QUESTIONS ON MULTICS

Answered by J. H. Saltzer

(Edited by H. Aiso)

Electrotechnical Laboratory

Editor's Note: This "Questions and Answers" on Multics is based on a series of lectures, which were given by Prof. J.H. Saltzer of MIT in summer of 1966 in Tokyo.

Page

1st Lecture (Aug. 16)

1. Q (H. Nishino, ETL):

Why is the associative memory attached to CPU ? I think it is also possible that the associative memory may be attached to the memory controller.

A:

All page and segment hardware logic is built into the Central Processing Unit for three reasons:

a. Complex logic is centralized in the processor module, increasing overall system reliability.

b. Page and segment logic is not needed for GIOC and Drum controller interface to the memory controller.

c. Access control information, generated in the CPU and needed on a per - CPU basis is stored in the associative memory.

2. Q (M. Hosaka, Univ. of Tokyo):

(1) How about the size of page ?

A:

Hardware facilities allow page of 64 to 1024 words. The superviser can simulate intermediate page sizes.

Q:

(2) How did you choose that size ?

A:

The page sizes are really minimum and maximum practical sizes, to permit experimentation.

Q:

(3) Does user see segments when programs are not written in machine language ? Usually which one assign segment, user or machine ?

A:

The programmer can think in terms of segments even when he is writing in high-level languages. The supervisor program assigns segment numbers to the programmer's segments, but the programmer is free to break up his program into segments as he chooses.

3. Q (I. Kimura, Univ. of Tokyo):

Is the supervisor visible to the user ?

A:

The users has many closed subroutine entry points available to him to perform various functions for him. Parts of the supervisor, for example, the missing page procedure, are not visible to the user.

4. Q (T. Yoshimura, Keio Univ.):

(1) Give us detailed description for "Missing Page Fault" concept.

A:

The page table may contain either a block address or a missing page bit in each of its entries. If a block address appears, the processor will automatically shift the address reference into the appropriate block. If a missing page bit appear, the processor will instead fault to the missing page procedure, which can then read the missing page into an appropriate block of core memory, replace the missing page bit (in the page table) with a block address, and restart the program.

Q:

(2) In what cases does supervisor set the extra bit in Page Map ON/OFF ?

A:

The missing page bit is set on for pages for which the programmer has not yet referenced or has not used for a long time.

5. Q (T. Uraki, Hitachi):

GE tried to simulate in order to determine number of associative registers. How much performance can be expected by using 16 associative registers, in comparison with by using 8 associative registers ?

A:

The precise improvement to be expected by increasing the associative registers is unknown, since we have no experience with procedure written for the segmented, paged environment. Rough measurements based on Fortran IV programs for the 635 computer have indicated that the average time/memory reference decrease slowly as the number of associative registers is increased beyond 16.

The question is one which we are very interested in learning the answer with programmer designed for 645.

6. Q (E. Wada, Univ. of Tokyo):

(1) When is the symbolic name of a segment translated to the segment number ?

A:

At the time of first reference to the segment by the procedure. If the particular data input to the procedure cause it to not reference some segment, that segment will never be assigned a segment number.

The technique of linking programs together dynamically is discussed in the FJCC 1965 papers, and will be reviewed later in this lecture series.

Q:

(2) As for the second reason of the paging, is the "enough interactive program" paged automatically, or does the programmer do something ?

A:

The supervisor (Basic File System) will automatically page the segments of an interactive program without any advice from the programmer. The supervisor can accept advice (which it may ignore) from the programmer concerning pages which will be needed soon or will not be used again.

2nd Lecture (Aug. 17)

1. Q (I. Kimura, Univ. of Tokyo):

Why has the linked list of memory in use backward links ?

Is it impossible to avoid it by shifting the used blocks to the tail of the list ?

A:

One can probably devise an algorithm by which the memory blocks are linked in one direction only, as in your suggestion. The double linked list has the advantage that it contains self-checking (redundant) information, which may be used to salvage the data base of core control, should a mishap (hardware or software) occur.

For example, following a system "crash", one could scan the linked list, checking for correctness of the back pointers. If they are all correct, one would be confident that this data base was not damaged. If there is an incorrect back pointer, this is a warning that perhaps core memory should be reloaded.

2. Q (H. Nishino, ETL):

Please explain the functions of the flags which the directory contains.

A:

The flags are interlocks which indicate, for example, that some user is reading the file, and that others should be careful before attempting to write in it.

The precise list of flags not yet completely specified.

3. Q (S. Doshita, Univ. of Kyoto):

Is it necessary to have any synchronization between scan of link listing and the scan of program execution among users ?

