Git

Tutorials

CI/CD with GitHub and Travis

Andrea Fornaia, Ph.D.

Department of Mathematics and Computer Science
University of Catania
Viale A. Doria, 6 - 95125 Catania Italy
fornaia@dmi.unict.it
http://www.cs.unict.it/~fornaia/
<table>
<thead>
<tr>
<th>Titolo</th>
<th>Paese</th>
<th>Anno</th>
<th>Stagioni</th>
<th>Episodi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breaking Bad</td>
<td>Stati Uniti</td>
<td>2008-2013</td>
<td>5</td>
<td>62</td>
</tr>
<tr>
<td>Altered Carbon</td>
<td>Stati Uniti</td>
<td>2018</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>Modern Family</td>
<td>Stati Uniti</td>
<td>2009-2020</td>
<td>11</td>
<td>250</td>
</tr>
</tbody>
</table>
Tutorial 1: Local Operations
Init Local Repo

- Init new git repository

```bash
$ mkdir tutorial1
$ cd tutorial1
$ git init
```

- Perform basic configuration

```bash
$ git config --global user.name "Your Name"
$ git config --global user.email "your.name@email.com"
$ git config --global alias.graph "log --graph --decorate --oneline"
```

- Create a simple index.html and add it to a commit on master

```bash
$ echo "Hello from Index" > index.html
$ git add .
$ git commit -m "Simple index.html"
$ git graph
* 8d6e49d (HEAD, master) Simple index.html
```

- Open index.html on browser

```html
Hello from index
```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="css/styles.css">
  <title>NetSix.com</title>
</head>
<body>
  <div class="header">
    <h1>NetSix</h1>
  </div>
  <div class="container">
    <div class="item">
      <img src="img/Breaking_Bad.jpg" alt="Breaking Bad"/>
      <ul>
        <li><b>Titolo:</b> Breaking Bad</li>
        <li><b>Paese:</b> Stati Uniti</li>
        <li><b>Anno:</b> 2008-2013</li>
        <li><b>Stagioni:</b> 5</li>
        <li><b>Episodi:</b> 62</li>
      </ul>
    </div>
  </div>
  <div class="footer">
    <p>NetSix.com</p>
  </div>
</body>
</html>
Add css/styles.css

body {
    background: white;
    font-family: "Arial";
}

.header, .footer {
    background-color: blue;
    color: white;
    padding: 15px;
    text-align: center;
}

.header {
    margin-bottom: 8px;
    font-size: 16px;
}

.footer {
    margin-top: 8px;
    clear: both;
}

.container {
    margin: auto;
    display: table;
}

.item {
    margin: 10px;
    width: 300px;
    float: left;
}

.item img {
    width: 300px;
    height: 175px;
}

.item ul {
    list-style-type: none;
    margin: 0px;
    padding: 0px;
}
Add img/Breaking_Bad.jpg
Open index.html

Titolo: Breaking Bad
Paese: Stati Uniti
Anno: 2008-2013
Stagioni: 5
Episodi: 62
Commit everything on master

- Init new git repository

```bash
$ git add . (i.e. index.html, css/, img/)
$ git commit -m "First NetSix version"
$ git graph
* 6dcadda (HEAD, master) First NetSix version
* 8d6e49d Simple index.html
```

- We assume that everything on master branch reflects the code on production
Start working on a new feature

- Create and switch to a feature branch to add “Altered Carbon” details and image

```bash
$ git branch feature/altered-carbon
$ git checkout feature/altered-carbon
$ git graph
* 6dcadda (HEAD, master, feature/altered-carbon) First NetSix version
* 8d6e49d Simple index.html
```

- Edit index to add Altered Carbon details (add a new `div.item` inside `div.container`)

```html
<div class="item">
  <img src="img/Altered_Carbon.jpg" alt="Altered Carbon"/>
  <ul>
    <li><b>Titolo:</b> Altered Carbon</li>
    <li><b>Paese:</b> Stati Uniti</li>
    <li><b>Anno:</b> 2018</li>
    <li><b>Stagioni:</b> 1</li>
    <li><b>Episodi:</b> 10</li>
  </ul>
</div>
```
Commit on feature branch

```bash
$ git add index.html
$ git commit -m "Added Altered Carbon details"
$ git graph
* dfb8ac6 (HEAD, feature/altered-carbon) Added Altered Carbon details
* 6dcadda (master) First NetSix version
* 8d6e49d Simple index.html
```
Hotfix on master required!

