

The background features a large, semi-transparent watermark of the Tsinghua University logo. The logo is circular, with the text "TSINGHUA UNIVERSITY" at the top and "SINCE 2013" at the bottom. In the center is a stylized building with a triangular roof and five vertical bars of different colors (yellow, orange, red, purple, blue) below it.

Git的基本命令和工作流简介

陈晨@TiUP

The background features a large, semi-transparent watermark of the Tsinghua University seal. The seal is circular, with the text 'TSINGHUA UNIVERSITY' at the top and 'SINCE 2013' at the bottom. In the center is a stylized building with five vertical bars of different colors (yellow, orange, red, purple, blue) below it.

Git简介

Git特性

快照

直接记录快照，
而非差异比较

本地

近乎所有操作都是
本地执行

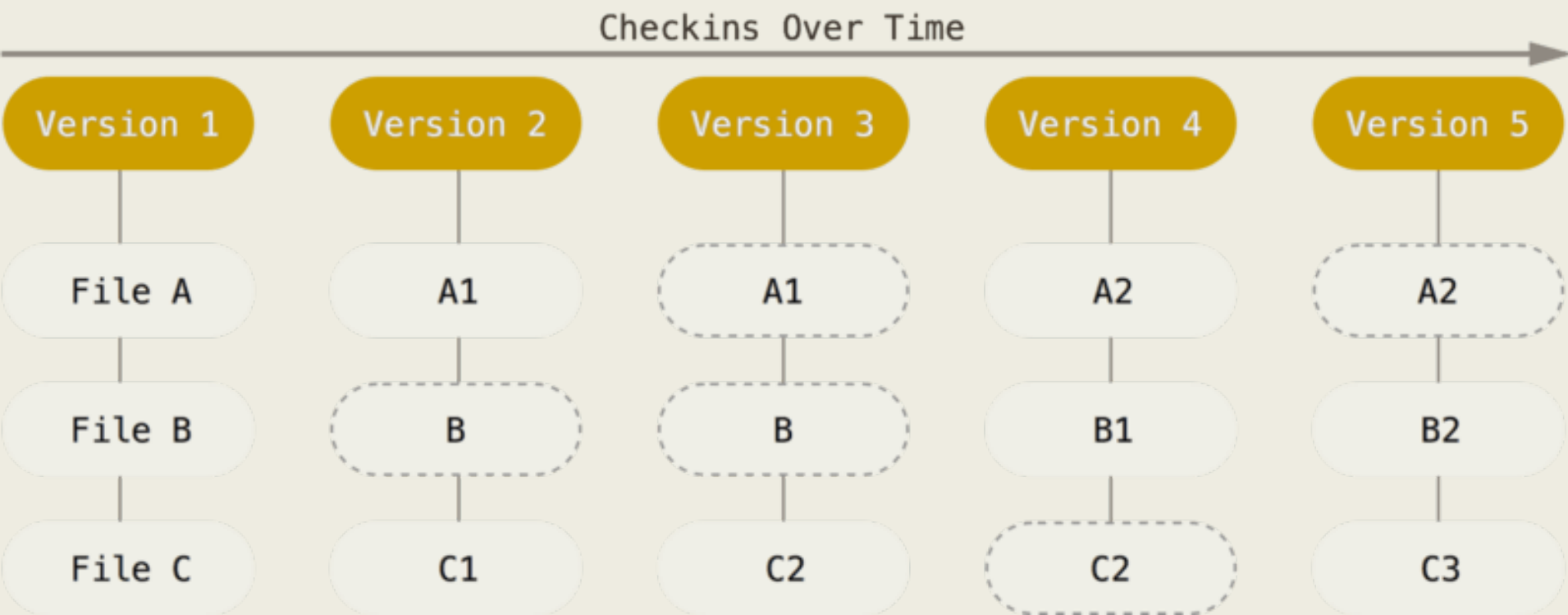
完整

Git 保证完整性，
SHA-1

添加

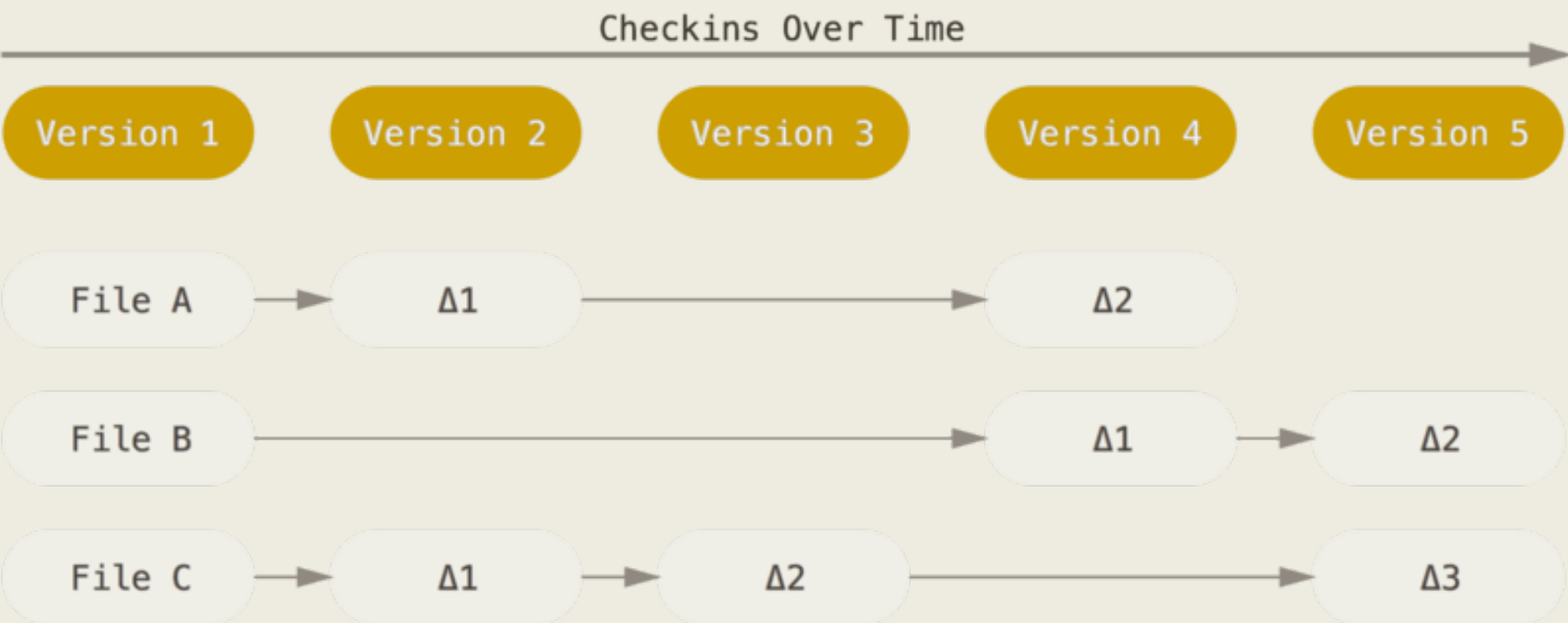
Git 一般只添
加数据。

Git的基本命令和工作流简介



Git存储项目随时间改变的快照

Git的基本命令和工作流简介



SVN等存储每个文件与初始版本的差异

Git特性

快照

直接记录快照，
而非差异比较

本地

近乎所有操作都是
本地执行

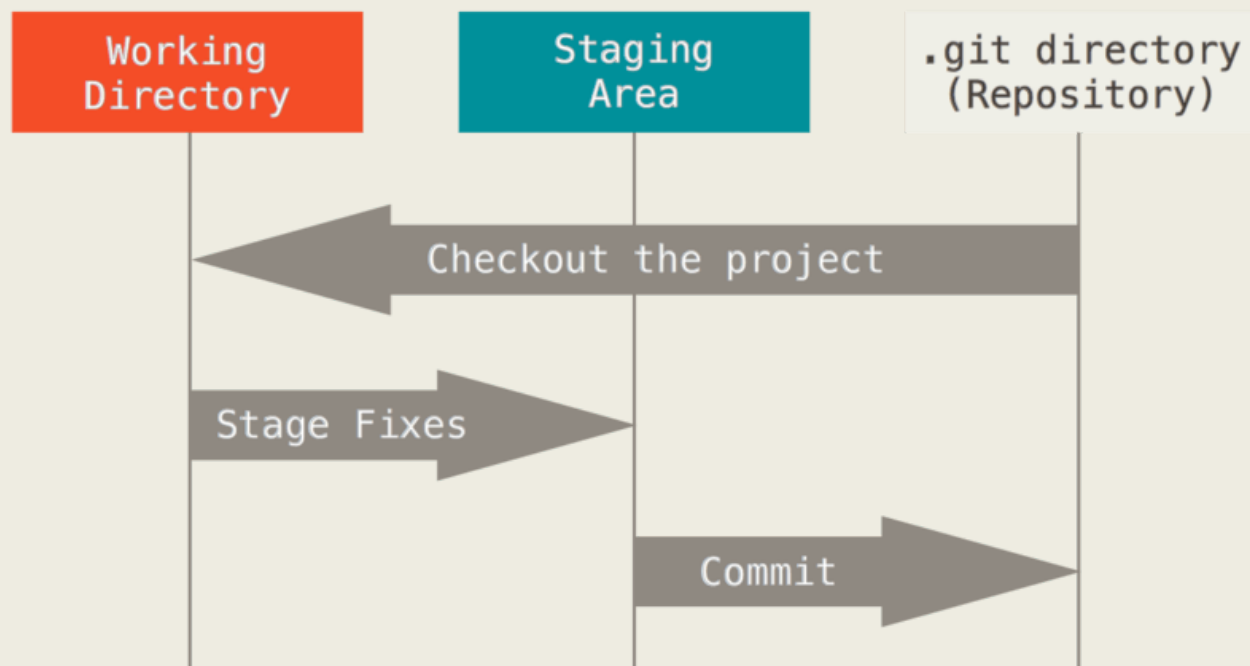
完整

Git 保证完整性，
SHA-1

添加

Git 一般只添
加数据。

Git的工作区域



工作目录、暂存区域以及 Git 仓库

The background of the slide features a large, semi-transparent circular logo of Tsinghua University. The logo contains a stylized building with five vertical bars of different colors (yellow, orange, red, purple, blue) in front of it. The text "TSINGHUA UNIVERSITY" is written in a circular path around the top, and "SINCE 2013" is written around the bottom, separated by two stars.

