
GIT REAL
2

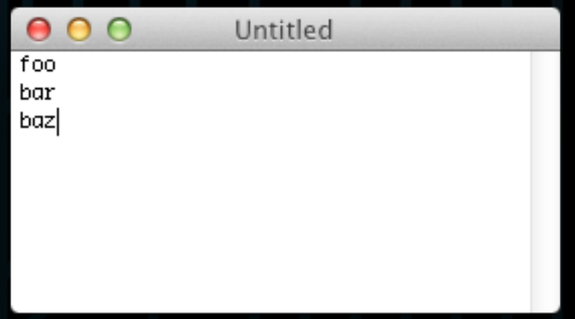
WORKING TOGETHER

CHAPTER 4

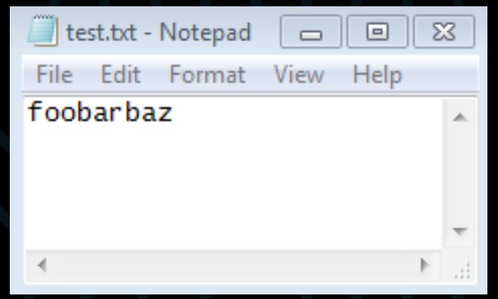
LINE ENDINGS

Different OSs store line separators differently

OS	Expects Character(s)
OSX/Linux	LF (LineFeed)
Windows	CR (Carriage Return) and LF pair



Some team members work on OSX and Linux



Some work on Windows

retrieving the wrong format means trouble!

LINE ENDINGS

Git can auto-correct formats for each OS

