How to Handle Merge Conflicts with eGit

1.) Try to push.
   1. If successful, stop, you're done.
   2. If fails, e.g. **[rejected – non-fast-forward]**, continue to next step
2.) Commit code (without push)
3.) Team → pull.
   1. If succeeds, resolve any **merge conflicts** and go to step 7.
   2. If fails, continue to next step.
4.) Team → Merge
5.) If auto merge succeeds
   1. Go to step 7
6.) If auto merge fails
   1. Manually merge files
7.) Team → **Merge Tool** to verify merges
8.) Open Window → Show View → Other → **Git Staging** to see the modified files
9.) Right click files in “**Unstaged Changes**” and add to Git Index
10.) Commit and Push changes

Fig. 1 Push Error
Fig. 2 Merge Dialog

![Merge 'master' dialog]

- **Local**
  - master 23a31df Changes

- **Remote Tracking**
  - origin/master d403215 go

**Merge options**
- Commit (commit the result)
- No commit (prepare merge commit, but don’t commit yet)
- Squash (merge changes into working directory, but don’t create merge commit)

**Fast forward options**
- If a fast-forward, only update the branch pointer
- If a fast-forward, create a merge commit
- If not a fast-forward, fail

[Buttons: Merge, Cancel]
Fig. 3 Manual Merge

Fig. 4 Merged result
Fig. 5 Merge Tool

Fig. 6 Git Staging