MPL-8 March 24, 1969

TO: Multics Performance Log

FROM: R. J. Feiertag

SUBJECT: Multics System Performance Certification Records

In order to maintain a log of performance information for evaluation of Multics systems, a standard information sheet has been written. This information sheet is called the "Multics System Performance Certification Record." The information recorded on this sheet is based upon data extracted from certification runs. A short description of the items on the sheet follows.

The running of the certifier is described in MOSN-26 and MSPM BV.7.03. There are three sources of information which must be consolidated to produce the Certification Record.

1. The console output of the user performing the certification run.

2. The console output of the processes created during the certification run.

3. The results of system metering printed on the on-line printer.

Each run is divided into two parts. The first part is that of creating and initializing processes used to perform the test. As the second part, a test script of commands is run on each created process. All metering is done separately for the two parts of the run thereby producing the two sets of numbers under "Measurements".

A description of each item now follows:

I Variable Settings: operating system

System being certified: This designation identifies the system being tested. It will either be an MSS or MTS number. Information describing each system can be found in Multics headquarters.
Certifier used: The certifier is a collection of programs which perform the actual certification test on the system, i.e., run the user scripts and make the appropriate measurements. Each of these collections of programs is given a distinct designation and the information necessary to determine the programs associated with each designation is kept in Multics headquarters. Usually the certfier used will be the one associated with the system being tested. However, in certain cases, the certifier included in the system may be replaced by a different version of the certifier. This change will be indicated under this heading.

Script used: The script is just the list of commands which are executed during certification. The list of commands associated with each script designation can be found in Multics headquarters.

Number of processes used: The certifier creates some number of processes and each process executes a script. The number of processes created will be printed on the user's console.

Typewriter output: The certifier can direct the output of the created processes either to a typewriter or to a file. The use of typewriters for output will be indicated on the user's console.

Number of lines output: This is the total number of lines output by the created processes to typewriters or files.

System Segment Table Size: This is the length, in decimal words, of the System Segment Table used in the system under the test. The length of the SST will be printed on the user's console.

Number of permanently wired pages: This is the number of blocks of core which are permanently wired. This figure will be printed on the user's console.

Maximum number of processes eligible for multiprogramming: This is the maximum number of processes, other than system processes, which may be simultaneously eligible for multiprogramming. It will also be printed on the user's console.

Maximum number of processes which may be loaded: This is the maximum number of processes, other than system processes, which may simultaneously remain loaded and is printed on the user's console.

Scheduling Quanta: This is the amount of time a user may run before losing his position in the queue that he is presently in. A separate quantum is assigned to each queue and the time for each queue is printed on the user's console.

II Hardware Configuration:

Amount of Core Memory; Number of Processors, Firehose Drum, Disk, Date of Certification Run, and Time of Certification Run: These items will all be printed on the user's console.

IV Measurements

a. CPU Time Breakdown: The times listed in this section are all extracted from the Traffic controller data block.

1. Time used by subject processes: During process creation this is the total of the amount of time each created process had a processor in order to initialize itself and the amount of time spent in the initializer creating the processes. The amount of time charged to each created process and the initializer during process creation will be printed on the user's console. During command sequence this is the total time of the four processes. These times will also be printed on the user's console.

2. Time spent loading processes: This is the time charged to the Loader Daemon and is printed on the user's console.

3. Time spent in File System Daemon: This is the time charged to the File System Daemon, it is printed on the user's console.

4,5,6 Idle Times: The time the system spends in the idle process is broken down into three categories. True idle time occurs when the system actually has nothing to do. This means that no processes are either ready or waiting. It is also possible to have idle time when there are processes on the ready list. This occurs when all those processes which are eligible are in the wait state and those processes on the ready list which cannot become eligible, are prohibited from running. This idle time is the price that is paid for having multiprogramming control and is the idle time due to eligibility control. There is a third idle time which does not properly fall into either of these categories. This occurs when no processes are on the ready list and there is a process in the wait state. This is not true idle time because even though no processes are ready there is someone being served even though he happens to be waiting. Also this is not idle time due to eligibility control because no one is being kept from running because he is not eligible. It is possible that if someone were ready he would be able to run. These three times will be read from the user's console.

b. Breakdown of CPU times used by subject processes--these times are extracted from the system meter table and the traffic controller data block.

