PROBLEMS INVOLVED IN SYSTEMS MANAGEMENT FACILITIES

by1264

THOMASYNE MAE CAMPBELL

B. A., Rosary College, 1967

A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Statistics and Computer Science

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1970

Approved by:

K. E. Kemp

Major Professor

ACKNOWLEDGEMENT

I wish to express my appreciation to Mr. Charles S. Walker, Assistant Professor of Computer Science at Kansas State University. Mr. Walker, although not a member of my committee, rendered invaluable guidance, direction, and assistance in all phases of the preparation of this paper.

TABLE OF CONTENTS

CHAPTER		PAGE
I.	INTRODUCTION	1
	Objective of the Study	2
	Definition of Terms	2
II.	SYSTEM MEASUREMENT	3
	History	3
	Purposes of System Measurement	3
	Problems Involved in System Measurement	4
	Information Required	5
III.	ANALYSIS OF THE SMF OPTION	6
	Data Set Organization	6
	Entries and Exit Routines	10
	Space and Time Requirements	13
	Timing and Allocation of Charges	15
	Reenterable Code	19
	Testing Exit Routines	20
IV.	OTHER ACCOUNTING FACILITIES	21
	MUSIC	21
	SMS/360	22
	Computer Performance Monitor II	23
V.	SUMMARY AND CONCLUSIONS	24
BIBLIOGRAPHY		26
APPENDICES		28
Appendix A.	Definition of Terms	29
Appendix B.	Table of Common Exit Parameter Area	37

LIST OF TABLES

TABLE		PAGE
I.	SYSTEM AND JOB STATISTIC INFORMATION	7
II.	EXIT ROUTINE FEATURES	11
III.	COMMON EXIT PARAMETER AREA	37

CHAPTER I

INTRODUCTION

The past twenty years have seen the rapid expansion of an extremely powerful tool--namely the electronic digital computer. The technology associated with these machines has advanced rapidly, permitting the development of capabilities which only a few short years ago would have staggered the imagination. Man has been equally progressive with respect to the application of this tool in his endeavors. In particular the area of management has built swiftly upon the advantages offered by computer processing. Although the computer only served as a glorified bookkeeping machine in its earliest applications, its usage quickly advanced.

Today there are complex product scheduling and planning programs, sales forecasting and analysis routines, elaborate simulations and much work is going into the development of large scale information systems. Yet, despite the rapidity with which management has seized upon the computer for assistance in making better business decisions, there has been little movement by computer center managements to bring management technology to bear upon the administration of computer services.

OBJECTIVE OF THE STUDY

Although there are many management problems involved in operating any computer installation, this paper is focused directly upon the most fundamental task of the bookkeeping of the machine itself. The purpose of this paper is to research system measurement techniques. The historical basis of system measurement, its purposes and problems, will be discussed. In particular, an analysis will be given of the "System Management Facility" option provided by IBM. The last topic to be covered concerns other accounting facilities available.

DEFINITION OF TERMS

The reader should be familiar with the information presented in the following publications:

IBM System/360 Operating System: Concepts and Facilities, Form C28-6535-4.

IBM System/360 Operating System: Introduction to Control Program Loglo, Form &28-6605-3.

IBM System/360 Operating System: MVT Control Program Logic Summary, Form Y38-6658-2.

IBM System/360 Operating System: System Programmer's Guide, Form C28-6550-3.

Definitions of terms particularly pertinent to the research can be found in Appendix A.[1]

---

[1] IBM System/360 Operating System: Concepts and Facilities (IBM Systems Reference Library, Form C28-6535-4. Poughkeepsie, New York: International Business Machine Corp., 1969), pp. 57-65.

CHAPTER II

SYSTEM MEASUREMENT

HISTORY

In the earliest days of computers, general purpose machines were classified under either one of two headings--scientific or commercial. Scientific computers were most generally concerned with arithmetic operations, while commercial machines tended toward I/O operations. So system performance evaluation was handled in a like manner. Scientific computers were judged by the speed of arithmetic instructions. Commercial computers, on the other hand, were evaluated by the speed of reading a card, writing a record, etc.

However, many changes in computing systems have evolved. I/O buffering, direct access storage, channels and operating systems were introduced. As computing systems become more and more complex, it has been increasingly difficult to measure their performance.

PURPOSES OF SYSTEM MEASUREMENT

System Measurement's purposes are eight. First, it provides data, which upon examination and analysis, improves typical program speeds. Second, it furnishes "I/O-Compute ratios" necessary to properly schedule under multiprogramming (MFT/MVT in OS/360) configurations. Third, it supplies information needed to establish efficient programming standards in any language. Fourth, system performance evaluation

enables the gathering of materials, which aid in the continual balance of channel and device utilization, directly increasing average throughput capacity and speed. Fifth, it assists in the evaluation of proposed equipment changes. Sixth, with its use, data needed to optimize system application and data organization can be inspected, thus reducing CPU wait time and increasing throughput. Seventh, data needed to analyze system residence requirements is provided, which can help decrease system overhead in terms of both core and speed. Lastly, it provides an accurate basis on which users can be charged for resources used.

PROBLEMS INVOLVED IN SYSTEM MEASUREMENT

In addition to the purposes inherent in the concept of system measurement, there are also some particular problems involved in assembling the information required. The first problem arises for the fact that a sound medium (data base) must be available on which to record systems measurement information. Thus a permanently opend data set is required. When the above data set becomes full, it is necessary to have an alternate data set ready, and also the former must be dumped into the file of previously collected system measurement information.

