

# Git branching

Jim Hester  
RStudio

 @jimhester

 @jimhester\_

<http://happygitwithr.com/branches.html>

Branching means that you take a detour from the main stream of development and do work without changing the main stream

# Why Branch?

# How do you create a branch?

```
git branch issue-5
```

```
git checkout issue-5
```

```
git checkout -b issue-5
```

# How do you change branches?

`git checkout issue-5`

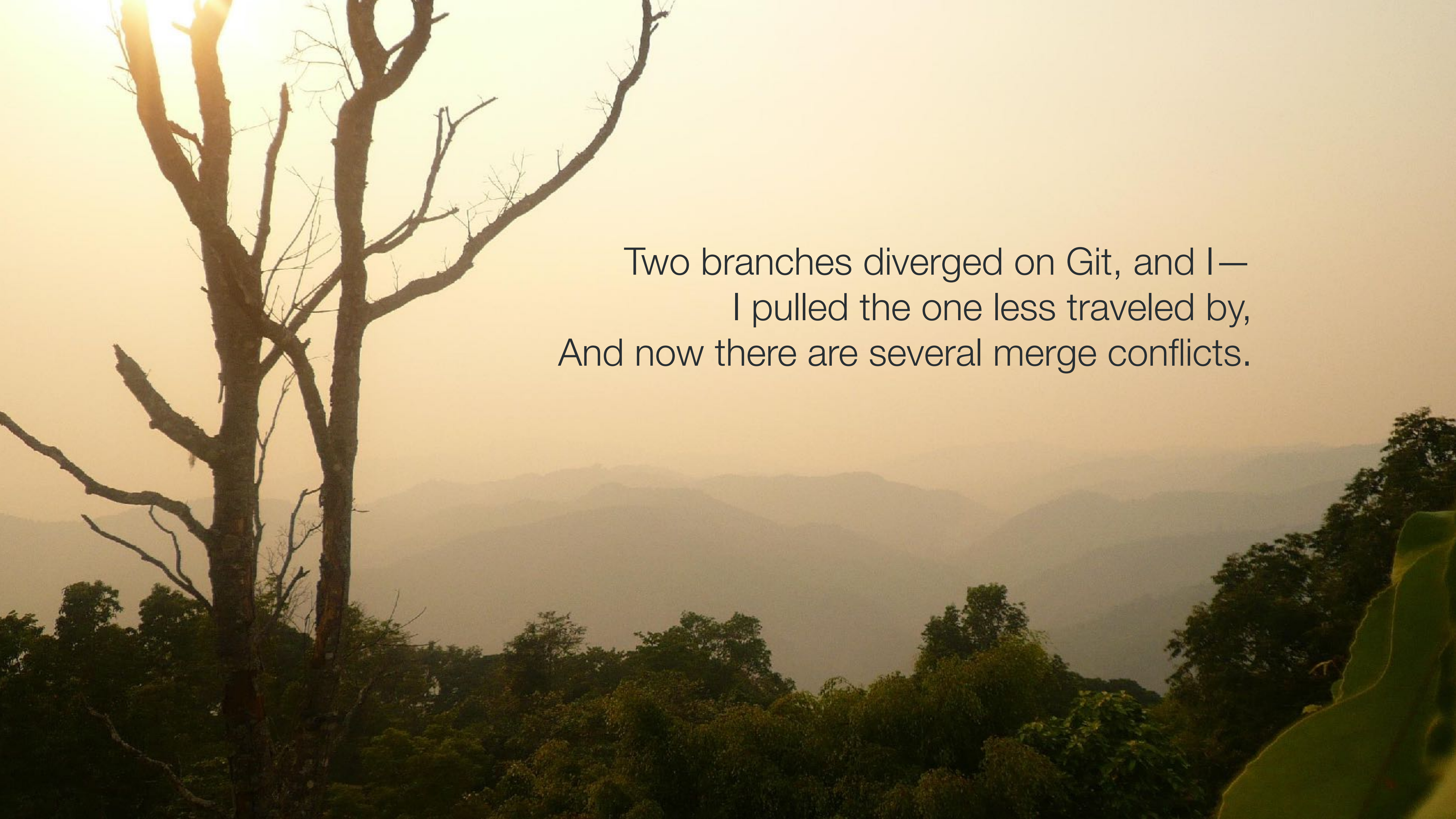
Git won't let me switch?

```
git commit --all -m WIP
```

```
git checkout xyz
```

```
git reset HEAD^1
```