A:

The link list scan is completely independent of central processor switching among users. Generally, the processor will switch from one user to another every 2 - 100 ms., while the storage block list might be relinked every 500 - 5,000 ms.

4. Q (H. Aiso, ETL):

(1) How many blocks do you need between "panic threshold" and "no core free" ?

A:

The number of blocks (exact number is yet unknown) depends on the number of cascaded interrupts which the system can encounter simultaneously. (Also of significance is the number of free blocks of core needed for temporary storage by the procedure.)

Q:

(2) Who decides the values of the two thresholds ?

A:

The panic threshold is determined mechanically from consideration of the system configuration and the "worst case" interrupt condition.

The refresh threshold is determined experimentally.

It is set so that the panic threshold is rarely encountered.

Conceivably, one could have the system dynamically set the refresh threshold as it "learns" of the characteristics of its users.

With our present state of knowledge, this approach would be very risky.

5. Q (K. Nitta, NEC):

Q:

As for the access right in access control list, how is it different between "read" and "execute" for users ?

A:

Access to "read" without access to "execute" would allow a file to be used as a source of data but not as a procedure segment.

Access to "execute" without access to "read" would allow the file to be used only as a procedure segment. This combination would probably be useful for a "proprietary" program for which there is a charge for each use.

6. Q (A. Fujii, Oki):

(1) How much values do you take now, about the refresh and panic thresholds.

A:

The thresholds have not been established. In the initial version of the operating system, they are parameter which may be adjusted by the system staff while the system is running. At some future time the system might attempt to adjust the parameter itself.

Q:

(2) Are there any special figure of Drum Control Words compared with other control words for instance control word for I/O or control word for transmission line ?

A:

The drum controller is a separate design from the GIOC, and data and data control words are of completely different format.

However, the function of the data control words is basically the same.

7. Q (T. Yoshimura, Keio Univ.):

Why is it necessary to have two queues (software queue and hardware queue for the management of a drum ?

A:

The software queue is a compact overflow queue for the case in which the more extended hardware queue would exceed a reasonable size. The hardware queue must be placed in absolute core memory (not in a paged segment as is the software queue) and therefore changing its size is awkward.

8. Q (E. Wada, Univ. of Tokyo):

(1) All use bits in page table must be cleared after the linked list is sorted. Is it right ?

A:

When the linked list of pages in use is resorted, all use bits are also set to zero.

Q:

(2) By which module is drum file map created ?

A:

The drum file map is created and managed entirely by the Drum Interface Module. In this way an efficient format may be used, but drum characteristics are confined to the DIM.

Q:

(3) Each hardware queue table entry corresponds to a definite sector, I understand.

So, is the size of hardware queue 16 ?

A:

The hardware queue is extended to whatever length seems appropriate to the DIM. Probably enough entries are maintained to keep the drum active for 2 or more revolutions.

Q:

(4) Which central processor does sorting of the linked list ?

A:

The linked list may be sorted by any processor ... whichever one discovers that it is time to resort it.

3rd Lecture (Aug. 18)

1. Q (T. Yoshimura, Keio Univ.):

What is the relation between dump capabilities in Basic File System and conventional "checkpoint" capabilities in existing programming systems ?

A:

Certain programs will find that the backup facility of the file system can replace the usual checkpoint facility.

Then programs include these which compute small amounts at widely separated intervals, and need "checkpoints" to guarantee safety of data between computations.

Programs which compute for 3 hours, taking checkpoint requests with requests to the file system to "back-up" certain files. In these cases checkpoints are programed basically the same as before.

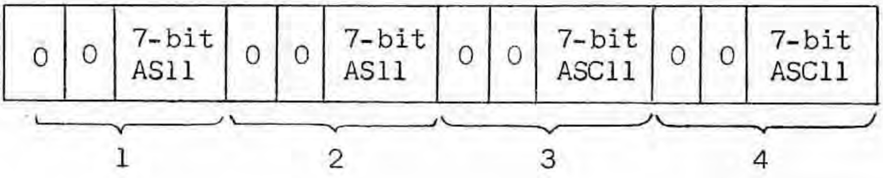
2. Q (T. Uraki, Hitachi):

The ASCII is originally a seven-bit code. On the other hand, the word size of the GE 645 is 36 bits. Then, it seems that six-bit characters can be easily handled.

What kind of code do you adopt as an internal code ? What is the size of the internal code ?

A:

Internally, the seven bit ASCII character are stored right-adjusted in a 9-bit field, four 9-bit fields per word. The loading bits in each 9-bit field are zeros.



