Report	Meaning	Response
WAITING TYPE3	TYP3	EXE	Servant called to a different distributed object server using RMI/IIOP	1. Investigate the delay on the other server. The delay may point to session caches 2. Look for any network problems 3. Avoid outbound calls
WAITING TYPE 4	TYP4	BTE	OTS call to RRS. Occurs only in controller when controller is trying to commit a distributed transaction	1. Investigate the delay on the other server 2. Look for any network problems 3. Consider combining application into one server to avoid delay
WAITING REGIST TO WORKTABLE	WORK.	BTE	An indication of contention within the controller while trying to process concurrent requests	If delay is excessive, consider adding another controller and splitting work off to it
WAITING OTHER_PRODUCT	OTHER	BTEI	ndicates a configuration problem in DNS or TCP/IP	Check to make sure all the DNS servers are running. You might want to look at OPING or ONSLOOKUP
WAITING DISTRIB	DIST	BTE	Controller as a client went outbound waiting for a response	1. Investigate the delay on the other server 2. Look for any network problems 3. Consider combining application into one server to avoid delay
WAITING SESS_NETWORK	REMT	BTE	Time spent waiting for a TCP/IP session to be established on the network	The two session delays should be observable in conjunction with TYP3 delays. Look at TCP/IP configuration
WAITING SESS_SYSPLEX	SYSP	BTE	Time spent waiting for a TCP/IP session to be established on the sysplex	The two session delays should be observable in conjunction with TYP3 delays. Look at TCP/IP configuration
WAITING REGULAR_THREAD	REGT	BTE	Waiting for a thread in the controller. Work is bottlenecked in the controller because it is receiving more requests than it can process	Split the controller
WAITING SSL_THREAD.	SSLT	BTE	Waiting for an SSL thread in the controller. Work is bottlenecked in the controller because it is receiving more requests for SSL handshakes than it can process	Split controller in increase SSL threads 1. Increase SSL threads 2. Look at SSL configuration 3. Split the controller to increase SSL threads

State	Code	Report	Meaning	Response
WAITING SESS_LOCALMVS.	LOCL	BTE	Time spent communicating with a different distributed object server using local optimized communication	1. Investigate the delay on the other server 2. Avoid outbound calls

9.6.4 Performance monitoring

When monitoring a WebSphere workload, there is no significant difference from other transactional applications with respect to RMF. In this section, we discuss parts of the Workload Activity report for a reporting class running in enclaves.

Workload Activity report (focusing on the transactions)

Example 9-25 is a Workload Activity report for a reporting class associated with a WebSphere workload, running in enclaves. It corresponds to the WLM definitions made to the CB subsystem.

Example 9-25 Workload Activity report

```
REPORT BY: POLICY=LSA510               REPORT CLASS=WASE
                             DESCRIPTION =LSA510 WAS EBUSINESS WORKLOAD

TRANSACTIONS    TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O-- --SERVICE RATES--  PAGE-IN RATES ---STORAGE----
AVG      2.28   ACTUAL                 147   SSCHRT   1.5  ABSRPTN    38580   SINGLE    0.0  AVG     0.00
MPL      2.28   EXECUTION              101   RESP     1.8  TRX SERV   38580   BLOCK     0.0  TOTAL   0.00
ENDED    6777   QUEUED                  45   CONN     1.2  TCB        229.7   SHARED    0.0  CENTRAL 0.00
END/S   22.59   R/S AFFINITY             0   DISC     0.3  SRB          0.0   HSP       0.0  EXPAND  0.00
#SWAPS      0   INELIGIBLE               0   Q+PEND   0.3  RCT          0.0   HSP MISS  0.0
EXCTD       0   CONVERSION               0   IOSQ     0.0  IIT          0.0   EXP SNGL  0.0  SHARED  0.00
AVG ENC  2.28   STD DEV                201                 HST          0.0   EXP BLK   0.0
REM ENC  0.00                                              APPL %      76.6   EXP SHR   0.0
MS ENC   0.00
```

- AVG is the average number of active transactions running during the RMF interval.
- MPL is the average number of transactions resident in storage during the measurement interval. This field is identical to AVG, but only includes the address spaces swapped in.
- ENDED is the number of transactions that ended during the interval, and END/S is the number of transactions that ended per second. If the reporting class is set up correctly, this is a direct measure of the application throughput as seen by WebSphere.
- AVG ENC is the average number of enclaves concurrently active at any point in time. This information may be useful to size storage requirements or system recovery aspects.
- TRANS.-TIME contains the transaction time, as seen by Workload Manager. This is from the time the transaction is placed on the server region WLM queue until the time the transaction is completed.
  - ACTUAL is the actual amount of time required to complete the work submitted under the service class. This is the total response time.
  - QUEUED is the average time the WebSphere transaction was delayed on the WLM queue waiting for server address space availability. The time can increase under full load conditions, if the number of servers in MAX_SRS is too low. The use of WLM Dynamic Application Environment may decrease such time, because based in goal and delay for server reasons, WLM controls dynamically the number of server address spaces.
- Note that the STORAGE field is always zero for an enclave type of report. Enclaves are associated with address spaces; however, by design no central storage values are reported.

- The APPL% field indicates the CPU captured time incurred on behalf of all activities which are part of the enclave. It is expressed as a percentage of one CP time used over the interval. Note that this represents *all* the CPU activity across all address spaces spanned by the transaction, including DB2 and CICS if the transaction contains JDBC or JCA connectors.

  No activity (or response time) information is reported by WLM within the CICS assigned service class or report class.
- From the previously defined fields, it is possible to calculate a characteristic of the workload, the average CP cost per transaction.

  Using the fields for the report class WASE in Figure 9-25, which identifies a WebSphere application workload, you can derive that over the measurement interval:

  – 2.28 transactions were concurrently active, all of them running under enclaves.

  – A total of 6777 transactions ended, which translates into an average throughput of 22.59 transactions per second.

  – The average response time was 147 milliseconds.

  – Over the measurement interval, APPL% indicates that one CP was busy 76.6% of the time to service WASE. Since the measurement interval is 5 minutes, this translates into:

  $$\text{CP time} = 300 \times 0.766 = 229.8 \text{ sec}$$

  – Over the same interval, 6777 transactions have been processed. We can derive the average CP cost in millisecond per transaction:

  $$\text{CP time / Trans} = \frac{229.8 \times 1000}{6777} = 33.9 \text{ ms}$$

  This calculation does not take into consideration the apportioning of the non-captured time.

Workload Activity report (focusing on the STC address spaces)

We recommend that server address space activity, which does not run under an enclave, should be assigned to a service class in the STC group. This processing time is associated with garbage collector, or memory leak.

If you also have defined a report class, you can obtain a workload report for the server region.

Example 9-26 Workload Activity report for WebSphere Application Server address space (extract)

```
REPORT BY: POLICY=LSA510                    REPORT CLASS=WASS
                                          DESCRIPTION =LSA510 WAS SERVER AS ACTIVITY
TRANSACTIONS            --SERVICE RATES--   PAGE-IN RATES      ----STORAGE----
AVG      2.00           ABSRPTN    181961   SINGLE     0.0     AVG     56146.9
MPL      2.00           TRX SERV   181961   BLOCK      0.0     TOTAL    112293
ENDED       0           TCB           8.0   SHARED     0.0     CENTRAL  112293
END/S    0.00           SRB           0.3   HSP        0.0     EXPAND     0.00
#SWAPS      0           RCT           0.0   HSP MISS   0.0
EXCTD       0           IIT           0.0   EXP SNGL   0.0     SHARED  3216.83
AVG ENC  0.00           HST           0.0   EXP BLK    0.0
REM ENC  0.00           APPL %        2.8   EXP SHR    0.0
MS ENC   0.00
```

There are three major differences in the interpretation of the data since the reported activity is address-space based:

- The TRANSACTION AVG indicates the number of server region address spaces active over the interval. Using this field, you can monitor the evolution of the number of servers between the MIN_SRS and MAX_SRS settings.
- STORAGE values are now filled in.

- Under normal conditions, the `APPL%` is typically very low. However, gradual increase in APPL% may be an indication of excessive garbage collector activity caused by a heap size too small, or a memory leak.

Workload Activity report (response time distribution)

The Workload Activity report provides response times for all service class periods and response time distribution information. The response time distribution is provided per service class period, for each service where a response time objective is defined. This is much more meaningful to the performance analyst than the average response time value.

Example 9-27 Response time distribution (partial view)

```
----------RESPONSE TIME DISTRIBUTION----------
    ----TIME----    --NUMBER OF TRANSACTIONS--   -------PERCENT-------  0  10  20  30  40  50  60
     HH.MM.SS.TTT    CUM TOTAL         IN BUCKET    CUM TOTAL   IN BUCKET  |..|..|..|..|..|..|..|..|..
  <  00.00.00.250         3716              3716         36.9        36.9  >>>>>>>>>>>>>>>>>
  <= 00.00.00.300         4129               413         41.0         4.1  >>>
  <= 00.00.00.350         4601               472         45.7         4.7  >>>
  <= 00.00.00.400         5041               440         50.0         4.4  >>>
  <= 00.00.00.450         5363               322         53.2         3.2  >>
  <= 00.00.00.500         5633               270         55.9         2.7  >>
  <= 00.00.02.000         9022              1732         89.5        17.2  >>>>>>>>>
  >  00.00.02.000        10075              1053          100        10.5  >>>>>>
```

Note that the interpretation of the data requires some knowledge of the application workload. If you have a coherent J2EE application, response time distribution would be concentrated into one peak, but if the application contains a mix of static html pages and J2EE transactions, the response time distribution may show two peaks reflecting the two different types of transactions, as we observe in the current case.

From this information, it is also possible to set an achievable percentile response time goal, a value commonly used in establishing service level agreements.

9.6.5 Performing problem diagnosis

In 6.2, “Running a performance health check” on page 180, we describe a methodology for diagnosing performance problems. This approach can be used for a WebSphere workload in the same way. Nevertheless, we add to this some considerations specific to a WebSphere workload.

You should look at the logical user requests that are not meeting response time expectations.

- If the logical user request response times within WebSphere server regions are OK, check components before server regions. This is mostly network-related and is outside the scope of this redbook.
- If the user request response times within WebSphere server regions are not satisfactory, examine these user requests more closely and continue.
- Map the logical user requests into WebSphere transactions. To the extent that you can’t map user requests to WebSphere transactions, you may need to guess and make assumptions.
- Define more than one period to honor the trivial transactions with more importance and more difficult goals.
- Segregate the WebSphere transactions into sets of good and bad transactions based on response times.
- Examine the resources used in each component of all bad transactions and identify common features and/or anomalies.
- Form hypotheses that explain 80% of the observations of good and bad transaction sets.

- Test these hypotheses, one by one, by gathering additional information.
- Be prepared to repeat this process when the identified problem has been resolved.

10

Using RMF to investigate Linux performance

This chapter describes RMF functions that can help you investigate Linux performance problems.