Git基本命令

基本命令 : start a working area

init

初始化一个空仓库

clone

将仓库克隆到本地

基本命令 : work on the current change

add

将文件添加到暂存区

reset

将文件从暂存区/仓库
恢复到工作目录

mv

移动文件

基本命令 : work on the current change

rm

删除文件

stash

保存工作目录

stash pop

弹出工作目录
(跨分支pop是把双刃剑)

基本命令 : grow, mark and tweak your common history

checkout

切换分支或者
恢复工作目录文件

diff

查看提交与提交之间
或者工作目录和提交
之间的不同

commit

将修改记录到仓库

基本命令 : grow, mark and tweak your common history

branch

查看、新建
或删除分支

merge

将多个提交历史
合并

rebase

变基

tag

查看、删除、创建
标签

基本命令 : examine the history and state

status

查看工作区状态

log

查看提交记录

show

查看提交历史

基本命令 : collaborate

fetch

从某个仓库下载提交

pull

从某个仓库下载提交
并合并进当前分支

push

把提交提交到
远程仓库

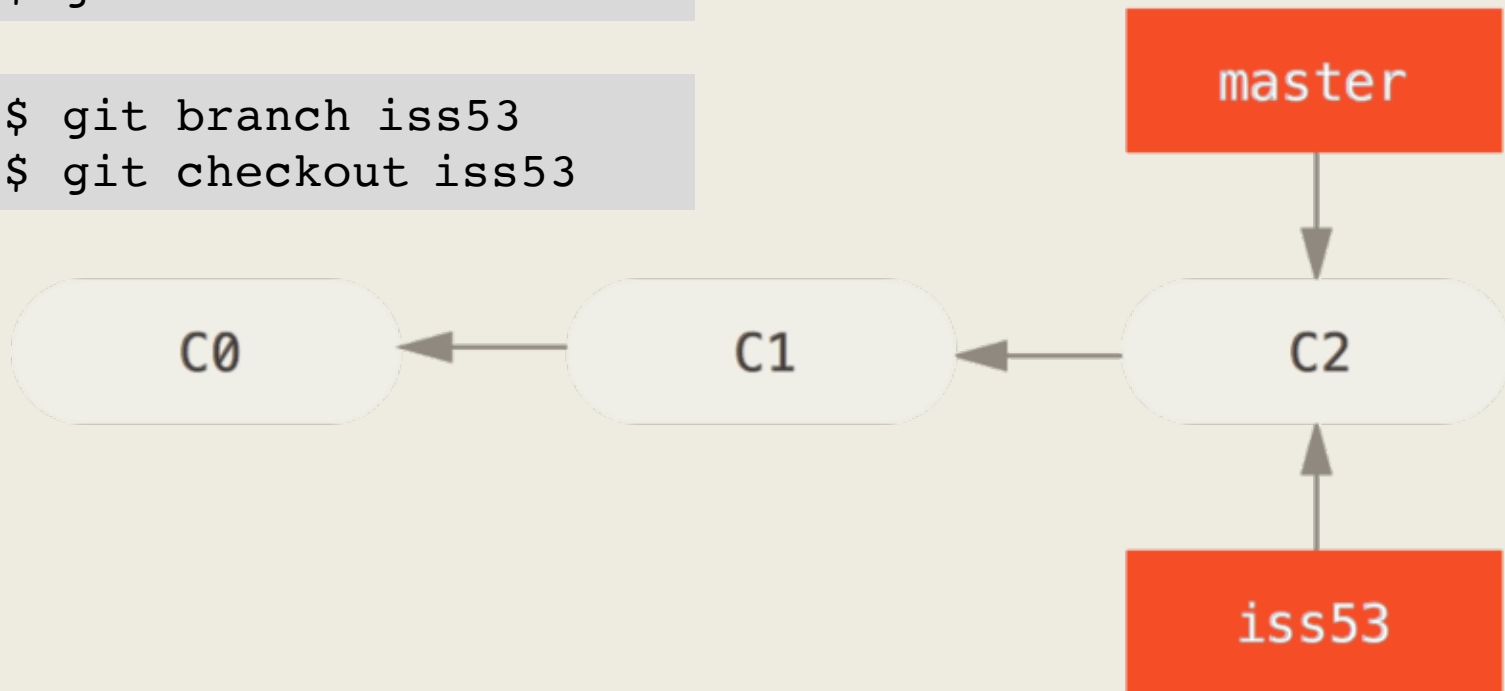


Git分支

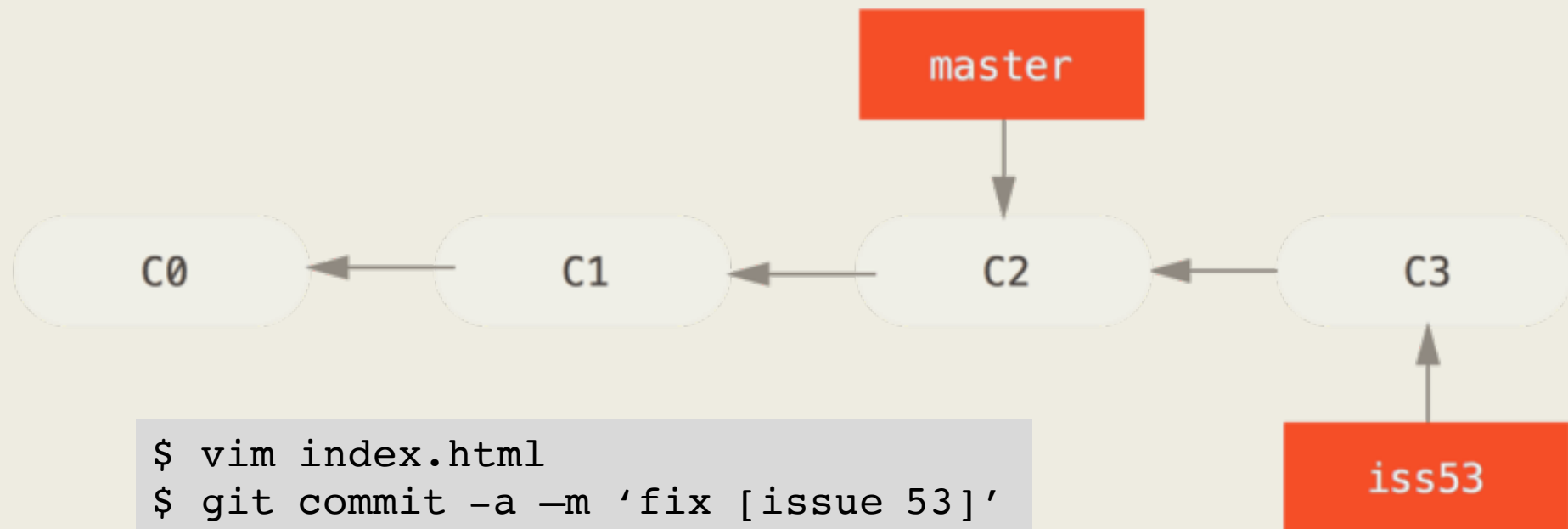
Git分支：创建分支

```
$ git checkout -b iss53
```

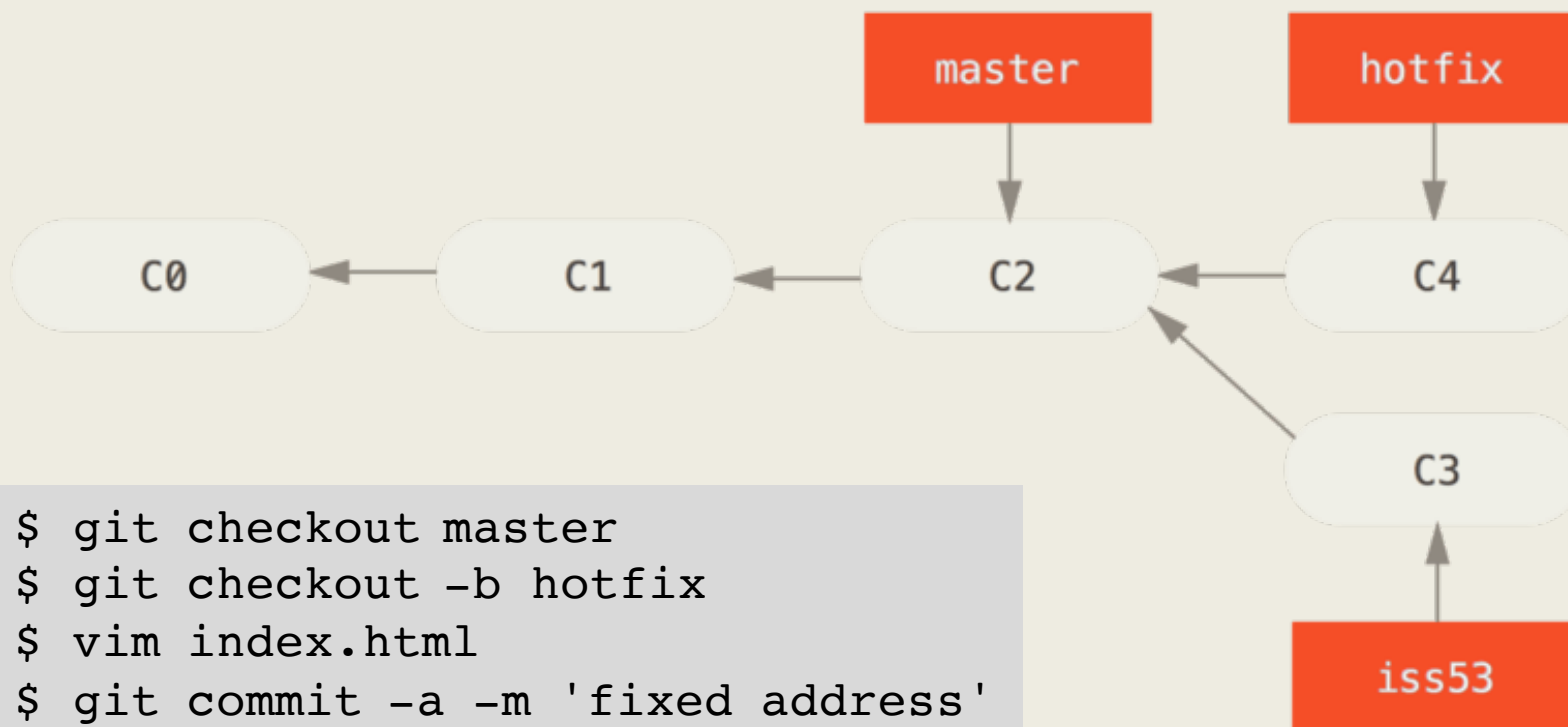
```
$ git branch iss53  
$ git checkout iss53
```



