NAME

quilt – tool to manage series of patches

SYNOPSIS

quilt [-h] command [options]

DESCRIPTION

Quilt is a tool to manage large sets of patches by keeping track of the changes each patch makes. Patches can be applied, un-applied, refreshed, etc. The key philosophical concept is that your primary output is patches.

With quilt, all work occurs within a single directory tree. Commands can be invoked from anywhere within the source tree. They are of the form **quilt cmd** similar to CVS, svn or git commands. They can be abbreviated as long as the specified part of the command is unique. All commands print some help text with **quilt cmd -h.**

Quilt manages a stack of patches. Patches are applied incrementally on top of the base tree plus all preceding patches. They can be pushed on top of the stack (**quilt push**), and popped off the stack (**quilt pop**). Commands are available for querying the contents of the series file (**quilt series**, see below), the contents of the stack (**quilt applied**, **quilt previous**, **quilt top**), and the patches that are not applied at a particular moment (**quilt next**, **quilt unapplied**). By default, most commands apply to the topmost patch on the stack.

Patch files are located in the *patches* sub-directory of the source tree (see EXAMPLE OF WORKING TREE below). The *QUILT_PATCHES* environment variable can be used to override this location. When not found in the current directory, that subdirectory is searched recursively in the parent directories (this is similar to the way *git* searches for its configuration files). The *patches* directory may contain sub-directories. It may also be a symbolic link instead of a directory.

A file called *series* contains a list of patch file names that defines the order in which patches are applied. Unless there are means by which series files can be generated automatically, it is usually provided along with a set of patches. In this file, each patch file name is on a separate line. Patch files are identified by path names that are relative to the *patches* directory; patches may be in sub-directories below this directory. Lines in the series file that start with a hash character (#) are ignored. You can also add a comment after each patch file name, introduced by a space followed by a hash character. When quilt adds, removes, or renames patches, it automatically updates the series file. Users of quilt can modify series files while some patches are applied, as long as the applied patches remain in their original order.

Different series files can be used to assemble patches in different ways, corresponding for example to different development branches.

Before a patch is applied (or ‘‘pushed on the stack’’), copies of all files the patch modifies are saved to the .pc/*patch* directory. The patch is added to the list of currently applied patches (.pc/applied-patches). Later when a patch is regenerated (**quilt refresh**), the backup copies in .pc/*patch* are compared with the current versions of the files in the source tree using GNU diff.

Documentation related to a patch can be put at the beginning of a patch file. Quilt is careful to preserve all text that precedes the actual patch when doing a refresh. (This is limited to patches in unified format; see **diff** documentation).

The series file is looked up in the .pc directory, in the root of the source tree, and in the patches directory. The first series file that is found is used. This may also be a symbolic link, or a file with multiple hard links. Usually, only one series file is used for a set of patches, so the patches sub-directory is a convenient location.

The .pc directory and its sub-directories cannot be relocated, but it can be a symbolic link. While patches are applied to the source tree, this directory is essential for many operations, including taking patches off the stack (**quilt pop**), and refreshing patches (**quilt refresh**). Files in the .pc directory are automatically removed when they are no longer needed, so there is no need to clean up manually.

QUILT COMMANDS REFERENCE

add [-P patch] {file} ...

Add one or more files to the topmost or named patch. Files must be added to the patch before being modified. Files that are modified by patches already applied on top of the specified patch cannot be added.

-P patch

Patch to add files to.

annotate [-P patch] {file}

Print an annotated listing of the specified file showing which patches modify which lines. Only applied patches are included.

-P patch

Stop checking for changes at the specified rather than the topmost patch.

applied [patch]

Print a list of applied patches, or all patches up to and including the specified patch in the file series.

delete [-r] [--backup] [patch|-n]

Remove the specified or topmost patch from the series file. If the patch is applied, quilt will attempt to remove it first. (Only the topmost patch can be removed right now.)

-n Delete the next patch after topmost, rather than the specified or topmost patch.

-r Remove the deleted patch file from the patches directory as well.

--backup

Rename the patch file to patch˜ rather than deleting it. Ignored if not used with ‘-r’.

diff [-p n|-p ab] [-u|-U num|-c|-C num] [--combine patch|-z] [-R] [-P patch] [--snapshot] [--diff=utility] [--no-timestamps] [--no-index] [--sort] [--color[=always|auto|never]] [file ...]

