1.7.10

Git v1.7.10 Release Notes ==========================

Compatibility Notes ----------

- From this release on, the "git merge" command in an interactive session will start an editor when it automatically resolves the merge for the user to explain the resulting commit, just like the "git commit" command does when it wasn’t given a commit message.

```
If you have a script that runs "git merge" and keeps its standard
input and output attached to the user's terminal, and if you do not
want the user to explain the resulting merge commits, you can
export GIT_MERGE_AUTOEDIT environment variable set to "no", like
this:

#!/bin/sh
GIT_MERGE_AUTOEDIT=no
export GIT_MERGE_AUTOEDIT

to disable this behavior (if you want your users to explain their
merge commits, you do not have to do anything).  Alternatively, you
can give the "--no-edit" option to individual invocations of the
"git merge" command if you know everybody who uses your script has
Git v1.7.8 or newer.
```

- The "--binary/-b" options to "git am" have been a no-op for quite a while and were deprecated in mid 2008 (v1.6.0). When you give these options to "git am", it will now warn and ask you not to use them.

- When you do not tell which branches and tags to push to the "git push" command in any way, the command used "matching refs" rule to update remote branches and tags with branches and tags with the same name you locally have. In future versions of Git, this will change to push out only your current branch according to either the "upstream" or the "current" rule. Although "upstream" may be more powerful once the user understands Git better, the semantics "current" gives is simpler and easier to understand for beginners and may be a safer and better default option. We haven’t decided yet which one to switch to.

Updates since v1.7.9 ----------

UI, Workflows & Features

- various "gitk" updates.
  - show the path to the top level directory in the window title
  - update preference edit dialog
  - display file list correctly when directories are given on command line
  - make "git-describe" output in the log message into a clickable link
  - avoid matching the UNIX timestamp part when searching all fields
  - give preference to symbolic font names like sans & monospace
  - allow comparing two commits using a mark
  - "gitk" honors log.showroot configuration.

- Teams for localizing the messages from the Porcelain layer of commands are starting to form, thanks to Jiang Xin who volunteered to be the localization coordinator. Translated messages for simplified Chinese, Swedish and Portuguese are available.

- The configuration mechanism learned an "include" facility; an assignment to the include.path pseudo-variable causes the named file to be included in-place when Git looks up configuration variables.

- A content filter (clean/smudge) used to be just a way to make the recorded contents "more useful", and allowed to fail; a filter can now optionally be marked as "required".

- Options whose names begin with "--no-" (e.g. the "--no-verify" option of the "git commit" command) can be negated by omitting "no-" from its name, e.g. "git commit --verify".

- "git am" learned to pass "-b" option to underlying "git mailinfo", so that a bracketed string other than "PATCH" at the beginning can be kept.

- "git clone" learned "--single-branch" option to limit cloning to a single branch (surprise!); tags that do not point into the history of the branch are not fetched.

- "git clone" learned to detach the HEAD in the resulting repository when the user specifies a tag with "--branch" (e.g., "--branch=v1.0"). Clone also learned to print the usual "detached HEAD" advice in such a case, similar to "git checkout v1.0".

- When showing a patch while ignoring whitespace changes, the context lines are taken from the postimage, in order to make it easier to view the output.

- "git diff --stat" learned to adjust the width of the output on wider terminals, and give more columns to pathnames as needed.

- "diff-highlight" filter (in contrib/) was updated to produce more aesthetically pleasing output.

- "fsck" learned "--no-dangling" option to omit dangling object information.
- "git log -G" and "git log -S" learned to pay attention to the "-i" option. With "-i", "log -G" ignores the case when finding patch hunks that introduce or remove a string that matches the given pattern. Similarly with "-i", "log -S" ignores the case when finding the commit the given block of text appears or disappears from the file.
- "git merge" in an interactive session learned to spawn the editor by default to let the user edit the auto-generated merge message, to encourage people to explain their merges better. Legacy scripts can export GIT_MERGE_AUTOEDIT=no to retain the historical behavior. Both "git merge" and "git pull" can be given --no-edit from the command line to accept the auto-generated merge message.
- The advice message given when the user didn’t give enough clue on what to merge to "git pull" and "git merge" has been updated to be more concise and easier to understand.
- "git push" learned the "--prune" option, similar to "git fetch".
- The whole directory that houses a top-level superproject managed by "git submodule" can be moved to another place.
- "git symbolic-ref" learned the "--short" option to abbreviate the refname it shows unambiguously.
- "git tag --list" can be given "--points-at <object>" to limit its output to those that point at the given object.
- "gitweb" allows intermediate entries in the directory hierarchy that leads to a project to be clicked, which in turn shows the list of projects inside that directory.
- "gitweb" learned to read various pieces of information for the repositories lazily, instead of reading everything that could be needed (including the ones that are not necessary for a specific task).
- Project search in "gitweb" shows the substring that matched in the project name and description highlighted.
- HTTP transport learned to authenticate with a proxy if needed.
- A new script "diffall" is added to contrib/; it drives an external tool to perform a directory diff of two Git revisions in one go, unlike "difftool" that compares one file at a time.

Foreign Interface

- Improved handling of views, labels and branches in "git-p4" (in contrib).

- "git-p4" (in contrib) suffered from unnecessary merge conflicts when p4 expanded the embedded RCS-like keywords; it can be now told to unexpand them.
- Some "git-svn" updates.
- "vcs-svn"/"svn-fe" learned to read dumps with svn-deltas and support incremental imports.
- "git difftool/mergetool" learned to drive DeltaWalker.

Performance

- Unnecessary calls to parse_object() "git upload-pack" makes in response to "git fetch", have been eliminated, to help performance in repositories with excessive number of refs.

Internal Implementation (please report possible regressions)

- Recursive call chains in "git index-pack" to deal with long delta chains have been flattened, to reduce the stack footprint.
- Use of add_extra_ref() API is now gone, to make it possible to cleanly restructure the overall refs API.
- The command line parser of "git pack-objects" now uses parse-options API.
- The test suite supports the new "test_pause" helper function.
- Parallel to the test suite, there is a beginning of performance benchmarking framework.
- t/Makefile is adjusted to prevent newer versions of GNU make from running tests in seemingly random order.
- The code to check if a path points at a file beyond a symbolic link has been restructured to be thread-safe.
- When pruning directories that has become empty during "git prune" and "git prune-packed", call closedir() that iterates over a directory before rmdir() it.

Also contains minor documentation updates and code clean-ups.

Fixes since v1.7.9 ----------

Unless otherwise noted, all the fixes since v1.7.9 in the maintenance releases are contained in this release (see release notes to them for details).

- Build with NO_PERL_MAKEMAKER was broken and Git::I18N did not work with versions of Perl older than 5.8.3. (merge 5eb660e ab/perl-i18n later to maint).

- "git tag -s" honored "gpg.program" configuration variable since 1.7.9, but "git tag -v" and "git verify-tag" didn’t. (merge a2c2506 az/verify-tag-use-gpg-config later to maint).

- "configure" script learned to take "--with-sane-tool-path" from the command line to record SANE_TOOL_PATH (used to avoid broken platform tools in /usr/bin) in config.mak.autogen. This may be useful for people on Solaris who have saner tools outside /usr/xpg[46]/bin.

- zsh port of bash completion script needed another workaround.

□□□ #2
□ □□□ □□□ 2026-10-31 19:49:13 UTC
□ □□□ □□□ 2026-10-31 19:49:14 UTC