NAME

baseline – rcs utility

USAGE

baseline *[options] [file-specifications]*

SYNOPSIS

Baseline permits you to set a baseline version in one or more RCS files.

DESCRIPTION

Baseline adds a baseline (placeholder) version to one or more RCS archives. The effective check-in time for the file is the file's last check-in date (before **baseline** is invoked)

Assuming that the file was originally checked in using **checkin**, then the check-in, modification and baseline dates all are the same.

A baseline version consists of a group of files which can be referenced by a common version number. The baseline program uses for this purpose 2.0, 3.0, etc. Version 1.0 is never used, since the first development version is, by convention, 1.1.

After baselining, you may use **checkout** to obtain a particular set of files. For example,

```
checkout -r2.0 file
```

obtains the baseline "2.0" version, while

```
checkout -r2 file1 file2 file3
```

obtains the last development version beginning with "2.0", but before "3.0".

OPTIONS

Baseline recognizes several options:

–*integer*
: the new baseline version, specified as a whole number after a hyphen, e.g.,

```
baseline -2 file1 file2 file3
```

If no baseline version is specified, the most recent baseline version (as recorded in the **RCS,v permit**-file) is assumed. This is done to permit additional files to be added to an existing baseline.

–a permits **baseline** to process files in directories beginning with ".".

–l leaves files locked after baselining. Also, if files are locked but not modified, no error is reported.

–L allows **baseline** to follow symbolic links to directories.

–m*msg*
: adds the string "–– *msg*" to the default baseline revision message "**BASELINE** *date*".

–n runs **baseline** in no-op mode, showing where permissions would be purged, and listing files which would be baselined.

–p the "purge" option purges existing access lists in the archives to prevent locks while the baselining process is proceeding.

–R permits **baseline** to run recursively when it encounters a non-RCS directory.

–v makes the baseline process more verbose by causing permit, checkin and checkout to show their actions.

OPERATIONS

Before invoking **baseline**, ensure that all of the files to be baselined have been checked into *rcs* (e.g., with "**checkin** –u"). This is easily done using the directory editor **ded**.

Baseline operates only upon files which have been checked into *rcs* (i.e., a corresponding archive file exists

in the subdirectory "RCS").

Baseline requires that all of the working files have been checked in (i.e., with the "−u" option so that they are retained as working files).

Baseline first invokes **permit** (to ensure that you have permission to place locks in the directory, then to purge the archive access lists, and finally to set the baseline-version for the directory). It then invokes **checkin** and **checkout** (which must be in your path) to first place a lock on the specified files, and then to check-in the file with the new, baseline version.

If any files are found which are locked, **baseline** prints an error message and halts. Similarly, an attempt made to baseline a file with a lower version than has been archived will result in an error.

ENVIRONMENT

Baseline is written in C, and runs under POSIX systems.

FILES

Baseline uses the following files

permit the RCS directory-permissions utility

checkin
the RCS check-in package

checkout
the RCS check-out package

ANTICIPATED CHANGES

Baseline does not currently handle branch deltas.

SEE ALSO

checkin, checkout, ded, permit

AUTHOR:

Thomas E. Dickey <dickey@invisible-island.net>