

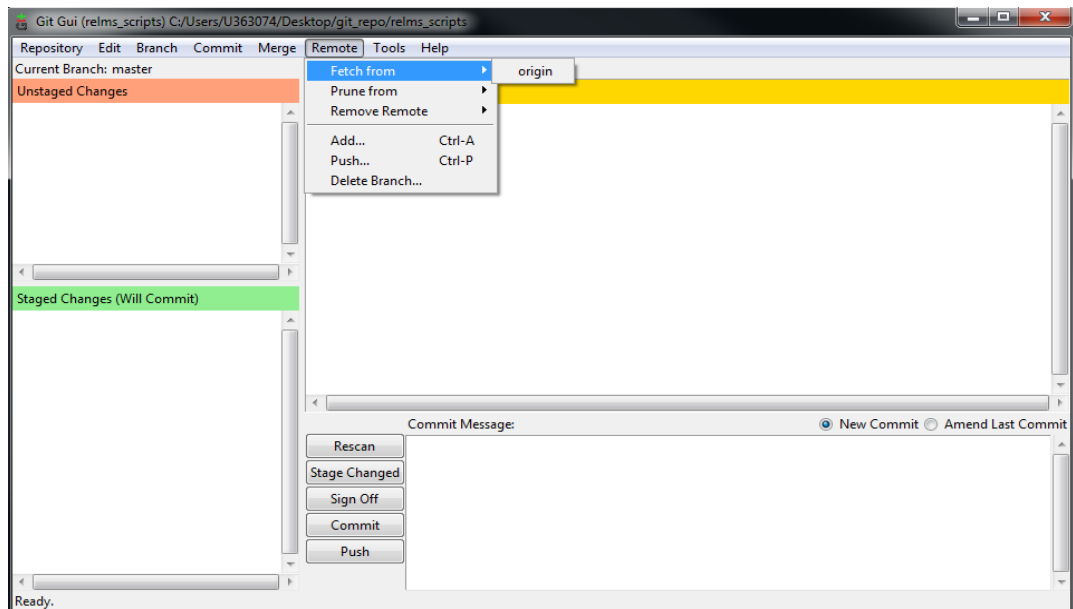
Git-Blit Tutorial

We use Git-Blit to allow automation teams to maintain their automation assets within a central repo. This ensures that all of the automation is in sync and no one is overriding another person's changes while working on the same project. Git-Blit allows for version control across the automation assets- if the code breaks it can be reverted back to a recent version without losing too much data.

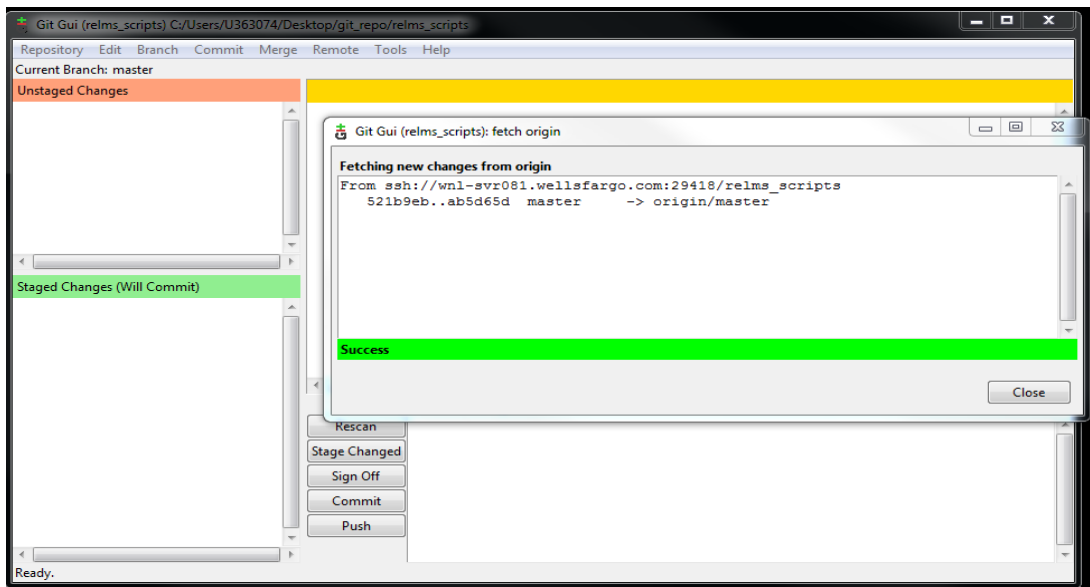
It is an open source tool that we use extensively because it gives us complete control over installation and security. Security is in check because it can easily be deployed within the firewall. Below are the steps that walk you through the Git-Blit Setup and basic usage (checking-in, checkin-out, etc.):