Git分支：切换分支

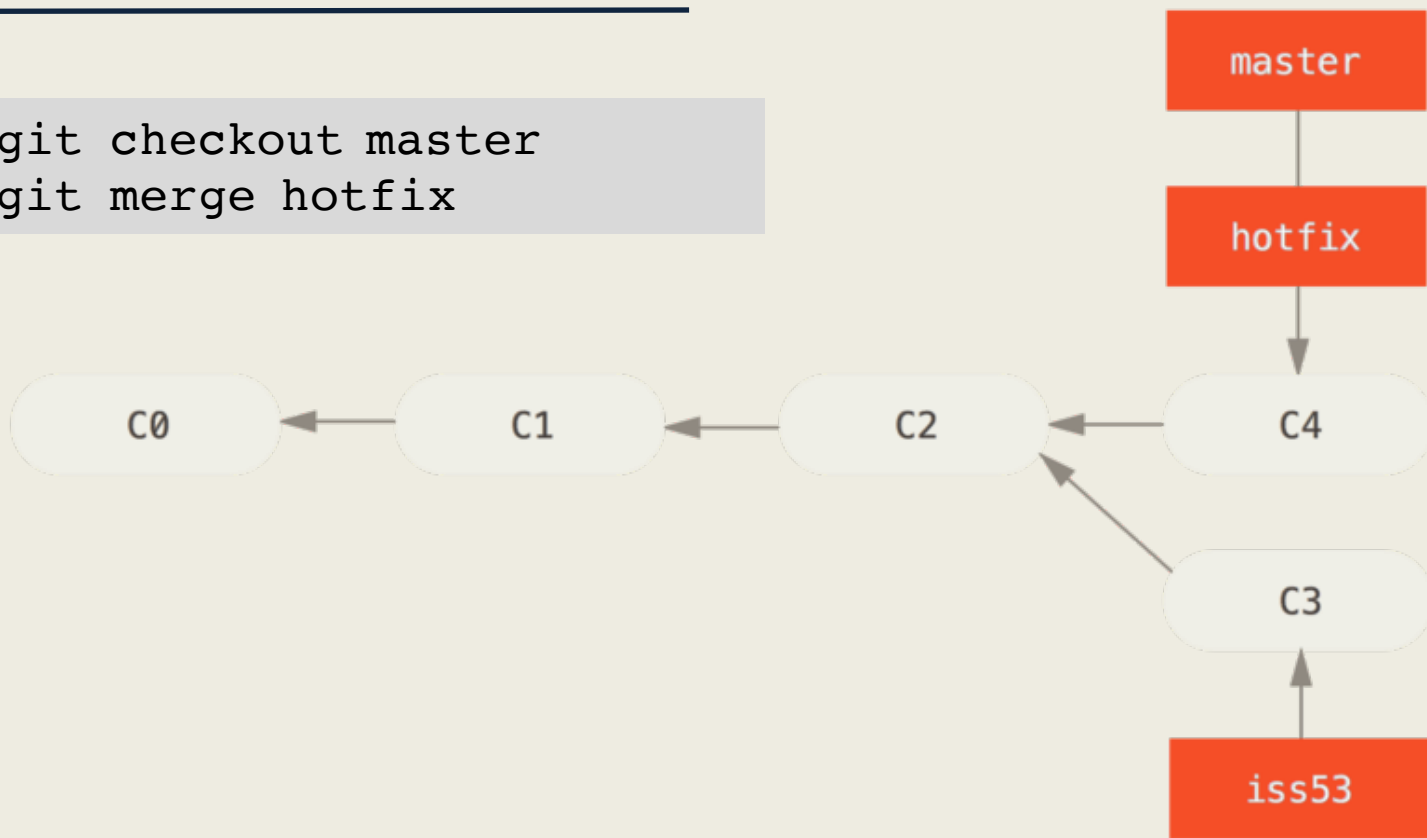


Git分支：切换分支

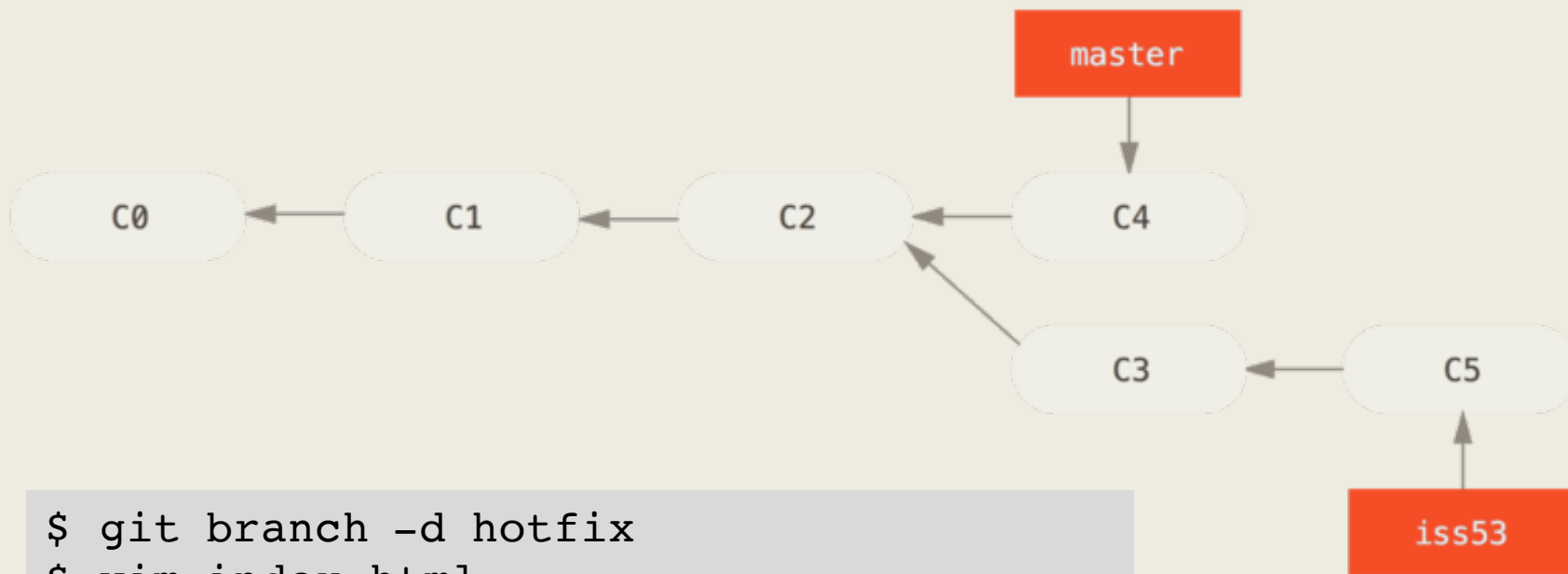


Git分支：合并分支

```
$ git checkout master  
$ git merge hotfix
```



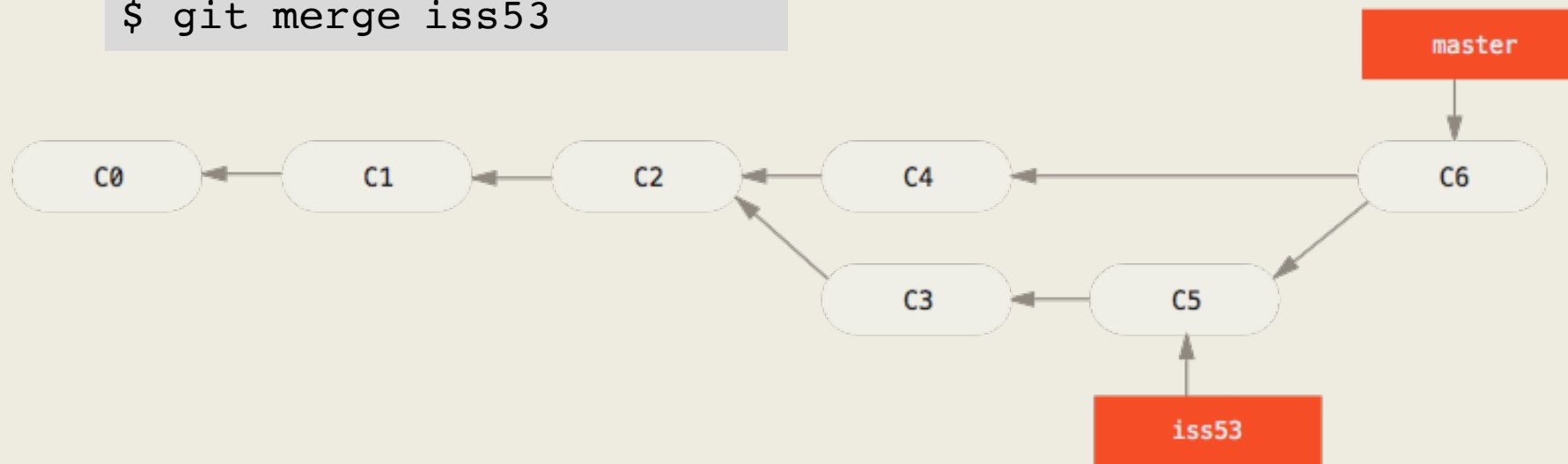
Git分支：合并分支



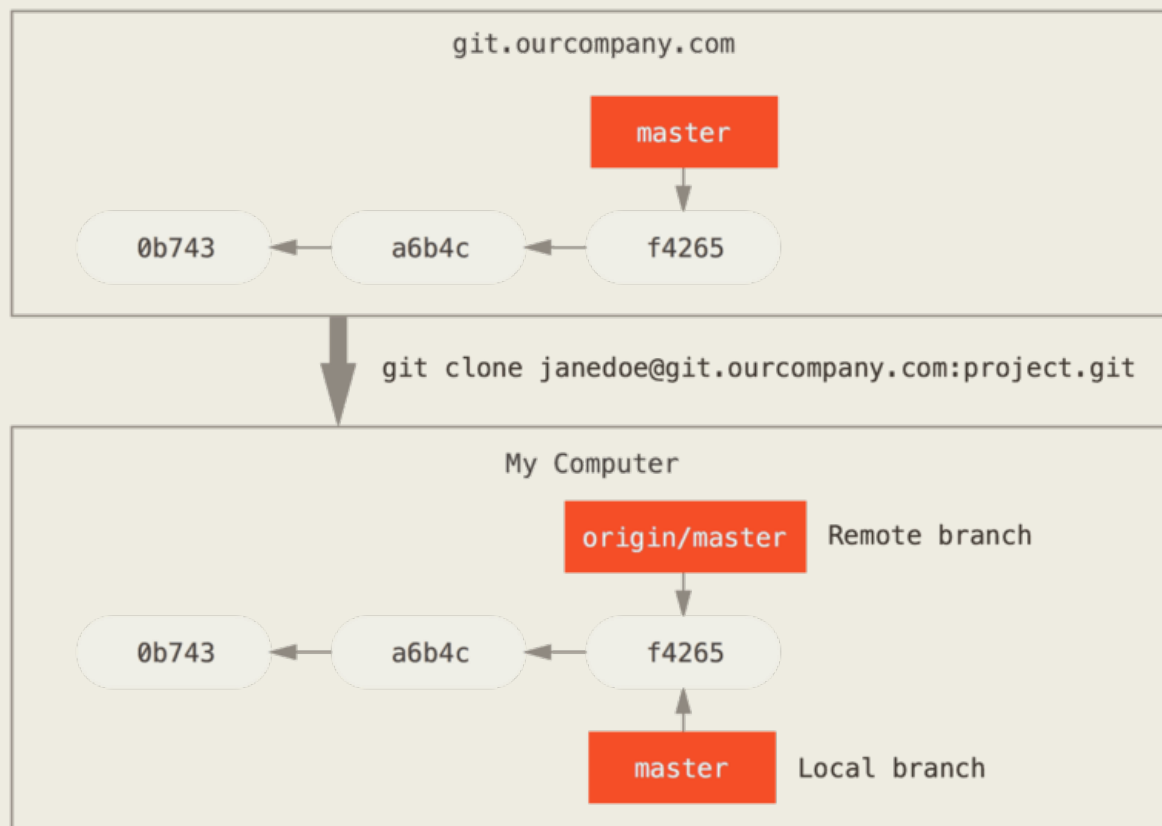
```
$ git branch -d hotfix  
$ vim index.html  
$ git commit -a -m 'finished [issue 53]'
```