Second, it is necessary to have control and validation entries. A program entry, hook, or interface for user validation, and control of JCL (Job Control Language) is required. Another entry is required at Job (and Step) initiation time

with information from the JFCB (Job File Control Block) made available to the exit routine. Also, an entry should be accessible at both Step termination and Job termination time, with termination status and job statistics available to the exit routine. At Step termination, the exit routine should be able to request termination of the Job. Entries when time, printed lines or punched cards have been exceeded, would be allowed, before action is taken and the routine should be capable fo voiding the overflow condition.

Third, is the problem of properly timing and allocating charges to the jobs running in the stream. It is necessary to have accurate accounts of CPU time, I/O activity, and CPU wait time for each job.

Fourth, is the problem of degradation of space and throughput caused by including system measurement in the operating system.

Fifth, in and MVT configuration, it is necessary that all exit routines be written in a reenterable fashion.

Lastly, a most thorough testing of routines to be used must be made, since they will become part of the control program.

INFORMATION REQUIRED

When the notion of implementing a measurement facility was considered, the Systems Management Fask Force of SHARE

suggested certain information for collection.[2] The information for collection for IBM OS/360 by the first group deals with job level information, while the second category concerns step level information. The third group treats the periodic job ans step data, and lastly, the fourth group entails the system use profile information. Below in Table I are listed the data requested within each group. Groups One and Two should be collected in a data set, and also the information should be placed on SYSOUT in message format. The information described in Group Three is suggested to be collected at intervals under the control of the operator and by user code in validation entries. Group Four information collection should be initiated by the operator or control program at validation entry. In effect, it is recorded according to time intervals specified at intiiation as a parameter.

CHAPTER III

ANALYSIS OF THE SMF OPTION

DATA SET ORGANIZATION

Systems Management Facilities (SMF) is "an option of the System/360 Operating System selectable at system generation in conjunction with the Multiprogramming with a variable number

---

[2]J. M. Noerr, "Status Report of the Systems Management Task Force," SHARE Secretary Distribution, Form C4819 (August, 1967), pp. 4-5.

TABLE I

SYSTEM AND JOB STATISTIC INFORMATION

Group One--Job Level	Group Two--Step Level
Job Log number	Step Log number
Job card fields after validation entry	EXEC card fields after validation entry
Entry time stamp	Step CPU time
Initiator time stamp	Unit address of allocated devices
Exit time Stamp	Output and input quantities (if selected for step level, eliminate at job level)
Priority	Step termination status
Output quantities (lines and code)	Maximum core used
Job, CPU time	
Termination status	
Maximum core used	
SYSIN reader identification	
Reader/interpreter time	
SYSOUT writer identification by class	
Output writer time	

Group Three--Periodic Job and Step Data	Group Four--System Use Profile
Job input queue time	Time-of-day
Job output queue time	Total memory in use
Step start time stamp	Number of active jobs
Data set description (include DD statements)	Number of passive jobs
Device use time	Number of devices in use by channel and type
	Systems work space in use
	Work input queue lengths
	Work output queue lengths
	Channel queue lengths
	Seek queue lengths
	Number of readers in use
	Number of active tasks
	Timer queue length
	Number of writers in use by class

of tasks configuration."[3] It is a new option implemented by IBM, which consists of a new set of load modules and modifications to existing load modules in the control program. SMF very simply monitors each mob from initiation to termination and records data about the jobs. The option consists of two parts--one to collect data, and the second to analyze it.

The SMF option does provide for a permanently open data set. This data set may reside on tape. If the SMF data set is resident on tape only one data set is required. At IPL (Initial Program Loading), a tape drive is designated to the SMF data set, and this of course means that the particular tape drive is dedicated to SMF and will not be available to any other jobs running in the system. Cost of a tape drive per month is $500 plus a share of the cost of the controller. The data set may take up only 20 volumes of space, at which time a new tape must be mounted, and the IPL procedure repeated. It should be noted, however, if a tape is used, data may be processed directly from it by report and analysis programs, without the use of a dump utility program.

If the SMF data set is to be resident on a direct access volume, two data sets must be allocated at IPL. The two data sets need not be defined on the same device, however the two data sets must be of the same type. When the first data set becomes filled, automatic switching occurs. The dump utility

---

[3] IBM System/360 Operating System: Planning for System Management Facilities (IBM Systems Reference Library, Form C28-6712-0. Poughkeepsie, New York: International Business Machine Corp., 1969), p. 5.

program must then be used to transfer the data to tape. When this is concluded, the data set is then restored and ready for use.

When dumping must occur, a message is written to the console. Dumping can be initiated by two methods. First, a job specifying the dump utility procedure may be held in the job queue until dumping is needed. Second, the JCL to initiate dumping may be read into the job stream from the reader. The literature does not indicate that the data set is saved if there is an error that occurs while dumping. If it is not, data not yet recorded will not be retrievable.

SMF facilities also provide for buffering the data before it is entered on one of the data sets. The size of the buffer is installation dependent, and is specified at IPL. The buffer provided is divided into two equal parts to allow overlapping of blocking and writing. Specifics about the size of buffer that should be specified can be found in the IBM/360 Operating System--Planning for System Management Facilities manual.