Git Fetch

1. Open Git GUI
2. Click on **Remote\Fetch From\Origin**

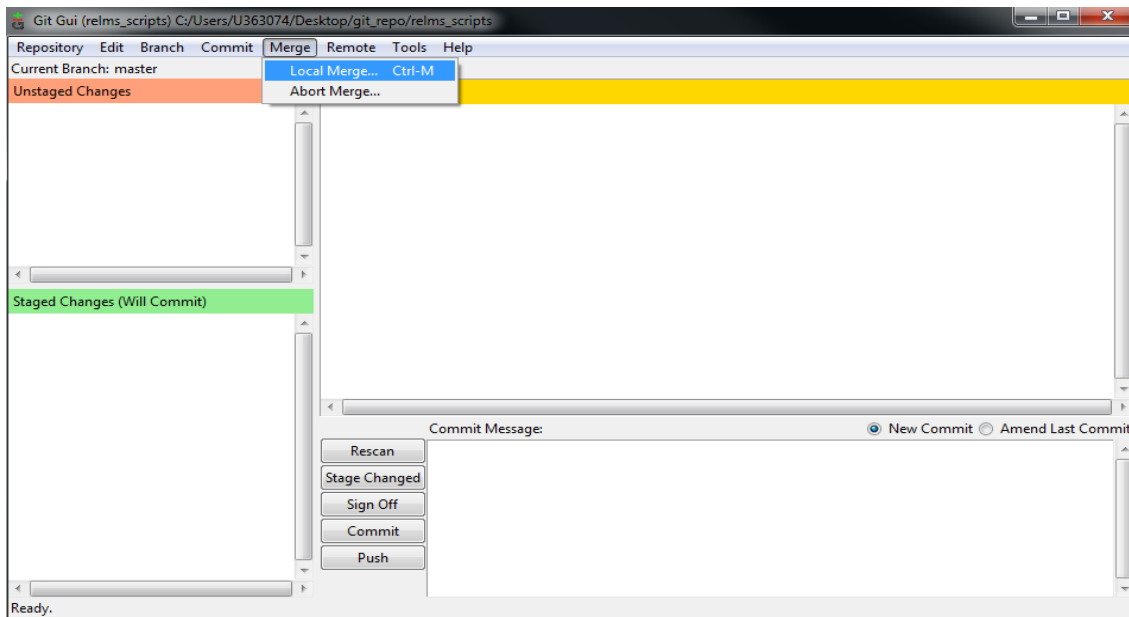


3. Wait till you see Fetching new changes from Origin Success message as shown in the following screenshot.

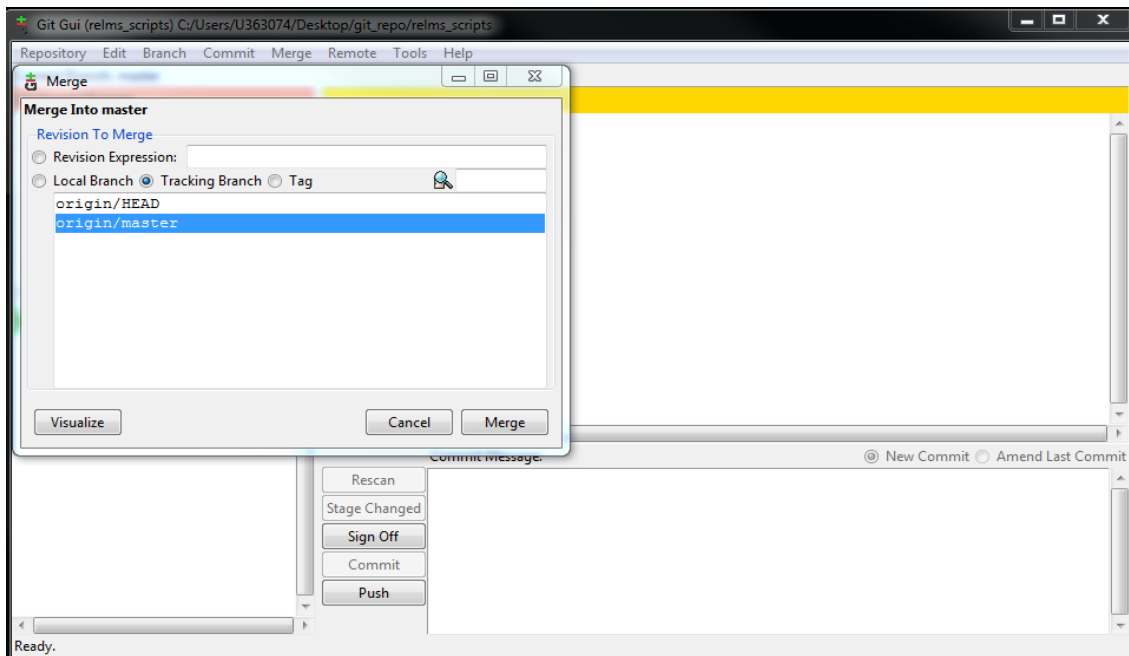


Merge Changes Locally

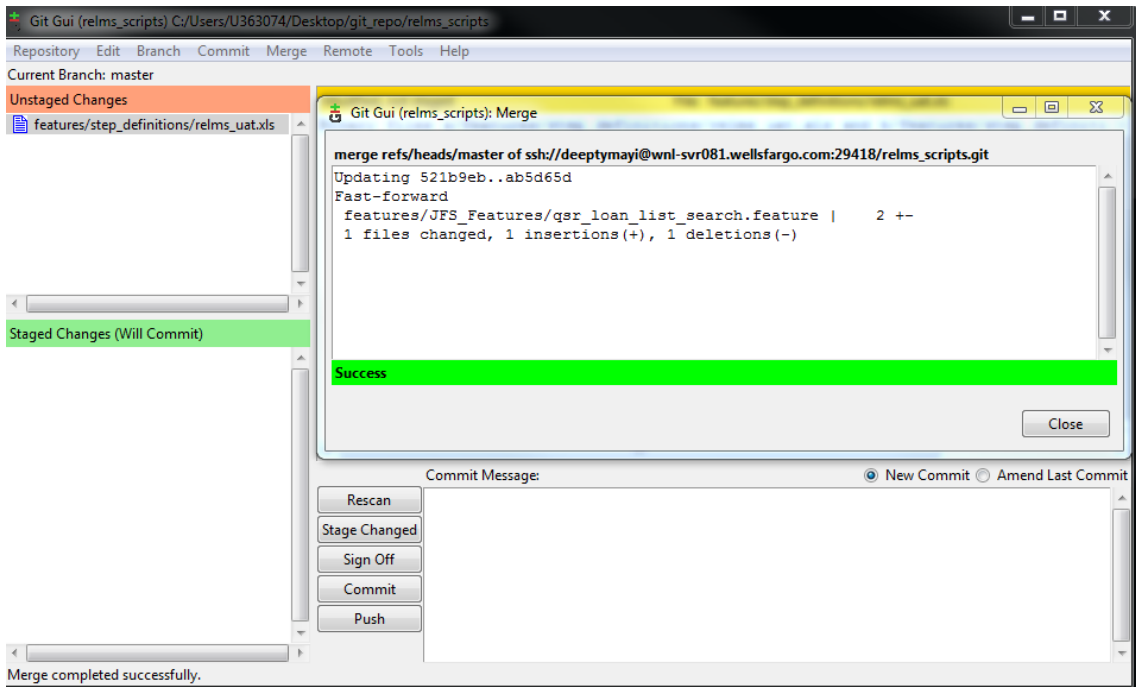
1. Click on Merge Tab and click Local Merge