Git分支：合并分支

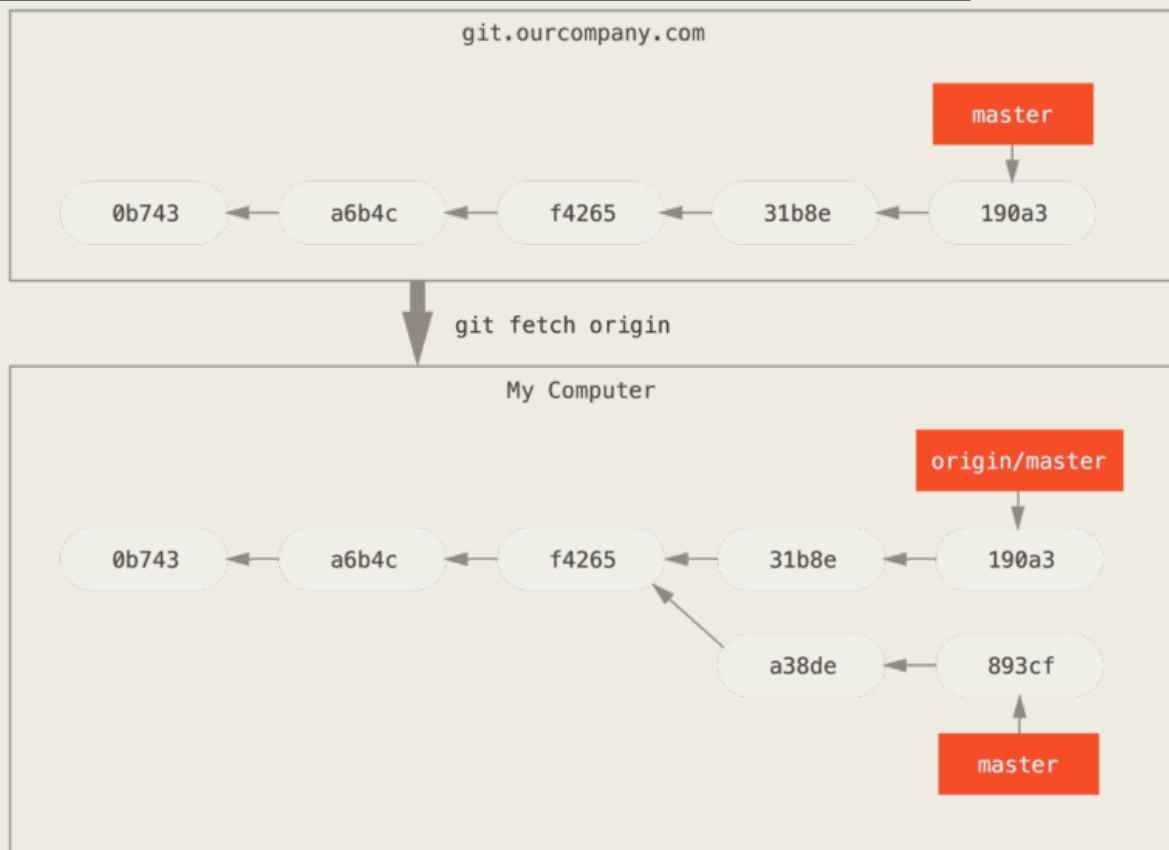
```
$ git checkout master  
$ git merge iss53
```



Git分支 : 远程分支(remote)/(branch)



Git分支 : 远程分支(remote)/(branch)



Git分支 : 遇到冲突时的分支合并

```
→ testgit git:(master) git pull
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/doublechen/testgit
   c73f285..bb74072  master      -> origin/master
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.
→ testgit git:(master) x □
```

Git分支：遇到冲突时的分支合并

```
11     <meta charset="utf-8">
12     <meta http-equiv="X-UA-Compatible" content="IE=11,IE=10,IE=
13 <<<<<< HEAD
14     <title>登录 - RUC</title>
15 =====
16     <title>登录 - BJTiUP团队</title>
17 >>>>>> bb74072b1498ecadf300b307fc90e305d42b6b37
18     <meta name="viewport" content="width=device-width, initial
19     <meta name="renderer" content="webkit">
20     <meta name="viewport" content="initial-scale=0.8">
21     <link rel="stylesheet" href="https://bjtiup.tiup.cn/static/
```

Git分支

: 遇到冲突时的分支合并

(1) 为什么会产生冲突

因为不同的开发者同时修改了同一文件的同一行

(2) 如何处理

人工将代码修改为最终正确的样子然后 `add commit push`

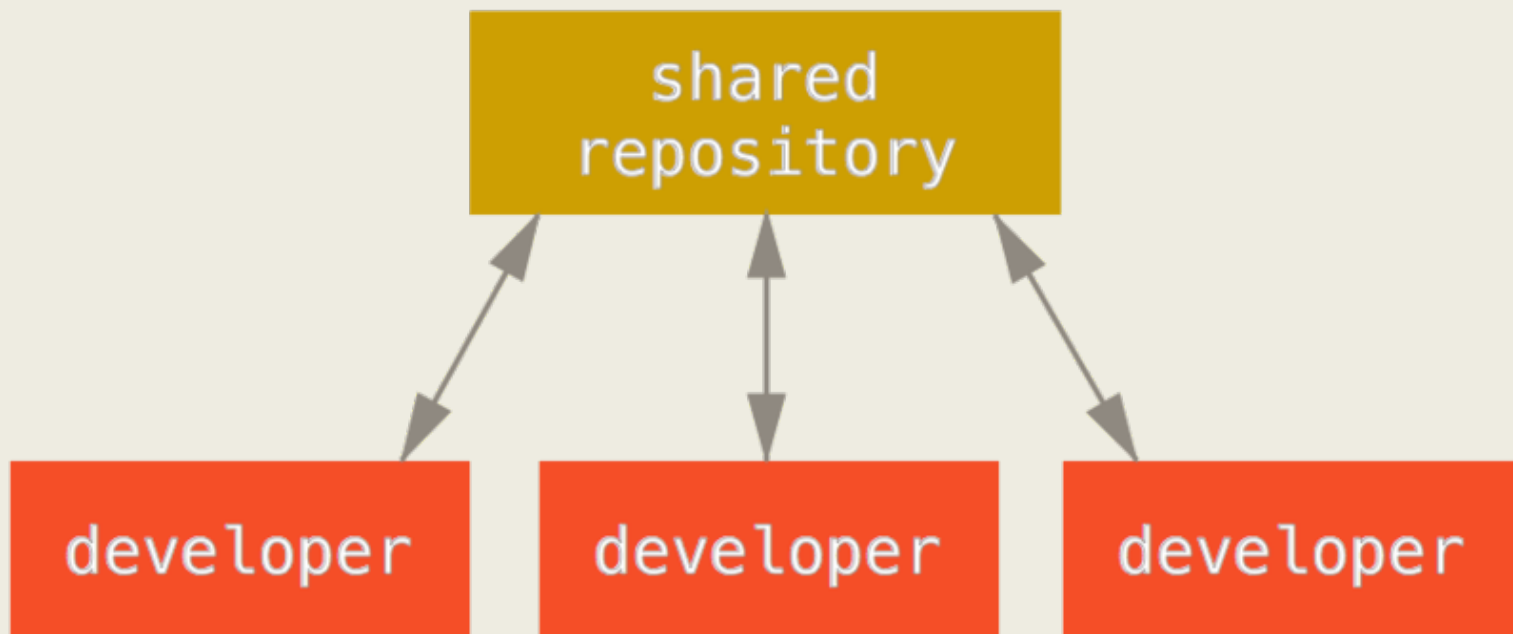


Git workflow

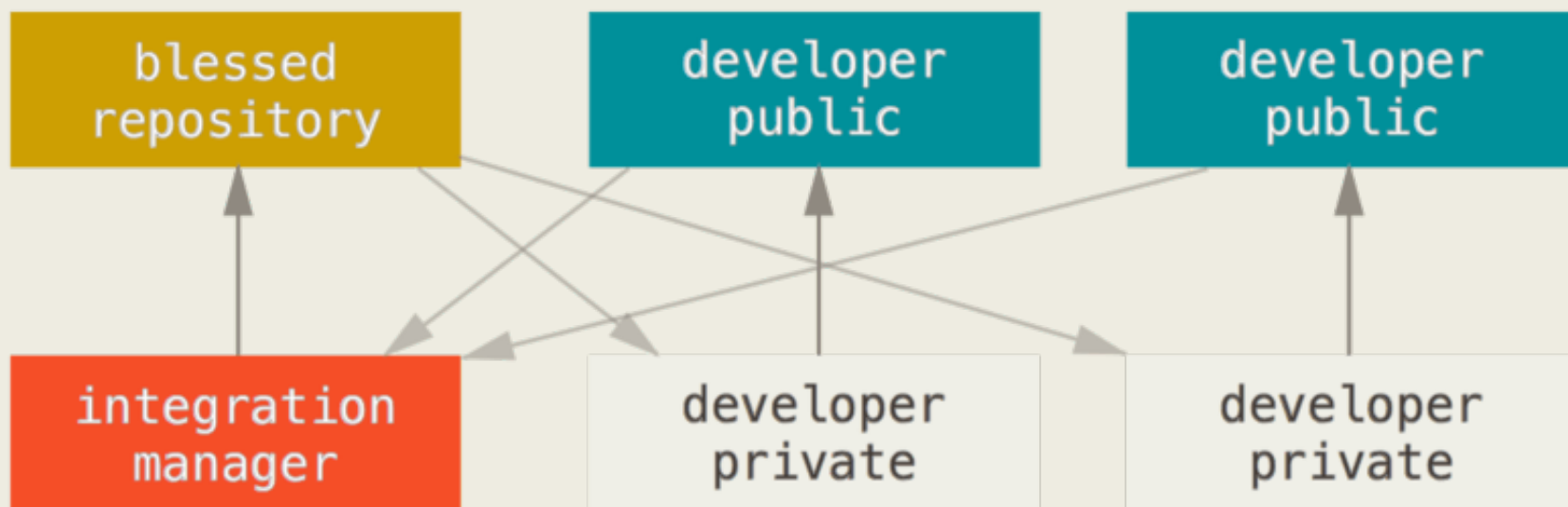
Git工作流：

- (1) 集中式工作流
- (2) 集成管理者工作流

Git工作流：集中式工作流



Git工作流：集成管理者工作流



贡献：提交准则

1、不要把空白错误（空白错误是指行尾的空格、Tab 制表符，和行首空格后跟 Tab 制表符的行为）提交上去。运行git diff --check

```
bash
lib/simplegit.rb:5: trailing whitespace.
+   @git_dir = File.expand_path(git_dir)
lib/simplegit.rb:7: trailing whitespace.
+
lib/simplegit.rb:20: trailing whitespace.
+end
(END) git config --global core.whitespace cr-at-eol

diff --git a/src/apps/notice/Lib/Service/UserNoticeService.class.php b/src/apps/notice/Lib/Service/UserNoticeService.class.php
index aa8c9c3..66ded02 100755
--- a/src/apps/notice/Lib/Service/UserNoticeService.class.php
+++ b/src/apps/notice/Lib/Service/UserNoticeService.class.php
@@ -1594,7 +1594,7 @@ class UserNoticeService{
     $model = MC::$user_notice_index';
     $sql = "select mid,istop from s_user_notice_index where status=2 and isdraft=0 and schoolid = ".SCHOOL_ID."";
+    $sql = "select mid,istop from s_user_notice_index where status=2 and isdraft=0 and schoolid = ".SCHOOL_ID."";

