IBM®

TECHNICAL BULLETIN

ROCHESTER INTERMEDIATE SYSTEMS CENTER

System/36 Performance Monitoring and Tuning

G360-1006-0
July 1985

IBM®

TECHNICAL BULLETIN

ROCHESTER INTERMEDIATE SYSTEMS CENTER

System/36 Performance Monitoring and Tuning

by: Al Brown

G360-1006-0
July 1985

First Edition (July 1985)

Changes may be periodically made to the information herein; any such changes will be reported in subsequent editions or Technical Newsletters.

Table of Contents

1.0 PREFACE

The subject of this technical bulletin is performance evaluation: the process of tuning your system:

- To meet your performance expectations

- To optimize use of the system resources

There is no "cookbook" method to do a performance evaluation that will automatically provide you with specific performance solutions to apply to your system; the responsibility of identifying performance improvements for your system rests with you, the user. As a result, do not expect to find sure-fire performance solutions outlined in this publication; there are no easy answers. Rather, we attempt to provide necessary information for you to evaluate the performance of your system in a disciplined way, with some degree of confidence that the evaluation will succeed in identifying your system's problem areas and specific solutions for your particular installation.

2.0 INTRODUCTION

As observed in many existing systems, problems such as the following frequently hamper performance evaluations:

- Users have not always clearly defined what they expect from the system. Unsatisfactory performance is described vaguely, and tuning efforts are not based on a specific problem description.

- Installations often apply "corrective" actions to the system based on the reputation of those actions rather than in response to a problem area identified in the system.

- Faced with a morass of measurements of the system, users often don't know which measurements to focus on or how to judge the reported values of the measurements on which they do focus.

Such problems can be minimized by approaching the performance evaluation in a disciplined way.

A disciplined performance evaluation should include these steps:

1. Set performance objectives

   Define the work the system is expected to do for your installation. At the same time, define the performance objectives and the factors affecting performance.

2. Identify and, if necessary, obtain or develop required measurement tools

   The goal of the performance evaluation is to make the most effective use of the system resources in order to meet the installation's performance objectives. Because a measurement tool is the basic means to determine the use of system resources--and to determine if you are meeting your objectives--any performance evaluation is limited by the measurement tool available.

3. Evaluate performance-related factors when setting up the system

   These factors include configuration and initialization parameters that have performance implications and basic performance factors that should be evaluated when you initially install SSP.

4. Measure against the performance objectives

Once the system is meeting its availability and reliability requirements, measure against the performance objectives established in step 1. If objectives are met, skip to step 8. Otherwise, continue with step 5.

5. Focus on areas of probable significant performance improvement

When performance objectives are not met, focus on areas of probable significant performance improvement.

6. Identify and implement corrective actions

Identify and implement corrective actions for the areas of probable improvement.

7. Evaluate the results of the actions implemented

Remeasure against the objectives. Often, one bottleneck will hide other bottlenecks in the system. If performance objectives are still not met, return to step 5.

8. Monitor performance

Once the objectives are met, monitor performance because of the changing nature of any system. Workload management and I/O resource management are key areas that require continual monitoring. By monitoring the system and its use of major resources, you can address performance problems before they become critical.

This technical bulletin gives a method for monitoring the performance of System/36, using the System Measurement Facility (SMF). By following the method suggested in this document, the personnel in an installation responsible for performance monitoring and tuning and capacity planning should be able to:

- Check if the system meets the currently specified throughput/load and response time requirements,
- Detect trends that could produce poor performance,
- Determine the cause of poor performance if it arises,
- Have a good basis for capacity planning.

Other Technical Bulletins ...

The following System/36 Rochester Technical Bulletins (newsletters) are available:

- System/36 8809 Tape Support, G360-1005.
- System/36 Query/36 Design Guide, G360-1007.
- System/36 Data Dictionary System Design Guide, G360-1008.
- System/36 Internals, G360-1009.
- System/36 Advanced Disk Data Management, G360-1010.

3.0 PERFORMANCE MONITORING AND TUNING OVERVIEW

General Concepts

Performance monitoring and tuning means keeping track of a system's operation so that its performance characteristics are understood.

The suggested method consists of:

- Calculating the values of performance variables
- Detecting when you are in a dangerous zone
- Keeping a history of performance to allow the trends in the system to be followed

In this section, we differentiate between two steps:

- Performance control: a quick snapshot of system performance.

  "Control values" are used for performance control. These values can be classified into two groups:

  - User-oriented (load, response time, etc.)
  - System-oriented (processors and disk utilizations, swapping, etc.)

- Performance analysis: an in-depth view of the performance indicators.

  "Utilizations and counters," which give the profile of the job/program/task, down to the transaction/function level, are used for performance analysis.

Measured Interval

In this suggested methodology, you should take the "peak hour" as the basis of time to do the performance monitoring.

This necessitates:

- Determining the period of greatest load, by observation on several days, with a duration of about an hour.
- Periodically assessing that this interval is still representative.
- All the numerical reports from the several system measurements will refer to this interval, explicitly specified by the START and STOP times.

So, you may graph the output from the performance tool for the complete day, but the analysis must refer to the same period each day to allow trends to be followed.

3.1 PERFORMANCE CONTROL

Objective

To verify that:

- The throughput/load and response time specifications are being satisfied for the overall system, or for any particular group of transactions.
- The system is stable (that is, no wide variation in performance from day to day).
- The control values don't reach the warning or action levels (thresholds).

Performance control is useful also:

- To determine workload trends.
- As input to capacity planning. You might look at an SMF report of your job cycle to make a determination if you have enough resource to:
  - Add additional applications.
  - Verify the best time of day to schedule new jobs.

Frequency

- Initially: Daily, to establish base values.
- Normally: Once per week or month, depending on current threshold level.

  This will give you a feel for how your system reacts or what it look like when response time and job throughput are normal. Then if something does change and you really need SMF (System Measurement Facility) output, you will have previous reports that you can compare with reports after the change occurred to find out what happened.
- During performance problems: daily.

- After an important change or new application: Daily, to establish new base values.

The system load grows. When the load on a system changes due to increases in the volume of transactions, the performance tool monitor should be run to get performance data showing the current loading on the system. This data can be used to help the user make projections as to the capability of the present configuration to handle the additional volume.

You anticipate changes to applications. If there are plans to make functional changes to one or more of the applications on a system, the impact of the changes needs to be understood. To evaluate the impact of the changes, performance data should be collected and used to project the impact of the changes.

Users are complaining about performance. Performance analysis should be performed if users are complaining that the response time is less than what they need or that the total system throughput is less. Using SMF data can assist the user in getting an insight into where the problems lie (processors, disk, or memory) and validate the concerns of the users.

An application backlog exists. If there are additional applications that are to be installed, there will be questions as to the potential impact on current performance of the new applications. Measuring the current applications and, separately, measuring the new applications in a prototype environment will provide data as to the performance impact on the existing applications as well as an idea of the performance of the new applications, both with the current workload as well as future workload.

You may want to run it while you are testing new applications or run schedules to see the effect they are going to have on the overall system performance.

Costs are becoming an issue. The output of SMF can be used to assist the user to determine what, if any, additional hardware or programming changes should be made to increase performance or throughput. SMF can assist the user in determining the effect of changes to hardware as well as to programs. Using this data, decisions can be made as to the best course of action to take to improve performance at the least cost, both in time and money.

You install a new release of the software. Collect data before a release is installed.

3.2 PERFORMANCE ANALYSIS

Objective

- If there are performance problems:

  - Identify which analysis values have reached the alarm levels (that is, warning or action levels).

  - Determine actions to correct those values. (Usually this means studing the hardware configuration, software options, or file distribution.)

- If there are no performance problems:

  - Identify the "typical" analysis values for this installation.

  - Assess whether the values are stable or a trend is indicated.

  The performance analysis data is useful also for models and capacity planning.

Frequency

- Daily, if there are performance problems

- Periodically, if there are no performance problems.

3.3 PERFORMANCE TUNING

What is performance tuning?

Tuning involves:

Changing work load characteristics in an attempt to improve performance without adding hardware.

Why performance tuning?

To meet batch and interactive throughput requirements.

When should you do performance tuning?

Whenever the operating environment changes.

The use of System/36 resources varies from one installation to another and even within one installation. Because of this, SSP provides you with commands to create and change resources and resource values that affect the responsiveness and throughput of your system. For example, you can change the task work area size, change print spool file sizes, and set task execution priorities.

Performance tuning offers suggestions only, no guarantees or warranties are expressed or implied. Tuning should be systematic and tightly controlled and, where possible, change should be made to a single system value or attribute. You should document your tuning process for future tuning needs.

Tuning your system to meet your needs can include the following:

- Evaluating job performance, which includes evaluating the internal activities of the system and the external behavior of a job.
- Examining the programming techniques used in your application programs and managing system resources.

Note:

Before you begin tuning your system, you should be familiar with main storage task management and disk data management concepts and functions.

3.3.1 BASIC TUNING STEPS

```
Step 1.   Initial Allocations:

          .    Memory
          .    Job parameters
          .    System values


Step 2.   Measurement:                      <---------------
                                            -----------      |
          .    Workload environments                   |     |  These
                                                       |     |  steps
                                                       |     |   are
                                                       |     |  repeated
Step 3.   Observation/evaluation/analysis: <-------    |     |   until
                                              --------       |performance
          .    User feedback                           |     |satisfies
          .    Work backlog                            |     |   the
          .    SMF reports                             |     |requirements
                                                       |     |
Step 4.   Adjustments: <---------------------------          |
                        --------------------------------------

          .    Tuning knobs
          .    Scheduling
          .    Education
          .    Application/program changes
          .    Hardware modification
```

3.3.2 PERFORMANCE TUNING KNOBS

Resource variables you may consider changing to improve performance include:

- Memory:
  - Buffering and blocking factors
  - Program sizes
- Disk:
  - File and library placement
- Job parameters:
  - Work scheduling
  - Execution Priority
- System values:
  - Task work area size
  - Print spool size
  - Print buffer size
- Application/program characteristics:
  - Logic flow and controls
  - Programming techniques
  - Display screen techniques

Each resource variable should be reviewed and considered in the performance analysis.

If tuning does not help, your next step is an upgrade analysis in which you change the hardware configuration:

- Disk size (that is, the number of disk arms)
- Memory size
- Optional processors:
  - DSC
  - Stage 2 processors (CSP on model 5360 only)
  - MLCA (for more communications lines)
- Model change, depending on current model installed:
  - From a model 5364 to a model 5362 to a model 5360

4.0 DEFINING PERFORMANCE OBJECTIVES

The definition of performance objectives has two goals:

- To state what is expected of the system in specific terms for each category of work (for example, trivial versus nontrivial transactions) at each distinct period of time (for example, prime shifts versus off-shifts and peak periods within each shift).

- To understand and document the resources required to meet the objectives defined as the first goal.

From the nature of these two goals, the definition of performance objectives is iterative. You should expect to update your performance objectives as the workload changes, as better understanding of resource requirements is gained, as the resource requirements of the work change, and as turnaround and response time requirements change. Detailed performance objectives, however, will make such changes noticeable and will help identify solutions to performance problems that arise because of the changing nature of the workload. The definition of performance objectives is not a trivial task, but it is essential to a disciplined performance analysis and to plan for new applications and additional work.

To define performance objectives, follow these steps:

1. Define the terms in which to specify objectives.

2. Determine how the objectives will be measured. Note discrepancies between the measurements and what the user sees.

3. Document the current workload--amount and categories of work. For example:

   - Interactive -- trivial and nontrivial transactions

   - Batch -- job classes or groups

   Examine existing workload categories for their effectiveness in distinguishing work that requires different priorities and different objectives.

4. Set objectives for each category of work.

5. Measure and document resources used by each category of work.

   a. Correct any ineffective classifications of work, based on resources used.

   b. Determine controls to enforce resource usage rules for the different categories.

   c. Review the reasonableness of the objectives in light of resource requirements, and set capacity limits for each category of work.

6. Measure against the objectives.

   a. If measured objectives meet defined objectives, monitor performance.

   b. If measured objectives do not meet defined objectives, analyze the system to identify problem areas.

The following sections describe each step in detail.

4.1 STEPS 1 & 2: SELECTING AND MEASURING OBJECTIVES

The first step in defining performance objectives is to choose the terms in which you will specify objectives. There are two basic types of performance objectives:

- User-oriented, which reflect the way an end user would rate the services provided by the system.

- System-oriented, which reflect the workload that must be supported on a system level.

User-oriented objectives include response time for interactive work and turnaround time for batch work. System-oriented objectives include batch throughput, interactive transaction rate, and the number of concurrent interactive users.

The distinction between system-oriented and user-oriented objectives is not merely academic. Achieving optimal system-oriented objectives (that is, getting as much work through the system as possible) implies achieving the highest use of system resources--processor, main storage, channels, control units, communications lines and devices. Achieving optimal user-oriented objectives implies having the availability of any resource when it is required. Ensuring the proper workload mix in terms of required resources can help avoid conflicts between meeting user-oriented and system-oriented objectives.

In addition, to ensure that you have the proper workload mix of required resources, let unmet user-oriented and system-oriented performance objectives direct the initial focus of a tuning effort. When investigating user-oriented performance problems, you must first determine to what resource(s) the work is being denied access and try to increase its access to that resource(s). The initial focus is on task management--favoring access of one type of work over other types of work to needed resources. When system-oriented problems occur, the focus is on resource management--identifying the critical resource(s) in the system and increasing the effective use of the resource(s).

When choosing the terms in which to define your objectives, you must also determine how the objectives will be measured and reported. For user-oriented objectives, you must note any differences between the measured objectives and what the user sees. Times reported by measurement tools are usually system elapsed times and do not include delays such as job output distribution and polling delays at a terminal.

4.2 STEP 3: DOCUMENTING THE WORKLOAD

After deciding on and defining the terms in which to measure performance, the next step in writing performance objectives is to understand and document the current workload on the system. This includes the amount and the categories of work:

- Priority of the work
- Different periods of time during which objectives and priorities vary for the same work
- Resource requirements of the work
- Types of users requiring different objectives
- Ability to track and report on work according to the installation's needs, such as breakdowns by department

Start with the categories of work that were defined under the prior system: batch job classes, trivial and nontrivial interactive transaction types, and so on. Usually, however, these categories were not fully defined. When initially defining the categories, the resource requirements will probably reflect expected resource usage; measuring actual resource usage on the system is described in more detail in step 5.

Once the categories are defined, review them for apparent errors. The purpose of the different categories is to distinguish work according to different resource requirements, different objectives that must be met, different priorities, and so on. For example, all jobs submitted from similar development groups in different locations are expected to receive the same turnaround time. However, because of distribution of the completed work to different locations, and possible time differences in actually returning output to the submitters, you might want to further separate this work--to give priority, for example, to jobs that must be distributed to locations in different time zones, where delays in turnaround time can have a significant effect on the users.

The importance of fully documenting the workload cannot be overemphasized. Some of the most significant performance gains to be achieved in the system are accomplished by means of <u>workload management</u>. The ability to manage the workload is directly proportional to having detailed knowledge of the workload.

4.3 STEP 4: SETTING THE OBJECTIVES

Once the workload is categorized, set objectives for each category of work and document and summarize your performance objectives.

Many installations state objectives for a percentage of the transactions in a class--for example, 90% of the interactive transactions should receive a three-second response time; 85% of the jobs in class A should receive turnaround time of one hour. If you state objectives in these terms, also set an objective for the "leftover" percentage of transactions--in the preceding example, for the 10% of the interactive transactions and the 15% of the jobs in class A. For long-running transactions or jobs, you might want to specify the objectives in terms of service units or processor time received per second. Ensure that objectives are set for 100% of the work in your system.

Note: When setting user-oriented objectives, be sure you take into account any time the user sees that is not reflected in the measurement of the objective. For example, if the interactive trivial transactions require a four-second response time, you might set the objective to three seconds to account for polling delays not reflected in SMF measurements of response time.

4.4 STEP 5: MEASURING RESOURCE REQUIREMENTS

Once the system is configured and is close to your reliability and availability objectives, measure the resources actually being used by the different categories of work. To do this, you must choose the means by which you will measure resource consumption (for example, service units, seconds, number of events such as the number of I/Os, and so on) and the tools by which the resources will be measured (that is, SMF and user-written programs). Essentially, you want to identify the amounts of processor, main storage, and I/O resources required for each category of work.

By assigning each distinct category of work to a separate performance group, you can obtain data on the processor, main storage, and I/O operations consumed by each category from the SMF workload activity report.

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

- Batch jobs whose resource consumption places them in the top ten percent of their class in terms of resource use might require reclassification.

- If the resource data varies widely for a particular job class, that is, there is no distinct pattern--that job class might require redefinition or a tolerant performance objective.

The resource data collected will further define the categories of work; from this data, you can set resource limits for each category.

5.0 PERFORMANCE ANALYSIS WITH SYSTEM MEASUREMENT FACILITY (SMF)

SMF is part of the System Support Program (SSP). It provides snapshots of system resource usage while running applications. It can be used to evaluate how efficiently system resources are being utilized. This could include:

- Optimizing present performance by determining if you need to allocate system resources differently
- Evaluating the effect of newly added applications on resources
- Determining efficient application loading and scheduling
- Determining the cause of a performance problem
- Judging the need for additional main storage and disk space

By giving the user access to specific information about system resource usage while he is running his applications, he is able to make intelligent decisions on changes which could be made to increase performance.

To interpret the SMF results optimally, it is necessary to know how S/36 works internally, especially the main storage data management (MSDM) and the disk data management (disk DM) functions. You can find detailed information about S/36 MSDM and disk DM in the System/36 Concepts and Programmer's Guide (SC21-9019 Chapters 8 & 13).

SMF can also be used as an education tool to provide:

- The ability to understand the behavior of a system, such as where can you expect throughput and/or response time to begin falling off, and what is the corresponding effect on batch throughput?
- Information on whether there is an imbalance in the use of processors, disk, or memory, thereby allowing corrections to be made by adding additional hardware capacity or changing the operating environment.
- The ability to set realistic expectations for system performance based on the current operational environment and the system configuration.

Use SMF to provide estimates of the effect of application programming and tuning changes on interactive throughput and response times.

The tool can be used to assist in determining what changes to a system will be the most beneficial to improve system performance and to assist you in making decisions regarding trade-offs between programming and hardware changes.

Note (System/34 users only):

The IBM System/36 hardware is all new; the majority of the software is new or has been rewritten or revised. The noteworthy software exception is the user interface which remained the same as the System/34 interface. Because of this difference, System/34 SMF values should not be compared to System/36 SMF values. Most of the same usages (utilizations) and counters still report the same things as they did on the System/34, but since the hardware is different they are not comparable.

SMF actually performs three functions:

- Data collection
- Communications data collection
- Report writing

These functions:

- Are started by the SMF, SMFSTART, SMFSTOP, and SMFPRINT procedures and can be called by a user-written procedure. This enables you to put the SMF procedures in a job stream.
- Help you analyze the use of system resources.

The SMF provides system information, counters and utilization data for the processors, disk, diskette, display stations, printers, communications lines, active tasks, and main storage.

SMF samples various system resource usage indicators at specified intervals while user programs are running. The report program lists configuration data (that is, the system configuration at the time SMF was run), statistical data, event counter data, and summary data.

5.1 SMF STORAGE REQUIREMENTS

SMF Storage Requirements:

. Data collection program without communications and/or data by task	6K
. Data collection program with communications and/or data by task	8K
. Report writer program	48K

6.0 COLLECTING PERFORMANCE ANALYSIS DATA

The measurement and collection of performance data is easy to do and can be done at an acceptable system overhead cost (generally 2-5% which is well within industry standards).

The data collected is an audit trail of performance and must be reduced and organized into meaningful reports. A key step is to inspect the data to make sure that it represents typical application work. SMF provides programs that allow the data to be reduced in various ways:

- In summary, mini, detail, or all reports
- Over specified time periods/intervals
- By system, job, and program
- Right down to the interactive transaction/function level (that is, the interactive transaction/function was the only activity entered during the measured time interval for a task)

You must understand the characteristics of the workload and of the transactions. This will mean studying the exceptional periods of performance, both good and bad, and tracking the key performance indicators.

You should collect performance data over a period of several days of operation in which there are various interactive and batch workloads. Gather data during periods of moderate workload, in which the system memory is not overcommitted. In addition, performance data should be gathered during other periods in which the workload is heavy.

The point of obtaining measurements for a variety of workload conditions is to observe the changes in the swapping characteristics of the system. In particular, it is important to see the differences between the moderate and heavy workloads in terms of:

- MSP and CSP utilizations

- Disk utilization

- Physical disk I/O per second requests

- Swap-in vs swap-out characteristics

- Translated calls to translated loads ratio

The value of SMF is greatest if it represents your complete work cycle. It can also be of great value if used during your peak workload when the system is being taxed the most. Many problems will show up at this time. If you are trying to determine if you have enough disk or memory on your system, this tool can help you check.

When collecting performance data, take into account the nontypical situations that might skew the data. Nontypical situations can include:

- Programmers. They can be a very disruptive force on the system. Not so much when they do SEU type work, but more when they build a test environment and then start testing the programs. Check to see if their portion of the entire workload looks reasonable.

  Consistently high usage by programmers may be justification for a machine just for programmers or, at the least, better scheduling of programmer activity.

- System operators. See how much resource they used.

- A line being down while taking measurements or for an unusual batch job that was run during the measurement period.

- Peak periods and periods of good response. Study both types of conditions to understand the differences in swapping and waiting characteristics for transactions.

- The application mix. Is it representative of the typical environment?

- The number of active users. Who does most of the transactions and when? Is this typical?

- Batch priority. What priority does batch run under? If equal to or greater than interactive work, then an adjustment may need to be made to reflect the impact that batch work will have on competing for system resources. This could be done by converting that portion of batch work running at a higher priority to low priority.

7.0 TYPE OF SMF OUTPUT

The intended use of SMF is to provide the user with information that will allow him to predict or have a general idea of:

- MSP and CSP, disk, communications, and memory utilizations
- The effect on system performance of changes to programming or hardware

The data provided by SMF helps the user to correlate throughput and response time to:

- Counters (count of items), and
- Device (resource) utilizations

The utilization measurements are computed by various methods. They are not intended to be exact values of device or resource utilization, but they can be used to identify trends in system usage.

Response time deals with interactive transactions. It is a measurement of the amount of elapsed time from the depression of an Enter key (or command key or Roll key) until an answer is displayed back at the work station. Consider it a measurement of end user satisfaction and of end user productivity potential. The shorter the response time, the more productive the end user could be.

Throughput is expressed in terms of transactions per hour and can be thought of as a business-oriented measurement. In other words, X number of transactions per hour must be accomplished or the business suffers. Sometimes the unit of measure itself is the major problem. This is because business volumes are expressed in different terms, such as orders per day or receipts per day, and thus must be converted into transactions per hour. In addition, not all applications are designed or implemented the same, thus one order could mean one transaction under one scheme and many transactions in another.

Not all transactions are the same. To be more specific about a transaction--in terms of characteristics that will allow you to do capacity planning--you need to know how much MSP/CSP and how many physical disk I/O requests will be required, on the average, per transaction.

After SMF has measured a specified time interval, you can specify the type of output to be produced by the report writer program in the form of either or both of the following:

- A report
- A disk file containing 80-byte summary records

7.1 REPORTS

Four types of reports can be produced:

- SUMMARY
- MINI
- DETAIL
- ALL

DETAIL is the default for the report option.

