SVN To Git Migration

Portfolio & DevOps Toolchain - Internal

Exported on 12/21/2022

Table of Contents

1 Step 1: Get the SVN repository users and map them with Git repository users	34
2 Step 2: Clone the SVN repository on the local system and convert to the git repository	5
3 Step 3: Convert SVN tag to Git tag	6
4 Step 4: Convert the SVN branch to the Git branch	7
5 Step 5: Create an empty git repository, this will be the target repository	8
6 Step 6: Push the local converted repository to target the empty repository: .	11

Before we start the Git repository migration, make sure the following pre-requisite is in place.

- The powerful system, If you are dealing with a number of SVN repositories at the same time.
- Good and continue network bandwidth.
- Git and TortoiseSVN client installs on your system.

Below is the very high-level step for SVN to Git repository migration.

- 1. Get the SVN repository users and map them with Git repository users
- 2. Clone SVN repository on the local system and convert to git repository
- 3. Convert SVN tag to Git tag
- 4. Convert SVN branch to Git branch
- 5. Create an empty git repository, this will be the target repository.
- 6. Push local converted repository to target empty repository.

Now let's start the repository preparation.



(i) Note

In this example, we have used the SVN "web" repository and Vishalmande, HitendraChauhan as mapped git users.

1 Step 1: Get the SVN repository users and map them with Git repository users

The AUthors/Users from the SVN repository along with historical commits can be extracted using the below command.

 $svn \ log \ -q \ --trust-server-cert \ --username \ E0492097 \ --password \ passwords \ >svn://svn.tcc.etn.com/MTS/TDM/Web/ | awk -F '|' '/^r/{sub("^ ", "", $2); sub(" $", "", $2); print $2" = "$2" < "$2">"}'| sort \ -u \ > authors-transform.txt$

Here user name and password will be your SVN repository credentials.

```
E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone
$ svn log -q --trust-server-cert --username E0492097 --password Vm6152021 svn:/
/svn.tcc.etn.com/MTS/TDM/Web/ |awk -F '|' '/^r/{sub("^ ", "", $2); sub(" $", "",
$2); print $2" = "$2" <"$2">"}'| sort -u > authors-transform.txt

E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone
$ |
```

Mapp SVN users to Git users, seem below screenshot.

```
Section of the composition of th
```

2 Step 2: Clone the SVN repository on the local system and convert to the git repository

Clone SVN repository by using following command

git svn clone svn://svn.tcc.etn.com/MTS/TDM/Web/ --authors-file=./authors-transform.txt -T trunk -b branches -t tag

Here, -T Trank à Trunk Directory.

- -b branches à branches directory.
- -t tag à tags directory.

Once you are done with Clone, the result will look like the one below. Also, you will find the SVN repository folder along with .git metadata.

```
MINGW64:/c/Users/E0492097/OneDrive - Eaton/Documents/SVN CLone
                                                                                                                   X
                          Framework/Model/obj/Debug/G2Model.pdb
                          TDMWebService/bin/G2Model.dll
  685 = af81276a04bdce9649dc314824e5f52b4e950282 (refs/remotes/origin/Release_1_6
 9_1_Salem2)
Auto packing the repository in background for optimum performance. See "git help gc" for manual housekeeping.
Enumerating objects: 9078, done.
Counting objects: 100% (9078/9078), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8827/8827), done.
Writing objects: 100% (9078/9078), done.
Total 9078 (delta 6608), reused 0 (delta 0)
Removing duplicate objects: 100% (256/256), done.
Computing commit graph generation numbers: 100% (685/685), done.
Updating files: 100% (1075/1075), done.
Checked out HEAD:
   svn://svn.tcc.etn.com/MTS/TDM/Web/trunk r680
creating empty directory: TDMConnector/LabviewSupport creating empty directory: TDMEnterprise/TDMDataImport/TDMDataImport/bin/Release creating empty directory: db/transactions
creating empty directory: db/txn-protorevs
  0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone
```

3 Step 3: Convert SVN tag to Git tag

Convert the SVN tag into Git tag by using the below command.

If you have multiple tags then:

for t in (git for-each-ref--format='%(refname:short)' refs/remotes/tags); do git tag <math>(t/tags)/ \$t && git branch -D -r \$t; done

If you have few tags then:

for t in \$(git for-each-ref --format='%(refname:short)' refs/remotes/tags); do git tag -a "\$t" -m"say farewell to SVN" "/tags/\$t"; done

```
E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/Web (master)

$ for t in $(git for-each-ref --format='%(refname:short)' refs/remotes/tags); do git tag ${t/tags\//} $t && git branch -D -r $t; done

E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/Web (master)

-$ |
```

4 Step 4: Convert the SVN branch to the Git branch

Convert the SVN branches to Git branches by using the following command.