We feel strongly that the advantages of a large character set outweigh greatly consideration of packing six characters/word.

3. Q (K. Nitta, NEC):

I would like to know the reason why incremental dump tapes should be kept forever. Because, if complete dump tape includes all information on storage, the incremental dump tape containing the previous information of the latest complete dump seems not to be necessary.

A:

One could have a choice of keeping either all complete dump tapes or all incremental dump tapes. (Since dump tapes are the bottom level of multi-level storage system, one of these must be kept forever.) It is expected that the volume of incremental dump tapes will be smaller, since they must contain duplication of files or used successive weekly complete dump tapes.

4. Q (E. Wada, Univ. of Tokyo):

(1) When the whole drum and disk memory were reloaded from the dumped tape, are the users on the remote consoles informed of it ?

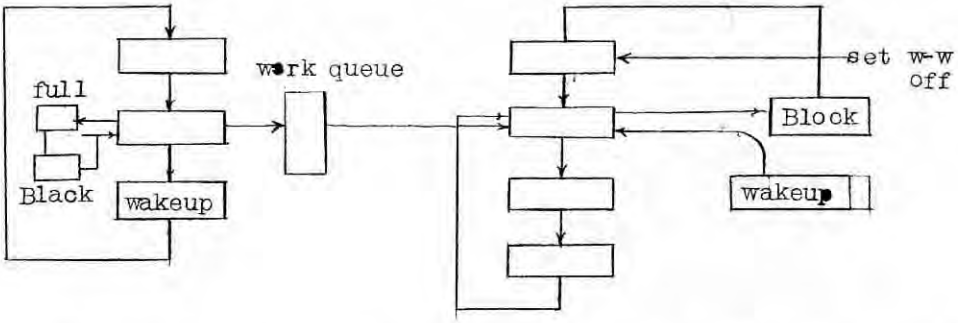
A:

Users are informed of a disk reload in one of several ways:

a. Files which could not be reloaded are "trapped" so that a reference to them will get the user a comment.
b. If files could not be reloaded, a message is placed in the user's file directory, giving him a list of files so lost.
c. A general message, which automatically prints out when any user logs in, is inserted in the system.

Q:

(2)



My suggestion is regarding to "Block and Wake-up" and "Wake-up Wait" strategy:

i Wake-up Wait switch is set "on" only if wake-up is ignored. I have not decided yet how to design if w-w is already "on" at this time.

ii Wake-up wait switch is set "off" when Block senses this switch being "on" and returns immediately. The purpose of this suggestion is to remove "Set w-w switch off" box in process B, although I have not yet considered whether this system works correctly.

A:

The technique works, if process "B" is prepared to accept an occasional call to Block which returns immediately despite an empty work queue. (This situation will occure if "B" works ahead in the queue without between quene entities.)

Q:

(3) Regarding to the canonical form conversion, how is "Back Space" used to erase or rub out mistyped characters ?

A:

The "backspace" character is exclusively interpreted to mean move carriage back in preposition for "overstriking", not for retyping. A separate convention is used for erasing mistyped character.

4th Lecture (Aug. 19)

1. Q (T. Yoshimura, Keio Univ.):

How does the supervisor know that a process is authorized to order Quit other processes ?

A:

The validation of calls to Wake-up and to Block is carried out in a higher layer of the supervisor. The traffic controller module makes no check on authorization.

2. Q (T. Uraki, Hitachi):

(1) Is it possible that the inter-process communication is done between different users ?

A:

If the users consent, inter-process communication is allowed. The pattern of inter-process communication between users is simpler than between processes of a single user, since there is no need for a call-passing mechanism.

Instead, the wait coordinator, and a procedure named "Intercom" which checks validity, are all that are used.

Q:

(2) Is the system clock an incrementing counter or a decrementing counter ? If it is an incrementing counter, it seems to be reasonable that an interrupt occurs when

the content of calendar clock is larger than that of a alarm clock.

A:

The system clock increments. The interrupt is generated whenever the alarm clock contains a number smaller than or equal to the system clock. (The inequality sign was reversed by mistake in lecture.)

3. Q (H. Yamamoto, KDD):

What does 'a priority "Label"' mean ? Is it related to the level of "Queues'" ?

A:

The label is an integer priority value for the process. In the case of multi-level scheduling algorithm, the priority label is the queue number. The label is merely a method of identifying where in the queue to place a new process.

4. Q (H. Kawai, ETC):

Are the interrupt cells checked by CPU every memory reference ?