You should print only the information needed:

- Use the SUMMARY report to find the times of maximum/minimum values.
- Use the MINI report for limited utilization rates, SEC, and I/O counts for desired subinterval.
- Use the DETAIL report for utilization rates and task status for desired subinterval.
- Use the ALL report for complete information about specific periods and for communications data, data by task, and utilizations.

Printed output is divided into four sections. Each section contains an SMF report. Because each report has some duplication, only the parts that are different are included in each of the following sections.

7.1.1 SMF SUMMARY REPORT

The following table shows the types of information that are listed on the SUMMARY report:

SUMMARY Printed Report	IPL configuration information
	Summary information

This section contains a complete SUMMARY report. The information in the SUMMARY report is printed for each report and will not be included in these materials.

SMF SUMMARY Report (Part 1 of 6)

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 1

I P L C O N F I G U R A T I O N

MAIN STORAGE SIZE	1024 K	SMF DATA COLLECTION DATE.	85/02/07
DISK CAPACITY	799.14 MB	RELEASE/MODIFICATION LEVEL. . . .	03/00
TASK WORK AREA SIZE	1100 BLOCKS	CONFIGURATION MEMBER NAME	S10CNFG
3262 PRINTER SUPPORTED. . . .	Y	COMMUNICATION LINES SUPPORTED . .	1,2,3,4,5,6,7,8
SPOOLING SUPPORTED.	Y	AUTOCALL LINES SUPPORTED.	NONE
REMOTE WORKSTATIONS SUPPORTED	Y	X.21 LINES SUPPORTED.	NONE
COMM CONTROLLER ATTACHED. . .	Y	X.25 LINES SUPPORTED.	NONE
DSC ATTACHED.	Y	TAPE DRIVES SUPPORTED	2
WSC ATTACHED.	Y	DISK DRIVES ATTACHED.	4
		SYSTEM MODEL NUMBER	53602

SMF SUMMARY Report (Part 2 of 6)

REPORT DATE - 2/07/85   DATA COLLECTION FILE - SMF.RAY.   SYSTEM/36 MEASUREMENT FACILITY   PAGE 2

---------- SMF SUMMARY ----------

START TIME	14.13.23.637
STOP TIME	14.14.23.742
ELAPSED TIME	00.01.00.105
SNAPSHOT INTERVAL	1.00.000

-------------------- S U M M A R Y   U S A G E --------------------

	AVERAGE	MAXIMUM	TIME MAXIMUM OCCURRED
MAIN STORAGE PROCESSOR. . . .	85 %	85 %	14.14.23.742
CONTROL STORAGE PROCESSOR . .	26 %	26 %	14.14.23.742
WORKSTATION CONTROLLER QUEUE.	4 %	4 %	14.14.23.742
WORKSTATION CONTROLLER. . . .	4 %	4 %	14.14.23.742
DATA STORAGE CONTROLLER . . .	6 %	6 %	14.14.23.742
DATA STORAGE ATTACHMENT . . .	23 %	23 %	14.14.23.742
DISK 1.	41 %	41 %	14.14.23.742
DISK 2.	2 %	2 %	14.14.23.742
DISK 3.	4 %	4 %	14.14.23.742
DISK 4.	4 %	4 %	14.14.23.742
COMMUNICATION LINE 1.	** %	** %	NOT ACTIVE/COLLECTED
COMMUNICATION LINE 2.	** %	** %	NOT ACTIVE/COLLECTED
COMMUNICATION LINE 3.	** %	** %	NOT ACTIVE/COLLECTED
COMMUNICATION LINE 4.	** %	** %	NOT ACTIVE/COLLECTED
COMMUNICATION LINE 5.	** %	** %	NOT ACTIVE/COLLECTED
COMMUNICATION LINE 6.	** %	** %	NOT ACTIVE/COLLECTED
COMMUNICATION LINE 7.	** %	** %	NOT ACTIVE/COLLECTED
COMMUNICATION LINE 8.	** %	** %	NOT ACTIVE/COLLECTED
TASK WORK AREA.	68 %	68 %	14.14.23.742
ASSIGN/FREE SPACE	80 %	80 %	14.14.23.742
TOTAL STORAGE COMMITMENT. . .	135 %	135 %	14.14.23.742
ACTIVE STORAGE COMMITMENT . .	96 %	96 %	14.14.23.742
ACTUAL STORAGE COMMITMENT . .	65 %	65 %	14.14.23.742

------------ S U M M A R Y   S Y S T E M   E V E N T   C O U N T E R S ------------

	TOTAL	PER MINUTE	MAXIMUM	TIME MAXIMUM OCCURRED
MAIN STORAGE TRANSIENT CALLS. .	10	10.0	10	14.14.23.742

SMF SUMMARY Report (Part 3 of 6)

TRANSLATED TRANSFER CALLS . . .	331	331.0	331	14.14.23.742
ASYNCHRONOUS TRANSFER CALLS . .	2	2.0	2	14.14.23.742
MAIN STORAGE TRANSIENT LOADS. .	5	5.0	5	14.14.23.742
TRANSLATED TRANSFER LOADS . . .	11	11.0	11	14.14.23.742
MAIN STORAGE LOADER REQUESTS. .	51	51.0	51	14.14.23.742
SWAPS IN.	5	5.0	5	14.14.23.742
SWAPS OUT	0	0.0	0	00.00.00.000
SWAPS OUT, FORCED	0	0.0	0	00.00.00.000
TASK WORK AREA READ OPS	18	18.0	18	14.14.23.742
TASK WORK AREA WRITE OPS. . . .	11	11.0	11	14.14.23.742
MAIN STORAGE CLEAR OPS.	36	36.0	36	14.14.23.742
CONTROL STORAGE TRANSIENT CALLS	37	37.0	37	14.14.23.742
CONTROL STORAGE TRANSIENT LOADS	0	0.0	0	00.00.00.000
CONTROL STORAGE LOADER REQUESTS	0	0.0	0	00.00.00.000
SPOOL SEGMENTS ALLOCATED. . . .	4	4.0	4	14.14.23.742
SPOOL ENTRIES ALLOCATED	0	0.0	0	00.00.00.000
SPOOL EXTENTS ALLOCATED	0	0.0	0	00.00.00.000
GENERAL WAITS	0	0.0	0	00.00.00.000
DISK RECORD WAITS	0	0.0	0	00.00.00.000
TASK WORK AREA EXTENTS.	0	0.0	0	00.00.00.000
JOB INITIATIONS	2	2.0	2	14.14.23.742
JOB STEP INITIATIONS.	4	4.0	4	14.14.23.742
MRT ATTACHES.	0	0.0	0	00.00.00.000
MRT LOADS	0	0.0	0	00.00.00.000
JOB TERMINATIONS.	1	1.0	1	14.14.23.742
JOB STEP TERMINATIONS	2	2.0	2	14.14.23.742
ABNORMAL TERMINATIONS	0	0.0	0	00.00.00.000
DISK LOCKS SATISFIED.	2	2.0	2	14.14.23.742
DISK LOCKS EXPIRED.	0	0.0	0	00.00.00.000
ASSIGN/FREE EXTENSIONS.	0	0.0	0	00.00.00.000
ASSIGN/FREE REDUCTIONS.	0	0.0	0	00.00.00.000
PREEMPTIVE TASK DISPATCHES. . .	2013	2013.0	2013	14.14.23.742
RESOURCE TIMEOUTS	157	157.0	157	14.14.23.742
MAIN STORAGE PROCESSOR TIMEOUTS	222	222.0	222	14.14.23.742
WKSTN BUFFER READ RETRIES . . .	1	1.0	1	14.14.23.742
L-1 STORAGE RELEASES W/O SWAP .	14	14.0	14	14.14.23.742
L-1 STORAGE RELEASES W/ SWAP. .	0	0.0	0	00.00.00.000

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 4

L-2 STORAGE RELEASES W/O SWAP .	0	0.0	0	00.00.00.000
L-2 STORAGE RELEASES W/ SWAP. .	0	0.0	0	00.00.00.000
L-3 STORAGE RELEASES W/O SWAP .	0	0.0	0	00.00.00.000
L-3 STORAGE RELEASES W/ SWAP. .	0	0.0	0	00.00.00.000
L-4 STORAGE RELEASES W/O SWAP .	0	0.0	0	00.00.00.000
L-4 STORAGE RELEASES W/ SWAP. .	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000
NOT USED.	0	0.0	0	00.00.00.000

S U M M A R Y I / O C O U N T E R S

	TOTAL	PER MINUTE	MAXIMUM	TIME MAXIMUM OCCURRED
DISK 1 READ OPS.	587	587.0	587	14.14.23.742
DISK 1 WRITE OPS	230	230.0	230	14.14.23.742
DISK 1 SCAN OPS.	26	26.0	26	14.14.23.742
DISK 1 SEEK OPS.	796	796.0	796	14.14.23.742
DISK 2 READ OPS.	16	16.0	16	14.14.23.742
DISK 2 WRITE OPS	44	44.0	44	14.14.23.742
DISK 2 SCAN OPS.	3	3.0	3	14.14.23.742
DISK 2 SEEK OPS.	26	26.0	26	14.14.23.742
DISK 3 READ OPS.	52	52.0	52	14.14.23.742
DISK 3 WRITE OPS	0	0.0	0	00.00.00.000
DISK 3 SCAN OPS.	49	49.0	49	14.14.23.742

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 5

DISK 3 SEEK OPS.	96	96.0	96	14.14.23.742
DISK 4 READ OPS.	117	117.0	117	14.14.23.742
DISK 4 WRITE OPS	20	20.0	20	14.14.23.742
DISK 4 SCAN OPS.	1	1.0	1	14.14.23.742
DISK 4 SEEK OPS.	5	5.0	5	14.14.23.742
DISKETTE 1 READ OPS.	0	0.0	0	00.00.00.000
DISKETTE 2D READ OPS	0	0.0	0	00.00.00.000
DISKETTE 1 WRITE OPS	0	0.0	0	00.00.00.000
DISKETTE 2D WRITE OPS.	0	0.0	0	00.00.00.000
DISKETTE SEEK OPS.	0	0.0	0	00.00.00.000
72MD AUTO LOADER REQUESTS. . .	9	9.0	9	14.14.23.742
DISKETTE HEAD CONTACT REVS . .	0	0.0	0	00.00.00.000
LOCAL DISPLAY STATION OPS. . .	27	27.0	27	14.14.23.742
LOCAL PRINTER OPS.	0	0.0	0	00.00.00.000
REMOTE DISPLAY STATION OPS . .	0	0.0	0	00.00.00.000
REMOTE PRINTER OPS	0	0.0	0	00.00.00.000
3262 PRINTER OPS	0	0.0	0	00.00.00.000
1255 MICR OPS.	0	0.0	0	00.00.00.000
TAPE 1 READ BYTES.	0 K	0.0 K	0 K	00.00.00.000
TAPE 1 WRITE BYTES	0 K	0.0 K	0 K	00.00.00.000
TAPE 1 REWIND OPS.	0	0.0	0	00.00.00.000
TAPE 1 HITCHBACK OPS	0	0.0	0	00.00.00.000
TAPE 2 READ BYTES.	0 K	0.0 K	0 K	00.00.00.000
TAPE 2 WRITE BYTES	0 K	0.0 K	0 K	00.00.00.000
TAPE 2 REWIND OPS.	0	0.0	0	00.00.00.000
TAPE 2 HITCHBACK OPS	0	0.0	0	00.00.00.000
DISK 1 SEEK OPS GT 1/3 DISK. .	6.4 %	****	6.4 %	14.14.23.742
DISK 2 SEEK OPS GT 1/3 DISK. .	0.0 %	****	0.0 %	00.00.00.000
DISK 3 SEEK OPS GT 1/3 DISK. .	0.0 %	****	0.0 %	00.00.00.000
DISK 4 SEEK OPS GT 1/3 DISK. .	0.0 %	****	0.0 %	00.00.00.000
DISK 1 AVERAGE SEEK LENGTH . .	45 CYL	****	45 CYL	14.14.23.742
DISK 2 AVERAGE SEEK LENGTH . .	3 CYL	****	3 CYL	14.14.23.742
DISK 3 AVERAGE SEEK LENGTH . .	2 CYL	****	2 CYL	14.14.23.742
DISK 4 AVERAGE SEEK LENGTH . .	58 CYL	****	58 CYL	14.14.23.742

------------ S U M M A R Y D S A A N D T A P E U S A G E ------------

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 6

	AVERAGE	MAXIMUM	TIME MAXIMUM OCCURRED
DISK 1	37.5 %	37.5 %	14.14.23.742
DISK 2	2.0 %	2.0 %	14.14.23.742
DISK 3	3.6 %	3.6 %	14.14.23.742
DISK 4	3.4 %	3.4 %	14.14.23.742
DISKETTE TO MAIN STORAGE	0.0 %	0.0 %	00.00.00.000
DISKETTE TO DSC.	0.0 %	0.0 %	00.00.00.000
ONE BUFFER TO M/C STORAGE.	41.2 %	41.2 %	14.14.23.742
ONE BUFFER TO DSC.	0.4 %	0.4 %	14.14.23.742
TWO BUFFERS TO M/C STORAGE	2.4 %	2.4 %	14.14.23.742
TWO BUFFERS TO DSC	0.0 %	0.0 %	00.00.00.000
ONE BUFFER EACH TO M/C AND DSC . .	0.0 %	0.0 %	00.00.00.000
TAPE 1	0.0 %	0.0 %	00.00.00.000
TAPE 2	0.0 %	0.0 %	00.00.00.000
TAPE 1 DATA TRANSFER	0.0 %	0.0 %	00.00.00.000
TAPE 2 DATA TRANSFER	0.0 %	0.0 %	00.00.00.000
TAPE 1 START/STOP DATA TRANSFER. .	0.0 %	0.0 %	00.00.00.000
TAPE 2 START/STOP DATA TRANSFER. .	0.0 %	0.0 %	00.00.00.000

7.1.2 SMF MINI REPORT

The following table shows the types of information that are listed on the MINI report:

MINI Printed Report	IPL configuration information
	Communications configuration data, if active and selected
	. Statistics for each sample interval: - Device usage rates - Significant counters with the following exceptions: -- Disk record waits is not on a per task basis but the total for all tasks -- Data by file information is not reported - Number of active tasks
	Summary information

This section contains a SMF MINI report except for the SUMMARY information.

SMF MINI Report (Part 1 of 1)

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 1

I P L C O N F I G U R A T I O N

MAIN STORAGE SIZE	1024 K	SMF DATA COLLECTION DATE.	85/02/07
DISK CAPACITY	799.14 MB	RELEASE/MODIFICATION LEVEL. . . .	03/00
TASK WORK AREA SIZE	1100 BLOCKS	CONFIGURATION MEMBER NAME	S10CNFG
3262 PRINTER SUPPORTED. . . .	Y	COMMUNICATION LINES SUPPORTED . .	1,2,3,4,5,6,7,8
SPOOLING SUPPORTED.	Y	AUTOCALL LINES SUPPORTED.	NONE
REMOTE WORKSTATIONS SUPPORTED	Y	X.21 LINES SUPPORTED.	NONE
COMM CONTROLLER ATTACHED. . .	Y	X.25 LINES SUPPORTED.	NONE
DSC ATTACHED.	Y	TAPE DRIVES SUPPORTED	2
WSC ATTACHED.	Y	DISK DRIVES ATTACHED.	4
		SYSTEM MODEL NUMBER	53602

SNAPSHOT TIME - 14.14.23.742 SAMPLE INTERVAL - 1.00.105

DEVICE USAGE RATES

MAIN STORAGE PROCESSOR	85 %
CONTROL STORAGE PROCESSOR. . .	26 %
WORKSTATION CONTROLLER QUEUE .	4 %
WORKSTATION CONTROLLER	4 %
DATA STORAGE CONTROLLER. . . .	6 %
DATA STORAGE ATTACHMENT. . . .	23 %
DISK 1	41 %
DISK 2	2 %
DISK 3	4 %
DISK 4	4 %

NUMBER OF ACTIVE TASKS 23

COMMUNICATION LINE USAGE

COMMUNICATION LINE 1 . . .	** %
COMMUNICATION LINE 2 . . .	** %
COMMUNICATION LINE 3 . . .	** %
COMMUNICATION LINE 4 . . .	** %
COMMUNICATION LINE 5 . . .	** %
COMMUNICATION LINE 6 . . .	** %
COMMUNICATION LINE 7 . . .	** %
COMMUNICATION LINE 8 . . .	** %

TASK WORK AREA

TASK WORK AREA SIZE . . .	1510
TASK WORK AREA USAGE. . .	68 %
TASK WORK AREA EXTENTS. .	1

SYSTEM EVENT AND I/O COUNTERS

TRANSLATED TRANSFER CALLS . . .	331
TRANSLATED TRANSFER LOADS . . .	11
SWAPS IN.	5
SWAPS OUT	0
DISK RECORD WAITS	0
L-3 STORAGE RELEASES W/O SWAP .	0
L-3 STORAGE RELEASES W/ SWAP. .	0
L-4 STORAGE RELEASES W/O SWAP .	0
L-4 STORAGE RELEASES W/ SWAP. .	0
DISK 1 SEEK OPS GT 1/3 DISK .	6.4 %
DISK 2 SEEK OPS GT 1/3 DISK .	0.0 %
DISK 3 SEEK OPS GT 1/3 DISK .	0.0 %
DISK 4 SEEK OPS GT 1/3 DISK .	0.0 %

7.1.3 SMF DETAIL REPORT

The following table shows the types of information that are listed on the DETAIL report:

DETAIL Printed Report	IPL configuration information
	Communications configuration data, if active and selected
	. Statistics for each sample interval: - Device usage rates - Task work area usage - Task status - Storage totals
	Summary information

This section contains a SMF DETAIL report except for the SUMMARY information.

SMF DETAIL Report (Part 1 of 2)

```
REPORT DATE -  2/07/85     DATA COLLECTION FILE - SMF.RAY                SYSTEM/36 MEASUREMENT FACILITY                 PAGE    1

          ---------------------------- I P L   C O N F I G U R A T I O N ----------------------------
          MAIN STORAGE SIZE . . . . . .    1024 K        SMF DATA COLLECTION DATE. . . . .            85/02/07
          DISK CAPACITY . . . . . . . .  799.14 MB       RELEASE/MODIFICATION LEVEL. . . .               03/00
          TASK WORK AREA SIZE . . . . .    1100 BLOCKS   CONFIGURATION MEMBER NAME . . . .             S10CNFG
          3262 PRINTER SUPPORTED. . . .      Y           COMMUNICATION LINES SUPPORTED . .     1,2,3,4,5,6,7,8
          SPOOLING SUPPORTED. . . . . .      Y           AUTOCALL LINES SUPPORTED. . . . .                NONE
          REMOTE WORKSTATIONS SUPPORTED      Y           X.21 LINES SUPPORTED. . . . . . .                NONE
          COMM CONTROLLER ATTACHED. . .      Y           X.25 LINES SUPPORTED. . . . . . .                NONE
          DSC ATTACHED. . . . . . . . .      Y           TAPE DRIVES SUPPORTED . . . . . .                   2
          WSC ATTACHED. . . . . . . . .      Y           DISK DRIVES ATTACHED. . . . . . .                   4
                                                         SYSTEM MODEL NUMBER . . . . . . .               53602

SNAPSHOT TIME - 14.14.23.742  SAMPLE INTERVAL - 1.00.105

--------------- D E V I C E   U S A G E   R A T E S ---------------          ---- T A S K   W O R K   A R E A ----
MAIN STORAGE PROCESSOR . . . . 85 %    COMMUNICATION LINE 1. . . . . ** %    TASK WORK AREA SIZE. . . . . .  1510 BL
CONTROL STORAGE PROCESSOR. . . 26 %    COMMUNICATION LINE 2. . . . . ** %    TASK WORK AREA USAGE . . . . .    68 %
WORKSTATION CONTROLLER QUEUE .  4 %    COMMUNICATION LINE 3. . . . . ** %    TASK WORK AREA EXTENTS . . . .     1
WORKSTATION CONTROLLER . . . .  4 %    COMMUNICATION LINE 4. . . . . ** %
DATA STORAGE CONTROLLER. . . .  6 %    COMMUNICATION LINE 5. . . . . ** %
DATA STORAGE ATTACHMENT. . . . 23 %    COMMUNICATION LINE 6. . . . . ** %
DISK 1 . . . . . . . . . . . . 41 %    COMMUNICATION LINE 7. . . . . ** %
DISK 2 . . . . . . . . . . . .  2 %    COMMUNICATION LINE 8. . . . . ** %
DISK 3 . . . . . . . . . . . .  4 %
DISK 4 . . . . . . . . . . . .  4 %

-------------------------------------------- T A S K   S T A T U S --------------------------------------------
                                   PROG           REQ  WS                        ATTRIBUTE PROG  ----- STATUS ------        TWS
  JOB       PROCEDURE  PROGRAM     SIZE   TYPE    CNT  OPS PRIORITY USER ID  EXEC STOR SWAPS SWAP --WAIT--  SCHD  TWS  SWAPS

1                      SYS TASK     0 K  CMD-PR            SYSTEM            RENT NUC              NUC EC LW            0 K
2 W3141319  SMFSTART   $SMFML       8 K  SMF               SYSTEM RON        RELD SWAP             IN                   0 K
3                      SYS TASK    8*K  *SDLC-M            SYSTEM            RELD NSW              NSW EC               0 K
4                      SYS TASK     6 K *SDLC-I            SYSTEM            RELD SWAP             IN  EC               0 K
```

SMF DETAIL Report (Part 2 of 2)

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 2

5			SYS TASK	8*K	EM3270			SYSTEM		RELD	NSW	NSW	EC		0 K
6			SYS TASK	8 K	BSC3270			SYSTEM		RELD	SWAP	IN	EC		8 K
7			SYS TASK	24 K	APPC			SYSTEM		RELD	SWAP	IN	EC		0 K
8			SYS TASK	10 K	*SYSTEM			SYSTEM		REUS	REFR	OUT	EC LW		0 K
9			SYS TASK	18 K	PEER			SYSTEM		RELD	SWAP	IN	EC		0 K
10			SYS TASK	24 K	SNA-RJE			SYSTEM		RENT	REFR	IN	EC		2 K
11			SYS TASK	6 K	SNA3270			SYSTEM		RELD	SWAP	IN	EC		0 K
12			SYS TASK	14 K	*CSNA			SYSTEM		REUS	REFR	IN	EC		0 K
13			SYS TASK	14 K	RWS			SYSTEM		RELD	SWAP	IN	EC		0 K
14			SYS TASK	2 K	*SYSTEM			SYSTEM		RELD	SWAP	IN	EC		0 K
15	X1132407	SDA	‡SASF	32 K	SRT	1	7	V-MED	BSW	RELD	SWAP	IN			0 K
16	W8134016	INDEX	‡TETX2	46 K	SRT	1		MEDIUM	MARYJANE	RELD	SWAP	OUT	EC LW		0 K
17	X2131350	IDDUDISK	‡DSIN	8 K	SRT	1		V-MED	DO	RENT	REFR	IN	EC LW		500 K
18	X5134426	STAFF	DISPLY	60 K	SRT	1		V-MED	DBM	RELD	SWAP	IN	EC LW		0 K
19	W1093734	OFCSTART	SCHEDULR	6 K	SRT			V-MED	RON	RENT	REFR	IN	TM LW	INIT	6 K
20	W1141354	CATALOG	$LABEL	24 K	SRT	1	16	V-MED	RON	RELD	SWAP	IN	EC LW		2 K
21	W5141137	FSEDIT	FSED2	32 K	SRT	1		MEDIUM	RCK	RELD	SWAP	IN	EC LW		0 K
22	W5140902	CNP05A	‡RPG	28 K	SRT			V-LOW	MARIE	RELD	SWAP	IN			0 K
23	W5140927	CNP05A	‡RPG	28 K	SRT			V-LOW	MARIE	RELD	SWAP	IN			0 K

-- S T O R A G E   T O T A L S --

SYSTEM WORK -- SPACE --	SIZE	SWAP STAT	SWAPS	ACTIVE USERS	DEMAND COUNT
WORK STATION	4 K	IN	5	2	30
TRACE	48 K	NSW		0	0
INDEX INSERT	0 K			0	0
ACTIVE PROCS	2 K	IN		0	5
BATCH BSC	0 K			0	0
FORMAT INDEX	4 K	IN		0	4
SPELL CHECK	0 K			0	0
HELP AREA	12 K	IN		0	3
FMS I/O SUBR	6 K	OUT		0	0
FMS FOLDER	0 K			0	0
SPELL CHECK	20 K	OUT		0	0

NUCLEUS.	134 K
NONSWAPPABLE PROGRAM SPACE .	16 K
NONSWAPPABLE WORK SPACE. . .	48 K
TOTAL NONSWAPPABLE SPACE. . .	198 K
SYSTEM PROGRAM SPACE	152 K
USER PROGRAM SPACE	398 K
SWAPPABLE WORK SPACE	566 K
TOTAL SWAPPABLE SPACE	1116 K

USER AREA SPACE AVAILABLE. .	826 K
ACTUAL STORAGE COMMITMENT. .	65 %
ACTIVE STORAGE COMMITMENT. .	96 %
TOTAL STORAGE COMMITMENT . .	135 %
TOTAL A/F SPACE SIZE	104.0 K
ASSIGN/FREE SPACE USAGE. . .	80 %
LARGEST AVAIL A/F SEGMENT. .	2048 BYT
A/F SEGMENTS AVAILABLE . . .	104

7.1.4 SMF ALL REPORT

The following table shows the types of information that are listed on the ALL report:

ALL Printed Report	IPL configuration information
	Communications configuration data, if active and selected
	. Statistics for each sample interval: - Device usage rates - Task work area usage - Task status - I/O and SEC information by task, if selected - Terminated task data - User file access counters - System file access counters - Storage totals - System event counters (SEC) - I/O counters - Data storage attachment (DSA) usage - Communications line usage, if active and selected
	Summary information

This section contains a SMF ALL report except for the SUMMARY information.

