

**COMP 311 / 311H**

**Software Engineering**

*SCM & Git/Github*

**Sung Kim**

# Software Configuration Management (SCM)

- Version control systems
- Issue (bug) tracking systems
- Build management systems

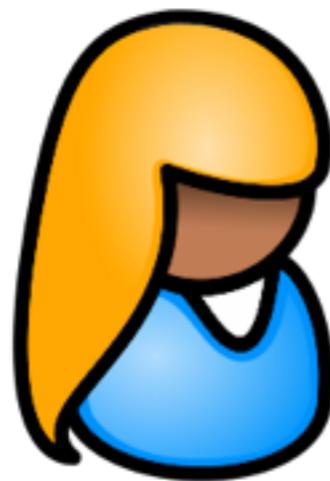
# Why version control system?

- Disk is cheap so I made many copies
  - fold\_2015\_9\_1
  - fold\_2015\_9\_3
  - ...

# Why version control system?

- Disk is cheap so I made many copies
  - fold\_2015\_9\_1
  - fold\_2015\_9\_3
  - ...
- How to revert (partially)?
- How can I work with others (simultaneously)?

# Software development



Build



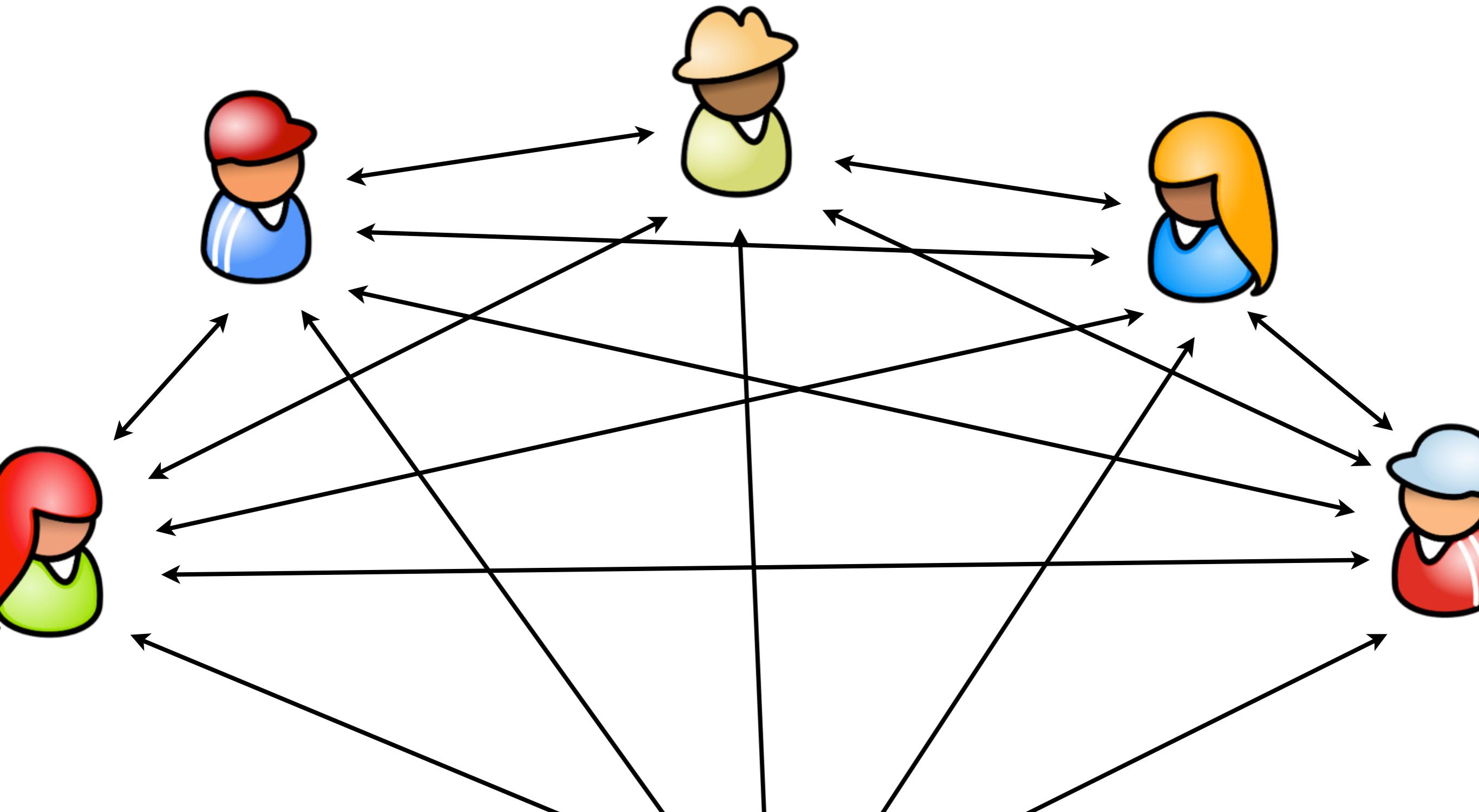
# Software development



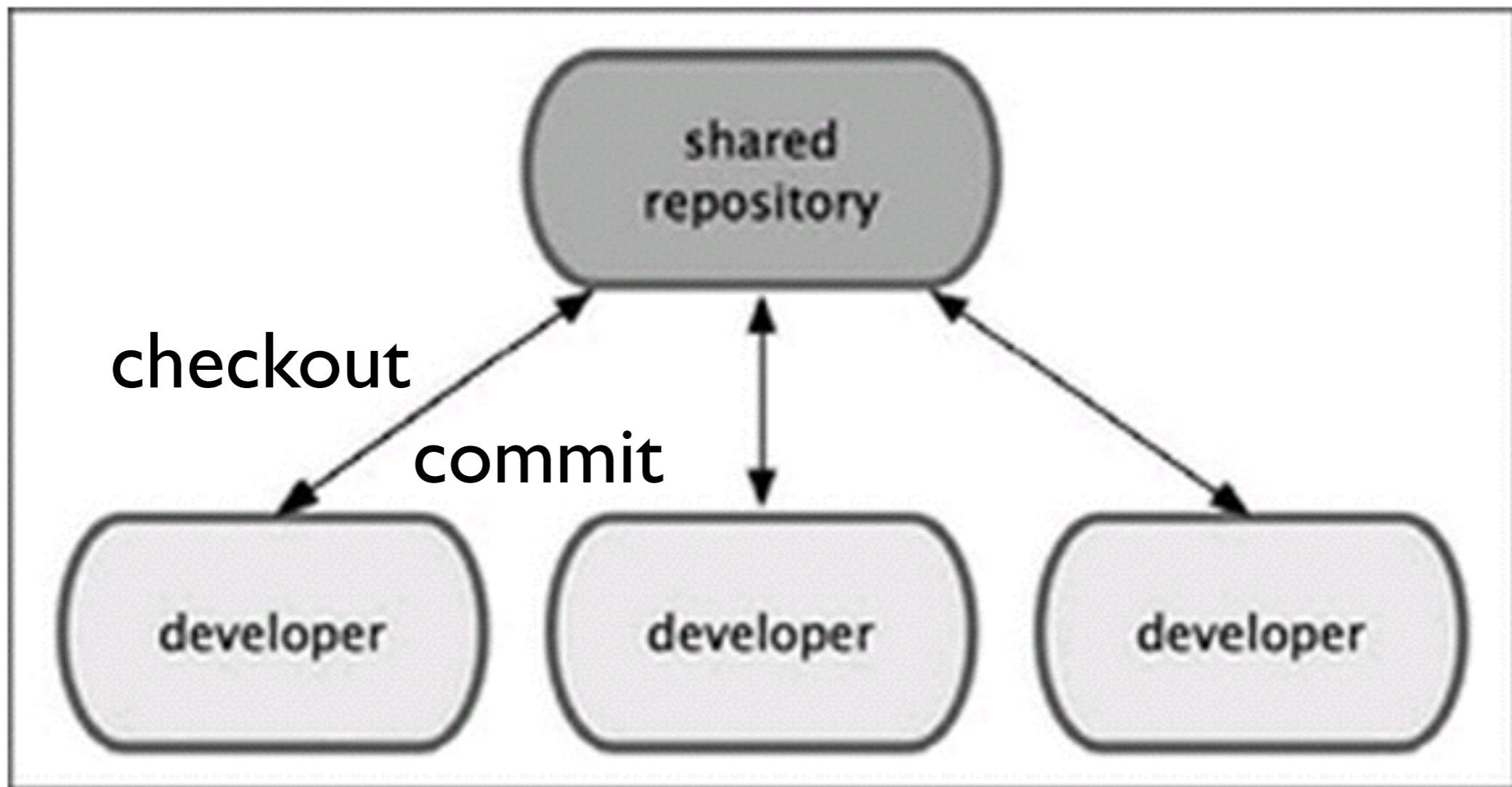
Build



# Collaboration

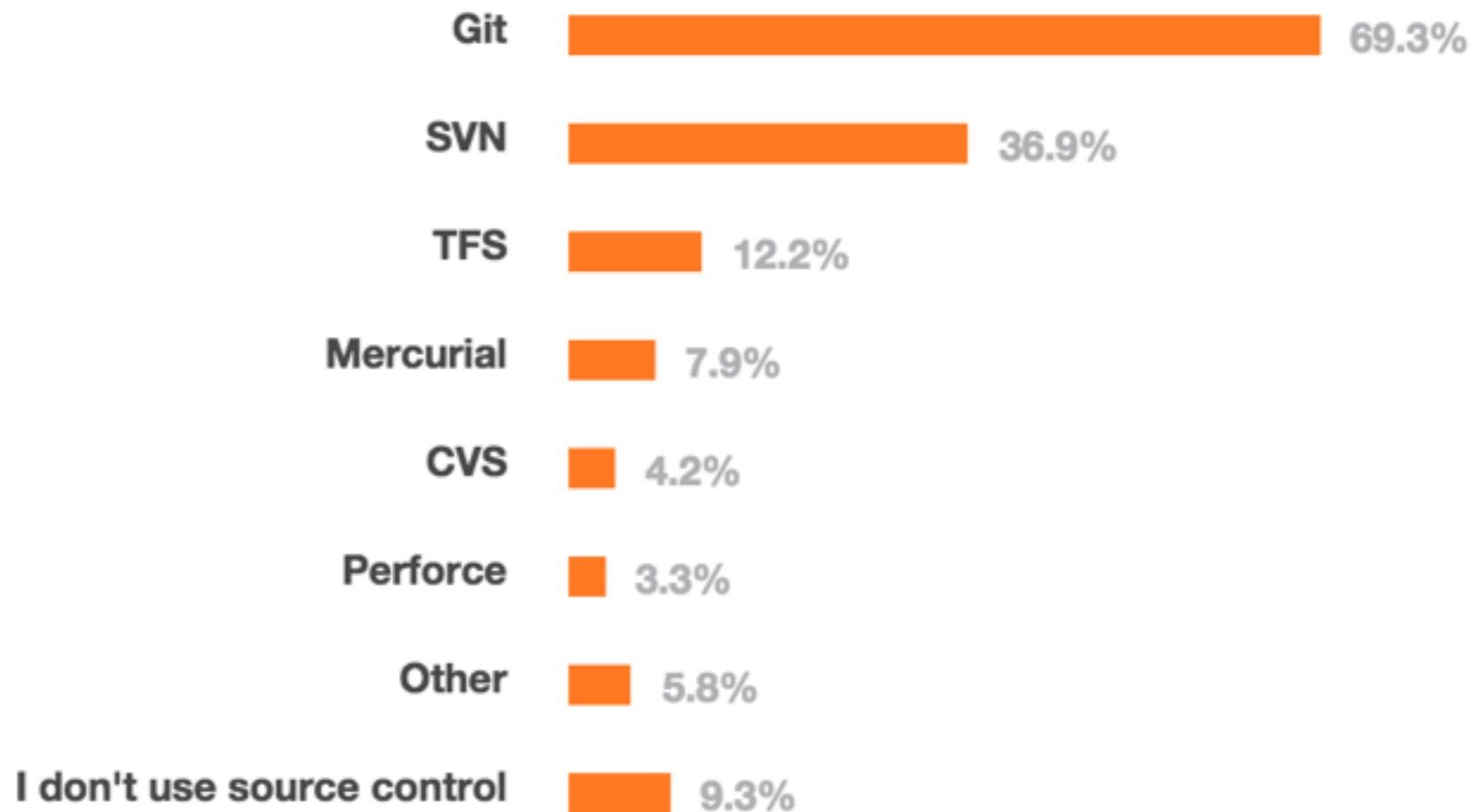


# Version Control System



# 2015 Developer Survey

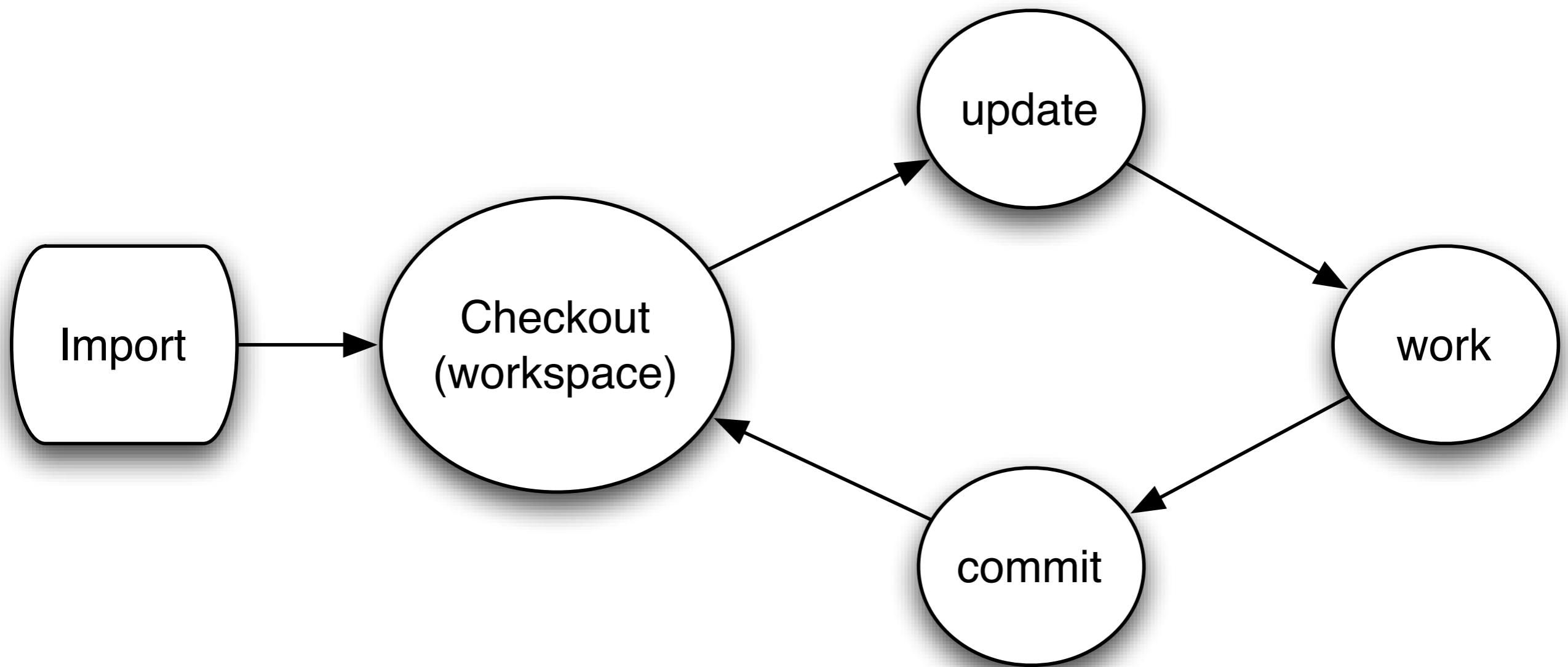
## VI. SOURCE CONTROL



# Subversion (SVN)

- Open-source version control system
- Compatible successor to the widely used Concurrent Versions System (CVS)
- Change set based (rather than file revision based)
- Directory structure
  - trunk (usually have your projects in trunk)
  - tags
  - branches

# SVN - life cycle



# SVN Workflow

- Create a repository (once)
- Import or checkout the repository
  - Work in your workspace

# Typical work cycle

- Update your working copy
  - `svn update`
- Make changes
  - `svn add`
  - `svn delete`
  - `svn copy`
  - `svn move`
- Examine your changes
  - `svn status`
  - `svn diff`
  - `svn revert`
- Merge others' changes
  - `svn merge`
  - `svn resolved`
- Commit your changes
  - `svn commit`

# Understanding diff

*original:*

1 This part of the  
2 document has stayed the  
3 same from version to  
4 version. It shouldn't  
5 be shown if it doesn't  
6 change. Otherwise, that  
7 would not be helping to  
8 compress the size of the  
9 changes.

10  
11 This paragraph contains  
12 text that is outdated.  
13 It will be deleted in the  
14 near future.

15  
16 It is important to spell  
17 check this dokument. On  
18 the other hand, a  
19 misspelled word isn't  
20 the end of the world.  
21 Nothing in the rest of  
22 this paragraph needs to  
23 be changed. Things can  
24 be added after it.

*new:*

1 This is an important  
2 notice! It should  
3 therefore be located at  
4 the beginning of this  
5 document!

6  
7 This part of the  
8 document has stayed the  
9 same from version to  
10 version. It shouldn't  
11 be shown if it doesn't  
12 change. Otherwise, that  
13 would not be helping to  
14 compress anything.

15  
16 It is important to spell  
17 check this document. On  
18 the other hand, a  
19 misspelled word isn't  
20 the end of the world.  
21 Nothing in the rest of  
22 this paragraph needs to  
23 be changed. Things can  
24 be added after it.

25  
26 This paragraph contains  
27 important new additions  
28 to this document.

# Understanding diff

```
--- /path/to/original      ''timestamp''
+++ /path/to/new          ''timestamp''
@@ -1,3 +1,9 @@
+This is an important
+notice! It should
+therefore be located at
+the beginning of this
+document!
+
    This part of the
    document has stayed the
    same from version to
@@ -5,16 +11,10 @@
    be shown if it doesn't
    change. Otherwise, that
    would not be helping to
-compress the size of the
-changes.
-
    -This paragraph contains
    -text that is outdated.
    -It will be deleted in the
    -near future.
+compress anything.
```

# Understanding diff - unified diff

```
--- /path/to/original      ''timestamp''
+++ /path/to/new          ''timestamp''
@@ -1,3 +1,9 @@
+This is an important
+notice! It should
+therefore be located at
+the beginning of this
+document!
+
 This part of the
 document has stayed the
 same from version to
@@ -5,16 +11,10 @@
 be shown if it doesn't
 change. Otherwise, that
 would not be helping to
-compress the size of the
-changes.
-
-This paragraph contains
-text that is outdated.
-It will be deleted in the
-near future.
+compress anything.
```

meta information

added lines (+)

context

line information

deleted lines (-)

# svn diff is handy

```
$ svn status  
M      foo.c  
$ svn diff
```

Index: foo.c

```
=====--- foo.c (revision 128)  
+++ foo.c (working copy)  
@@ -1003,7 +1003,7 @@  
     return ERROR_ON_THE_G_STRING;  
  
     /* Do something in a loop. */  
- for (i = 0; i < txns->nelts; i++)  
+ for (i = 0; i < txns->nelts; i--)  
{  
    status = do_something(i);  
    if (status)
```

# side diff with GUI

JDK / jdk / openjdk / 6-b14 / java.lang.Long

25 changed segments	<a href="#">Expand All</a>	<a href="#">Collapse All</a>	<a href="#">7-b147</a>	<a href="#">6-b14</a>
1	<pre>/*</pre>		<pre>/*</pre>	