Produces a diff of the specified file(s) in the topmost or specified patch. If no files are specified, all files that are modified are included.

-p n Create a -p n style patch (-p0 or -p1 are supported).

-p ab Create a -p1 style patch, but use a/file and b/file as the original and new filenames instead of the default dir.orig/file and dir/file names.

-u, -U num, -c, -C num

Create a unified diff (-u, -U) with num lines of context. Create a context diff (-c, -C) with num lines of context. The number of context lines defaults to 3.

--no-timestamps

Do not include file timestamps in patch headers.

--no-index

Do not output Index: lines.

-z Write to standard output the changes that have been made relative to the topmost or specified patch.

-R Create a reverse diff.

-P patch

Create a diff for the specified patch. (Defaults to the topmost patch.)

--combine patch

Create a combined diff for all patches between this patch and the patch specified with -P. A patch name of ‘-’ is equivalent to specifying the first applied patch.

--snapshot

Diff against snapshot (see ‘quilt snapshot -h’).

--diff=utility

Use the specified utility for generating the diff. The utility is invoked with the original and new file name as arguments.

--color[=always|auto|never]

Use syntax coloring (auto activates it only if the output is a tty).

--sort Sort files by their name instead of preserving the original order.

edit file ...

Edit the specified file(s) in $EDITOR after adding it (them) to the topmost patch.

files [-v] [-a] [-l] [--combine patch] [patch]

Print the list of files that the topmost or specified patch changes.

-a List all files in all applied patches.

-l Add patch name to output.

-v Verbose, more user friendly output.

--combine patch

Create a listing for all patches between this patch and the topmost or specified patch. A patch name of '-' is equivalent to specifying the first applied patch.

fold [-R] [-q] [-f] [-p strip-level]

Integrate the patch read from standard input into the topmost patch: After making sure that all files modified are part of the topmost patch, the patch is applied with the specified strip level (which defaults to 1).

-R Apply patch in reverse.

-q Quiet operation.

-f Force apply, even if the patch has rejects. Unless in quiet mode, apply the patch interactively: the patch utility may ask questions.

-p strip-level

The number of pathname components to strip from file names when applying patchfile.

fork [new_name]

Fork the topmost patch. Forking a patch means creating a verbatim copy of it under a new name, and use that new name instead of the original one in the current series. This is useful when a patch has to be modified, but the original version of it should be preserved, e.g. because it is used in another series, or for the history. A typical sequence of commands would be: fork, edit, refresh.

If new_name is missing, the name of the forked patch will be the current patch name, followed by '-2'. If the patch name already ends in a dash-and-number, the number is further incremented (e.g., patch.diff, patch-2.diff, patch-3.diff).

graph [--all] [--reduce] [--lines[=num]] [--edge-labels=files] [-T ps] [patch]

Generate a dot(1) directed graph showing the dependencies between applied patches. A patch depends on another patch if both touch the same file or, with the --lines option, if their modifications overlap. Unless otherwise specified, the graph includes all patches that the topmost patch depends on. When a patch name is specified, instead of the topmost patch, create a graph for the specified patch. The graph will include all other patches that this patch depends on, as well as all patches that depend on this patch.

--all Generate a graph including all applied patches and their dependencies. (Unapplied patches are not included.)

--reduce

Eliminate transitive edges from the graph.

--lines[=num]

Compute dependencies by looking at the lines the patches modify. Unless a different num is specified, two lines of context are included.

--edge-labels=files

Label graph edges with the file names that the adjacent patches modify.

-T ps Directly produce a PostScript output file.

grep [-h|options] {pattern}

Grep through the source files, recursively, skipping patches and quilt meta-information. If no filename argument is given, the whole source tree is searched. Please see the grep(1) manual page for options.

-h Print this help. The grep -h option can be passed after a double-dash (--). Search expressions that start with a dash can be passed after a second double-dash (-- --).

header [-a|-r|-e] [--backup] [--dep3] [--strip-diffstat] [--strip-trailing-whitespace] [patch]

Print or change the header of the topmost or specified patch.