It provides an overview of the important performance areas of Linux, and how RMF monitors and reports performance data for Linux.

10.1 Linux

IBM embraced Linux early on because open standards are the key to integrating business from end-to-end, not only because they help simplify business integration, but also because they can help a company to deploy new solutions more quickly and accelerate the time to market for new products and services.

Linux, because it is open, also provides you with the flexibility to choose the best applications for your business needs and helps your investment to provide value in the future. By making it easier for multiple applications and middleware to work together, Linux can help boost productivity within your enterprise as well as with your key suppliers and partners.

zSeries servers offer one of the world's most reliable and scalable environments for Linux, ideal for running new and open eBusiness applications right alongside existing core business applications. zSeries servers help make it easy to manage multiple diverse workloads while balancing resources across those workloads for optimal performance and maximum utilization.

Here we discuss how to use RMF to monitor the performance of your Linux on zSeries effectively.

10.1.1 Linux on zSeries

You can install Linux on zSeries in a virtual machine running under the control of the z/VM operating system or in a logical partition (LPAR).

When running in an LPAR, the classic RMF components provide performance data about that LPAR, for example, the Postprocessor Partition Data report or the Monitor III CPC report.

But using the RMF Linux data gatherer (rmfpms), you also get a view inside your Linux operating system and what's going on there. The data is available either via RMF Performance Monitoring (RMF PM) or using the RMF Web browser interface. This view is available for Linux running in an LPAR or running as VM guest.

Here we focus on how to use rmfpms and RMF PM to get performance data.

For more details about the installation and use of rmfpms, refer to "RMF Linux data gatherer" on page 138. For information about the installation and use of RMF PM, refer to "RMF Performance Monitoring" on page 74.

10.1.2 Linux performance metrics

RMF supports the following performance areas of Linux, which are discussed in this chapter:

- Linux system and application resource metrics
- Network resource metrics
- CPU resource metrics
- CPU load average
- File system resource metrics
- Memory resource metrics

System resource metrics

RMF provides metrics to monitor your Linux system itself and also your Apache HTTP server.

Linux System metrics

Linux is a multiprogramming environment. Whenever an application is started, a process is created. Multiprogramming is supported by giving each process in the runnable state queue an opportunity to run on the CPU in turn. The procedure for swapping between runnable processes is to name a context switch.

rmfpms also offers Linux kernel-related metrics:

- System - rate of context switches (per second)

  Context switches occur when the operation switches from running one process to running another.

- System - rate of processes created (per second)

In Figure 10-1 we see that the `rate of processes created (per second)` is increasing. An increasing number of processes leads to an increasing `rate of context switches (per second)`, as we also can see in the report. At the time stamp 14:00:00, about 36 processes are created in one minute. We have here the highest number of context switches. After the peak, at 14:01:00, the number of processes created by the system decreases, and the context switch rate decreases, too. This means that most of the created processes have ended at 14:01:00; otherwise, the context switch rate would be constant.

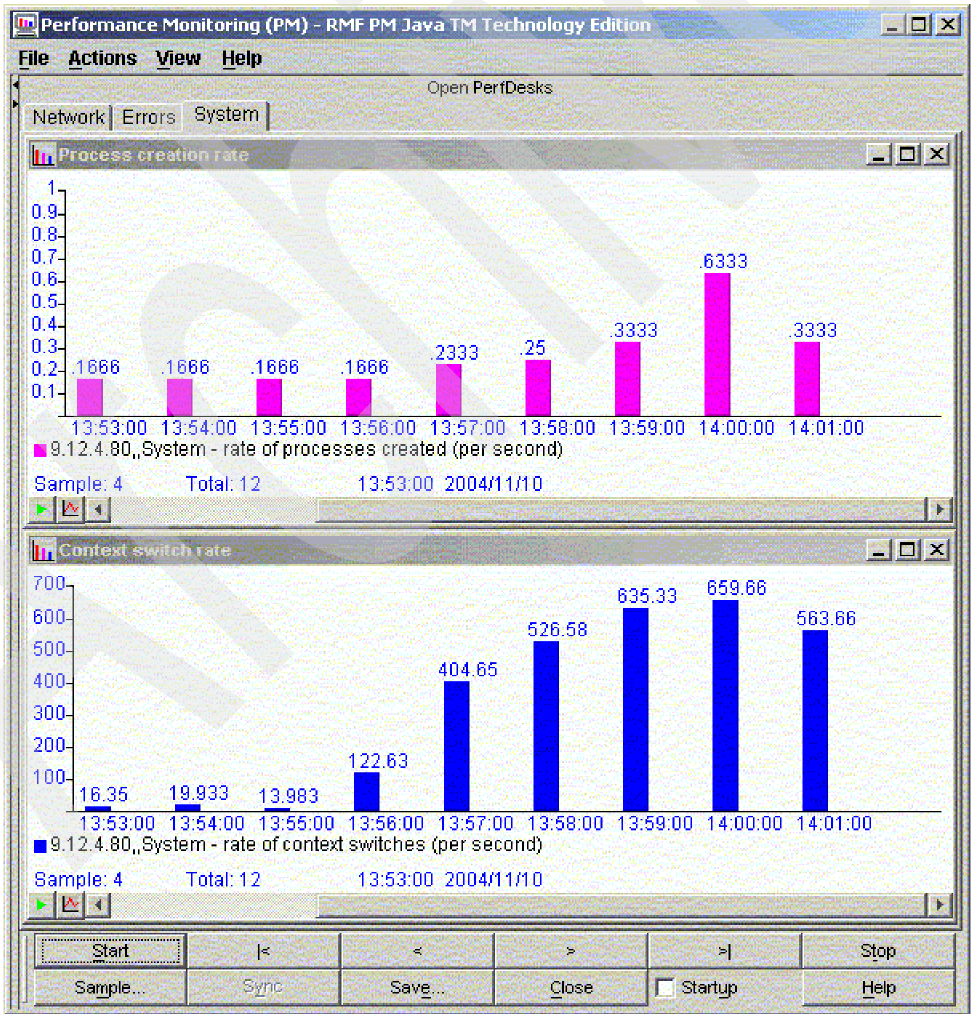


Figure 10-1 System metrics

Apache HTTP resource metrics

When talking about performance of a Web server, one thing is important: the throughput. This throughput for Web servers is measured by using two different units. In both cases, large numbers are indicators for good performance.

- Requests per second

  This is usually the first number you should review when benchmarking a Web server.

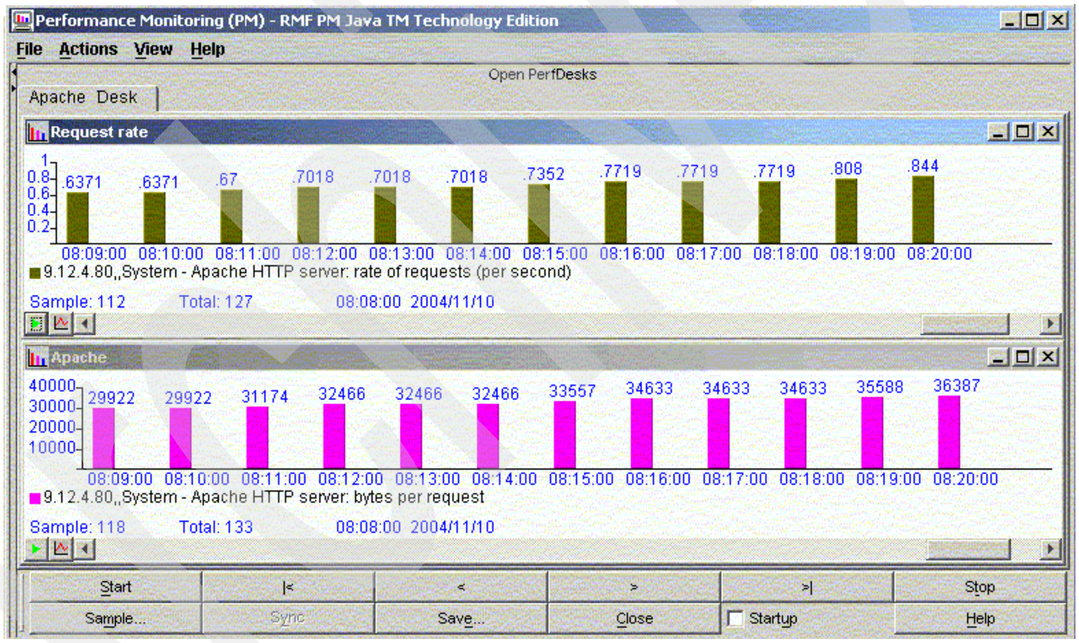
- Bits transmitted per second

  This information is the bandwidth of your Web server, and tells you whether the Web server is saturating the wire.

Therefore, we focus on the following two metrics of rmfpms:

- System - Apache HTTP server: rate of requests (per second)
- System - Apache HTTP server: bytes per request

Figure 10-2 shows an Apache HTTP server with less but growing load, only about one request per second. In relation to the growing number of requests, the transfer rate is growing as well. This is good because it means that we have no contention.



🟩 9.12.4.80,,System - Apache HTTP server: rate of requests (per second)

Sample: 112 Total: 127 08:08:00 2004/11/10

🟪 9.12.4.80,,System - Apache HTTP server: bytes per request

Sample: 118 Total: 133 08:08:00 2004/11/10

☐ Startup

Figure 10-2 Apache HTTP server metrics

Network resource metrics

For applications using network services, the network performance is an important part of their response time. You can have superb CPU and I/O power on your system, but your network could still be a weak link in the chain. Therefore, the network throughput is important.

One important metric is the number of transmitted packets per second. But since the size of a packet is not constant, we need the number of transmitted bytes also. This is important since small packets can result in larger CPU usage. Also, the number of transmit errors is important because corrupted packets are resent, causing additional network load.

We focus on the following rmfpms metrics:

- Net - packets received per second
- Net - packets transmitted per second
- Net - bytes received per second
- Net - bytes transmitted per second
- Net - receive errors per second
- Net - transmit errors per second

In Figure 10-3, we see that the number of transmitted packets is twice as high as the number of received packages. For example, at 13:20:00 there are 1496 transmitted packets and 670 received packets. But when we have a look at the transferred byte rate, we see that we received around 32 KB and transmitted only about 46 Bytes. So a transmitted package is much smaller than a received packet. The package size of transmitted and received packages stays fairly constant.

