

Using a GitHub *git* Repository as a Read-Only Mirror for an *svn* Repository

David T. Ashley (dashley@gmail.com)

Introduction and Overview

I (Dave Ashley) use *svn* (Subversion) at home as a version control system. However, *git* and *GitHub* are very popular, and I wish to keep a copy of my public projects under *git* on *GitHub*. This requires an export from *svn* to *git*, as well as interaction with *GitHub*.

This document describes the procedures I use to initially export a complete *svn* repository to *git* and *GitHub*, and to incrementally export subsequent *svn* commits to *git* and *GitHub*.

Git vs. *TortoiseGit* *git* Executables

At the time I first tried to fully export an *svn* repository to a *git* repository, I had both *Git* and *TortoiseGit* installed.

I discovered that the *git* binary included with *Git* seemed to handle the *svn* export incorrectly, eventually terminating with the error “*git-svn migration fatal: not a valid SHA1 ...*”.

The *git* binary included with *TortoiseGit* seemed to handle the *svn* export correctly.

To use the *TortoiseGit* *git* binary from the *Bash* shell program distributed with *Git*, a longer path must be used. That is why “*C:/Program Files/Git/bin/git.exe*” rather than “*git*” is used in some of the instructions that follow. Using “*git*” alone would use the *git* executable distributed with *Git*, which does not do the export from *svn* correctly.

However, other than the export from *svn* to *git*, the *Git* *git* executable works correctly.

Notes on My Home Setup

I use *svn* hidden behind *SSH* port forwarding. For that reason, in the following procedures, the host for my *svn* repository is *localhost*. A typical *svn* URL in my home setup is "*svn://localhost/dtapublic*".

Initial *svn* to *git* Export

1. Create the file used to transform an *svn* author name to a *git* author name (if necessary). My file has this content and is named *authors-transform.txt*:

```
dashley = David T. Ashley <dashley@gmail.com>
```

2. Perform the initial export:

```
$ "C:/Program Files/Git/bin/git.exe" svn clone svn://localhost/dtapublic -A authors-transform.txt
```

The transformation will proceed at the rate of approximately 20 *svn* revisions per minute. A subdirectory named *dtapublic* will be created to contain the *git* repository and sandbox. As the last phase, a *git* checkout will be performed.


3. Create a *git* repository on *GitHub*.

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

Repository name

 dtashley ▾


 /

dtapublic ✓


Great repository names are short and memorable. Need inspiration? How about **refactored-spoon**.

Description (optional)

David T. Ashley's Public Projects

☒  **Public**

Anyone can see this repository. You choose who can commit.

☐  **Private**


You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾



 |

Add a license: **None** ▾ 

Create repository

GitHub also provides instructions about what can be done with the repository.

Quick setup — if you've done this kind of thing before

 Set up in Desktop or **HTTPS** **SSH** 

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# dtapublic" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/dtashley/dtapublic.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/dtashley/dtapublic.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

4. `cd` into the new *git* repository (the subsequent *git* commands must be done from within the repository).

```
$ cd dtapublic/
```

5. Execute the *git remote add* command below.

```
$ git remote add origin https://github.com/dtashley/dtapublic.git
```

6. Execute the *git push* command below. This pushes the local repository to GitHub.

```
$ git push -u origin master
Enumerating objects: 5953, done.
Counting objects: 100% (5953/5953), done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4931/4931), done.
Writing objects: 100% (5953/5953), 63.54 MiB | 1.12 MiB/s, done.
Total 5953 (delta 2765), reused 0 (delta 0)
remote: Resolving deltas: 100% (2765/2765), done.
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/dtashley/dtapublic/pull/new/master
remote:
To https://github.com/dtashley/dtapublic.git
```