```


贡献

: 提交准则

2、尝试让每一个提交成为一个逻辑上的独立变更集

这可以使同事开发者必须审查你的改动时尽量让事情容易些。当你之后需要时这个方法也会使拉出或还原一个变更集更容易些。

3、有一个创建优质提交信息的习惯会使 Git 的使用与协作容易的多

一般情况下，信息应当以少于 50 个字符（25个汉字）的单行开始且简要地描述变更，接着是一个空白行，再接着是一个更详细的解释（包括做改动的动机和它的实现与之前行为的对比）。

贡献：提交准则

修改的摘要（50 个字符或更少）

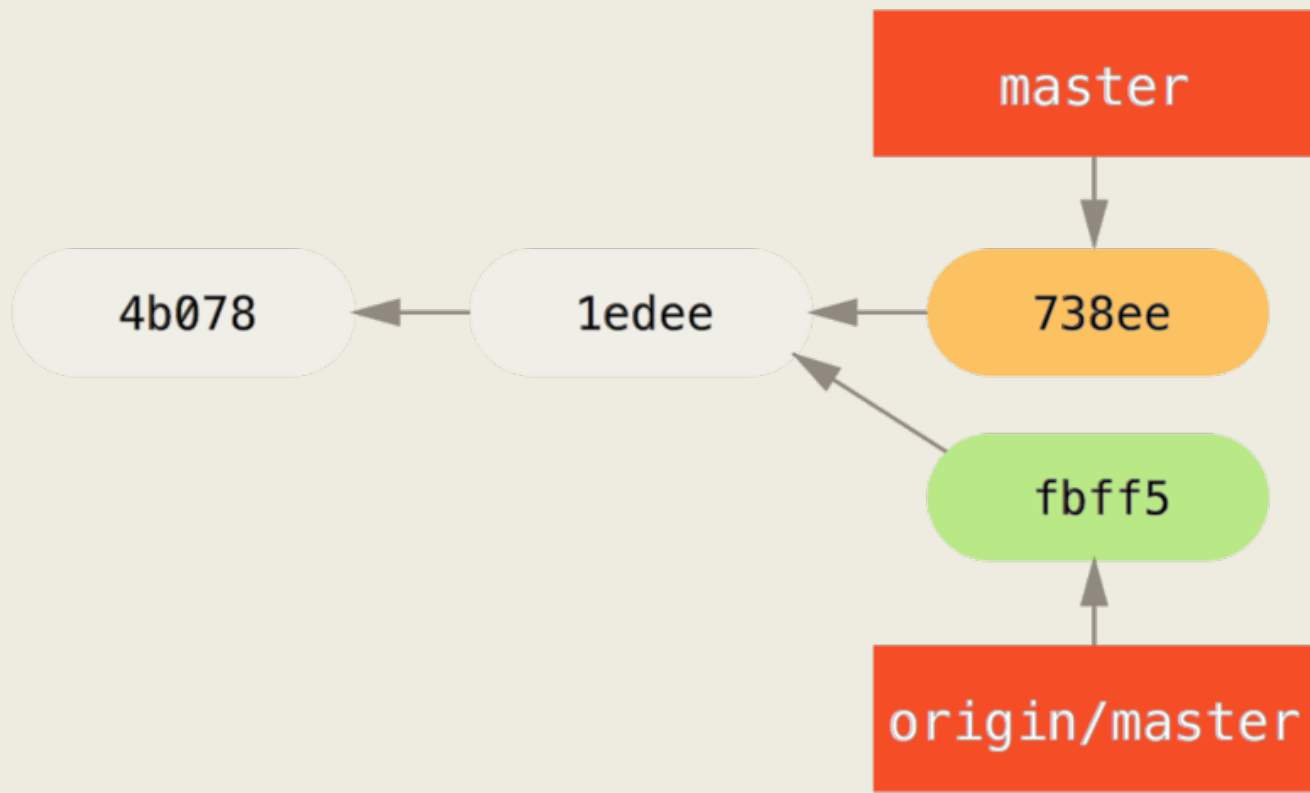
如果必要的话，加入更详细的解释文字。在大概 72 个字符的时候换行。在某些情形下，第一行被当作一封电子邮件的标题，剩下的文本作为正文。分隔摘要与正文的空行是必须的（除非你完全省略正文）；如果你将两者混在一起，那么类似变基等工具无法正常工作。

空行接着更进一步的段落。

- 句号也是可以的。
- 项目符号可以使用典型的连字符或星号前面一个空格，之间用空行隔开，但是可以依据不同的惯例有所不同。

贡献

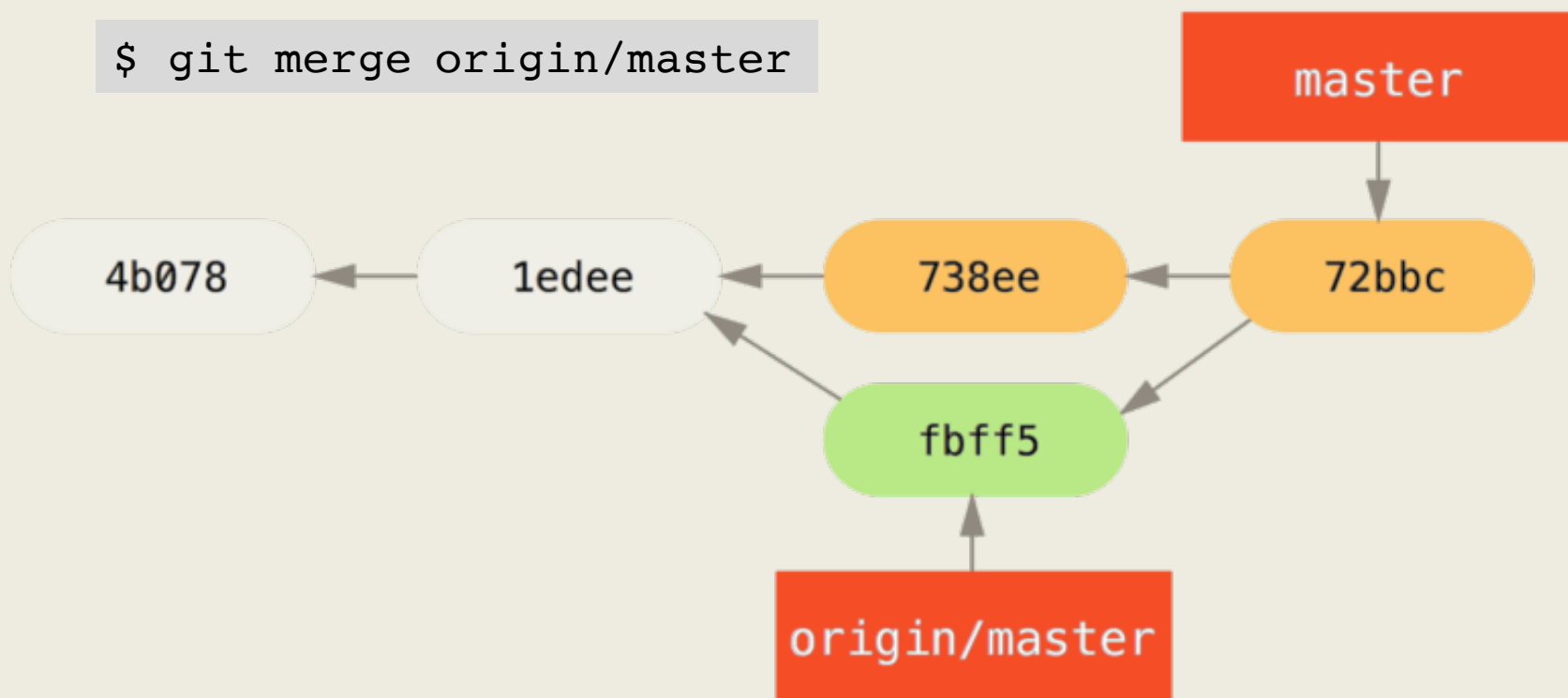
: 私有小型团队 John



贡献

: 私有小型团队 John

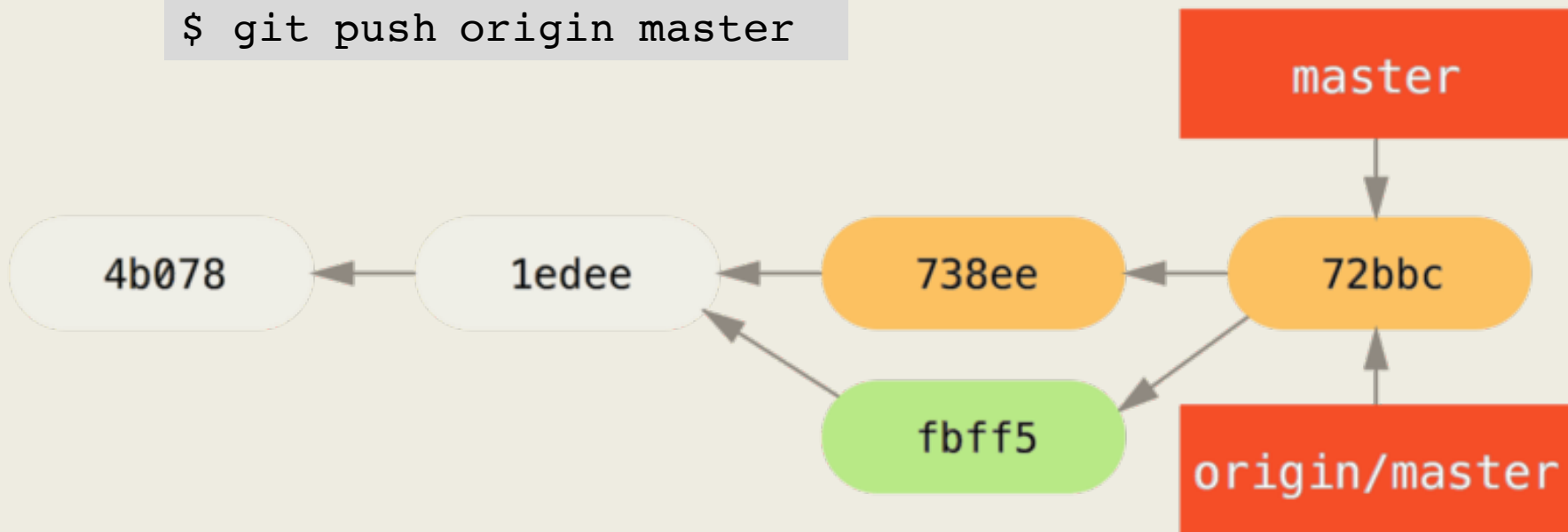
```
$ git merge origin/master
```



贡献

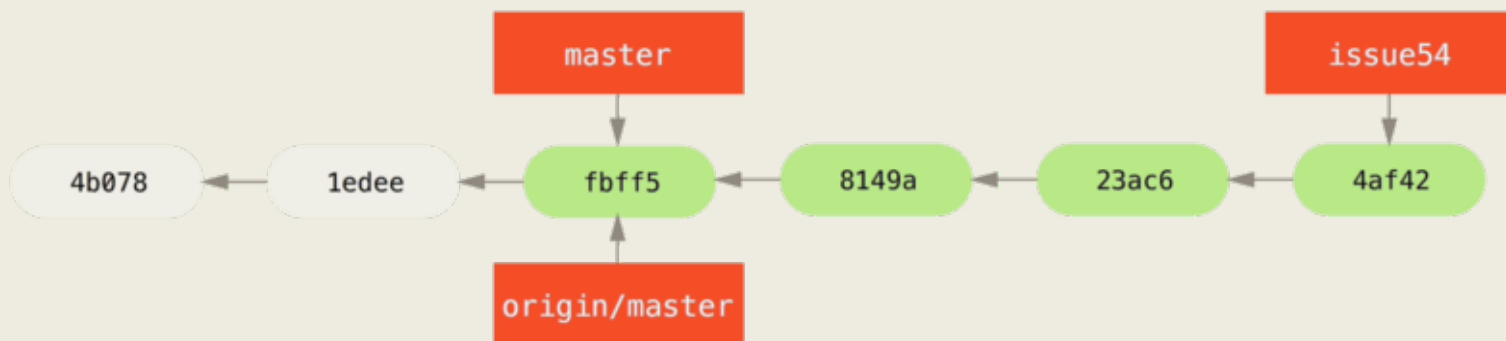
: 私有小型团队 John

```
$ git push origin master
```