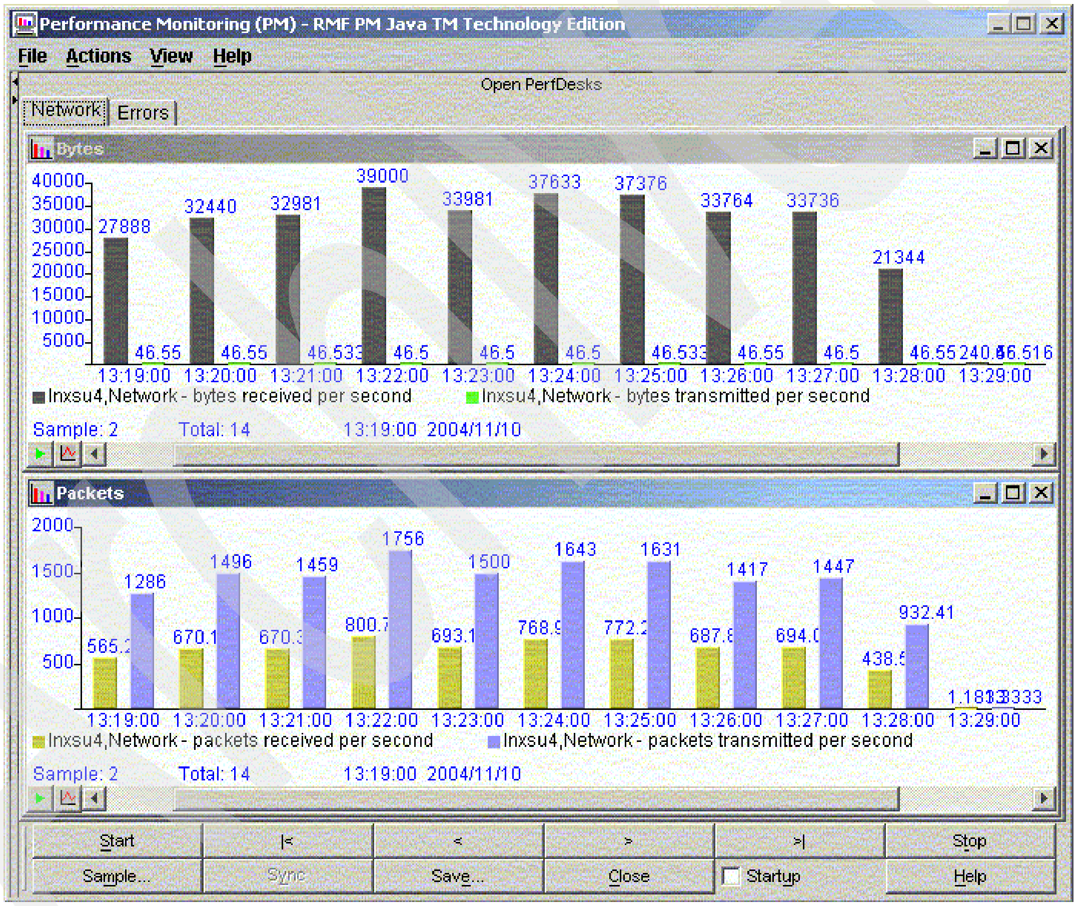


Figure 10-3 Network throughput

Figure 10-4 shows us a report that tells us how good the network is, no receive or transmit errors at all.

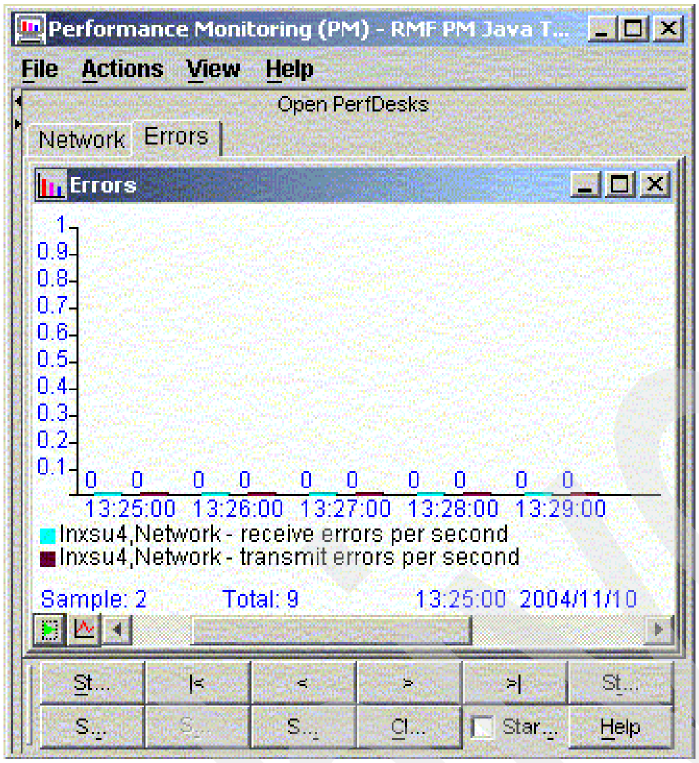


Figure 10-4 Network errors

CPU resource metrics

For servers whose primary role is that of an application or database server, the CPU is a critical resource and often a source of performance bottlenecks.

Therefore, we have to check similar characteristics as we do when monitoring a z/OS host system:

- What is the utilization of the CPU; is it close to its limit; do we have any capacity left?
- Is there any CPU constraint; how much does it hurt us; is there only one process waiting for the CPU resource or do we have a long queue of processes that are being delayed?
- Which are the heavy CPU using processes; do we need to move some workloads to other systems if we have delays?

Therefore, we focus on the CPU metrics:

- CPU - load average

  Average length of the queue of processes ready to run. This is the number of processes that are waiting for the busy CPU. This is actually the major indication for CPU contention because it works in virtualized and non-virtualized environments quite well. A high load average in relation to the number of processors means that your system is short on CPU because lots of processes are waiting for CPU resources.

- CPU - % cpu total active time

  Percentage of CPU total active time, averaged over all CPUs.

- CPU - % cpu total active time by processor

  Percentage of CPU total active time by processor.
- CPU - % cpu time total by process

  Percentage of all CPU time by process.

All CPU percentages externalized from rmfpms are purely based on Linux data, with operating system view. So if you see, for example, that a process used 30% cpu, it means that this process used 30% of all CPU resources this operating system was able to use, but it does not tell you how many CPU resources the operating system has got from PR/SM LPAR or from z/VM hypervisor.

Figure 10-5 gives us an overview of the CPU usage. The `% cpu total active time` shows that sometimes we have a heavy load on our system, for instance, using up to 54% of the CPUs at 10:52:00. On the `% cpu total active time by processor`, we see the load on our two CPUs: cpu0 and cpu1. The `load average` shows that we have some minor delays. At time stamp 10:52:00, we have an average processor queue length of 0.91. This means that for 91% of the 60-second interval, one process was waiting on the processor queue.

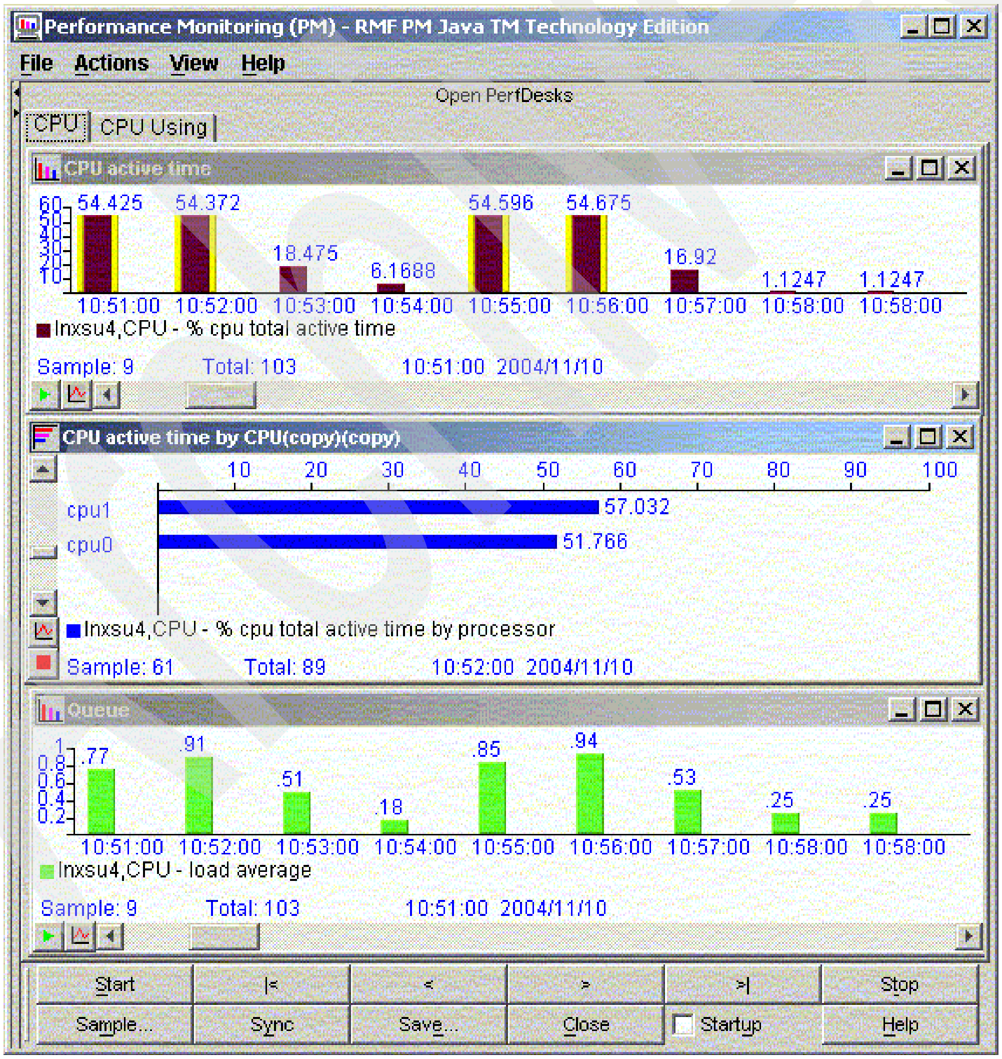


Figure 10-5 CPU usage

In Figure 10-6, the `% cpu time total by process` shows that our Apache HTTP server is a heavy CPU user, while the other processes are minor CPU users.

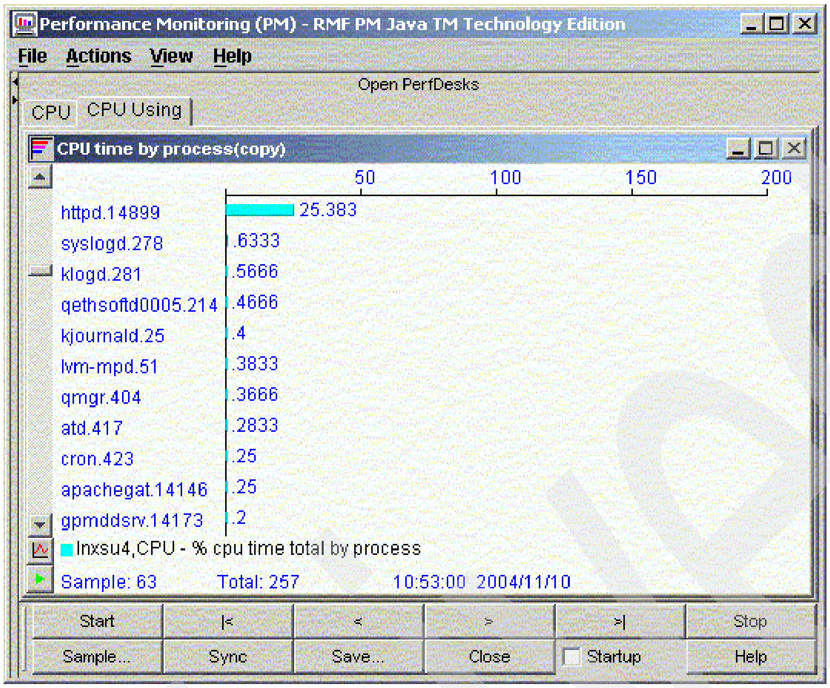


Figure 10-6 CPU usage by process

File system resource metrics

For the file system, RMF offers either real Linux file system metrics, or DASD performance data.