2. Choose the “**Tracking Branch**” where you want the merge to take place and click **Merge** button

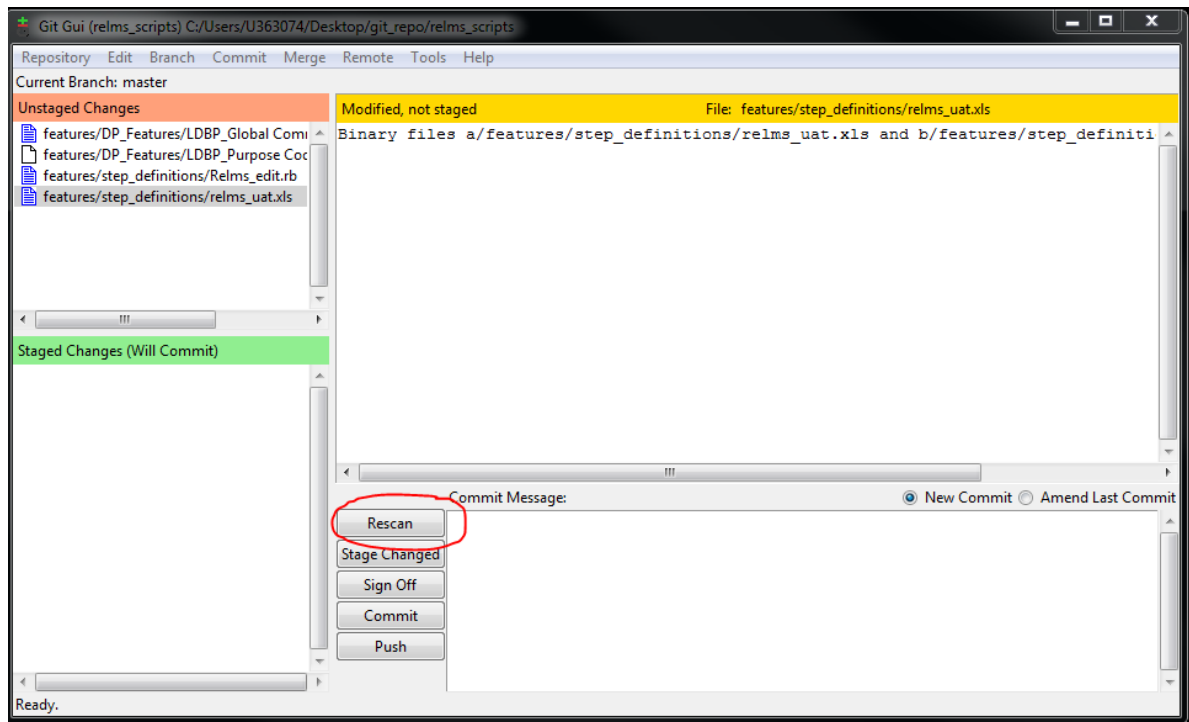


3. Wait to see the **Merge Success** message and close the Merge pop up window.

