Identification

The Absentee User-Process-Group

K.J. Martin, B.A. Tague (see note)

DRAFT

date 7/21/66

Purpose

Some of the operations which a user might wish to perform on Multics do not require his presence at a remote terminal. When a user wishes to operate as absentee, i.e., not present at a remote terminal, he declares this fact to the system, using the "absentee" command (BX.3.04). This section discusses the concepts involved in managing an absentee user.

Discussion

No user may operate on Multics unless he first logs in as described in ~~the~~ previous BQ.1.05 section. Immediately after logging in the user has three processes - an Overseer, a Device Manager and a Working Process - which are known collectively as a user-process-group. Every interactive user, i.e., present at a remote terminal, has at least these three processes plus any others which he has created. To say that an interactive user wishes to operate as absentee is to say that another process-group is to be created for the user. The absentee process-group uses files rather than a remote terminal for input and output.

Before a user can initiate an absentee process-group there must exist a file in the file system which contains information suitable for use as a command input. The user is responsible for creating the file. When the user initiates an absentee process-group he specifies the file to be used as input. He may, if he likes, specify a file to be used for output.

The absentee command is called and executed in the Working process like any

other command. The duties of the command are light. It verifies that the file specified as input does exist. If a file is specified for output, it checks on that file, creating it if it does not exist. If no file is specified for output, the command creates and names one, placing it in the user's working file directory and telling the user the name given to it. The user must have at least read access to the input file and write access the output file. Any errors, such as improper access to a file or a non-existent input file cause the absentee process-group not to be created. The command is discontinued at the point that the error occurs.

The absentee command now places the names of the input and output files and the user profile of this user in a data base of requested absentee jobs. The absentee command calls the wakeuser procedure to wakeup the Absentee Monitor process.

The Absentee Monitor process has two modes of operation. In one mode it acts much like the Answering Service, creating Overseer processes and calling them, then destroying them upon their return. The Absentee Monitor gives each process which it creates a process-group-wide data base as specified by the user profile contained in an entry of the data base of requested absentee jobs. The overseer procedure is called in that process, passing as arguments the input and output files specified in the same entry of the data base of requested absentee jobs.

The identification of the Overseer process is placed in a monitor data base which the Absentee Monitor uses in its other mode of operation. This other mode is discussed later.

The overseer procedure called by the Monitor is different in some respects from the overseer described in BO.1.04. In particular, the login mode is not needed since the user is already fully identified. The call to the listener procedure is made at an entry point which avoids processing the second part of logging in. Figure 1 of BO.1.04 shows these differences between the actions of the overseer procedure as called by the Answering Service and as called by the Absentee Monitor.

The second mode in which the Absentee Monitor operates is as director of scheduling for the absentee process-groups under its control. The scheduling of absentee process-groups is separate from the scheduling of interactive process-groups. Individual absentee process-groups can be removed from the direction of the Monitor when necessary. For instance, a process-group which runs as part of a reservation is not controlled by the absentee monitor because it must operate under conditions which the absentee monitor cannot determine or control. (Reservations are discussed in MSPM Section BT.1.00).

It must be possible for a user to regain control of his absentee process-group. He must be able to log in again (creating another process-group) and issue a quit to the absentee Overseer from the other process-group. An entry in the intercom data base (described in BD.8.03) of the Overseer makes it possible for specified users to note the quit event. In particular, all other process-groups belonging to the same user are allowed to do so. The user need only login in and execute the quit_absentee command which calls the wakeuser procedure (described in BD.8.03) giving the identification of the absentee process-group. The quit handler in the absentee Overseer, in addition to stopping the Working Process notes that the process-group is absentee so the Overseer blocks before asking to read the next input line. The quit-absentee command is one of a class useable by restricted process-groups

which could not log in normally because of system load. Therefore, a user can always quit an absentee process-group.

After having quit an absentee process-group in most cases the user wishes to assume positive control of the process-group from his remote console. The join_absentee command causes the console which the user is using to be attached to the absentee process-group, making it interactive. The interactive process-group which executed the join_absentee command is effectively logged out and replaced by the formerly absentee process-group. The details of join_absentee are explained in BX.3. . After the join_absentee command has been executed, the newly interactive process-group is at command level waiting for input from the console. At this point the user may examine the files which were the input and output files to determine if any mishaps occurred. Corrections may be made as in any interactive process-group.

The logout_absentee command is also available for use after a user has quit an absentee process-group. This command sends a logout wakeup to the absentee process-group, causing it to log out. The input and output files remain intact and can be examined by the user at any time.

A command run as part of the absentee process-group may encounter difficulties which make it necessary to block the absentee process-group and wait until some repair work (as described above) has been done. The listen procedure of the Overseer is prepared to note an error status upon a return from the shell procedure in the Working process. The absentee instance is then blocked in the Overseer. This blocked condition is much like having been quit and requires the same sort of repair work.

When an absentee process-group is completed (has logged out) or reaches a point where it must be blocked until help comes, the user to whom it belongs should be notified. If he is logged in at an interactive process-group, the inter-process-group communication facility )BD.8.03) is used to notify him. If he is not logged in, a notification is sent to him via the system mail facilities.

Note:

The author of this section, K.J. Martin, is being transferred to another subproject. B.A. Tague joined the Central System Control subproject shortly before publication of this section. Although he was not involved in the authorship of this section, he is aware of the issues and will be working on the task. Therefore, comments and criticism of the ideas presented may be directed to him.