贡献

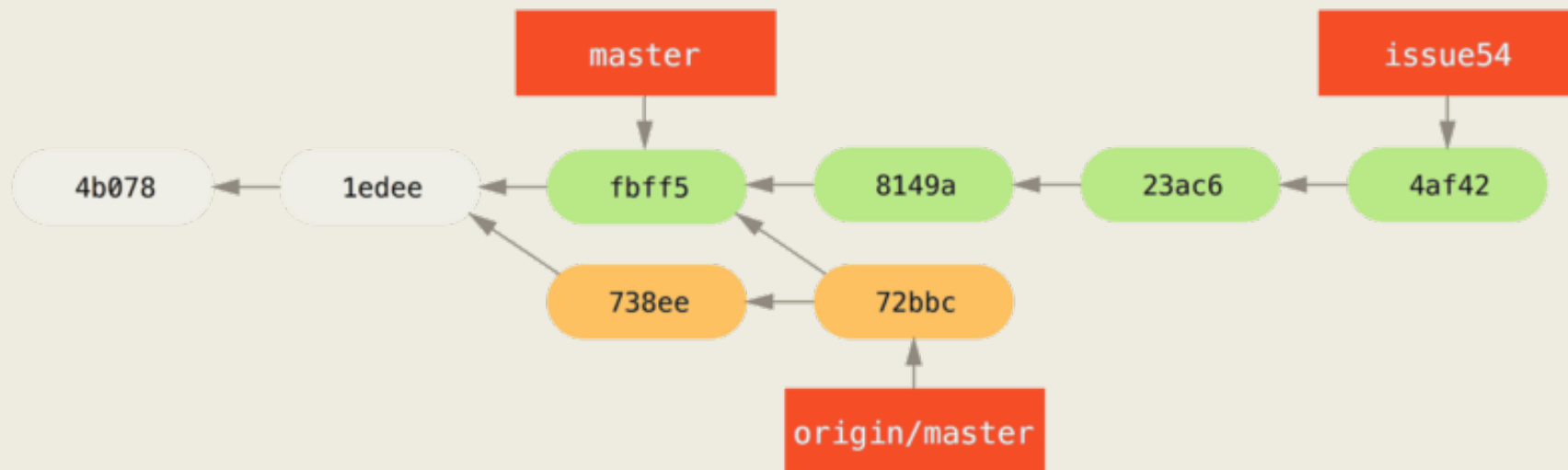
: 私有小型团队 Jessica



贡献

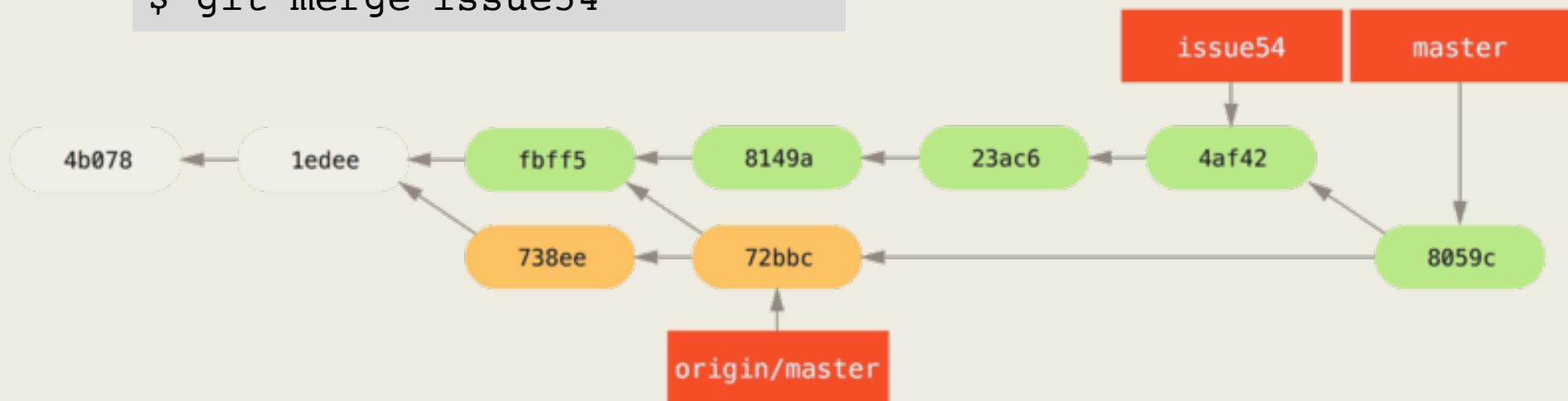
: 私有小型团队 Jessica

```
$ git fetch origin
```



贡献：私有小型团队 Jessica

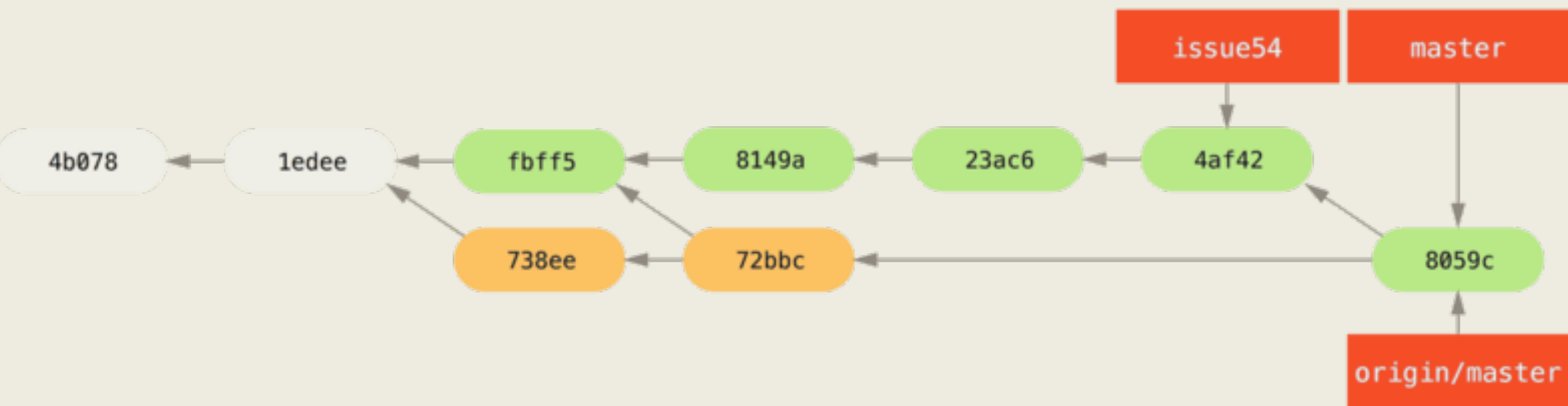
```
$ git checkout master  
$ git merge origin/master  
$ git merge issue54
```