SMF writes thirteen types of records to the SMF data set. Each record written to the data set contains a header, with the type of record to follow indicated. The types of records written to the data set are an IPL record, wait time record, dump header record, dump trailer record, step termination record, job termination record, output writer record, data lost record, I/O configuration record, vary online record, allocation recovery record, vary offline record, and end-of-day

record. These records are written automatically to the SMF data set at appropriate times.

ENTRIES AND EXIT ROUTINES

SMF provides five interfaces from which user written exit routines can be entered. These are:

1. From the reader/interpreter of the job scheduler just before each job control statement is interpreted.
2. From the initiator/terminator when a job is selected for initiation.
3. From the initiator/terminator when a step is selected for initiation.
4. From the initiator/terminator when a step and/or job is terminated.
5. From the timer second level interruption handler when a specific CPU or wait time limit for the job or step is exceeded.[4]

All or any of the interfaces can be used. If the interface is not to be used a dummy routine is inserted automatically. The installation written exit routines allow a further monitoring of each job, with the additional ability to cancel or terminate jobs, change, create, or delete fields, determine accounting charges, write additional records to the SMF data set, and open and write records to other data sets. The requirements necessary for writing records to SYSOUT can be found in the IBM/360 Operating System--Planning for System Management Facilities manual.

A summary of each exit routine, the parameters passed it, and when it is called may be found in Table II. Standard

---

[4] Ibid., p. 7.

TABLE II

EXIT ROUTINE FEATURES*

Exit Routine	Parameters Passed	Entered	Return
Job Validation IEFUJV	Standard JCL Image JCL Type	Each JCL Card Job Enqueue	Continue or cancel
Job Initiation IEFUJI	Standard Programmer Name Priority Account Fields	Job Initiation	Continue or cancel
Step Initiation IEFUSI	Standard Step Name Step Account Fields	Step Initiation	Continue or cancel
Termination IEFACTRT	Standard Programmer Name Job CPU Time Job Account Flds Step CPU Time Step Account Flds Completion Code SMF Record	Step Termination	Continue or cancel; Write or Skip SMF record
Time Limit IEFUTL	Standard Type Code	Job CPU Time Limit Exceeded Step CPU Time Limit Exceeded Job Wait Time Limit Exceeded	Continue with new Time limit or cancel

*IBM System/360 Operating System: Planning for System Management Facilities (IBM Systems Reference Library, Form C28-6712-0. Poughkeepsie, New York: International Business Machines Corporation, 1969), p. 11.

parameters passed to each of the exit routines include job name, time stamp, system ID, user ID, step number installation data, and a communication field. A fuller description of these parameters may be found in Appendix B.

Note that exit routine IEFUJV receives control at an encounter of a JCL card, and once before the job is queued for scheduling for the purpose of validation. JOB and EXEC fields may also be checked at exits for job and step initiation respectively. The accounting fields are placed in a formatted list in the "Job Initiation" and "Step Initiation" exits, thus making validation far more efficient at these exits. However, this exit provides the only opportunity at which priority can be assigned. In the interest of degradation of time and space, it seems that exit IEFUJV should only be implemented for priority purposes, if it is to be implemented at all. If installation records are to be created, they must be defined in this exit.

Each exit routine must save registers on entrance, and restore registers upon return. The entry point name and the name used in the call of the exit must be identical. Routines can communicate with each other with the use of the user communication field and identification. These parameters are passed to each exit routine.

Exit routines can write to the SMF data set via the SMFWTM macro instruction. Further information on this macro, and the use of it, may be found in the IBM/360 Operating System--Planning for System Management Facilities manual. If

installation defined data sets are being used, additional DD statements are needed in the appropriate procedures.

> A data set used by routine IEFUJV requires a DD statement in the reader/interpreter cataloged procedure, a data set used by routines IEFUJI, IEFUSI, and IEFACTRT requires a DD statement in the initiator procedure, and a data set used by routine IEFUTL requires a DD statement in the current job step.[5]

SPACE AND TIME REQUIREMENTS

In considering the implementation of the SMF option, additional space requirements and degradation of throughput are of major concern to the installation. The storage areas that must be expanded lie in two separate areas. First there is a need for more main storage, and second there are additional auxiliary storage requirements.

In main storage, sizes must be enlarged in the nucleus, system queue area, and the system task area (if exit routines are to be used). SMF requires an additional 1500 bytes in the nucleus. Space in the system queue area is required for the following blocks: timing control table, SMF control table, and the SMF buffer. Also space is needed for communication between exit routines if this is desired. One TCT (Timing Control Table) is created for each active step. Approximate size of this TCT is equal to 100 bytes plus four times the maximum number of DD's per step plus eight times the number of devices in each DD statement.[6] The SMF control table occupies

---

[5] Ibid., p. 17.

[6] Ibid., p. 7.

sixty-eight bytes. A minimum buffer length is four hundred bytes. The additional space needed for communication between exit routines is dependent upon the maximum number of jobs being executed at the same time. No estimate is given indicating the approximate number of bytes needed for the communication field.

In addition to the space needed for the SMF data set, more auxiliary storage is also needed for the System Libraries. The following System Libraries require expanding:

SYS1.LINKLIB--5 tracks
SYS1.SVCLIB--2 tracks
SYS1.PARMLIB--1 track
SYS1.MACLIB--1 track
SYS1.NUCLEUS--1 track[7]

No value is given for the amount of time added by the use of the SMF option. However, it is noted that the extended overhead will depend on factors such as "extent and blocksize of the SMF data sets, the types of devices to which the data sets are assigned, the types of SMF records written, and the execution time of the exit routines."[8]

Some consideration to system overhead is given, however. SMFDEFLT parameters are provided in the SMF option. These parameters signify which SMF operations are to be performed. Thus the option is a flexible one. The SMFDEFLT parameters are specified at each IPL. They are eight:

---

[7] Ibid., p. 8.