- We need to fix the theme on production (master), from “blue & white” to “black & red”, **NOW!**

```bash
$ git checkout master
$ git checkout -b hotfix/black-red-theme
$ git graph --all (to see all branches, not only the active one)
* dfb8ac6 (feature/altered-carbon) Added Altered Carbon details
* 6dcadda (HEAD, master, hotfix/black-red-theme) First NetSix version
* 8d6e49d Simple index.html
```

- Fix the css style

```css
body {
- background: white;
+ background: black;
}

.header, .footer {
- background-color: blue;
- color: white;
+ background-color: #222;
+ color: red;
}

.item li {
- background-color: blue;
+ background-color: red;
}

.item li:hover {
- background-color: green;
+ background-color: gray
}
```
Commit on hotfix branch

```
$ git add css/style.css
$ git commit -m "Fixed style"
$ git graph
* 6e5304b (HEAD, hotfix/black-red-theme) Fixed style
  * dfb8ac6 (feature/altered-carbon) Added Altered Carbon details
  /
* 6dcadda (master) First NetSix version
* 8d6e49d Simple index.html
```

There are no Altered Carbon details, which are still on a separate feature branch!
Merge hotfix branch into master

$ git checkout master
$ git merge hotfix/black-red-theme (will be a fast-forward merge)
$ git graph --all
* 6e5304b (HEAD, master, hotfix/black-red-theme) Fixed style
  | * dfb8ac6 (feature/altered-carbon) Added Altered Carbon details
  |
* 6dcadda First NetSix version
* 8d6e49d Simple index.html

- Master (i.e. production code) has been merged with fixes
- We can delete the hotfix branch

$ git branch -d hotfix/black-red-theme
$ git graph --all
* 6e5304b (HEAD, master) Fixed style
  | * dfb8ac6 (feature/altered-carbon) Added Altered Carbon details
  |
* 6dcadda First NetSix version
* 8d6e49d Simple index.html
Let’s get back to the feature

$ git checkout feature/altered-carbon
$ Add Altered_Carbon.jpg to img/
$ git add img/Altered_Carbon.jpg
$ git commit -m "Added Altered Carbon image"
$ git graph --all
* c314b0a (HEAD, feature/altered-carbon) Added Altered Carbon image
* dfb8ac6 Added Altered Carbon details
| * 6e5304b (master) Fixed style
|/
* 6dcadda First NetSix version
* 8d6e49d Simple index.html

NetSix

The updated style is still on the master branch

<table>
<thead>
<tr>
<th>Titolo: Breaking Bad</th>
<th>Titolo: Altered Carbon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paese: Stati Uniti</td>
<td>Paese: Stati Uniti</td>
</tr>
<tr>
<td>Stagioni: 5</td>
<td>Stagioni: 1</td>
</tr>
<tr>
<td>Episodi: 62</td>
<td>Episodi: 10</td>
</tr>
</tbody>
</table>

NetSix.com
Merge the completed feature

$ git checkout master
$ git merge feature/altered-carbon
$ git branch -d feature/altered-carbon
$ git graph
* 3ccb7b5 (HEAD, master) Merge branch 'feature/altered-carbon'
|\ 3ccb7b5 (HEAD, master) Merge branch 'feature/altered-carbon'
| * c314b0a Added Altered Carbon image
| * dfb8ac6 Added Altered Carbon details
| * 6e5304b Fixed style
|/
* 6dcadda First NetSix version
* 8d6e49d Simple index.html
Tutorial 2: Remote Operations
Create a new repo on GitHub