SMF ALL Report (Part 1 of 7)

---------- I P L   C O N F I G U R A T I O N ----------

MAIN STORAGE SIZE	1024 K	SMF DATA COLLECTION DATE.	85/02/07
DISK CAPACITY	799.14 MB	RELEASE/MODIFICATION LEVEL. . . .	03/00
TASK WORK AREA SIZE	1100 BLOCKS	CONFIGURATION MEMBER NAME	S10CNFG
3262 PRINTER SUPPORTED. . . .	Y	COMMUNICATION LINES SUPPORTED . .	1,2,3,4,5,6,7,8
SPOOLING SUPPORTED.	Y	AUTOCALL LINES SUPPORTED.	NONE
REMOTE WORKSTATIONS SUPPORTED	Y	X.21 LINES SUPPORTED.	NONE
COMM CONTROLLER ATTACHED. . .	Y	X.25 LINES SUPPORTED.	NONE
DSC ATTACHED.	Y	TAPE DRIVES SUPPORTED	2
WSC ATTACHED.	Y	DISK DRIVES ATTACHED.	4
		SYSTEM MODEL NUMBER	53602

SMF ALL Report (Part 2 of 7)

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 2

SNAPSHOT TIME - 14.14.23.742 SAMPLE INTERVAL - 1.00.105

--------------- D E V I C E   U S A G E   R A T E S ---------------

MAIN STORAGE PROCESSOR . . . . 85 %
CONTROL STORAGE PROCESSOR. . . 26 %
WORKSTATION CONTROLLER QUEUE . 4 %
WORKSTATION CONTROLLER . . . . 4 %
DATA STORAGE CONTROLLER. . . . 6 %
DATA STORAGE ATTACHMENT. . . . 23 %
DISK 1 . . . . 41 %
DISK 2 . . . . 2 %
DISK 3 . . . . 4 %
DISK 4 . . . . 4 %

COMMUNICATION LINE 1. . . . . ** %
COMMUNICATION LINE 2. . . . . ** %
COMMUNICATION LINE 3. . . . . ** %
COMMUNICATION LINE 4. . . . . ** %
COMMUNICATION LINE 5. . . . . ** %
COMMUNICATION LINE 6. . . . . ** %
COMMUNICATION LINE 7. . . . . ** %
COMMUNICATION LINE 8. . . . . ** %

----- T A S K   W O R K   A R E A -----

TASK WORK AREA SIZE. . . . . . . 1510 BL:
TASK WORK AREA USAGE . . . . . . 68 %
TASK WORK AREA EXTENTS 1

--------------------------------- T A S K   S T A T U S ---------------------------------

	JOB	PROCEDURE	PROGRAM	PROG SIZE	TYPE	REQ CNT	WS OPS	PRIORITY	USER ID	ATTRIBUTE EXEC	ATTRIBUTE STOR	PROG SWAPS	STATUS SWAP	STATUS --WAIT--	SCHD	TWS	TWS SWAPS
1			SYS TASK	0 K	CMD-PR			SYSTEM		RENT	NUC		NUC	EC LW		0 K	
2	W3141319	SMFSTART	$SMFML	8 K	SMF			SYSTEM	RON	RELD	SWAP		IN			0 K	
3			SYS TASK	8*K	*SDLC-M			SYSTEM		RELD	NSW		NSW	EC		0 K	
4			SYS TASK	6 K	*SDLC-I			SYSTEM		RELD	SWAP		IN	EC		0 K	
5			SYS TASK	8*K	EM3270			SYSTEM		RELD	NSW		NSW	EC		0 K	
6			SYS TASK	8 K	BSC3270			SYSTEM		RELD	SWAP		IN	EC		0 K	
7			SYS TASK	24 K	APPC			SYSTEM		RELD	SWAP		IN	EC		8 K	
8			SYS TASK	10 K	*SYSTEM			SYSTEM		RELD	SWAP		IN	EC		0 K	
9			SYS TASK	18 K	PEER			SYSTEM		REUS	REFR		OUT	EC LW		0 K	
10			SYS TASK	24 K	SNA-RJE			SYSTEM		RELD	SWAP		IN	EC		0 K	
11			SYS TASK	6 K	SNA3270			SYSTEM		RENT	REFR		IN	EC		2 K	
12			SYS TASK	14 K	*CSNA			SYSTEM		RELD	SWAP		IN	EC		0 K	
13			SYS TASK	14 K	RWS			SYSTEM		REUS	REFR		IN	EC		0 K	
14			SYS TASK	2 K	*SYSTEM			SYSTEM		RELD	SWAP		IN	EC		0 K	
15	X1132407	SDA	‡SASF	32 K	SRT	1	7	V-MED	BSW	RELD	SWAP		IN	EC		0 K	
16	W8134016	INDEX	‡TETX2	46 K	SRT	1		MEDIUM	MARYJANE	RELD	SWAP		IN			0 K	
17	X2131350	IDDUDISK	‡DSIN	8 K	SRT	1		V-MED	DO	RENT	REFR		OUT	EC LW		0 K	
													IN	EC LW		500 K	

SMF ALL Report (Part 3 of 7)

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 3

	JOB	PROCEDURE	PROGRAM	PROG SIZE	TYPE	REQ CNT	WS OPS	PRIORITY	USER ID	ATTRIBUTE EXEC	STOR	PROG SWAPS	SWAP	----- STATUS ------ --WAIT--	SCHD	TWS	TWS SWAPS
18	X5134426	STAFF	DISPLY	60 K	SRT	1		V-MED	DBM	RELD	SWAP		IN	EC LW		0 K	
19	W1093734	OFCSTART	SCHEDULR	6 K	SRT			V-MED	RON	RENT	REFR		IN	TM LW	INIT	6 K	
20	W1141354	CATALOG	$LABEL	24 K	SRT	1	16	V-MED	RON	RELD	SWAP		IN	EC LW		2 K	
21	W5141137	FSEDIT	FSED2	32 K	SRT	1		MEDIUM	RCK	RELD	SWAP		IN	EC LW		0 K	
22	W5140902	CNP05A	#RPG	28 K	SRT			V-LOW	MARIE	RELD	SWAP		IN			0 K	
23	W5140927	CNP05A	#RPG	28 K	SRT			V-LOW	MARIE	RELD	SWAP		IN			0 K	

	JOB	PROC/TYPE	MSP USAGE	---- DISK 1 ---- READ	SCAN	WRITE	---- DISK 2 ---- READ	SCAN	WRITE	---- DISK 3 ---- READ	SCAN	WRITE	---- DISK 4 ---- READ	SCAN	WRITE
1		CMD-PR	1%	2	3	2	8	0	0	0	0	0	0	0	0
2	W3141319	SMF	0%	1	0	0	0	0	0	0	0	0	1	1	1
3		SDLC-M	0%	0	0	0	0	0	0	0	0	0	0	0	0
4		SDLC-I	0%	0	0	0	0	0	0	0	0	0	0	0	0
5		EM3270	0%	0	0	0	0	0	0	0	0	0	0	0	0
6		BSC3270	0%	0	0	0	0	0	0	0	0	0	0	0	0
7		APPC	0%	0	0	0	0	0	0	0	0	0	0	0	0
8		SYSTEM	0%	0	0	0	0	0	0	0	0	0	0	0	0
9		PEER	0%	0	0	0	0	0	0	0	0	0	0	0	0
10		SNA-RJE	0%	0	0	0	0	0	0	0	0	0	0	0	0
11		SNA3270	0%	0	0	0	0	0	0	0	0	0	0	0	0
12		CSNA	0%	0	0	0	0	0	0	0	0	0	0	0	0
13		RWS	0%	0	0	0	0	0	0	0	0	0	0	0	0
14		SYSTEM	0%	1	0	1	0	0	0	0	0	0	0	0	0
15	X1132407	SDA	4%	15	15	37	0	3	0	2	49	0	111	0	19
16	W8134016	INDEX	0%	0	0	0	0	0	0	0	0	0	0	0	0
17	X2131350	IDDUDISK	0%	0	0	0	0	0	0	0	0	0	0	0	0
18	X5134426	STAFF	0%	0	0	0	0	0	0	0	0	0	0	0	0
19	W1093734	OFCSTART	0%	3	0	2	0	0	0	0	0	0	5	0	0
20	W1141354	CATALOG	1%	6	7	4	0	0	0	0	0	0	0	0	0
21	W5141137	FSEDIT	0%	0	0	0	0	0	0	0	0	0	0	0	0
22	W5140902	CNP05A	40%	257	0	56	4	0	21	0	0	0	0	0	0
23	W5140927	CNP05A	39%	245	0	63	4	0	21	0	0	0	0	0	0

SMF ALL Report (Part 4 of 7)

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 4

	JOB	PROC/TYPE	PRNTR -OPS-	WKSTN COUNT	XIENT CALLS	XXFER CALLS	GEN WAITS	REC WAITS	JOB STEPS	RES T-OUT	MSP T-OUT	NOT USED
1		CMD-PR	0	3	2	30	0	0	0	1	1	0
2	W3141319	SMF	0	0	0	3	0	0	0	0	0	0
3		SDLC-M	0	0	0	0	0	0	0	0	0	0
4		SDLC-I	0	0	0	0	0	0	0	0	0	0
5		EM3270	0	0	0	0	0	0	0	1	0	0
6		BSC3270	0	0	0	0	0	0	0	0	0	0
7		APPC	0	0	0	0	0	0	0	0	0	0
8		SYSTEM	0	0	0	0	0	0	0	0	0	0
9		PEER	0	0	0	0	0	0	0	0	0	0
10		SNA-RJE	0	0	0	0	0	0	0	0	0	0
11		SNA3270	0	0	0	0	0	0	0	0	0	0
12		CSNA	0	0	0	0	0	0	0	0	0	0
13		RWS	0	0	0	0	0	0	0	0	0	0
14		SYSTEM	0	0	0	1	0	0	0	1	0	0
15	X1132407	SDA	0	3	5	149	0	0	1	17	1	0
16	W8134016	INDEX	0	0	0	0	0	0	0	0	0	0
17	X2131350	IDDUDISK	0	0	0	0	0	0	0	0	0	0
18	X5134426	STAFF	0	0	0	0	0	0	0	0	0	0
19	W1093734	OFCSTART	0	0	1	22	0	0	1	1	0	0
20	W1141354	CATALOG	0	0	1	71	0	0	1	3	2	0
21	W5141137	FSEDIT	0	0	0	0	0	0	0	0	0	0
22	W5140902	CNP05A	699	0	0	17	0	0	0	66	110	0
23	W5140927	CNP05A	675	0	0	17	0	0	0	67	108	0

T E R M I N A T E D   T A S K   D A T A

	JOB	PROC/TYPE	MSP USAGE	DISK 1 READ	DISK 1 SCAN	DISK 1 WRITE	DISK 2 READ	DISK 2 SCAN	DISK 2 WRITE	DISK 3 READ	DISK 3 SCAN	DISK 3 WRITE	DISK 4 READ	DISK 4 SCAN	DISK 4 WRITE
1	W1141349	SYSLIST	0%	3	1	2	0	0	0	0	0	0	0	0	0

JOB	PROC/TYPE	PRNTR -OPS-	WKSTN COUNT	XIENT CALLS	XXFER CALLS	GEN WAITS	REC WAITS	JOB STEPS	RES T-OUT	MSP T-OUT	NOT USED

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 5

1	W1141349 SYSLIST	0	0	0	21	0	0	1	0	0	0

USER FILE ACCESS COUNTERS

FILE LABEL	DATE CREATED	JOBNAME	FILE TYPE	FILE ORG	BLOCK LOC	LENGTH	DISK LOC	DATA READ	DATA WRTE	INDEX READ	INDEX SCAN	INDEX WRTE	REC WTS	GET LOG	GET PHYS	UPDATE LOG	UPDATE PHYS	DELETE LOG	DELETE PHYS	ADD LOG	ADD PHYS
#SD.X1	85/02/07		R	D	16348	378	A1	5	3	0	0	0	0	31	8	22	0	0	0	0	0
#S2.X1	85/02/07		R	D	308168	20	A4	105	20	0	0	0	0	281	125	117	0	0	0	0	0
*$OUACQ	85/02/06		R	I	294731	168	A4	1	0	2	0	0	0	2	3	0	0	0	0	0	0

SYSTEM FILE ACCESS COUNTERS

FILE LABEL	DATE CREATED	JOBNAME	FILE TYPE	FILE ORG	BLOCK LOC	LENGTH	DISK LOC	DATA READS	DATA SCANS	DATA WRTES
#SYSTASK					3259	1100	A1	12	0	11
#LIBRARY					4359	11500	A1	34	15	0
#SYSWORK					650	109	A1	10	10	9
#SYSHIST					759	2500	A1	11	0	11
#TRACE00					37143	50	A1	0	0	31
CSLIB					0	650	A1	5	0	0
$WORK	85/02/07	W5140902	S	S	15899	40	A1	0	0	56
$WORK	85/02/07	W5140927	S	S	15939	40	A1	0	0	63
MISCA1					0	78204	A1	503	2	26
##SPOOL1					125234	1498	A2	17	0	46
MISCA2					78204	78204	A2	0	3	0
#SDALIB			R	L	173309	500	A3	53	49	0
SMF.RAY	85/02/07		R	D	301381	200	A4	1	1	2
MISCA4					234612	78204	A4	2	0	2
*#SD.X1	85/02/07	X1132407	R	D	16348	378	A1	20	0	0
*X1132407	85/02/07	X1132407	S	D	301581	78	A4	0	0	20

STORAGE TOTALS

SYSTEM WORK SWAP ACTIVE DEMAND NUCLEUS. . . . . 134 K USER AREA SPACE AVAILABLE. . 826 K

SMF ALL Report (Part 6 of 7)

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 6

— SPACE —	SIZE	STAT	SWAPS	USERS	COUNT
WORK STATION	4 K	IN	5	2	30
TRACE	48 K	NSW		0	0
INDEX INSERT	0 K			0	0
ACTIVE PROCS	2 K	IN		0	5
BATCH BSC	0 K			0	0
FORMAT INDEX	4 K	IN		0	4
SPELL CHECK	0 K			0	0
HELP AREA	12 K	IN		0	3
FMS I/O SUBR	6 K	OUT		0	0
FMS FOLDER	0 K			0	0
SPELL CHECK	20 K	OUT		0	0

NONSWAPPABLE PROGRAM SPACE .	16 K
NONSWAPPABLE WORK SPACE. . .	48 K
TOTAL NONSWAPPABLE SPACE. . .	198 K
SYSTEM PROGRAM SPACE	152 K
USER PROGRAM SPACE	398 K
SWAPPABLE WORK SPACE	566 K
TOTAL SWAPPABLE SPACE	1116 K

ACTUAL STORAGE COMMITMENT. .	65 %
ACTIVE STORAGE COMMITMENT. .	96 %
TOTAL STORAGE COMMITMENT . .	135 %
TOTAL A/F SPACE SIZE	104.0 K
ASSIGN/FREE SPACE USAGE. . .	80 %
LARGEST AVAIL A/F SEGMENT. .	2048 BYT
A/F SEGMENTS AVAILABLE . . .	104

S Y S T E M   E V E N T   C O U N T E R S

MAIN STORAGE TRANSIENT CALLS. .	10
TRANSLATED TRANSFER CALLS . . .	331
ASYNCHRONOUS TRANSFER CALLS . .	2
MAIN STORAGE TRANSIENT LOADS. .	5
TRANSLATED TRANSFER LOADS . . .	11
MAIN STORAGE LOADER REQUESTS. .	51
SWAPS IN.	5
SWAPS OUT	0
SWAPS OUT, FORCED	0
TASK WORK AREA READ OPS	18
TASK WORK AREA WRITE OPS. . . .	11
MAIN STORAGE CLEAR OPS.	36
CONTROL STORAGE TRANSIENT CALLS	37
CONTROL STORAGE TRANSIENT LOADS	0
CONTROL STORAGE LOADER REQUESTS	0
SPOOL SEGMENTS ALLOCATED. . . .	4
SPOOL ENTRIES ALLOCATED	0
SPOOL EXTENTS ALLOCATED	0
GENERAL WAITS	0
DISK RECORD WAITS	0

TASK WORK AREA EXTENTS.	0
JOB INITIATIONS	2
JOB STEP INITIATIONS.	4
MRT ATTACHES.	0
MRT LOADS	0
JOB TERMINATIONS.	1
JOB STEP TERMINATIONS	2
ABNORMAL TERMINATIONS	0
DISK LOCKS SATISFIED.	2
DISK LOCKS EXPIRED.	0
ASSIGN/FREE EXTENSIONS.	0
ASSIGN/FREE REDUCTIONS.	0
PREEMPTIVE TASK DISPATCHES. . .	2013
RESOURCE TIMEOUTS	157
MAIN STORAGE PROCESSOR TIMEOUTS	222
WKSTN BUFFER READ RETRIES . . .	1
L-1 STORAGE RELEASES W/O SWAP .	14
L-1 STORAGE RELEASES W/ SWAP. .	0
L-2 STORAGE RELEASES W/O SWAP .	0
L-2 STORAGE RELEASES W/ SWAP. .	0

L-3 STORAGE RELEASES W/O SWAP .	0
L-3 STORAGE RELEASES W/ SWAP. .	0
L-4 STORAGE RELEASES W/O SWAP .	0
L-4 STORAGE RELEASES W/ SWAP. .	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0
NOT USED.	0

SMF ALL Report (Part 7 of 7)

REPORT DATE - 2/07/85 DATA COLLECTION FILE - SMF.RAY SYSTEM/36 MEASUREMENT FACILITY PAGE 7

I / O COUNTERS

DISK 1 READ OPS	587	DISK 3 READ OPS	52	DISKETTE 1 READ OPS	0
DISK 1 WRITE OPS.	230	DISK 3 WRITE OPS.	0	DISKETTE 2D READ OPS.	0
DISK 1 SCAN OPS	26	DISK 3 SCAN OPS	49	DISKETTE 1 WRITE OPS.	0
DISK 1 SEEK OPS	796	DISK 3 SEEK OPS	96	DISKETTE 2D WRITE OPS	0
DISK 1 SEEK OPS GT 1/3 DISK .	6.4 %	DISK 3 SEEK OPS GT 1/3 DISK .	0.0 %	DISKETTE SEEK OPS	0
DISK 1 AVERAGE SEEK LENGTH. .	45 CYL	DISK 3 AVERAGE SEEK LENGTH. .	2 CYL	1255 MICR OPS	0
DISK 2 READ OPS	16	DISK 4 READ OPS	117	72MD AUTO LOADER OPS.	9
DISK 2 WRITE OPS.	44	DISK 4 WRITE OPS.	20	DISKETTE HEAD CONTACT REVS. . .	0
DISK 2 SCAN OPS	3	DISK 4 SCAN OPS	1	LOCAL DISPLAY STATION OPS . . .	27
DISK 2 SEEK OPS	26	DISK 4 SEEK OPS	5	LOCAL PRINTER OPS	0
DISK 2 SEEK OPS GT 1/3 DISK .	0.0 %	DISK 4 SEEK OPS GT 1/3 DISK .	0.0 %	REMOTE DISPLAY STATION OPS. . .	0
DISK 2 AVERAGE SEEK LENGTH. .	3 CYL	DISK 4 AVERAGE SEEK LENGTH. .	58 CYL	REMOTE PRINTER OPS.	0
TAPE 1 READ BYTES	0 K	TAPE 2 READ BYTES	0 K	3262 PRINTER OPS.	0
TAPE 1 WRITE BYTES.	0 K	TAPE 2 WRITE BYTES.	0 K		
TAPE 1 REWIND OPS	0	TAPE 2 REWIND OPS	0		
TAPE 1 HITCHBACK OPS.	0	TAPE 2 HITCHBACK OPS.	0		

D S A AND TAPE USAGE

DSA DEVICE USAGE		DSA BUFFER USAGE		TAPE USAGE	
DISK 1.	37.5 %	ONE BUFFER TO M/C STORAGE . . .	41.2 %	TAPE 1.	0.0 %
DISK 2.	2.0 %	ONE BUFFER TO DSC	0.4 %	TAPE 2.	0.0 %
DISK 3.	3.6 %	TWO BUFFERS TO M/C STORAGE. . .	2.4 %	TAPE 1 DATA TRANSFER.	0.0 %
DISK 4.	3.4 %	TWO BUFFERS TO DSC.	0.0 %	TAPE 2 DATA TRANSFER.	0.0 %
DISKETTE TO MAIN STORAGE. . . .	0.0 %	TWO BUFFERS TO M/C AND DSC. . .	0.0 %	TAPE 1 START/STOP DATA TRANSFER	0.0 %
DISKETTE TO DSC	0.0 %	TOTAL DSA BUFFER USAGE.	23.2 %	TAPE 2 START/STOP DATA TRANSFER	0.0 %

7.2 DISK FILE (DATA COLLECTION FILE)

You have access to the file and input specifications for the data collection files for SMF (see Appendix C in the System Measurement Facility Guide). With this information, you have the option of creating some simple procedures and/or programs to monitor counters for a specific area. This will allow you:

- To monitor your system, or
- To check up on changes you are making on the system.

An example of this would be for communications. To get the communications error reports you must select the print all option. You may find it to your advantage to write a program to give you just the primary usages and counters plus the communications usages.

