MPL-45

TO: Multics Performance Log

FROM: A. Sekino

SUBJECT: Results of Daily Multics Performance Measurements
(Period: December 1 through December 4)
and Some System Operational Statistics

DATE: December 11, 1969

The recent results of daily Multics performance measurements are given below. During the measurement period, the operational statistics are usually collected using "hmu", "rm", "tcm", and "fsm" in a second process. The typical statistics obtained by these commands are given in Table I and it was found that these statistics clearly show the degradation of the heavily loaded system, as shown in Figure I.

1. Results of Daily Measurements

time unit = second
average = per interaction

Date	Run No.	No. of users	System	Total CPU time	Total No. of P. F.	Total Real Time	Average CPU Time	Average No. of P. F.
12/1	MPM71*	26	4.7f	--	--	--	--	--
12/1	MPM72*	21	4.7f	--	--	--	--	--
12/2	MPM73	22	4.7f	47.558	2465	2670	.720	37
12/3	MPM74	28	4.7f	51.085	2631	2730	.774	39
12/4	MPM75	24	4.9	48.353	2626	2490	.732	39
12/4	MPM76*	23	4.9f	52.826	2723	2730	.800	41

*Only the first half of the script was successfully run.

**This measurement was done in the evening (i.e., 22:21 - 23:07). It seems that the evening users are much heavier than the daytime users.

2. Typical System Statistics

The system statistics were collected during the daily performance measurement period. Some of them are given below for two cases, namely the 24 user case and the 27-28 user case of the 4.7f system.

Table I Typical System Statistics (4.7f)

No. of users	24	27 - 28
obtained from:	MPM67, 68	MPM66, 74
PDP8 user		
total CPU time	49.3 sec.	50.3 sec.
total No. of P.F.	2520	2570
total real time	2610 sec.	2640 sec.
paging rate	38.3%	38.4%
tcm		
ave. queue length	8.7	14.0
total idle time	36%	50%
multi-prog. idle time	36%	50%
other idle times	0%	0%
M.T.B. Interactions	2.4 sec.	3.3 sec.
$P\{P.T. < 4 \text{ sec.}\}$	96.5%	96%
fsm		
drum reads	24.0/sec.	19.9/sec.
drum writes	12.9/sec.	10.8/sec.
total drum access	36.9/sec.	30.6/sec.
disk reads	2.25/sec	2.83/sec.
disk writes	1.61/sec.	1.62/sec.
total disk access	3.86/sec.	4.44/sec.

paging rate : Portion of time spent for paging ($= \frac{\text{total No. of P.F.}}{\text{total CPU time}} \times 7.5 \times 10^{-3}$)

M.T.B. Interactions : Mean time between interactions observed by the system

$P\{P.T. < 4 \text{ sec.}\}$: Probability that the processing time for each interaction is less than four seconds.

3. Degradation of the Heavily Loaded System

The system statistics collected so far clearly show the way in which the system is degraded. They are plotted as functions of the number of simultaneous users in Figure I. Some observations on Figure I are given below (see Table I also).

(a) Average CPU queue length: The length increases from 6 (at 22 user point) to 14 (at 28 user point).

(b) Multi-programming idle time: The percentage of the multi-programming idle time goes up from 30% to 50% as the number of simultaneous users increases from 20 to 28.

(c) Other idle times: All of them stay around 0% in the same range of the number of users.

(d) Drum access and Disk access: As the number of users increases from 24 to 27-28, the total number of times that the drum is accessed decreases, while the total number of times that the disk is accessed (expecially read access) increases. It seems that the disk transfer rate is almost its maximum (see Talbe I).

(e) Mean time between interactions: As the number of users increases, the mean time between interactions measured by the traffic controller metering procedure increases. This means that the input rate of the commands from a user is almost completely controlled by the system output rate (i.e., the slow system response time).

(f) These observations suggest that the computational capability of the system substantially decreases as the number of users increases from 20 users to 28 users, which is the normal range of the system load. Therefore, it seems that the system is a little over-loaded. However, the installation of high-speed disks is expected to improve this situation; it is expected to reduce the CPU queue length and the multi-programming idle time.

