2.22.1

Git 2.22.1 Release Notes ==========

Fixes since v2.22 ----------

- A relative pathname given to "git init --template=<path> <repo>" ought to be relative to the directory "git init" gets invoked in, but it instead was made relative to the repository, which has been corrected.

- "git worktree add" used to fail when another worktree connected to the same repository was corrupt, which has been corrected.

- The ownership rule for the file descriptor to fast-import remote backend was mixed up, leading to unrelated file descriptor getting closed, which has been fixed.

- "git update-server-info" used to leave stale packfiles in its output, which has been corrected.

- The server side support for "git fetch" used to show incorrect value for the HEAD symbolic ref when the namespace feature is in use, which has been corrected.

- "git am -i --resolved" segfaulted after trying to see a commit as if it were a tree, which has been corrected.

- "git bundle verify" needs to see if prerequisite objects exist in the receiving repository, but the command did not check if we are in a repository upfront, which has been corrected.

- "git merge --squash" is designed to update the working tree and the index without creating the commit, and this cannot be countermanded by adding the "--commit" option; the command now refuses to work when both options are given.

- The data collected by fsmonitor was not properly written back to the on-disk index file, breaking t7519 tests occasionally, which has been corrected.

- Update to Unicode 12.1 width table.

- The command line to invoke a "git cat-file" command from inside "git p4" was not properly quoted to protect a caret and running a broken command on Windows, which has been corrected.

- "git request-pull" learned to warn when the ref we ask them to pull from in the local repository and in the published repository are different.

- When creating a partial clone, the object filtering criteria is recorded for the origin of the clone, but this incorrectly used a hardcoded name "origin" to name that remote; it has been corrected to honor the "--origin <name>" option.

- "git fetch" into a lazy clone forgot to fetch base objects that are necessary to complete delta in a thin packfile, which has been corrected.

- The filter_data used in the list-objects-filter (which manages a lazily sparse clone repository) did not use the dynamic array API correctly---'nr' is supposed to point at one past the last element of the array in use. This has been corrected.

- The description about slashes in gitignore patterns (used to indicate things like "anchored to this level only" and "only matches directories") has been revamped.

- The URL decoding code has been updated to avoid going past the end of the string while parsing %-<hex>-<hex> sequence.

- The list of for-each like macros used by clang-format has been updated.

- "git push --atomic" that goes over the transport-helper (namely, the smart http transport) failed to prevent refs to be pushed when it can locally tell that one of the ref update will fail without having to consult the other end, which has been corrected.

- "git clean" silently skipped a path when it cannot lstat() it; now it gives a warning.

- A codepath that reads from GPG for signed object verification read past the end of allocated buffer, which has been fixed.

- "git rm" to resolve a conflicted path leaked an internal message "needs merge" before actually removing the path, which was confusing. This has been corrected.

- The "git clone" documentation refers to command line options in its description in the short form; they have been replaced with long forms to make them more recognisable.

- The configuration variable rebase.rescheduleFailedExec should be effective only while running an interactive rebase and should not affect anything when running a non-interactive one, which was not the case. This has been corrected.

- "git submodule foreach" did not protect command line options passed to the command to be run in each submodule correctly, when the "--recursive" option was in use.

- Use "Erase in Line" CSI sequence that is already used in the editor support to clear cruft in the progress output.

- The codepath to compute delta islands used to spew progress output without giving the callers any way to squelch it, which has been fixed.

- The code to parse scaled numbers out of configuration files has been made more robust and also easier to follow.
- An incorrect list of options was cached after command line completion failed (e.g. trying to complete a command that requires a repository outside one), which has been corrected.
- "git rebase --abort" used to leave refs/rewritten/ when concluding "git rebase -r", which has been corrected.
- "git stash show 23" used to work, but no more after getting rewritten in C; this regression has been corrected.
- "git interpret-trailers" always treated ’#’ as the comment character, regardless of core.commentChar setting, which has been corrected.
- Code clean-up to avoid signed integer overlaps during binary search.
- "git checkout -p" needs to selectively apply a patch in reverse, which did not work well.
- The commit-graph file is now part of the "files that the runtime may keep open file descriptors on, all of which would need to be closed when done with the object store", and the file descriptor to an existing commit-graph file now is closed before "gc" finalizes a new instance to replace it.
- Code restructuring during 2.20 period broke fetching tags via "import" based transports.
- We have been trying out a few language features outside c89; the coding guidelines document did not talk about them and instead had a blanket ban against them.
- The internal diff machinery can be made to read out of bounds while looking for --funcion-context line in a corner case, which has been corrected.

Also contains various documentation updates, code clean-ups and minor fixups.

□□□ #2
□ □□□ □□□ 2026-10-31 19:56:36 UTC
□ □□□ □□□ 2026-10-31 19:56:37 UTC