On Unix-like systems (Linux, OSX, etc.)

```
$ git config --global core.autocrlf input
```

Changes CR/LF to LF on **commit**

(fixes any Windows line endings that get introduced)

On Windows systems

```
> git config --global core.autocrlf true
```

Changes LF to CR/LF on **checkout**

(but converts back to LF on commit)

On Windows-**only** projects

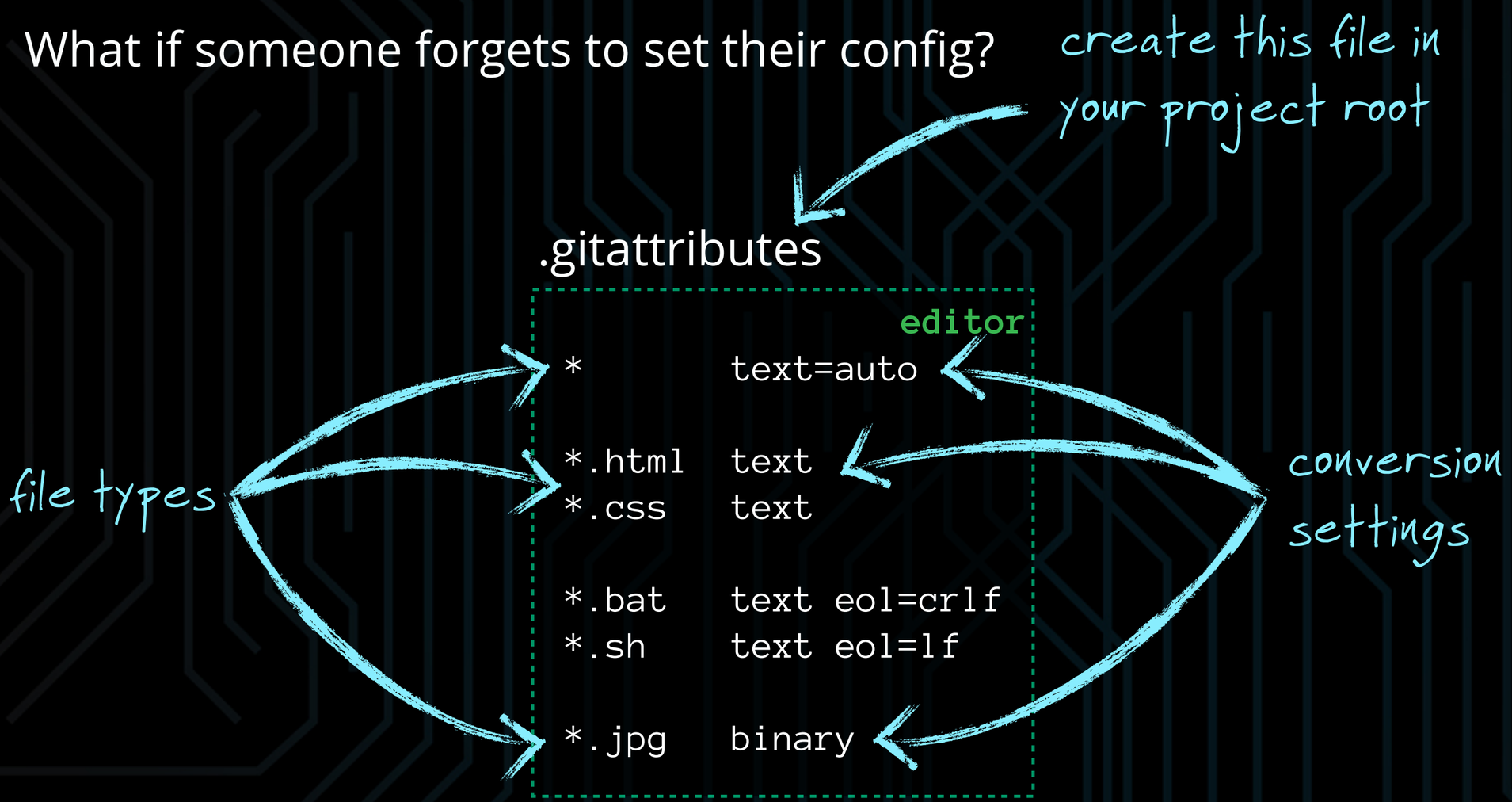
```
> git config core.autocrlf false
```