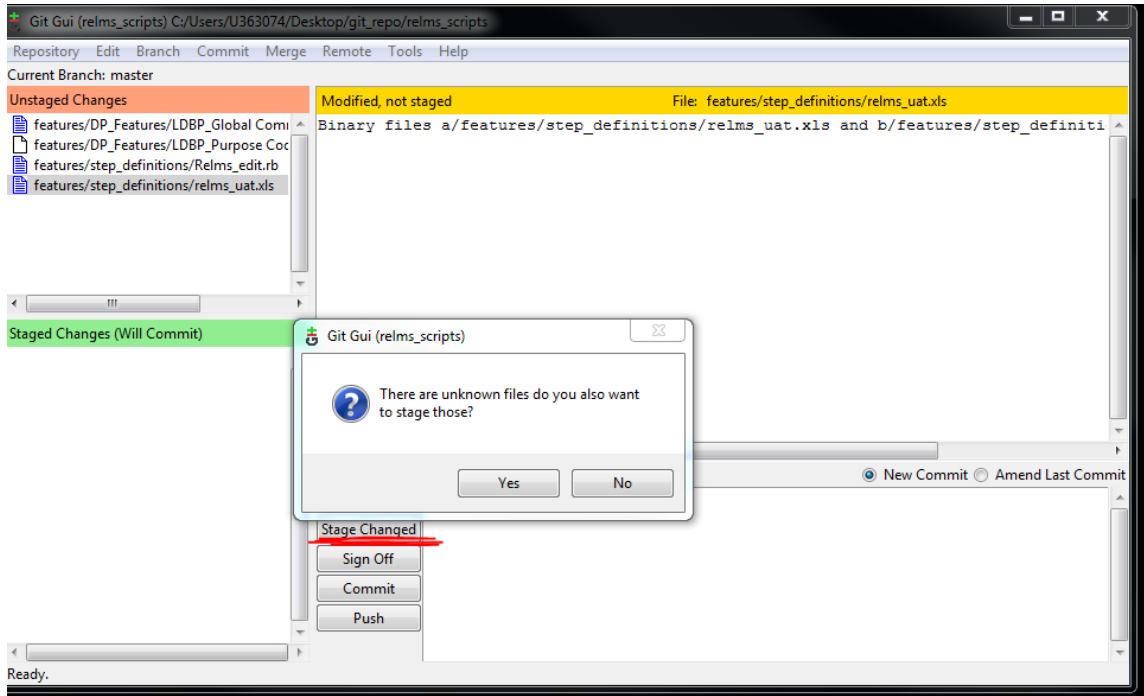


Pushing Your Local Changes to Git Repository

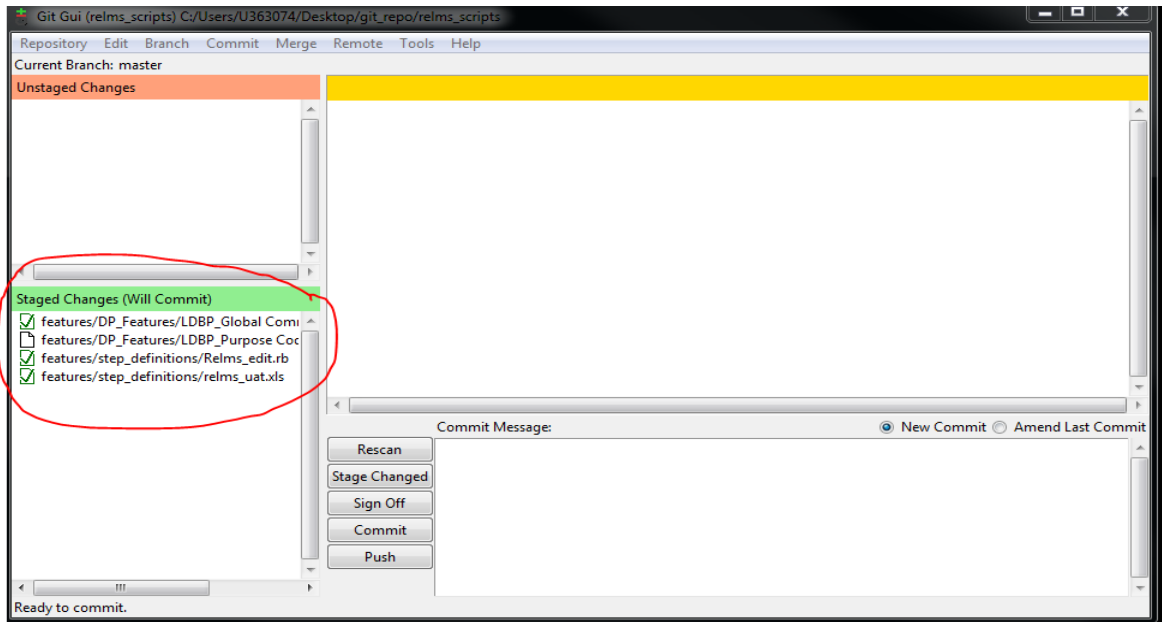
1. Click “**Rescan**” to see your local changes.