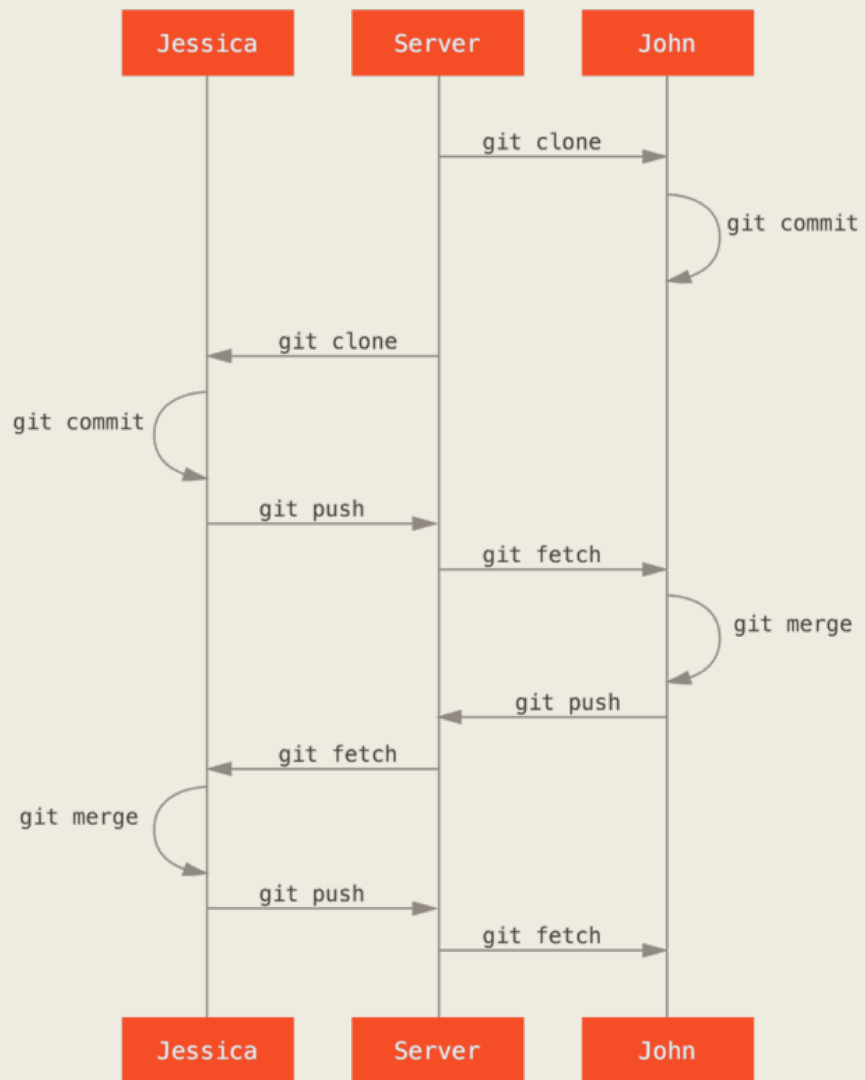
贡献

: 私有小型团队 Jessica

```
$ git push origin master
```

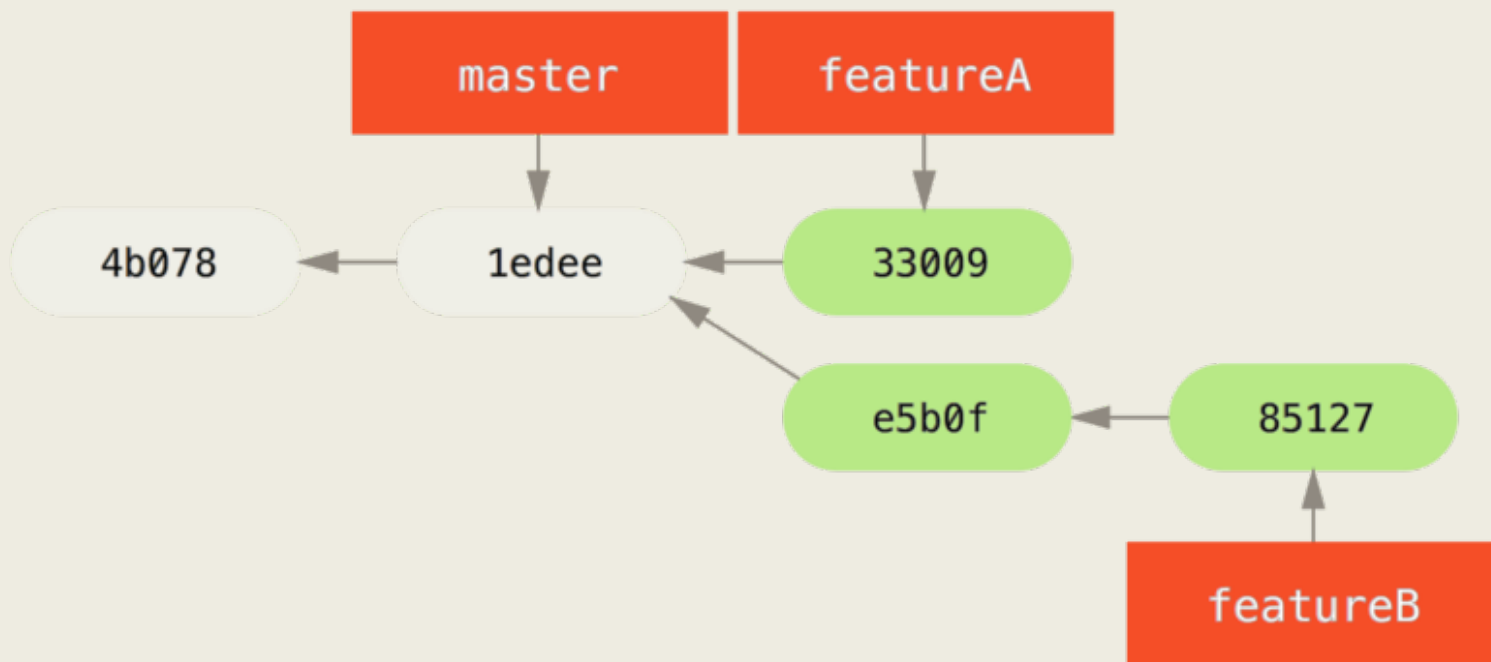


贡献：私有小型团队

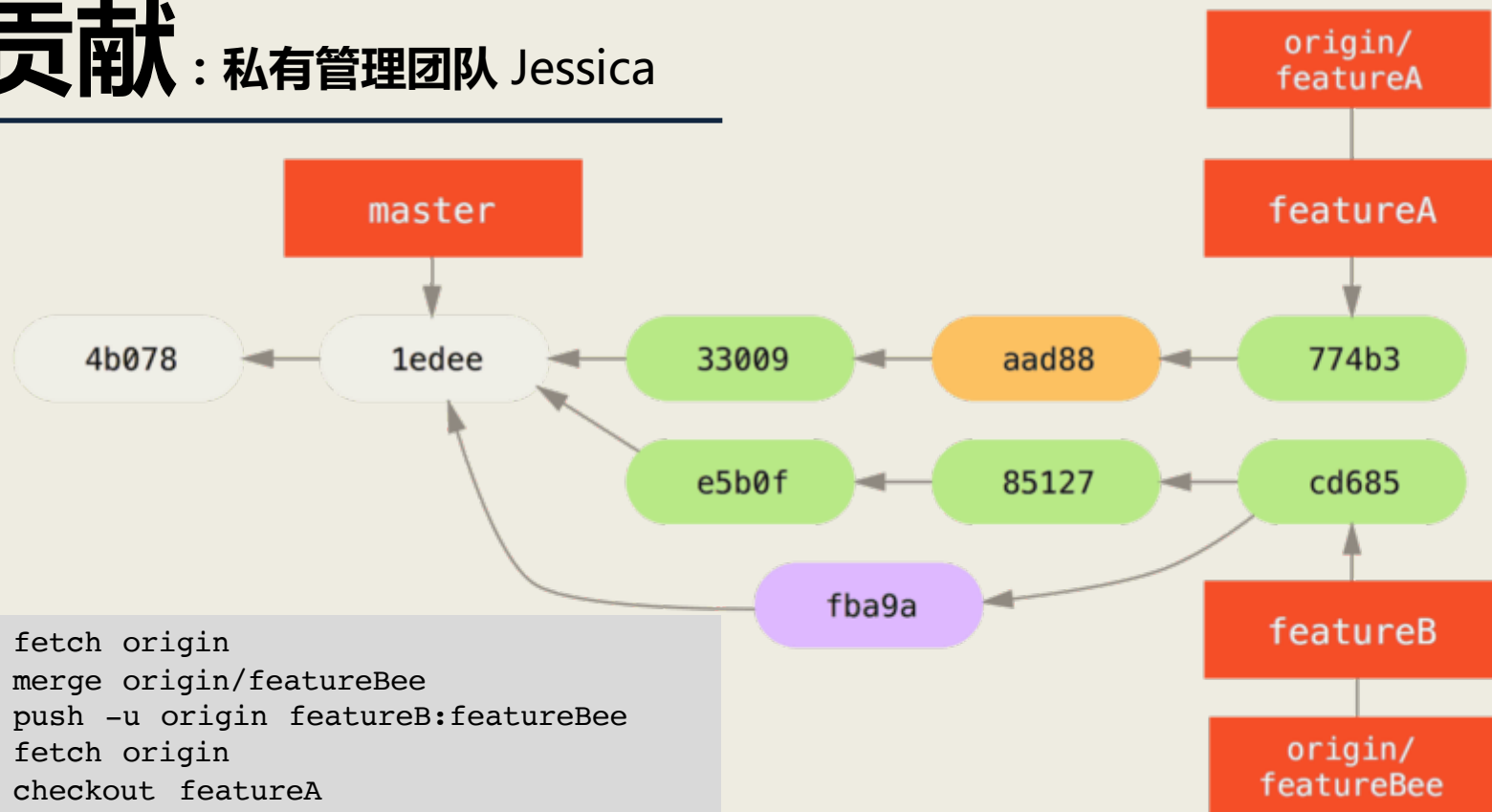


贡献

: 私有管理团队 Jessica

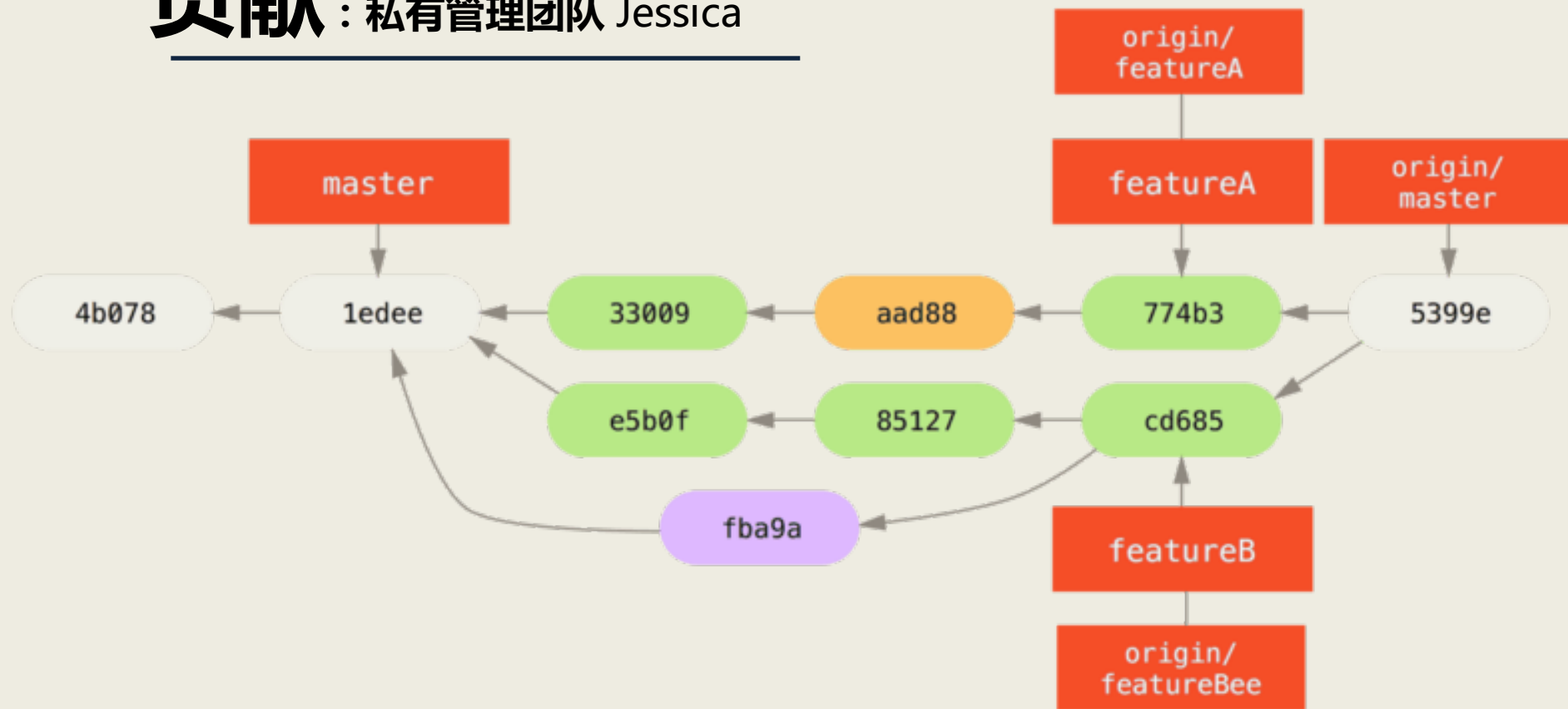


贡献：私有管理团队 Jessica

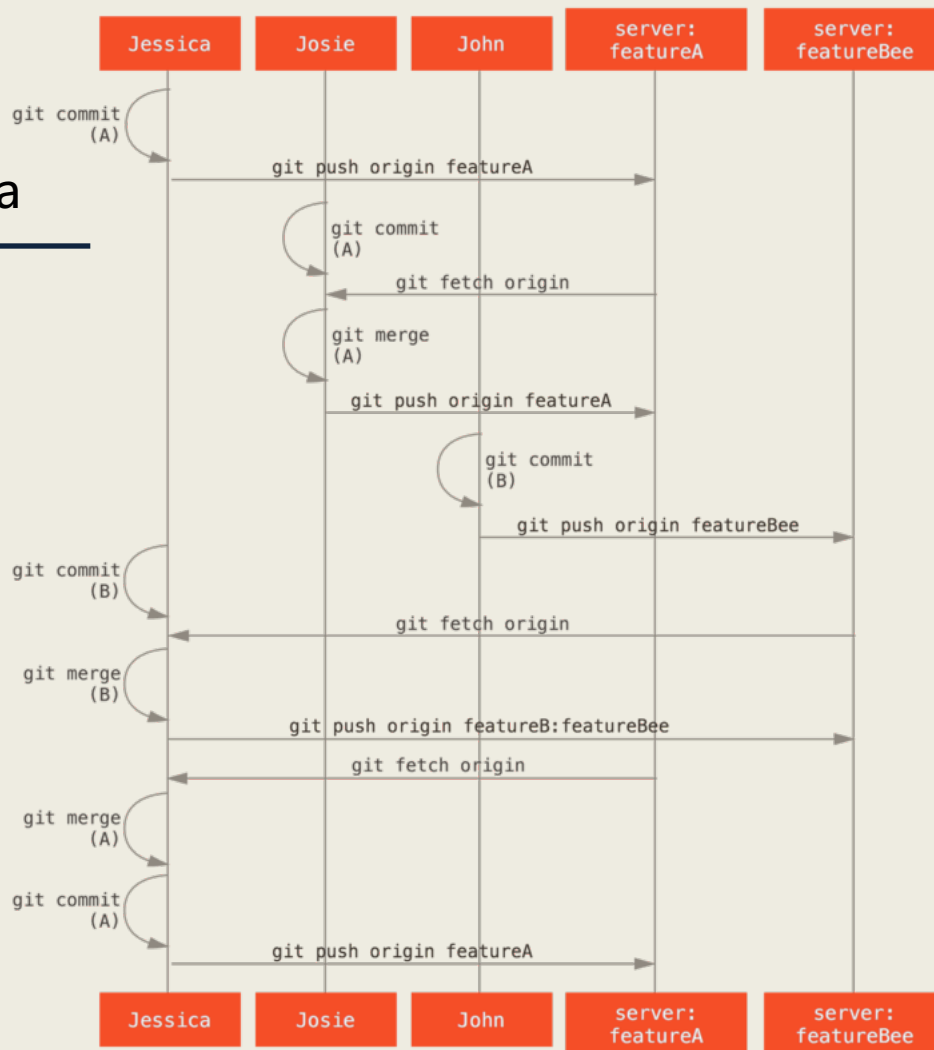


```
$ git fetch origin
$ git merge origin/featureBee
$ git push -u origin featureB:featureBee
$ git fetch origin
$ git checkout featureA
$ git merge origin/featureA
$ git commit -am 'small tweak'
$ git push
```

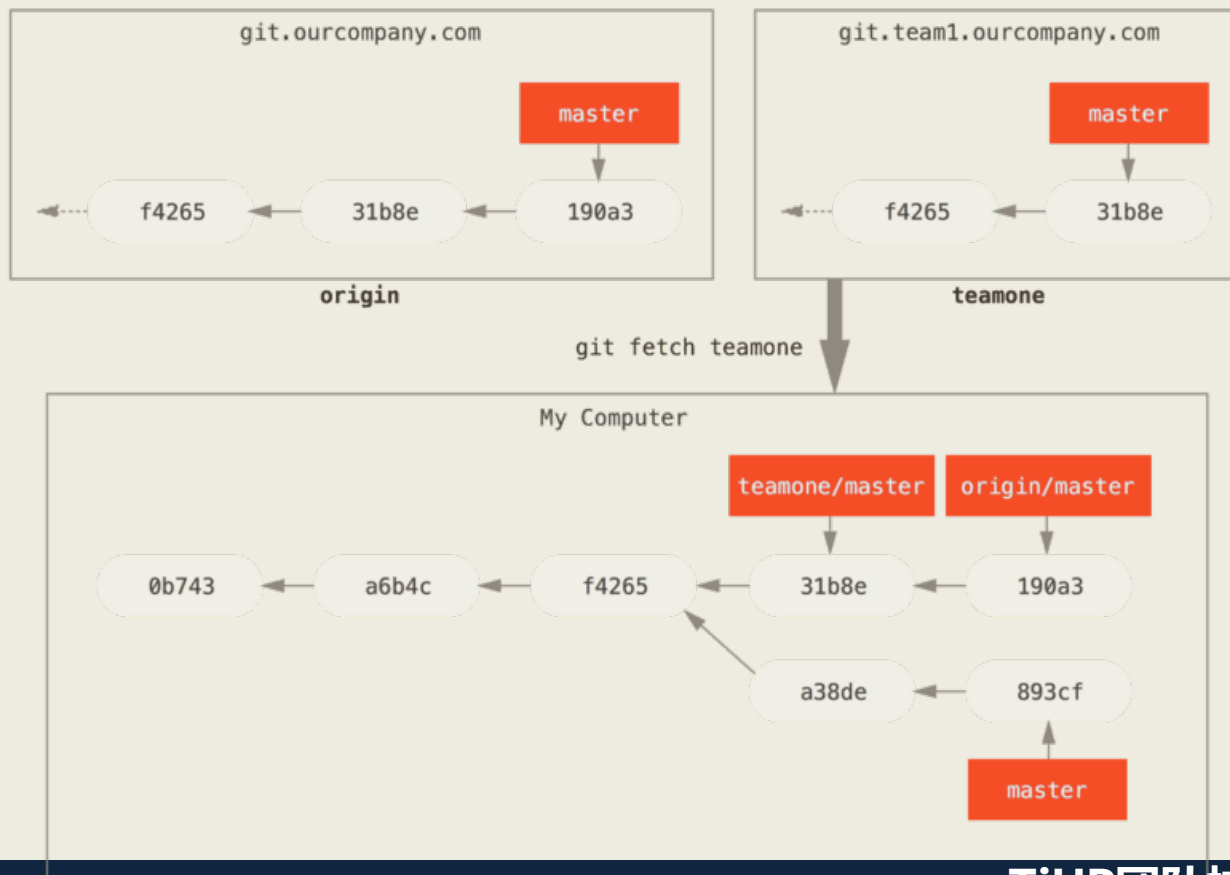
贡献：私有管理团队 Jessica



贡献：私有管理团队 Jessica



贡献：派生的公开项目





Tips

Tips : 修改代码前先pull 可能可以避免部分冲突

```
→ tiUP git:(tiupng) git pull
Updating cdc0434..654d8d0
Fast-forward
 src/apps/position/Lib/Service/UserApplyService.class.php      | 13 ++++++-----
 src/apps/schoolwidget/Lib/Action/BfsuCardAction.class.php     | 22 ++++++++-----
 src/apps/servcenter/Lib/Action/Info/TableAction.class.php     | 1 +
 src/apps/servcenter/Lib/Service/ServExportService.class.php   | 5 +++--
 src/apps/servcenter/Lib/Service/ServTblService.class.php      | 10 ++++++----
 src/core/ThinkPHP/Extend/Service/LoginService.class.php       | 4 +++-
 .../exceltemplate/235/02-分会各系列各级专业技术岗位推荐人员一览表.xlsx | Bin 10346 -> 10342 bytes
 src/static/sample/gangweizhicheng/7.3、同行专家鉴定意见.docx | Bin 20041 -> 21163 bytes
 src/tiup.yml.sample                                           | 2 ++
 9 files changed, 33 insertions(+), 24 deletions(-)
```

Tips : 随手add , 完成提交

```
→ salon git:(master) X gst
```

```
On branch master
```

```
Your branch is up-to-date with 'origin/master'.
```

```
Changes to be committed:
```

```
(use "git reset HEAD <file>..." to unstage)
```

```
    modified:   Git的基本命令和工作流简介.pptx
```

```
Changes not staged for commit:
```

```
(use "git add <file>..." to update what will be committed)
```

```
(use "git checkout -- <file>..." to discard changes in working directory)
```

```
    modified:   Git的基本命令和工作流简介.pptx
```

Tips : commit前diff一下

```
diff --git a/src/apps/notice/Lib/Service/UserNoticeService.class.php b/src/apps/notice/Lib/Service/UserNoticeService.class.php
index aa8c9c3..d61752f 100755
--- a/src/apps/notice/Lib/Service/UserNoticeService.class.php
+++ b/src/apps/notice/Lib/Service/UserNoticeService.class.php
@@ -1594,12 +1594,13 @@ class UserNoticeService{

    $model = M('s_user_notice_index');

-    $sql = "select nid,istop from s_user_notice_index where status=2 and isdraft=0 and schoolid = ".SCHOOL_ID.";