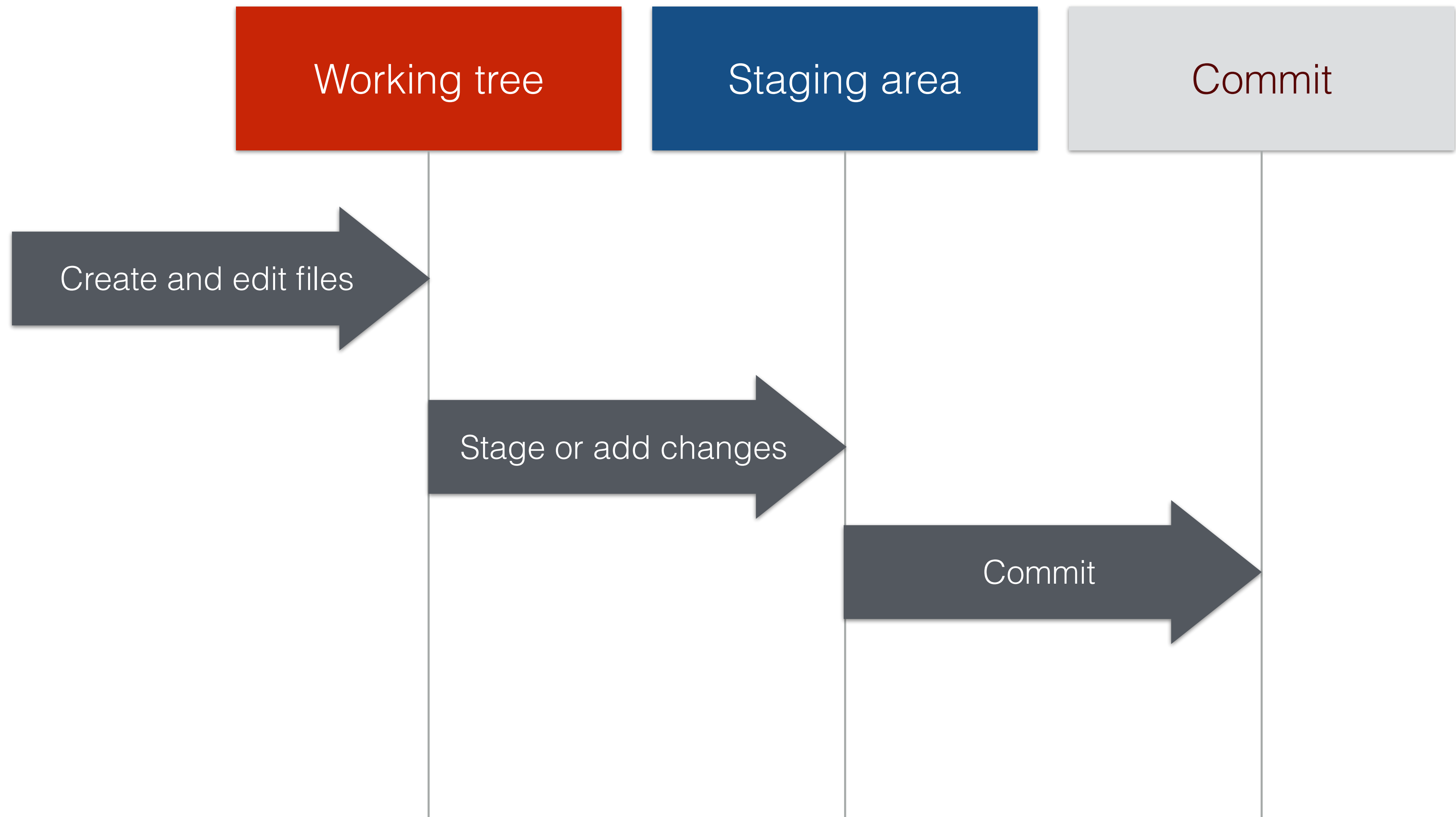
Two branches diverged on Git, and I—  
I pulled the one less traveled by,  
And now there are several merge conflicts.

