

Hands-on Git to Set Up Three Branches

Kate Hedstrom
ARSC, UAF

Work Along?

```
git config --global user.name "me"
```

```
git config --global user.email "me@work"
```

```
git config --global color.ui "auto"
```

- **This information goes into \$HOME/.gitconfig**
- **Find some code to track...**

Download ROMS

```
git svn clone -r 526 --username name  
https://www.myroms.org/svn/src/trunk MyDir
```

- **DO NOT leave out the “-r 526”**
 - If you do, you will download the entire history
 - Downloading the entire history will be very slow because svn is slow (compared to git)

See What You Have

- **Change into your new directory**
- **“ls” to see the ROMS files**
- **“ls -a” shows hidden files – in this case the .git directory**
- **“more .git/config” to see git’s view of the repo**

Ignoring Files

- **Edit .gitignore**

- **I put in stuff like ocean*, Build, etc**

```
git add .gitignore
```

```
git diff
```

```
git commit
```

```
gitk
```

Branches

- **The goal at the end of this is to have three branches:**
 - master – clean copy of the Rutgers trunk
 - Circle – some special files for only the CIRCLE test case
 - my_stuff – for my personal changes (includes Tide_bay files)

Circle Fixes

- **Go to and download: wget**
<http://www.arsc.edu/~kate/ROMS/HK/circle.diffs>
- **Best to keep the master branch clean so we create a new branch:**

```
git branch my_stuff
```

```
git checkout my_stuff
```

```
patch -p1 < circle.diffs
```

```
git diff
```

Check in the Tide_bay Fixes

```
git status
```

```
git add Apps/Tide_bay
```

```
git commit -m 'Tide Bay setup'
```

A Branch for the Circle Problem

```
git branch circle
```

```
git checkout circle
```

```
git add Apps/Circle makefile.circle
```

```
git status
```

```
git commit -a -m "Circle test case"
```

```
git branch
```

Create a Conflict

- **Edit ROMS/Version to change the date in your branch**

```
git checkout my_stuff
```

```
[edit ROMS/Version]
```

```
git status
```

```
git commit -a
```

```
git status
```

- **Fetch the latest code into master:**

```
git checkout master
```

```
git svn rebase
```

Now for that Conflict

```
git checkout my_stuff
```

```
git merge master
```

```
git status
```

View the Conflict

- **In the file:**

```
<<<<<<< HEAD
```

```
ROMS/TOMS Framework:  December 30, 2010
```

```
=====
```

```
ROMS/TOMS Framework:  January 1, 2010
```

```
>>>>>>> Playing around.
```

```
=====
```

```
Copyright (c) 2002-2010 The ROMS/TOMS Group
```

Fix the Conflict

- **Clean up the code using Hernan's date**
- **Tell git the problem has been solved:**

```
git add ROMS/Version
```

```
git commit
```

Rest of Update

- **Have to update the circle branch too:**

```
git checkout circle
```

```
git merge master
```

```
git branch
```

Last Thing...

- **You always want to check “git status” and make sure it’s clean before fetching an update**
- **Ditto for switching branches unless you want the change to be in the new branch instead**

Remote Example

```
git remote add cygnus ssh://kate@cygnus.arsc.edu/~/  
feedme.git/
```

```
git remote update
```

```
git branch -a
```

```
git checkout remotes/cygnus/kate_svn
```

```
git checkout -b kate_svn
```

```
git branch
```

- **Here, cygnus is the name of the remote site in our .git/config file – a handle**
- **Kate_svn is a branch in that repo**