2	<pre>* Copyright (c) 1994, 2009, Oracle and/or its affiliates. All rights reserved.</pre>		<pre>* Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved.</pre>	
3	<pre>* DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS FILE HEADER.</pre>		<pre>* DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS FILE HEADER.</pre>	
4	<pre>*</pre>		<pre>*</pre>	
5	<pre>* This code is free software; you can redistribute it and/or modify it</pre>		<pre>* This code is free software; you can redistribute it and/or modify it</pre>	
6	<pre>* under the terms of the GNU General Public License version 2 only, as</pre>		<pre>* under the terms of the GNU General Public License version 2 only, as</pre>	
7	<pre>* published by the Free Software Foundation. Oracle designates this</pre>		<pre>* published by the Free Software Foundation. Sun designates this</pre>	
8	<pre>* particular file as subject to the "Classpath" exception as provided</pre>		<pre>* particular file as subject to the "Classpath" exception as provided</pre>	
9	<pre>* by Oracle in the LICENSE file that accompanied this code.</pre>		<pre>* by Sun in the LICENSE file that accompanied this code.</pre>	
10	<pre>*</pre>		<pre>*</pre>	
11	<pre>* This code is distributed in the hope that it will be useful, but WITHOUT</pre>		<pre>* This code is distributed in the hope that it will be useful, but WITHOUT</pre>	
		<a href="#">Show 7 unchanged lines</a>		
19	<pre>* Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA.</pre>		<pre>* Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA.</pre>	
20	<pre>*</pre>		<pre>*</pre>	
21	<pre>* Please contact Oracle, 500 Oracle Parkway, Redwood Shores, CA 94065 USA</pre>		<pre>* Please contact Sun Microsystems, Inc., 4150 Network Circle, Santa Clara,</pre>	
22	<pre>* or visit <a href="http://www.oracle.com">www.oracle.com</a> if you need additional information or have any</pre>		<pre>* CA 95054 USA or visit <a href="http://www.sun.com">www.sun.com</a> if you need additional information or</pre>	
23	<pre>* questions.</pre>		<pre>* have any questions.</pre>	
24	<pre>*/</pre>		<pre>*/</pre>	
25				
		<a href="#">Show 318 unchanged lines</a>		
344	<pre>* a nonnegative value), except that the first character may be an</pre>		<pre>* a nonnegative value), except that the first character may be an</pre>	
345	<pre>* ASCII minus sign {@code '-'} ({&lt;code&gt;'#92;u002D'&lt;/code&gt;}) to</pre>		<pre>* ASCII minus sign {@code '-'} ({&lt;code&gt;'#92;u002D'&lt;/code&gt;}) to</pre>	
346	<pre>* indicate a negative value or an ASCII plus sign {@code '+'}</pre>		<pre>* indicate a negative value. The resulting {@code long} value is</pre>	
347	<pre>* ({&lt;code&gt;'#92;u002B'&lt;/code&gt;}) to indicate a positive value. The</pre>		<pre>* returned.</pre>	
348	<pre>* resulting {@code long} value is returned.</pre>			
349	<pre>*</pre>		<pre>*</pre>	
350	<pre>* &lt;p&gt;Note that neither the character {@code L}</pre>		<pre>* &lt;p&gt;Note that neither the character {@code L}</pre>	
		<a href="#">Show 18 unchanged lines</a>		
369	<pre>* &lt;li&gt;Any character of the string is not a digit of the specified</pre>		<pre>* &lt;li&gt;Any character of the string is not a digit of the specified</pre>	
370	<pre>* radix, except that the first character may be a minus sign</pre>		<pre>* radix, except that the first character may be a minus sign</pre>	
371	<pre>* {@code '-'} ({&lt;code&gt;'#92;u002d'&lt;/code&gt;}) or plus sign {@code</pre>		<pre>* {@code '-'} ({&lt;code&gt;'#92;u002d'&lt;/code&gt;}) provided that the</pre>	
372	<pre>* '+') ({&lt;code&gt;'#92;u002B'&lt;/code&gt;}) provided that the string is</pre>		<pre>* string is longer than length 1.</pre>	
373	<pre>* longer than length 1.</pre>			
374	<pre>*</pre>		<pre>*</pre>	
375	<pre>* &lt;li&gt;The value represented by the string is not a value of type</pre>		<pre>* &lt;li&gt;The value represented by the string is not a value of type</pre>	
376	<pre>*     {@code long}.</pre>		<pre>*     {@code long}.</pre>	
377	<pre>* &lt;/ul&gt;</pre>		<pre>* &lt;/ul&gt;</pre>	
378	<pre>*</pre>		<pre>*</pre>	
379	<pre>* &lt;p&gt;Examples:</pre>		<pre>* &lt;p&gt;Examples:</pre>	
380	<pre>* &lt;blockquote&gt;&lt;pre&gt;</pre>		<pre>* &lt;blockquote&gt;&lt;pre&gt;</pre>	
381	<pre>* parseLong("0", 10) returns 0L</pre>		<pre>* parseLong("0", 10) returns 0L</pre>	
382	<pre>* parseLong("473", 10) returns 473L</pre>		<pre>* parseLong("473", 10) returns 473L</pre>	
383	<pre>* parseLong("+42", 10) returns 42L</pre>			
384	<pre>* parseLong("-0", 10) returns 0L</pre>		<pre>* parseLong("-0", 10) returns 0L</pre>	
385	<pre>* parseLong("-FF", 16) returns -255L</pre>		<pre>* parseLong("-FF", 16) returns -255L</pre>	
		<a href="#">Show 37 unchanged lines</a>		