A:

The Central Processor checks for an interrupt signal between every pair of successive memory references. Thus it is possible for the CPU to be interrupted between a segment descriptor word access and a page table word access. For this reason, a substantial amount of information about the processor state must be stored at the instant of interruption.

5. Q (A. Fujii, Oki):

(1) How many descriptor base registors does CPU have ?

A:

Each CPU has 8 descriptor base registors.

Q:

(2) What is the special relation between a paticular CPU and memory control at the interrupt operation ?

A:

The 645 system hardware forces us to designate one processor or "control" for each memory controller; that processor takes all interrupts directed to that memory controller.

Another processor may be control for another memory controller. The work implied by the interrupts is distributed among the processors by the process exchange.

Q:

(3) Are there any paticular hardware corresponding to "Process Exchange" ?

A:

The Process Exchange uses the "load descriptor base register" instruction, and the system interrupts in its operation. However there is no special hardware designed to lighten the work of the process exchange. It is interesting to speculate on which parts of the process exchange might be implemented by a hardware device.

Q:

(4) How many program steps is "Process Exchange" composed of ?

A:

As yet, this is unknown. The bulk of the code is in the scheduler. From 7094 experience, the scheduler might consist of 100 PL/I program statements.

6. Q (E. Wada, Univ. of Tokyo):

(1) When a process calls the Swap-DBR subroutine and the processes are switched, how is the return address of the former process remembered ? Does "call stack" have a special segment number ?

A:

The stack pointer of the call stack is stored in the process table entry for the process. Since all processes probably share a common linkage segment for the basic supervisor, they will in fact all use the same segment number for the call stack when operating in the supervisor. Such organization is not required, however.

Q:

(2) Does the box "switches to address space of process being scheduled" use the Swap-DBR subroutine ? If so, does this subroutine work differently when called by this box ?

A:

Swap-DBR has two entry points, named "Swap-DBR", and "Ready-Him". "Ready-Him" is used to wake-up another process. Both are placed in the same segment so that only 1 segment must have a common segment number in all processes.

Q:

(3) Explain precisely the Quit box in the process exchange.

A:

The Quit module operates as follow; (Process "A" calls Quit (B) to turn off process "B".)

a. If "B" is already blocked, do nothing.

b. If "B" is ready, remove "B" from ready list.

c. If "B" is running, set a "Quit interrupt" for "B".

If a process receive a Quit interrupt, it calls Block.

Q:

(4) Does the process table have the entry of processor number for running processes in order to set Quit interrupt to the processor which is processing the process to be quitted ?

A:

Yes.

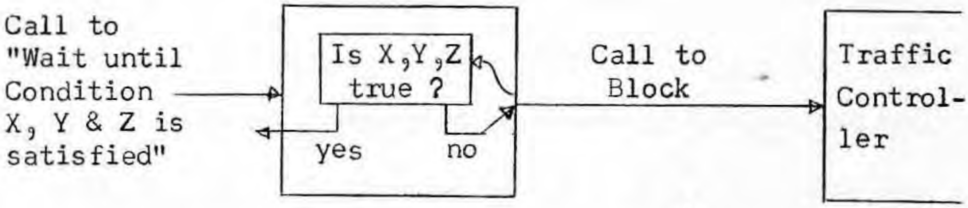
5th Lecture (Aug. 20)

1. Q (I. Kimura, Univ. of Tokyo):

Is it possible to start a computation in response to the completion of two other computations ?

A:

The Traffic controller only provides the primitive facility to Block until any signal arrives. The user may program a more elaborate procedure thus:



This organization is chosen so that the user may specify an arbitrary condition for wake-up. No assumption about the user's needs is built in to the Traffic Controller.

2. Q (H. Kawai, ETL):

(1) When an external interrupt occurs, it takes too much time to pass through Process Exchange. How do you avoid that ?

A:

Part of the cost of generalized organization is an increase in the overhead cost of running the system. The generalization usually pays for itself, however, in a simple system to maintain, debug, and operate.

In the case of interrupt handling, it is not expected that the time required to call "Wake-up" will be significantly longer than in any other organization of a system in which interrupt handling is not given special priority. In fact, the system is generally organized to minimize the number of interrupts which arrive at the system.

Q:

(2) Are interrupt handlers wired-down ?

A:

All interrupt handlers are wired-down. This is because an interrupt may occur while handling a missing-page fault. Also, the relative priority of the interrupted process and the interrupt are unknown; it will be bad practice to Block a high priority process while waiting for a page to service a low-priority interrupt.

3. Q (E Wada, Univ. of Tokyo):

(1) What does the ratio in the interrupt frequency table mean ?