+    $sql = "select nid,istop from s_user_notice_index where status=2 and isdraft=0 and schoolid = ".SCHOOL_ID." ";

    if(strlen($type) > 0){
        $sql .= " and type like '%".$type."%";
    }

+    if(is_array($nids) && $nids){
        $sql .= 'and nid in (';
        foreach ($nids as $nid) {
@@ -1622,7 +1623,7 @@ class UserNoticeService{
    }

    $sql .= "(`range` = 5 and uids like '%".$uid."%') or `range` = 1";
-    $sql .= ")";
+    $sql .= ")";z

    $sql .= " order by ".$order;
    $sqlcount = str_replace('nid', 'count(*)', $sql.");");
~
```

Tips : 编写规范的commit信息 git log --no-merges --grep "up"

```
commit c340a32103c1808024da7833bc71010bf4fa416a
Author: Your Name <you@example.com>
Date: Thu Dec 8 10:13:59 2016 +0800
```

up

```
commit f16337fd245038c5641968c620544dcdbdc6d3381
Author: Your Name <you@example.com>
Date: Tue Nov 22 14:12:11 2016 +0800
```

up

```
commit f6d47d31b38d97902f644c802abb8bedeb7fa1db
Author: Your Name <you@example.com>
Date: Tue Nov 22 14:06:24 2016 +0800
```

update

```
commit 94a0ad1bd9ae6546aa9d99e8064899461bf38a29
Author: Your Name <you@example.com>
Date: Tue Nov 22 14:02:12 2016 +0800
```

update;

```
commit 62574157a2c1fddc2d7029bc990062d4a2bf48f5
Author: 1111@tiup.us>
Date: Thu May 5 14:48:21 2016 +0800
```

update

```
commit cf9848ddcccebd728bef2ee6427dacb4a01166c
Author: 1111@tiup.us>
Date: Wed May 4 17:44:46 2016 +0800
```

up

```
commit a78de789804d92885b29397091172440883cdac8
Author: 1111jp@tiup.us>
Date: Wed May 4 14:07:41 2016 +0800
```

update

```
commit 32c9053829e1fd91691c7d665855a7382867f4eb
Author: 1111@tiup.us>
Date: Thu Apr 21 11:00:28 2016 +0800
```

up

启用 .git/hooks/commit-msg

Tips : 关注 push 和 pull 的结果

```
→ testgit git:(master) git push
To https://github.com/doublechen/testgit.git
 ! [rejected]          master -> master (fetch first)
error: failed to push some refs to 'https://github.com/doublechen/testgit.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

Tips : 合并遇到冲突一定要解决 git show

```
11     <meta charset="utf-8">
12     <meta http-equiv="X-UA-Compatible" content="IE=11,IE=10,IE=
13 <<<<<< HEAD
14     <title>登录 - RUC</title>
15 =====
16     <title>登录 - BJTiUP团队</title>
17 >>>>>> bb74072b1498ecadf300b307fc90e305d42b6b37
18     <meta name="viewport" content="width=device-width, initial
19     <meta name="renderer" content="webkit">
20     <meta name="viewport" content="initial-scale=0.8">
21     <link rel="stylesheet" href="https://bjtiup.tiup.cn/static/
```

Tips : 不要在工作中实战你不确定会带来什么结果的命令



Ref

(1) <https://git-scm.com/book/zh/v2/>

(2) RUN “git help” OR “git help -a”



谢谢