Does no conversion

(conversion isn't needed if everyone expects CR/LF)

GIT ATTRIBUTES FILE

What if someone forgets to set their config?



GIT ATTRIBUTES FILE

File type - designates which files rule applies to

`*`	Matches any type without an extension-specific rule
`*.html`	Matches all HTML files
`*.jpg`	Matches all JPEG files

GIT ATTRIBUTES FILE

Conversion settings

`text=auto`	Choose conversion automatically
`text`	Treat files as text - convert to OS's line endings on checkout, back to LF on commit
`text eol=crlf` `text eol=lf`	Convert to specified format on checkout, back to LF on commit
`binary`	Treat files as binary - do no conversion

GIT ATTRIBUTES FILE

Some typical rules

```
* text=auto
```

By default, auto-convert line endings

```
*.html text
*.css  text
```

Treat HTML and CSS files as text

(not needed if you have the line above)

```
*.jpg binary
*.png binary
```

Treat image files as binary

```
*.sh  text eol=lf
*.bat text eol=crlf
```

Keep shell scripts in Unix format,
batch files in Windows format

CHERRY-PICK

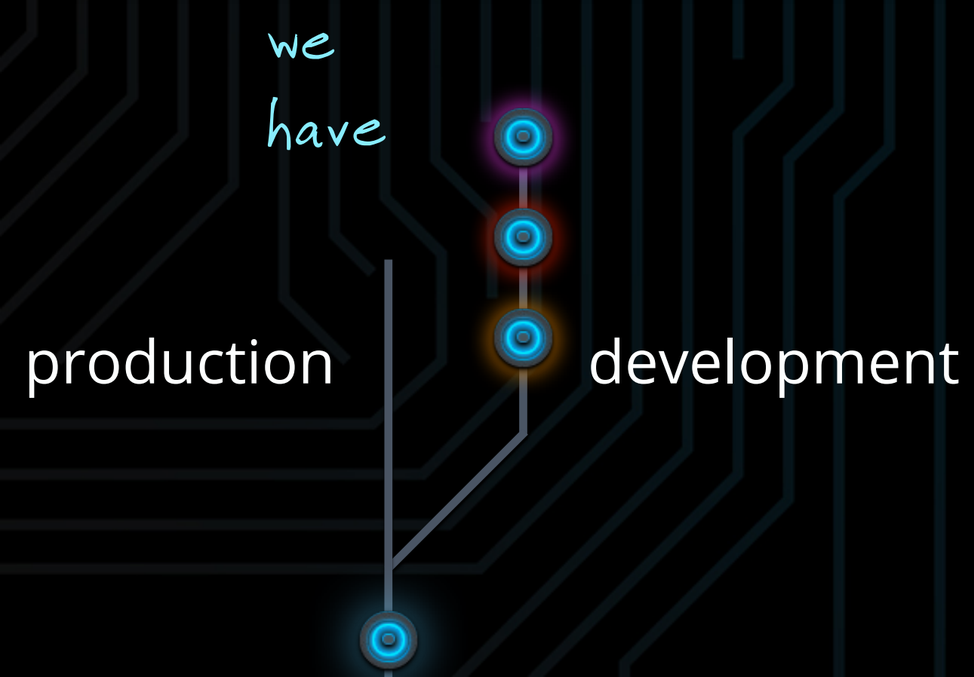
"production" branch

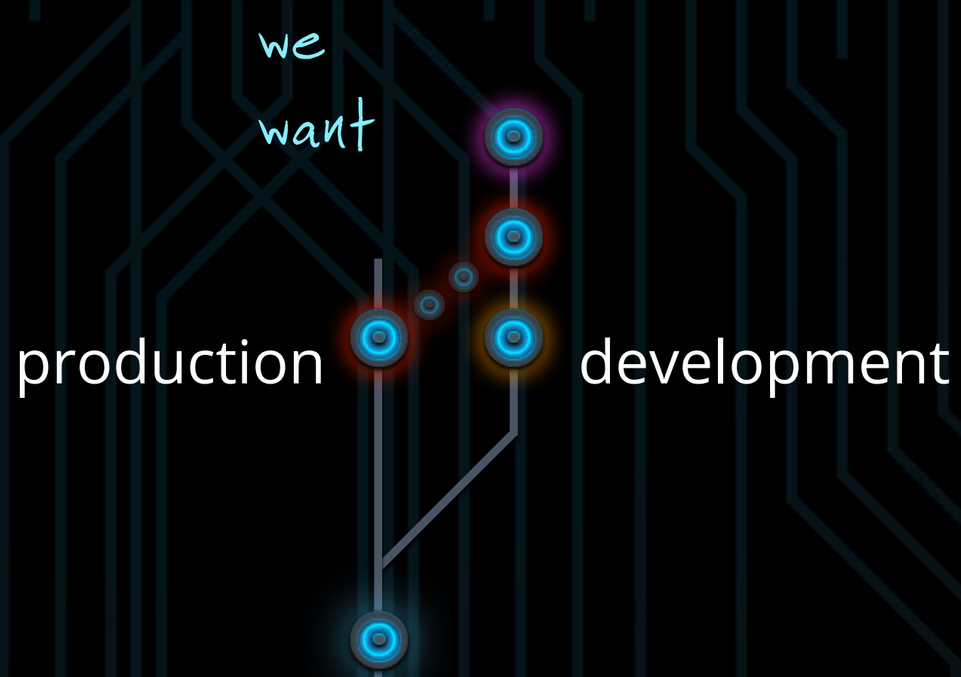
```
$ git log --oneline
3d7225f Add guppies.
```

"development" branch

```
$ git log --oneline
c400bf0 Add Cthulhu.
53212e5 Add clownfish.
55ae374 Add sharks.
3d7225f Add guppies.
```