A:

The statement that "interrupts on coming from the 7094 disk 50% of the time" means that approximately half the time that the system is in operation, the channel to the disk is in operation. The other half of the time, the channel is idle.

Q:

(2) Is there any ready list for GIOC in the process table ?

A:

The GIOC hardware includes facilities to multiplex its 8 hardware channels among the several thousand apparent channels, so that the apparent channels always appear "ready to go" to the central processor. Thus no ready list is needed. (If you like, the GIOC "ready list" is stored in the GIOC hardware.)

Q:

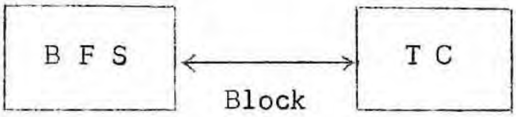
(3) Can the process table grow infinitely ?

A:

The process table is divided into two parts, the Active Process Table, kept in core memory, and the Known Process Table, paged onto drum/disk. The Known Process Table may grow to any size.

Q:

(4)



When missing segment fault takes place, is the T.C. called ?

A:

The Traffic Controller is not called explicitly on missing-segment faults (unless to temporarily wait for a table to be released by another process) but since missing-page faults occur, the Traffic Controller will be called.

6th Lecture (Aug. 22)

1. Q (I. Kimura, Univ. of Tokyo):

Why is the basic file system not recursive ?

A:

A recursive procedure requires a push-down stack. In the case of the missing-page procedure, the push-down stack must be <u>wired-down</u>. If recursive missing page faults were permitted, one would have to predict the maximum depth of recursion in the worst possible case, and then allocate <u>permanently</u> enough space for the push-down stack. For many cases their allocated area would go unused. Therefore, the missing-page routines are not recursive.

2. Q (T. Uraki, Hitachi):

(1) Does the reasonable price of the typewriter, you mentioned, include the cost of the interface to the telephone-line ?

A:

It includes parallel-to-serial (and back) conversion to a D.C. telephone line. It does not include so called "data-set" or other special communication facilities required to provide D.C. service on an A.C. telephone line.

Q:

(2) Could you explain in more detail about the item 5 of the desirable characteristics of the typewriter ?

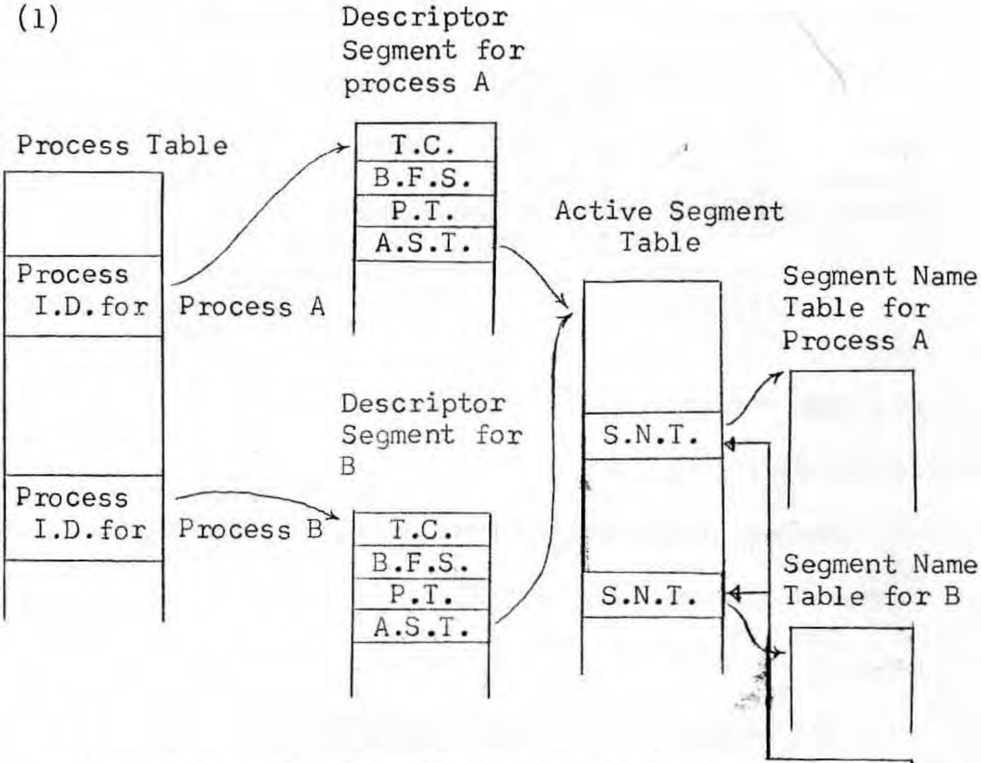
(5. Quit channel always open.)

