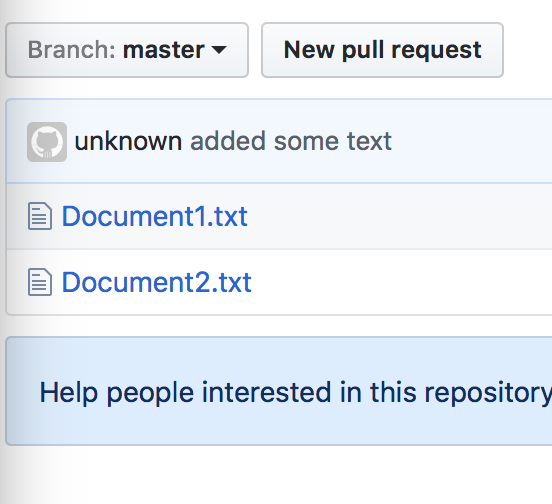
One of the workflows for software development teams using Git is the *feature branch* workflow. In this workflow, developers create short-lived branches for each feature. When a feature is complete, a developer sends a pull request for that branch to be merged into the master branch.

Part A of this document describes a scenario in which a pull request for a feature branch is sent and the branch merged without any conflicts.

Part B of this document describes a scenario in which pull requests for two different feature branches are sent and the second merge results in conflicts.

**The starting point**

Remote repository



Local Repository

iMac:GitPractice birdb$ git status

On branch master

Your branch is up-to-date with 'origin/master'.