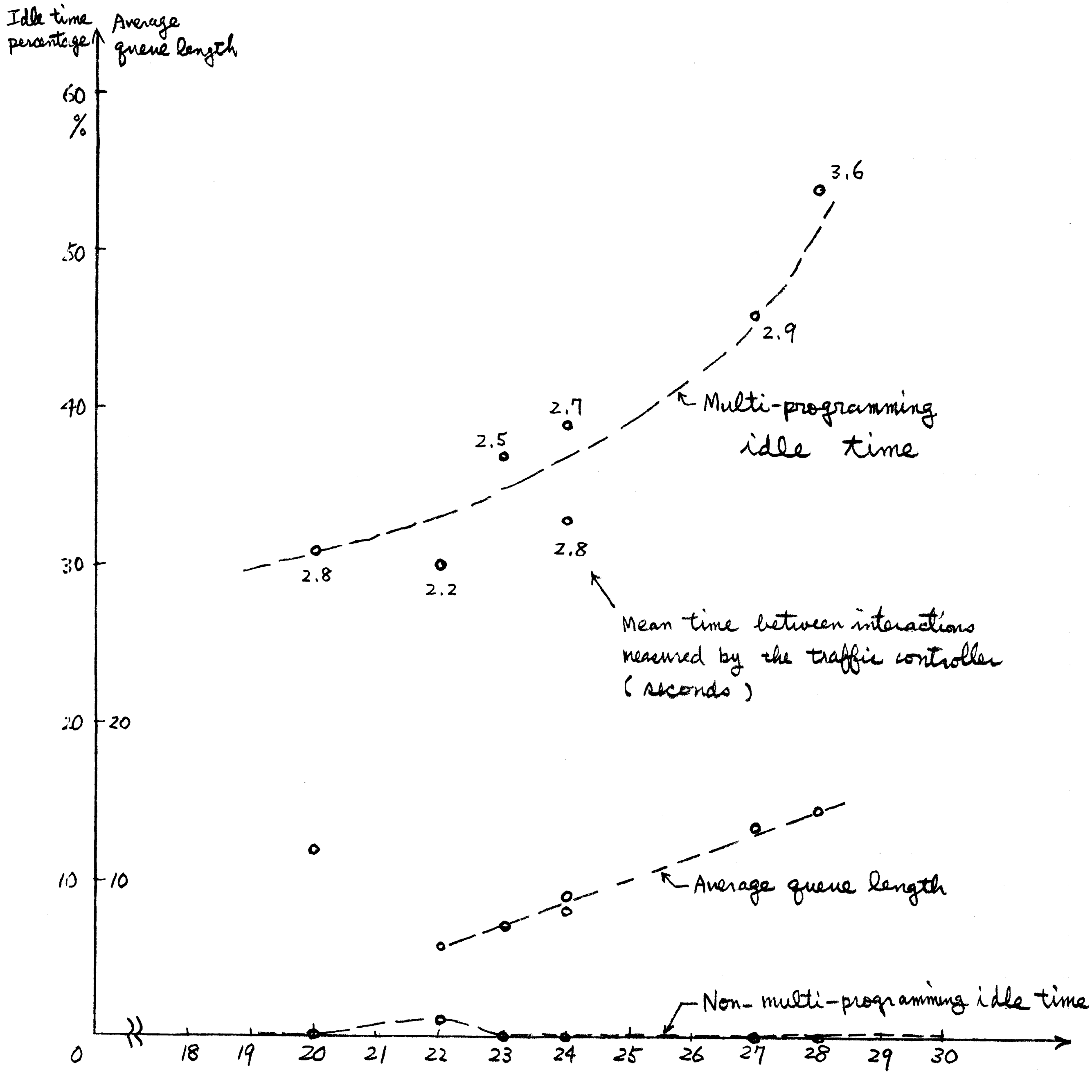

Fig. I. CPU idle times, average queue length, and mean time between interactions as functions of # of users