

TO: MSPM Distribution  
FROM: K. J. Martin  
SUBJ: BX.2.02  
DATE: 07/07/67

The listener has changed in two ways:

- 1) It no longer participates in logging out a user;
- 2) It takes special actions following a quit.

TO: MSPM Distribution  
FROM: K. J. Martin  
SUBJECT: BX.2.02  
DATE: 07/19/68

The listener has changed in two ways:

- 1) It is no longer active following a quit.
- 2) The overseer does not place "arguments" in a special data base; rather, it does all necessary attachments for the process.

## Identification

Listener procedure  
K. J. Martin

## Purpose

The listener procedure's function is to read complete command sequences (which may require one or more calls to read from the input stream) and call the Shell (BX.2.00) passing the command sequence as an argument.

## Discussion

The listener procedure operates in the working process of a user-process-group. (See BQ.3.00 for an overview of user-process-groups.) The presence of the listener in a user's working process defines that user's subsystem by virtue of the fact that in normal operation all command input is passed to the Shell for interpretation.

The listener is essentially a read loop on the user's input stream. It reads successive lines of input until a complete command sequence has been received, then calls the Shell. The listener must be prepared to recognize when it has read a complete command sequence and when it has not.

Three Shell punctuation characters are of interest to the listener: the escape character (normally %), and the literal string delimiters (normally ` to open a string and ' to close a string). The Shell punctuation characters are contained in the segment, shell\_char (BX.2.01). Both the listener and the Shell reference shell\_char to determine the current values of punctuation characters.

The last character of every input line is the new line character. The listener recognizes two conditions which indicate that it has not received a complete command sequence and must read more input:

- 1) The new line character is immediately preceded by the Shell escape character;
- 2) The new line character is imbedded in an incomplete literal string.

### Implementation of Listener

The listener is called in the working process and proceeds as follows:

- 1) Initialize for the cancel command following an unclaimed signal (see BY.11.05 and BX.3.11) by calling `unclaimed_signal$init` passing a label variable within the listener as an argument.
- 2) Check the `project_restrictions` segment of the process profile (BQ.4.03) to determine the setting of the `restricted_shell` indicator. If it is on, the Shell should be called indicating that pathnames are not legal as commands.
- 3) Print a ready message (R `xx.x yy.y zz`) where `xx.x` is the real time from the start of the current command (i.e., from the previous wait message - see step 8), `yy.y` is the CPU time used since the last ready message, and `zz` is the number of page waits (times the process was required to wait for a page to be read from the drum) since the last ready message.
- 4) Read from the user's input stream.
- 5) Scan the input read in step 4 (or by the quit handler) to check for a complete command sequence.
- 6) If the input is an incomplete command sequence, update arguments for the read call (so that the next line is read in immediately following the last) and go to step 4. Otherwise,
- 7) Call the Shell passing the complete command sequence.
- 8) Print a wait message (W `xxxx.x`) on the console where `xxxx.x` is the current time.
- 9) On return from the Shell, check the user's housekeep option. If the option is off go to step 3. Otherwise (the option is on) return. The overseer procedure interprets a return from the listener as a request for housekeeping. The overseer destroys the current working process, creates another, and calls the listener in it.

The listener is a loop reading input from the stream `user_input` and calling the Shell. The loop can be broken by any of:

- 1) housekeeping - a return to the overseer;
- 2) a quit signalled by the user depressing his break key;
- 3) execution of the logout command;
- 4) an automatic logout imposed on the process group by the system.