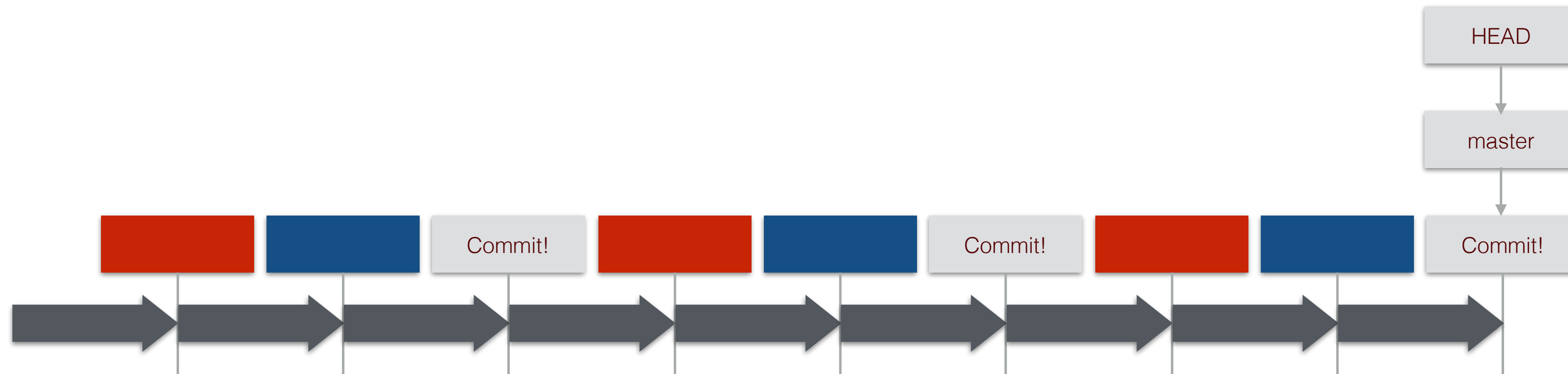


Two branches diverged on Git, and I—  
I pulled the one less traveled by,  
And now there are several merge conflicts.

<https://twitter.com/marktimemedia/status/469462640314421248>

Michelle Schulp - <http://mynameismichelle.com/git-frost/>





do stuff  
stage it  
commit it

lather, rinse, repeat

**Working directory**  
**Working tree**

current state of filesystem

**Index**  
**Staging area**

Proposed next commit  
snapshot

**HEAD**

Last commit snapshot,  
parent of next commit



# “do stuff”

Working tree

Two branches diverged on Git, and I—

Staging area

Commit

“stage it”

`git add xyz`

Working tree

Two branches diverged on Git, and I—

Staging area

Two branches diverged on Git, and I—

Commit

“commit it”

git commit

Working tree

Two branches diverged on Git, and I—

Staging area

Two branches diverged on Git, and I—

Commit

Two branches diverged on Git, and I—

# “do (more) stuff”

Working tree

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Staging area

Two branches diverged on Git, and I—

Commit

Two branches diverged on Git, and I—



# “stage it”

# git add abc

Working tree

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Staging area

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Commit

Two branches diverged on Git, and I—

# “commit it”      git commit

Working tree

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Staging area

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Commit

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

“undo it”      `git reset HEAD^1`

Working tree

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Staging area

Two branches diverged on Git, and I—

Commit

Two branches diverged on Git, and I—

# “commit it”      git commit

Working tree

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Staging area

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Commit

Two branches diverged on Git, and I—  
I pulled the one less traveled by,



# “soft undo” `git reset --soft HEAD^1`

Working tree

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Staging area

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Commit

Two branches diverged on Git, and I—

# “commit it”      git commit

Working tree

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Staging area

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

Commit

Two branches diverged on Git, and I—  
I pulled the one less traveled by,

# “hard undo” `git reset --hard HEAD^1`

Working tree

Two branches diverged on Git, and I—

Staging area

Two branches diverged on Git, and I—

Commit

Two branches diverged on Git, and I—

Git won't let me switch?

```
git commit --all -m WIP
```

```
git checkout xyz
```

```
git reset HEAD^1
```



# How do you merge branches?

`git merge issue-5`

Dealing with conflicts

git status

# On branch master

# You have unmerged paths.

# (fix conflicts and run "git commit")

#

# Unmerged paths:

# (use "git add <file>..." to mark resolution)

#

# both modified: index.html

#

# no changes added to commit (use "git add" and/or "git commit -a")

<<<<<<< HEAD:index.html

```
<div id="footer">contact :  
email.support@github.com</div>
```

=====

```
<div id="footer">  
  please contact us at  
  support@github.com  
</div>
```

>>>>>>> issue-5:index.html

[rstd.io/wtf-seattle](https://rstd.io/wtf-seattle)

[activities/git-branch-01.R](#)

# Git remotes

Jim Hester  
RStudio

 @jimhester

 @jimhester\_

<http://happygitwithr.com/remotes.html>



Remote repositories are versions of your project that are hosted on the Internet or another network

# Multiple Remotes?

How do you see what remotes you already have?

```
git remote -v
```

How do you add a new remote?

```
git remote add happygit https://xyz.git
```

# Remote nicknames

origin

upstream

github

jennys

How do you retrieve data from a remote?

```
git fetch https://xyz.git
```

```
git fetch happygit
```

# How do you merge the remote data?

```
git merge happygit/master master
```

```
git pull happygit master
```



# How do you send data to a remote?

```
git push origin master
```

```
git push happygit feature-1
```

How do you change remotes?

```
git remote rename happygit upstream
```

```
git remote set-url upstream https://newurl.git
```

# The fork remote rotation

```
git clone https://github.com/a/repo
```

...

```
fork repo on github
```

```
git remote rename origin upstream
```

```
git remote add origin https://github.com/me/repo
```

[rstd.io/wtf-seattle](https://rstd.io/wtf-seattle)

[activities/git-remotes-01.R](#)