A:

The user must have available at all times a button with which he can signal the computer (with intention of interrupting his program). This button should operate whether or not his program is "listening" for input from the typewriter, or, for example, when the program is typing out.

3. Q (E. Wada, Univ. of Tokyo):

(1)



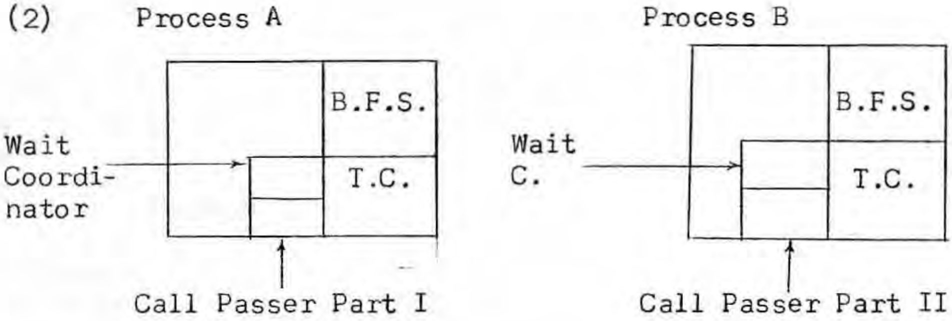
How can we distinguish two S.N.T. in the Active Segment Table for processes A and B ?

A:

The Segment Name Table for process #4, say, is a segment with a unique tree name, or: root> process directory base> #4> segment name table then the symbolic name of a S.N.T. may be fabricated knowing only the process number.

Q:

(2)



Are the both wait coordinators in processes A and B identical ?

And may I understand that A has only Part I of Call Passer and B has only Part II of it ?

A:

The Interprocess Communication Facility is a procedure from the resource management library which is usually shared by all users. However, a user may use a different wait coordinator for one or more of his processer if he is administratively entitled to do so. Process A and B are whichever portions of the call-passer they happen to need; in the example given in the lecture, Process A only used Part I; Process B only used part II.

7th Lecture (Aug. 23)

1. Q (E. Wada, Univ. of Tokyo):

(1) What do you mean by user family ?

A:

A family of processes created to service a single user.

Q:

(2) Can we type command line "brief pℓ/1", after PL/1 has started translation by typing after "pℓ/1 abcd" command ?

A:

One may follow the following sequence:

a. type "pℓ/1 abcd"

b. press "quit" button

c. type "save" command

d. type "brief pℓ/1" command

e. type "resume" command

Q:

(3) Is there any systematic way to translate source program segment name to object program segment name of PL/1 ?

A:

For example, a source file might be named "abcd·pℓ".

One would type "pℓ/1 abcd"; PL/1 translator would add "pℓ"
to name "abcd".

PL produces files "abcd·text" and "abcd•link".

One then types "abcd" command; Search module then looks
for "abcd·text".

2. Q (A. Sasaki, ETL):

Segments and Files are in one to one correspondence. How is it accomplished ?

What's the relation between the concepts of paging and segmentation ?

A:

One-to-one correspondence is not on appropriate way to view segment-file relation. Rather,

a. a File is a linear collection of information, given a symbolic name, and placed in secondary storage.

b. a Segment is a linear section of the available address space of a user.

The Information Storage System allows only one method of access to a file. One must map the file into a segment number (by asking the file system for a segment # of a file.) He can then address the information directly, by segment addressing. The file system perform the mapping by use of missing-page retrieval machinery.

3. Q (T. Maruyama, Fujitsu):

In "give call" from overseer to shell, the argument indicating the location of a command line is represented in overseer's address space.

Does not this mean that the same address space must be shared between overseer and shell ?

A:

Only the segment containing the command line (or other arguments) need be common between the two processor.

Since shared segments do not need to have the same segment number in all sharing processor, the address space do not need to be identical, even for the shared segments.

The location of the command line is passed from the overseer to the shell symbolically. That is, the tree name of the segment and the relative location within the segment are passed. The shell then can Map the segment into its own address space.

8th Lecture (Aug. 24)

1. Q (I. Kimura, Univ. of Tokyo):

(1) Are line-printers available ?

A:

Line printers are normally attached to the system for those applications requiring bulk output, although the volume of bulk output from a time-shared system is often much lower than from a batch-processed system.
Normally, only output which is honestly needed is requested from a time-shared system, since output requests are queued, and actual output may be delayed for a while, (a batch-processing system, it is customary to request all output which might conceivably be useful.)