File system metrics

An important concept to remember is that to Linux, everything is a file. Therefore, the file system itself is very important to the operating system. If the file system runs out of space, this may have serious impact on the operating system. Therefore it is important to have the file system monitored.

So, instead of the % used metrics, let's focus on the metrics that show the available free space:

- Filesystem - % free by file system
- Filesystem - % of space free
- Filesystem - available (in MB) by file system

In Figure 10-7 we see that, overall, about 14% free space is left. Additionally, we see that it is even worse. Of our two file systems /dev/dasddb1 and shmfs, the file system shmfs is completely empty with 251 MB. So it seems that this file system is not used at all, but on our main file system /dev/dasddb1, there is less than 4% space left, or only about 80 MB is left. Therefore, we should take a look and free some space, or move some files to the currently unused file system.

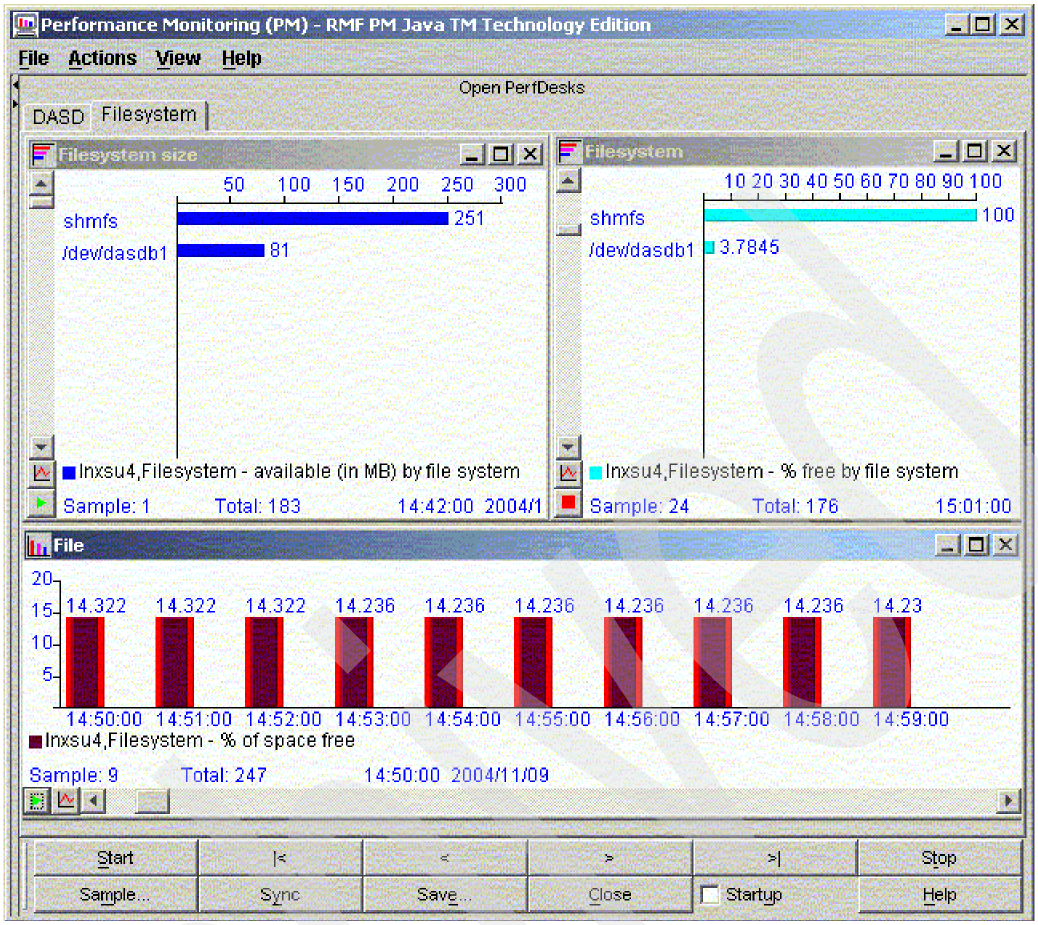


Figure 10-7 Linux file system metrics

DASD performance

DASD metrics are supported by RMF. There are two important metrics:

- Filesystem - DASD I/O requests per second
- Filesystem - DASD I/O average response time per request

The number of DASD I/O requests per second tells you the I/O usage of the DASD from the Linux system. One request of the Linux file system may not result in one request to the DASD.

Therefore, DASD I/O average response is the overall measure of the health of a device's operation. I/O response time is a component of user response time. It is often the dominant component. High I/O response times can delay user jobs and increase response times to unacceptable levels.

Figure 10-8 shows the behavior that occurs normally with an increase of DASD activity will cause you to see the response times increase. In the report example, it is a result of an application that started to write huge files to the file system. We also see that even if the number of DASD I/O requests has stabilized, the response time still gets worse.

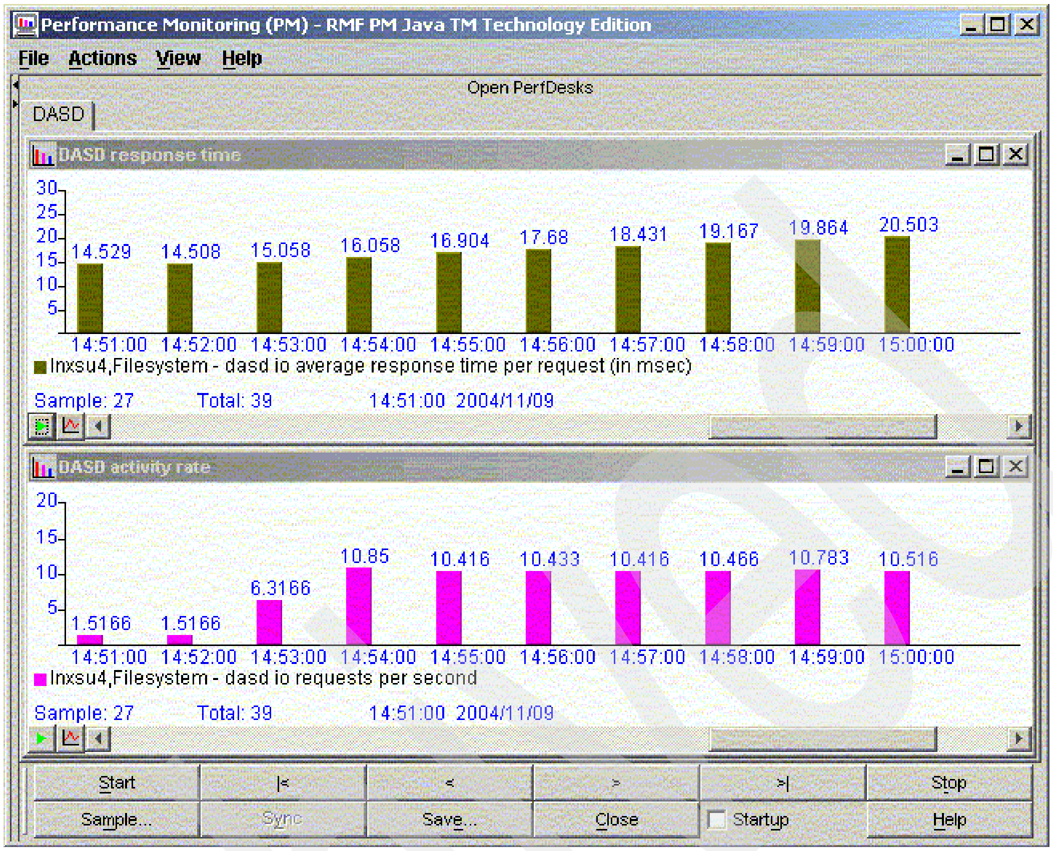


Figure 10-8 Linux DASD I/O

Memory resource metrics

On a Linux system, many programs run at the same time; these programs support multiple users and some processes are used more often than others. Some of these programs use a portion of memory while the rest are sleeping (idle). The operating system uses an algorithm to control which programs will use physical memory, and which are paged out. Page space is a file created by the operating system on a disk partition to store user programs that are not currently used. A page fault occurs when a process tries to access a virtual address that is not in physical memory.

In Linux, page faults are either minor or major. A major fault requires an I/O operation to complete, such as a page swap from disk. Minor faults are handled without an I/O and include things such as a Copy on Write request for a shared page or a request for a zeroed page. Accessing the disk will slow your application down considerably.

Therefore, it is important to monitor the swap space itself and the paging activity.

- Memory - used swap space in MB
- Memory - free swap space in MB
- Memory - number of pages swapped in per second (in 4 KB pages)
- Memory - number of pages swapped out per second (in 4 KB pages)

Figure 10-9 shows a system with no memory problems at all. No swap space is used, and 140 MB is left. Also no page activity is shown.