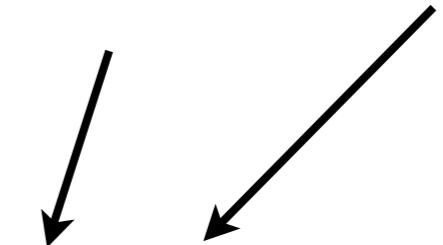
# wdiff (word diff)

```
- // White line Check
+ if(lineContent.trim().length() < 1){
+   if(lineContent.trim().length() == 1){
+     numOfAddedWhiteLines++;
```

```
// White line Check
```

```
if(lineContent.trim().length() <== 1) {
    numOfAddedWhiteLines++;
```

removed      added



# Eclipse diff view (GUI + wdiff)

The screenshot shows the Eclipse Java Source Compare interface comparing two versions of the `MetricsRepository.java` file. The left pane shows the code at revision 598, and the right pane shows the code at revision 499. The diff highlights changes in the `doubleSpacePattern` and `tabIndentationPattern` declarations, and the corresponding logic for double-space and tab-indentation checks.

```
Java Source Compare ▾
```

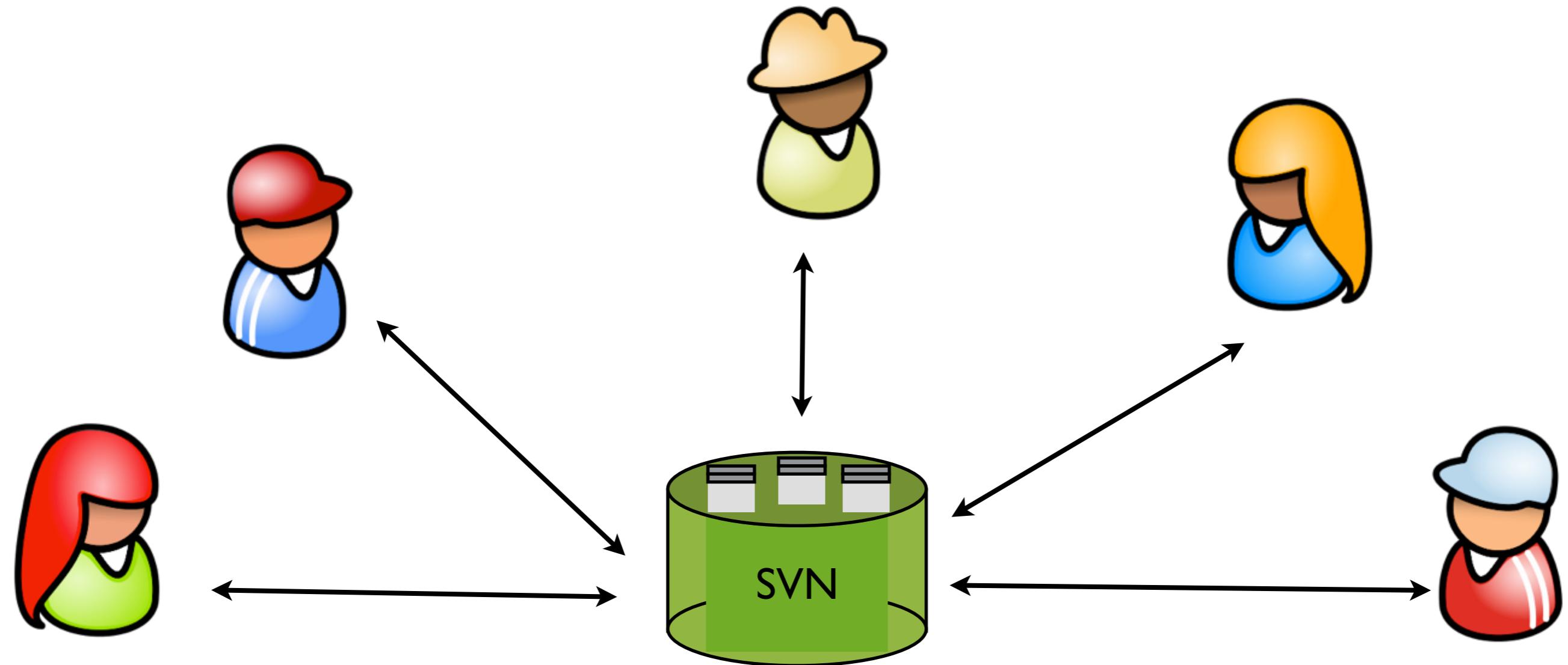
```
http://nforge.cse.ust.hk/svn/patch/trunk/MetricsExtractor-FSE-Eclips...st/cse/pam/patch/metrics/repository/MetricsRepository.java [Rev:598]
```

```
http://nforge.cse.ust.hk/svn/patch/trunk/MetricsExtractor-FSE-Eclips...st/cse/pam/patch/metrics/repository/MetricsRepository.java [Rev:499]
```