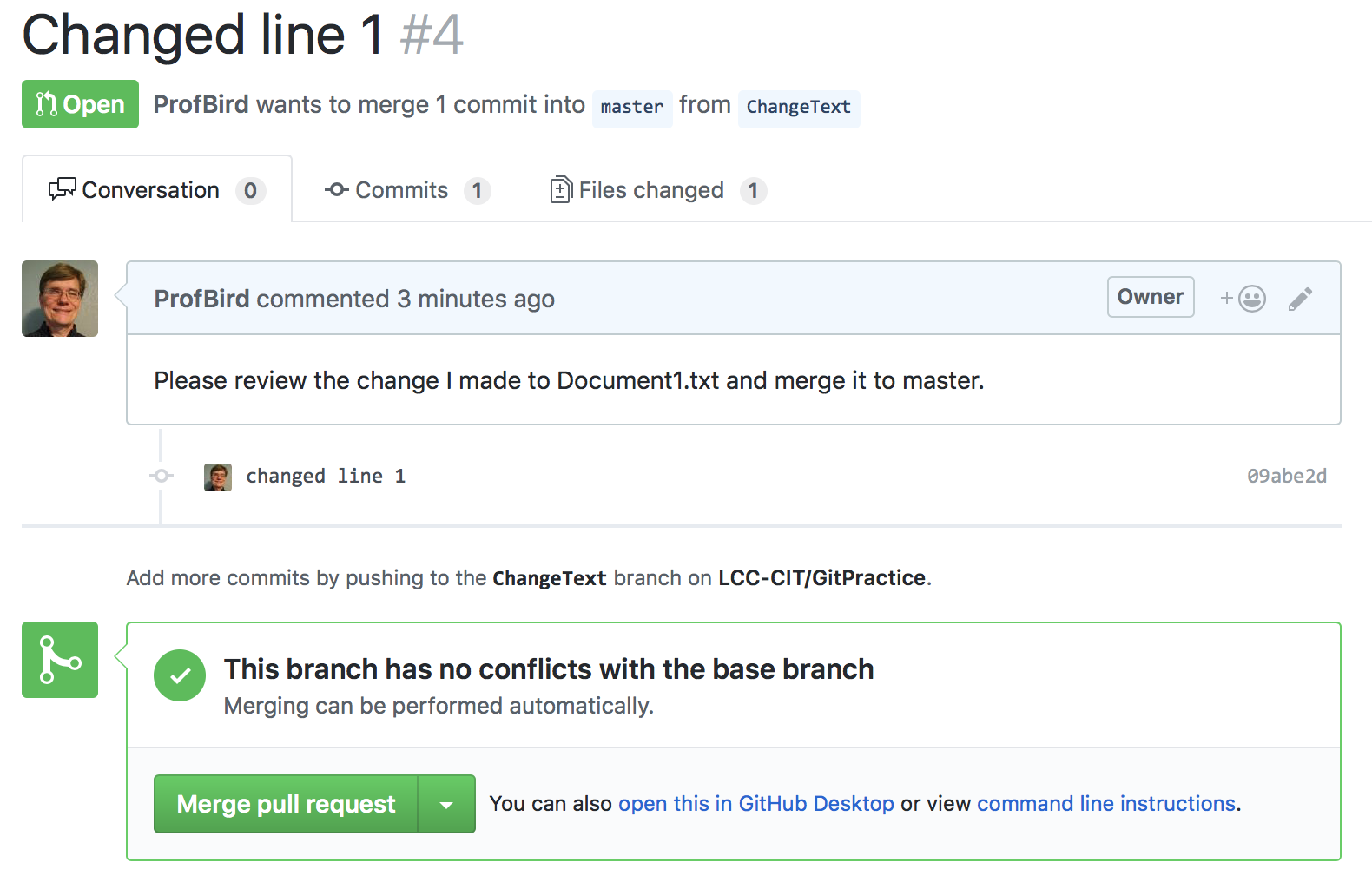
nothing to commit, working tree clean

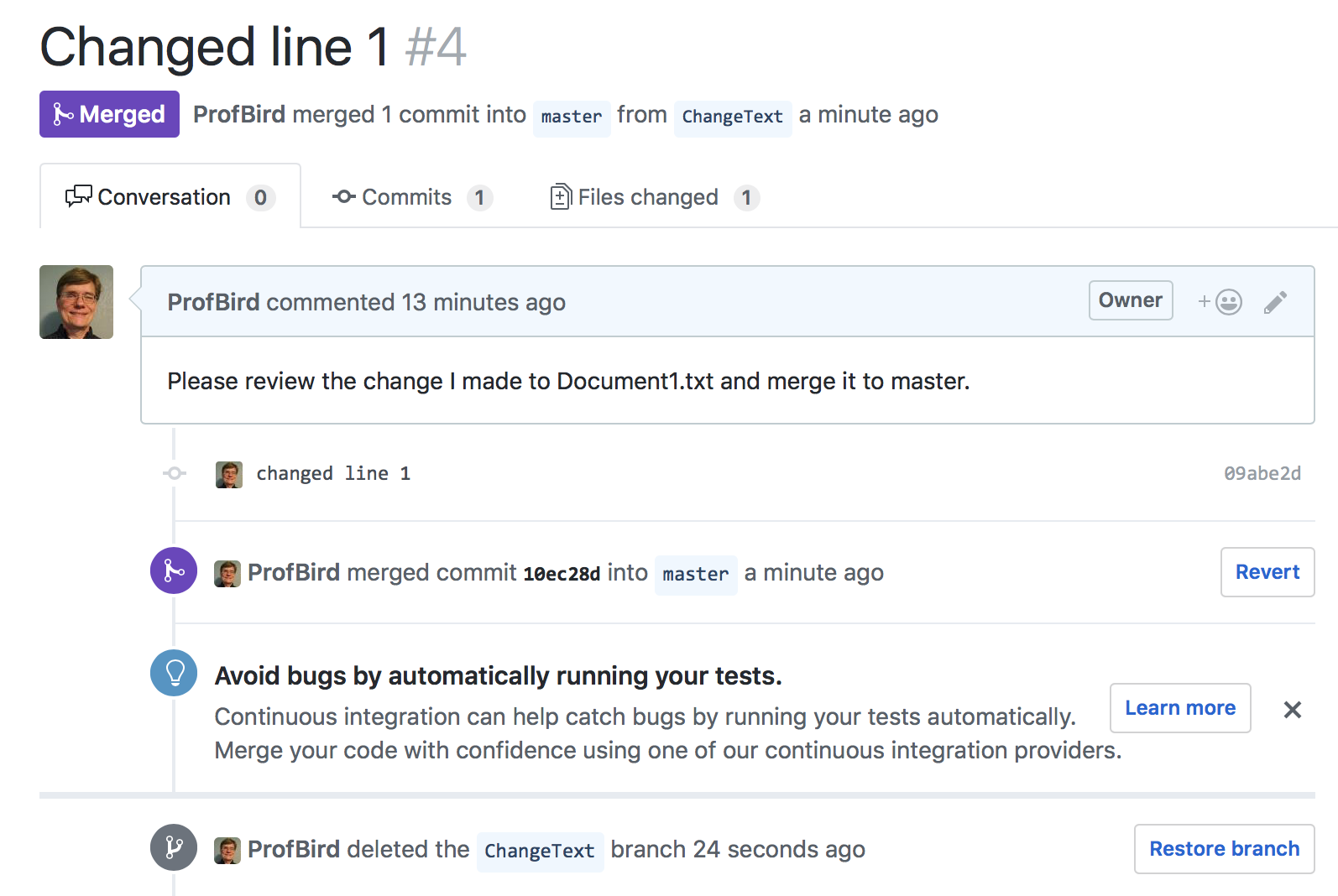
**A. Making changes on a branch that won’t conflict when merging**