Q:

(2) Tree names may be infinitely long. How is this problem fixed ?

A:

Some rather long limit on tree name size is imposed by storage limitations in the directory branch. The limit is long enough that it is unlikely that any user will find any difficulty. The cost in search time of exploring deep tree extensions will probably effectively limit useful tree depth much more shortly than will name-size restrictions.

2. Q (T. Yoshimura, Keio Univ.):

(1) Question on System Skeleton: What is the difference between Personnel Registory and User Profiles ?

A:

The Personnel Registory contains information which must be kept secure; e.g., passwords and personal data.

The User Profiles, on the other hand, are not secret; therefore they are placed in different directories with different access control lists.

Also, a single person may wish to have a different user profile when he works on a different project.

Hence there may be several user profiles for one person.

On the other hand, there is only one entry in Personnel Registory for one person.

Q:

(2) Question on Link and Search: Are fault tags restored when a segment is paged-out ?

A:

No. The segment descriptor words are set to "missing-segment" when a segment will be trapped. However, it is never necessary to "Map" the segment into the address space again, so fault tags are not restored.

Briefly, one may describe the actions occurring on the various faults as follows:

Missing-Page: Retrieve the missing page from secondary.

Missing-Segment: Locate the segment on secondary, and create a page table.

Fault Tag 2: Map a <u>named</u> segment into the address space of this process.

Miscellaneous

1. Q (K. Tamura, ETL):

Is it possible to use the Multic System for a real time experiment ?

A:

The Multics traffic controller can offer a service such as the following: User will receive 1 ms of computation within 10 ms of presenting a signal to the system. He may present such a signal as often as once every 500 ms.

Such a user places a severe load on the system, and must be scheduled in advance. It may not ever be possible to have 2 (or 3) such users on the system simultaneously. With this facility, a class of real-time experiments may be controlled by programs running in a Multics system.

2. Q (T. Yoshimura, Keio Univ.):

How do you simulate real-time situations in debugging system ?

A:

We have no special techniques for simulating read-time events at debugging time. The 645 interpreter, which runs on the 635, simulates only 1-processor, 1-memory, no-I/O system. Thus no "real-time events" can be simulated in their environment.

Our only defense against bugs of this kind is to carefully organize the system to minimize timing-dependent errors.

3. Q (E. Wada, Univ. of Tokyo):

Please show us some interesting applications of Time-Sharing.

A:

A recent issue of Scientific American carried an article "Molecular Model Building by Computer describing use of a time-shared display at Project MAC.

A project named "TIP" --- Technical Information Project" is exploring the possible uses of a shared computer to library information retrieval.

Work on a large computer system by several people can be greatly speeded if they can simultaneously share a single system and exchange idea, subroutines, and information easily.

One can use the computer to store and edit (and later type out) documents. This application can speed up composition of a paper, reduce proofreading need, and is generally of interest to the printing industry.

There are many other applications, but these four are representative.

4. Q (I. Kimura, Univ. of Tokyo):

Can a user say: "Please do my job when the computer is free", and be charged less ?

A:

The user may place his job in one of several "absentee" queues, with reduced rates appropriate to the priority which they enjoy. Only 1 or 2 jobs from such a queue we allowed to proceed at once, and their priority is controlled to make sure they do not compete with interactive users.

5. Q (M. Tsujigado, Fujitsu):

(These questions were not asked in lecture)

(1) Is there any approach to the Multi-Access Computer other than that of Project-MAC and IBM/360-67 ? Especially

(i) another project underway except in small computers such as GE 235, PDP-1, PDP-5, SDS-930, 940 and Σ 7.

(ii) another approach undertaken now

A:

I do not know of any other large scale projects than the ones you mention. A number of real-time control system with family resemblence to time-sharing systems are being designed for the Apollo (Man-in-Space) project.

Q:

(2) The rate of central site to remote site. How many percentages does the central site from of all the cost of Multics system ?

A:

Extrapolating from present CTSS experience, one might expect a need for a total 1000 consoles if the system can accomodate 200 simultaneous users. Assuming consoles at \$1500 each, and the central computer at \$5. x 10^6, the consoles represent about 25 percent of the system cost. One should also include the value of the communications lines and, if used, the switching network. Thus, depending on circumstances and improvements in cost/performance, one might expect the remote equipment to represent anywhere from 20-50 percent of the cost of the system.

Q:

(3) Is the research of Project MAC public ? If public, when may we read the internal program of Multics written in PL/I language ?