-a, -r, -e

Append to (-a) or replace (-r) the existing patch header, or edit (-e) the header in $EDITOR. If none of these options is given, print the patch header.

--strip-diffstat

Strip diffstat output from the header.

--strip-trailing-whitespace

Strip trailing whitespace at the end of lines of the header.

--backup

Create a backup copy of the old version of a patch as patch˜.

--dep3

When editing (-e), insert a template with DEP-3 headers. DEP-3 is http://dep.debian.net/deps/dep3/ Patch Tagging Guidelines.

import [-p num] [-R] [-P patch] [-f] [-d {o|a|n}] patchfile ...

Import external patches. The patches will be inserted following the current top patch, and must be pushed after import to apply them.

-p num

Number of directory levels to strip when applying (default=1)

-R

Apply patch in reverse.

-P patch

Patch filename to use inside quilt. This option can only be used when importing a single patch.

-f Overwrite/update existing patches.

-d {o|a|n}

When overwriting in existing patch, keep the old (o), all (a), or new (n) patch header. If both patches include headers, this option must be specified. This option is only effective when -f is used.

The quilt meta-data is now initialized.

mail {--mbox file|--send} [-m text] [-M file] [--prefix prefix] [--sender ...] [--from ...] [--to ...] [--cc ...] [--bcc ...] [--subject ...] [--reply-to message] [--charset ...] [--signature file] [first_patch [last_patch]]

Create mail messages from a specified range of patches, or all patches in the series file, and either store them in a mailbox file, or send them immediately. The editor is opened with a template for the introduction. Please see /usr/share/doc/quilt/README.MAIL for details. When specifying a range of patches, a first patch name of '-' denotes the first, and a last patch name of '-' denotes the last patch in the series.

-m text

Text to use as the text in the introduction. When this option is used, the editor will not be invoked, and the patches will be processed immediately.

-M file

Like the -m option, but read the introduction from file.

--prefix prefix

Use an alternate prefix in the bracketed part of the subjects generated. Defaults to ‘patch’.

--mbox file

Store all messages in the specified file in mbox format. The mbox can later be sent using formail, for example.

--send

Send the messages directly.

--sender

The envelope sender address to use. The address must be of the form ‘user@domain.name’. No display name is allowed.

--from, --subject

The values for the From and Subject headers to use. If no --from option is given, the value of the --sender option is used.

--to, --cc, --bcc

Append a recipient to the To, Cc, or Bcc header.

--charset

Specify a particular message encoding on systems which don’t use UTF-8 or ISO-8859-15. This character encoding must match the one used in the patches.

--signature file

Append the specified signature to messages (defaults to ˜/.signature if found; use ‘-’ for no signature).

--reply-to message

Add the appropriate headers to reply to the specified message.

new [-p n|-p ab] {patchname}

Create a new patch with the specified file name, and insert it after the topmost patch. The name can be prefixed with a sub-directory name, allowing for grouping related patches together.

-p n Create a -p n style patch (-p0 or -p1 are supported).

-p ab Create a -p1 style patch, but use a/file and b/file as the original and new filenames instead of the default dir.orig/file and dir/file names.

Quilt can be used in sub-directories of a source tree. It determines the root of a source tree by searching for a patches directory above the current working directory. Create a patches directory in the intended root directory if quilt chooses a top-level directory that is too high up in the directory tree.

next [patch]

Print the name of the next patch after the specified or topmost patch in the series file.

patches [-v] [--color[=always|auto|never]] {file} [files...]

Print the list of patches that modify any of the specified files. (Uses a heuristic to determine which files are modified by unapplied patches. Note that this heuristic is much slower than scanning applied patches.)

-v Verbose, more user friendly output.

--color[=always|auto|never]

Use syntax coloring (auto activates it only if the output is a tty).

pop [-afRqv] [--refresh] [num|patch]

Remove patch(es) from the stack of applied patches. Without options, the topmost patch is removed. When a number is specified, remove the specified number of patches. When a patch name is specified, remove patches until the specified patch end up on top of the stack. Patch names may include the patches/ prefix, which means that filename completion can be used.

-a Remove all applied patches.

-f Force remove. The state before the patch(es) were applied will be restored from backup files.