In the local Repository

1. Create a new branch  
   (This branch will contain all the files currently in the master branch)  
     
   iMac:GitPractice birdb$ git checkout -b ChangeText  
   Switched to a new branch 'ChangeText'
2. Edit Document1.txt by changing line 1 from “Document 1” to “Document1.txt”  
     
   iMac:GitPractice birdb$ vi Document1.txt
3. Commit the change to the ChangeText branch  
     
   iMac:GitPractice birdb$ git commit -a -m "changed line 1"  
   [ChangeText 09abe2d] changed line 1  
    1 file changed, 1 insertion(+), 1 deletion(-)
4. Send a pull request. Note: there is no conflict! This is because no one else made changes to the same file on the master branch after I started my branch.

In the Remote Repository on GitHub



1. Merge the ChangeText branch into the master branch and delete the ChangeText branch (deleting the ChangeText branch is optional)  
   Note that I set a bad example here. I didn’t add a comment before merging the pull request.  
     
   

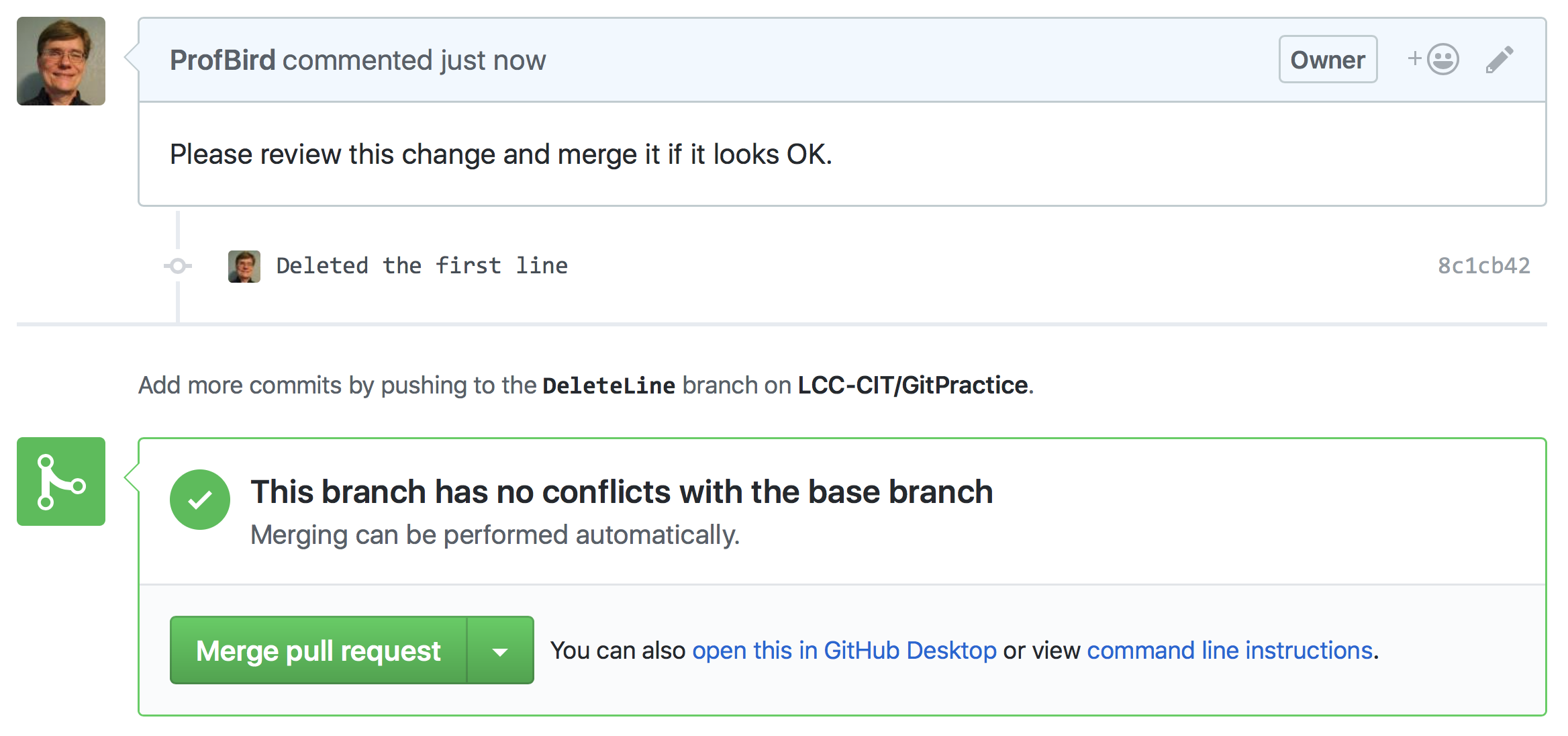
In the local Repository