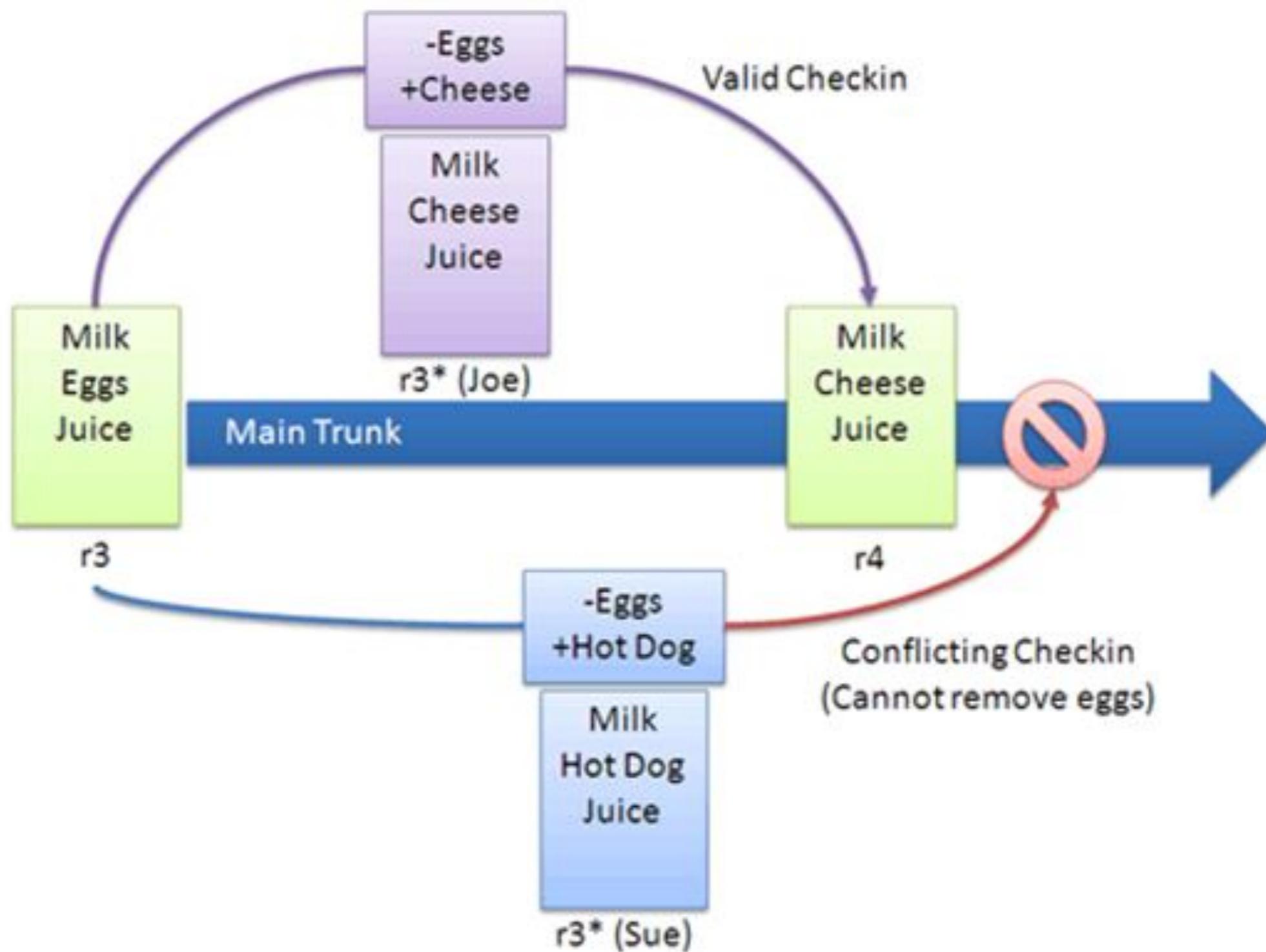
```
List<Integer> addedWordsInIdentifiers = new ArrayList<Integer>();  
List<Integer> deletedWordsInIdentifiers = new ArrayList<Integer>();  
  
Pattern doubleSpacePattern = Pattern.compile("[a-zA-Z0-9_\\(\\)\\{\\}]+(\\s{2})[a-zA-Z0-9_\\(\\)\\{\\}]");  
Pattern tabIndentationPattern = Pattern.compile("^([\\t\\r\\n]+)");  
DiffType diffType = new DiffType(patch.getData());  
for(CodeChurn codeChurn : diffType.getCodeChurns()){  
    if (!codeChurn.getFileName().endsWith(".java")){  
        continue;  
    }  
  
    // Inspect Added Lines  
    for(Integer addedLine : codeChurn.getAddedLines().keySet()){  
        String lineContent = codeChurn.getAddedLines().get(addedLine);  
  
        addedLineLengths.add(lineContent.length());  
  
        // White line Check  
        if(lineContent.trim().length() == 1){  
            numOfAddedWhiteLines++;  
        }  
  
        // Double Space Check  
        Matcher doubleSpaceMatcher = doubleSpacePattern.matcher(lineContent);  
        while(doubleSpaceMatcher.find()){  
            numOfAddedDoubleWhiteSpaces++;  
        }  
  
        //Indentation Check - Tab Indentation  
        Matcher indentationMatcher = tabIndentationPattern.matcher(lineContent);  
        if(indentationMatcher.find()){  
            numOfAddedIndentations += indentationMatcher.group(1).length();  
        }  
  
        //Indentation Check - Four Space Indentation  
        String removeSpaceIndentation = lineContent.substring(1).replaceAll("^(\\s{4})","");
        numOfAddedIndentations += (lineContent.length() - removeSpaceIndentation.length()) / 4;  
  
        for(int i=0; i < lineContent.length(); i++){  
            //Punctuation Marks  
            if(lineContent.charAt(i) == '.' || lineContent.charAt(i) == ','  
                || lineContent.charAt(i) == ';' || lineContent.charAt(i) == '('  
                || lineContent.charAt(i) == ')')  
                numOfAddedPunctuationMarks++;  
        }  
    }  
}
```

```
List<Integer> addedWordsInIdentifiers = new ArrayList<Integer>();  
List<Integer> deletedWordsInIdentifiers = new ArrayList<Integer>();  
  
Pattern doubleSpacePattern = Pattern.compile("[a-zA-Z0-9_\\(\\)\\{\\}]+(\\s{2})[a-zA-Z0-9_\\(\\)\\{\\}]");  
Pattern indentationPattern = Pattern.compile("^([\\t\\r\\n]+)");  
DiffType diffType = new DiffType(patch.getData());  
for(CodeChurn codeChurn : diffType.getCodeChurns()){  
    if (!codeChurn.getFileName().endsWith(".java")){  
        continue;  
    }  
  
    // Inspect Added Lines  
    for(Integer addedLine : codeChurn.getAddedLines().keySet()){  
        String lineContent = codeChurn.getAddedLines().get(addedLine);  
  
        addedLineLengths.add(lineContent.length());  
  
        // White line Check  
        if(lineContent.trim().length() < 1){  
            numOfAddedWhiteLines++;  
        }  
  
        // Double Space Check  
        Matcher doubleSpaceMatcher = doubleSpacePattern.matcher(lineContent);  
        while(doubleSpaceMatcher.find()){  
            numOfAddedDoubleWhiteSpaces++;  
        }  
  
        //Indentation Check  
        Matcher indentationMatcher = indentationPattern.matcher(lineContent);  
        if(indentationMatcher.find()){  
            numOfAddedIndentations += indentationMatcher.group(1).length();  
        }  
  
        for(int i=0; i < lineContent.length(); i++){  
            //Punctuation Marks  
            if(lineContent.charAt(i) == '.' || lineContent.charAt(i) == ','  
                || lineContent.charAt(i) == ';' || lineContent.charAt(i) == '('  
                || lineContent.charAt(i) == ')'  
                || lineContent.charAt(i) == '?')  
                numOfAddedPunctuationMarks++;  
        }  
    }  
}
```

# Now we can work simultaneously!



# Conflicts



# Resolving conflicts

```
$ svn commit -m "Updated README"  
Sending      README  
Transmitting file data .svn: Commit failed (details follow):  
svn: Out of date: '/myproject/README'
```

```
$ svn update  
C  README  
Updated to revision 6.
```

# Resolving conflicts

```
$cat README

<<<<< .mine
This is fun stuff!
=====
This is a documentation file
>>>>> .r6
```

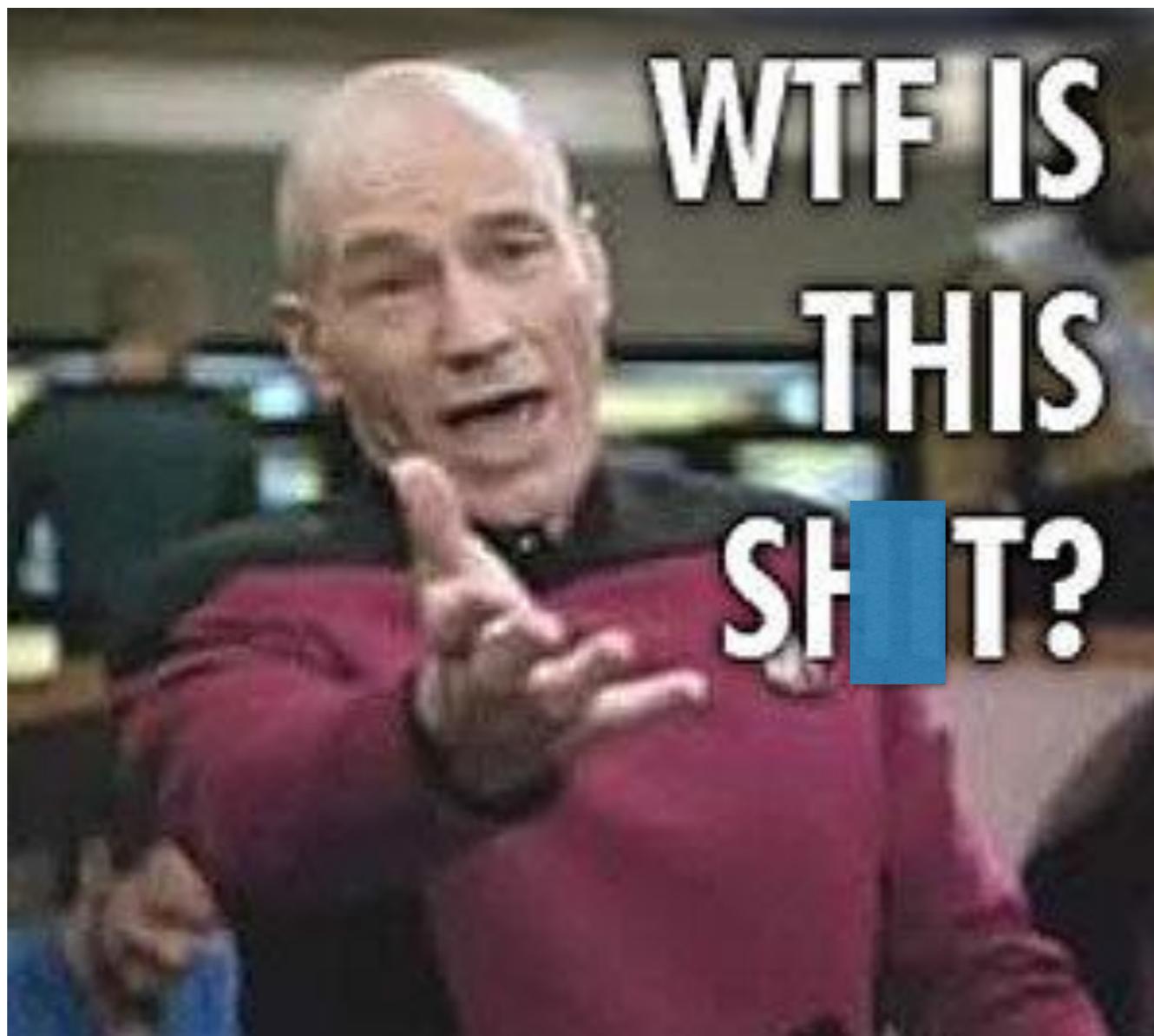
# Resolving conflicts: What Should I do?

1. Scrap your changes, and go with the current work from your colleague.
2. Keep your changes, and dump whatever your colleague did.
3. Merge both versions to a new version

# I. Scrap your changes

```
$ svn revert README  
Reverted 'README'  
$ svn update README  
At revision 6.
```

2. Keep your changes, and dump whatever your colleague did.



2. Keep your changes, and dump  
whatever your colleague did.



### 3. Merge both versions to a new version



- Manually edit the file (README)
  - remove the marks and add what you want to do
- SVN won't commit this yet, so you need to make it is resolved!

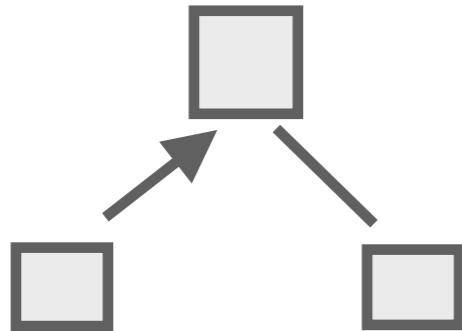
```
$ svn resolved README
Resolved conflicted state of 'README'
```

# Avoiding conflicts

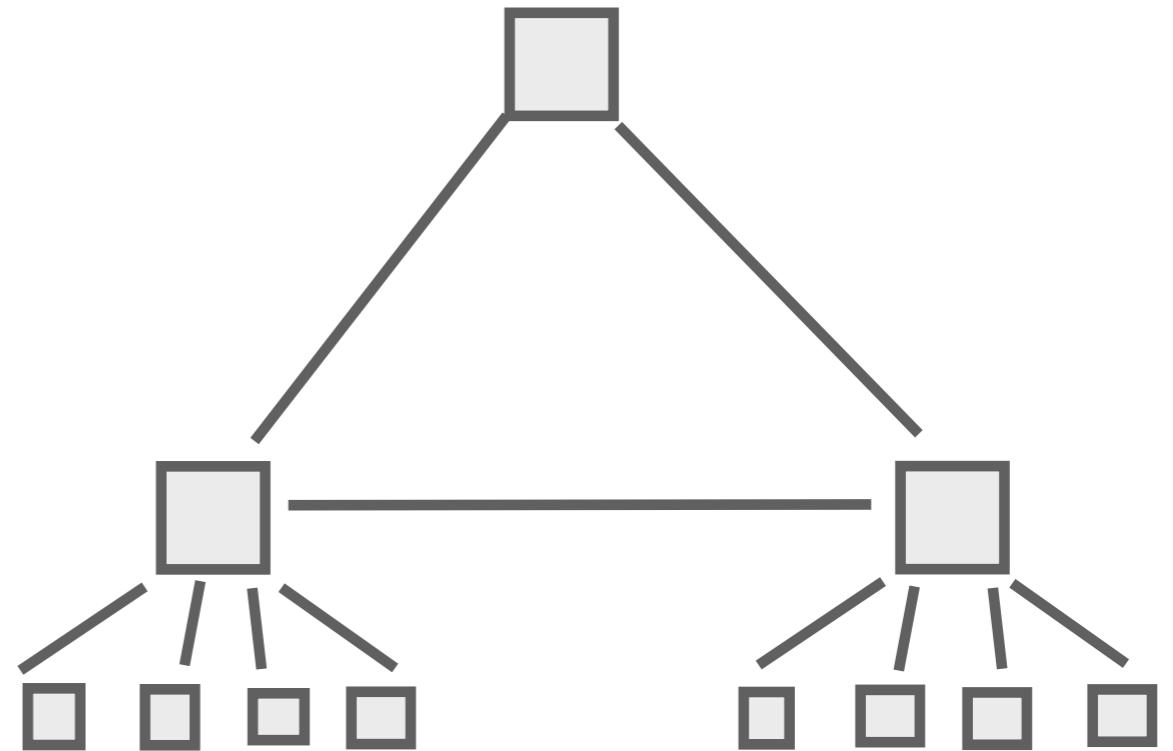
- It won't happen too often
  - SVN automatically merges if different parts are changed. (good or bad)?
- do svn update frequently
- svn lock/unclock
- Refactoring
  - better modularization