[8] Ibid.

OPT=1 Only system and job data is to be collected.
2 System, job and step data is to be collected.

EXT=No Exits will not be taken.
Yes Exits will be taken.

JWT=n Job wait time limit.

BUF=n Size of the SMF buffer.

SID=xx Identifies the system on which SMF is active.

MDL=nn Identifies the model number of the system's CPU.

OPI=Yes Operator is allowed to modify parameters.
No Operator is not allowed to modify parameters.

MAN=NONE No records will be written to the SMF data set.
USER Only user records will be sritten.
ALL Both SMF and user records will be written.

PRM= Specifies primary data set.

ALT= Specifies alternate data set.
Data sets are specifice by volume, serial [9]
number, device address and data set label.

TIMING AND ALLOCATION OR CHARGES

Since the "time received by the exit routines provides one of the most direct means of allocating charges, it is a critical aspect of system measurement. At step termination and job termination, exit routine IEFACTRT receives step and job CPU time. At job termination CPU time is passed as the sum of step CPU times. In addition, the start and stop time for processing of the job by the reader/interpreter and the device type and class of the reader device are included in the "Job Termination" record passed also to the exit routine.[10] Also

---

[9] Ibid., pp. 37-39.

[10] Ibid., p. 10.

the number of records in the SYSIN data sets is recorded. Output devices are listed by an entry recording with device class, type, and address and the EXCP count for the data.[11]

The Interval Timer (or the hardward clock) is used to collect CPU time. It is incremented every 16.67 milliseconds. The Systems Management Tasks Force indicated that this is too long an interval and inaccurate job accounting may occur if it is to be used. They reported a need for a high resolution timer and also a real-time clock that is independent of the CPU button.[12] At present it is presumed there is no replacement for the Interval Timer.

Three sources of potential error in collecting CPU time with the current Interval Timer were disclosed by Douglas Aircraft.[13] These are:

> 1. The resolution of the Interval Timer (the hardware timer) is approximately 16.6 milliseconds. A job may receive control, execute some instructions and lose control all within a time interval and not be charged any time at all. On the other hand, a job may receive control just as the timer is decremented and the task is charged for more time than is used. Thus, the timing of each interval a job has control has a potential error of + or - 16.6 milliseconds.
>
> 2. When a channel is transferring data to or from main memory, the CPU could be delayed and channel interference will occur. The Interval Timer continues to run during this time.

---

[11] Ibid., p. 9.

[12] Noerr, op. cit., p. 5.

[13] Ralph Lawrence, "Timing and Running Jobs under MVT," SHARE Secretary Distribution, Form C5018 (July, 1968), p. 3.

3. When an I/O interrupt occurs the IOS (Input-Output Supervisor) processes the interrupt prior to the dispostion of the interrupt. Thus the interrupted task is charged for the interrupt even though the I/O may be for another task.

Wait time and clocking also provide a problem in the multiprogramming environment. CPU time is charged as the time available rather than time used. At present the highest priority job is charged for the total time it is in the computer (minus system task time). Each lower priority job is charged for the wait time of the job next above it in priority.[14] This also allows for inconsistencies in job timing within jobs of the same priority. Obviously job priority is then determined by the sequence in which they are loaded onto the machine. The cause of allocation of time in this manner is attributed to the need of deterring jobs that would lock up the system. It suggests a need for the use of possibly two internal clocks--one for runaway and one for accurate CPU counting.[15]

In addition to the use of time (CPU) as a "scale" for the allocation of charges, there are also supplementary measures that can be utilized. Three factors should influence the choice of the additional measurements. First, the measurements should relate accurately to the resources allocated to the job. Second, only measures that can be obtained easily should be used.

---

[14] Ibid., p. 2.

[15] Noerr, op. cit., p. 6.

Third, only measures that can be deciphered, without problem, by the user should be used.[16]

Some of the additional measurements that can be used are indicated below:

Elapsed Time, the time from the start to the finish of a job.

Compute Time, the actual CPU time used by a job.

Memory Time, the integral over time of memory space occupied by a job.

Productive Memory Time, the integral over time of memory space occupied by a job while CPU cycles were being used.

Input-Output Time, the sum of a job's I/O times for file references employing device.

Input-Output Number, the number of I/O operations initiated by a job for file references employing device.

Cards Read, the number of input cards for a job.

Cards Punched, the number of output cards for a job.

Lines Printed, the number of output lines for a job.

Set Ups required by a job (tape, reels, disk packs, etc.)

File Storage, the integral over time of storage space for a user's file on unit.[17]

Douglas Aircraft has suggested the use of essentially four of the above elements. They indicate the inclusion of the following list as additional parameters in their accounting

---

[16]Norman R. Nielsen, "Flexible Pricing: An Approach to the Allocation of Computer Resources," AFIPS Conference Proceedings Fall Joint Computer Conference, 33:524, Fall, 1968.

[17]Ibid., p. 523.

facility.