for b in \$(git for-each-ref--format='%(refname:short)' refs/remotes); do git branch \$b refs/remotes/\$b && git branch -D -r \$b; done

```
E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/Web (mas ter)

$ for b in $(git for-each-ref --format='%(refname:short)' refs/remotes); do git branch $b refs/remotes/$ b && git branch -D -r $b; done Deleted remote-tracking branch origin/1.2.1 (was 076195d).

Deleted remote-tracking branch origin/1.2.144 (was a1ef248).

Deleted remote-tracking branch origin/originalTrunk (was 99402ef).

Deleted remote-tracking branch origin/Release_1_4_1 (was 592ed46).

Deleted remote-tracking branch origin/Release_1_4_2 (was 9d4d3el).

Deleted remote-tracking branch origin/Release_1_6 (was be56b1).

Deleted remote-tracking branch origin/Release_1_65 (was 5b38f3d).

Deleted remote-tracking branch origin/Release_1_65 (was 5b38d4ed).

Deleted remote-tracking branch origin/Release_1_6_8 (was 4503cd7).

Deleted remote-tracking branch origin/Release_1_6_9 (was fa8ebdf).

Deleted remote-tracking branch origin/Release_1_6_9.1 (was 52e0f8c).

Deleted remote-tracking branch origin/TDMCoreAPI (was 7122d30).

Deleted remote-tracking branch origin/TDMCoreAPI (was 7122d30).

Deleted remote-tracking branch origin/TDMTestAPI (was 60054a9).

Deleted remote-tracking branch origin/report_fixes (was 7be665e).

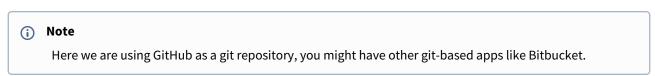
Deleted remote-tracking branch origin/report_fixes (was 3b004a9).

Deleted remote-tracking branch origin/report_fixes (was 3b004a9).

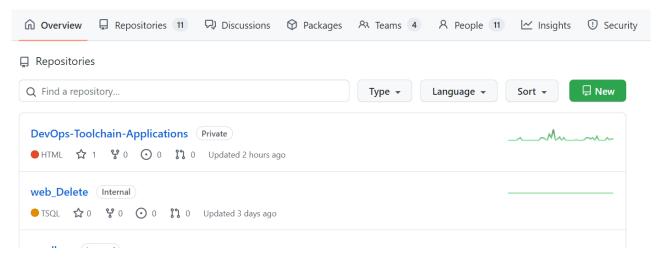
Deleted remote-tracking branch origin/report_fixes (was 3b004a9).
```

Now at this point, your SVN repository is converted to the git repository and ready to push to the remote repo.

5 Step 5: Create an empty git repository, this will be the target repository.



Log in to your GitHub¹account and create a new empty repository by clicking on the "New" button.



Fill in the repository details, the same repo name which is cloned on the local system (recommended)

¹ https://github.com/

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.



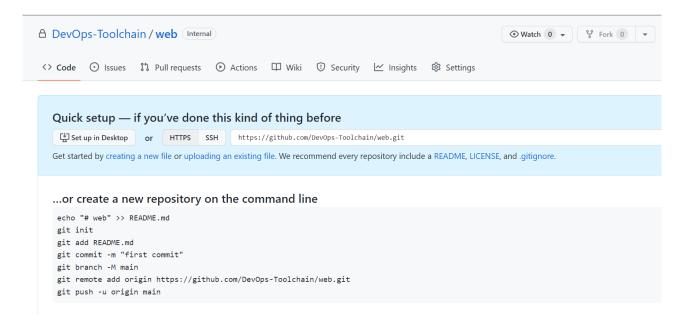
Click on the "Create" button.

Initialize this repository with: Skip this step if you're importing an existing repository. Add a README file This is where you can write a long description for your project. Learn more. Add .gitignore Choose which files not to track from a list of templates. Learn more. .gitignore template: None ▼ Choose a license A license tells others what they can and can't do with your code. Learn more. License: None ▼

(i) You are creating an internal repository in the DevOps-Toolchain organization (Eaton Corporation).

Create repository

The empty git repository will look like the below:



6 Step 6: Push the local converted repository to target the empty repository:

Now commit the git repository, and add all commits in the stage.

```
E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/Web (mas $ git commit -m "svn to git" On branch master nothing to commit, working tree clean E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/Web (mas ter) $ git add --all E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/Web (mas
```

Update the origin in the local git repository to the remote empty repo.

```
E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/Web (master)
$ git remote add origin https://vishalmande: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@github.com/DevOps-Toolchain/web.git

E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/Web (mas
```