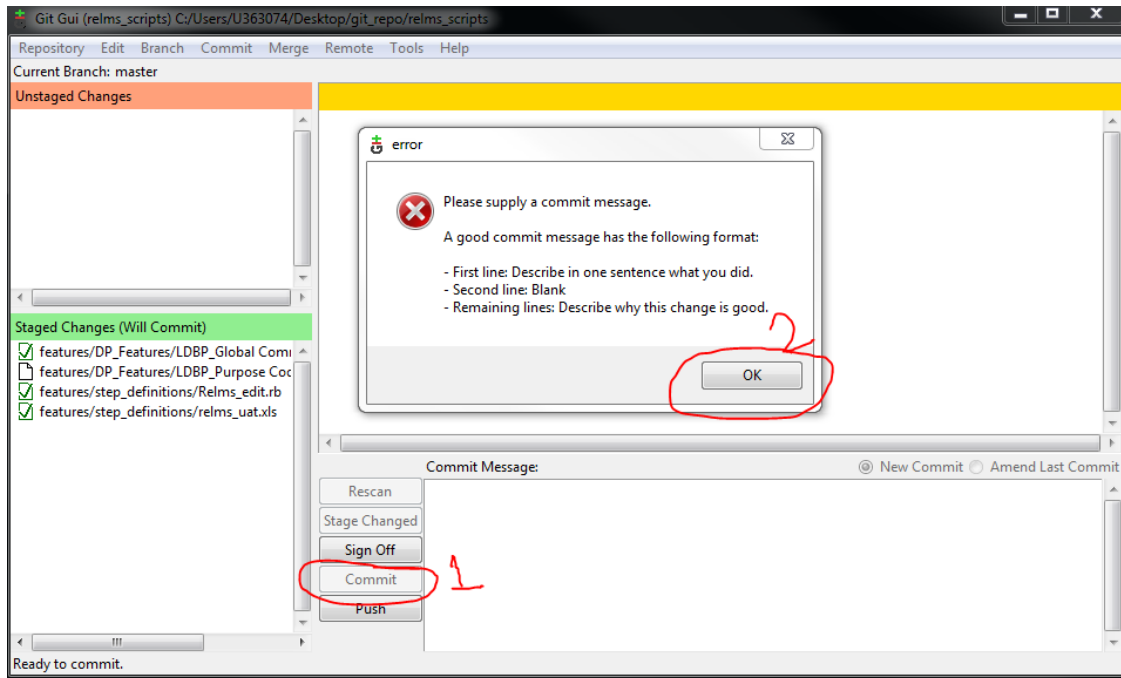
- Click on “Stage Changed” to stage your local changes to the Git Repository. Click “YES” on the browser alert.



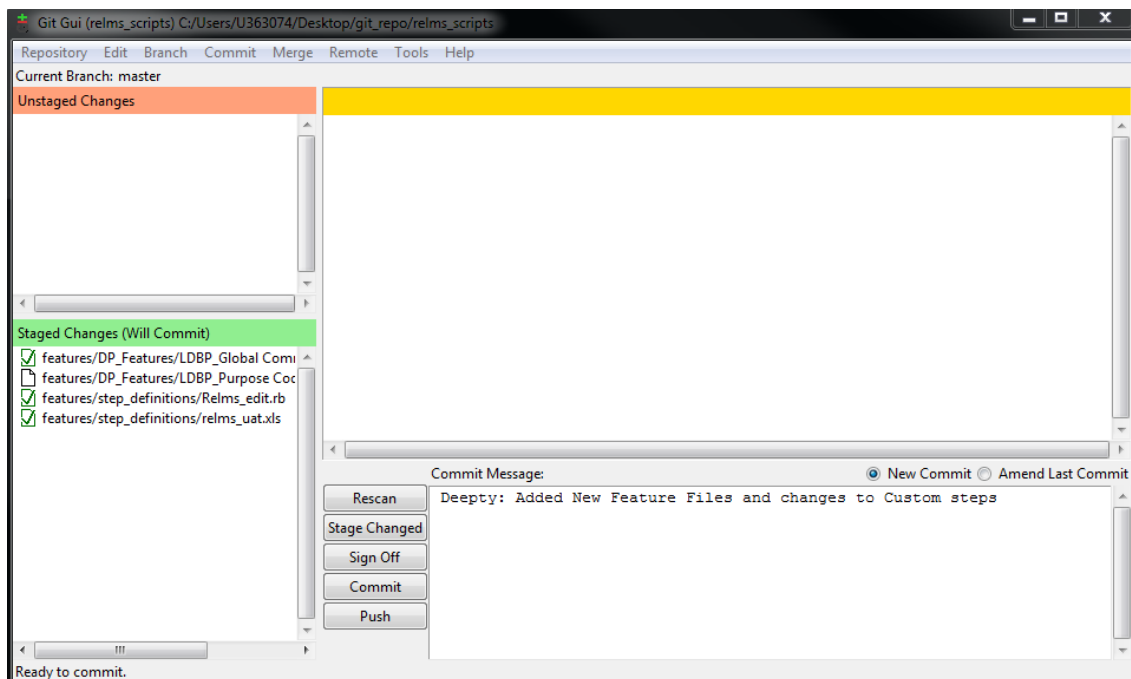
- Once you click “Yes” you should be able to see all the new changes marked under the Staged Changes Column.