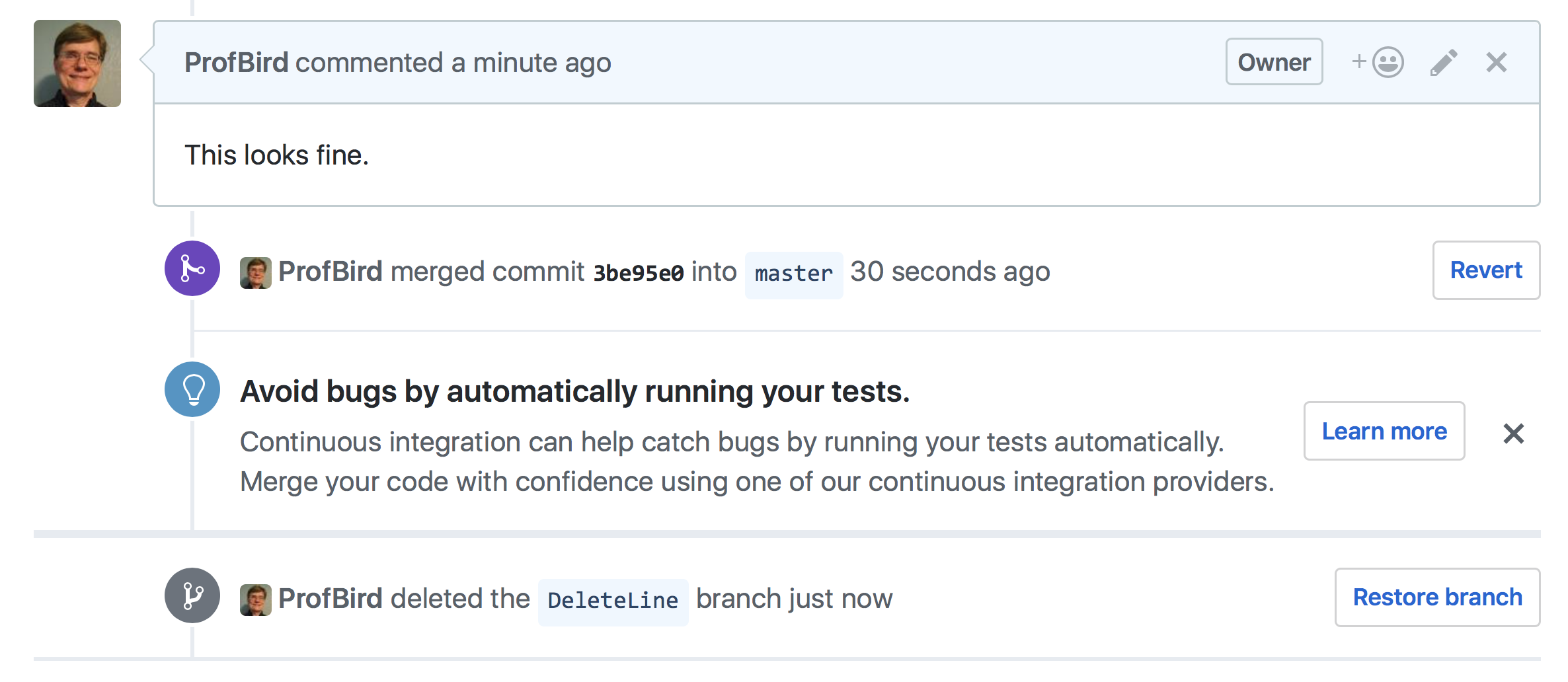
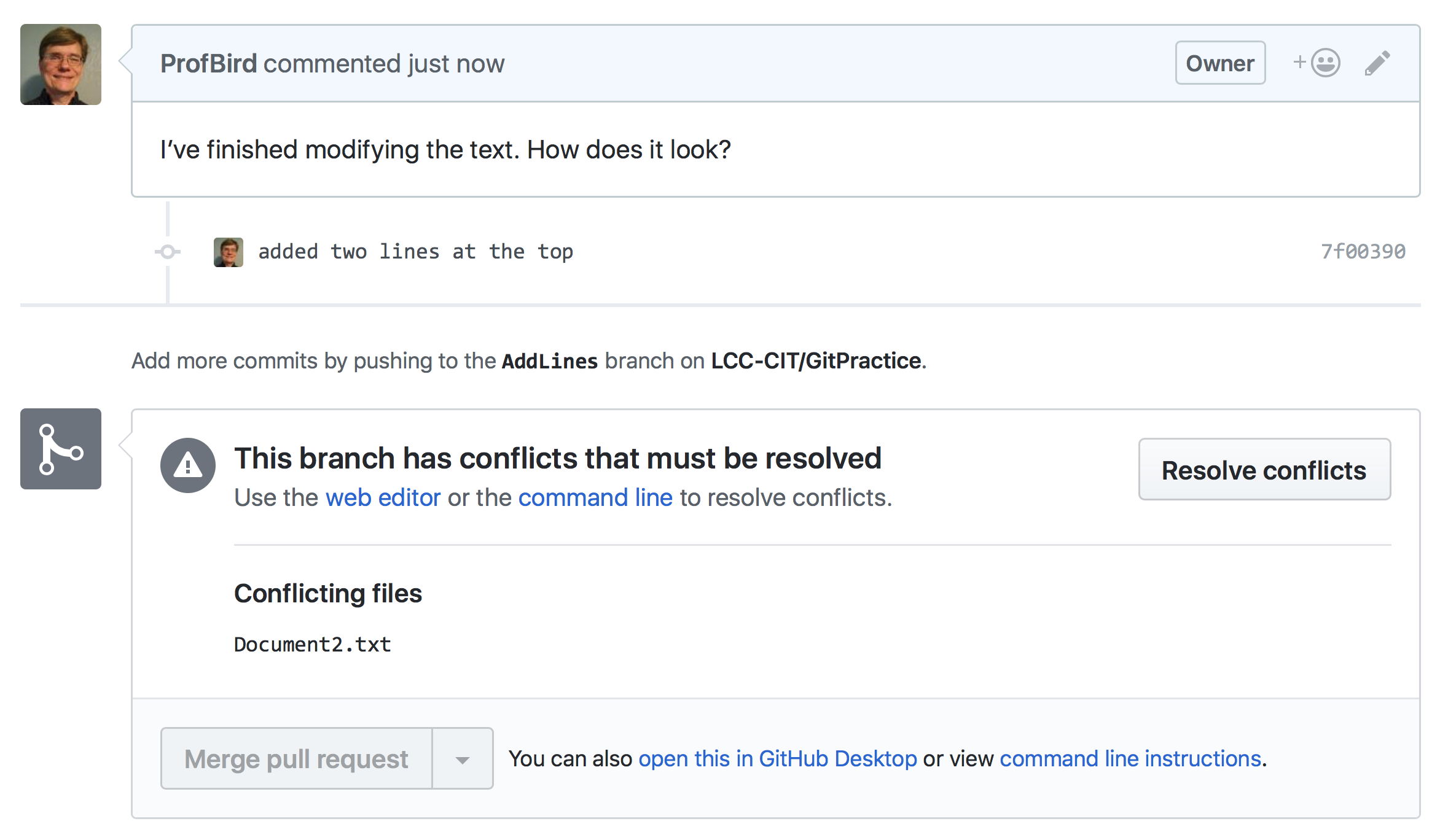
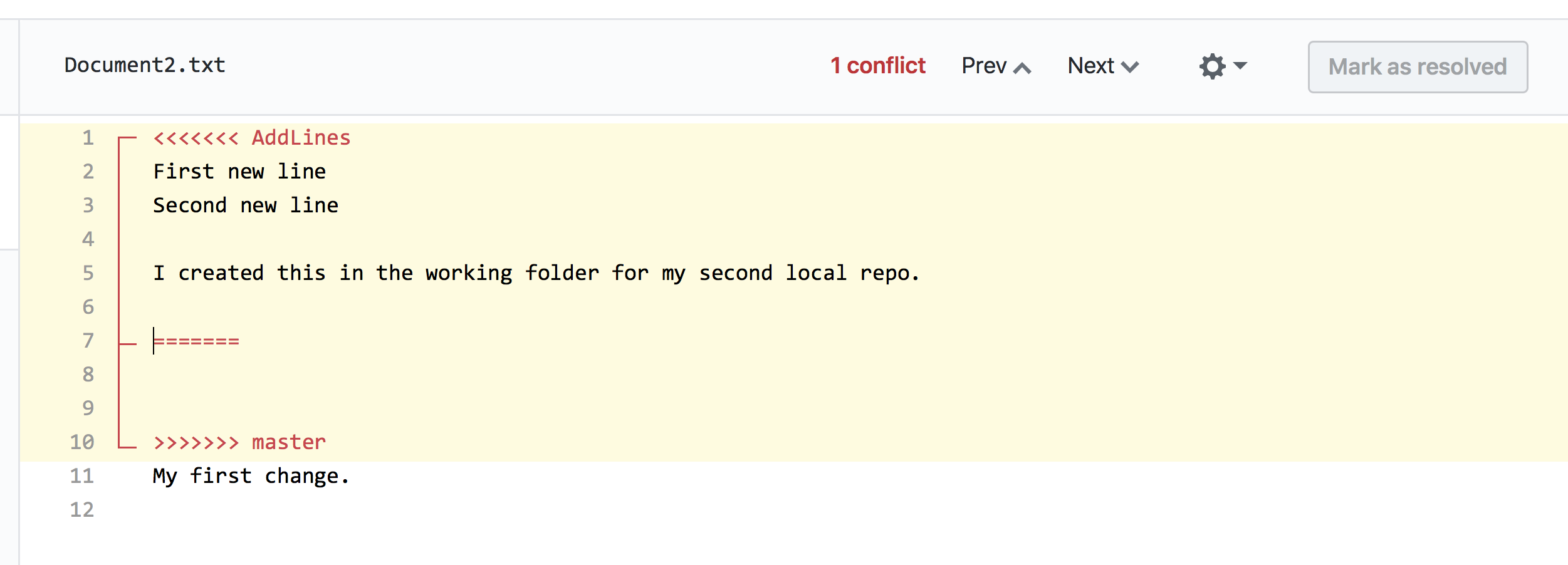
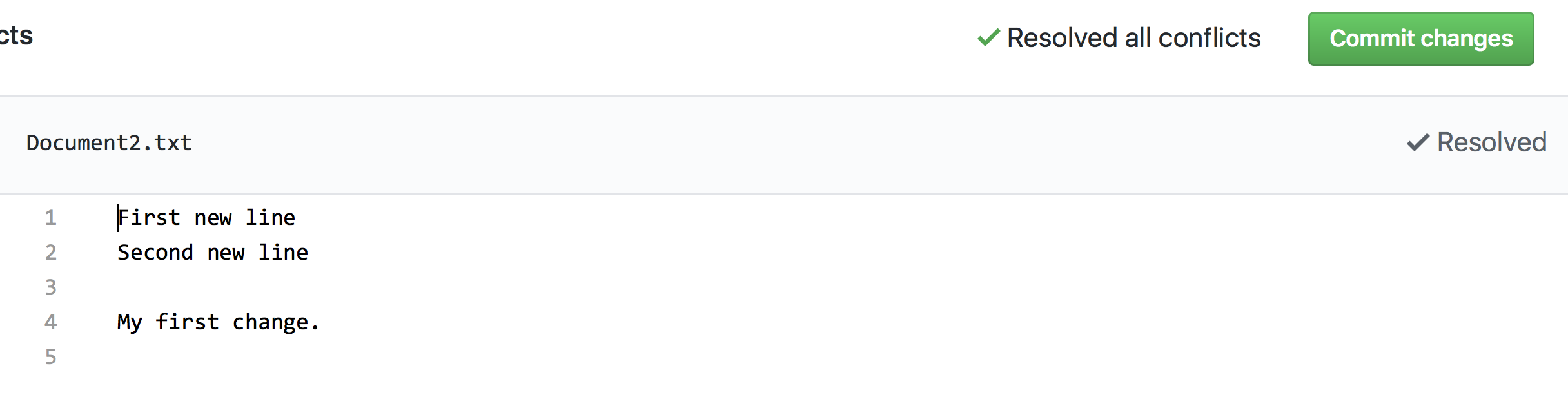
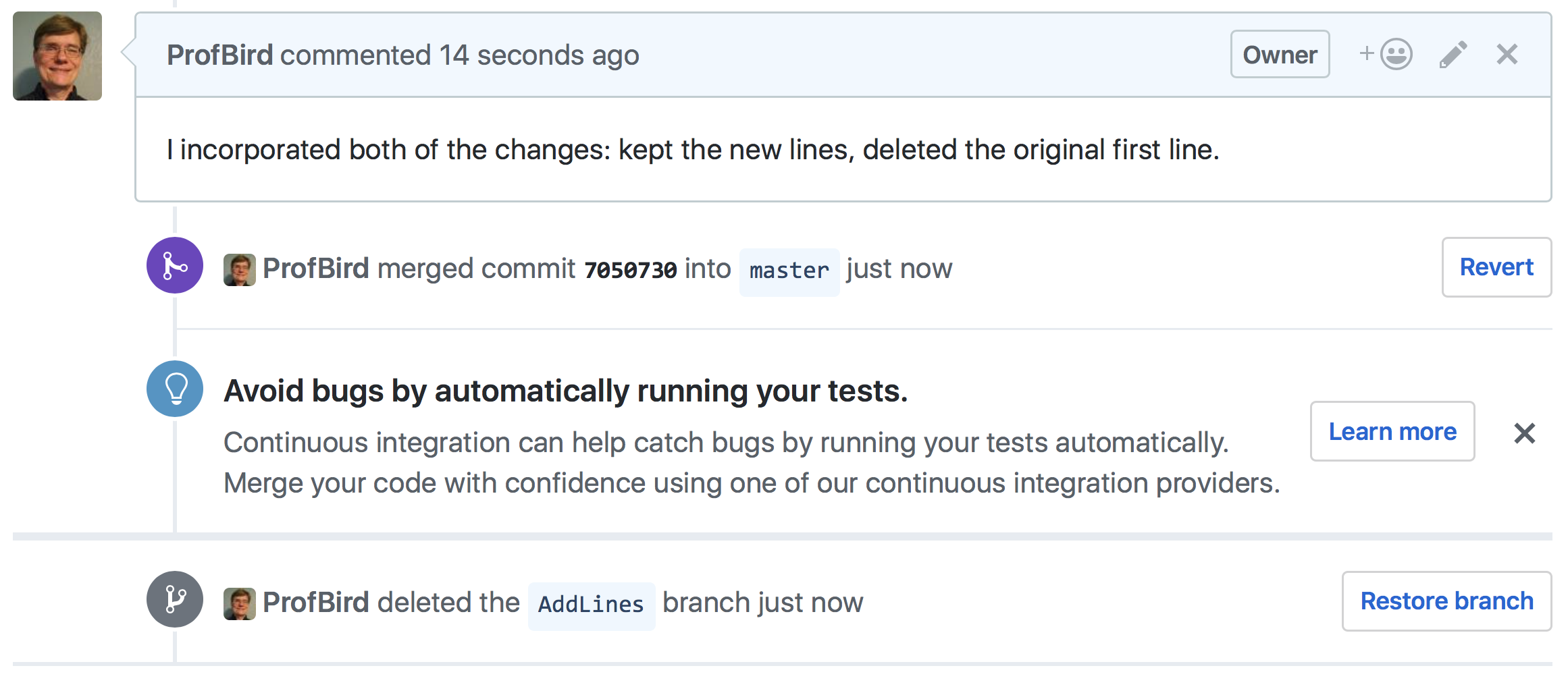
1. Pull the master branch from the remote repository  
     