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

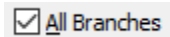
Incremental *svn* to *git* Export

1. Run the git fetch command below.

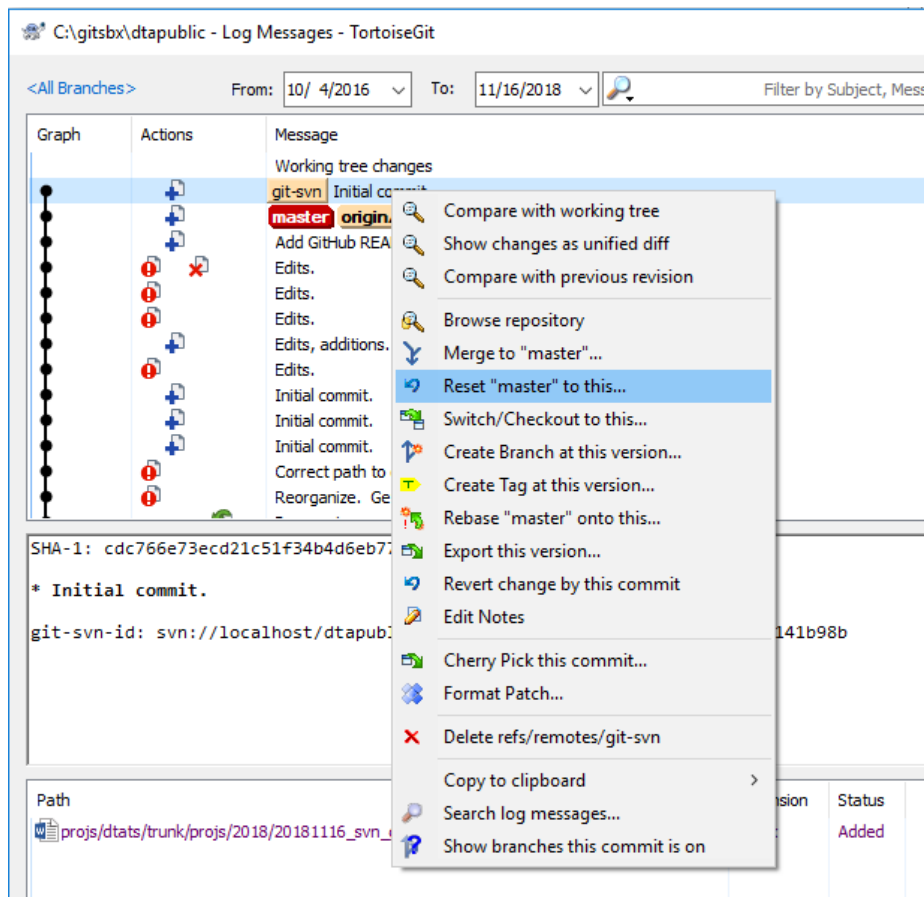
```
$ "C:/Program Files/Git/bin/git.exe" svn fetch -A
"C:/svnsbx/dtapublic/projs/dtats/trunk/projs/2018/20181116_svn_git_mirror/authors-transform.txt"

A      proj s/dtats/trunk/proj s/2018/20181116_svn_git_mirror/authors-transform.txt
r255 = 98be7c63b17c13af3ac838588f1fee9281172560 (refs/remotes/git-svn)
```