# Git

SVN one server



if someone make changes, everyone else can see it



multiple servers, each person has their own servers

# git - the simple guide

just a simple guide for getting started with git. no deep shit ;)



by Roger Dudler

credits to @tfnico, @fhd and Namics

this guide in deutsch, español, français, indonesian, italiano, nederlands, polski, português, русский, тürkçe,

ไทย, 日本語, 中文, 한국어 Vietnamese

please report issues on github

# create a new repository

create a new directory, open it and perform a

`git init`

to create a new git repository.

# workflow

your local repository consists of three "trees" maintained by git. the first one is your **Working Directory** which holds the actual files. the second one is the **Index** which acts as a staging area and finally the **HEAD** which points to the last commit you've made.



# add & commit

You can propose changes (add it to the **Index**) using

```
git add <filename>
```

```
git add *
```

This is the first step in the basic git workflow. To actually commit these

changes use

```
git commit -m "Commit message"
```

Now the file is committed to the **HEAD**, but not in your remote repository yet.

# branching

Branches are used to develop features isolated from each other. The *master* branch is the "default" branch when you create a repository. Use other branches for development and merge them back to the master branch upon completion.



# branching

create a new branch named "feature\_x" and switch to it using

```
git checkout -b feature_x
```

switch back to master

```
git checkout master
```

and delete the branch again

```
git branch -d feature_x
```



# update & merge

```
git merge <branch>
```

in both cases git tries to auto-merge changes. Unfortunately, this is not always possible and results in *conflicts*. You are responsible to merge those *conflicts* manually by editing the files shown by git. After changing, you need to mark them as merged with

```
git add <filename>
```

before merging changes, you can also preview them by using

```
git diff <source_branch> <target_branch>
```

# log

in its simplest form, you can study repository history using.. `git log`

You can add a lot of parameters to make the log look like what you want.

To see only the commits of a certain author:

`git log --author=bob`

To see a very compressed log where each commit is one line:

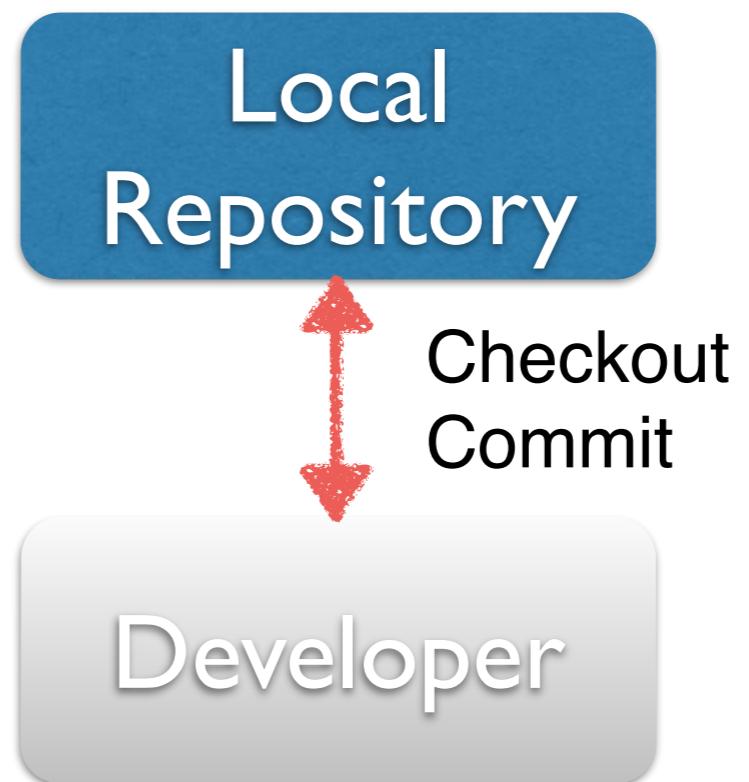
`git log --pretty=oneline`

Or maybe you want to see an ASCII art tree of all the branches,

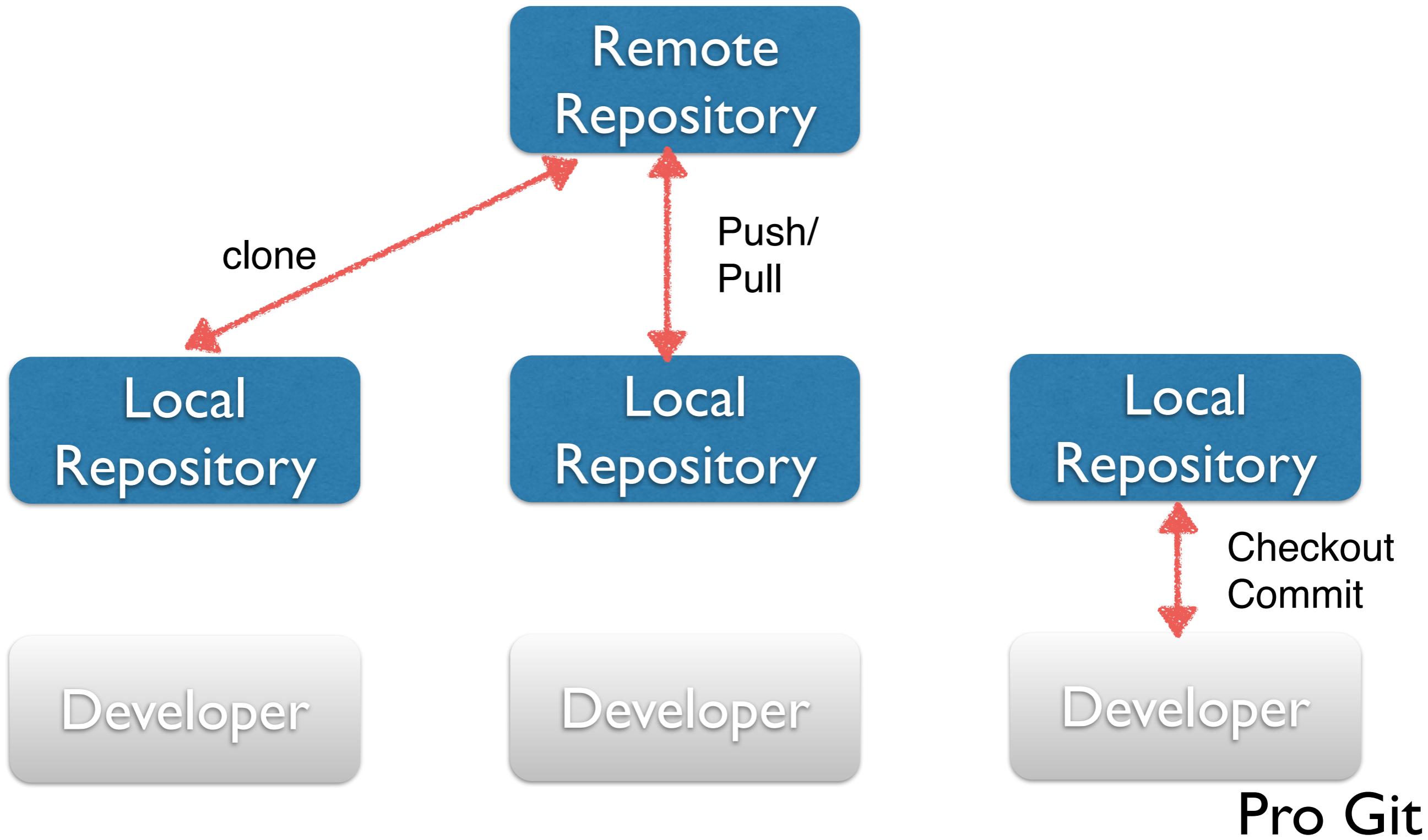
decorated with the names of tags and branches:

`git log --graph --oneline --decorate --all`

# Git basic workflow



# Git basic workflow



# checkout a repository

create a working copy of a local repository by running the command

```
git clone /path/to/repository
```

when using a remote server, your command will be

```
git clone username@host:/path/to/repository
```

# pushing changes

Your changes are now in the **HEAD** of your local working copy. To send those changes to your remote repository, execute

```
git push origin master
```

Where What

Change *master* to whatever branch you want to push your changes to.

If you have not cloned an existing repository and want to connect your repository to a remote server, you need to add it with

```
git remote add origin <server>
```

Now you are able to push your changes to the selected remote server

# branching

remote repository

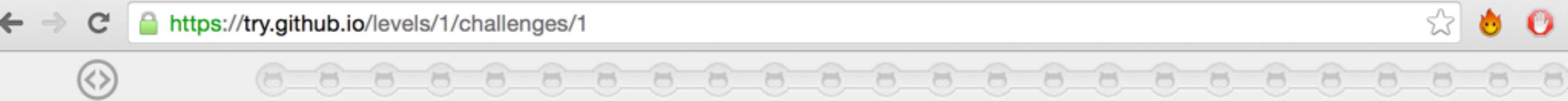
```
git push origin <branch>
```

# update & merge

to update your local repository to the newest commit, execute

`git pull`

in your working directory to *fetch* and *merge* remote changes.



## 1.1 · Got 15 minutes and want to learn Git?

Git allows groups of people to work on the same documents (often code) at the same time, and without stepping on each other's toes. It's a distributed version control system.