   iMac:GitPractice birdb$ git checkout master  
   Switched to branch 'master'  
   our branch is behind 'origin/master' by 2 commits, and can be fast-forwarded.  
    (use "git pull" to update your local branch)  
     
   iMac:GitPractice birdb$ git pull origin  
   Updating a253bfe..10ec28d  
   Fast-forward  
    Document1.txt | 2 +-  
    1 file changed, 1 insertion(+), 1 deletion(-)
2. Delete the local ChangeText branch (optional)  
     
   iMac:GitPractice birdb$ git branch -d ChangeText  
   Deleted branch ChangeText (was 09abe2d).  
     
   1915200-M20158s-iMac:GitPractice birdb$ git status  
   On branch master  
   Your branch is up-to-date with 'origin/master'.  
   nothing to commit, working tree clean

**B. Making changes on branches that will conflict when merging**

In the local Repository

1. Create a new branch off of the master  
     
   iMac:GitPractice birdb$ git checkout -b AddLines  
   Switched to a new branch 'AddLines'
2. Add two new lines at the top of Document2.txt:  
     
   First new line  
   Second new line
3. Commit the change on the AddLines branch  
     
   iMac:GitPractice birdb$ git commit -a -m "added two lines at the top"  
   [AddLines 7f00390] added two lines at the top  
    1 file changed, 4 insertions(+), 1 deletion(-)
4. Start another branch off of master  
     
   iMac:GitPractice birdb$ git checkout master  
   Switched to branch 'master'  
     
   Your branch is up-to-date with 'origin/master'.  
   iMac:GitPractice birdb$ git checkout -b DeleteLine  
   Switched to a new branch 'DeleteLine'
5. Delete the first line in Document2.txt. The deleted line is:  
     
   I created this in the working folder for my second local repo.
6. Commit the change on the DeleteLine branch  
     
   1915200-M20158s-iMac:GitPractice birdb$ git commit -a -m "Deleted the first line"  
   [DeleteLine e2bf07c] Deleted the first line  
    1 file changed, 1 insertion(+), 1 deletion(-)

Send a pull request for the DeleteLine branch and delete the branch (deletion is optional)  
  
In the Remote Repository on GitHub  
  


1. Merge the DeleteLine branch  
     
   
2. Send a pull request for the change on the AddLines branch  
   Notice that this time I do have conflicts, because I made changes on another branch that branched from the master after this one did.  
   
3. Click on “Resolve Conflicts” and view the conflicting lines in Document2.txt  
   The empty space between the ======== and >>>>>>>>> shows that for the section shown above, there is nothing in the master.  
     
   
4. Decide how you want to resolve the conflict and edit the file. Then click “Mark as resolved”  
     
   
5. Commit the changes, add a comment, merge the pull request, and delete the branch (deletion is optional).  
     
   

In the local Repository

1. Pull the master branch from the remote repository  
     
   iMac:GitPractice birdb$ git checkout master  
   Switched to branch 'master'  
   Your branch is up-to-date with 'origin/master'.  
     
   iMac:GitPractice birdb$ git pull origin  
   Updating 3be95e0..7050730  
   Fast-forward  
    Document2.txt | 3 ++-  
    1 file changed, 2 insertions(+), 1 deletion(-)  
     
   iMac:GitPractice birdb$ git status  
   On branch master  
   Your branch is up-to-date with 'origin/master'.  
   nothing to commit, working tree clean
2. Delete the local feature branches (optional)  
     
   iMac:GitPractice birdb$ git branch -d AddLines  
   Deleted branch AddLines (was 7f00390).  
     
   iMac:GitPractice birdb$ git branch -d DeleteLine  
   Deleted branch DeleteLine (was 8c1cb42)  
     
   iMac:GitPractice birdb$ git status  
   On branch master  
   Your branch is up-to-date with 'origin/master'.  
   nothing to commit, working tree clean