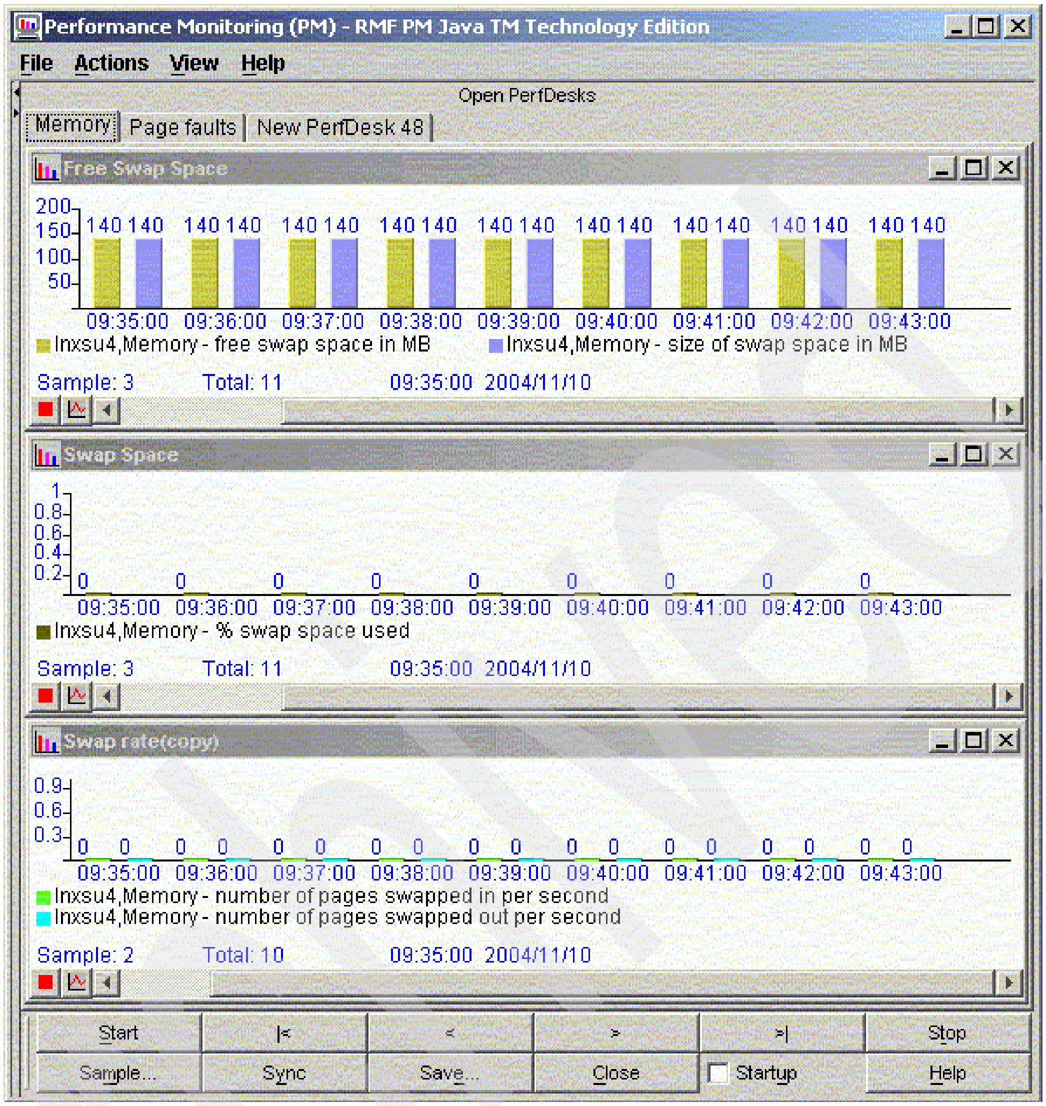


Figure 10-9 Swap space metrics

When we have a problem, we need to drill down to discover which processes are suffering from the page fault. We also want to know what are the big memory consumers in our system? Perhaps we can resolve this situation by moving applications to another Linux system instead of spending more memory.

We have a look at the following metrics:

- Memory - virtual memory size by process (in bytes)
- Memory - minor page fault rate including children by process
- Memory - major page fault rate including children by process

Additionally, the resident set size of a process is the in-memory portion of the virtual address space of a process and sometimes is an interesting metric. This can give you an idea of the memory consumption of the given process.

Figure 10-10 shows a slight change of the situation. We see a few major page faults, which result in I/O operations. We also see that the Apache HTTP server processes are using about 90 MB memory.

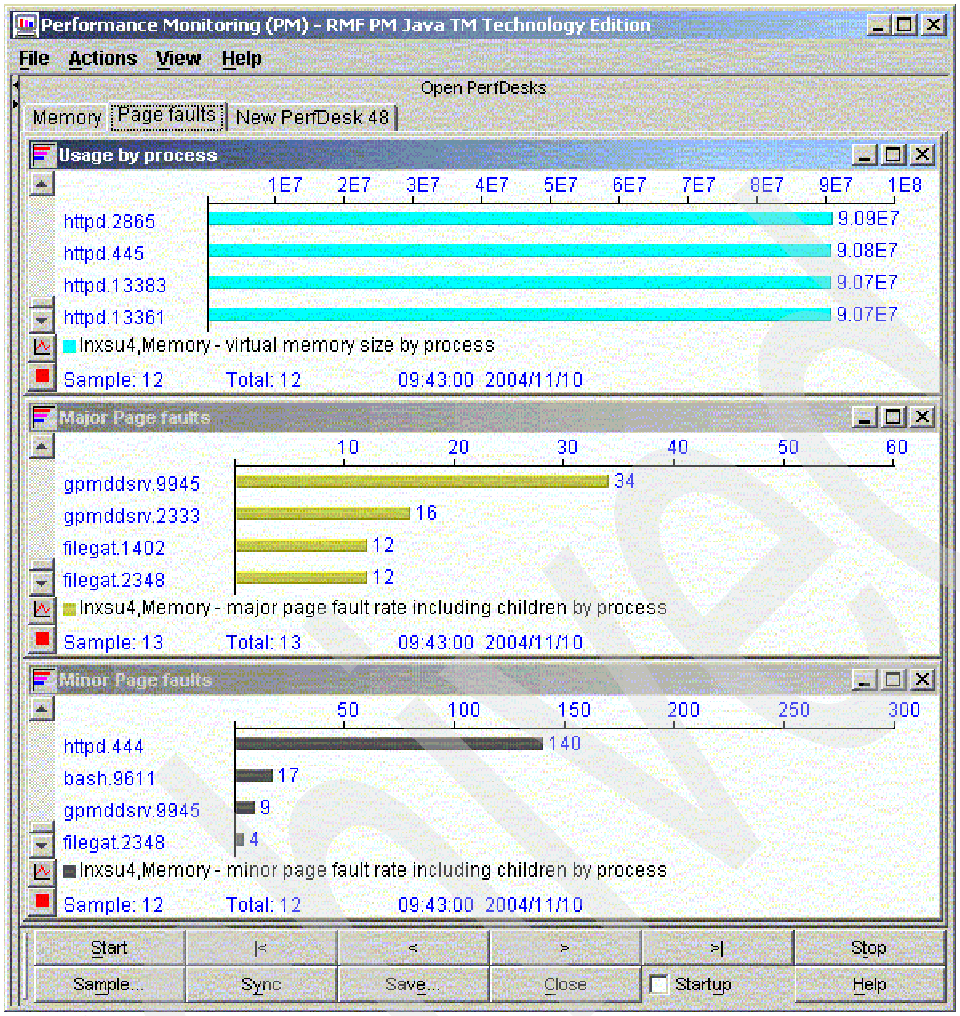

Performance Monitoring (PM) - RMF PM Java TM Technology Edition
File Actions View Help
Open PerfDesks
Memory
Page faults
New PerfDesk 48
Usage by process
1E7 2E7 3E7 4E7 5E7 6E7 7E7 8E7 9E7 1E8
httpd.2865 9.09E7
httpd.445 9.08E7
httpd.13383 9.07E7
httpd.13361 9.07E7
lnxsu4,Memory - virtual memory size by process
Sample: 12 Total: 12 09:43:00 2004/11/10
Major Page faults
10 20 30 40 50 60
gpmddsrv.9945 34
gpmddsrv.2333 16
filegat.1402 12
filegat.2348 12
lnxsu4,Memory - major page fault rate including children by process
Sample: 13 Total: 13 09:43:00 2004/11/10
Minor Page faults
50 100 150 200 250 300
httpd.444 140
bash.9611 17
gpmddsrv.9945 9
filegat.2348 4
lnxsu4,Memory - minor page fault rate including children by process
Sample: 12 Total: 12 09:43:00 2004/11/10
Start
|<
<
>
>|
Stop
Sample...
Sync
Save...
Close
□ Startup
Help

Figure 10-10 Detailed memory view

A

Formulas and laws in performance management

This appendix describes some of the laws, rules, and principles used frequently in performance management. The items covered are:

- Little's Law
- Markov's Equation
- 80/20 Rule
- Partition's Law
- Law of the Diminished Returns
- Principle of Locality

Little’s law

Figure A-1 illustrates Little’s law.

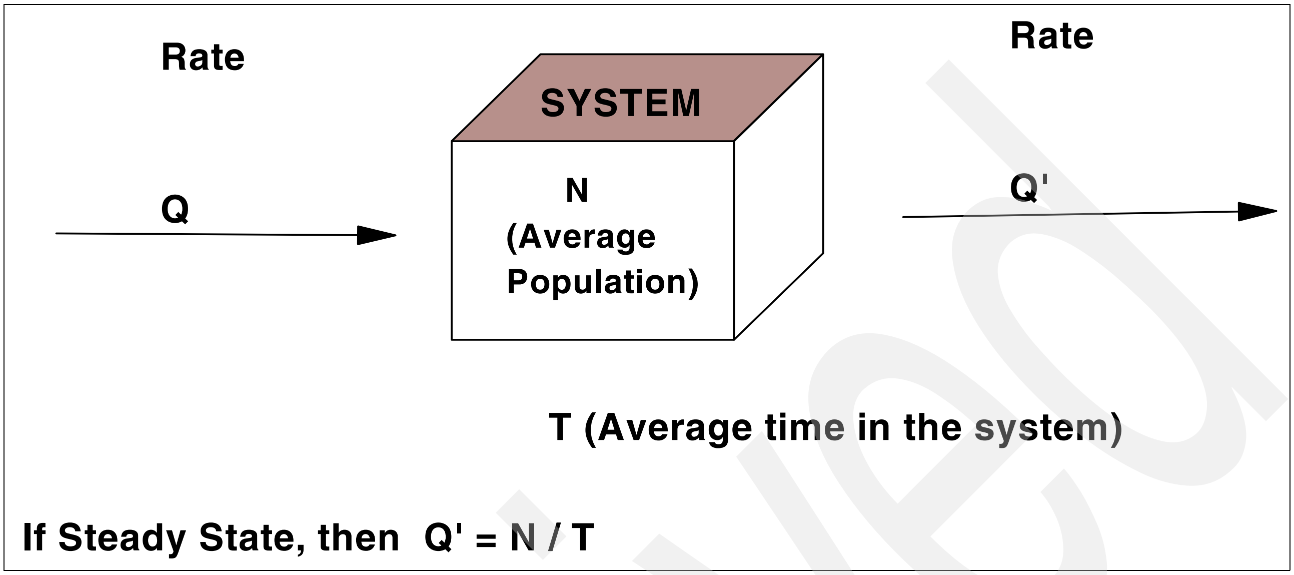

Rate
SYSTEM
Rate
Q
N
(Average
Population)
Q'
T (Average time in the system)
If Steady State, then Q' = N / T

Figure A-1 Little’s law

Imagine a system (for example, a computer or a bank branch office) where elements (transactions in a computer, customers in a bank branch office) are entering at a specific rate Q and are leaving at a specific rate Q', and where T is the average time spent by each element in the system and N is the average population of elements in the system. If the system is in steady state, we can say that Q' is equal to N/T. In an approximation, we may say that in this state Q=Q'.