-R Always verify if the patch removes cleanly; don’t rely on timestamp checks.

-q Quiet operation.

-v Verbose operation.

--refresh

Automatically refresh every patch before it gets unapplied.

previous [patch]

Print the name of the previous patch before the specified or topmost patch in the series file.

push [-afqvm] [--fuzz=N] [--merge[=merge|diff3]] [--leave-rejects] [--color[=always|auto|never]] [--refresh] [num|patch]

Apply patch(es) from the series file. Without options, the next patch in the series file is applied. When a number is specified, apply the specified number of patches. When a patch name is specified, apply all patches up to and including the specified patch. Patch names may include the patches/ prefix, which means that filename completion can be used. The mtime of all touched files will be exactly the same to prevent time skews.

-a Apply all patches in the series file.

-q Quiet operation.

-f Force apply, even if the patch has rejects.

-v Verbose operation.

--fuzz=N

Set the maximum fuzz factor (default: 2).

-m, --merge[=merge|diff3]

Merge the patch file into the original files (see patch(1)).

--leave-rejects

Leave around the reject files patch produced, even if the patch is not actually applied.

--color[=always|auto|never]

Use syntax coloring (auto activates it only if the output is a tty).

--refresh

Automatically refresh every patch after it was successfully applied.

refresh [-p n|-p ab] [-u|-U num|-c|-C num] [-z[new_name]] [-f] [--no-timestamps] [--no-index] [--diffstat] [--sort] [--backup] [--strip-trailing-whitespace] [patch]

Refreshes the specified patch, or the topmost patch by default. Documentation that comes before the actual patch in the patch file is retained.

It is possible to refresh patches that are not on top. If any patches on top of the patch to refresh modify the same files, the script aborts by default. Patches can still be refreshed with -f. In that case this script will print a warning for each shadowed file, changes by more recent patches will be ignored, and only changes in files that have not been modified by any more recent patches will end up in the specified patch.

-p n Create a -p n style patch (-p0 or -p1 supported).

-p ab Create a -p1 style patch, but use a/file and b/file as the original and new filenames instead of the default dir.orig/file and dir/file names.

-u, -U num, -c, -C num

Create a unified diff (-u, -U) with num lines of context. Create a context diff (-c, -C) with num lines of context. The number of context lines defaults to 3.

-z[new_name]

Create a new patch containing the changes instead of refreshing the topmost patch. If no new name is specified, ‘-2’ is added to the original patch name, etc. (See the fork command.)

--no-timestamps

Do not include file timestamps in patch headers.

--no-index

Do not output Index: lines.

--diffstat

Add a diffstat section to the patch header, or replace the existing diffstat section.

-f Enforce refreshing of a patch that is not on top.

--backup

Create a backup copy of the old version of a patch as patch˜.

--sort Sort files by their name instead of preserving the original order.

--strip-trailing-whitespace

Strip trailing whitespace at the end of lines.

remove [-P patch] {file} ...

Remove one or more files from the topmost or named patch. Files that are modified by patches on top of the specified patch cannot be removed.

-P patch

Remove named files from the named patch.

rename [-P patch] new_name

Rename the topmost or named patch.

-P patch

Patch to rename.

revert [-P patch] {file} ...

Revert uncommitted changes to the topmost or named patch for the specified file(s): after the revert, 'quilt diff -z' will show no differences for those files. Changes to files that are modified by patches on top of the specified patch cannot be reverted.

-P patch

Revert changes in the named patch.

series [--color[=always|auto|never]] [-v]

Print the names of all patches in the series file.

--color[=always|auto|never]

Use syntax coloring (auto activates it only if the output is a tty).

-v Verbose, more user friendly output.

setup [-d path-prefix] [-v] [--sourcedir dir] [--fuzz=N] [--slow|--fast] {specfile|seriesfile}

Initializes a source tree from an rpm spec file or a quilt series file.

-d Optional path prefix for the resulting source tree.

--sourcedir

Directory that contains the package sources. Defaults to '.'.

-v Verbose debug output.

--fuzz=N

Set the maximum fuzz factor (needs rpm 4.6 or later).

