Published: 05/15/68

Identification

Introduction to Absentee Computations in Multics
H. M. Deitel

Purpose

This section summarizes the features of the Multics mechanism for handling absentee computations, and describes those features which are available to the user for controlling his absentee computations.

Discussion

A Multics user may run his computation in either of two modes, namely interactive or absentee.

An interactive computation (IC) is controlled by a user who enters commands at a remote terminal, receives responses from the system at that terminal, and enters additional commands based upon previous responses. The interactive mode affords the user precise control over his computation and allows the user to make major changes of strategy at run time. The interactive mode is particularly useful for program debugging and for implementing programs which "talk" with non-programmer users (administrators, scientists, flight reservation personnel, etc.)

An absentee computation (AC) does not require interaction with the user. The user submits an absentee computation as a file of commands (basically identical to the commands the user would enter if running the same computation interactively). The system maintains a queue of all such files and initiates new absentee computations at its own convenience. The absentee mode frees the user from having to be present to control his computation and is particularly useful for running checked out programs.

A batch computation is basically the same as an absentee computation except for the manner in which the file of commands is submitted to the system. An absentee computation command file is generally submitted via a remote terminal, whereas a batch computation command file is generally submitted as a card deck at the central computer installation.

Summary of Features

1. An AC may be initiated by

   a. an IC

   b. another AC

   c. a batch user.

2. An AC for a particular user may be terminated by

   a. any of that user's IC's

   b. any of that user's AC's.

3. A computation of a particular user may obtain status information about any of that user's computations (both AC's and IC's). Detailed information about each of a user's AC's is always available to that user, regardless of whether his AC's are waiting to be run or are currently running.

4. The system enqueues requests to initiate new AC's so that these AC's may be initiated in the future at a time which the system feels is opportune.

5. The system may temporarily suspend service to a number of AC's so that an increased load of IC's may be more effectively serviced (and vice versa).

6. The system may automatically resume service to suspended AC's when the IC load decreases. (Note: Suspending an IC corresponds to automatically logging it out. The only way for a suspended IC to continue operation is for the interactive user to log in again and RESUME the IC. The system cannot automatically resume suspended IC's.)

7. An interactive user may interrupt his IC at any point and convert it to an AC.

8. An interactive user may interrupt any one of his AC's at any point and convert it to his current IC.

9. The system administrative personnel may specify an apportionment of system resources between AC's and IC's in order to emphasize a particular mode during certain periods of system operation. In effect, this apportionment partitions the system into two distinct systems, one for running IC's and one for running AC's. The AC-IC apportionment may range anywhere from 0%-100% to 100%-0%.

10. Jobs may be submitted to Multics in a batch-processing format as a deck of cards. Such jobs, once entered into the system, are handled by the absentee mechanism in the same manner as AC's requested by IC's or other AC's.

11. The system makes shutdown transparent to AC's by suspending any AC's running at shutdown and automatically resuming the suspended AC's at startup time.

12. The apportionment of resources between AC's and IC's is normally done for various long periods of system operation called "shifts". However, the demands upon the system made by AC's and IC's over the short term may vary significantly and frequently. The system caters to short term variations in demand by making modifications to the resource apportionment over short periods called "integration periods". The Load Control mechanism (see section BQ.5.00) compares the current demands upon the system with the current shift's resource apportionment, and makes necessary adjustments in AC and IC loads to insure that good quality service is provided to all running computations regardless of any short term surges in demand. For example, if the AC-IC apportionment is 25%-75% and the AC demands decrease to only 20%, Load Control allows enough new IC's to be initiated to bring the IC usage up to 80%. When the short term variations are of greater magnitude, Load Control may not match a decrease in one mode of usage with an increase in the other. Instead, it may match a 20% decrease with only a 10% increase. This provides a damping effect which helps to prevent the current usage of system resources from varying too significantly from the desired shift resource apportionment.

13. The system administrative personnel may terminate any AC (or IC) which appears to be a "troublemaker".

14. A user may have many running AC's (and many IC's) at one time, but the number can be administratively limited.

15. AC input is normally taken from a user-specified file called an Absentee Computation Descriptor File. AC output is normally placed in another user-specified file called an Absentee Computation Output File. The user may alternatively specify dedicated resources (in place of files) for AC input/output.

16. A user requiring dedicated resources should place an advance reservation for them. When the reservation becomes due, the system automatically initiates the AC.

17. A user may specify (for his own protection) any running time limit for each of his AC's. If the time limit is exceeded, the system automatically SAVE's the AC so that parti¬l results are not lost. If the user does not specify a time limit for an AC, then a default value is assumed by the system, again for the user's own protection.

18. In a system such as Multics in which many computations are simultaneously competing for service, it is desirable to provide some sort of priority mechanism to allow a user to express the importance of his computation. The Multics user is provided with a choice of <u>priority streams</u> in which he may request his AC be run. High priority streams provide service at higher cost, while low priority streams may provide slower service but at reduced cost. If the user does not specify in which priority stream he wants his computation to run, the system automatically inserts the computation into the <u>standard stream</u>. Since absentee computations do not necessarily have to be initiated immediately, a series of waiting queues is provided, one for each priority stream. When Load Control decides that more absentee computations may be placed into a particular stream, it informs the mechanism which handles the absentee queues to initiate an appropriate number of absentee computations from the waiting queue for that stream.