Another example might be if you are trying to determine how many job steps are involved in a new program you are using. You could write a program to show you totals for the task that was active. These numbers could then be used to see if the new job was efficient and possibly the time of day you should schedule it.

Three types of files can be created:

- SUMMARY
- DETAIL
- ALL

ALL is the <u>default</u> file type.

The table below shows the types of information that are contained in each file:

SUMMARY Report File	IPL configuration information
	Summary information
DETAIL Report File	IPL configuration information
	Communications configuration data, if active and selected
	. Statistics for each sample interval: - Device usage rates - Task work area usage - Task status - Storage totals
	Summary information
ALL Report File	IPL configuration information
	Communications configuration data, if active and selected
	. Statistics for each sample interval: - Device usage rates - Task work area usage - Task status - I/O and SEC information by task, if selected - Terminated task data - User file access counters - System file access counters - Storage totals - System event counters (SEC) - I/O counters - Data storage attachment (DSA) usage - Communications line usage, if active and selected
	Summary information

8.0 MONITORING PERFORMANCE

It is important to keep in mind that tuning your system is an ongoing process. You should execute SMF on a regular basis, not just when performance problems become apparent. You should be able, once your objectives are met, to identify potential performance problems before they become crises.

Producing the same type of report or reports at periodic intervals should be a standard procedure at most installations. Data obtained in this way will enable you to determine the trends and the usual values for your installation. Once you establish these values for individual workloads, you can monitor performance. Any variation to the values usually observed can be indicative of a performance problem and should be investigated.

Note: Limit the number of reports produced on a regular basis to those that you can examine quickly and easily. Use exception reporting to produce reports when specified thresholds are exceeded.

A measurement tool such as SMF can be run continuously with little degradation to the system's performance.

The following sections describe the key SMF measurements that you should examine regularly to monitor performance with maximum effectivness.

8.1 PRIMARY AND SECONDARY: COUNTERS AND UTILIZATIONS

Primary and secondary counters and utilizations are identified as those most likely to help in solving a performance problem and which best reflect overall system performance. These values can be found in the SMF summary report.

Primary Utilizations and Counters

The primary utilizations and counters are those values that should be examined first when using SMF.

Secondary Utilizations and Counters

The secondary utilizations and counters are the values that should be considered after the primary values.

Note: Where guidelines are appropriate, they are given (see the "Tuning Guidelines" section).

The following listing shows the primary and secondary (i.e.,significant) counters and utilizations. These are generally the <u>first</u> indicators of an SMF output that need to be evaluated.

SMF Significant Counters and Utilizations Chart		
Element	Type Element	
	Counter	Utilization
Primary		
Main storage processor		X
Control storage processor		X
Disk usage (disks 1-4)		X
Communication lines		X
Translated calls	X	
Translated loads	X	
Swaps in	X	
Swaps out	X	
Disk record waits	X	
Disk seek ops GT 1/3 disk		X
User area disk activity (UADA)	X	
Secondary		
Workstation controller		X
Spool extents allocated	X	
Task work area extents	X	
Storage release with swap (L3-L4)	X	
Storage release without swap (L3-L4)	X	
Disk 1-4 reads, scans, writes, seeks	X	
Local printer ops	X	
Local work station ops	X	
Job step initiations	X	
Resource timeouts	X	
Main storage processor timeouts	X	

Note: The above utilizations and counters can be found on the SMF SUMMARY listing.

Utilization measurements of the major resources--such as MSP, CSP, disk, and remote lines--are the most important. Utilization values can give the user insight to performance capabilities and trends. Some examples are:

- The relative limit or capacity of a resource. In other words, you can only get 100% out of the resource. The current value tells you at a glance where you are relative to resource saturation. Comparing the utilization values of the different resources might also point out where the system bottlenecks are. The user looks for ways to control the utilization of resources to keep them within acceptable guidelines or to keep various components in balance with one another.

  A calculated percentage (0-100) indicates the relative activity of each device to the others. Over 60% is significant. If the activity is in the 60 to 85% range, some performance degradation is probably occurring because of the large request queue at the device. If the device busy value is over 85%, you could be experiencing severe queuing problems and performance degradation.

- How much you may have to wait for a resource. That is to say, the greater the utilization of a resource (when multiple users are competing for that resource), then the greater the probability that you will have to wait your turn for the resource. This is the phenomenon that various queuing theories are designed to estimate.

- Is the distribution of utilization uniform or the way that you want it? For example, in the case of disk utilization, you want the disk drives to be utilized as evenly as possible. The same arguments hold true for remote lines. In the case of MSP utilization, it may be acceptable to run at high utilizations, but the important thing is to run with the right proportions for interactive work, system overhead, and batch work.

More on the relationship of throughput and utilization.

As throughput increases,

> which is generally the case when more work stations are added to the system,
>
> and/or more applications are added,
>
> which means more Enter keys are depressed per hour,

then the utilization of system resources is going to increase as well. As utilization increases, so does the probability that:

- More users will have to wait,
- The queues will be longer, and
- There will be more contention for resources.

8.2 ADDITIONAL COUNTERS TO TRACK

There are several counters you can track that indicate work change on your system. To take advantage of this, run SMF on a regular basis. Daily or biweekly is probably the best. The counters that you can follow are:

- Job step initiations
- Local display station ops
- Remote display station ops
- Local printer ops
- Remote printer ops
- 3262 printer ops
- Total reads, writes and scans for the disks

Those that would probably offer the greatest benefit are the Job Step Initiations, Local Display Station Ops, Remote Display Station Ops and the Total Reads, Writes and Scans for the Disks.

The printer ops counters may be useful if you do a lot of printing and want to know if it is increasing. The number of 3262 Printer Ops is the same as the number of lines printed.

The best method to track these counters for an indication of work is by plotting the counters. Two examples are shown in SMF Utilizations and Counters Usage Chart. Note that over the long haul the Display Station Ops are increasing. You may want to add the local and remote display station ops or plot just the local display station ops. You may want to use several counters together or plot them separately. For instance, suppose you plotted job step initiations and display station ops as shown. Over time the job step initiations remained constant but the display station ops went up. It may indicate that the operators are doing more work in the programs they are running. Perhaps more items are being entered per order.

Plotting the disk operations will indicate whether the number of accesses to the disks is increasing. Plotting each disk drive separately is not recommended since a change in file placement or scratch file allocation will change the results but not the total disk accesses. Add all the disk operations for reads, writes and scans and plot the final number. You could also plot all the reads and/or writes separately.

You may plot these counters as shown in SMF Utilizations and Counters Usage Chart, by per-minute values or by total counts. If you plot the per-minute values, make sure you include only your busy periods of the day and make sure to plot the same periods every time. For example, if you run SMF from 6:00 AM to 8:00 PM and you want to plot the per-minute values, do one SMF report from 8:00 AM to 12:00 noon and another from 1:00 PM to 5:00 PM and average the two reports values. You do not want the other times of the day because the operators did not start until 8:00 AM, went to lunch and quit at 5:00 PM. If your daily processing is the same Monday through Thursday and is different on Friday, keep two plots--one of the activity from Monday through Thursday and one for Fridays. You may then decide to run SMF three days a week for plotting purposes: Monday, Wednesday and Friday.

9.0 ANALYSIS OF SMF OUTPUT

Enhanced Functions

There are several new features on System/36 SMF compared to System/34 SMF. These include I/O and SEC data by task and SMFDATA as well as new counters detailing display station operations and job information. (See the section on Counters to Track below for a description on use of the new counters and Data by File.)

I/O (input/output) and SEC (system event counter) data by task provide information about each task that is running. This is done on a sample interval basis. Some of the usages and counters reported are listed below.

- Main storage processor usage
- Disk reads, writes and scans are given for each disk drive
- Printer ops
- Work station counts which is the number of enter, function and command keys pressed
- Disk record waits
- Job steps
- Resource time outs
- Main storage processor time outs

Tasks that ended during an interval have their final counts reported in the Terminated Task Data section of the SMF report. Details on Counters below.

SMFDATA is a procedure that creates a sequential file containing the information that would be contained in a printed SMF report. This file can be accessed by programs written in languages such as RPG II and COBOL. The layout of the records in this file is shown in the SMF Guide, Appendix C. This was new in Release 2.

The data by file SMF option is new in Release 3 and will list the number of accesses made to all the open files on a snapshot basis. Both physical and logical accesses are listed, as well as the index and data portions for indexed files. User files are listed separately from the system files. However, all libraries are included in the system file access counters area. Appendix D in the Release 3 SMF Guide has a sample program (FILEPROG) that shows what you can do with data by file when you use the SMFDATA procedure.

On the following pages are examples of what can be done with SMFDATA.

- Sample of the output from RPG II programs.
- Note that files and libraries are listed by usage with the most-used first. This removes the guess work.

9.1 SMFDATA EXAMPLE #1

Program - FILEPROG

LABEL	DATE	TYPE	ORG	LOCATION	LENGTH	DISK	TOTAL OPS	GET	UPDATE	DELETE	ADD	READ	SCAN	WRITE
FILEB1	84/08/08	R	I	11497	2355	A1	9650	7725	1925	0	0	0	0	0
FILEB2	84/08/08	R	I	152545	2355	A2	8805	7063	1742	0	0	0	0	0
#SYSTASK				967	1500	A1	7362	0	0	0	0	4340	0	3022
FILEB3	84/08/08	R	I	145480	2355	A2	7126	5659	1467	0	0	0	0	0
FILEB4	84/08/08	R	I	136388	2355	A2	6917	5490	1427	0	0	0	0	0
D457RAMP		R	L	9997	1500	A1	3968	0	0	0	0	3356	612	0
#LIBRARY				2467	7500	A1	2346	0	0	0	0	2216	130	0
CSLIB				0	650	A1	1875	0	0	0	0	1853	0	22
FILEC1	84/08/09	R	S	48521	59	A1	1686	0	0	0	1686	0	0	0
FILEB5	84/08/08	R	I	129651	2355	A2	1413	1116	297	0	0	0	0	0
FILEC2	84/08/09	R	S	110037	59	A2	1388	0	0	0	1388	0	0	0
FILEC3	84/08/09	R	S	48580	59	A1	1301	0	0	0	1301	0	0	0
FILEA1	84/08/08	R	I	110096	2027	A2	1043	1043	0	0	0	0	0	0
FILEA2	84/08/08	R	I	37074	2027	A1	972	972	0	0	0	0	0	0
#SYSHIST				717	250	A1	965	0	0	0	0	482	0	483
FILED3	84/08/08	R	I	143125	2355	A2	746	746	0	0	0	0	0	0
FILED4	84/08/08	R	I	23272	2355	A1	744	744	0	0	0	0	0	0
FILED1	84/08/08	R	I	150190	2355	A2	724	724	0	0	0	0	0	0
FILED2	84/08/08	R	I	30009	2355	A1	717	717	0	0	0	0	0	0
FILEA3	84/08/08	R	I	27982	2027	A1	622	622	0	0	0	0	0	0
FILEA4	84/08/08	R	I	138743	2027	A2	612	612	0	0	0	0	0	0
FILEC4	84/08/09	R	S	109978	59	A2	433	0	0	0	433	0	0	0
FILEE1	84/08/08	R	I	41456	2355	A1	360	360	0	0	0	0	0	0
MISCA2				78204	78204	A2	360	0	0	0	0	360	0	0
FILEE2	84/08/08	R	I	147835	2355	A2	347	347	0	0	0	0	0	0
#SYSWORK				650	67	A1	300	0	0	0	0	181	68	51
FILEE3	84/08/08	R	I	140770	2355	A2	270	270	0	0	0	0	0	0
FILEE4	84/08/08	R	I	32364	2355	A1	269	269	0	0	0	0	0	0
FILEA5	84/08/08	R	I	132006	2027	A2	95	95	0	0	0	0	0	0
SMF.8091	84/08/09	R	D	109778	200	A2	60	0	0	0	0	0	0	60
FILED5	84/08/08	R	I	43811	2355	A1	54	54	0	0	0	0	0	0
FILEE5	84/08/08	R	I	127296	2355	A2	54	54	0	0	0	0	0	0
MISCA1				0	78204	A1	48	0	0	0	0	48	0	0

9.2 SMFDATA EXAMPLE #2

Program - SNAPS

SNAPSHOT TIME.	11.37.07	11.38.07	11.39.08	11.40.08	11.41.09
MAIN STORAGE PROCESSOR	66 %	59 %	60 %	56 %	66 %
CONTROL STORAGE PROCESSOR. . .	68 %	71 %	71 %	71 %	76 %
DISK 1 USAGE	71 %	84 %	79 %	78 %	76 %
DISK 2 USAGE	23 %	24 %	31 %	33 %	32 %
DISK 3 USAGE	22 %	28 %	35 %	41 %	40 %
DISK 4 USAGE	43 %	33 %	54 %	62 %	43 %
DISK 1 SEEK OPS GT 1/3 DISK. .	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %
DISK 2 SEEK OPS GT 1/3 DISK. .	17.4 %	11.4 %	21.9 %	14.6 %	17.8 %
DISK 3 SEEK OPS GT 1/3 DISK. .	4.0 %	12.3 %	22.0 %	19.7 %	14.1 %
DISK 4 SEEK OPS GT 1/3 DISK. .	34.1 %	31.5 %	41.2 %	35.5 %	36.9 %
DISK RECORD WAITS.	0	1	0	0	1
TRANSLATED TRANSFER CALLS. . .	1125	1171	759	726	810
TRANSLATED TRANSFER LOADS. . .	174	204	170	94	134
TRANSLATED XFER CALLS/LOADS. .	6.5	5.7	4.5	7.7	6.0
SWAPS IN	294	355	335	379	408
SWAPS OUT.	277	321	290	338	347
COMMUNICATION LINE 1 USAGE . .	0.0 %	62.6 %	57.3 %	52.5 %	62.5 %
COMMUNICATION LINE 1 ERRORS. .	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %
COMMUNICATION LINE 2 USAGE . .	0.0 %	74.7 %	69.8 %	78.2 %	75.1 %
COMMUNICATION LINE 2 ERRORS. .	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %
COMMUNICATION LINE 3 USAGE . .	0.0 %	0.4 %	7.5 %	14.5 %	14.3 %
COMMUNICATION LINE 3 ERRORS. .	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %
COMMUNICATION LINE 4 USAGE . .	0.0 %	0.0 %	8.6 %	16.6 %	16.3 %
COMMUNICATION LINE 4 ERRORS. .	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %
NUMBER OF ACTIVE TASKS	33	32	36	35	35

9.3 SMFDATA EXAMPLE #3

Program - TASK

PROCEDURE NAME.	NB04A
JOB NAME.	R1122627
MAXIMUM PROGRAM SIZE.	28 K
MAXIMUM REQUEST COUNT	4
TOTAL WORKSTATION OPS	1170
TOTAL PROGRAM SWAPS	40
MAXIMUM TWS SIZE.	0 K
TOTAL TWS SWAPS	0
MAXIMUM MSP USAGE	3 %
AVERAGE MSP USAGE	3 %
TOTAL PRINTER OPS	0
TOTAL WORKSTATION COUNT . . .	534
TOTAL RECORD WAITS.	0
TOTAL JOB STEPS	0
TOTAL DISK 1 READS.	730
TOTAL DISK 1 SCANS.	279
TOTAL DISK 1 WRITES	486
TOTAL DISK 2 READS.	409
TOTAL DISK 2 SCANS.	355
TOTAL DISK 2 WRITES	0
TOTAL DISK 3 READS.	0
TOTAL DISK 3 SCANS.	0
TOTAL DISK 3 WRITES	0
TOTAL DISK 4 READS.	0
TOTAL DISK 4 SCANS.	0
TOTAL DISK 4 WRITES	0

9.4 USAGES: CLARIFICATION AND UNDERSTANDING

Assign/Free Space Usage

This usage gives the percentage of the used assign/free space. But the System/36 is a demand machine and dynamically takes as much space as it needs from the user area and adds it to the assign/free space. The system will also return unused assign/free space to the user area. As a result, the assign/free space is constantly changing but the per cent of assign/free space used will not change very much because it is calculated by dividing the spaced used by the total space. Since both the space used and the total space are changing, the usage values really do not tell you anything.

Total Storage Commitment

This usage can be misleading because of the way people use it. The problem is that operators use this value as an indication of how main storage is being utilized. Often they are wrong because there are programs in the system called NEP-MRTs. These programs contribute to the total storage commitment but do NOT contribute to the UADA since they do not have users connected to them and they are swapped out. A second contributing factor is that other interactive programs remain active while the operators take breaks, answer the telephone, and so on. A third contributing factor is task work space and system work space that is allocated and swapped out with no demand for their use. Because of the above, use the UADA as an indication of how main storage is being used. Anyone who obtains more main storage based on total storage commitment (they want better performance) may be disappointed unless their UADA values (swapping activity) are also high enough to show that the main storage was needed. Systems exist that have UADA value less than 100 and total storage commitment values in excess of 200%. Adding main storage to these systems will not improve performance. (See Swapping -- User Area Disk Activity (UADA) section below.)

Priority

The task status section of SMF contains a column labeled priority. The tasks are listed from highest to lowest priority. Priorities are assigned to a task as HIGH, MEDIUM or LOW. If you did not assign a priority to a task, its priority will be V-MED (Variable Medium) or V-LOW (Variable Low), except for the system routines whose priority will be SYSTEM or MEDIUM (for example SPOOL). Even if you assigned a priority to a task, the system may temporarily assign a different priority for special situations. It is recommended that you do not assign priorities but let the system assign the priority. Priority should only be used as an exception condition. Using a high priority for a given job will impact all other jobs negatively.

All tasks that did not have a priority assigned to them begin with a base priority of 129. If a task has an assigned priority of HIGH, MEDIUM or LOW, it begins with a base priority of 192, 128 or 64 respectively. The lowest priority in the system is 64, the highest is 255. All priorities equal to or greater than 240 are SYSTEM priorities. The following chart lists the priorities from high to low.

USAGE	FIXED				VARIABLE		
	DEFINED		VALUE	SWITCH	DEFAULT	VALUE	SWITCH
SYSTEM	SYSTEM	1	255				
		2					
		3					
		4					
		5					
		6					
		7	240				
USER	HIGH		239	YES		239	
				or			
			192	ON			
	MEDIUM		191				NO
			128		NORMAL		or
	LOW		127			69	OFF
			64				

When a task is initiated, it starts out with a base priority. To the base, a value of 4 is added for each display station attached to the task. This is called its run time priority. For example, a MRT program with 3 display stations attached will have a base priority of 129 and a run time priority of 141 (129+12). Each time the task does a work station read operation the priority of the task is lowered by 4. A check is then made to determine if the priority is lower than to its run time priority. If not, the task will execute at that priority. This ensures that all tasks will be given an equal share of the processing.

When a resource time-out occurs, the system checks to determine if it is this task's first resource time-out since it was swapped into main storage. If it is, the system does nothing except mark the task as having had its first resource time-out. If it is not, the priority of the task is lowered by 4. Unless the user assigned a priority to the task of HIGH, MEDIUM or LOW, the priority can be lowered by resource time-outs until it reaches the priority just above LOW. If the task has been assigned a priority of HIGH, MEDIUM or LOW, it will only be allowed to lower the priority to the base priority for HIGH, MEDIUM or LOW. When the task again does a work station read operation, it will again check to determine if the priority is lower than the base priority for that task. If it is lower, the priority is restored to its run time priority.

Batch tasks with a priority assigned will execute at that priority. Batch tasks without a priority assigned and that do no work station read operations will have their priority lowered each time a resource time-out occurs until the priority reaches the priority just above LOW. It will remain there until the job ends or a work station read should have a lower priority than interactive jobs to promote good response times for the interactive jobs.

9.5 PROCESSORS

9.5.1 PRIMARY

Main Storage Processor (MSP)

Use the MSP utilization to compare with other runs. The question to ask is, What changed to produce the increase or decrease?

If the average MSP exceeds 60%, begin to look for the tasks that are using the MSP more than others. These can be found by collecting I/O and SEC data by task and determining the MSP usage for each task. If the response times have slowed, consider rescheduling the task that is using the most MSP. If the response times have not degraded, then assume that you are using the MSP to its full extent. If the average MSP exceeds 80%, then you probably have a problem and should definitely reschedule one or more tasks.

One way to reschedule batch jobs without being concerned about starting them at the correct time is to submit all batch jobs through the JOBQ. This will prevent multiple batch jobs from running by stopping the JOBQ.

When the system is being used for multiple batch jobs or many interactive jobs, the MSP will almost always, at some point in the day, peak at some value over 80%. This is nothing to worry about. Certain functions, usually batch type jobs (sorts, compiles, and so on), will use the MSP heavily. Avoid giving two or more batch jobs high priority. The system may appear to be hung because the batch jobs will struggle for the system resources while locking out the interactive jobs.

Control Storage Processor (CSP)

As with the MSP, use this value when comparing with other runs. Again the question to ask is, What changed to produce the increase or decrease? Some of the tasks that cause the CSP to be heavily used are programs written in FORTRAN such as the Business Graphics Utilities (BGU), programs written in BASIC such as BRADS/36, and tracing via the TRACE procedure.

Two other functions in the CSP are communications polling and disk I/O operations. These two functions could be done in some of the other System/36 processors. However, you must purchase them since they are not part of the base system. The Data Storage Controller (DSC) will reduce the CSP usage by approximately 10-15%. These CSP usage reductions will be true only if the CSP is heavily used, that is, greater than 65%.

The 5362 system unit also does the Work Station Controller (WSC) functions in the CSP, unless the Workstation Expansion feature is installed. Installation of this feature will reduce the CSP usage by approximately 5-10% assuming the CSP usage is greater than 65%.

The Stage 2 CSP would also help to reduce the CSP usage since it is a faster processor. However, you must purchase it. It could reduce the utilization by approximately 10%-12% if the CSP is greater than 65%.

Average CSP usage exceeding 85% will probably result in slower response times. If this occurs, reschedule one or more jobs that are causing the problem, or buy a DSC, MLCA or Stage 2 processor or all three. Consider the costs and where the next bottleneck will occur before buying additional hardware. If the average CSP exceeds 65%, you should begin to plan or consider changes since the system is approaching a problem situation.

The 5362 system unit also does the Work Station Controller (WSC) functions in the CSP, unless the Workstation Expansion feature is installed. Installation of this feature will reduce the CSP usage by approximately 5-10% assuming the CSP usage is greater than 65%.

9.5.2 SECONDARY

Workstation Controller

The Workstation Controller handles all I/O operations to locally attached display stations and work station printers. Average usage below 60% will not cause any problems. Average usage above 80% is probably affecting the response times. One solution is to reschedule some jobs and/or reduce the amount of processing the workstation controller has to do. Processing can be reduced by cutting down on the amount of data sent and received from display stations. You could use Put Override and Erase Input Fields in your formats rather than resending the entire format. Additionally you could reduce the number of fields that have to be processed by the workstation controller. An example of this is a screen that has ten column headings, each defined in the SFGR specifications as a separate constant field. Defining the ten column headings as one constant field will reduce the processing done by the workstation controller. Keeping the fields in the formats in order will also reduce processing. For example, if you have three columns of fields, the workstation controller will have less columns. The only other approach is to add an external controller (5294) using a high speed remote line (56,000 BPS) and modem eliminators. This "in-house local" remote line could handle some of the workstation controller load and thus eliminate the problem. Response times at these display stations should not be much different than experienced by the locally attached display stations. However, consider the cost before implementing this solution.

9.6 DISK

9.6.1 PRIMARY

Disk 1-4

An important step is to balance the disk usage. Try to get the disk usages within 10-15% of each other with the least usage on disk A1.