In other words: The average number of things in the system is the product of the average rate at which things leave the system and the average time each one spends in the system, and if there is a gross *flow balance* of things entering and leaving, the exit rate is also the entry rate. Peter Denning succinctly phrases this rule as: *The average number of objects in a queue is the product of the entry rate and the average holding time.*

There are many applications for Little's law, such as in the following formula:

```
Q = N / (Tr + Tt)
```

where:

N Average number of users sending transactions (logged on)
Tr Response time
Tt Average thinking time of these users

Using this formula together with RMF data, z/OS systems programmers can derive the User Thinking time (Tt). Then the programmers can get some good-natured revenge for complaints they hear by calling the users when their thinking time is slow.

```
Avg_IOSQ TIME = IOS_ Avg_ Queue_Length / IOs_Rate
```

This formula is used by RMF to capture the average IOS queue time during an I/O operation. The IOS queue time is the time the I/O request spends, on average, delayed in the operating system queue because the previous I/O request from the same operating system towards the same device has not completed yet.

Markov's equation

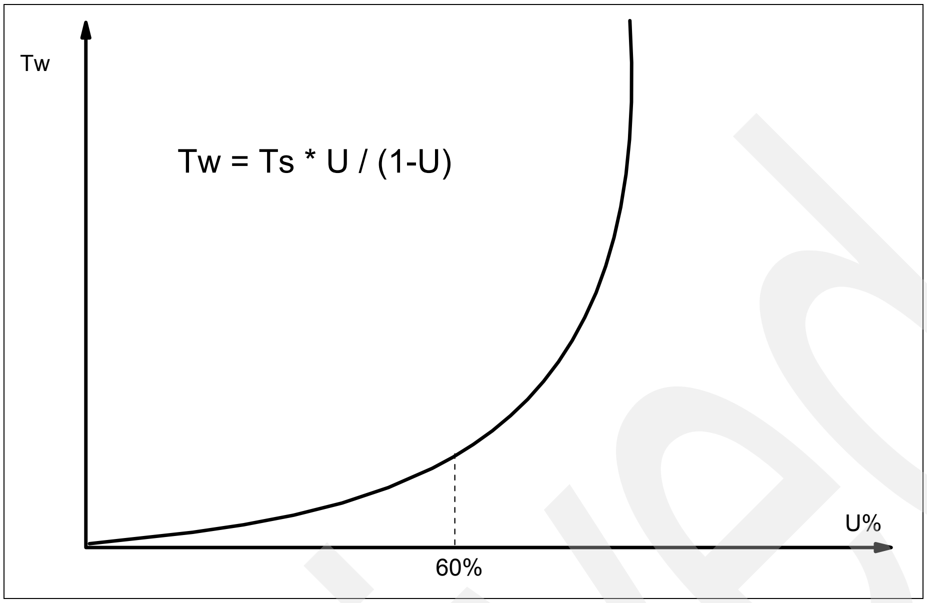


Figure A-2 Markov's equation curve

Markov's equation, graphed in Figure A-2, shows the function `Tw= f(Ts,U)` that is, the relationship of Average Wait Time (Tw) with Service Time (Ts) and the Utilization (U):

```
Tw = Ts * U / (1 - U)
```

The formula applies when:

- There is only one server
- The system is in steady state
- The distribution of the inter arrival time (time between two consecutive transactions using the server) is exponential.

This is a simplification of the queuing theory described in the M/M/1 formula where both the inter arrival time and the service time are exponential. For additional details on the M/M/1 formula, refer to the redpaper *A Solution to the Multiserver Priority Queuing Model,* REDP-3969. As you can see in Figure A-2, for utilizations greater than 60% the Tw rises drastically, causing a degradation in Tw and consequently in Tr (average response time).

When U is trending to one, Tw trends to infinity. It means that, if an operating system is unable to block transactions entering the system when the demand is too high, the system may run out of resources soon. When CPU reaches 100%, the memory could be exhausted by so many ongoing transactions.

z/OS, in contrast to other operating systems, has mechanisms for workload rejection. Therefore, many z/OS installations are able to work at close to an average CPU utilization of 100% for a long time without the need of an IPL although sometimes you might assist to a failure, mainly due to serialization.

Regarding the CPU resource, it is possible to have an average utilization greater than 35% in z/OS. If the critical work (z/OS itself and online applications, for example) consumes up to such a figure, all the increase in Tw is experienced by the low priority transactions such as batch. In order to make the previous statement true, we need to have a preemptable operating system, such as z/OS. *Preemptability* here means to respect the dispatching priorities of the dispatchable units (TCB/SRB) associated with the transactions.

80/20 Rule

This rule, illustrated in Figure A-3, states that 80% of the transactions consume 20% of the resources and 20% of the transactions consume 80% of the resources. It implies that the trivial ones (the majority) are responsible for 20% of the consumption.

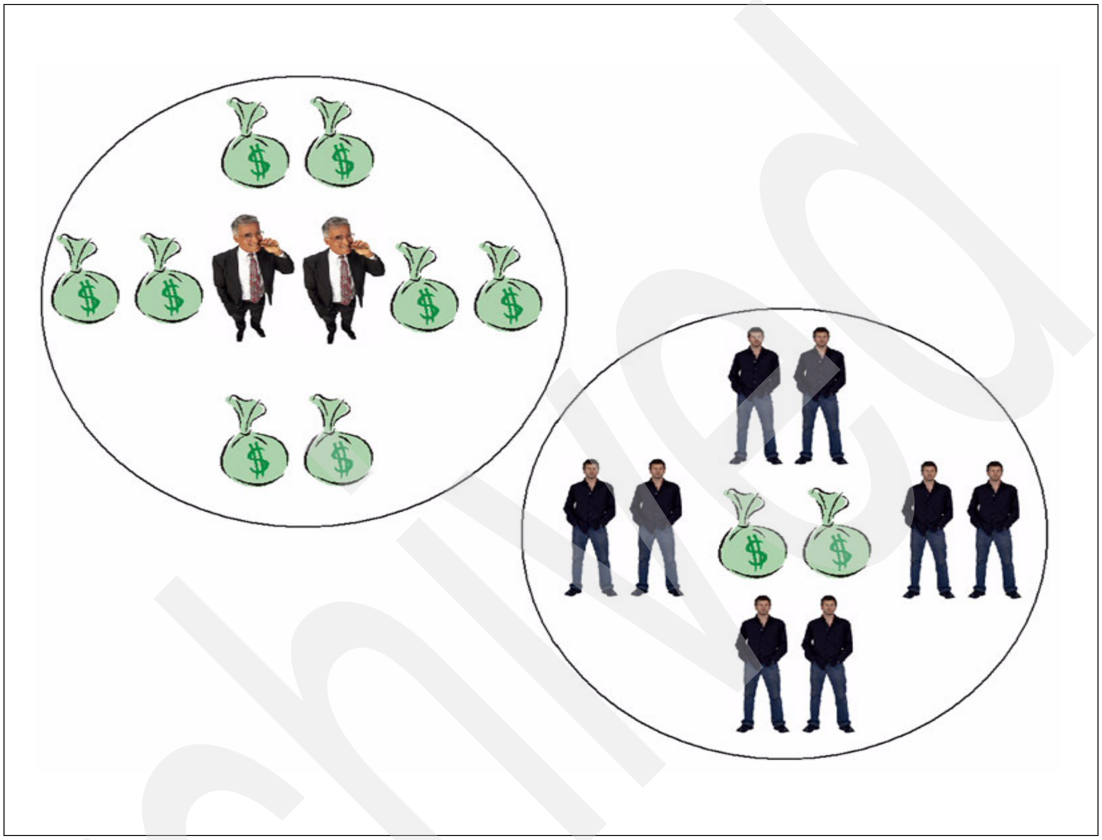


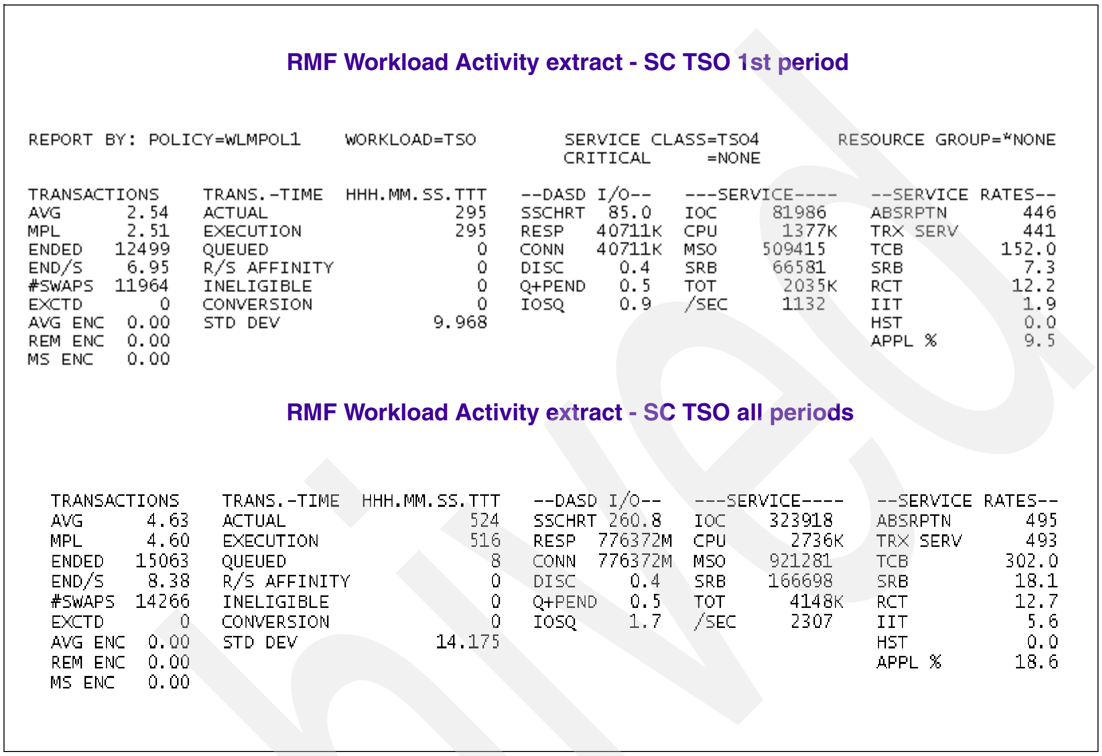
Figure A-3 The distribution of the economy in a society

A major consequence of this rule is the idea of prioritizing trivial online transactions. It helps performance in general because:

- Low burning in the system results from the low system consumption.
- 80% of the user population is happy.
- Getting rid of online transactions frees z/OS of control blocks manipulation, thus saving CPU cycles. The next transaction will come many seconds later (allowing for user thinking time).

80/20 rule and service class periods

Figure A-4 illustrates an RMF Workload activity report for the transactions associated with the TSO Service Class.



