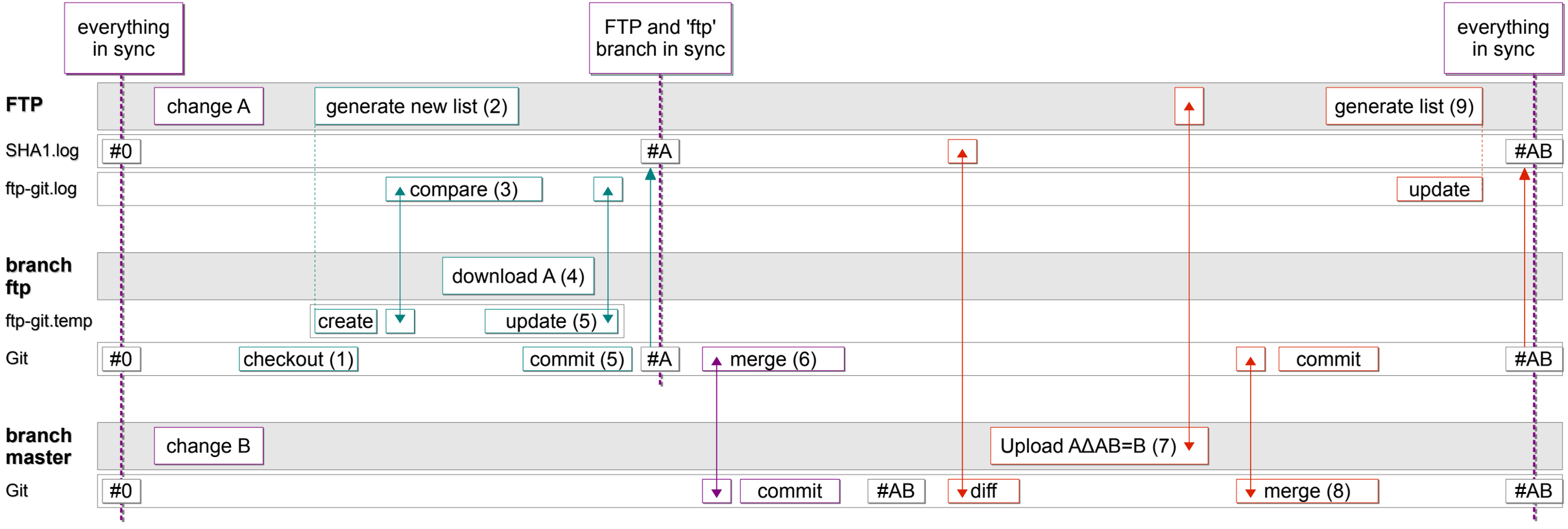
Synchronizing working directory with FTP



(1) Switch to branch 'ftp'. This branch is used for tracking the FTP server, so first of all we need to get the branch up to date.
(2) Generate and download the new list. Store temporarily.
(3) Compare the new list to the one in the working directory.
(4) Download the delta of the comparison. This equals the set of changes that were made on the server since the last sync.
(5) Now the FTP server and the 'ftp' branch are in sync, so we can upload the new list to the server, commit the downloads and upload the new SHA1 to the server as well.
(6) If there were actual changes from the download, ask the user to merge the 'ftp' branch back onto the master branch. This is where the users can check what has changed and decide how to handle it. After that, the master branch contains both the changes made on the server (A) and the local changes (B).
(7) Upload the delta between the FTP server and the master branch. That eqals the changes made locally (B).
(8) Since the FTP server has been updated, we have to update the 'git' branch as well to keep them in sync. We only uploaded B, so a simple (fast-forward) merge with the master is enough, we do not need to download from the server again.
(9) Step 8 was a shortcut for downloading, so we still need to bring the list up to date. After that, we update the SHA1 on the server with the one from the 'ftp' branch, and everything is in sync again.

external conditions
ftp-git down sync
git-ftp up sync