For this use case, we do not init the repo with the README file
Push local repo

- We want to init the remote repository with the local repository state
- Preserving the previous commit history (tutorial1)

$ cp -r tutorial1 tutorial2 (includes the .git)
$ git remote -v (no remote already set)
$ git remote add origin https://github.com/yourname/git-tutorial.git
$ git remote -v
origin https://github.com/yourname/git-tutorial.git (fetch)
origin https://github.com/yourname/git-tutorial.git (push)
$ git push -u origin master (set upstream master -> origin/master)

- This will sync the remote and add the remote branch references

$ git graph
* 3ccb7b5 (HEAD, origin/master, master) Merge branch 'feature/altered-carbon'
|`
| * c314b0a Added Altered Carbon image
| * dfb8ac6 Added Altered Carbon details
* | 6e5304b Fixed style
|/
* 6dcadda First NetSix version
* 8d6e49d Simple index.html
GitHub repo is in sync
Mimic a concurrent change

- We want to mimic a second developer with write access to the repo
- We clone the remote repository in another directory (tutorial2_mimic)

```bash
$ cd ../ (the directory containing tutorial2)
$ mkdir tutorial2_mimic
$ cd tutorial2_mimic
$ git clone https://github.com/yourname/git-tutorial.git .
$ git graph
  * 3ccb7b5 (HEAD, origin/master, master) Merge branch 'feature/altered-carbon'
  |
  | * c314b0a Added Altered Carbon image
  | * dfb8ac6 Added Altered Carbon details
  |   * 6e5304b Fixed style
  |
  * 6dcadda First NetSix version
  * 8d6e49d Simple index.html
```
First developer add a new feature

$ cd ../tutorial2
$ git checkout -b feature/modern-family
$ Edit index.html with Modern Family details
$ git commit -a -m "Added Modern Family details"
$ Add img/Modern_Family.jpg
$ git add .
$ git commit -m "Added Modern Family image"
$ git checkout master
$ git merge feature/modern-family
$ git branch -d feature/modern-family
First developer push the feature

$ git graph
* 1a4bf38 (HEAD, master) Added Modern Family image
* a456105 Added Modern Family description
* 3ccb7b5 (origin/master) Merge branch 'feature/altered-carbon'
  /
  | * c314b0a Added Altered Carbon image
  | * dfb8ac6 Added Altered Carbon details
  | | 6e5304b Fixed style
  |/
* 6dcadda (tag: v1.0) First NetSix version
* 8d6e49d Simple index.html

$ git push origin master

$ git graph
* 1a4bf38 (HEAD, origin/master, master) Added Modern Family image
* a456105 Added Modern Family description
* 3ccb7b5 Merge branch 'feature/altered-carbon'
  /
  | * c314b0a Added Altered Carbon image
  | * dfb8ac6 Added Altered Carbon details
  | | 6e5304b Fixed style
  |/
* 6dcadda (tag: v1.0) First NetSix version
* 8d6e49d Simple index.html
GitHub repo is in sync
Second developer add a new feature

$ cd ../tutorial2_mimimc
$ git checkout -b feature/vikings
$ Edit index.html with Vikings details
$ git commit -a -m "Added Vikings details"
$ Add img/Vikings.jpg
$ git add .
$ git commit -m "Added Vikings image"
$ git checkout master
$ git merge feature/vikings
$ git branch -d feature/vikings
Second developer (try to) push the feature

$ git graph
* 888ce21 (HEAD, master) Added Vikings image
* f682744 Added Vikings details
* 3ccb7b5 (origin/master) Merge branch 'feature/altered-carbon'
  /
  | * c314b0a Added Altered Carbon image
  | * dfb8ac6 Added Altered Carbon details
  | 6e5304b Fixed style
  / 6dcadda First NetSix version
* 8d6e49d Simple index.html