RMF Workload Activity extract - SC TSO 1st period

```
REPORT BY: POLICY=WLMPOL1    WORKLOAD=TSO        SERVICE CLASS=TSO4        RESOURCE GROUP=*NONE
                                                 CRITICAL      =NONE

TRANSACTIONS     TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--    ---SERVICE----    --SERVICE RATES--
AVG       2.54   ACTUAL                 295   SSCHRT  85.0    IOC      81986    ABSRPTN       446
MPL       2.51   EXECUTION              295   RESP   40711K   CPU      1377K    TRX SERV      441
ENDED    12499   QUEUED                   0   CONN   40711K   MSO     509415    TCB         152.0
END/S     6.95   R/S AFFINITY             0   DISC     0.4    SRB      66581    SRB           7.3
#SWAPS   11964   INELIGIBLE               0   Q+PEND   0.5    TOT      2035K    RCT          12.2
EXCTD        0   CONVERSION               0   IOSQ     0.9    /SEC      1132    IIT           1.9
AVG ENC   0.00   STD DEV              9.968                                     HST           0.0
REM ENC   0.00                                                                  APPL %        9.5
MS ENC    0.00
```

RMF Workload Activity extract - SC TSO all periods

```
TRANSACTIONS     TRANS.-TIME  HHH.MM.SS.TTT   --DASD I/O--     ---SERVICE----    --SERVICE RATES--
AVG       4.63   ACTUAL                 524   SSCHRT 260.8     IOC     323918    ABSRPTN       495
MPL       4.60   EXECUTION              516   RESP  776372M    CPU      2736K    TRX SERV      493
ENDED    15063   QUEUED                   8   CONN  776372M    MSO     921281    TCB         302.0
END/S     8.38   R/S AFFINITY             0   DISC      0.4    SRB     166698    SRB          18.1
#SWAPS   14266   INELIGIBLE               0   Q+PEND    0.5    TOT      4148K    RCT          12.7
EXCTD        0   CONVERSION               0   IOSQ      1.7    /SEC      2307    IIT           5.6
AVG ENC   0.00   STD DEV             14.175                                      HST           0.0
REM ENC   0.00                                                                   APPL %       18.6
MS ENC    0.00
```

Figure A-4 RMF workload activity report extracts for TSO SC

Usually when a transaction starts we do not know if it is trivial or not. Then, you should use the concept of periods in a service class (SC) in order to prioritize the trivial transactions. The existence of periods in a SC are used to distinguish trivial resource consumption transactions from the heavy ones. All transactions in such SC always start running in the first period (the trivial and the heaviest). In this period, you should define a higher importance and a more challenging goal to be obtained. Each non-last period has a keyword named DUR, that identifies the amount of service units consumed in this period by a transaction before being moved to the next period. Then, if this number was chosen correctly, all the trivial (80% of the total) should finish in the first period and enjoy a good (high) priority. The others, when reaching the duration limit, are migrated to another period, where the priorities (generated by value of the goal and importance) are not so high.

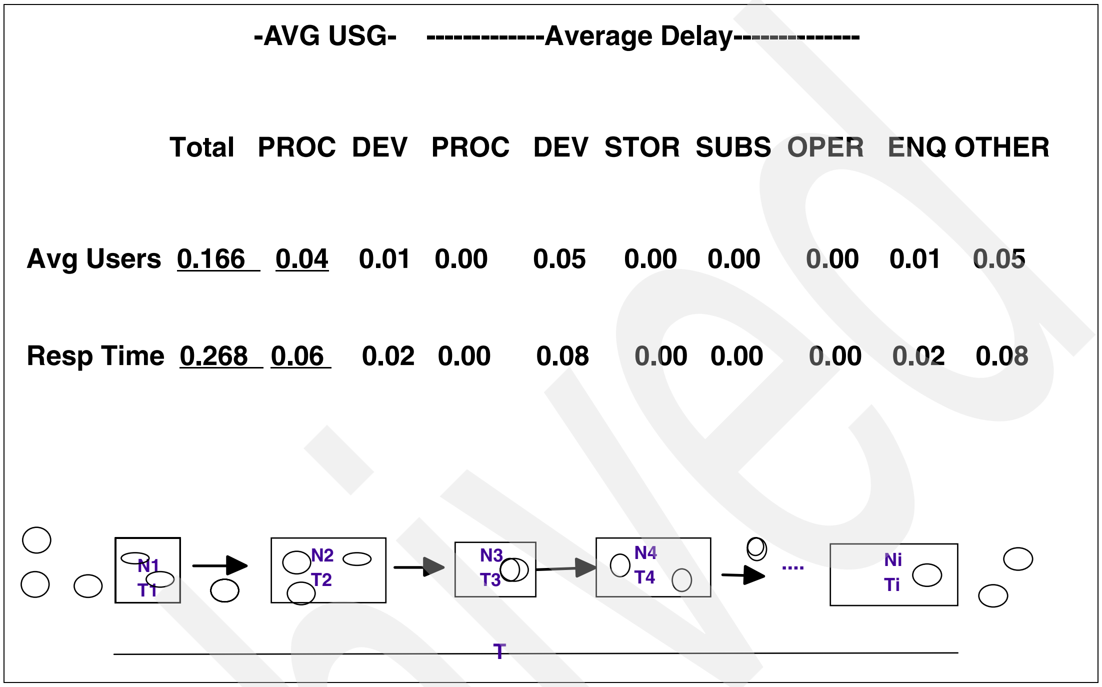
Looking at the sample RMF report, we want to verify that the duration value follows the 80/20 rule. We note the following from the report:

- Ended transactions in 1st Period: 12499 transactions
- Total: 15063 transactions

Then, 12499 / 15063 = 0.82 * 100 = 82% finish in the 1st period. This implies that the DUR parameter is numerically correct for the service class in this time interval as presented by RMF. If the ratio is higher than 80%, you are protecting more transactions that you should, and therefore you should decrease the DUR. If the ratio is less than 80%, do the opposite.

Partition's law

Let us imagine a process formed by serial stages, where transactions are processed as in a pipeline. In Figure A-5, we have *i* stages and the little circles represent the transactions going through the stages.



	-AVG USG-		------------Average Delay-------------							
	Total	PROC	DEV	PROC	DEV	STOR	SUBS	OPER	ENQ	OTHER
Avg Users	0.166	0.04	0.01	0.00	0.05	0.00	0.00	0.00	0.01	0.05
Resp Time	0.268	0.06	0.02	0.00	0.08	0.00	0.00	0.00	0.02	0.08

Figure A-5 RMF Group report extract

Ni is the average number of transactions being executed in the *i* stage. *Ti* is the average consumed in the *i* stage. If the flow of transactions is in a steady state, the partition's law states that:

```
Ti = (Ni / N) * T
```

Where:

N Average total population in the full process (all stages).
T Average total time in the full process (all stages).

Using this law, we can derive each partial time in each stage (an important figure performance wise), only by knowing the distribution of the population.

In Figure A-5, the extract of the Group RMF report shows a portion of a transaction response time. The time in each stage is calculated from the sampling of the average population in each stage. For example, for the pictured service class figures:

- N (average number of active transactions) is 0.166.
- T (average response time) is 0.268.

Then, the average time spent by those transactions executing CPU is:

0.04 / 0.166 * 0.268 = 0,06 msec

In a practical case, if you are not happy with the transaction average response time, you can see by the report that a major contributor to the response time is the I/O delay and consequently direct your efforts to improve your I/O.

Law of diminished returns (LDR)

Figure A-6 illustrates the law of diminished returns, which states:

> When one increases the amount of resources, one also increases the overhead to manage these resources. There is a point beyond which there is no gain in increasing the resource capacity.

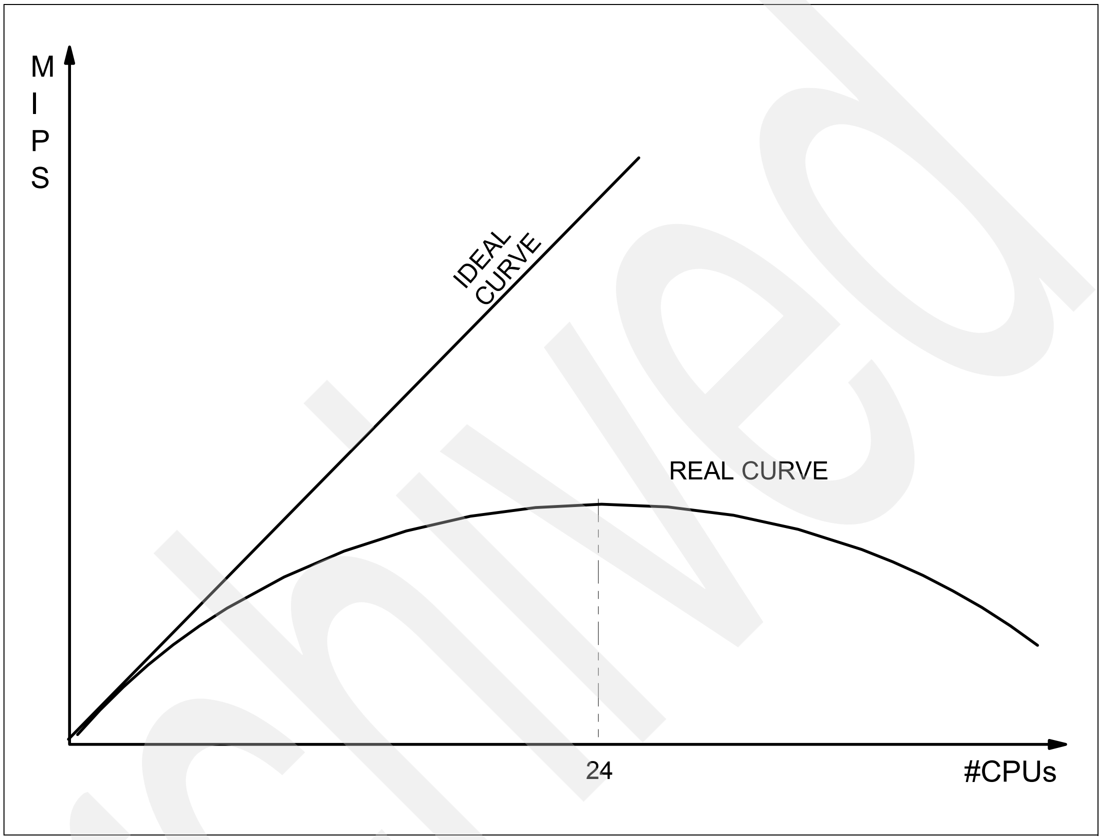
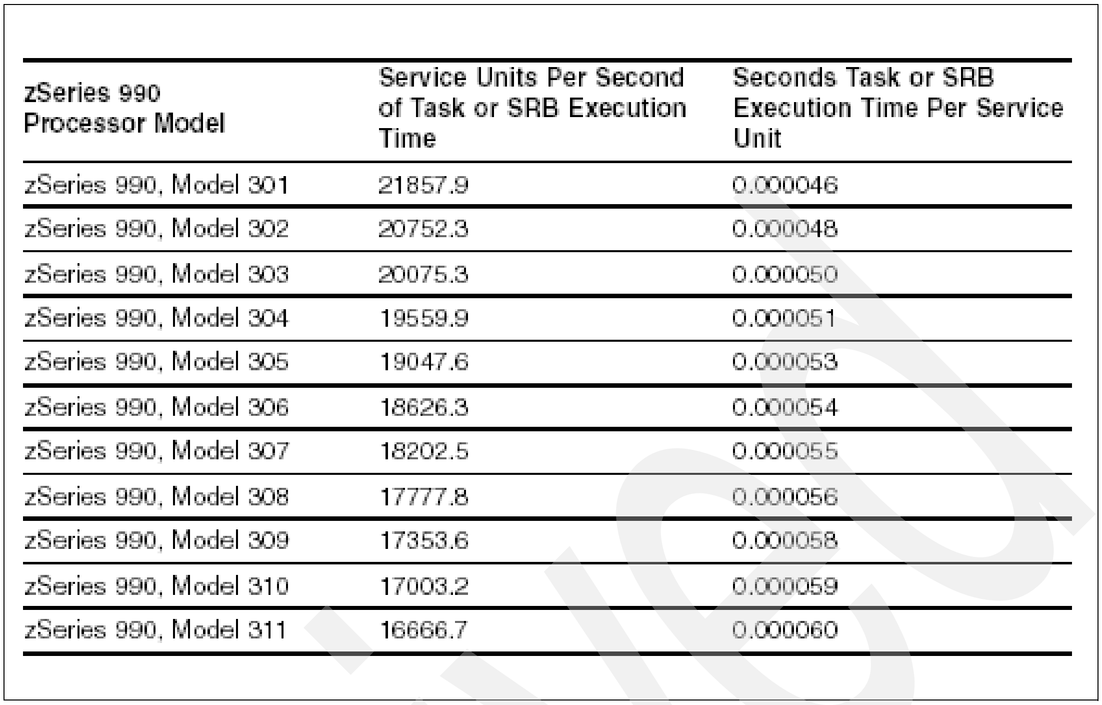


