

How to fix an Index which contains the same file name multiple times, but with different encodings

Background

The *Index* file (`.git/index`) contains meta-information related to the working tree and is constrained by Git to have exactly one entry per file (and stage level). While Git handles file names as raw *byte-sequences*, SmartGit interprets them as (*Unicode-Strings*) to display them more nicely. Certain parts of a byte sequence may not represent a valid encoding, hence the relevant parts of the byte sequence will be replaced by the 'Unicode replacement character' (U+FFFD). If the *Index* contains two byte sequences which differ only in 'unconvertible' parts, the resulting *Strings* may be identical. Such *Index* entries are confusing (not only SmartGit) and need to be fixed.

Note

This kind of problem might occur after updating to *msysgit 1.7.10* and not properly converting non-unicode file names.

Resolution

If an encoding problem in the *Index* is encountered, SmartGit will show an error message which contains information about the problematic file. The resolution is performed using Git from the command line:

- `cd` to your repository.
- Invoke `git status` to be sure you have a clean working tree and Index. If not clean, either commit or stash away your local modifications.
- For *msysgit*, on Windows: invoke `bash.exe` to enter Git's shell.
- Invoke `git ls-files | grep <readable-part-of-the-file-name>`.
- Inspect the output of the command and identify the offending entries: usually they will be almost identical, but only differ at certain positions where the *octal*/encoding (xyz) slightly varies. In the next step we will remove these offending entries from the repository (and re-add them again, if necessary). Hence, for every offending entry *<path-to-file>*:
- Make a backup of *<path-to-file>* and
- Invoke `git rm --cached "`printf "<path-to-file>`"` (Note: all the quotes are necessary!).

Note

On Windows, you may also work without `bash.exe`: just open a *command line (DOS box)* and invoke `git rm --cached <path-to-file-pattern>` with *<path-to-file-pattern>* being an unambiguous pattern of the file's path where the non-ASCII characters are replaced by ?. E.g.: `x?y.txt`.

- After all entries have been removed, invoke `git commit -m "clean up bad encoding of file names"`.
- If necessary, re-add the entries (this time with correct encoding) with `git add .` and finally commit with `git commit --amend -m "clean up bad encoding of file names"`.
- Now SmartGit should be able to open the repository. Don't forget to push your changes!