NAME

git−qapplied − list applied entries from the patch queue series file

SYNOPSIS

git qapplied [**−v** | **−−verbose**] [**−s** | **−−summary**] [**−1** | **−−last**]

DESCRIPTION

The **git qapplied** command displays, in series order, the names of all patches which have been registered as members of the active **Git−MQ** patch series, *and* which are currently in the *applied* state, (i.e. all registered and applied patches, in series order, up to and including the topmost applied patch, as indicated by the **git qtop** command).

If the **−−verbose** option is specified, the list is extended to include any *unapplied* patches which precede the topmost applied patch, in series order; each patch name is preceded by its sequence number in the series, and a single character flag, indicating its status, as described for the **git qseries** command.

If the **−−summary** option is specified, the commit summary line, if available in the patch header, is displayed following each patch name.

If the **−−last** option is specified, the output is reduced to the display of only the *one* patch, if any, which is the *applied* predecessor of the topmost applied patch; this is effectively equivalent to the output of the **git qprev** command.

OPTIONS

−1, **−−last**
Show only the predecessor of the current topmost patch.

−s, **−−summary**
Include summary lines from patch headers.

−v, **−−verbose**
Show patch sequence numbers and status flags.

EXIT STATUS

On successful completion, **git qapplied** reports an exit status code of 0. Any non-zero exit status code indicates that an error occurred.

COMPARISON WITH MERCURIAL QUEUES

Inspired by, and for the most part based on behavioural observation of **Mercurial**'s **MQ** extension, the **git qapplied** command exhibits fundamentally the same behaviour as its **hg qapplied** counterpart, *except* in the case when the **−−last** option is specified, *and* the patch immediately the topmost is *guarded*; (see the **CAVEATS AND BUGS** section, below).

CAVEATS AND BUGS

The output, when using the **−−last** option is not particularly useful; as implemented, it simply mimics the behaviour of the **git qprev** command, reporting the *most recently applied* predecessor of the topmost currently-applied patch. This differs from the behaviour of Mercurial's **hg qapplied −−last** command, which reports information relating to the *immediate* predecessor of the topmost applied patch, *regardless* of whether that predecessor is *applied*, *unapplied*, or *guarded*, but (unhelpfully) displays nothing at all, if the **−−verbose** option is not specified, when the predecessor is either *unapplied*, or *guarded*.

AUTHOR

SEE ALSO

git−qprev(1), **git−qseries**(1), **git−qtop**(1), **git−qunapplied**(1)