1-5 Fault Times: The times mentioned here are the total times necessary to process each type of fault for the created processes, the loader daemon, and file system daemon. The time for each type of fault is measured exclusive of the time necessary to process any faults which may occur during its processing. These times can be read from the output of the on-line printer under the heading of "TIME IN METERED PROCESSES".

6. Non-fault time: This is just the amount of time spent not processing any of the above five faults. Since there is no direct way of measuring this time it is obtained by subtracting the sum of the above fault times from the total time. The total time can be obtained from, but is not equal to, the total time of section IVa. Since this section is concerned with faults in all processes except the initializer, the total time here is the actual total time minus the sum at the initializer time and idle times. The actual total time, the initializer time, and the idle times will be read from the user's console.

c. Fault times and number--these figures are read from the system meter table. Here is recorded the total number of each type of fault and the average time to process each type of fault again exclusive of the time necessary to process faults occuring during its processing. These times are also measured for the created processes and the loader daemon and file system daemon. The numbers are read from the on-line printer output under the headings of "AVG. FAULT TIMES" and "TOTAL NUMBER OF FAULTS".

d. Average times seen by a process--these times are derived from the traffic controller data block and the output of the created processes.

1. Average real time for completion of a process: Real time is the actual time necessary to perform a task or command. The real time for completion of a process is the actual time necessary to perform the certification script for each process. This time is, of course, highly sensitive to the number of processes executing in the system. The real time for each process is printed at the end of its console output. These times are averaged to obtain this number.

2. Average process creation time: This is the average time necessary to create and initialize a process to command level. It is obtained by summing the process initialization times for each process and the initializer time for creating the processes as printed on the user's console and dividing by the number of created processes.

3. Average time for execution of command sequence: This is the average CPU time charged to each process for executing the test script. It is obtained by averaging the times charged to each created process during command sequence execution as printed on the user's console.

4. Time for CTSS to execute same command sequence: The CPU time necessary to execute the equivalent of the present certifier script has been measured using the PDP-8 simulator.

V Output of original run may be found in file labeled:

The original output of all runs will be kept on file in Multics headquarters. Each run will be filed numerically according to the number of the system being tested. Any deviation from this filing technique will be reported here.

This concludes the general description of a Certification Record. Some of the items above are not retrievable in the present certifier, so some areas must be left blank and others must be approximated by modified calculation techniques.

Most of the information under "Variable settings" and "Hardware configuration" are kept in various data bases and initialization routines. In the future the certifier will collect all this information and print it out, but at present the information must be determined by a search through miscellaneous listings.

The calculation of certain items under "Measurements" has been modified to fit the information available with certifiers "multics_test_c" and "multics_test_d".

a. CPU time breakdown--none of these numbers are available with the present certifier, however, the time used by subject processes during command sequence can be obtained by summing the Compute Times printed at the bottom of each created process's console output. The total CPU time charged during command sequence can be approximated by taking the difference between the last wait message printed on a created process's output and the first wait message printed on a created process's output.

b. Breakdown of CPU times used by subject processes--the non-fault and total times of this section are not directly calculable, however, the time used by subject process calculated in part (a) is a good approximation to the total time during command sequence of part (b). The non-fault time during command sequence can be derived from this by subtracting the sum of the measured fault times.

d. Average times seen by a process--the average process creation time can not be determined, but the average time for execution of command sequence can be calculated by dividing the time used by the subject processes during command sequence of part (a) by the number of created processes.