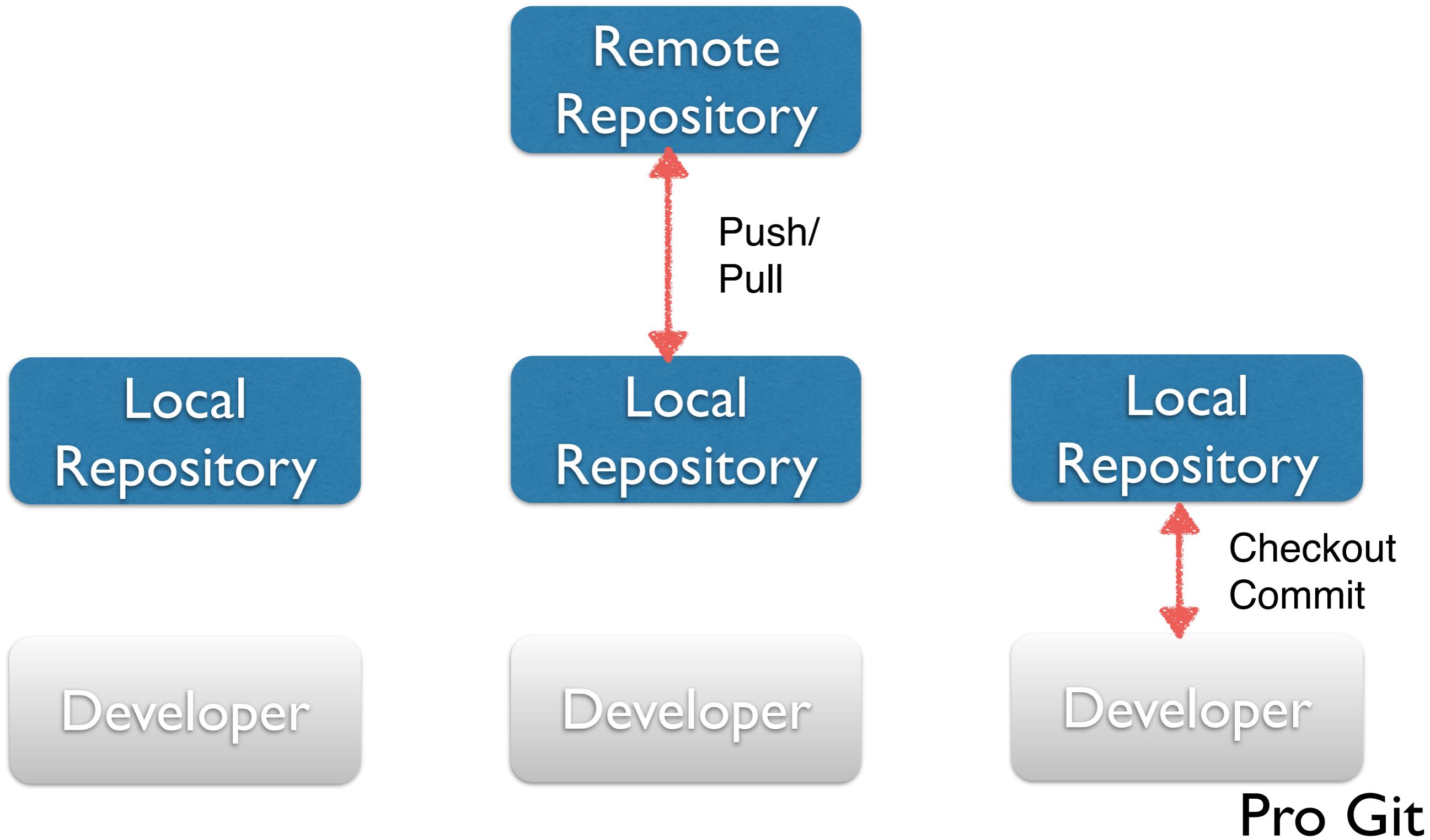
Our terminal prompt below is currently in a directory we decided to name "octobox". To initialize a Git repository here, type the following command:

→ **git init**

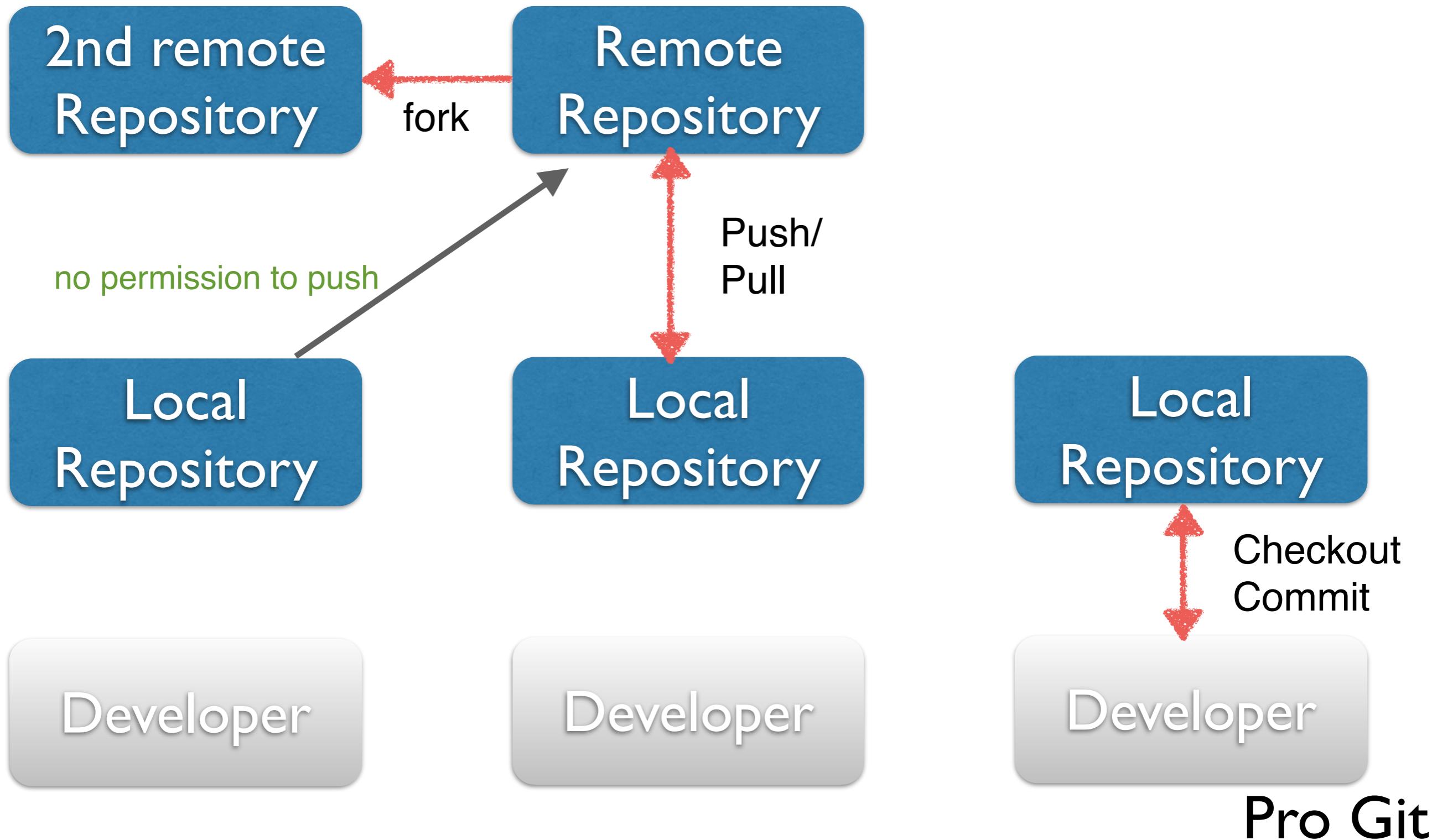


```
Press enter to submit commands  
>
```

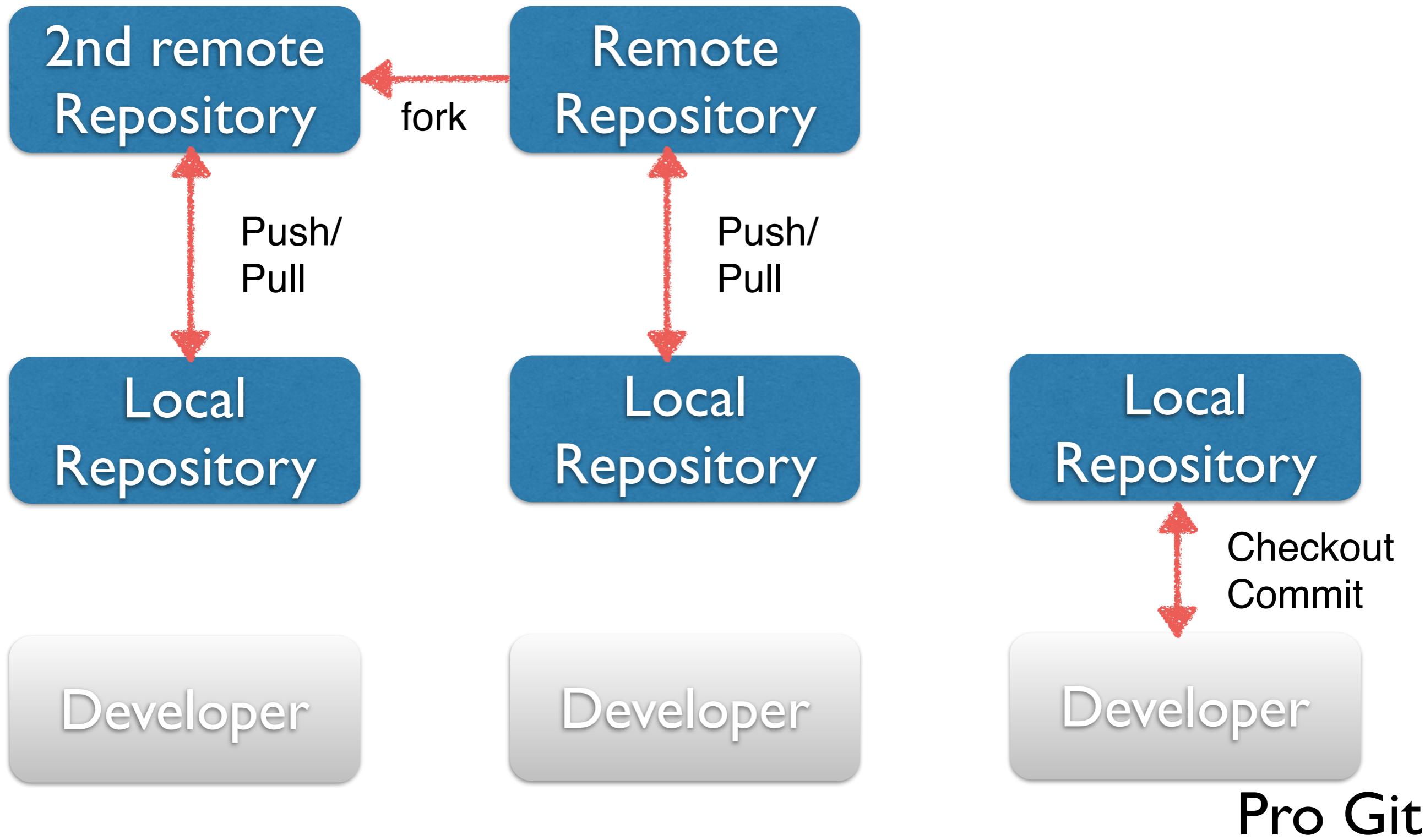
# Git basic workflow



# Git workflow



# Git workflow





This repository Search

Pull requests Issues Gist



COMP3111 / questionsJ

Where should we fork this repository?