Figure A-6 Diminished returns law curve

With regard to CPUs, the maximum number of such resources sharing the same memory and managed by the same code of z/OS is currently 24.

Beyond that, there is no performance gain for the moment. There are plans in the near future to decrease the overhead and to be able to go beyond 24 CPUs. There are many reasons for decreasing the speed of a CPU when sharing a common memory with other CPUs in a commercial environment. One of them has to do with L1 cache invalidation. Every modification in the content of L1 cache in one CPU, forces the overhead of verifying if such content is present in other L1 cache and if it is, a cross invalidation process is executed. Figure A-7 shows a table where the second column (Service Units per Second of Task or SRB Execution Time) indicates the SRM Constant. This metric measures the speed (in SUs per sec) of each processor in the different CECs. In the example of z990 models, you can observe that the CPU speed decays when the number of CPUs increases.



zSeries 990 Processor Model	Service Units Per Second of Task or SRB Execution Time	Seconds Task or SRB Execution Time Per Service Unit
zSeries 990, Model 301	21857.9	0.000046
zSeries 990, Model 302	20752.3	0.000048
zSeries 990, Model 303	20075.3	0.000050
zSeries 990, Model 304	19559.9	0.000051
zSeries 990, Model 305	19047.6	0.000053
zSeries 990, Model 306	18626.3	0.000054
zSeries 990, Model 307	18202.5	0.000055
zSeries 990, Model 308	17777.8	0.000056
zSeries 990, Model 309	17353.6	0.000058
zSeries 990, Model 310	17003.2	0.000059
zSeries 990, Model 311	16666.7	0.000060

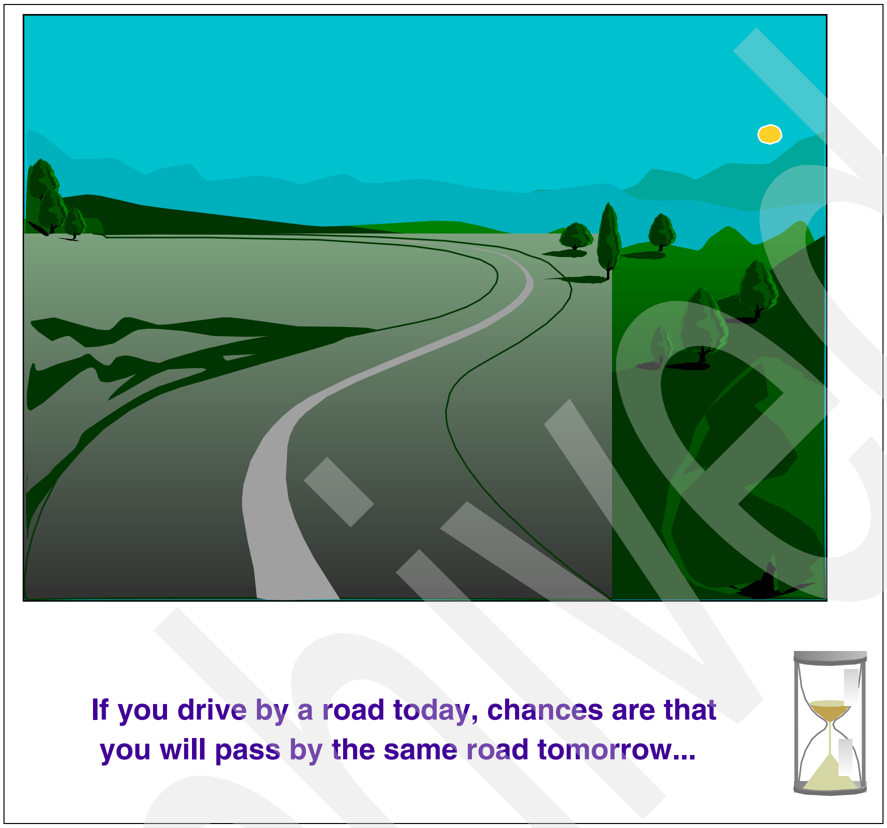
Figure A-7 SRM Constant table

However, in science and technology it is common to have computers with hundreds of processors without performance degradation. (The same is true of mammal brains, which have trillions of neurons.) This sort of machine is designed to solve non-commercial problems, where data is not scalar but in matrixes. When we add two matrixes, we add correspondent pairs of elements and parallel processing among several pairs can be done. This type of data allows massive independent parallel processing because the result of an operation with a correspondent pair of elements does not depend on the other pairs of operations. Therefore, adding more processors results in faster processing.

In a commercial environment the parallelism is used to execute several transactions at the same time, not to speed up each transaction. This happens because of the transactions' natural sequential logic.

Principle of locality

In a computational commercial environment the following principle is valid: *If a fact happens right now, there is a good chance that it will happen in the near future*.



If you drive by a road today, chances are that you will pass by the same road tomorrow...

Figure A-8 Principle of locality

We can explain such behavior with two examples:

- Instructions. Less than 20% of a program’s code is executed frequently; it is a sort of kernel containing the major logic to be applied to each I/O record. The other instructions are there (in the program) just for unusual (exceptional) events. So, if one instruction is executed now, there is a good chance that it belongs to the kernel and it will be executed again soon.
- Data. If a client in a bank is accessing their personal data now, there is a good chance that the client will do that shortly.

This behavior allows the implementation of a least recently used (LRU) algorithm, which keeps in memory the piece of code and data most referenced. This memory could be: the CEC Central storage (making feasible virtual storage), DASD Cache, DB2 buffer pool.

However, there are examples of processing where the principle does not apply and consequently the LRU algorithm does not apply, for example, in I/O sequential processing where every processed record is not going to be processed again.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see "How to get IBM Redbooks" on page 334. Note that some of the documents referenced here may be available in softcopy only.

- *zOS Intelligent Resource Director,* SG24-5952
- *zSeries Application Assist Processor (zAAP) Implementation*, SG24-6386
- *UNIX System Services z/OS Version 1 Release 4 Implementation*, SG24-7035
- *Hierarchical File System Usage Guide*, SG24-5482
- *VSAM Demystified,* SG24-6105
- *A Solution to the Multiserver Priority Queuing Model*, REDP-3969

Other publications

These publications are also relevant as further information sources:

- *RMF User's Guide,* SC33-7990
- *RMF Messages and Codes,* SC33-7993
- *RMF Programmer's Guide*, SC33-7994
- *RMF Report Analysis*, SC33-7991
- *RMF Performance Management Guide*, SC33-7992
- *z/OS MVS Initialization and Tuning Reference*, SA22-7592
- *z/OS MVS System Commands*, SA22-7627
- *z/OS MVS System Management Facilities (SMF)*, SA22-7630
- *z/OS SDSF Operation and Customization,* SA22-7670
- Michael Teuffel and Robert Vaupel*, Das Betriebssystem z/OS und die zSeries,* publisher Oldenbourg Verlag, Munich, Germany, 2004

Online resources

These Web sites and URLs are also relevant as further information sources:

- RMF Home page

  `http://www.ibm.com/servers/eserver/zseries/zos/rmf/`
- LINPACK Benchmark description and information

  `http://www.linpack.com/`

- zAAP Projection Tool Homepage

  http://www.ibm.com/servers/eserver/zseries/zaap/gettingstarted/
- Service Units definitions - SUs

  http://www.ibm.com/servers/eservers/zseries/srm/
- Sysplex Health Checker program

  http://www.ibm.com/servers/eserver/zseries/zos/downloads/
- LSPR product document with workload mix description

  http://www.ibm.com/servers/eserver/zseries/lspr/lsprmixwork.html
- WSC document with LSPR information and presentation

  http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS135
- Large System Performance Reference

  http://www.ibm.com/servers/eserver/zseries/lspr/zSerieszOS.html/

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

IBM

Redbooks

Effective zSeries Performance Monitoring Using Resource Measurement Facility

(0.5” spine)
0.475”<->0.873”
250 <-> 459 pages

Effective zSeries Performance Monitoring Using Resource Measurement Facility

Setting up and customizing RMF components

Understanding the new features and reports

Using RMF to monitor performance in real-world scenarios

This IBM Redbook provides a detailed look at Resource Measurement Facility (RMF), the IBM product designed to simplify management of single and multiple system workloads. RMF gathers data and creates reports that help system programmers and administrators to tune their system optimally, react quickly to system delays, and diagnose and remediate performance problems.

This redbook describes RMF functionality with special emphasis on the newest features, and also presents a review of the older, established components. Detailed instructions for setting up and customizing the components are provided. New features introduced are Spreadsheet Reporter, Distributed Data Server, Linux data gatherer, and Performance Monitor.

A high-level overview of performance analysis concepts is presented, along with a detailed discussion of performance metrics. This information is the foundation for an in-depth look at how RMF can be used to manage systems in the real world. Practical scenarios demonstrate how to use RMF to conduct overall performance evaluations and monitor batch and transactional workloads. The new reporting capabilities are illustrated with numerous examples, in particular those that support the latest workload licensing model, zAAP, UNIX System Services, WebSphere Application Server, and Linux.

SG24-6645-00 ISBN 0738492353