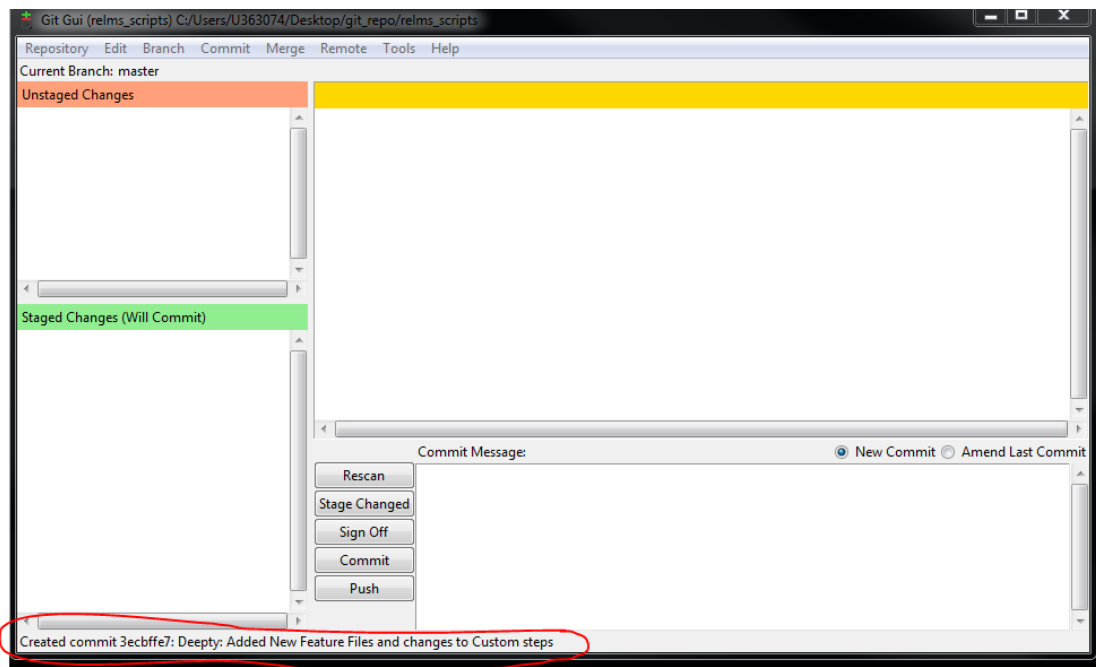
4. After staging the changes, click “**Commit**” to commit the changes to the Git Repository and then click “OK” on the browser alert.



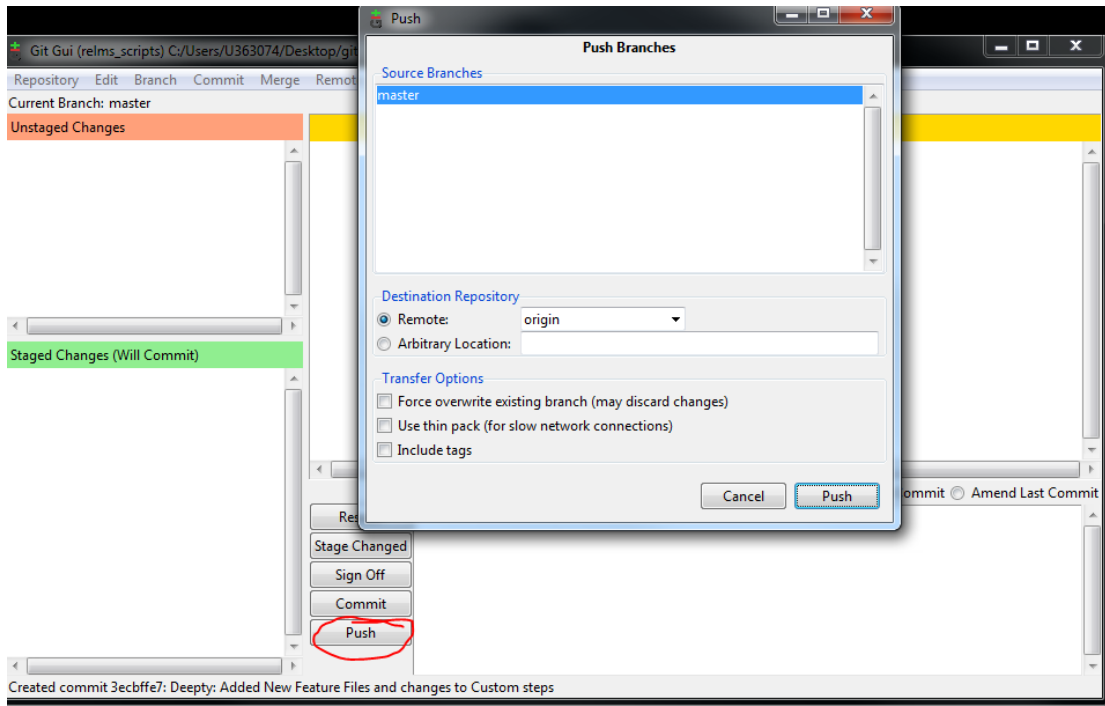
5. Add your custom Commit Message relevant to the changes that you are willing to commit to Git.