$ git push origin master (WILL FAIL! We need to integrate remote changes first)
Fetch and merge remote changes

$ git fetch
$ git graph --all
* 888ce21 (HEAD, master) Added Vikings image
* f682744 Added Vikings details
| * 1a4bf38 (origin/master) Added Modern Family image
| * a456105 Added Modern Family description
|/
* 3ccb7b5 Merge branch 'feature/altered-carbon'
|\n| * c314b0a Added Altered Carbon image
| * dfb8ac6 Added Altered Carbon details
* | 6e5304b Fixed style
|/
* 6dcadda First NetSix version
* 8d6e49d Simple index.html

$ git merge origin/master (solve conflicts, i.e. simply choose the item order)
$ git commit -a (complete the merge commit)
Push changes to remote

$ git graph --all
* 1ae52c2 (HEAD, master) Merge remote-tracking branch 'origin/master'
  \\ * 1a4bf38 (origin/master) Added Modern Family image
  * a456105 Added Modern Family description
  * 888ce21 Added Vikings image
  * f682744 Added Vikings details
  \\ * 3ccbb5 Merge branch 'feature/altered-carbon'
  \\ * c314b0a Added Altered Carbon image
  * dfb8ac6 Added Altered Carbon details
  * 6e5304b Fixed style
  \\ * 6dcadda First NetSix version
* 8d6e49d Simple index.html

$ git push origin master (master and origin/master will point the same commit)
GitHub repo is in sync
Also update first developer copy

$ cd ../tutorial2
$ git pull origin master
$ git graph --all
  * 1ae52c2 (HEAD, origin/master, master) Merge remote-tracking branch 'origin/master'
     |
     | * 1a4bf38 Added Modern Family image
     | * a456105 Added Modern Family description
     | | 888ce21 Added Vikings image
     | | f682744 Added Vikings details
     |/
     * 3ccb7b5 Merge branch 'feature/altered-carbon'
     |
     | * c314b0a Added Altered Carbon image
     | * dfb8ac6 Added Altered Carbon details
     | | 6e5304b Fixed style
     |/
     * 6dcadda First NetSix version
* 8d6e49d Simple index.html
Tutorial 3: Travis Pipeline
Link Travis to GitHub

Test and Deploy with Confidence
Easily sync your GitHub projects with Travis CI and you'll be testing your code in minutes!

1. Test and Deploy with Confidence
2. Authorize Travis CI for Open Source
3. Webhooks
Link Travis to a public repository

Danger Zone

Make this repository public
Make this repository visible to anyone.

Make public

Travis CI

My Account

afornaia

Sync account

A SINGLE PLACE FOR ALL YOUR BUILDS
You can now have all your public and private repositories together at travis-ci.com

Sign up for the beta

Repositories

Settings

Legacy Services Integration

Filter repositories

git-tutorial

GitHub Project

Settings
Generate GITHUB_TOKEN

GitHub: User > Settings > Developer settings > Generate new token

Travis CI: More options > Settings > Environment Variables

Copy and paste the generated token to Travis envs as GITHUB_TOKEN
Deploy automation to GitHub Pages

$ cd ../
$ mkdir tutorial3
$ git clone https://github.com/yourname/git-tutorial.git .
$ vim .travis.yml

```yaml
language: generic
script:
  - echo "Step 1"
  - echo "Step 2"
deploy:
  provider: pages
  github-token: $GITHUB_TOKEN
on:
  branch: master
```

$ git add .travis.yml
$ git commit -m "Added .travis.yml"
$ git push origin master
Check pipeline status on Travis
Check pipeline logs on Travis

```bash
$ git clone --depth=50 --branch=master https://github.com/afornaia/git-tutorial.git

$ export GITHUB_TOKEN=[secure]

$ bash -c 'echo $BASH_VERSION'

4.3.46(1)-release

$ echo "Step 1"

Step 1

The command "echo "Step 1"" exited with 0.

$ echo "Step 2"

Step 2

The command "echo "Step 2"" exited with 0.

$ rvm $(travis_internal_ruby) --fuzzy do ruby -S gem install dpl

Installing deploy dependencies

Logged in as @afornaia ()

Preparing deploy

Deploying application

No stash entries found.

Done. Your build exited with 0.
```
Automatic push on gh-pages
GH Pages activated by travis (settings)
Check the project page

yourname.github.io/git-tutorial
Add some tests: make it fail!