1. Job Step time
2. Region size
3. Number of tape data sets
4. Number of direct access data sets[18]

SMF does supply these parameters. As indicated previously, IEFACTRT receives step time and the sum of step times at Step and Job termination respectively. Also the amount of main storage requested and used is passed in the SMF record at each Step termination. I/O activity is reported as above. Additional measures can be implemented according to the amount of the SMF option used.

Note allocation of charge may not always be thought of as a linear operation of measurements taken. For instance, if there is a need for a huge amount of core by a job, this may deter all I/O activity at a particular installation. Even though the job requires very little I/O, a charge should be made for the "tie up" of I/O activity. This case suggests a quadratic rather than a linear allocation for core storage.

REENTERABLE CODE

The SMF options require that all exit routines be written in a reenterable manner. Reenterable code allows executing the same instructions at the same time by two tasks

---

[18]Lawrence, op. cit., p. 5.

which is necessary in an MVT configuration. SMF does allow the use of the GETMAIN and FREEMAIN macro instructions, both of which are reenterable. With the use of the user communication field a GETMAIN instruction can be used in the first exit. The address returned can be stored, passed, and used for all subsequent exit routines. A FREEMAIN would then be implemented in the last exit.

TESTING EXIT ROUTINES

Because exit routines will become part of the control program, it is imperative that they be thoroughly checked out before implementation of the SMF option. IBM does provide a procedure by which these exits can be tested. It is called TESTEXIT and can be found in SYS1.SAMPLIB.

TESTEXIT requires two changes in the option while it is being tested. First, subpool 0 must be specified in any GETMAIN instruction. Second, a special definition for SMFWTM must be given.

In order to implement the TESTEXIT procedure, four steps are necessary. First, exit routines must be placed in a partitioned data set. Second, a punched deck of TESTEXIT must be obtained. Third, the procedure followed for testing must be modified according to installation requirements found in the exit routines. Fourth, is the execution of the testing procedure.

Further information on suggested modifications, descriptions, etc., of the testing procedure may be found in IBM/360 Operating System--Planning for System Management Facilities manual.

CHAPTER IV

OTHER ACCOUNTING FACILITIES

MUSIC

MUSIC (Machine Utilization and Statistical Information Collection System) was developed by a member of the IBM World Trade Corporation (WTC) Field Systems Center in London, England. It is a set of modifications for MFT-II systems, which enable the gathering of more adequate accounting information than the accounting facilities presently supplied in OS/360.

Use of the system should be limited due to the incongruities existing between SMF and MUSIC. However, subsets of the package can be used. Items which can be promulgated and do not conflict with future support are: program CPU plus I/O interrupt time, maximum core used, job and step start and stop times, device allocation and step completion codes.[19] The addition of this data provides a far superior basis for billing, in contrast to what the simple accounting facility suggests.

---

[19]Beckerleg, "MFT-II Accounting Program (MUSIC)," (unpublished memo, IBM Headquarters, Poughkeepsie, New York, October, 1968), p. 2.

SMS/360

Systems Measurement Software/360 is a set of software products operating in IBM/360 OS and DOS environments. Two products are of particular interest. First, is PPE (Problem Program Efficiency). Second, is CUE (Configuration Utilization Efficiency). Both products are available for OS release 13 and greater. The software facilities were written by Boole and Babbage Inc.

PPE is concerned with measuring the efficiency of problem programs. It consists of two programs--the extractor and analyzer. The extractor must reside in the same partition as the problem program. It collects information about the problem program, and blocks it on tape or disk for further use by the analyzer. Degradation of time and space by the extractor are one to five percent and 5k respectively. Obviously, these are approximate values. The analyzer runs as a separate job requiring approximately 96k and five minutes. Information, such as the percentage of time spent in a particular section of the program, I/O Wait locations and times, and subroutines or macro usage are the types of data that PPE extracts and reports.

CUE operates in OS/360 under MFT and MVT configurations. It also is made up of an extractor and analyzer. The extractor of CUE operates as a separate job and collects information, such as CPU, channel and device utilization on random access device head movement, and on transient SVC usage.

The data collected is blocked and put on tape. The extractor adds approximately five percent overhead and used 6k storage.

The Analyzer processes the data collected by the extractor and outputs its findings in a report consisting of three major sub-reports (Equipment, Head Movement, and Supervisor Call). The extractor requires 96k and takes about five minutes for each hour of data extracted.

COMPUTER PERFORMANCE MONITOR II

Computer Performance Monitor II (CPM) is a hardward device, which extracts system measurement information. It was developed by Allied Computer Technology, Inc. and collects information by the attachment of measurement probes to the appropriate circuit pins on the computer. CPM allows for CPU clock cycles as fast as two hundred nanoseconds. Activities such as CPU wait time, execution time in supervisor mode, execution time in individual paritions, subroutine execution time, channel usage, device usage and frequencies and time of seeks on direct access storage devices are monitored, and information is collected about them.

An analysis and report program is supplied with the device. It supplies the installation with summary tables and histograms of the data collected according to the time interval specified. More detailed information can be reported by allocating a finer time interval to the report program.

CHAPTER V

SUMMARY AND CONCLUSIONS

In summary, Systems Management Facilities is an option, selectable at system generation, which monitors and collects information about jobs running in the system. The routines are divided into two parts--the first monitor and record information, while the second report and analyze the information.

SMF offers a great deal of installation flexibility within its framework. First, it permits a choice of tape or disk for storage of the SMF data set. Second, it allows the use of one or all of its five exits. Third, SMFDEFLT, which contains the options and parameters which control SMF operations are specified and may be altered at each IPL.