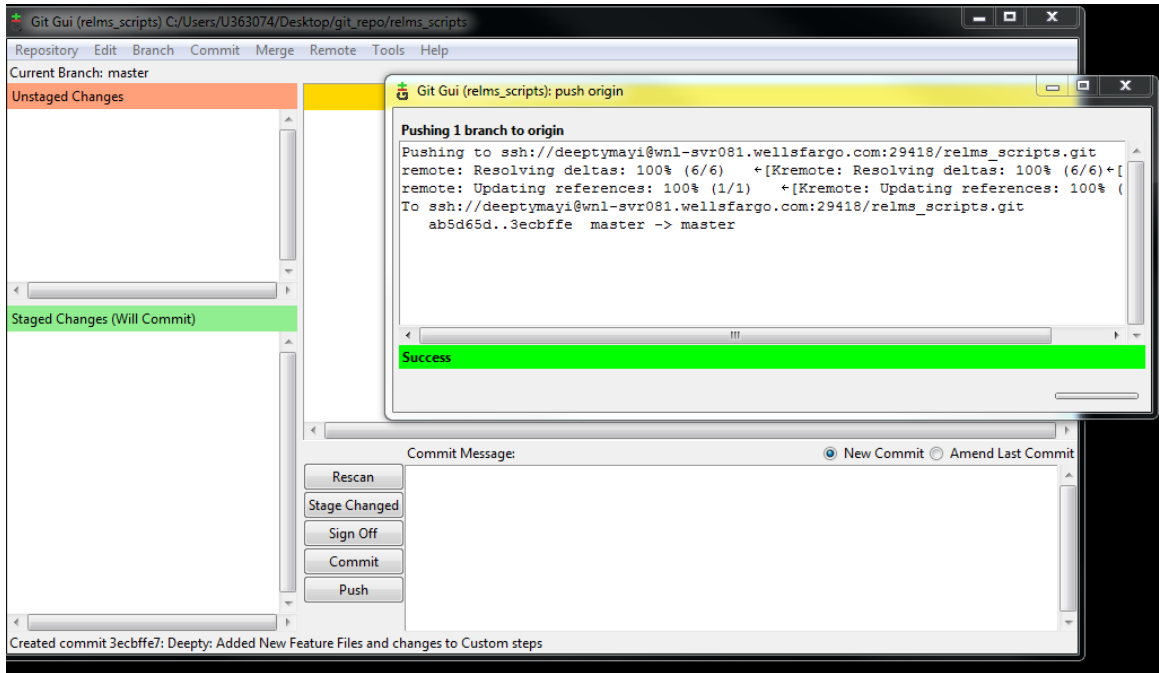
- Click “**Commit**” again and wait for your commit message to appear at the bottom of Git GUI page.



- Click “**Push**” on Git GUI and click “**Push**” again after verifying that the correct source branch is selected on the Push Branches pop up window. The application might ask you to enter your password for authentication so enter your password.



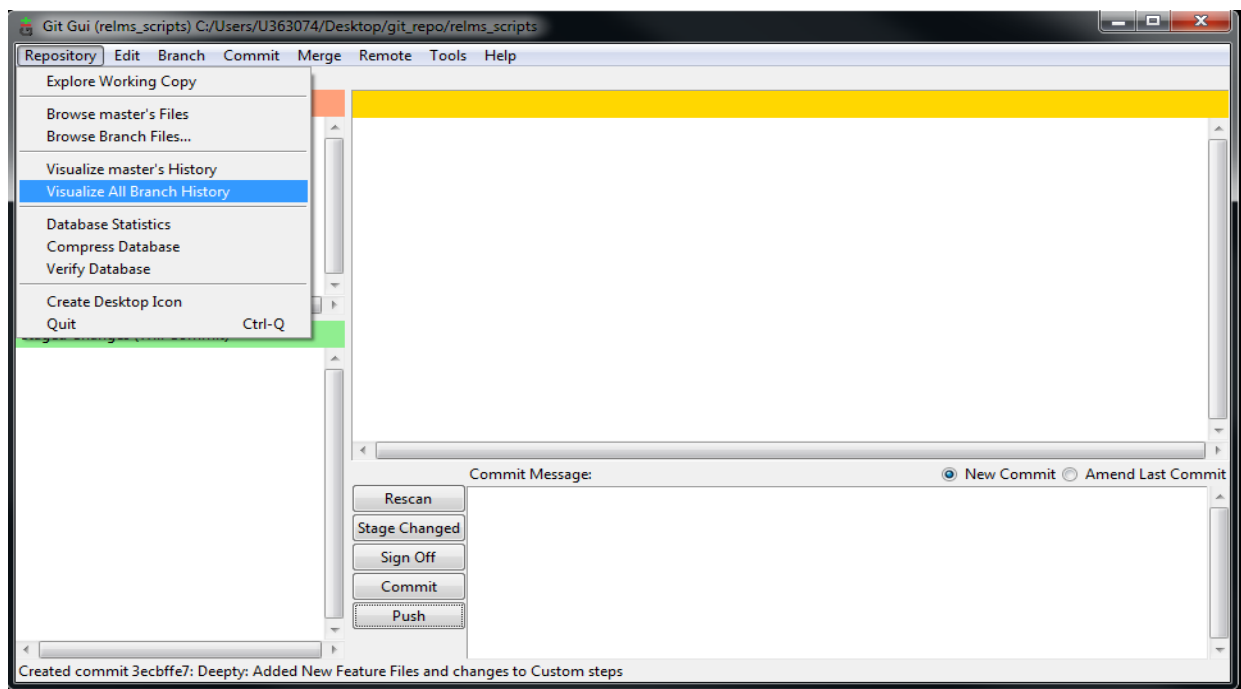
8. Wait to See “Success” on Push Origin pop up window.