$ vim .travis.yml

language: generic
script:
  - test $(grep '<div class="item">' index.html | wc -l) -le 6
  - test $(grep '<div class="item">' index.html | wc -l) -ge 5

deploy:
  provider: pages
  github-token: $GITHUB_TOKEN
  on:
    branch: master

$ git add .travis.yml
$ git commit -m "Added test execution to travis pipeline"
$ git push origin master
Check pipeline status

```
$ git clone --depth=50 --branch=master https://github.com/afornaia/git-tutorial.git

Setting environment variables from repository settings
$ export GITHUB_TOKEN=[secure]

$ bash -c 'echo $BASH_VERSION'
4.3.48(1)-release

$ test $(grep '<div class="item">' index.html | wc -l) -le 6
The command "test $(grep '<div class="item">' index.html | wc -l) -le 6" exited with 0.

$ test $(grep '<div class="item">' index.html | wc -l) -ge 5
The command "test $(grep '<div class="item">' index.html | wc -l) -ge 5" exited with 1.

Done. Your build exited with 1.
```
Tutorial 4: Pull Request
Push a feature branch

$ git checkout -b feature/hercules
$ make a little change to index.html, like a new line
  just to show how to handle a pull request with errors
$ git commit -a -m "Little change to the index.html"
$ git push -u origin feature/hercules

remote:
remote: Create a pull request for 'feature/hercules' on GitHub by visiting:
remote:   https://github.com/afornaia/git-tutorial/pull/new/feature/hercules
remote:
To https://afornaia@github.com/afornaia/git-tutorial.git
 * [new branch] feature/hercules -> feature/hercules
Branch feature/hercules set up to track remote branch feature/hercules from origin.

- Master is usually a protected branch!
- Only the repo maintainer should be able to push on master
- We want to create a pull request, asking to merge our changes to master branch
- Feature branch (like any other branch) will trigger travis tests
- If the feature branch passes the tests, the maintainer may choose to merge
- In case of a merge, this will cause a push on master branch
- Triggerring travis deploy on production (GH Pages)
- Pull request can now be closed!
Create a pull request

https://github.com/yourname/git-tutorial/pull/new/feature/hercules
Create a pull request

https://github.com/yourname/git-tutorial/pull/new/feature/hercules
Tests will fail

We can help the developer with code reviews, giving suggestions to complete the pull request.

The developer may push other commits to this feature branch (follow-up commits).

The pull request will be automatically updated with the new commit!
Add follow-up commits

$ Edit index.html with Hercules details
$ git commit -a -m "Added Hercules details"
$ Add img/Hercules.jpg
$ git add .
$ git commit -m "Added Hercules image"
$ git push origin feature/hercules

The maintainer can merge the pull request directly from GitHub (if it's a fast-forward)
Merge and close the pull request
Automatically deployed on production

<table>
<thead>
<tr>
<th>Branch: master</th>
<th>New pull request</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>afornaia</strong> Merge pull request #1 from afornaia/feature/hercules</td>
<td>Latest commit d62e338 7 minutes ago</td>
</tr>
<tr>
<td>css</td>
<td>Fixed style</td>
</tr>
<tr>
<td>img</td>
<td>Added Hercules image</td>
</tr>
<tr>
<td>.travis.yml</td>
<td>Added test execution to travis pipeline</td>
</tr>
<tr>
<td>index.html</td>
<td>Added Hercules details</td>
</tr>
</tbody>
</table>

Help people interested in this repository understand your project by adding a README.
That’s All!

(for now)