Special consideration must be given as to the amount of time and space available at the particular installtion. It seems that as these capacities increase, SMF becomes more tolerable. SMF increases main storage requirements for the control program. More auxiliary storage is required also. No estimate of degradation of throughput is available in the literature.

The Interval Timer (the hardware timer) still seems to be inadequate for timing and charging facilities. Consistent CPU times are available, however there are several sources of error in them. Consistent I/O times are impossible to obtain, and thus other means must be utilized in charging, if chargin is to remain at all consistent.

If total management of an installation is to be incurred, Periodic Job and Step data (Group 3) and System Use Profile Information (Group 4), as described earlier, must be provided. These groups of information, particularly Group 4, as a whole are not regarded in SMF. Supplemental routines such as those available from Boole and Babbage Inc. must be implemented to gather the information at this time.

BIBLIOGRAPHY

"Computer Performance Analysis." London, England: T. C. Hudson Associates Limited, 1969. (Pamphlet.)

"CPM II: The Second Generation Computer Performance Monitor." Santa Monica, California: Allied Computer Technology, Inc., 1968. (Pamphlet.)

"SMS/360 CUE (Configuration Utilization Efficiency) Product Summary." Palo Alto, California: Boole and Babbage Inc., 1968. (Pamphlet, Form S-17.)

"SMS/360 DPPE (DOS Problem Program Efficiency) Product Summary." Palo, Alto, California: Boole and Babbage Inc., 1968. (Pamphlet, Form S-21.)

Drummond, M. E. Jr. "A perspective on System Performance Evaluation," IBM System Journal, 8:252-264, Winter, 1969.

Lawrence, Ralph. "Timing and Chargin Jobs Running Under MVT," SHARE Secretary Distribution, Form C5018, July, 1968.

International Business Machines Corporation. IBM System/360 Operating System: Concepts and Facilities. IBM Systems Reference Library, Form C28-6535-4. Poughkeepsie, New York: International Business Machines Corporation, 1969.

International Business Machines Corporation. IBM System/360 Operating System: Introduction to Control Program Logic. IBM Program Logic, Form Y28-6605-3. Poughkeepsie, New Yrok: International Business Machines Corporation, 1967.

International Business Machines Corporation. IBM System/360 Operating System: MVT Control Program Logic Summary. IBM Program Logic, Form Y28-6658-2. Poughkeepsie, New York: International Business Machines Corporation, 1968.

International Business Machines Corporation. IBM System/360 Operating System: Planning for System Management Facilities. IBM Systems Reference Library, Form C28-6712-0. Poughkeepsie, New York: International Business Machines Corporation, 1969.

International Business Machines Corporation. IBM System/360 Operating System: Supervisor and Data Management Services. IBM System Reference Library, Form C28-6646-2. Poughkeepsie, New York: International Business Machines Corporation, 1968.

International Business Machines Corporation. IBM System/360 Operating System: System Programmer's Guide. IBM Systems Reference Library, Form C28-6550-3. Poughkeepsie, New York: Internal Business Machines Corporation, 1967.

"MFT-II Accounting Package--MUSIC: The Machine Utilization and Statistical Information Collection System." Poughkeepsie, New York: IBM Headquarters FSC, 1968. (Pamphlet.)

"MFT-II Accounting Program (MUSIC)." Poughkeepsie, New York: IBM Headquarters FSC, 1968. (Memo.)

Nielsen, Norman R. "Flexible Pricing: An Approach to the Allocation of Computer Resources," AFIPS Conference Proceedings Fall Joint Computer Conference, 33:521-531, Fall, 1968.

Noerr, J. M. "Status Report of the Systems Management Task Force," SHARE Secretary Distribution, Form C4819, August, 1967.

"SMS/360 PPE (Problem Program Efficiency) Product Summary." Palo Alto, California: Boole and Babbage Inc., 1968. (Pamphlet, Form S-16).

Stanley, W. I. "Measurement of System Operational Statistics," IBM Systems Journal, 8:299-308, Winter, 1969.

APPENDIX

APPENDIX A

DEFINITION OF TERMS

The following definitions were taken from the Glossary of IBM System/360 Operating System: Concepts and Facilities.

allocate: To grant a resource to, or reserve it for, a job or task.

asynchronous: Without regular time relationship; hence, as applied to program execution, unpredictable with respect to instruction sequence.

attribute: A characteristic; e.g., attributes of data include record length, record format, data set name, associated device type and volume identification, use, creation date, etc.

auxiliary storage: Data storage other than main storage.

block (records):
1. To group records for the purpose of conserving storage space or increasing the efficiency of access or processing.
2. A physical record so consituted, or a portion of a telecommunications message defined to be a unit of data transmission.

buffer (program input/output): A portion of main storage into which data is read, or from which it is written.

channel: A hardware device that connects the CPU and main storage with the I/O control units.

contents directory: A series of queues that indicate the routines either in a given region of main storage or in the link pack area.

control block: A storage area through which a particular type of information required for control of the operating system is communicated among its part.

control program: A collective or general term for all routines in the operating system that contribute to the management of resources, implement the data organization or communications conventions of the operations.

control section: The smallest separately relocatable unit of a program; that portion of text specified by the programmer to be an entity, all elements of which are to be loaded into contiguous main storage locations.