Watch 5

★ Star 6

Fork 21

### Description

Short description of this repository

48 commits



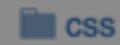
Branch: gh-pages

questionsJ

Merge pull request #9 from logchan/gh-page



hunkim authored 4 days ago



Created F



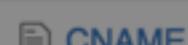
Created F



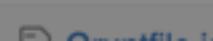
Added mo



Upated CSS



Changed the domain name



Added a file



@hunkim



@PAMSE



@daumkaka...



@COMP3111

or Cancel

<> Code

Issues 6

Pull requests 0

Wiki

Pulse

Graphs

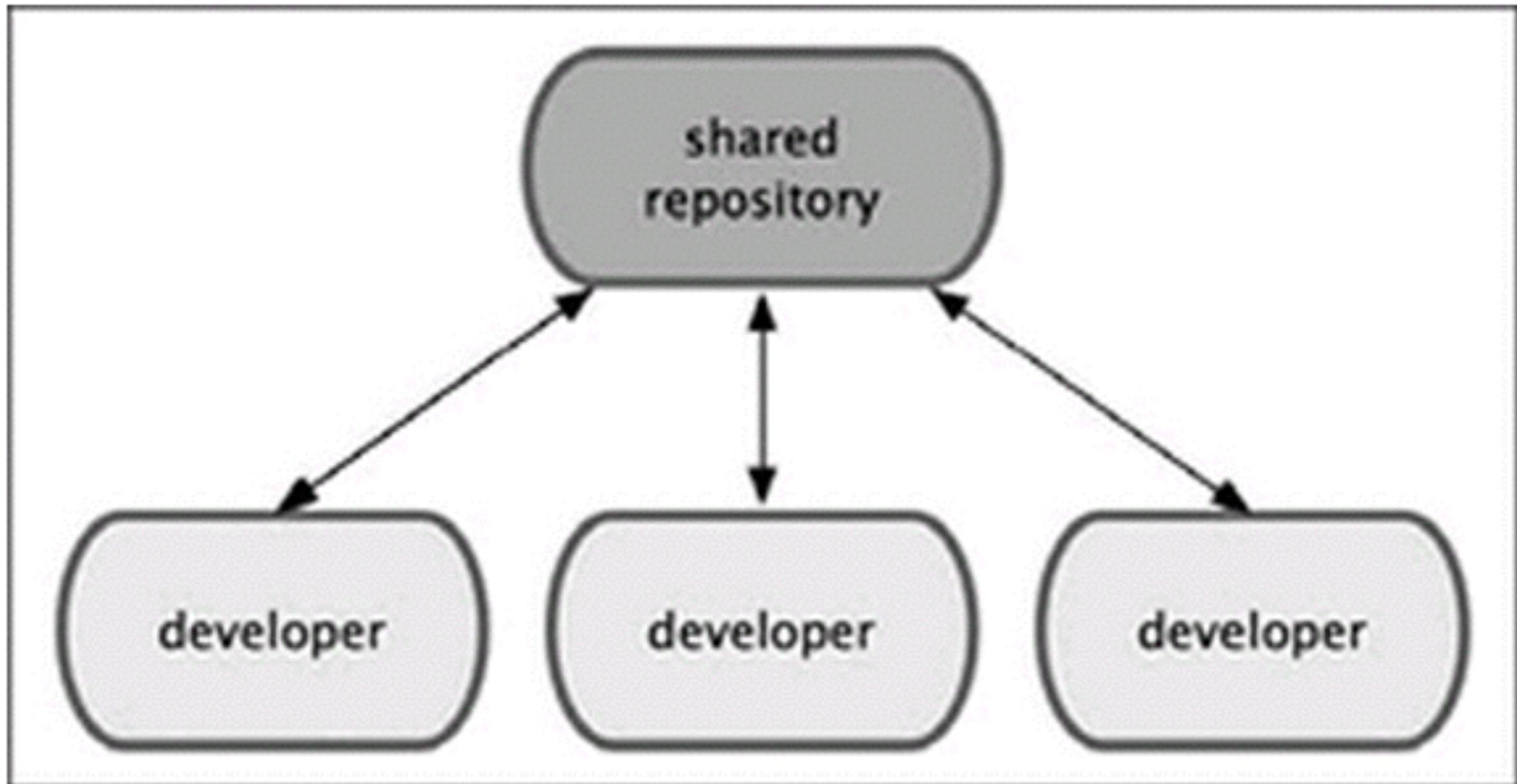
Settings

SSH clone URL

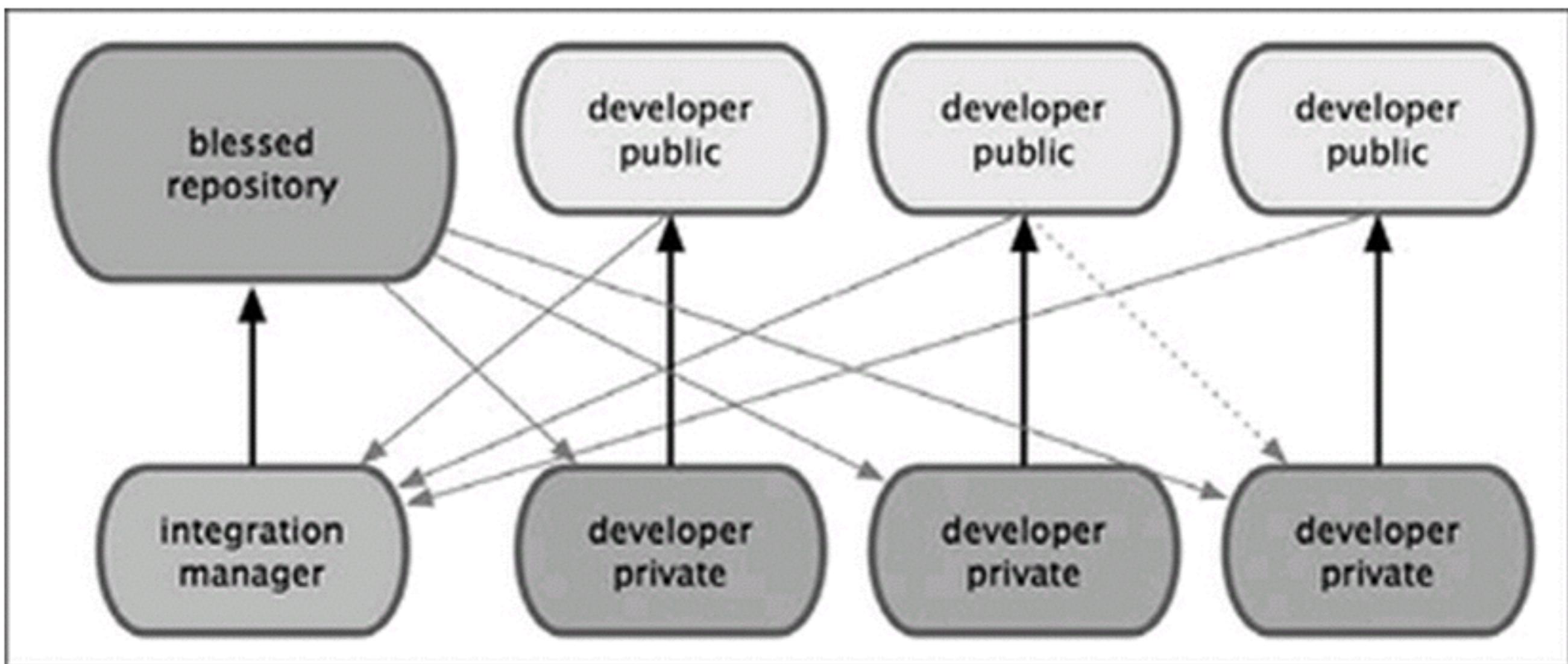
git@github.com:COM



# Basic workflow

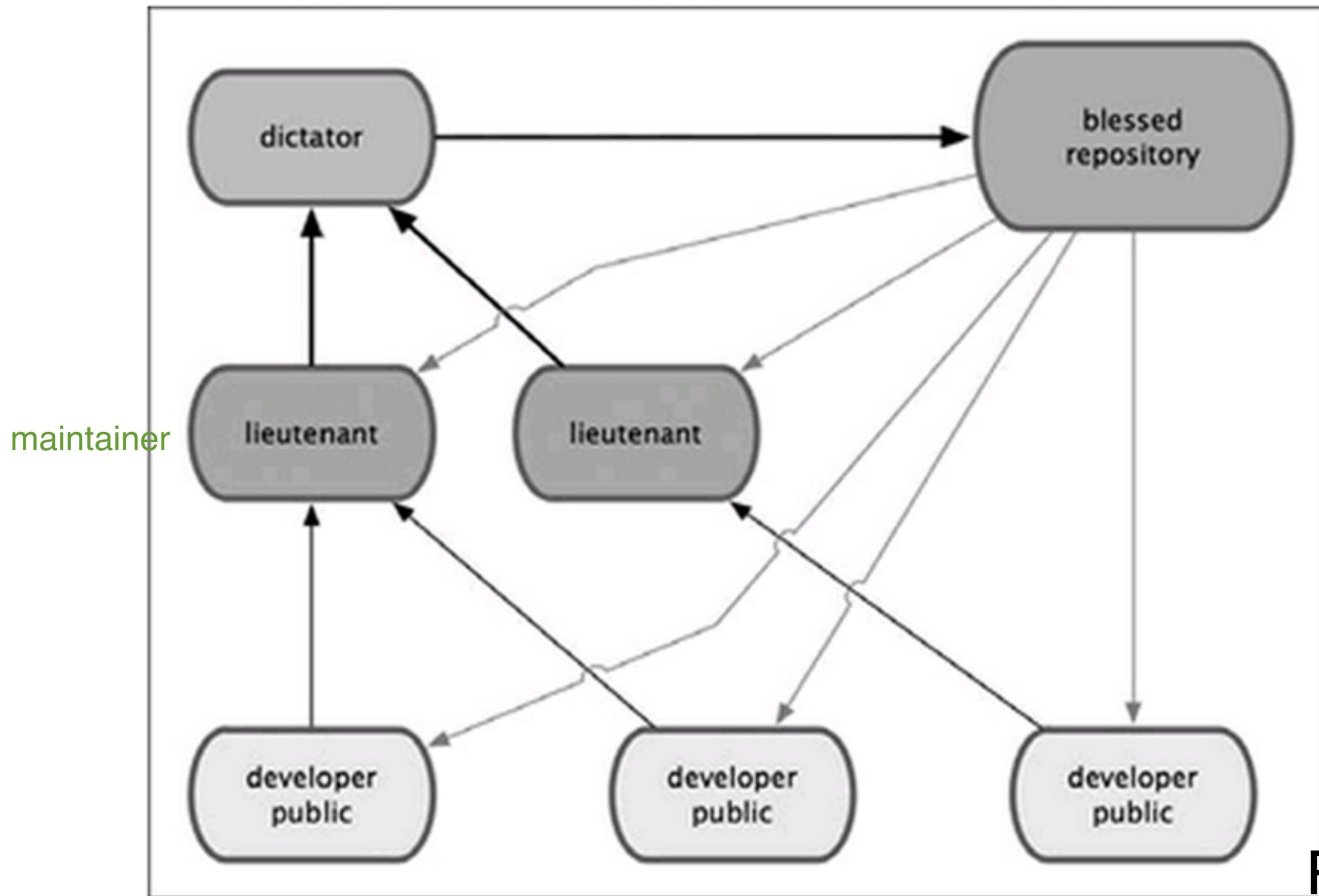


# Integration-manager Workflow



everyone can clone, but cannot push

# Benevolent dictator workflow



# Multiple git servers? Automatic deploy?

development  
server

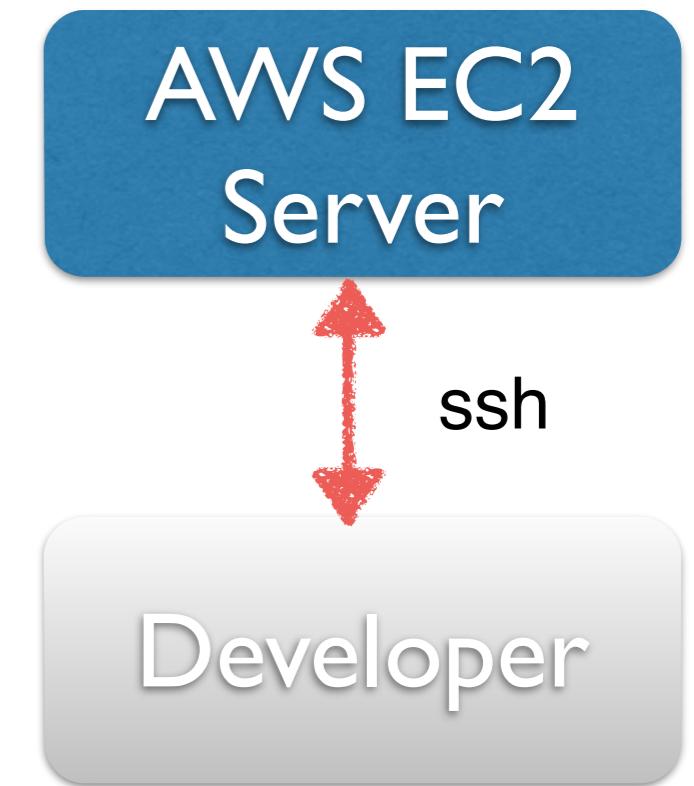
test server

...

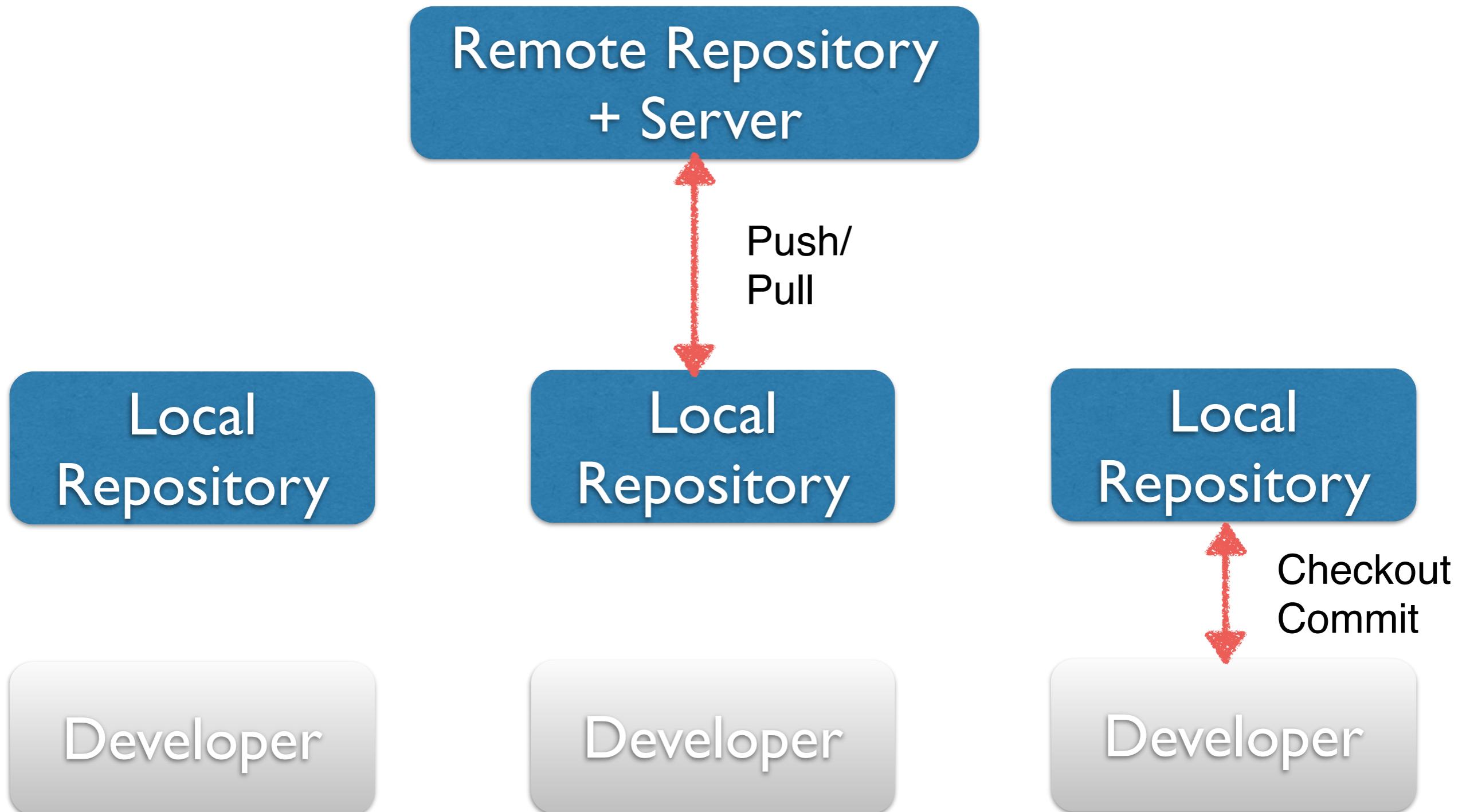
production  
server

Mac/PC  
(Local machine)

# Manage your web page using git?



# Manage your web page using git?



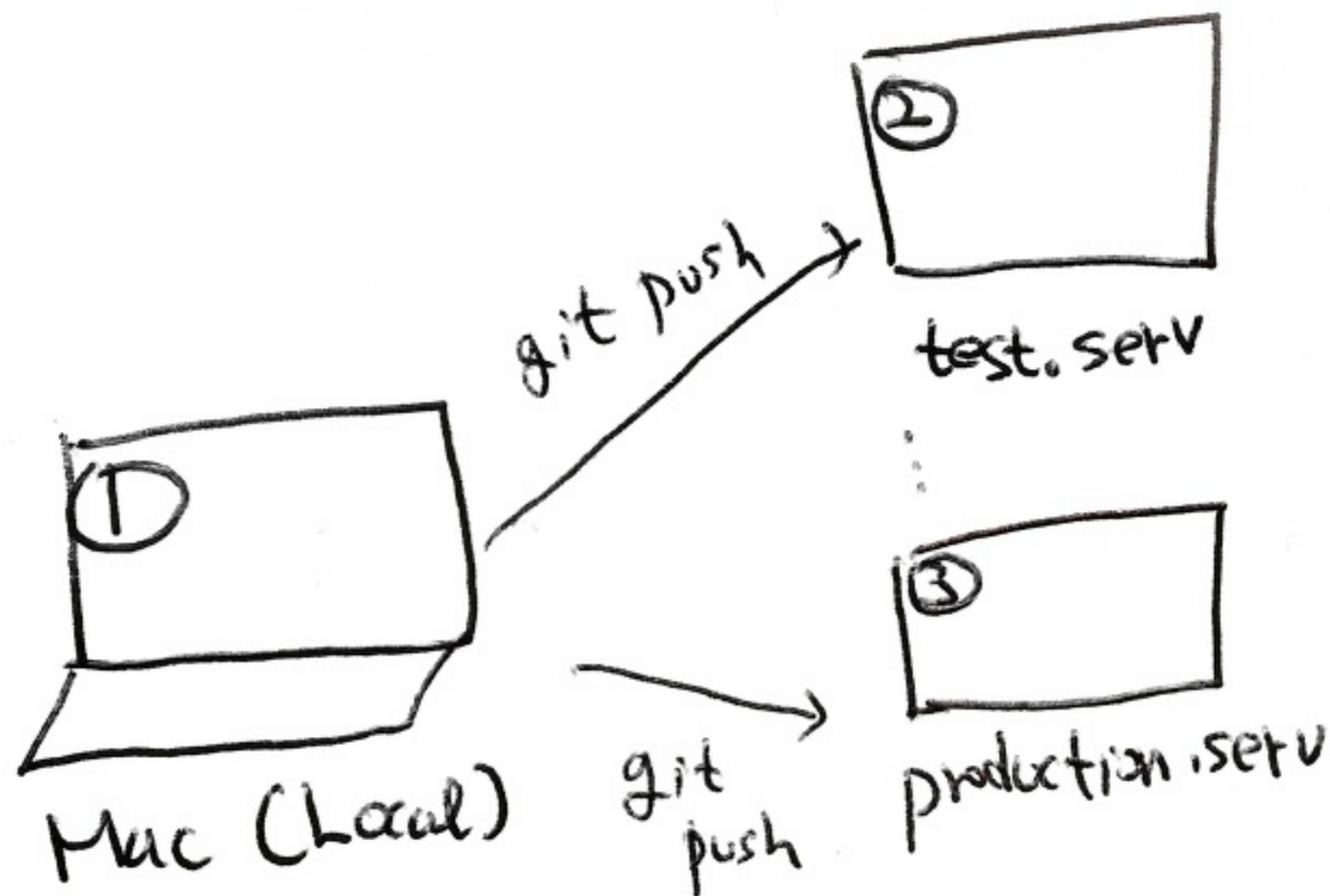
# Demo

set up your own git server

(no more privacy problem, other groups wont see our work)

(advanced, not compulsory)

- git (bare) + post-receive hook
- <http://toroid.org/ams/git-website-howto>
- Similar to Heroku