--slow

Use the original, slow method to process the spec file. This is the default for now, but that might change in the future. In this mode, rpmbuild generates a working tree in a temporary directory while all its actions are recorded, and then everything is replayed from scratch in the target directory.

--fast Use an alternative, faster method to process the spec file. In this mode, rpmbuild is told to generate a working tree directly in the target directory. If the input is a series file, it is assumed that all archives have been extracted manually beforehand.

shell [command]

Launch a shell in a duplicate environment. After exiting the shell, any modifications made in this environment are applied to the topmost patch.

If a command is specified, it is executed instead of launching the shell.

snapshot [-d]

Take a snapshot of the current working state. After taking the snapshot, the tree can be modified in the usual ways, including pushing and popping patches. A diff against the tree at the moment of the snapshot can be generated with ‘quilt diff --snapshot’.

-d Only remove current snapshot.

top

Print the name of the topmost patch on the current stack of applied patches.

unapplied [patch]

Print a list of patches that are not applied, or all patches that follow the specified patch in the series file.

upgrade

Upgrade the meta-data in a working tree from an old version of quilt to the current version. This command is only needed when the quilt meta-data format has changed, and the working tree still contains old-format meta-data. In that case, quilt will request to run ‘quilt upgrade’.

COMMON OPTIONS TO ALL COMMANDS

--trace

Runs the command in bash trace mode (-x). For internal debugging.

--quiltrc file

Use the specified configuration file instead of ~/.quiltrc (or /etc/quilt.quiltrc if ~/.quiltrc does not exist). See the pdf documentation for details about its possible contents. The special value "-" causes quilt not to read any configuration file.

--version

Print the version number and exit immediately.

EXIT STATUS

The exit status is 0 if the sub-command was successfully executed, and 1 in case of error.

An exit status of 2 denotes that quilt did not do anything to complete the command. This happens in particular when asking to push when the whole stack is already pushed, or asking to pop when the whole stack is already popped. This behavior is intended to ease the scripting around quilt.

EXAMPLE OF WORKING TREE

```
work/
 patches/
|     series         (list of patches to apply)
|     patch1.diff    (one particular patch)
|     patch2.diff
|     ...
 .pc/
|     .quilt_patches (content of QUILT_PATCHES)
|     .quilt_series  (content of QUILT_SERIES)
|     patch1.diff/   (copy of patched files)
|    |     ...
|     patch2.diff/
|    |     ...
|     ...
 ...
```

The patches/ directory is precious as it contains all your patches as well as the order in which it should be applied.

The .pc/ directory contains some metadata about the current state of your patch series. Changing its content is not advised. This directory can usually be regenerated from the initial files and the content of the patches/ directory (provided that all patches were regenerated before the removal).

EXAMPLE

Please refer to the pdf documentation for a full example of use.

CONFIGURATION FILE

Upon startup, quilt evaluates the file .quiltrc in the user's home directory, or the file specified with the --quiltrc option. This file is a regular bash script. Default options can be passed to any COMMAND by defining a QUILT_${COMMAND}_ARGS variable. For example, QUILT_DIFF_ARGS="--color=auto" causes the output of quilt diff to be syntax colored when writing to a terminal.

In addition to that, quilt recognizes the following variables:

EDITOR

The program to run to edit files. If it isn't redefined in the configuration file, $EDITOR as defined in the environment will be used.

LESS

The arguments used to invoke the pager. Inherits the existing value of $LESS if LESS is already set in the environment, otherwise defaults to "-FRSX".

QUILT_DIFF_OPTS

Additional options that quilt shall pass to GNU diff when generating patches. A useful setting for C source code is "-p", which causes GNU diff to show in the resulting patch which function a change is in.

QUILT_PATCH_OPTS

Additional options that quilt shall pass to GNU patch when applying patches. For example, recent versions of GNU patch support the "--reject-format=unified" option for generating reject files in unified diff style (older patch versions used "--unified-reject-files" for that).

You may also want to add the "-E" option if you have issues with quilt not deleting empty files when you think it should. The documentation of GNU patch says that "normally this option is unnecessary", but when patch is in POSIX mode or if the patch format doesn't allow one to distinguish empty files from deleted files, patch deletes empty files only if the -E option is given. Beware that when passing -E to patch, quilt will no longer be able to deal with empty files, which is why using -E is no longer the default.

