NAME

at, batch, atq, atrm − queue, examine, or delete jobs for later execution

SYNOPSIS

at [**−V**] [**−q** *queue*] [**−f** *file*] [**−mMlv**] *timespec ...*
at [**−V**] [**−q** *queue*] [**−f** *file*] [**−mMkv**] [**−t** *time*]
at −c *job* [...]
atq [**−V**] [**−q** *queue*]
at [**−rd**] *job* [...]
atrm [**−V**] *job* [...]
batch
at −b

DESCRIPTION

at and **batch** read commands from standard input or a specified file which are to be executed at a later time, using **/bin/sh**.

at   executes commands at a specified time.

atq   lists the user's pending jobs, unless the user is the superuser; in that case, everybody's jobs are listed. The format of the output lines (one for each job) is: Job number, date, hour, queue, and username.

atrm   deletes jobs, identified by their job number.

batch   executes commands when system load levels permit; in other words, when the load average drops below 1.5, or the value specified in the invocation of **atd**.

At allows fairly complex time specifications, extending the POSIX.2 standard. It accepts times of the form **HH:MM** to run a job at a specific time of day. (If that time is already past, the next day is assumed.) You may also specify **midnight, noon,** or **teatime** (4pm) and you can have a time-of-day suffixed with **AM** or **PM** for running in the morning or the evening. You can also say what day the job will be run, by giving a date in the form **month-name day** with an optional **year,** or giving a date of the form *MMDD*[*CC*]*YY*, *MM*/*DD*/[*CC*]*YY*, *DD.MM.*[*CC*]*YY* or [*CC*]*YY-MM-DD*. The specification of a date *must* follow the specification of the time of day. You can also give times like **now +** *count time-units,* where the time-units can be **minutes, hours, days,** or **weeks** and you can tell **at** to run the job today by suffixing the time with **today** and to run the job tomorrow by suffixing the time with **tomorrow.**

For example, to run a job at 4pm three days from now, you would do **at 4pm + 3 days,** to run a job at 10:00am on July 31, you would do **at 10am Jul 31** and to run a job at 1am tomorrow, you would do **at 1am tomorrow.**

If you specify a job to absolutely run at a specific time and date in the past, the job will run as soon as possible. For example, if it is 8pm and you do a **at 6pm today,** it will run more likely at 8:05pm.

The definition of the time specification can be found in */usr/share/doc/at/timespec*.

For both **at** and **batch**, commands are read from standard input or the file specified with the **−f** option and executed. The working directory, the environment (except for the variables **BASH_VERSINFO**, **DIS-PLAY**, **EUID**, **GROUPS**, **SHELLOPTS**, **TERM**, **UID**, and **_**) and the umask are retained from the time of invocation.

As **at** is currently implemented as a setuid program, other environment variables (e.g., **LD_LI-BRARY_PATH** or **LD_PRELOAD**) are also not exported. This may change in the future. As a work-around, set these variables explicitly in your job.

An **at −** or **batch −** command invoked from a **su**(1) shell will retain the current userid. The user will be mailed standard error and standard output from his commands, if any. Mail will be sent using the command **/usr/sbin/sendmail**. If **at** is executed from a **su**(1) shell, the owner of the login shell will receive the mail.

The superuser may use these commands in any case. For other users, permission to use at is determined by the files */etc/at.allow* and */etc/at.deny*. See **at.allow**(5) for details.

OPTIONS

−V   prints the version number to standard error and exit successfully.

−q *queue*

uses the specified queue. A queue designation consists of a single letter; valid queue designations range from **a** to **z** and **A** to **Z**. The **a** queue is the default for **at** and the **b** queue for **batch**. Queues with higher letters run with increased niceness. The special queue "=" is reserved for jobs which are currently running.

If a job is submitted to a queue designated with an uppercase letter, the job is treated as if it were submitted to batch at the time of the job. Once the time is reached, the batch processing rules with respect to load average apply. If **atq** is given a specific queue, it will only show jobs pending in that queue.

−m   Send mail to the user when the job has completed even if there was no output.

−M   Never send mail to the user.

−f *file*   Reads the job from *file* rather than standard input.

−t *time*   run the job at *time*, given in the format [[CC]YY]MMDDhhmm[.ss]

−l   Is an alias for **atq.**

−r   Is an alias for **atrm.**

−d   Is an alias for **atrm.**

−b   is an alias for **batch**.

−v   Shows the time the job will be executed before reading the job.

Times displayed will be in the format "Thu Feb 20 14:50:00 1997".

−c   cats the jobs listed on the command line to standard output.

FILES

/var/spool/cron/atjobs
/var/spool/cron/atspool
/proc/loadavg
/var/run/utmp
/etc/at.allow
/etc/at.deny

SEE ALSO

at.allow(5), **at.deny**(5), **atd**(8), **cron**(1), **nice**(1), **sh**(1), **umask**(2).

BUGS

The correct operation of **batch** for Linux depends on the presence of a *proc*- type directory mounted on */proc*.

If the file */var/run/utmp* is not available or corrupted, or if the user is not logged on at the time **at** is invoked, the mail is sent to the userid found in the environment variable **LOGNAME**. If that is undefined or empty, the current userid is assumed.

At and **batch** as presently implemented are not suitable when users are competing for resources. If this is the case for your site, you might want to consider another batch system, such as **nqs**.

AUTHOR

At was mostly written by Thomas Koenig.