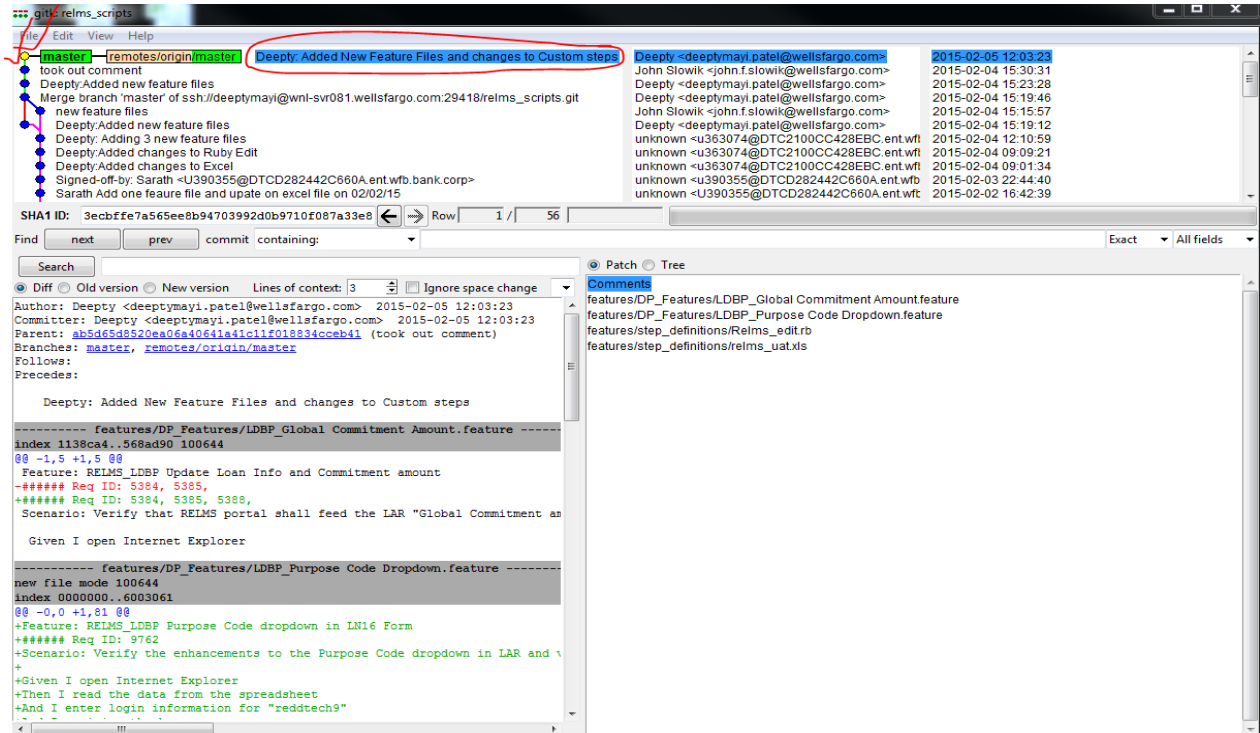
Validating that Changes have been Pushed Successfully

Once you have pushed your changes to the Git Repo, it is important to validate that you have all the changes pushed successfully and your changes are merged with changes made by other members of your team in the same Git branch. We can follow the steps below to validate the same:

1. Click on “**Repository**” tab on Git GUI and click “**Visualize All Branch History**”



- You should see the recent commit message added by you and your name at the top of the branch. The “Yellow” dot is your update. If the yellow dot is at the top, it means you have synced all the changes made by you and your team members and you have made the latest update/commit to Git Repository.



- You can also verify that your changes have been successfully pushed to Git Repository by logging in to Gitblit. The first commit under the recent activity list in “**My dashboard**” should be you.

Here is a screenshot:

The screenshot shows the Gitblit dashboard interface. At the top, there is a navigation bar with the Gitblit logo and links for 'my dashboard', 'repositories', 'activity', 'search', and 'filters'. Below the navigation bar, the page title is 'Welcome to Gitblit' with sub-links for 'shared', 'owned', 'active', and 'projects'. The main content area is titled 'recent activity' and shows a summary of activity over the last 7 days, including 24 commits by 3 authors. Two pie charts are displayed: 'active repositories' (100%) and 'active authors' (63% and 33%). The activity list shows the following commits:

- today, Thursday, February 5, 2015**
1 commit to master of relms_scripts by Deepty
 3ecbfff Deepty: Added New Feature Files and changes to Custom steps
- yesterday, Wednesday, February 4, 2015**
8 commits to master of relms_scripts by 3 authors
 ab5d65 took out comment
 521b9e Deepty: Added new feature files
 adf1ad Merge branch 'master' of ssh://deedlymay@wml-svr081.wellsfargo.com:29418/ir...
 24b9ba Deepty: Added new feature files
 79191f Deepty: Adding 3 new feature files
 3 more commits >
- 2 days ago, Monday, February 2, 2015**
7 commits to master of relms_scripts by 2 authors
 ba66d Sarath Add one feaure file and upate on excel file on 02/02/15
 e99fa6 New Feature files and updated Ruby steps for 2nd Feb
 a62ebd Signed-off-by: Sarath <U390355@DTC282442C660A.ent.wfb.bank.com>
 c76abe Merge branch 'master' of ssh://john@wml-svr081.wellsfargo.com:29418/relms_s...
 377fb1 Merged John's excel changes and ruby file changes.
 2 more commits >