IEEE P1003.2a Draft 8 – December 1991

5.5 `crontab` — Schedule periodic background work

5.5.1 Synopsis

`crontab` [*file*]

`crontab [ −e | −l | −r ]`

5.5.2 Description

The `crontab` utility shall create, replace, or edit a user's *crontab* entry; a crontab entry is a list of commands and the times at which they are to be executed. The new crontab entry can be input by specifying *file* or input from standard input if no *file* operand is specified, or by using an editor, if `−e` is specified.

Upon execution of a command from a crontab entry, the implementation shall supply a default environment, defining at least the following environment variables:

HOME A pathname of the user's home directory.

LOGNAME The user's login name.

PATH A string representing a search path guaranteed to find all of the standard utilities.

SHELL A pathname of the command interpreter. When `crontab` is invoked as specified by this standard, the value shall be a pathname for `sh` (see 4.56).

The values of these variables when `crontab` is invoked as specified by this standard shall not affect the default values provided when the scheduled command is run.

If standard output and standard error are not redirected by commands executed from the crontab entry, any generated output or errors shall be mailed, via an implementation-defined method, to the user.

5.5.3 Options

The `crontab` utility shall conform to the utility argument syntax guidelines described in 2.10.2.

The following options shall be supported by the implementation:

`−e` Edit a copy of the invoking user's crontab entry, or create an empty entry to edit if the crontab entry does not exist. When editing is complete, the entry shall be installed as the user's crontab entry.

`-l` (The letter ell.) List the invoking user's crontab entry.

`-r` Remove the invoking user's crontab entry.

5.5.4 Operands

The following operand shall be supported by the implementation:

file The pathname of a file that contains specifications, in the format defined in 5.5.5.2, for crontab entries.

5.5.5 External Influences

5.5.5.1 Standard Input

See Input Files.

5.5.5.2 Input Files

In the POSIX Locale, a crontab entry shall be a text file consisting of lines of six fields each. The fields shall be separated by `<blank>`s. The first five fields shall be integer patterns that specify the following:

(1) Minute (0−59)

(2) Hour (0−23)

(3) Day of the month (1−31)

(4) Month of the year (1−12)

(5) Day of the week (0−6 with 0=Sunday).

Each of these patterns can be either an asterisk (meaning all valid values), an element, or a list of elements separated by commas. An element shall be either a number or two numbers separated by a hyphen (meaning an inclusive range). The specification of days can be made by two fields (day of the month and day of the week). If month, day of month, and day of week are all asterisks, every day shall be matched. If either the month or day of month is specified as an element or list, but the day of week is an asterisk, the month and day of month fields shall specify the days that match. If both month and day of month are specified as asterisk, but day of week is an element or list, then only the specified days of the week shall match. Finally, if either the month or day of month is specified as an element or list, and the day of week is also specified as an element or list, then any day matching either the month and day of month, or the day of week, shall be matched.

The sixth field of a line in a crontab entry is a string that shall be executed by `sh` (see 4.56) at the specified times. A percent-sign character in this field shall be translated to a `<newline>` character. Any character preceded by a backslash (including the `%`) shall cause that character to be treated literally. Only the first line (up to a `%` or end of line) of the command field shall be executed by the command interpreter. The other lines shall be made available to the command as standard input.

Blank lines and those whose first non-`<blank>` character is # shall be ignored.

5.5.5.3 Environment Variables

The following environment variables shall affect the execution of `crontab`:

EDITOR This variable shall determine the editor to be invoked when the `−e` option is specified. The default editor shall be `vi`.

LANG This variable shall determine the locale to use for the locale categories when both **LC_ALL** and the corresponding environment variable (beginning with **LC_**) do not specify a locale. See 2.6.

LC_ALL This variable shall determine the locale to be used to override any values for locale categories specified by the settings of **LANG** or any environment variables beginning with **LC_**.

LC_CTYPE This variable shall determine the interpretation of sequences of bytes of text data as characters (e.g., single- versus multibyte characters in arguments and input files).

LC_MESSAGES This variable shall determine the language in which messages should be written.

5.5.5.4 Asynchronous Events

Default.

5.5.6 External Effects

5.5.6.1 Standard Output

If the `−l` option is specified, the crontab entry shall be written to the standard output.

5.5.6.2 Standard Error

Used only for diagnostic messages.

5.5.6.3 Output Files

None.

5.5.7 Extended Description

None.

5.5.8 Exit Status

The `crontab` utility shall exit with one of the following values:

- 0 Successful completion.
- >0 An error occurred.

5.5.9 Consequences of Errors

The user's crontab entry is not submitted, removed, edited, or listed.

5.5.10 Rationale. *(This subclause is not a part of P1003.2a)*

Usage, Examples

As an example of specifying the two types of days,

```
0 0 1,15 * 1
```

would run a command on the first and fifteenth of each month, as well as on every Monday. To specify days by only one field, the other field should be set to * (for example,

```
0 0 * * 1
```

would run a command only on Mondays).

The default settings of the **HOME**, **LOGNAME**, **PATH**, and **SHELL** variables that are given to the scheduled job are not affected by the settings of those variables when `crontab` is run; as stated, they are defaults. The text about ‘‘invoked as specified by this standard’’ means that the implementation may provide extensions that allow these variables to be affected at runtime, but that the user has to take explicit action in order to access the extension, such as give a new option flag or modify the format of the crontab entry.

History of Decisions Made

All references to a `cron` daemon and to `cron` *files* have been removed. Although historical implementations have used this arrangement, there is no reason to limit future implementations.

This description of `crontab` is designed to support only users with normal privileges. The format of the input is based on the System V `crontab`; however, there is no requirement here that the actual system database used by the `cron` daemon (or similar mechanism) use this format internally. For example, systems derived from BSD will likely have an additional field appended that indicates the user identity to be used when the job is submitted.

The `−e` option was adopted from the *SVID* as a user convenience, although it does not exist in all historical implementations.

5.6 `csplit` — Split files based on context

5.6.1 Synopsis

`csplit [−ks][−f` *prefix* `][−n` *number* `]` *file arg1 . . . argn*

5.6.2 Description

The `csplit` utility shall read the file named by the *file* operand, write all or part of that file into other files as directed by the *arg* operands, and write the sizes of the files.

5.6.3 Options

The `csplit` utility shall conform to the utility argument syntax guidelines described in 2.10.2.

The following options shall be supported by the implementation:

`−f` *prefix* Name the created files *prefix* `00`, *prefix* `01`, . . . , *prefixn* . The default shall be `xx00` . . . `xx`*n*. If the *prefix* argument would create a filename exceeding {NAME_MAX} bytes, an error shall result, `csplit` shall exit with a diagnostic message and no files shall be created.

`−k` Leave previously created files intact. By default, `csplit` shall remove created files if an error occurs.

`−n` *number* Use *number* decimal digits to form filenames for the file pieces. The default shall be 2.

`−s` Suppress the output of file size messages.