

February 1, 1969

TO: Multics Performance Log
FROM: J. H. Saltzer
SUBJECT: CTSS Performance on trivial requests

As a benchmark against which to compare Multics performance on trivial interactive tasks, some statistics about CTSS performance are provided.

A desk calculator command, in the form of a private "saved" file, is available on CTSS. It is capable of doing simple addition, subtraction, multiplication, and division of integers, and of remembering partial results by name for later use as operands. The computation required to satisfy any given request is about as trivial as for any CTSS command, requiring that the program be swapped into core memory, call the supervisor to read a single input line, scan the line and determine and do the computation wanted, and occasionally write out a line in reply.

The performance of CTSS in executing this program has been monitored by examining ready message statistics following uses of the command, and counting the number of interactions during the use of the command. Ten different "Production" uses of the command were monitored, involving about 400 interactions, on different days when CTSS load ranged from 10 to 19 users. The statistics examined in detail show variation of no more than 10% between runs with different numbers of users.

In summary, we observe the following:

(numbers in decimal)

Basic program size before loading	471 words
Size of loaded core image including 12-bit typewriter I/O library and support library	1241 words
Average CPU time charged per interaction	12 milliseconds
Average swap time charged per interaction	145 milliseconds

(numbers in decimal)

Disk swap time charged for the first interaction, if command has been used recently (sticky register avoids directory search) 260 milliseconds

Disk swap time charged for first interaction, if command has not been recently used and directory search is necessary 600 milliseconds

Thus the cost of each interaction with the desk calculator, once one gets started, is $145 + 12 = 157$ ms. At the current CTSS charging rates, this cost amounts to about 1 cent per interaction, plus about 4 cents to invoke the command.

Comments

1. The intrinsic capability of the 645 hardware should allow more than an order of magnitude improvement in the cost per interaction, since a multiprogrammed firehose drum should not require 100% of the machine for 145 milliseconds to get a 2000 word program (plus 1000 words of machine conditions) into core. One might expect to have to pay, in the best situation, for about 6 to 10 ms. of cpu (20% of the machine) and about 30 ms. of 3000 words of core tie-up (2% of the machine) while waiting for the program to arrive.
2. Multics system 1.8, with repeated use of a program so that page faults do not occur, seems to require on the order of 600 ms. of computation time alone for a trivial interaction. If computation for 20 page faults at 15 ms. each is added, the time is closer to 900ms. It appears that even if a factor of three is obtained in both of these figures (from I/O system typewriter DIM improvement and from page fault/file system improvement, respectively) it will still be necessary to work this area over some more. In particular,
 - a. Auditing of the computation path to locate the bottlenecks,
 - and b. Reorganization of the per process data bases to reduce the number of missing page faults, along the lines that have been recently proposed

will probably be necessary.