# I. local (mac)

everything still in local computer

```
$ mkdir website && cd website
```

```
$ git init
```

Initialized empty Git repository in /home/ams/website/.git/

```
$ echo 'Hello, world!' > index.html
```

```
$ git add index.html
```

inform others what I have done

```
$ git commit -q -m "The humble beginnings of my web site."
```

add and commit only done locally

## 2. Server I

\$ ssh ubuntu@test.serv if unsure can use ifconfig command to test where the file is

\$ mkdir website.git && cd website.git set up git server

\$ git init --bare

Initialized empty Git repository in /home/ubuntu/website.git/

\$ pwd

/home/ubuntu/website.git

# 3. post-hook (at server I)

```
$ cat > hooks/post-receive
```

```
#!/bin/sh
```

```
GIT_WORK_TREE=/var/www/html git checkout -f
```

```
$ chmod +x hooks/post-receive
```

execution permission to the program

root directory of my website  
changes will be pushed here

## 4. add remote at local

```
$ git remote add test ssh://ubutu@test.serv/  
home/ubuntu/website.git    can use remote -v command to double check
```

```
$ git push test +master:refs/heads/master
```

find ip address (the web address) back in the aws web

# Todo

- More about git <http://githowto.com/>
- Create your github account
- (git) Fork COMP3111 projects (JS/Android)
- (git) Clone from your (forked) repositories
- Make changes, commit, and push!