When files are allocated without a given drive preference, the system will attempt to allocate the file on the least-used drive. For temporary files, the system determines which drive to select based on the last hour's disk activity. For permanent files, the system determines which drive to select based on the total count of disk activity that has been recorded for the drives.

One of the most frequently asked questions is "When is a disk a bottleneck?" Generally, average disk usage of 40% or less is not a problem. On the other hand, it is probably safe to say that a disk is a bottleneck if the average usage is 85% or more. Start to become suspicious when the average usage exceeds 60%.

There are several steps that the user can take when the disk becomes a problem. First look at the Disk Seek Ops GT 1/3 Disk. By reducing the seeking distance, you will reduce the distance the disk arm must travel and thus reduce the disk usage or allow more disk operations to occur in the same time period. After the seek distances have been reduced as much as possible, then balance the disk usage. Both the reduction in seek distances and the balancing of the disks respectively are the procedures used to move files and libraries. See Disk Seek Ops GT 1/3 Disk below.

If the disks continue to cause problems after the seek reduction and disk balancing, you may want to reschedule some of the jobs that use the disk heavily or add additional disks. You must then balance the disk usage again. Collection of I/O and SEC data by task will show which jobs are the heavy disk users. If the swapping activity (see Swapping below) is causing the disk problem, you can add main storage. After adding main storage, rebalance the disks.

Disk Record Waits

Disk Record Waits indicate that two programs want the same record. At least one of the programs wants the record for update. The counter registers the number of waits that have occurred. The programs that are causing the disk record waits can be found by collecting I/O and SEC data by task. The column containing the disk record waits is labeled REC WAITS. It is then relatively easy to look at the files used in the procedure and determine which one is causing the disk record waits. You will also have to look at the other procedures that had the disk record wait. With Data by File in Release 3, the files that have the disk record waits can also be found. The column containing the disk record waits is labeled REC WTS.

To avoid getting disk record waits, reschedule one or more jobs. One of the common problems is to read a record for update and, while it is being updated, have another program request the same record from the same file. To overcome this problem, change the disposition (DISP) of the file in one or more procedures. DISP-SHR is the same as SHRMM, but if the file is only being read from one program while updated by another, the program that is only reading the file would benefit by using SHRRM.

The length of the disk record wait is application dependent, therefore the total number of disk record waits may not be a good indication that a problem exits. If you are experiencing waits but your response time is acceptable, you may not need to find the cause. On the other hand, if there are long responses on one or more display stations because of the disk record waits, then it is worth spending the time finding out which programs are causing the problem. Consider searching for the cause any time the disk record waits are affecting the response times.

Disk Seek Ops GT 1/3 Disk

Reducing the seek span for disk operations may be the most important thing you can do to improve the performance of the system. The disk seek ops GT 1/3 disk should be reduced to as close as possible to zero. The second most important thing is to balance the disk usage. By using the recommended file placements, your scratch and job files, as well as any transaction files created, will end up next to your most-used files (except on A1) when they are allocated by the system in the free space on disk.

The farther the disk arm must move to read, scan, or write data on the disk, the longer the disk operation will take to complete. When there are a large number of these long disk operations, your system performance is degraded. Use the CATALOG procedure and specify by LOCATION to find the current location of files and libraries to move as well as where to move them. Use the COPYDATA procedure to move files and the ALOCLIBR procedure to move libraries to reduce the seek span. The files and libraries that you should group together are those used most frequently. For example, an application that uses ten files may access four of them only a couple of times each during the execution. These files should be classified as least-used files. The execution should be classified as most-used files. Determine which files and libraries are the most used during your daily processing (probably less than 50). You can use the Data by File option available in Release 3 to determine the files. These files should be grouped together on the disk to reduce the seek span. Libraries should be included in this placement since they contain the formats that are being accessed every time a program writes a screen to a display station. They also contain the procedures and load members that you are running. Files and libraries used only for month-end processing should be classified as least-used files and libraries. However, you may want to place the heavily used month-end files next to your most-used files to reduce the seek span when you do month-end processing.

To achieve the placements, first compress the disk spindle you are going to move files and libraries on. The direction of compression is determined by the spindle you are working with.

To achieve the best results, drive utilization must be evenly shared across all drives:

```
1, 2, 4                                                                3
DRIVES                                                            DRIVES

      |             Most-Used           Least-Used                |
<---- | System      Files and           Files and           Free  |
      | Files     | Libraries         | Libraries       | Space   |
    1 |********|XXXXXXXXXXXXXXXXX|\\\\\\\\\\\\\\\\\\\|//////////| 1


      |                         Most-Used         Least-Used     |
----> | Free                    Files and         Files and      |
      | Space                 | Libraries       | Libraries      |
    2 |////////////////////|XXXXXXXXXXXXXXXXXX|\\\\\\\\\\\\\\\\|


      | Least-Used                 Most-Used                     |
<---- | Files and                  Files and               Free  |
      | Libraries       |          Libraries         |   Space   |
    3 |\\\\\\\\\\\\\\\\|XXXXXXXXXXXXXXXXXXXXXXXXXXXXX|//////////| 2


      |                              Most-Used      Least-Used   |
----> | Free                         Files and      Files and    |
      | Space                      | Libraries    | Libraries    |
    4 |/////////////////////////|XXXXXXXXXXXXXXXX|\\\\\\\\\\\\\\| 3
```

```
 ____
|****|   System Space
|____|

 ____
|////|   Free Space
|____|

 ____
|XXXX|   Most-used files and libraries
|____|

 ____
|\\\\|   Least-used files and libraries
|____|

 ______
<-----  |
----->  |---> Default direction of movement for files, folders, and
______  |     libraries during a COMPRESS.
```

When moving files and libraries on spindle A1, you should first do a:

COMPRESS A1,FREELOW

This will create free space next to the system files. Next move the files and libraries to this free space. Finally, do a:

COMPRESS A1,FREEHIGH

To move files and libraries on spindles A2, A3 and A4, do COMPRESS on free space. Then do another COMPRESS on the spindle.

COMPRESS A2,FREELOW

A2 for example; use the COPYDATA procedure to move the high-use file and specify the LOCATION parameter as to where the file is to placed (that is, low-end block value).

If the location is not specified, the file could be moved into the space just vacated by a previous COPYDATA procedure.

The spool file should be included in the seek span of the most-used files. To move the spool file you must IPL twice. For moving the spool file on spindles A2, A3, or A4, do the following: On the IPL SIGN ON screen for the first IPL, request the overrides. Take option 4 to request Change Print Spooling Status on the IPL OVERRIDES MENU. Answer Y to both Cancel Print Spooling and Delete Print Spool File. Make sure the spool file is empty before doing this or any data in the spool file will be lost. When IPL is complete do a COMPRESS to eliminate the "hole" on disk where the spool file was. Then do a CNFIGSSP. Take the option to Define Base SSP Values. This will allow you to increase or decrease the size of the spool file as well as specify the spindle preference for the spool file. After the CNFIGSSP is complete, IPL again but do not request the overrides for changing spool.

For moving the spool file on spindle A1 (you want it next to the system files), do a:

COMPRESS A1,FREELOW

To create space next to the system files instead of doing the plain COMPRESS. After the second IPL, you must then do a:

COMPRESS A1,FREEHIGH

When a reference is made to one-third of the disk, the area that is being referred to is one-third of the disk in either direction from the first third of the disk.

When placing files and libraries, you must consider your entire work cycle. This means that you must group the files that you use heavily over the entire week or two week period that comprise your work cycle. Never use just one day's SMF data to place files and libraries. Use SMF data from your entire work cycle. Don't spend an inordinate amount of time moving files and libraries. Determine which files and libraries you need to move and then move them. Check your results with additional SMF runs. After two to three iterations you should have your most-used files and libraries together. Even if you do not have a performance problem, you may want to reduce the seek span to either improve performance or to head off future problems. The same is true for balancing the disk usages. If you reorganize your files often and it causes those files to move from their desired location, consider writing a procedure that restores the files to their desired location after the reorganize.

Also note that if the disk is lightly used (less than 20%), eliminating the seeks greater than one-third of the disk will probably not perceivably change the performance. This is because the disk is so lightly loaded that queuing of the disk operations does not occur, thus reducing the seeks greater than one-third of the disk will have an insignificant effect.

File Placement Scenario

The data by file output can help you determine what files and libraries on your system have the most activity. Then you can place your files so the seek span is at a minimum, thus giving you better performance.

This scenario assumes:

- A one week work cycle, and
- Uses steps 1 through 5 from the sample program in Appendix D of the SMF Guide.

Daily processing ...

1. Run SMF and collect data by file. Create a new data collection file each day.

2. At the end of each day, run the SMFDATA procedure to create a SMF report file (You can find the formats in Appendix C of the SMF Guide).

3. Do Step 1 of the sample program. Step 1 reads the report file and creates a temporary file "REPORT1" containing only user (AHA, AHB) and system (AIA) file records.

4. Save file "REPORT1" on diskette.

   On Tuesday through Friday change this step to specify "ADD" which appends the disk file to the diskette file. At the end of the week, the "REPORT1" file contains all the file counts for the entire week.

5. Delete SMF data collection and report files.

End of the week processing ...

1. Restore the "REPORT1" file.

2. Do Step 2 of the sample program. Step 2 sorts the "REPORT1" by the file label field so records with the same file label are grouped together. A temporary file, "REPORTS1", is created.

3. Do Step 3 of the sample program. Step 3 reads file "REPORTS1", produces one record per file with a cumulative total of the number of disk operations in the field "TOTAL OPS", and writes the records to a temporary file "REPORT2".

4. Do Step 4 of the sample program. Step 4 sorts file "REPORT2" using field "TOTAL OPS" in descending order. A temporary file "REPORT2S", is created.

5. Do Step 5 of the sample program. Step 5 reads file "REPORT2S" and prints the file access report for your analysis. A file access report report contains:

   - File label
   - Creation date
   - File type
   - File organization
   - File location
   - File length
   - Disk location
   - Total physical operation
   - Individual physical counts

   The individual physical counts contain data for user files and system files. For user files, these counts include gets, updates, deletes, and adds. For system files, these counts include reads, writes, and scans.

File placement example for A1...

Disk A1 file placement:

```
      |                 Files and Libraries
      | System |     B       A       C C        |  Free    |
      | Files  |     1       2       1 3        |  Space   |
      |________|________________________________|__________|
```

1. Perform: COMPRESS A1,FREELOW

```
      |                           Files and Libraries
<---- | System |  Free   |     B       A       C C        |
      | Files  |  Space  |     1       2       1 3        |
      |________|_________|________________________________|
```

2.

a. Move high-use file with COPYDATA:

FILEB1,FILEC1,FILEC3,FILEA2,.....

```
      |                                    Files and Libraries
      | System | B C C A |  Free   |                          |
      | Files  | 1 1 3 2 |  Space  |                          |
      |________|_________|_________|__________________________|
```

b. Move high-use libraries with ALOCLIBR:

D457RAMP,.....

3. Perform: COMPRESS A1,FREEHIGH

```
      |        |Most-Used|      Least-Used           |         |
      |        |         |  Files and Libraries      |         |
----> | System | B C C A |                           |  Free   |
      | Files  | 1 1 3 2 |                           |  Space  |
      |________|_________|___________________________|_________|
```

File placement example for A2...

Disk A2 file placement:

```
      |                Files and Libraries                   |
      | Free    |                          B  B  B  B        |
      | Space   |                          5  4  3  2        |
      |_________|____________________________________________|
```

1. Perform: COMPRESS A2,FREELOW

```
      |                           Files and Libraries        |
<---- |      Free         |                B  B  B  B        |
      |      Space        |                5  4  3  2        |
      |___________________|__________________________________|
```

2.

a. Move high-use file with COPYDATA and specify location (low end):

FILEB2,FILEB3,FILEB4,FILEB5,.....

```
      |                           Files and Libraries        |
      | B  B  B  B        |                                  |
      | 2  3  4  5        |                                  |
      |___________________|__________________________________|
```

b. Move high-use libraries with ALOCLIBR:

LIBRxxxx,.....

3. Perform: COMPRESS A2,FREELOW

```
      |         | Most-Used  |       Least-Used              |
      |         |            |  Files and Libraries          |
<---- | Free    | B  B  B  B |                               |
      | Space   | 2  3  4  5 |                               |
      |_________|____________|_______________________________|
```

9.6.2 SECONDARY

Task Work Area Extents

If you are getting Task Work Area Extents, increase the size of the Task Work Area. When the extents are created they may be placed outside of the seek span of your most used files and libraries. This will increase the time it takes to do the disk operations (increases the seeks greater than one-third of the disk) and thus degrade performance. With Release 2 the Task Work Area can be increased by changing the size using CNFIGSSP and then doing an IPL.

Spool Extents Allocated

If the Spool Extents Allocated exceeds 2, increase the size of the spool file. When the spool extents are created they are probably placed outside of the seek span of your most-used files and libraries. This will increase the time it takes to do the disk operations (increases the seeks greater than one-third of the disk) and thus degrade performance. You may want to increase the size of the spool file if you get any extents that fall outside the seek span of your most-used files and libraries.

9.7 SWAPPING

9.7.1 PRIMARY

Swapping - User Area Disk Activity (UADA)

Swapping is the best indication of how main storage is being utilized. Do not use Storage Commitment since it can be misleading (see Total Storage Commitment). The User Area Disk Activity (UADA) is the sum of the swaps in, and swaps out, and translated transfer loads and must be calculated by the user since SMF does not calculate it. The UADA reflects the system disk activity in the user area. If the average UADA is around 400 per minute, more main storage will help performance unless the swaps-out counter is zero and almost all of the UADA is due to translated transfer loads. BSC-B could cause this phenomenon to occur (see Translated Transfer Calls/Loads). In the 200-400 per minute range, more main storage may help reduce swapping. However, there may or may not be any perceivable (by the operators) response time improvement. The same is true for batch run time improvement. Rescheduling some jobs may help. If the average UADA is below 200 per minute, it is doubtful that more main storage would be helpful.

Swaps in will occur on all systems even if there is more than enough main storage to contain all of the user tasks. These swaps in are occurring because of system programs that execute in the user area. These system programs are loaded into main storage as needed and may create periods when there is not enough main storage to contain all the tasks, both user tasks and system programs. This may not show up only at the instant of the snapshot interval.

It is not valid to add the Prog Swaps and TWS (Task Work Space) Swaps from the task status section, and the Swaps of the System Work Space (SWS) from the storage-totals section, and compare that total to the sum of the swaps in and swaps out in the System Event Counters section. The reasons are:

1. When programs go to end of job, their Prog Swaps and TWS Swaps counts are lost.

2. System programs are continually being released from storage and others are being loaded or swapped into storage. These swaps are not reported anywhere except in the swaps in and swaps out.

3. The initial load of a program is counted in the Prog Swaps but is not counted in the swaps in.

9.7.2 SECONDARY

Storage Releases W/O or W/ Swap

Before discussing the L-1 to L-4 Storage Releases, some discussion of storage releases in general is appropriate. W/O (without) or W/ (with) swap indicates whether the main storage for a program was swapped out prior to releasing the storage for use by another program. W/O swap indicates that the program was not swapped out and therefore must be a reentrant or reusable system program (TTC/TTL). When the program is needed again it will be reloaded from a library. W/ swap indicates that the program was swapped out. When the program is needed again it will be swapped in. Use these counters to tell what kind of swapping is occurring. Don't be concerned with L-1 and L-2 releases but look at the L-3 and L-4 releases as indicated below. L-1 to L-4 indicate the level of desirability, L-1 being the most desirable and L-4 the least.

L-1 Storage Releases

Voluntary long wait. A long wait occurs when a program or the system knows that it will be a long time before the program will be required again. The most common situation for a program to go to a long wait is when it has sent a format to a display station and is waiting for the Enter key to be pressed at that display station.

L-2 Storage Releases

The program swapped out had a significantly lower priority than the program that was swapped into main storage. Priority can range from 64 to 255. A significant difference in priority is 32 or greater. For more information see the discussion on Priority in Other Information.

L-3 Storage Releases

The priority of the program swapped into main storage was slightly greater than or equal to the priority of the program that was swapped out. "Slightly greater than" means a priority difference is less than 32. (See the discussion on Priority.) If the majority of the programs you are running have the same priority, this will be the normal situation. If not, you are approaching nonproductive swapping.

L-4 Storage Releases

The program swapped out had a higher priority than the program that was swapped into main storage. The most common situation for an L-4 storage release occurs when a program with a higher priority is currently in main storage and wants a system resource currently owned by a program with a lower priority that is not in main storage. The program with the higher priority releases its storage to allow the program with the lower priority to be swapped in to run and free up the owned resource. If the program with the higher priority did not release its storage, the programs would interlock, since the program with the lower priority cannot free the resource when it is not in main storage. If the number of L-4 storage releases exceeds 20 per minute, additional main storage would help. Tests have shown that when this counter exceeded 20, the UADA also exceeded 400 per minute. Also consider rescheduling jobs to reduce system activity and thus improve performance.

9.8 SYSTEM PROGRAMS

9.8.1 PRIMARY

Translated Transfer Calls/Loads (TTC/TTL)

A Translated Transfer Call (TTC) is a call to a system program that executes in the user area of main storage. The Translated Transfer Load (TTL) is the disk operations necessary to load the system program into the user area. The majority of these system programs are reentrant or reusable which means that these programs do not have to be reloaded from disk every time they are executed. Thus there will be more calls than loads.

The ratio of calls to loads can be tracked. As the ratio goes down, that is, it approaches 1 to 1, the swapping activity increases (the system programs must be continually reloaded to execute). At higher ratios, for example 8 to 1, the swapping activity decreases since the system program called is likely to be in main storage. If your system is constantly showing ratios of less than 2 to 1, you could probably benefit from adding more main storage so more of the system programs would remain resident, thus reducing disk operations and improving the system's performance.

If you are running Batch BSC (Binary Synchronous Communication) or BSC-B as it appears on an SMF printout, be aware that some of the system programs supporting it are reloadable. This means that they must be reloaded every time they are executed. Every call to those modules will result in a load and the ratio will approach 1 to 1. More main storage will not help this ratio and system performance if you are running BSC-B. The most common way to use BSC-B is through the telecommunications specifications of RPG.

9.9 DATA BY TASK

9.9.1 SECONDARY

I/O and SEC Data By Task

Look at these counters on a per-task basis only. They can be found in the Task Status section of a Detail or All report. Some of these have already been discussed on a system basis.

This shows you the MSP usage for each task active at the time of the snapshot. It will indicate which tasks are using the MSP more than the others.

Res T-Outs

Res T-Outs are the number of resource time-outs that occurred on a per-task basis. The average time-out interval is 500 milliseconds. When a work station read operation is performed, the task will get it if it does not use more than the allotted resource time. For interactive jobs, this value could indicate a task that uses the system heavily and, as a result, may be experiencing slow response times. It may be necessary to change the program to improve the performance. Every time a resource time-out occurs, the task has its priority lowered by 4. For more information see the discussion on Priority in the Other Information section. Batch jobs will probably have many resource time-outs because they do not do work station read operations. Their priority will normally be V-LOW and will not change with the resource time-out since it will already be as low as it can get.

MSP T-Outs

This is the number of main storage processor time-outs on a per-task basis. It means that a task used more than 100 milliseconds of processor time in a 200-millisecond period. This is the amount of processing a task can have before the system flags it as being interruptible. If another task of equal or higher priority is ready, the system gives control to it and places the interrupted task in the queue of tasks ready to use the processor. If not, the task can continue processing. The main storage processor time-out value is reset every 200 milliseconds. Therefore, a task may never get a main storage processor time-out if it does not use more than the allotted time. And, as a result, may be experiencing slow response times. It would also typically show a higher MSP usage than other tasks not experiencing main storage processor time-outs. It may be necessary to change the program to improve the performance.

Disk 1-4 Reads, Scans, and Writes

These counters show the number of reads, scans and writes on each disk on a per-task basis. It will indicate which tasks are doing the most disk operations and on which disks they are being done.

Prntr Ops

Prntr ops indicates the number of printer operations or lines printed on a per task basis. If the task is spooling the printer output, this is the number of print requests intercepted into the spool file. Print requests include requests for paper movement as well as printing of data.

Wkstn Count

Wkstn count shows the number of Enter keys, function keys and command keys that were pressed from any work station attached to the task.

Rec-Waits

Rec waits are disk record waits on a per-task basis. It lists the number of times each task had to wait to read or write a record that was being updated by another task.

Job Steps

Job steps are the number of job steps initiated on a per-task basis (the number of executed // RUN statements in the procedure). This counter indicates how many job steps were called from a given procedure. You would have to add the values from all the intervals during which the task (procedure) was active. This could be done by using SMFDATA and a program used to print the totals for the task.

9.10 COMMUNICATIONS

9.10.1 PRIMARY

Communication Line 1-8

Use the communication usages to compare with other runs. Again, what changed to cause the increase or decrease? An important area of concern is line retries or errors on the line. SMF only reports the errors with the communication information on an ALL report. This means you must print the entire report or use SMFDATA and write a program to print out just those counters you want. See the SMF Guide for information on SMFDATA and the record layouts of the file created by SMFDATA.

Long response time on remote display stations could be caused by a number of factors. It could be the line itself. A 4800-BPS line transfers 600 bytes per second. If you are receiving 300 bytes of data from a remote display station and sending a 1200-byte format back to that display, the response time will include 2.5 seconds of data transfer over the line. This does not include any wait time that may occur while the data or the format is waiting to use the line. This should not be a significant factor unless you are sending large amounts of data over the line. The line itself can have only one set of data or format on the line at a time. It is a single thread resource and, as a result, queuing may occur at either or both ends of the line. This will increase the response time. One thing that can be done is to use Put Overrides and Erase Input Fields wherever possible in your formats. These will reduce the amount of data that must be sent over the communication line. However, the bottom line is still, "What do you find to be acceptable response time?".

It can be difficult to determine if the line is a bottleneck because the average amount of communication line usage may be misleading. This is because an average usage in a given minute of 25% may be 15 seconds of 100% line usage (when everyone is trying to use the line) and 45 seconds of 0% usage. The average does not really show a problem but there is one. If you suspect this is happening on your system, try using a smaller snapshot interval. Then examine each snapshot interval to determine if the problem exists. The number of transmitting and receiving 150 bytes versus 1500 bytes will have a considerable effect on response time.

On the other hand, batch jobs such as MSRJE should drive the line at more than 80%, assuming the line speed is 9600 BPS or less, because the system should be able to supply 1000 bytes of data per second to transmit (80% of 1200 bytes is 960 bytes). These numbers depend on how busy the rest of the system is, the number of MSRJE sessions per line, and the number of communication lines running batch-type jobs. The busier the system, the more MSRJE sessions, and the more lines running batch jobs all contribute to reducing the throughput. You may improve line usage by increasing the pacing count for SNA subsystems or by sending larger blocks of data for BSC subsystems. For example, changing the pacing count for MSRJE from 2 to 7 may improve the performance. However, using higher pacing counts or sending larger blocks of data will tie up the line without improving response time if the job is interactive. Every subsystem has a different way to specify its blocking. See the appropriate reference manual for your communications subsystem.

10.0 TUNING GUIDELINES

The question is often asked:

"What are the "correct" values for the various SMF utilizations and counters?".