CPU (central processing unit): The unit of a system that contains the circuits that control and perform the execution of instructions.

data control block: A control block through which the information required by access routines to store and retrieve data is communicated to them.

data definition name (ddname): A name appearing in the data control block of a program which corresponds to the name field of a data definition statement.

data management: A general term that collectively describes those functions of the control program that provide access to data sets, enforce data storage conventions, and regulate the use of input/output devices.

data organization: A term that refers to any one of the data management conventions for the arrangement of a data set.

data set: The major unit of data storage and retrieval in the operating system consisting of a collection of data in one of several prescribed arrangements and described by control information that is accessible by the system.

dump (main storage):
1. To copy the contents of all or part of main storage onto an output device, so that it can be examined.
2. The data resulting from 1.
3. A routine that will accomplish 1.

dynamic area: That portion of main storage that is subdivided into regions or partitions for use by the programs performing job steps and system tasks. The dynamic area of storage is all the storage between the supervisor queue area and the link pack area.

entry point: Any location in a program to which control can be passed by another program.

extent: The physical locations on input/output devices occupied or reserved for a particular data set.

external reference: A reference to a symbol defined in another module.

external symbol: A control section name, entry point name, or external reference; a symbol contained in the external symbol dictionary.

fixed area: That portion of main storage occupied by the resident portion of the control program (nucleus).

initial program loading (IPL): As applied to the operating system, the initialization procedure which loads the nucleus and begins normal operations.

initiating task: The job management task of selecting jobs and preparing jobs for execution.

initiator/terminator: The job scheduler function that selects jobs and job steps to be executed, allocates input/output devices for them, places them under task control, and, at completion of the job, supplies control information for writing job output on a system output unit.

input stream: The sequence of control statements and data submitted to the operating system on an input unit especially activated for this purpose by the operator.

input work queue: A queue of summary information of job control statements maintained by the job scheduler, from which it selects the jobs and job steps to be processed.

installation: A general term for a particular computing system, in the context of the overall function it serves and the individuals who manage it, operate it, apply it to problems, service it, and use the results it produces.

job: A total processing application comprising one or more related processing programs, such as a weekly payroll, a day's business transactions, or the reduction or a collection of test data.

job control statements: Any one of the control statements in the input job stream that identifies a job or defines its requirements.

job management: A general term that collectively describes the functions of the job scheduler and master scheduler.

job processing: The reading of control statements from an input stream, the initiating of job steps defined in these statements, and the writing of SYSOUT messages.

job queue: (See input work queue.)

job scheduler: The control program function that control input job streams and system output, obtains input/output resources for jobs and job steps, attaches tasks corresponding to job steps, and otherwise regulates the use of the computing system by jobs. (See reader/interpreter, initiator/terminator, output writer.)

job (JOB) statement: The control statement in the input job stream that identifies the beginning of a series of job control statements for a single job.

job step: That unit of work associated with one processing program and related data. A cataloged procedure can comprise many job steps.

library:
1. In general, a collection of objects (e.g., data sets, volumes, card decks) associated with a particular use, and the location of which is identified in a directory of some type. In this context, see job library, link library, system library.
2. Any partitioned data set.

link pack area: The area of main storage that contains selected reenterable routines from SYS1.SVCLIB and SYS1.LINKLIB. The routines are loaded at IPL time, and can be used for all tasks in the system.

linkage: The means by which communication is effected between two routines or modules.

load module: The output of the linkage editor; a program in a format suitable for loading into main storage for execution.

macro instruction: A general term used to collectively describe a macro instruction statement, the corresponding macro instruction definition, the resulting assembler language statements, and the machine language instructions and other data produced from the assember language statements; loosely, any one of these representations of a machine language instruction sequence.

main storage: All addressable storage from which instructions can be executed or from which data can be loaded directly into registers.

MFT: Multiprogramming with a fixed number of tasks.

module (programming): The input to, or output from, a single execution of an assembler, compiler, or linkage editor; a source, object, or load module; hence, a program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading.

multiprogramming: A general term that expresses use of the computing system to fulfill two or more different requirements concurrently. Thus, it includes MFT and MVT.

multitask operation: Multiprogramming; called multitask operation to express parallel processing not only of many programs, but also of a single reenterable program used by many tasks.

MVT: Multiprogramming with a variable number of tasks.

nucleus: That portion of the control program that is loaded into the fixed area of main storage from SYS1.NUCLEUS at IPL time and is never overlaid by another part of the operating system.

nucleus initialization program (NIP): The program that initializes the resident control program. Through it, you may request last minute changes to certain options specified during system generation. The operator makes these changes through the console.

output stream: Diagnostic messages and other output data issued by the operating system or the processing program on output units especially activated for this purpose by the operator.

output work queue: A queue of control information describing system output data sets, which specifies to an output writer the location and disposition of system output.

output writer: A job scheduler function that transcribes specified output data sets onto a system output unit, indepently of the program that produced such data sets.

partition: A subdivision of the dynamic area that is allocated to a job step or a system task.

partitioned data set: Independent groups of sequentially organized data sets, each identified by a member name in the directory.

problem program: Any of the class of routines that perform processing of the type for which a computing system is intended, and including routines that solve problems, monitor and control industrial processes, sort and merge records, perform computations, process transactions against stored records, etc.

processing program: Any program capable of operating in the problem program mode. This includes IBM-distributed language processors, application programs, service and utility programs and user-written programs.

reader/interpreter: A job scheduler function that services an input job stream.