A:

In general, yes. Multics is a research project, and complete publication of any useful results of research is intended. It is likely that when completed and documented, that substantial parts of the documentation will in fact consist of PL/I programs, where such documentation is appropriate. Similarly, an edited version of the Multics System Programmers' Manual, when complete, will be a part of the final documentation. (It should be realized, of course, that Multics is an evolving system, and that any listing of the system programs is merely a "snapshot" of the state of evolution at a particular time. Our efforts to publish descriptions of CTSS have usually resulted in publications describing an obsolete system by the time the documents are available.)

Q:

(4) Multics is highly centralized system, while we can compare it with another concept named roughly connected computer complex. If you fave any opinion for such a concept, please let us know the advantages and disadvantages.

A:

Although "loosely connected" computer systems have been proposed, I do not recall examples of such systems having been placed in successful operation, and accomplishing objectives similar to that of a centralized system. The basic difficulty is that the information storage of the system is by nature centralized, or else there are bottlenecks in communication (data rate) between the "loosely connected" parts of the computer complex.

5. Q (H. Kawai, ETL):

(These questions were not asked in lecture)

(1) How high priority does the call-passer have in the Traffic Control ?

A:

Inter-process communication originating in the call-passer is not handled in any way differently than other inter-process signals. The scheduling algorithm does not base priority on the particular programs being executed, but rather on the resources consumed by the process executing the programs.

Q:

(2) What relation is there between I/O status class and interruption priority level ?

A:

There could be a one-to-one correspondence between the I/O status interrupts and interrupt priorities, or the I/O interrupts may be grouped into classes. Either technique is acceptable to the Traffic Controller; an adjustable table of processor mask patterns for each hardware interrupt determines the precise grouping into classes.

Q:

(3) Can AST be stored in secondary memory ?

A:

The Active Segment Table cannot be stored as a file in the directory structure on secondary memory, since it is the primary source of information as to where files on secondary memory are located. It could be placed on secondary memory outside the file structure, in known physical location, but this arrangement would defeat the purpose of having in core memory the information necessary to handle any missing-page fault. It would also require special-purpose machinery to be set up to keep track of where the table is stored, since it cannot be in the directory structure.

Q:

(4) How do you think to provide two systems for error tolerance ? (Two systems for a problem)

A:

We are not exploring the areas of running simultaneous computation with periodic cross-checking of results. This technique is used by persons requiring high reliability of individual results. Our reliability efforts are directed instead to high reliability of overall system "up time", thus the emphasis for example on duplexed components which share the load under normal conditions, rather than repeated calculations.

Q:

(5) Do you recommend to standardize the set of terminal commands?

A:

At this point in time standardizing terminal commands would be a futile task, since there is as yet no agreement as to what constitutes a standard set of <u>facilities</u> of a time-sharing system. The whole pattern of usage of a system can shift if some facility is not provided. (E.g., some systems do not permit information storage. Others do not permit on-line development of programs -- only usage of provided programs.) Until it is generally agreed that variations among some set of systems are insignificant differences of taste, standardization cannot be accomplished in any real sense.

Q:

(6) How long is the reasonable response time ?

A:

From the point of view of a human being performing interactive work at a console, adequate response time depends on his expectation. However, for trivial tasks, our experience with CTSS indicates that the human being is annoyed with response time longer than 15-20 seconds, he is satisfied with response within 1-15 seconds, and is unable to usefully benefit from response faster than 1 second. These are response times to typical messages, e.g., system commands or interactive program requests, and it is assumed that the user can type ahead of his program.

Q:

(7) How do you detect system saturation in Multics ?

A:

The primary tool in CTSS is average response time, as indicated by average queue lengths. Presumably a similar technique will work for Multics.

Q:

(8) Is it possible for the users of Multics to consult with

a program instructor in on-line mode ?

A:

This technique is expected to be easy to implement within the structure of the Multics I/O system.

Q:

(9) How do you detect and compensate temporary loss of the modules ?

A:

All duplexed hardware modules are identical, and programs do not depend on absolute device addresses, for example. Therefore, reconfiguring to compensate for a failed device merely lowers system capacity. Detection of failures is an area under study, but we do not presently have worked-out techniques.

Q:

(10) What is the relationship between the entry of directory and file-map ?

A:

A directory entry contains physical location information for the file. This physical location information can only be interpreted by the appropriate Device Interface Module. In particular, the Drum Interface Module interprets the physical location information to be a track address of file map, which contains a list of the track addresses of successive records of the actual file. Another DIM could equally well interpret the physical location information to be the track address of the first record of the file. The first word of that record would contain the track address of the next record, etc..