The answer is, those values do not exist. This section offers some guidelines but be aware that they are just that -- GUIDELINES, NOT RULES. You should use the range of values that you determine as good values for your system. The System/36 is a sophisticated machine that can have up to seven processors. Some systems will perform well with a given set of values and other systems will not perform as well with the same set of values. The difference is the applications being run. It is recommended that if SMF is run periodically when your system is performing well, you will be able to establish a range of values that will represent good values for your system. You should also measure the response times of your various job mixes when your system is performing well. These will be useful if you later experience a performance problem. Also keep some of these SMF runs on file in case you need to make detailed comparisons. Remember to compare "like" SMF runs that have the same job mixes.

The guidelines should help the majority of System/36 users. However, some changes may be made based on the guidelines and although there will be some improvement, it may not be perceivable by the operators. There may also be systems that will not exceed any of the guidelines but will still be performing poorly. In these cases the applications may be the cause of the problem.

Some guidelines to consider before doing any tuning on your system are:

1. Is your system response satisfactory?

2. What do you hope to gain from tuning?

3. Do you have enough information collected to know what to tune?

4. Have you evaluated the alternatives to tuning?

5. Are you prepared to live with your conclusions?

10.1 PRIMARY AND SECONDARY: UTILIZATIONS AND COUNTERS

The preceding sections contained information on what is considered to be the primary and secondary utilizations and counters. There is also information on some counters that can be used as an indication of work change.

The following chart is a guide to using the SMF output. Refer to these figures as you read about the utilizations and counters.

At first glance, the list of utilizations and counters looks very simple and quite ordinary--in fact there is nothing listed that should come as any great surprise to even a novice user. What is important, however, is to understand the real value of each of these performance indicators.

SMF Utilizations and Counters Usage Chart

Significant utilizations and counters	Alarm Levels	
	Warning	Action
Primary		
Main Storage Processor	60%	80%
Control Storage Processor	65%	85%
Disk Usage (disks 1-4)	60%	85%
Communication Lines	60%	80%
Translated Calls to Translated Load Ratio	2 to 1	1.5 to 1
Disk Record Waits	Dependent	Dependent
Disk Seek Ops GT 1/3 Disk (disks 1-4)	Dependent	Dependent
User Area Disk Activity is the total of:	200	400
. Translated Transfer Loads	Dependent	Dependent
. Swaps In	Dependent	Dependent
. Swaps Out	Dependent	Dependent
Secondary		
Work station Controller	60%	80%
Spool Extents Allocated	1	2
Task Work Area Extents	1	1
Storage Release with Swap (L3-L4)	Dependent	Dependent
Storage Release without Swap (L3-L4)	Dependent	Dependent
Disk 1-4 Reads, Scans, Writes, Seeks	Dependent	Dependent
Printer Ops	Dependent	Dependent
Work Station Ops	Dependent	Dependent
Job Step Initiations	Dependent	Dependent
Resource Timeouts	Dependent	Dependent
Main Storage Processor Timeouts	Dependent	Dependent

Notes:

1. Warning means you should make plans to solve a potential performance problem.

2. Action means you should implement your plans.

3. Dependent means the guideline value for that counter or utilization cannot be given because of its dependency on other variables.

4. Communications:

   - Both interactive and batch communication jobs are dependent on overall system activity.

   - Communications lines:

     - Batch jobs should drive the line at 80% or more if the line is 9600 BPS or less.

     - Interactive jobs are application dependent.

5. UADA - user area disk activity, the total of swaps in, swaps out, and translated transfer loads.

6. The number of work station operations versus the number of program swaps by job can directly affect the performance of the system. A low ratio, 1 to 1 or 1 to 2, is an indication that memory or redistribution of your workload may improve system performance.

7. The maximum number of disk operations per second is between 25 to 30 per disk spindle. This number is calculated by taking the total of the disk read, write, and scans. Then divide that number by the total time from the SMF reporting period.

8. The ratio between Translated Transfer Calls and Translated Transfer Loads should be greater than 2 to 1. If it is less than that ratio, additional main storage is probably needed. If the total of the swaps in, swaps out, and Translated Transfer Loads (UADA) are greater than 200 per minute you need to begin thinking about adding additional memory.

9. The primary and secondary utilizations and counters shown are only a part of the total number of counters that are used. For difficult problems you may need to use the other counters. You may also need to isolate a single task to see how the system responds. There are also many programming considerations that affect how the system utilizes resources.

10. Disk Seek GT 1/3 Disk: reduce as much as possible and try to get below the 10-15% range.

10.2 STANDARD PRACTICES AND PROCEDURES

The following items are standard practices and procedures for tuning:

- Establish a range of 'standard' system values:
  - Utilization values
  - Counter values
  - Include peak workload times

  Note: Because of different workloads and variables on your individual system, these values will not be completely standard from system to system.

- Establish a range of acceptable performance values:
  - Response times
  - Batch times

- Keep a history of catalogs by location:
  - Need for comparisons
  - Run catalogs on a regular basis

- Keep a history of SMF runs:
  - Run SMF before <u>any</u> hardware or software change.
  - Run SMF on a regular basis.
  - Summarize results - plot the counters you are tracking.
  - Keep reports on file or on diskette (SMFDATA version).

- <u>Never</u> tune a system based on one SMF run:
  - Include peak workload times.
  - Use a complete work cycle.

A large number of the performance problems are file placement problems. By keeping a history of your catalogs (VTOCs) by location you can compare the current catalog with a previous one when performance was good. If one or more of your key files has moved, it could explain the slow down. Make the comparison before running SMF.

We also recommend establishing what you are going to use as measurable criteria so that when your operators tell you the system is slow, you will be able to verify whether it is slower and how much slower. Anyone who has sat in front of a display station waiting for the input inhibited light to go out knows that perception can be a problem. Common responses to questions on the length of response time range from "one minute" to "forever." Measure the responses and compare these times to the ones you made when the system was performing well. Assuming it is slower, find out as much about the slowdown as possible. Is it common to all the operators or just the order entry operators? By narrowing down the problem, you will spend less time solving it. If you decide to run SMF, determine how long to run it and what snapshot interval to use. Recommended is the period of time that you are experiencing the problem. Use a one-minute interval. A ten-second snapshot interval may be appropriate (rarely) if you can get several snapshot intervals over one display station response. For example, if you are experiencing 40-second response times, you should get at least three snapshots by using the ten-second snapshot interval. You can then look at the task status section of SMF and try to determine what is happening.

Batch throughput can also be used as measurable criteria. Batch times in the history file can be printed or viewed on a display using the HISTORY procedure. If batch throughput is used, be sure to compare the same job mixes and be aware that batch times will probably vary much more than response times. Whatever is used as your measurable criteria, make sure that it is repeatable. Also make several measurements over a period of time (days, weeks) to insure the accuracy of the data.

When running SMF, use one minute as the snapshot interval, collect communications data if you have communications, and collect I/O (Input/Output) and SEC (System Event Counters) data by task. If you collect communications data, make sure you enter the correct line speeds. Since SMF cannot sense the line speed, it relies on you for the information. If you have not yet done any file placement or you need to do more file placement, collect user and system data by file. When you start SMF, change the SMF data collection file name from SMF.LOG to SMxxxxxx, where xxxxxx is the current date. This will help you to identify when the run was made since SMF will print the data collection file name at the top of each page of the report.

NEVER tune your system based on one SMF run. Make several runs before tuning your system. Make the runs using your complete work cycle and workload, including your peak workload times. Then use all of the data gathered in tuning your system. You should run SMF before making any hardware or software changes to your system. If you have problems after the changes, you will then have something to compare against to help you determine why the change had the effect it did.

10.3 IF THE SYSTEM RUNS WITHOUT PROBLEMS

Here are suggestions for using SMF:

1. Run SMF with a long interval for the entire day.

2. Print out the SUMMARY for the peak period.

3. Save the SMF printout for later reference.

The goals are to:

- Establish a range of or standard system values

  - Peak workload time
  - Utilization
  - Counter values

- Keep a history of SMF runs

  - Run SMF before any hardware or software changes.
  - Summarize the results on one chart and file it along with a description of what was executing.
  - Keep the SMF file on diskette and file it with the summary.
  - Keep a history of catalogs by location so you can make comparisons.

10.4 IF PERFORMANCE PROBLEMS OCCUR

Here are suggestions for using SMF:

1. Run SMF with a long interval (for example, 5 min) for the entire day. Average the totals.

2. Run SMF with a 60-second interval spanning the problem period.

3. Run SMF with a shorter interval (for example, 10-second, if response time is 30-40 seconds) spanning the problem period. (The summary report shows the maximum counters and the time at which the maximum occurred.)

4. Look for values that differ greatly from the typical runs.

5. Use the DETAIL or ALL print option to list the most important parts of the collected data.

6. Be sure to collect I/O and System Event Counter data by task.

10.4.1 STEPS IN A PERFORMANCE PROBLEM ANALYSIS

Performance problem analysis is divided into five steps, each of which is described in more detail in this section. Steps 1 and 2 are necessary preparation for a disciplined analysis: understanding the problem to be solved and gathering data to work with. Steps 3, 4, and 5 represent searching for the cause and solution to the problem. Each of these three steps requires more time, more effort, and/or more significant decisions than the preceding step: basic performance factors, applicable to all system control programs, are reviewed in step 3; the use of system resources and specific potential bottlenecks are investigated in step 4; doing a sensitivity analysis is discussed in step 5; and solutions that are considered last resorts are included in step 6. The amount of time spent on each step depends on how quickly the problem is solved or how quickly the solutions investigated in a specific step are rejected. (Note: Carefully review steps 1 and 3; experience indicates that these steps are most often ignored in practice.)

Steps

1. Describe the performance problem in terms of the objectives that are not being met.

   a. User-oriented objectives
   b. System-oriented objectives

2. Take a basic set of measurements.

3. Review basic performance factors.

   a. Hardware configuration
   b. I/O resource usage
   c. Control of users that monopolize resources
   d. Resource contention
   e. Swapping characteristics

   REMEASURE and REEVALUATE:

   Problem changed? Go to step 1.
   Problem solved? Monitor performance.

4. Identify potential bottlenecks in the system and assess the impact of the bottleneck.

   a. System-oriented problems--resource management

      1. Processors
      2. I/O resources
      3. Swapping
      4. Translated calls to translated loads ratio
      5. Communications lines speed

   b. User-oriented problems--workload management

5. Perform a sensitivity analysis (that is, validate suspected bottlenecks and quantify their relative impact).

   REMEASURE and REEVALUATE:

   Problem changed? Go to step 1.
   Problem solved? Monitor performance.

6. Investigate alternative solutions.

   a. Modify performance objectives
   b. Reconfigure hardware
   c. Obtain more hardware
   d. Modify system support program (SSP) parameters

The following descriptions of the steps do not reflect their interactive nature. The process of remeasuring and reevaluating (steps 3 and 5) is necessary every time potential solutions are identified and implemented. The possible results are:

- The problem is solved and no new problems are identified; the analysis is complete.

- The immediate problem being addressed is solved but another problem becomes apparent. This is not unusual, because one bottleneck often disguises another. In this event, the analysis begins again with step 1.

- The problem remains and further investigation is required; the analysis continues.

10.4.1.1 STEP 1. DESCRIBE THE PERFORMANCE PROBLEM

The purpose of this step is to be as specific as possible in describing the performance problem(s) in terms of the objectives that are not being met. The problem description should include the extent of the problem (which will help you predict the effectiveness of possible solutions) and the specific type of work experiencing the problem (which will help you identify possible trade-offs). For example, "Fifty percent, as opposed to the desired ninety percent, of the trivial interactive transactions during peak hours are receiving three-second response time" is a good problem description; "the system is sluggish" is not.

Problem statements will be specific if your performance objectives were specific. Two types of objectives, based on two types of measurements, are defined.

- User-oriented objectives -- response time for a specific class of transactions or turnaround time for a specific application or class of jobs.

  -and-

- System-oriented objectives -- batch throughput, interactive transaction rate, or number of concurrent terminal users.

Note which type of objective is not being met in your system--it will affect the approach taken to identify bottlenecks, as described in step 4. If both user-oriented objectives and system-oriented objectives are not being met, investigate the system-oriented problems first.

Approach

If you are dissatisfied with the performance of your system, you should ask yourself, "What am I trying to achieve, faster interactive response time, more batch throughput, or improved print performance?"

Next, think about when the problem first started:

- Did you make any changes to the system about that time, such as adding additional devices, new applications, or new programs?

- Did the workload increase?

The answers to these questions may give you a clue as to where to look for the problem. Look for symptoms of performance in:

- Display station response time during interactive jobs.

- Printer throughput (the number of lines printed within a given time period).

- Batch throughput (the number of batch jobs processed within a given time period).

As mentioned earlier, the tool offers many ways to reduce and report the collected data.

Just reviewing the summary reports will give the overall perspective required to do a good job of using SMF.

If some jobs and/or time periods look suspect, detail listings can be run on demand for the jobs and/or time periods in question.

Often the place to start is to:

- Determine what the largest component of response time is.

10.4.1.2 STEP 2. TAKE A BASIC SET OF MEASUREMENTS

The purpose of the basic set of measurements is to provide enough data to focus on specific potential problem areas in the system. "Enough" is defined by:

- The type of information you gather.
- The number of samples you take.

For example, you could compare your performance measurements against your performance objectives specified on performance tracking worksheets and focus attention on any performance objectives that are not being met.

Taking Basic Measurements

Basic measurements to take are:

- System-wide resource usage, to help identify those resources that are under- or over-used. Use the SMF summary report option.
- Resource usage by task.

In addition, SMF provides data on system and task statuses, data by task, and I/O activities, data that is frequently needed for detailed performance analysis.

Whether to use additional information, beyond what is gathered by the basic set of measurements, can be determined by the conclusions reached from reviewing the basic measurements.

Taking Sample Measurements

It is important to have a multiple number of samples--for the times when performance problems are occurring and for the times when performance objectives are being met--and to document the workload at the time of each sample. This is necessary to judge whether the reported value of any specific measurement indicates a possible problem area in the system. Recognition of values that might indicate a problem is usually based on the fact that a number is high or low compared to one of the following:

- A value usually observed at your installation--for example, processor use observed under similar workload conditions or the average processor use over a period of time. If multiple samples have been gathered, or if SMF is executed on a regular basis, you will be able to identify usual values for your installation.

- A target value obtained from experience-based guidelines, comparison with similar installations, or similar sources. This newsletter provides target values for many of the measurements used in the suggested analysis. (See Analysis of SMF Output.)

In either case--usual values or target values--the value should not become a goal. It should be used only to determine if further investigation of a measurement is indicated. Circumstances might have changed sufficiently for a "usual" value to no longer be valid, and target values are not necessarily appropriate for your installation. In general, usual values are a more effective yardstick than target values.

Once a basic set of measurements has been taken, a major problem is determining which of those measurements to focus on.

10.4.1.3 STEP 3. REVIEW BASIC PERFORMANCE FACTORS

The objective of this step is to ensure that basic tuning factors--those that are applicable to all systems--have been addressed before spending a lot of time and effort analyzing applications specifically. The factors that should be addressed at this point have these characteristics:

- The effect (positive or negative) of varying the factor is fairly predictable.
- The installation can easily exercise control over the factor, for example, by using installation procedures or external system interfaces such as initialization parameters.

Based on experience with existing systems, the following factors are important to consider at this stage:

- How adequate is the hardware configuration in light of workload requirements?
- Are I/O resources being used effectively and efficiently, such as by using file placement and monitoring the configuration of communications lines and system values?
- Swapping characteristics.
- Are there controls to prevent users from monopolizing system resources?
- Is there any resource contention?

These are the same factors that should be addressed before system configuration and initialization.

10.4.1.4 STEP 4. IDENTIFY POTENTIAL BOTTLENECKS

This step and the next represent the body of the analysis. The approach to identify and correct bottlenecks in the system depends on the type of performance problem being experienced--whether system-oriented objectives or user-oriented objectives are not being met (see step 1). There are two major areas of control in the system:

- Resource management
- Workload management

Although they overlap, the:

1. Initial focus is on resources used for both system-oriented and user-oriented problems.

2. Secondary focus is on workload management for user-oriented performance problems.

Analyzing System-Oriented Problems

In the case of a system-oriented problem, the goal is to identify the resources that could be used more effectively and to improve their use. The resources of the system can be reduced to five basic resources: the processors, swapping, communications lines, control units, and devices. (Swapping, although considered separately, is really an interrelated combination of main storage, communications lines, control units, and devices.) Bottlenecks that affect throughput will affect one (or more) of these resources and be reflected in measurements of its use. The initial focus of the analysis is determined by answering the question, "Is the system waiting too much?"

Analysing User-Oriented Problems

In the case of a user-oriented problem, the analysis has already focused on a specific application, class of batch jobs, or trivial and nontrivial interactive transactions. The goal is to determine why the work is being delayed--to what resource(s) it is not receiving access--and to increase its access to the needed resource(s). Increasing access to resources by the work in question implies either improved resource management or trade-offs for other work in the system (that is, workload management). If resource management cannot be improved, the analysis must focus on task management.

There is no firm order in which to investigate the possible causes--if a specific resource is critical in your installation, investigate causes related to it first. For example, if storage is critical, start by checking excessive swapping and the translated calls to translated loads ratio. However, you cannot always focus on one cause until other causes have been eliminated. For example, if the work is in storage and is not obtaining sufficient access to the processor, you cannot assume the work is being denied access to the processor unless you know it is not waiting due to I/O or a record lock contention.

When the causes of wait time are identified, initial solutions can involve increasing the access to the resources causing the delay, via specifying preference of the work in question over other work in the system (that is, by workload management). Means of doing this include specifying or changing execution priority, job queue priority, the performance objective, and for batch the number of jobs that is allowed to execute concurrently. Use of such controls implies trade-offs for other work in the system. If such trade-offs cannot be tolerated, the direction of the analysis must shift to making optimal use of the resource(s) in question--or setting priority for the objectives of different types of work, modifying objectives, or considering additional hardware.

10.4.1.5 STEP 5. PERFORM A SENSITIVITY ANALYSIS

After completing step 4, you should have identified one or more bottlenecks that might be responsible for the performance problem. Further analysis should reveal that the suspected bottleneck:

- Has no effect on the performance problem being investigated, or

- Has an effect on the performance problem, but the possible benefit of reducing or eliminating the bottleneck does not justify the tuning effort required to address the bottleneck, or

- Has a significant effect on the performance problem and justifies the runing effort to reduce it.

To determine which of the preceding cases is true, you can use a technique called <u>sensitivity analysis</u> to analyze the suspected bottleneck.

Sensitivity analysis is a graphic display technique that maps the objectives being met against a measurement that reflects a suspected bottleneck. The resulting curve (or lack of one) illustrates the relationship (or lack of a relationship) between the objectives and the suspected bottleneck.

Measurement Sampling

To obtain useful measurements, gather samples at several different times during a day. These sampling periods must be numerous enough to provide a variety of values and measurements that allow any deviations caused by the bottleneck to stand out. Also, short sampling periods provide a quick way to get many snapshots of a specific workload without collecting long workload histories. While these values should be closely related in terms of sample time, you don't have to synchronize them down to the smallest fraction of a second. The values will still be reasonable when the bottleneck is significant (greater than 5% of elapsed time).

For example, when analyzing response time, use a sample period of five minutes or less for online interactive work. When analyzing throughput, measure throughput in small units of time such as 15 minutes or a half hour. When plotting throughput, plot a count of steps or transactions executed in that period (rather than response time) against the use of different resources. It does not matter if the steps or jobs do not terminate within the sampling period or interval. Their presence if reflected in the system loads that the throughput is being evaluated against.

Note: When analyzing throughput for long-running batch steps, make the sample monitoring period proportionately longer.

Sensitivity analysis allows you to graphically plot response time and throughput against the suspected channel bottleneck. From the graph you can determine:

- If a correlation exists between the degradation and device usage.

  -and-

- If the bottleneck justifies taking corrective action.

10.4.1.6 STEP 6. REVIEW ALTERNATIVE SOLUTIONS

When performance problems are not solved after following steps 4 and 5, you should review such alternative solutions as:

- Modifying performance objectives
- Reconfiguring system (including modifying system values)
- Obtaining more hardware
- Redesigning application

However, there is overlap between the steps of a performance analysis as outlined here.

For example, any of these "alternative solutions" might have been identified in step 4 as the most effective solution to a bottleneck. The question of additional hardware resources might have been addressed in step 3, in reviewing the reasonableness of the current hardware configuration. The key to exploring these "alternative solutions" at any point in the analysis is the ability to predict the probability of meeting your objectives by means of other solutions--workload balancing, increasing resource use, decreasing unproductive resource use, and so on. Detailed performance objectives will allow you to identify minor changes to the objectives' priorities as solutions to resource bottlenecks identified in step 4.

System modifications should be avoided. Before considering any system modification--or any solution--ensure that it addresses a bottleneck identified in your system. There are no all-embracing solutions among the performance suggestions, despite the reputation of some. Each solution must be considered in light of your installation and its specific bottlenecks.

10.5 USER-ORIENTED GUIDELINES

Introduction

Good system performance is achieved by effectively managing basic system resources:

- Main storage
- Disk
- Main storage processor (MSP) and control storage processor (CSP)
- Work station communications

Key performance indicators are:

- Throughput (that is, transactions per hour)
- Response time (in seconds)
- Transaction characteristics:
  - CPU (MSP and CSP) transaction
  - Disk I/O transaction
- Resource utilization:
  - MSP and CSP (system, interactive, and batch)
  - Disk arms
  - Memory (that is, main storage)
  - Communications lines
- Number of active users

The discussions in this section assume that the reader is familiar with the System/36 and how it works.

This section contains discussions of things you might want to change to improve performance on your System/36. This is not all-inclusive but lists several things to look at. Also note that some of the items have previously been discussed. Suggested areas where performance can be affected are the folowing.

User-Controlled Components

The System/36 allows you to assume an important role in managing and scheduling jobs. You can affect:

- How your programs use main storage by using different processing priorities
- The order in which jobs are to be processed by the system by using different priorities in the job queue
- The number of concurrently executing user programs
    - Workload (wait in OCL)

Program Execution Environment

To optimize system performance, consider:

- Number of concurrently executing user programs
- User program size
- Record lengths of opened files
- Data and index blocking factors
- Number of alternative indexes created per file

Program Characteristics

Program performance depends on:

- Whether the initial program has overlays
- Execution priority
- Number of files accessed
- File placement
- Buffer size
- Whether buffers are appended
- Whether the storage index is used
- Other system activity

10.5.1 USER RESPONSE TIME

For the purpose of this discussion, response time will be defined as shown below:

Queue Time	Service Time
Response Time	

User response time on System/36 depends on the following:

- Wait queue time (that is, the time spent on the active program list queue)
- Service time (that is, the time spent to complete the transaction)
- Communications line time
- Other system activity:
  - Relative priority
  - Disk input/output operations
  - Storage commitment

<u>Objective</u>

- Set transaction guidelines:
  - Processing per transaction
- Use time slice efficiently:
  - Minimize service time
  - Minimize queueing time

10.5.1.1 WAIT QUEUE TIME

Active Program List Queue

A task is placed on the active program list queue:

- When it is first loaded/initiated.
- By a "long wait" status.
- When it exceeds its resource utilization time interval (R-Time), its priority is adjusted and then placed on the ready task list queue.

Each task receives a new time slice (400 ms) after being placed on the active program list queue.

Ready Task List Queue

A task is placed on the ready task list queue:

- When:
  - It is loaded/initiated or and it has all the needed resources to execute
  - Its execution priority has been adjusted as a result of exceeding its resorce time interval (R-time)
- When it exceeds its main storage processor time interval (MSP-Time)

  Each task receives a new MSP-Time interval (100 ms) after being placed on the ready task list queue, but not a new R-Time.
- When it is preempted by a significantly higher priority task (that is, 32 priority levels higher).