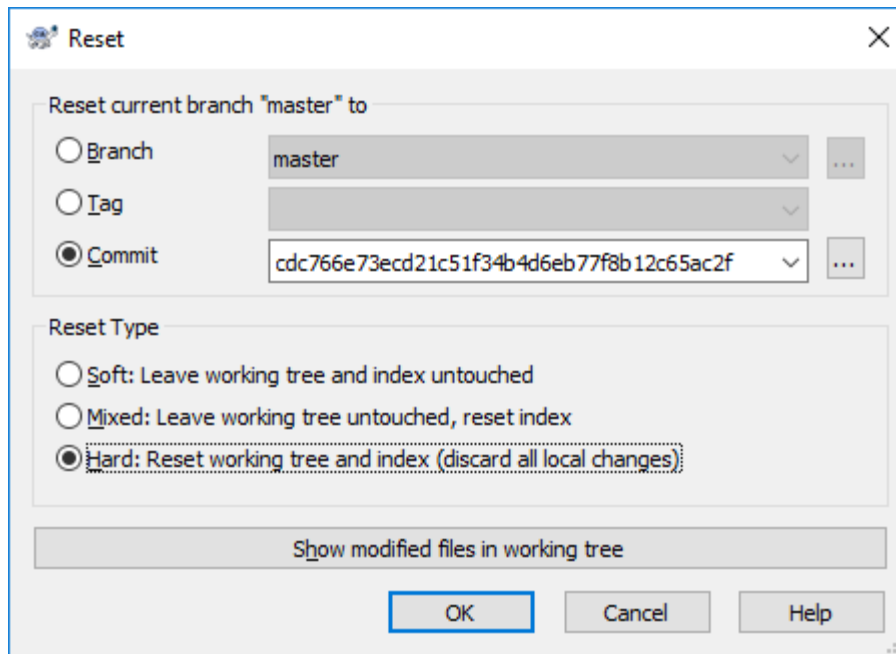
2. Right-click on the local *git* repository and choose *TortoiseGit* -> *View Log*. Ensure that “*All Branches*” is checked.



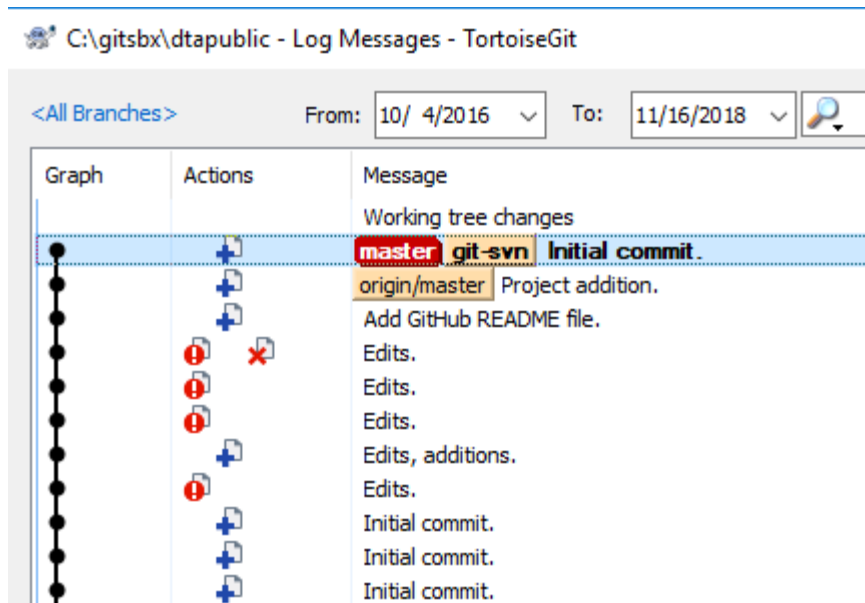
3. The most recent commits should be shown at the top of the log. Right click on the topmost commit and choose *Reset “master” to this....*



Choose "hard" reset to keep the *git* sandbox synchronized with the repository.



At this point, *origin/master* will be lagging *master*.



4. Issue the *git push* command shown below. This will cause *origin/master* to become coincident with *master*.

```
dashley@DTA-T420-B MINGW64 /c/gitsbx/dtapublic (master)
$ git push
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 4 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (9/9), 827 bytes | 413.00 KiB/s, done.
Total 9 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/dashley/dtapublic.git
  9425e85..98be7c6 master -> master
```


C:\gitsbx\dtapublic - Log Messages - TortoiseGit

<All Branches> From: 10/ 4/2016 To: 11/16/2018

Graph	Actions	Message
		Working tree changes
	+	master origin/master git-svn Initial commit.
	+	Project addition.
	+	Add GitHub README file.
	!	Edits.
	!	Edits.
	!	Edits.
	+	Edits, additions.
	!	Edits.
	+	Initial commit.
	+	Initial commit.
	+	Initial commit.