QUILT_DIFFSTAT_OPTS

Additional options that quilt shall pass to diffstat when generating patch statistics. For example, "-f0" can be used for an alternative output format. Recent versions of diffstat also support alternative rounding methods ("-r1", "-r2").

QUILT_PATCHES

The location of patch files, defaulting to "patches".

QUILT_SERIES

The name of the series file, defaulting to "series". Unless an absolute path is used, the search algorithm described above applies.

QUILT_PATCHES_PREFIX

If set to anything, quilt will prefix patch names it prints with their directory (QUILT_PATCHES).

QUILT_NO_DIFF_INDEX

By default, quilt prepends an Index: line to the patches it generates. If this variable is set to anything, no line is prepended. This is a shortcut to adding --no-index to both QUILT_DIFF_ARGS and QUILT_REFRESH_ARGS.

QUILT_NO_DIFF_TIMESTAMPS

By default, quilt includes timestamps in headers when generating patches. If this variable is set to anything, no timestamp will be included. This is a shortcut to adding --no-timestamps to both QUILT_DIFF_ARGS and QUILT_REFRESH_ARGS.

QUILT_PAGER

The pager quilt shall use for commands which produce paginated output. If unset, the values of GIT_PAGER or PAGER is used. If none of these variables is set, "less -R" is used. An empty value indicates that no pager should be used.

QUILT_COLORS

By default, quilt uses its predefined color set in order to be more comprehensible when distiguishing various types of patches, eg. applied/unapplied, failed, etc.

To override one or more color settings, set the QUILT_COLORS variable in following syntax - colon (:) separated list of elements, each being of the form <format name>=<foreground color>[;<background color>]

Format names with their respective default values are listed below, along with their usage(s). Color codes(values) are standard bash coloring escape codes. See more at http://tldp.org/LDP/abs/html/colorizing.html#AEN20229

diff_hdr Used in 'quilt diff' to color the index line. Defaults to 32 (green).

diff_add Used in 'quilt diff' to color added lines. Defaults to 36 (azure).

diff_mod Used in 'quilt diff' to color modified lines. Defaults to 35 (purple).

diff_rem Used in 'quilt diff' to color removed lines. Defaults to 35 (purple).

diff_hunk Used in 'quilt diff' to color hunk header. Defaults to 33 (brown/orange).

diff_ctx Used in 'quilt diff' to color the text after end of hunk header (diff --show-c-function generates this). Defaults to 35 (purple).

diff_cctx Used in 'quilt diff' to color the 15-asterisk sequence before or after a hunk. Defaults to 33 (brown/orange).

patch_fuzz
Used in ’quilt push’ to color the patch fuzz information. Defaults to 35 (purple).

patch_fail Used in ’quilt push’ to color the fail message. Defaults to 31 (red).

series_app Used in ’quilt series’ and ’quilt patches’ to color the applied patch names. Defaults to 32 (green).

series_top Used in ’quilt series’ and ’quilt patches’ to color the top patch name. Defaults to 33 (brown/orange).

series_una Used in ’quilt series’ and ’quilt patches’ to color unapplied patch names. Defaults to 0 (no special color).

In addition, the **clear** format name is used to turn off special coloring. Its value is 0; it is not advised to modify it.

The content of QUILT_COLORS supersedes default values. So the value diff_hdr=35;44 will get you the diff headers in magenta over blue instead of the default green over unchanged background. For that, add the following content to ˜/.quiltrc (or /etc/quilt.quiltrc):

```
QUILT_DIFF_ARGS="--color"
QUILT_COLORS=’diff_hdr=35;44’
```

AUTHORS

Quilt started as a series of scripts written by Andrew Morton (patch-scripts). Based on Andrew’s ideas, Andreas Gruenbacher completely rewrote the scripts, with the help of several other contributors (see AUTHORS file in the distribution).

This man page was written by Martin Quinson, based on information found in the pdf documentation, and in the help messages of each commands.

SEE ALSO

The pdf documentation, which should be under /usr/share/doc/quilt/quilt.pdf. Note that some distributors compress this file. **zxpdf**(**1**) can be used to display compressed pdf files.

diff(**1**), **patch**(**1**).