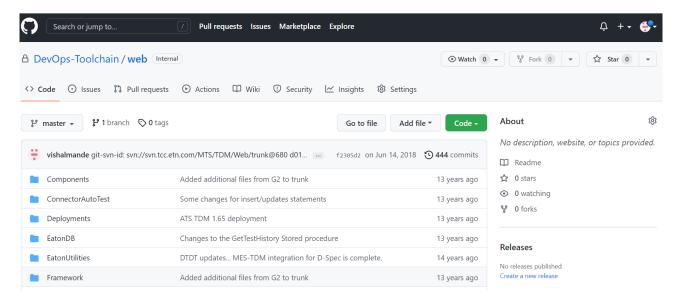
Push the master branch to the remote repository.

```
E0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/web (master)

$ git push -u origin master
Enumerating objects: 5815, done.
Counting objects: 100% (5815/5815), done.
Delta compression using up to 8 threads
Compressing objects: 100% (1654/1654), done.
Writing objects: 100% (5815/5815), 89.55 MiB | 1.24 MiB/s, done.
Total 5815 (delta 4479), reused 5258 (delta 4059)
remote: Resolving deltas: 100% (4479/4479), done.
To https://github.com/DevOps-Toolchain/web.git

* [new branch] master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

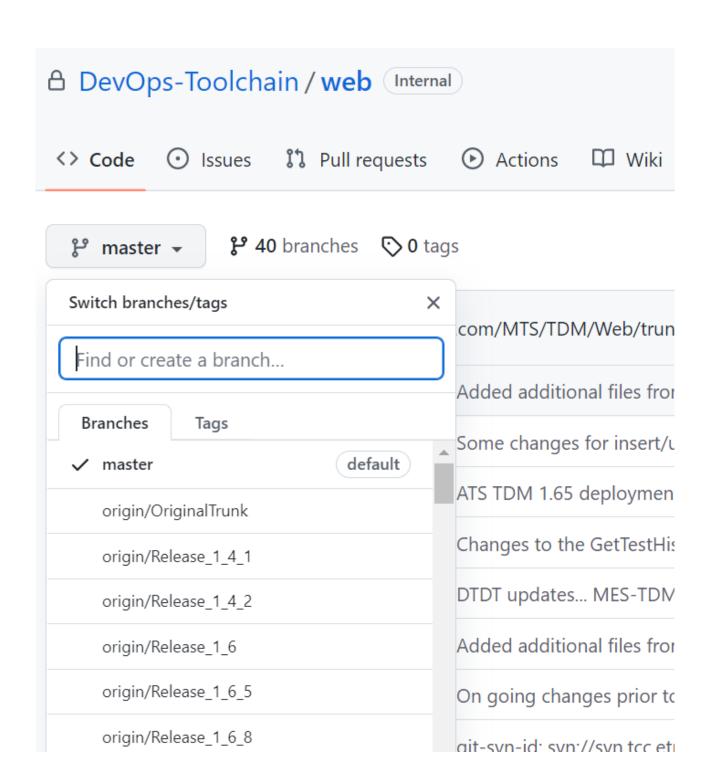
Check your repository data in the remote repository along with the history



Push all other tags/branches to the remote repository.

```
0492097@INPUNWHP6152451 MINGW64 ~/OneDrive - Eaton/Documents/SVN CLone/Test/Web
  (master)
$ git push -u origin --all
Enumerating objects: 5141, done.
Counting objects: 100% (4201/4201), done.
Delta compression using up to 8 threads
Compressing objects: 100% (1575/1575), done.
Writing objects: 100% (3602/3602), 123.35 MiB | 203.08 MiB/s, done.
Total 3602 (delta 2302), reused 3107 (delta 1860)
remote: Resolving deltas: 100% (2302/2302), completed with 259 local objects.
     https://github.com/DevOps-Toolchain/web.git
                                    origin/1.2.1 -> origin/1.2.1
origin/1.2.1@4 -> origin/1.2.1@4
      [new branch]
      new branch
      [new branch]
                                     origin/OriginalTrunk -> origin/OriginalTrunk
                                    origin/Release_1_4_1 -> origin/Release_
origin/Release_1_4_2 -> origin/Release_
      [new branch]
       new branch
                                     origin/Release_1_6 -> origin/Release_1_6
      [new branch]
                                     origin/Release_1_65 -> origin/Release_1_65
      [new branch]
                                    origin/Release_1_6_5 -> origin/Release_1_6_5
origin/Release_1_6_8 -> origin/Release_1_6_8
      [new branch]
      [new branch]
      [new branch]
                                     origin/Release_1_6_9 -> origin/Release_1_6_9
                                    origin/Release_1_6_9_1 -> origin/Release_1_6_9_1 origin/Release_1_6_9_1_Salem2 -> origin/Release_1_6_9_1_Sal
      [new branch<sup>-</sup>
      [new branch]
em2
                                    origin/TDMCoreAPI -> origin/TDMCoreAPI
origin/TDMTestAPI -> origin/TDMTestAPI
origin/report_fixes -> origin/report_fixes
origin/report_fixes@26 -> origin/report_fixes@26
origin/tags/1-2-b -> origin/tags/1-2-b@27
origin/tags/1-2-b@27 -> origin/tags/1-2-c
      [new branch]
       new branch
      [new branch]
      [new branch]
             branch]
      [new
      [new branch
                                    origin/tags/1-2-c -> origin/tags/1-2-c origin/tags/1-2-c@34 -> origin/tags/1-2-c@34 origin/tags/1-2-d
      [new branch]
       new branch
      [new branch]
```

Check all tags and branches data in the remote repository along with historical data and commits.



You have done a good job! Now from here, you can make the svn repository read-only and eventually stop using it and start using git. You have my best wishes for your git journey from here. $\ensuremath{\mathfrak{C}}$