We need **just** the clownfish page in production ASAP





CHERRY-PICK

"production" branch

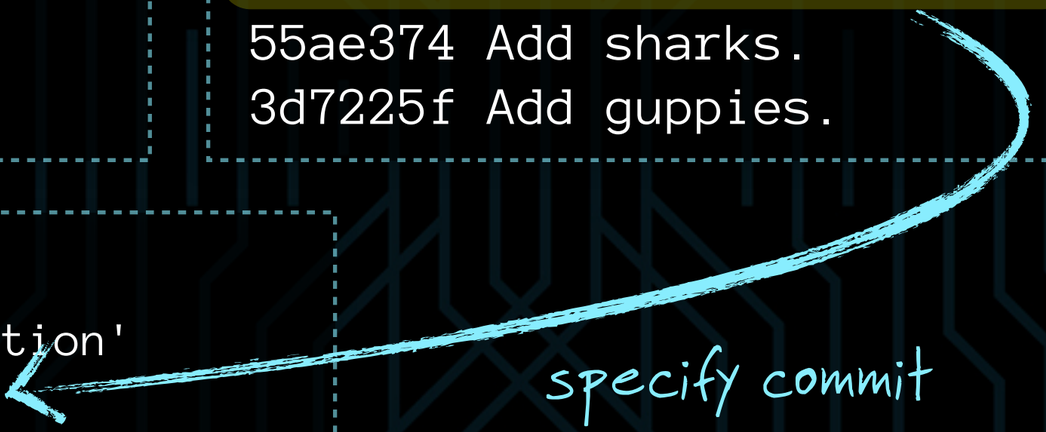
```
$ git log --oneline
3d7225f Add guppies.
```

"development" branch

```
$ git log --oneline
c400bf0 Add Cthulhu.
53212e5 Add clownfish.
55ae374 Add sharks.
3d7225f Add guppies.
```

We need **just** the clownfish page in production ASAP

```
$ git checkout production
Switched to branch 'production'
$ git cherry-pick 53212e5
[production 80a16e2] Add clownfish.
 1 file changed, 1 insertion(+)
 create mode 100644 clownfish.html
```



CHERRY-PICK

"production" branch

```
$ git log --oneline
80a16e2 Add clownfish.
3d7225f Add guppies.
```

"development" branch

```
$ git log --oneline
c400bf0 Add Cthulhu.
53212e5 Add clownfish.
55ae374 Add sharks.
3d7225f Add guppies.
```

production

development

We copied a single commit to the current branch

CHERRY-PICK - EDIT COMMIT

We want a different message on the cherry-picked commit...

```
$ git cherry-pick --edit 5321
```

--edit lets you change messages

editor

```
Add clownfish.
#
# It looks like you may be committing a cherry-pick.
# If this is not correct, please remove the file
#   .git/CHERRY_PICK_HEAD
# and try again.

# Please enter the commit message for your changes...
```

Shows an editor

CHERRY-PICK - EDIT COMMIT

We want a different message on the cherry-picked commit...

```
$ git cherry-pick --edit 5321
```

--edit lets you change messages

editor

```
Pull in clownfish from development.
#
# It looks like you may be committing a cherry-pick.
# If this is not correct, please remove the file
#   .git/CHERRY_PICK_HEAD
# and try again.

# Please enter the commit message for your changes...
```

Enter your message, save, and quit

```
$ git log --oneline
03785d9 Pull in clownfish from development.
3d7225f Add guppies.
```