Note:

For performance, however, queuing is an effect, not a cause.

10.5.1.1.1 ACTIVE PROGRAM LIST QUEUE

A "long wait" status

- The task gives up the MSP and is placed on the active program list queue regardless of how quickly the user responds to the request.

- Second to service time, user response time can be most significant if the application requires a rapid succession of responses and the system is heavily loaded.

Example:

Assume the user requires three display screens (menus with options) in succession in order to determine the type of work to be accomplished.

Consideration:

- The task requires three time slices in order to display three screens, because after each work station write/read operation (that is, Active-to-Long Wait (AW) condition), the task is placed on the active program list queue and receives a new time slice. (No real work has been accomplished.)

A task exceeds its resource utilization time interval (R-Time)

A "long service time" (that is, a transaction that requires more than one time slice to complete) also affects user response time because of the possibility of the task being queued and swapped after each R-Time exhaustion.

Case 2

Assume each transaction for a task requires approximately 21 physical disk operations.

Consideration:

- A minimum of two time slices would be required (excluding execution time spent in the MSP and CSP) to complete one transaction, because only 16 physical disk operations can be completed before the task's R-Time (400 ms) is exceeded:

```
Formula = (16 x 3) x 8.192 ms = 406 ms).
           |    |         |
           |    |         ---System time unit
           |    -----------I/O timer unit factor for disk
           ---------------Number of physical disk operations
```

10.5.1.1.2 READY TASK LIST QUEUE

A task exceeds its main storage processor time interval (MSP-Time)

In order to limit the influence of a processor-bound task, if one task has been executing consecutively for more than 100 MS (main storage time-out, checked every 200 MS), it can be preempted by a task of equal or higher priority. User response time is affected because of the possibility of the task being queued and swapped after each MSP-Time exhaustion.

Considerations:

- Active-to-Ready (AR). The task gives up the MSP because it exceeds its MSP-Time and there are other tasks of equal or higher priority waiting in the ready task list queue. The task is put into the ready task list queue in FIFO order within its priority group.

- If the task has remaining R-Time as a result of the MSP-Time being exceeded, it is placed on the ready task list queue and receives a new MSP-Time. Otherwise:

  1. It is placed on the active program list queue

  2. Its priority is adjusted

  3. It receives a new time-slice (500 ms average)

10.5.1.2 SERVICE TIME

Service time includes six major components:

1. Swaps. A program's modules and/or task work spaces (TWSs) are required; however, they are not in main storage.

2. Disk accesses are required to perform any application.

3. Execution of program code of all components. May have the smallest impact on service time unless there are many complex iterations per transaction.

4. Use of system services. Includes items like the command processor services routines, error message handling, and so on. Can be big contributors to service time.

5. Work station file handling. Remote work stations performing multiple put operations using large numbers of fields and literals, and so on, all contribute to service time.

6. Index maintenance. Could be significant when a transaction causes many alternative indexes to be maintained, especially opened alternative indexes, which require immediate maintenance.

10.5.1.3 SUMMARY

The largest contributor to service time "tends to be" the number of physical I/O operations that are performed.

Minimizing service time:

- Is the single most important performance element related to interactive program design objectives.

- Has a very positive effect on user response time, not only because a task takes less time to execute, but because there will be less queueing for all system resources.

The key to finding and fixing I/O related performance problems is response time: the length of time it take to complete an I/O operation. Response time can have a dramatic effect on performance, particularly with online and interactive products such as SEU, DEF, QUERY/36, and DW/36. The following discussion addresses the various response time elements and the factors that can affect them.

10.5.2 FOCUSING ON PROCESSOR-BOUND USERS

When experiencing processor constraints, one useful approach is to identify and focus on those users who are processor-bound; that is, which use an above average amount of processor (MSP and CSP) time.

10.5.2.1 CONTROLLING DOMINANT USERS

A common performance-related axiom is that "10% of the users use 90% of the system." By obtaining SMF reports on the amounts of I/O, main storage, and processor service of end-users, you can identify the dominant users at your installation.

Controlling dominant users depends upon:

- Who is the user,
- What is the user's priority
- Similar installation-dependent factors.

Some examples are:

- If more than one batch job uses significant amounts of main storage, you can prevent the jobs from executing concurrently by assigning and placing them on the job queue.
- Specify LOW execution priority for a group of jobs with heavy processor (MSP and/or CSP) demands.

10.5.3 OPERATIONAL BOTTLENECKS

Operational bottlenecks can be a significant source of system degradation, but they are often ignored because they are not usually reflected in system measurements. Before doing detailed performance analysis, you should check for operational bottlenecks. If they exist, but are not detected, the analysis can be long and futile.

Avoiding and Recognizing Operational Bottlenecks

Two important ways of avoid operational bottlenecks are:

- Educating operators
- Involving the operations staff early in the installation planning

Of these two ways, educating operators is the most effective means of avoiding operational problems. For example: On System/36, operators have a high degree of control over the system, and their actions can seriously impact performance. If the operator starts seven batch jobs via the EVOKE OCL statement, all seven try to run. Thus, to make the proper operational decisions for specific circumstances, operators must understand basic SSP concepts and architecture.

When operational bottlenecks do occur, they are often not obvious, and system measurements generally do not reflect them. Thus, the best way to recognize common operational bottlenecks is by being aware of, and observing, them. Some common operational bottlenecks to watch for are over-initiation, forms mounting, shift changes, reply delays, and volume mounting for diskettes and tapes.

Handling Operational Bottlenecks

Solutions to operational bottlenecks are often "easier said then done." For example, if delays occur in volume mounting or responding to console messages or messages from users, an obvious solution is to mount volumes and reply quickly. But implementing this solution could require more than a suggestion or an edict to the operators. To determine the cause of the delay, you could review the adequacy of the operations staff or the effectiveness of their procedures, such as excessive paperwork required of operators.

The following topics describe how to handle some operational problems observed in existing systems, such as:

- Number of batch jobs executing concurrently
- Forms mounting
- Shift changes
- Input and output delays
- Reply delays
- Tape or diskette volume mounting

Number of batch jobs executing concurrently ...

System and application programmers should also educate the operators on the reasons for the recommended values; such education can help them withstand pressure from priority users complaining about their work. It has been observed that when a user complains about a job, the operator often responds by starting another batch job, often resulting in further system degradation.

Forms Mounting ...

Potential problems in this area include jammed or empty printers and operators not responding to output-related console messages. To help avoid operational bottlenecks in forms mounting, reconsider the criteria for special forms, and use the FORMS OCL statement. The FORMNO parameter on the FORMS OCL statement specifies the forms number of the printer forms to be used for the printed output from the display station session.

If a forms number is specified, the system prompts the operator controlling the printer to install the forms with the specified forms number in the printer if the specified forms are not already installed.

Shift Changes ...

At some installations, operators quiesce the system to finish work on one shift before switching to the next shift. This action can waste many system resources.

Input and Output Delays ...

SMF reports do not include delays caused by entering work into the system and distributing output. Watch for this problem when complaints of slow turnaround time occur predominately for areas under control of remote operational groups and jobs. Check the HISTORY log for clues.

Reply Delays ...

Besides ensuring that console messages receive responses quickly, you should examine and rewrite applications that issue an excessive number of console messages or prompts. Old applications, written for earlier systems, often depend unnecessarily on the operator (for example, to supply time-of-day information).

Volume Mounting ...

Besides asking operators to respond quickly to mount requests, you should consider having controls for volume mounting.

For example, you can examine the job mix and decide to run large diskette and tape jobs on an off-shift.

10.5.4 DISK

If the largest response time component is the disk component (and quite frequently this is the case), the next step is to determine the makeup of the physical disk I/O per transaction.

This information leads to determining whether the physical disk I/O is caused by either or both:

- User application
- Memory overcommitment

Another factor that can influence the disk component of response is the average disk arm utilization. High disk arm utilization causes additional waiting.

Now, the question becomes one of reducing disk I/Os by any or all of the following:

- Adding memory
- Improving the application design
- Spreading the existing disk I/O accesses across more disk arms to reduce the effects of disk arm utilization

The Concept of I/O Contention

Many problems reported as poor performance turn out to have nothing to do with the SSP. Approximately 75% of the problems reported can be traced to some kind of I/O contention. Channel loading, control unit or device contention, data set placement (that is, files, folders, and libraries), high swapping active, and file sharing are the major culprits. As a general tuning rule, unless it is obvious that the problem is somewhere else, examine the disk I/O first.

<u>Arm Contention</u>

Arm contention is caused by two or more active data sets on the same volume and/or long seek distances. The effects of arm contention can appear in queue time and service time. Service time is increased when the arm is required to move long distances from one data set to the next. Generally, any I/O request with a seek distance greater than 1/3 of the disk drive should be examined.

Queue time can also be affected by arm contention. The increase in service time when long seeks are present increases the device busy condition and, therefore, increases the probability that the request will be queued.

Arm contention can also be caused by too many active data sets on a disk drive. This condition usually shows up as high device busy (high queue time) but reasonable service time.

In any case, excessive arm contention indicates some sort of data set placement activity is needed, except for folders (placement of folders can not be specified). It may require moving high activity data sets to low use disk drives or reallocating data sets closer to one another to reduce arm movement.

Another condition that can cause excessive service times is disk drives with large VTOCs that are searched frequently. Generally, this type of disk drive has many small data sets. This condition may require moving some data sets to a different volume to reduce VTOC search time. The performance exposure here is not necessarily to the jobs accessing data on this disk drive but to other more critical applications that cannot access their data because the channel and control unit are busy doing a VTOC search (scan).

A similar condition can occur with large program libraries and their directory searches. Situations like this may require splitting the library across several disk drives.

Reducing Disk I/O

Disk I/O time can be reduced by:

1. Using the DBLOCK and IBLOCK parameters on the // FILE statement to change the blocking of data and index respectively.
2. Minimizing disk seek ops greater than 1/3 disk.
3. Drive 1 should be the least utilized (system libraries reside there).
4. Using multiple small libraries instead of one large library:
   - Less queueing for a library
   - Shorter disk scan of the directory for specified member
5. There are other times that you will need to redistribute your files and libraries to obtain balancing. When you:
   - Add memory to your system, you will change your swapping rate and reduce the disk A1 usage. You may need to move files and/or libraries on A1.
   - Add another disk drive, you will need to move files and libraries to that disk.
   - Add or revise applications, you will probably need to redistribute your files and libraries since it will change the usage on the disks.
   - Make changes in your work pattern, such as rescheduling some batch jobs, you will probably need to redistribute your files and libraries to balance disk usage (jobs rescheduled).
6. Eliminating unnecessary swapping operations by deleting unopened alternative indexes that you do not use frequently and rebuilding them when you need to use them.
7. Reorganizing high usage overlaid or segmented programs to reduce the jumping around from one part of the program to another (reduce the number of swaps).
8. Reducing the number of external calls in a job/program, such as moving subprograms into the main program.
9. Adding more disk arms. This will reduce the average disk arm utilization and thus improve average response time.
10. Adding memory to the system (if possible):
    - Can reduce the average disk I/O per transaction
    - May reduce the disk I/O per second rate if the increased throughput does not make up the differences, which means lower disk arm utilization and less waiting (that is, making the disk I/O that does occur more productive).

Time spent waiting for the disk can be reduced by:

1 Reducing the number of disk I/Os in other jobs;

2 Increasing the number of disk arms by adding additional disk devices (if possible).

Record lock wait time can be reduced by:

1 Redesigning the application to not hold the lock as long

2 Avoiding the use of control records or files.

10.5.5 MEMORY

If the largest response time component is the memory component, the next step is to determine if the memory constraint is caused by either or both:

- User application
- Memory overcommitment

There is no precise formula to predict the effect that the following items will have on memory swapping.

- More memory should reduce swapping.
- More work stations should increase swapping.
- New applications should increase swapping.

Adding more memory can have very positive effects on system performance and user response time. More memory:

- Results in reducing the MSP time and CSP time per transaction
- Can increase the throughput level that the system will normally run at (because more processor and disk resources are available and users can enter transactions quicker, more frequently)

The net result is improved average response time and/or increased throughput.

Often, experiments can be done on an existing system to understand better what to expect when more memory is added to the system. Some things to try include:

- Suspend batch work, resulting in less contention for resources.
- Cut the number of active users in half. This should also help stabilize the interactive environment.

While you are performing these experiments, you will need to measure performance with the SMF. This is because you need to observe and understand the changes and/or trends taking place in the various operating environments.

10.5.6 JOB PARAMETERS

Job Scheduling

An effective way to improve performance of the system is to reschedule jobs:

1. Unrestricted evoking of batch jobs can cause performance to degrade. One question to ask is whether or not a particular job is critical. In other words, could the job be run at some other time and still satisfy the requestor's needs. Run these jobs at night or over lunch or during a period of light system activity.
2. When two or more batch programs are run at the same time, they:
   - Might take longer to run than if they were run separate (one after the other).
   - Have a tendency to adversely affect response time of interactive programs.

   One way to keep batch programs from adversely affecting interactive programs is to put the batch programs on the job queue. Have the requestors submit their batch jobs via the JOBQ rather than evoking them. This would prevent multiple batch jobs from running at the same time and also allow you to stop the batch activity by stopping the JOBQ.
3. Build a job stream to run at night (it could be out of the JOBQ) and spool the printed output. Use the HOLD parameter on the // PRINTER statement. This will allow the output to be printed but will also hold the output in the spool file. If the printing failed for any reason, you can still print it the next morning. If the output is good, delete the spool entry.
4. You may use the // WAIT statement and request a time (by specifying the TIME parameter) of known light system activity. This will suspend the task until the time specified, when it will then begin execution. Use the // WAIT only for 1 or 2 jobs since the jobs will use some system resources while waiting for the timer to expire.
5. Interactive programs that do a lot of calculating or do a lot of disk reading and writing might have poor or inconsistent response times. The system might treat these interactive programs the same as batch programs. Therefore, you may not want other batch programs running on the system when these types of interactive programs are running. Also, you may want to assign medium priority to these types of interactive programs.
6. You may need to reschedule interactive jobs to avoid disk record waits.
7. During heavy interactive loads, avoid interactive program debug.

Priority

System/36 was designed to provide equitable service to all tasks. Specification of high and medium execution priorities might cause normal priority interactive programs to process slower, because assigning high tells the system to favor it over the other tasks.

Considerations

You may want to assign execution priority to programs based upon the following:

- Main storage size of the program. You may want to assign a lower execution priority to programs that use a larger amount of main storage.

- Whether the program can be run as an interactive or batch program. You may decide to assign higher execution priority to interactive programs than to batch programs, depending on your processing requirements.

- Amount of main storage processing time the program uses. You may decide that a program that uses a large amount of main storage processor time should have a lower execution priority than a program that does not use as much main storage processor time.

- How much elapsed time it take the program to run. You may want to assign a high execution priority for a job that does not take a long time to execute and complete its processing on the system.

- What kinds of demands the program makes on system resources, such as disk files. You may want to base your assignment of execution priority to a job based on the total demands it makes on the system. For example, a program that uses four files and a printer may have a lower priority assigned than a program that uses only one file and does not use a printer.

- How important the program is to your data processing requirements and deadline schedules. You may want programs that are extremely important to your organization assigned a high priority every time they are run.

Guidelines

In general, let the system assign execution and job queue priorities to your jobs; the normal priority can usually handle your job needs.

When assigning priority levels to jobs, your main goal is to have the system process the maximum number of jobs in the least amount of time.

You may want to use the execution and job queue priorities to establish groups of jobs with certain characteristics. For example, you may want to:

- Assign all certain jobs a job queue priority or a execution priority.
- To run your testing jobs with one execution priority and your production programs with another execution priority.
- To assign execution priority to programs based upon the following criteria (that is, should be for critical jobs or schedules only). Use execution priorities sparingly; however, if you must use them:

  1 Use low priority for batch programs (these programs might be swapped more often). Long-running batch tasks should have their priority set to low to prevent their interfering with the interactive programs. Low priority will keep a task below those tasks with normal priority which have had their priority fall into the V-LOW range because of a large number of resource time outs.
  2 Use medium priority for your important interactive programs that perform lots of calculations or disk read and write operations; for example, order entry applications.
  3 Use high priority for extremely important programs or for efficiently coded interactive programs. If you feel that high priority is necessary:
     - Be aware that you will probably degrade the performance of the other tasks.
     - You should not assign high priority to more than one batch job; if you do, your interactive jobs may slow down considerably.

Time spent waiting for executing can be reduced by:

1 Increasing the priority of the job;
2 Running jobs one at a time from the batch job queue;
3 Reducing the execution priority value used by other jobs;
4 Increasing CSP power by upgrading to the Stage 2 processor (if possible).

Spool

There are several considerations when discussing spool parameters.

1 Spool segment size defaults to 10 blocks. Read Spool File Storage Estimates in Chapter 3 of <u>Changing Your System Configuration</u> (SC21-9052) to determine what your size should be, based on the amount of printing you do.

Segment size relates to:

- Average length of a print file
- Average number of pages printed per print file
- The number of print files in the spool files at one time
- The amount of contiguous space allocated in the spool file
- The number of accesses in the spool file:
  - Large: Fewer accesses, but waste space if allocated to large
  - Small: More efficient use of spool file space, but more accesses to spool file
- Performance:
  - Can have an effect on performance

2 The number of spool buffers can range from 1 to 8, each 256 bytes long. Recommendations should be based on the amount of printing you are doing and the type of printer you use:

- If you use a 5219 or 5256 set the number of buffers to 1.
- If your printer is a 5224, 5225-1 or 5225-2, set the number of buffers to 2.
- If your printer is a 5225-3, 5225-4 or 3262, set the buffer based on the amount of printing you are doing. If your printer is running:
  - Eight hours a day, set the number of buffers to 7 or 8.
  - Six hours, set to 5 or 6.
  - Four hours, set to 3 or 4.
  - Two hours a day or less, set the buffer to 2.
- The setting depends on the amount of printing you do and whether or not you are driving the printer close to its rated speed.
- The more buffers you assign, the fewer disk operations are needed to get the printing off the disk. It will take that area away from your user area and may cause an increase in swapping.

3 Spool writer high priority is also an option. This is not recommended unless you do lots of printing all day long, the printing must be done as quickly as possible and the printer is not running within 90% of its rated speed.

4 Spool extents allocated should be minimized to keep the seek span as small as possible. See the earlier discussion in Secondary Usages and Counters.

5 You may specify the spool intercept buffer size with the ACTIVITY parameter in the // PRINTER statement in OCL.

- Segment size relates to:
  - The number of accesses in the spool intercept buffer:
    - Large: Fewer accesses, but waste space if allocated to large.
    - Small: More efficient use of spool file space, but more accesses to spool file on disk
- If a program does a lot of printing, such as a 50-page report, then specify ACTIVITY-HIGH. This will make the spool intercept buffer 10 sectors.
- If a program does only one page of printing, specify ACTIVITY-LOW. This will make the spool intercept buffer one sector.
- If a program does not open a print file, the spool intercept buffer will not be allocated.

OCL Procedure

The importance of OCL performance is directly related to its frequency of execution. OCL performance is:

- Critical for frequently executed interactive jobs
- Less critical for longer running batch jobs
- History file logging:

  1 To log an OCL statement requires a minimum of two disk operations.
  2 When creating a procedure, specify logging until the procedure is debugged. Then change the logging to no log.
  3 You can use the // LOG statement to turn logging ON or OFF individually for each display station.
  4 Log only during debugging.
  5 Each OCL statement requires 2-3 disk ops.

Avoid:

HISTORY-YES	HISTORY FILE

PROCA
// LOAD PROG
// FILE NAME-FILE1
// FILE NAME-FILE2
// FILE NAME-FILE3
// RUN
	PROCA TIME WS
	// LOAD PROG
	// FILE NAME-FILE1
	// FILE NAME-FILE2
	// FILE NAME-FILE3
	// RUN

NOTE: HISTORY FILE LOGGING REQUIRES 3 DISK ACCESSES PER OCL STATEMENT.	

Use:

HISTORY-NO	HISTORY FILE

PROCA
// LOAD PROG
// FILE NAME-FILE1
// FILE NAME-FILE2
// FILE NAME-FILE3
// RUN
	PROCA TIME WS

(Minimize logging of OCL statements to the history file.)

- // * messages

  Avoid:

  - Requires 17 disk operations with text
  - Requires 21 disk operations with MIC#

  Use:

  - Promopt screen
  - INFORMSG-NO:
    - -- Overrides the OCL // * 'MESSAGE' statement
    - -- In procedure OCL, it stays active for procedure
    - -- As a command, it stays active for session

- Nested substitutions

  Avoid:

  - // IF DATAF1-?1'?2?'?MAST
  - Requires 21 disk operations with MIC# (minimize nested substitutions)

- File defaults

  Avoid:

  ```
  // IF FILE-EMPLMAST,UNIT-F2,LABEL-EMPLMAST
  ```

  Use file defaults when possible:

  ```
  // IF FILE-EMPLMAST
  ```

- Program switches

  Avoid:

  Local data area to condition steps

  Use:

  External switches

- Use GOTO and TAG statements rather than several redundant IF expressions. Use one IF expression and a GOTO expression to reduce the time needed to evaluate several IF expressions. The statements skipped by the GOTO and TAG expressions are not processed. For example, rather than doing this:

```
// IF ?1?/Y LOAD $MAINT
// IF ?1?/Y RUN
// IF ?1?/Y COPY FROM-#LIBRARY,NAME-TEST,LIBRARY-P,
// TO-PRINT
// IF ?1?/Y END
```

  Do this, which avoids duplicating the tests for parameter 1 by using GOTO and TAG statements:

```
// IFF ?1?/Y GOTO A
//   LOAD $MAINT
//   RUN
//   COPY FROM-#LIBRARY,NAME-TEST,LIBRARY-P,TO-PRINT
//   END
// TAG A
.
.
.
```

- Use ELSE statements if you have more than one IF expression and only one of the expressions can be true. All ELSE statements are skipped after a true IF expression. For example, rather than doing this, which processes all three statements even though only one of the statements will be true:

```
// IF ?2?/T SWITCH 1XXXXXXX
// IF ?2?/J SWITCH X1XXXXXX
// IF ?2?/S SWITCH XX1XXXXX
```

  Do this, which stops processing after the first true condition:

```
// IF ?2?/T             SWITCH 1XXXXXXX
//   ELSE IF ?2?/J      SWITCH X1XXXXXX
//     ELSE IF ?2?/S    SWITCH XX1XXXXX
```

- Combine IF expressions where possible. The remainder of a statement is not processed after a false condition. For example, rather than doing this (which wastes space in the library):

```
// IF ?2?/T GOTO NEXT
// IF ?2?/J GOTO NEXT
// IF ?2?/S GOTO NEXT
// GOTO ERROR
// TAG NEXT
.
.
.
// RETURN
// TAG ERROR
// PAUSE 'ERROR IN PARAMETER 2'
// CANCEL
```

  Do this, which checks the value of parameter 2 and if it is not equal to T, J, or S, the ERROR is processed:

```
// IFF ?2?/T IFF ?2?/J IFF ?2?/S GOTO ERROR
.
.
.
// RETURN
// TAG ERROR
// PAUSE 'ERROR IN PARAMETER 2'
// CANCEL
```

- Avoid using the informational message (// *) statement to display prompting messages (such as: ENTER MEMBER NAME or ENTER LIBRARY NAME). Use the PROMPT OCL statement and a display format instead. The advantages are:

  - More information can be displayed.
  - Fewer disk operations are required.
  - For remote display stations, fewer data transmissions are made. The // * statement must save the current display contents, show the message, and reshow the display after the procedure ends. The PROMPT statement just shows the display format without having to save the current display contents.

