Distributed version control with git — a brief introduction

Oscar Nierstrasz
Why git?
A recipe for disaster!
The git object model
A “blob” is *content* under version control (a file)
You can have *trees* of blobs (directories of files)
A “commit” is a tree of blobs
(a set of changes)
Most commits modify (or merge) earlier commits.
You can “tag” an interesting commit
A graph of commits may belong to a branch.
"HEAD "is the current branch
Let’s focus on *commits* and *branches*.
Basic git
Create a git repo

mkdir repo
cd repo
git init
Tell git to “stage” changes

git add ...

- HEAD
  - master
  - C0
Commit your changes

git commit ...
Collaborating
John

Local repo

```
git clone ...
```

Public repo

```
git clone ...
```

Jane

Local repo

```
C1
C0
```

```
C1
C0
```

```
C1
C0
```
John

Local repo

```
<table>
<thead>
<tr>
<th>master</th>
</tr>
</thead>
<tbody>
<tr>
<td>C2</td>
</tr>
<tr>
<td>C1</td>
</tr>
<tr>
<td>C0</td>
</tr>
</tbody>
</table>
```

```
git pull
```

Public repo

```
<table>
<thead>
<tr>
<th>master</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
</tr>
<tr>
<td>C0</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>master</th>
</tr>
</thead>
<tbody>
<tr>
<td>C3</td>
</tr>
<tr>
<td>C1</td>
</tr>
<tr>
<td>C0</td>
</tr>
</tbody>
</table>
```

Local repo

```
<table>
<thead>
<tr>
<th>master</th>
</tr>
</thead>
<tbody>
<tr>
<td>C2</td>
</tr>
<tr>
<td>C1</td>
</tr>
<tr>
<td>C0</td>
</tr>
</tbody>
</table>
```

(noting new to pull)
Public repo

Local repo

Local repo

**git push**

John

Jane

C0

C1

C2

master

C0

C1

C2

C0

master

C0

C1

C2

C3

C1

C0

Thursday, September 15, 11
NB: git pull = fetch + merge
Branching and merging
“origin” refers to the remote repo

HEAD

master

origin/master

C1

C0
...git commit ...
git checkout tryout

NB: `git checkout -b ...` = branch + checkout

- C0
- C1
- C2
- master
- origin/master
- tryout
- HEAD

Thursday, September 15, 11
git fetch origin

C1
master

C2
C3
tryout

C4
C5
origin/master

C6
origin/idea

git fetch origin

Thursday, September 15, 11
git merge origin/master origin/idea
git push

HEAD
master
origin/master

C0
C1
C2
C3
C4
C5
C6
C7

tryout

origin/idea

Thursday, September 15, 11
More to git
More to git …

> Merging and mergetool
> Squashing commits when merging
> Resolving conflicts
> User authentication with ssh
> gitx and other graphical tools
> git configure — remembering your name
> git remote — multiple remote repos
> github — an open source public repo
> …
Resources

http://git-scm.com/

https://github.com/

http://www.slideshare.net/chacon/getting-git

http://oreilly.com/
You are free:
▪ to copy, distribute, display, and perform the work
▪ to make derivative works
▪ to make commercial use of the work

Under the following conditions:

**Attribution.** You must attribute the work in the manner specified by the author or licensor.

**Share Alike.** If you alter, transform, or build upon this work, you may distribute the resulting work only under a license identical to this one.

▪ For any reuse or distribution, you must make clear to others the license terms of this work.
▪ Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

[http://creativecommons.org/licenses/by-sa/3.0/](http://creativecommons.org/licenses/by-sa/3.0/)