reenterable: The attribute of a load module that allows the same copy of the load module to be used concurrently by two or more tasks.

region: A subdivision of the synamic area that is allocated to a job step or a system task.

relocation: The modification of address constants required to compensate for a change of origin of a module or control section.

resource: Any facility of the computing system or operating system required by a job or task and including main storage, input/output devices, the central processing unit, data sets, and control and processing programs.

resource manager: A general term for any control program function responsible for the allocation of a resource.

seek: To position the access mechanism of a direct access device at a specified location.

sequential scheduling system: A form of the job scheduler that reads one input stream and executes only one job step at a time from that input stream.

service program: Any of the class of standard routines that assist in the use of a computing system and in the successful execution of problem programs, without contributing directly to control of the system or production of results, and including utilities, simulators, test and debugging routines, etc.

stacked job processing: A technique that permits multiple job definitions to be grouped (stacked) for presentation to the system, which automatically recognizes the jobs, one after the other. More advanced systems allow job definitions to be added to the group (stack) at any time and from any source, and also honor priorities.

supervisor: As applied to the operating system, a routine or routines executed in response to a requirement for altering or interrupting the flow of operations through the central processing unit, or fro performance of input/output operations, and, therefore, the medium through which the use of resources is coordinated and the flow of operations through the central processing unit is maintained; hence, a control routine that is executed in supervisor state.

supervisor queue area: The main storage area, adjacent to the fixed area, that is reserved for control blocks and tables built by the control program.

SVC routine: A control program routine that performs or initiates a control program service specified by a supervisor call.

synchronous: Occurring concurrently, and with a regular or predictable time relationship.

SYSIN: A name conventionally used as the data definition name of a data set in the input job stream.

SYSOUT: An indicator used in data definition statements to signify that a data set is to be written on a system output unit.

system input unit: A device specified as a source of an input job stream.

system macro instruction: A predefined macro instruction that provides access to operating system facilities.

system output unit: An output device, shared by all jobs, onto which specified output data is transcribed.

system task: A control program function that is performed under control of a task control block.

SYS1.LINKLIB: The partitioned data set that contains the IBM-supplied processing programs and part of the nonresident portion of the control program. It may also contain user-written programs.

SYS1.SVCLIB: The partitioned data set that contains the non-resident SVC routines, non resident error-handling routines, and access method routines.

task: A unit of work for the central processing unit form the standpoint of the control program; therefore, the basic multi-programming unit under the control program.

task control block (TCB): The consolidation of control information related to a task.

throughput: A measure of system efficiency; the rate at which work can be handled by a computing system.

turn-around time: The elapsed time between submission of a job to a computing center and the return of results.

user: Anyone who requires the services of a computing system.

volume: All that portion of a single unit of storage media which is accessible to a single read/write mechanism.

wait state (system): The condition of the CPU when all operations are suspended. This condition is indicated by a bit setting in the current program status word.

wait state (task): The condition of a task when it is unperformable because some event such as the completion of an I/O operation has not occurred.

work queue entry: The control blocks and tables created from one job in an input stream and placed in the input work queue or in one of the output work queues.

writing task: The job management task of transferring system messages and SYSOUT data sets from the direct access volume on which they were initially written to a specified output device.

APPENDIX B

TABLE OF COMMON EXIT PARAMETER AREA

The following table describes the contents of the parameters common to all exit routines.

TABLE III

COMMON EXIT PARAMETER AREA*

Displacement from Pointer	Length (in bytes)	Data Format	Description
0	8	EBCDIC	Job Name
8	4	Binary	Reader start time (in hundredth of seconds) for the job.
12	4	Packed decimal	Reader start date for the job. Format is OOYYDDDF. (F is a sign.)
16	4	EBCDIC	System identifier (2 bytes) and model number (2 bytes).
20	8	EBCDIC	User identification field.
28	4	Binary	Number of the step being processed, in high-order byte. The other 3 bytes are reserved.
32	4	Binary	User communication field.

*IBM System/360 Operating System: Planning for System Management Facilities (IBM Systems Reference Library, Form C28-6712-0. Poughkeepsie, New York: International Business Machines Corporation, 1969), p. 17.

PROBLEMS INVOLVED IN SYSTEMS MANAGEMENT FACILITIES

by

THOMASYNE MAE CAMPBELL

B. A., Rosary College, 1967

AN ABSTRACT OF A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Statistics and Computer Science

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1970

System measurement is concerned with gathering statistics about the operating system performance and jobs running under the control of the system. It was the objective of this paper to research and analyze operating system measurements. The historical basis of operating system measurement, its purposes and problems, and the necessary statistics, were discussed.

In particular "Systems Management Facilities" (SMF) was analyzed. It is a system measurement option of the System/360 operating system, applicable to multiprogramming with a variable number of tasks configuration. Analysis was done with regard to problems such as data set requirements, additional system overhead incurred, timing and allocation of charges, re-enterable code, and testing of the option. In addition to the above discussion, a brief description of other accounting facilities was given. They included System Measurement Software/360 (SMS/360), Machine Utilization and Statistical Information Collection (MUSIC), and Computer Performance Monitor II (CPM II).

SMF was found to be an extremely flexible system. Data set requirements of the option were provided adequately. It seemed as the amount of time and space available increases, the option becomes more desirable. The time of individual jobs cannot be measured accurately, thus billing is still a

very difficult task. Supplementary facilities must be employed if all necessary information is to be collected.