- After you have tested your procedures, stop the logging of the OCL statements to the history file. You may only need to have the OCL statements logged when you are creating and testing your procedure. You can stop the logging by either of two ways:

  - The source entry utility (SEU) has an end of job option that allows you to specify whether the statements should be logged.
  - The LOG command or OCL statement can specify whether the statements should be logged.

- If you have many comments in your procedure, you should put a RETURN statement at the end of the procedure and put your comments after the RETURN. This way the system processes the RETURN statement and your comments are not processed (thus saving the amount of time the system would otherwise have used to read the comments). For example:

```
// ...
// ...         (statements in the procedure)
// ...
// RETURN
*
*
*              (comments)
*
*
```

- Use your own libraries for your applications; that is, run procedures and programs from a library other than the system library (#LIBRARY). The system library has a very large directory, and therefore more time is needed to search for a library member in the system library than for the same member in one of your libraries.

  Also, the SSP always searches the current library first, and if the member is not found, then it searches the system library.

- Use substitution expressions to concatenate values. For example:

```
// IFF ?1?/0 IFF ?1?/1 GOTO ERROR
// SWITCH XXX?1?XXX?1?X
```

  If the first parameter is 1, the SWITCH statement will be:

```
// SWITCH XXX1XX1X
```

  If the first parameter is 0, the SWITCH statement will be:

```
// SWITCH XXX0XX0X
```

- Concatenate values to create unique names. For example the ?WS? expression, which substitutes the current display station ID, can be used to create a file name that will be unique for each display station:

```
// FILE NAME-OUTPUT,LABEL-FILEA?WS?
```

  This allows more than one operator to use the procedure containing this statement because each display station would have its own unique work file. The program refers to the output file as OUTPUT, and if an operator at display station W1 ran the procedure, the actual name of the file would be FILEAW1.

- Use IF conditional expressions to avoid making the system operator respond to an informational message when a procedure is sent to the job queue or when the procedure is started by the EVOKE OCL statement or an SSP-ICF evoke operation. For example:

```
// IF JOBQ-NO IF EVOKED-NO * 'Procedure running'
```

  This example would display the message only when the procedure was being run from the display station; that is, not from the job queue and not evoked.

- Change the value of a parameter, which allows an operator to use fewer keystrokes. For example:

```
// * 'ENTER 1 TO PROCESS MONTHLY; 2 TO PROCESS WEEKLY'
// IF      ?1R?=1 EVALUATE P1='MONTHLY'
//  ELSE IF ?1?=2  EVALUATE P1='WEEKLY'
//        ELSE  CANCEL
INVENTRY ?1?
```

  If the operator enters 1, the procedure INVENTRY MONTHLY is run; if the operator enters 2, the procedure INVENTRY WEEKLY is run. If neither 1 or 2 is entered, the procedure is cancelled.

- Mixed OCL statements

  Avoid:

```
// IF
// FILE.....
// IF
// FILE.....
```

  Use:

```
// IF
// IF
// FILE.....
// FILE.....
```

  (System transients are maximized once called)

- Place OCL comment statements at the end of the procedure
- Use single line statement instead of multiple lines (that is, continuation lines)

Region and Set Procedures

You can improve performance by increasing the region size for the following system procedures:

- BLDFILE
- BLDINDEX
- COMPRESS
- COPYDATA
- KEYSORT
- RESTORE
- SAVE
- SORT

10.5.7 APPLICATION/PROGRAM CHARACTERISTICS

File Processing

1. When you process files sequentially (consecutively), use blocking to reduce disk accesses. This is true for update as well as read and write. For example, if you are updating a file with DISP-SHR and have 10 records per block, when you read the first record all 10 records will be brought into storage. Then, when you update the first record only those sectors that contain the first record will be written back to disk. The other records will not be written to disk. Next you read the second record and, since it is already in the buffer (even if it is in the same sector as the first record), a disk access will not be necessary unless someone changed the second record. Disk data management keeps track of which records in main storage are "current." Note that the updated records are written back to the disk all the time but the read records are read from the disk only every tenth time.
2. You do not have to recompile your programs to take advantage of blocking; you only have to change the // FILE statement to change blocking (DBLOCK and IBLOCK parameters).
3. As you block, be careful to keep the execution size to 64K or less. If the program size plus the buffers used for blocking cause the execution size to exceed 64K, the buffers will reside in TWS (that is, TWS plus 16 KB). Since the TWS has to be mapped to and also swaps independently of the program, performance could be adversely affected.
4. Two parameters control blocking of both the data records and the index:
   - The first is DBLOCK and allows you to specify how many records you want in the data buffer.
   - The second is IBLOCK and allows you to specify how many index entries you want in the index buffer.

   This can be quite useful if you are processing an indexed file sequentially by key. The IBLOCK parameter will reduce the disk accesses by keeping a large number of index entries in main storage.

5 Indexed files with and without alternative indexes:

- Read only gives the same performance
- Write/update:
  - Files with alternative indexes give slower performance because:
    - All indexes are updated before the program resumes execution
    - Adds require update to each related index
    - Deletes affect each related index
    - Key updating requires a delete and add to the index and a check of each related index

6 Duplicate keys:

- The system checks each add/update unless "BYPASS-YES" is specified
- Specify "BYPASS-YES" on the // file OCL statement to eliminate unnecessary checking
- If possible, include the checks in the application program

MRT Programs

If you use MRT programs, be careful not to overload them. Too many display stations attached to a MRT can cause response time problems because a display station attached to the MRT must do all its processing, including disk and printer operations, before any other display station can enter the MRT to do its processing. Therefore, the response time for a display station may include the time that it had to wait before it could begin processing. As a result, MRT programs work best when there is lots of keying necessary at a display station before the enter key is pressed. This allows one display station to do its processing while the others are keying, thus minimizing the interference.

Work Station Formats

The maximum number of formats is 256. However:

- The Format Index area is located in the System Work Space (SWS) on System/36 if the number of formats in the format member is 32 or less.
- If the number of formats is more than 32 the format index area is located in Task Work Space (TWS). Since the SWS is a shared resource, less swapping will occur since the system will swap TWS before it will swap SWS.

Acknowledge Operator Input

Operator interaction with a display station is usually conversational; for example, an operator makes an inquiry, and the display station shows the requested data. For this reason, you should be concerned about how long an operator waits for System/36 to respond.

If a program takes a relatively long time to respond to an operator, you might want to display an in-process message immediately after the program receives the operator's input. For example, you might want to do this for a program that does extensive calculations with the operator's input. Acknowledging operator responses at remote display stations might affect their response times.

10.6 SYSTEM-ORIENTED GUIDELINES

10.6.1 MAIN STORAGE PROCESSOR (MSP)

A MSP usage that is significantly higher than normal may indicate a processor-bound job, which could be increasing response time at your display station.

Processor-bound jobs may be:

- Programs that perform:
  - Many calculations
  - A sort
- Program compilations
- Batch programs running with:
  - High priority: consider rescheduling these jobs, or lowering their priority
  - Other batch programs: consider rescheduling some of the batch programs or sequencing them using the job queue.

10.6.2 CONTROL STORAGE

Considerations

Depending on the size of your control storage relocatable area and the functions you run there, all functions may not be able to run concurrently. For example, if you have System/36 Model 5360 stage 1, and if you run BASIC and FORTRAN programs at the same time, you cannot also run the data compression, extended trace, SMF communications, or 1255 MICR functions.

The following is a list of functions that run in the relocatable control storage area:

- Diskette Functions
- Data Communications
- BASIC
- FORTRAN
- Data Compression
- Extended Trace
- SMF Communications
- 1255 MICR

Data communications and diskette functions are always guaranteed space to run. If enough space is not available for the functions, an error message is issued. To recover from the error, you must either wait for some programs to finish or cancel some programs. You can either cancel the program requesting the storage (option 3), or cancel some other programs that are using one of the other functions. This will free up the space for the request program. The following information will help you understand and schedule which functions can run concurrently on your system.

For System/36 Model 5360; programs that can run concurrently:

BASIC	FORTRAN	DATA COMPRESSION	EXTENDED TRACE	SMF COMMUNICATIONS	1255 MICR
X	X				
X			X	X	X
	X		X	X	X
X		X	X		
	X	X	X		
		X	X	X	X

Notes:

- All programs can run concurrently on System/36 Model 5360 with Stage 2 control storage.
- Data compression, BASIC, and FORTRAN are mutually exclusive.
- To determine if your system is a Stage 2 system, look on the label on the inside of the control panel cover. In the top center of the label, immediately to the right of the words Processor Check, will be the words Stage 2. If Stage 2 does not appear on the label, you have a Stage 1 system.

For System/36 Model 5362; programs that can run concurrently:

BASIC	FORTRAN	DATA COMPRESSION	EXTENDED TRACE	SMF FACILITY
X			X	X
X		X	X	
	X		X	X
	X	X	X	
		X	X	X

Notes:

- All programs can run concurrently on System/36 Model 5362 with a work station controller.

- BASIC and FORTRAN are mutually exclusive.

10.6.3 CONTROL STORAGE PROCESSOR (CSP)

CSP time can be reduced by:

1. Simplifying the processing or moving some of it to batch;
2. Reducing the number of physical disk I/O operations;
3. Eliminating unnecessary file opens and closes;
4. Increasing CSP power by upgrading to the Stage 2 processor (if possible);
5. Eliminating unnecessary external calls.

10.6.4 TASK WORK AREA (TWA)

The task work area is a file which is used by each task on the system. The space in the task work area is used as needed, dictated by the load on the system at any point in time. Many factors affect the amount of space needed in the task work area. Among these factors are the following:

1 Number of display stations. Each display station attached to the system, except the system console, requires 65 blocks. The system console area is calculated into the minimum size of the task work space. All display stations configured as subconsoles require an additional 4 blocks.

2 Job queue. If the job queue is configured and active, an additional 30 blocks are required.

3 User programs/system utilities. Each of these tasks is allocated 26 blocks, regardless of the size of the program.

4 Evoked programs. If programs are evoked on the system, 30 blocks per program are required.

5 SPOOL. Each active SPOOL writer requires 3 blocks.

6 Communications. Each piece of active communications support requires additional task work space.

7 Task dump area. This area is dynamically allocated as required.

8 Task work space. This area is allocated enough space to containthe TWS, rounded up to the next 2K page.

9 The task work area will automatically be extended by the system if more space is required than is available at any point in time.

10 Make sure task work area (TWA) is large enough to support workload:

- You can specify from 444 blocks through 6553 blocks
- Use the default value for the size of the task work area

11 If you decide to change the size of the TWA, run the system measurement facility (SMF) to determine the size:

```
TASK WORK AREA SIZE.......  1100 BLKS
TASK WORK AREA USAGE......    60 %
TASK WORK AREA EXTENTS....     0
```

12 If you establish a TWA that is approximately 20% greater than the extended area that the system assigns during periods of high activity on your system, you may be able to cut down on unnecessary extents.

11.0 HOW TO USE SMF?

11.1 SMFSTART PROCEDURE

- You can start SMF many ways:
  - Enter the command SMFSTART and go directly to the start screen.
  - Enter the command SMF to go to the main SMF menu.
  - Via the main HELP screen:
    - Select option 8: Problem Determination and Services
    - Select option 2: Run Service Aid Procedures
    - Select option 4: Use System Measurement Facility.

- From the SMF menu select option 1: Collect SMF data for the SMF report.

- Data collection time interval:

  - In order to properly monitor the performance of System/36 it is best to execute SMF over several time periods. This will ensure that the most accurate data regarding the system's activities are received.

  - The user has the option of selecting statistical samples or snapshots of the system's resources to be taken in intervals starting from 10 seconds to every 5 minutes.

  - It is recommended to use a one minute time-interval over the period where you are experiencing the problem.

  - A ten second snapshot may be appropriate (rarely) if you can get several snapshots over one display station response.

- Size of data collection file:

  - This prompt allows you to select the size of your data collection file.
  - The size can be from 1 to 312815 blocks in length.
  - The size will vary depending on the length of time you plan to run SMF.
  - You should start out using the default of 200 blocks.

- Collect communications data:

  - Your response to this prompt specifies whether you wish to collect data about your communications environment.
  - Additional time is required to collect this data, and additional space in the data collection file is used.
  - If you choose to collect communications usage data, you must specify the correct line speed for the lines on which you want to collect data.
  - Failure to enter the correct line speeds will result in erroneous output data.
  - If no line speed information is entered, no communications data is collected by SMF, even if you specify Y for this prompt.
  - "ALL" must be specified as the report option in order to print communication data.

- Name of the data collection file:

  - It is recommended to change the default data collection file name from SMF.LOG to SMxxxxxx where xxxxxx is the current date.
  - Once you begin analyzing your system's performance over several different periods this will help you to identify when the run was made since SMF will print the data collection file name at the top of each page of the report.

- Collect I/O and SEC data by task:

  - Your response to this prompt specifies whether you want to collect counters separately for each task as well as for the entire system.
  - Enter a "Y" to generate a detailed report of the system's resources.

- Collect user and system data by file:

  - Your response to this prompt specifies whether you want to collect
    access counts for user and system files on your disk.
  - "N" indicates that no data by file will be collected.
  - "Y" indicates that data for both user and system files will be collected.
  - "U" indicates that data for only user files will be collected.
  - You cannot collect data for system files only.

11.2 SMFSTOP PROCEDURE

- You can stop SMF many ways.
  - Enter the command SMFSTOP and go directly to the stop screen.
  - Enter the command SMF to go to the main SMF menu.
  - Via the main HELP screen:
    - -- Select option 8: Problem Determination and Services
    - -- Select option 2: Run Service Aid Procedures
    - -- Select option 4: Use System Measurement Facility.

- From the SMF menu select option 2: Stop collecting SMF data for the SMF report.

- The SMFSTOP procedure has no parameters and no help display.

- Pressing the Enter key at the SMFSTOP display will immediately stop the data collection program. Consequently, the last reported sample interval might be shorter than the specified interval. The reported usage statistics are still accurate, however, because they are based on the full sample interval.

- The system operator can also stop the data collection program by entering the STOP SYSTEM control command. The data collection program then waits for the next sample interval to pass before stopping.

- The system will automatically stop SMF if it is left running longer than 24 hours.

11.3 SMFPRINT PROCEDURE

- You can print SMF reports many ways:
  - Enter the command SMFPRINT and go directly to the print screen.
  - Enter the command SMF to go to the main SMF menu.
  - Via the main HELP screen.
    - -- Select option 8: Problem Determination and Services
    - -- Select option 2: Run Service Aid Procedures
    - -- Select option 4: Use System Measurement Facility.

- From the SMF menu select option 3: Print the SMF report.

- Your response to these prompts specify the type of report to be produced by the report writer program.

- Report opions:

  - DETAIL:

    Prints configuration, device utilization, tasks active, storage totals information, and summary data.

  - ALL:

    Prints configuration, device utilization, tasks active, storage totals, system event and I/O counters, communications and SMF summary data.

  - MINI:

    Prints configuration, device utilization, significant system event and I/O counters, and SMF summary data.

  - SUMMARY:

    Prints configuration information and SMF summary data.

Note: It is recommended to initially generate Summary reports. This report will help the user to identify potential problem areas. Once problem areas are identified, then more detailed reports can be run to help identify the problem.

- Delete data collection file after printing?
  - "Y" - Delete the data collection file
  - "N" - Do not delete data collection file

Note: It is recommended that you select "N" for this prompt until you have retrieved all data you will need.

- Name of Data Collection File:

  - Name of the SMF data collection file that contains the data to be printed.

- Starting date:

  - Starting date of the data to be reported from the data collection file. (YY/MM/DD)

- Starting time:

  - Your response to this prompt specifies a beginning time in hours, minutes and seconds (hhmmss) for the report.

- Ending time:

  - Your response to this prompt indicates an ending time in hours, minutes and seconds (hhmmss) for the report.

11.4 SMFDATA PROCEDURE

- This procedure enables the user to create a report file from the SMF data collection file.

- Once the file is created, the user can extract various fields from the SMFDATA file with an appropriate application program.

- The user could use a custom written inquiry program or take advantage of currently available programs like Query, Retrieval/36 or BRADS/36 to generate a report highlighting selected areas of the SMF report.

- The report file program can be run while the data collection program is still active (the data is formatted from the existing data collection file) or after it has ended.

- You can create an SMF data file many ways:

  - Enter the command SMFDATA and go directly to the create report file screen.
  - Enter the command SMF to go to the main SMF menu.
  - Via the main HELP screen:
    - Select option 8: Problem Determination and Services
    - Select option 2: Run Service Aid Procedures
    - Select option 4: Use System Measurement Facility.

- From the SMF menu select option 4: Create a SMF report file from the SMF menu.

- Report option:

  - Your response to this prompt specifies the type of file to be created by the report file program.

  - Three types of files can be created:
    - SUMMARY
    - DETAIL
    - ALL

  - Chapter 3, SMF Reports, in the SMF Guide gives a detailed description of the types of information that can be contained in the file.

  - ALL is the default value for this parameter.

- Delete data collection file.
  - Your response to this prompt specifies whether the data collection file should be removed from the disk after the report file program ends:
    - "Y" indicates that the file should be deleted.
    - "N" indicates the file should not be deleted.
    - "N" is the default value for this parameter. It is recommended that this be selected until you have completed your use of the information.
- Name of data collection file:
  - Your response to this prompt specifies the name of the data collection file to be used as input for the report file program.
  - SMF.LOG is the default value for this parameter.
- Name of the report file:
  - Your response to this prompt specifies the name you want to call the report file that will contain the data from the data collection file.
  - SMF.DATA is the default value for this parameter.
- Starting date:
  - Your response to this prompt specifies a beginning date in year, month, and day (yymmdd) for the report file program. All 6 digits must be entered.

- Starting time:

  - Your response to this prompt specifies a beginning time in hours, minutes, and seconds (hhmmss) for the report file program.
  - Any time from 000000 through 235959 may be entered. All 6 digits must be entered.
  - The report file program diagnoses invalid times (for example, 240000), and an error message is displayed.
  - The default value for this parameter is blank or 000000, which means that the report file should begin with the first record in the data collection file.
  - Only samples recorded at or after this beginning time are processed by the report file program.

- Ending time:

  - Your response to this prompt specifies an ending time in hours, minutes, and seconds (hhmmss) for the report file program.
  - After the program begins creating the report file, it will continue to copy data until the end time is reached.
  - Any time from 000000 to 235959 may be entered. All 6 digits must be entered.
  - The report file program diagnoses invalid times, and an error message is displayed.
  - The default value for this parameter is blank or 000000, which means that the copying of data should end with the last record in the data collection file.

12.0 CONCLUSIONS

SMF can be a helpful tool for tuning your system and for planning your capacity. However, you must run SMF regularly to realize the benefits.

As changes are made to improve performance, the SMF values may show that there has been improvement. The operators, on the other hand, may not perceive any improvement. Before investing in more hardware to improve performance, spend some time making sure that the hardware will give you the improvement you desire. You must weigh the cost of additional hardware against the performance gain to determine if the hardware is justified. It is easier to determine when additional disk or main storage will noticeably improve performance than when additional or faster processors will noticeably improve performance.

Generally, unless you change something in your system such as file placement, as the usage and counter values go up, your response times will degrade. Small changes in the values probably will not show any degradation.

After you have run SMF and followed the recommendations and guidelines, reevaluate your performance. If your performance is still not what you want, you have two choices: Accept the performance you are getting or look elsewhere for the cause of the poor performance you are experiencing. The first place to look is at your applications.

SMF offers you some excellent tools to evaluate performance of the IBM System/36. Usually you can improve performance and plan for future growth by spending a few hours setting up some programs and procedures that will enable you to track the system's performance. The experience will be valuable by providing some insight and a better understanding of how your system works.

13.0 APPENDIX A: PROGRAM OPTIMIZATION

Approach

Monitor the program to identify where time and resources are consumed.

Profiling the program

- Method 1: Using System Measurement Facility (SMF)

- Method 2: Augment the Program

13.1 METHOD 1: USING SYSTEM MEASUREMENT FACILITY (SMF)

Use SMF for a short time to profile each transaction and/or function. The steps are:

Step 1. After signing on to System/36, the main help menu is displayed, select the following options:

- A. Select Option 8: Problem Determination and Service.
- B. Select Option 2: Run Service Aid Procedures.
- C. Select Option 4: Use System Measurement Facility (SMF)

  (You can use the SMF procedure "SMFSTART" to replace items A through C.).
- D. Select Option 1: Collect SMF Data.
  1. Data collection time interval in minutes and seconds.
     - Specify a sufficient time interval that would include the following items:
       - a) Time to key the transaction/function to be profiled.
       - b) Time to process the transaction.
       - c) Wait time--time used as a buffer between each test.
  2.
     - Name of data collection file.
     - Specify the name of the data collection file you want SMF to write data to.
  3. Collect I/O and SEC data by task?
     - Your response to this prompt specifies whether you want to collect counters separately for each task as well as for the entire system.
       - Enter a "Y" to this prompt.

Step 2.

- A. Execute the application to the point of entering the first transaction/function to be profiled.
- B. Take the time of day for the system.
- C. Wait until the specified time interval expires (this will separate application startup from actual profile testing).

Step 3.

- A. Enter first or next transaction/function.
- B. Wait until the specified time interval expires (this will keep each transaction/function profile separate).

Step 4. Repeat step 3 until finished.

Step 5. Print SMF report using the SMF procedure "SMFPRINT." Your response to this prompt specifies the type of report to be produced by the report writer program. Specify the following options, others are optional:

A. Print option: Specify "ALL".
B. Name of file: Specify the name you entered in step 1.D.2.
C. Use the time from step 2.B for printing time intervals:
   1. Print interval "from" limit.

      Your response to this prompt specifies a beginning time in hours, minutes, and seconds (hhmmss) for the report.
   2. Print interval "to" limit.

      Your response to this prompt specifies an ending time in hours, minutes, and seconds (hhmmss) for the report.

   Print only the intervals at or after the beginning time and before the ending time are processed.

Step 6. Evaluation

The information printed on the SMF report is divided into 11 sections. Evaluate the following secion:

- Section: Task Status Information.
- Topic: I/O and System Event Counters (SECs) Data by Task
- Items:

A. MSP Usage: The percentage of main storage processor time used by the task. You would prefer this item to be a high percentage of the MSP usage and RES T -Out to be zero. If so, the transaction/function completes within one time slice.

B. RES T-Out: The number of resource utilization time-outs for the task. A value of zero means no additional time slices were required. A value greater than zero and a system that is swapping (swap outs occurring) present the possibility of the task being swap-eligible after each time-out.

C. MSP T-Out: The number of main storage processor time-outs for the task. A value of zero means that the tasks do not use more than 100 MS of processor time in a 200-MS period.

Note: Most application programs will have one or more functions that cause time-outs to occur.

D. Disk 1-4: Indicates the number of read, scan, and write operations on the disk drive 1-4 required to complete the transaction.

E. Evaluate remaining primary and secondary utilizations and counters.

13.2 METHOD 2: AUGMENT THE PROGRAM

Build a program that augments the program with bookkeeping statements necessary to gather its own profile:

- Routines that use the most time should receive the greatest effort toward optimization.

- Most programs have one critical point responsible for the majority of execution time:

  - Normally, 5% of the code accounts for more than 50% of its execution time. Two important implications for programmers:

    - Decreasing the run time of 5% of the code speeds up the entire program tremendously.

    - Any efficiency improvement in the 95% of the code makes little difference in the overall performance of the program.

IBM®

Printed in U.S.A.

G360-1006-0