New commit has new message

CHERRY-PICK - CUSTOMIZE COMMIT

"production" branch

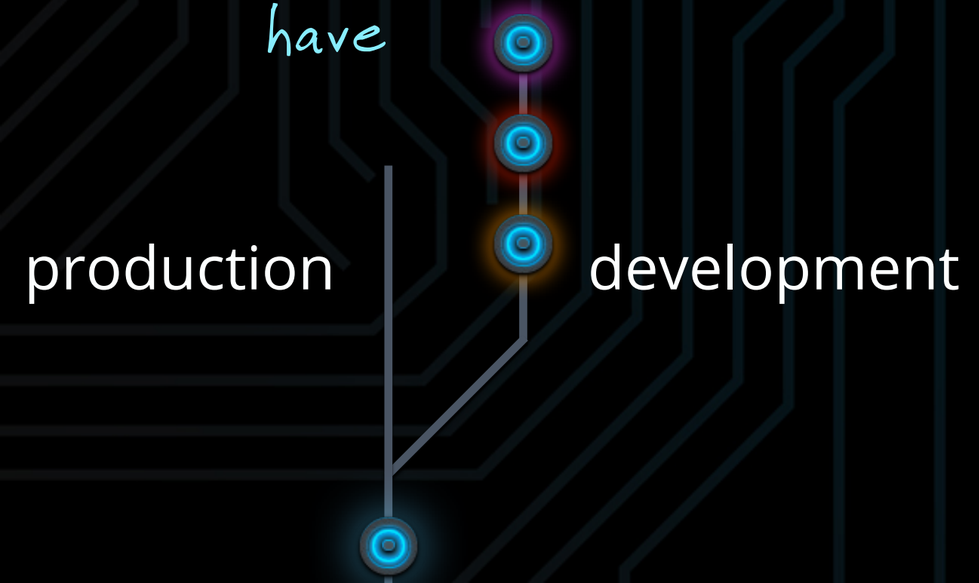
```
$ git log --oneline
3d7225f Add guppies.
```

"development" branch

```
$ git log --oneline
c400bf0 Add Cthulhu.
53212e5 Add clownfish.
55ae374 Add sharks.
3d7225f Add guppies.
```

We need to cherry-pick **and combine** these commits

we have



we want

CHERRY-PICK - CUSTOMIZE COMMIT

"production" branch

```
$ git log --oneline
3d7225f Add guppies.
```

"development" branch

```
$ git log --oneline
c400bf0 Add Cthulhu.
53212e5 Add clownfish.
55ae374 Add sharks.
3d7225f Add guppies.
```

```
$ git cherry-pick --no-commit 53212e5 55ae374
```

```
$ git status
# On branch production
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#  new file:   clownfish.html
#  new file:   shark.html
```

--no-commit pulls in changes and stages them, but doesn't commit

CHERRY-PICK - CUSTOMIZE COMMIT

```
$ git commit -m "Add clownfish and sharks (in separate tanks)."
[production bc75856] Add clownfish and sharks (in separate tanks).
 2 files changed, 2 insertions(+)
 create mode 100644 clownfish.html
 create mode 100644 shark.html
```

Now, manually commit the changes

production development

We cherry-picked the changes from two commits and made a single commit

CHERRY-PICK

We want to track which commit we cherry-picked from

```
$ git cherry-pick -x 5321
[production 6bbeedc] Add clownfish. (cherry picked from commit
53212e591c75100b9b462d3a4e8bdae56d9facae)
 1 file changed, 1 insertion(+)
 create mode 100644 clownfish.html
```

```
$ git log --oneline
6bbeedc Add clownfish. (cherry picked from commit
53212e591c75100b9b462d3a4e8bdae56d9facae)
3d7225f Add guppies.
```

-x adds source SHA to commit message

Only useful with public branches; don't use for local branches

CHERRY-PICK

We want to track who cherry-picked the commit along with the original committer

```
$ git cherry-pick --signoff 5321
[production ef0c80c] Add clownfish.
 Author: Gregg <gregg@example.com>
 1 file changed, 1 insertion(+)
 create mode 100644 clownfish.html
```

--signoff adds current user's name to commit message

```
$ git log
commit ef0c80cf5bb0357f8f39e0ae76628043ccf0ee91
Author: Gregg <gregg@example.com>
Date:   Wed Mar 20 22:20:07 2013 -0700

    Add clownfish.

    Signed-off-by: Jane <jane@example.com>
...
```

committed by...

signed off by...