

# Working with Git Branches

---

## UNDERSTANDING GIT BRANCH BASICS

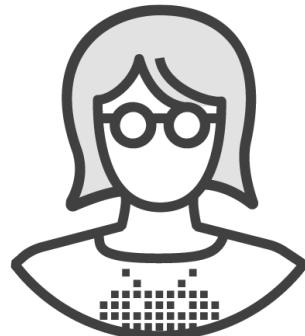


**Craig Golightly**

SENIOR SOFTWARE CONSULTANT

@seethatgo [www.seethatgo.com](http://www.seethatgo.com)





# Oops Path



“Quick” Fix



How to save work?



Commit on a branch



```
git checkout -b quickfix
```

◀ Create a branch called “quickfix” and move to that branch

```
git add *
```

◀ Stage the changes

```
git commit -m "start of changes  
for quickfix"
```

◀ Commit the changes in the “quickfix” branch

```
git checkout master
```

◀ Master is now clean and changes are saved in “quickfix” branch





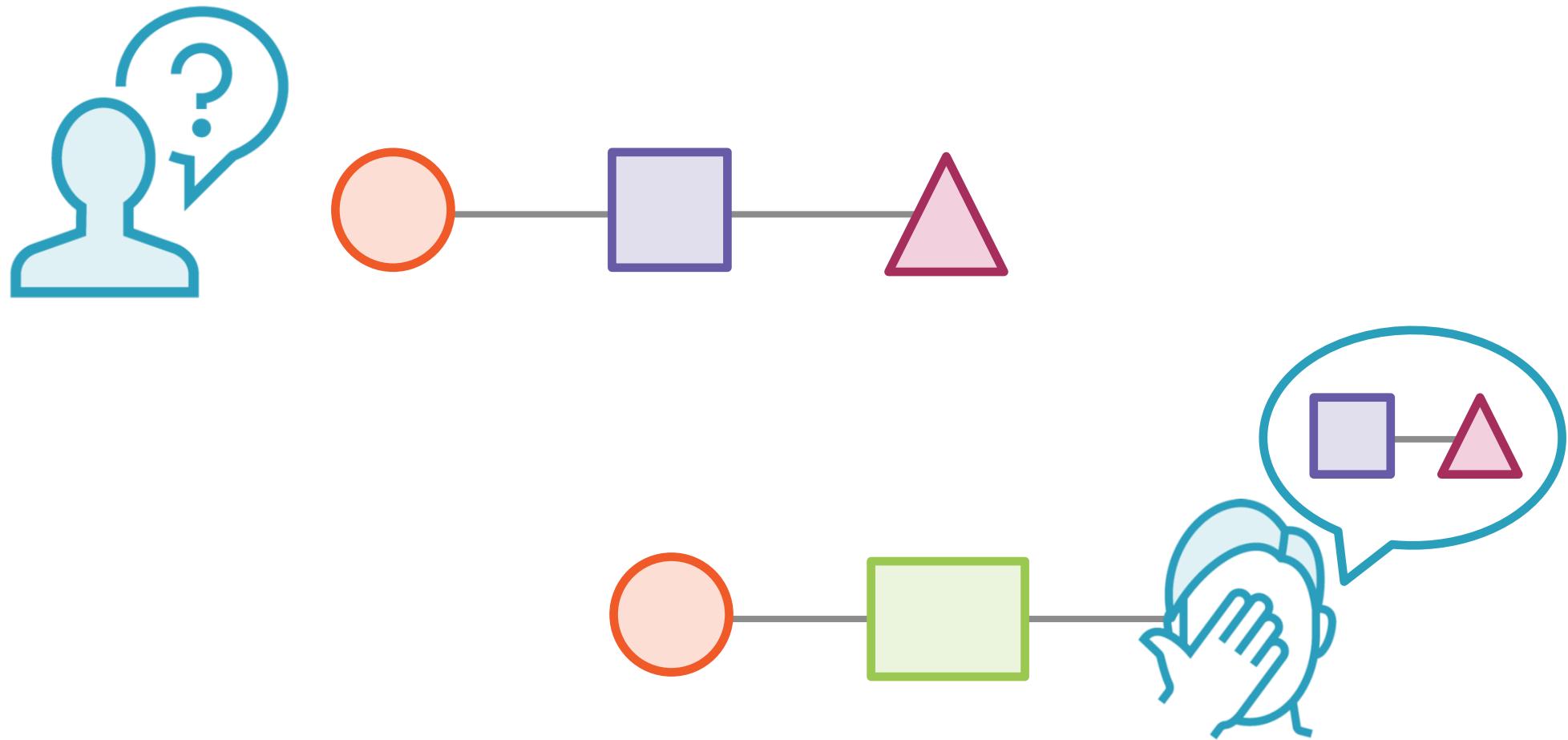
```
git branch
```

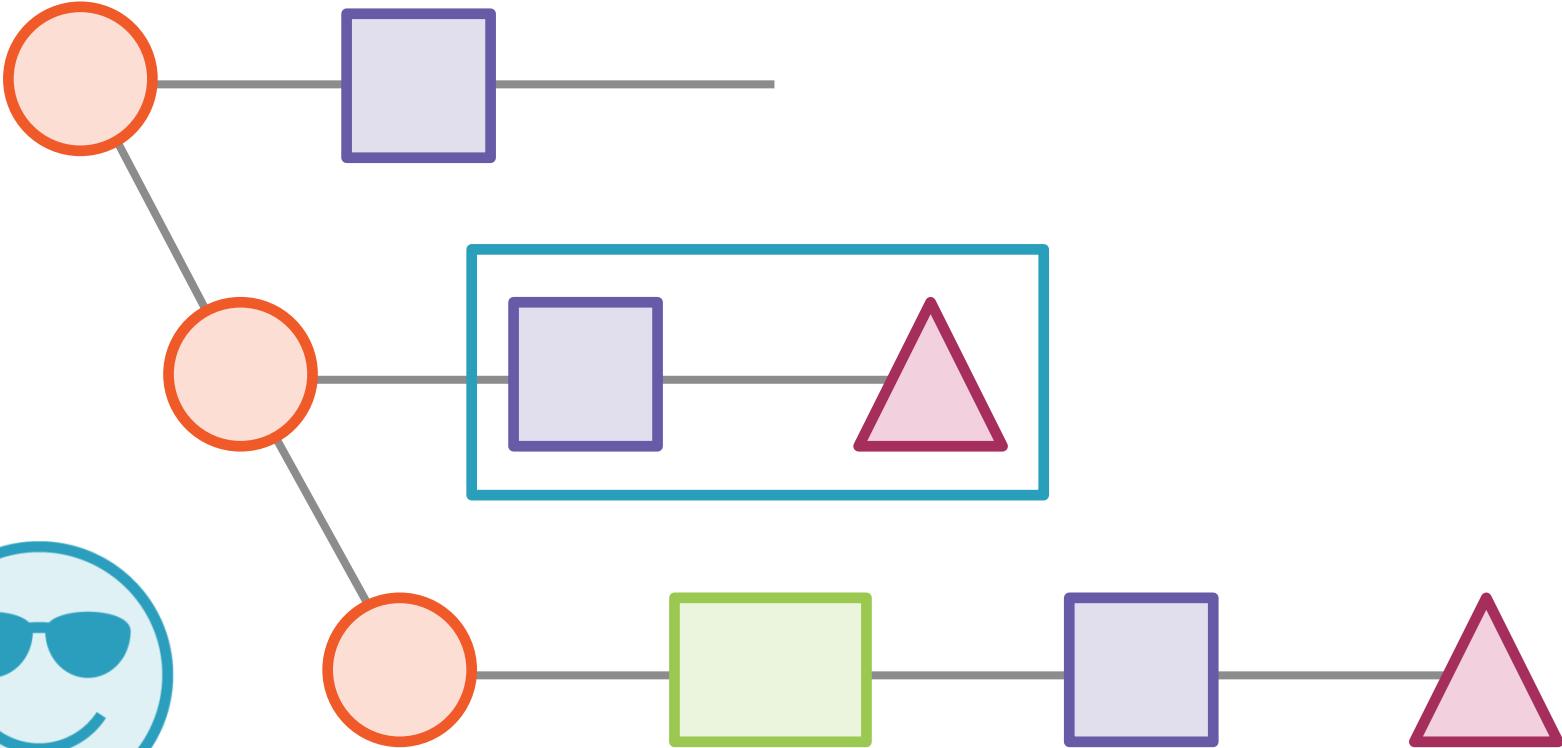
◀ List your branches

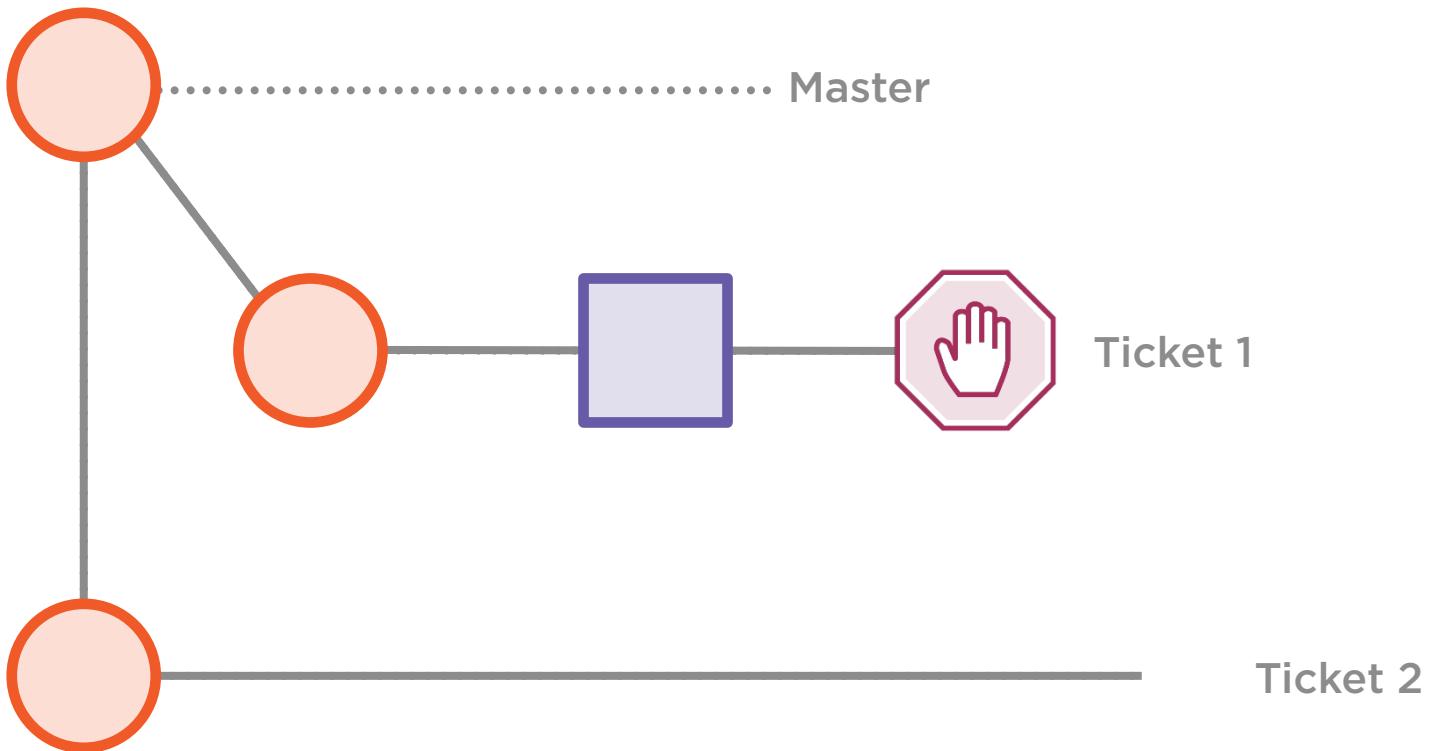
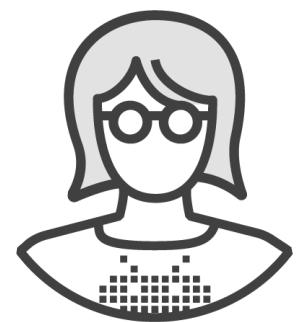
```
git checkout <branch-name>
```

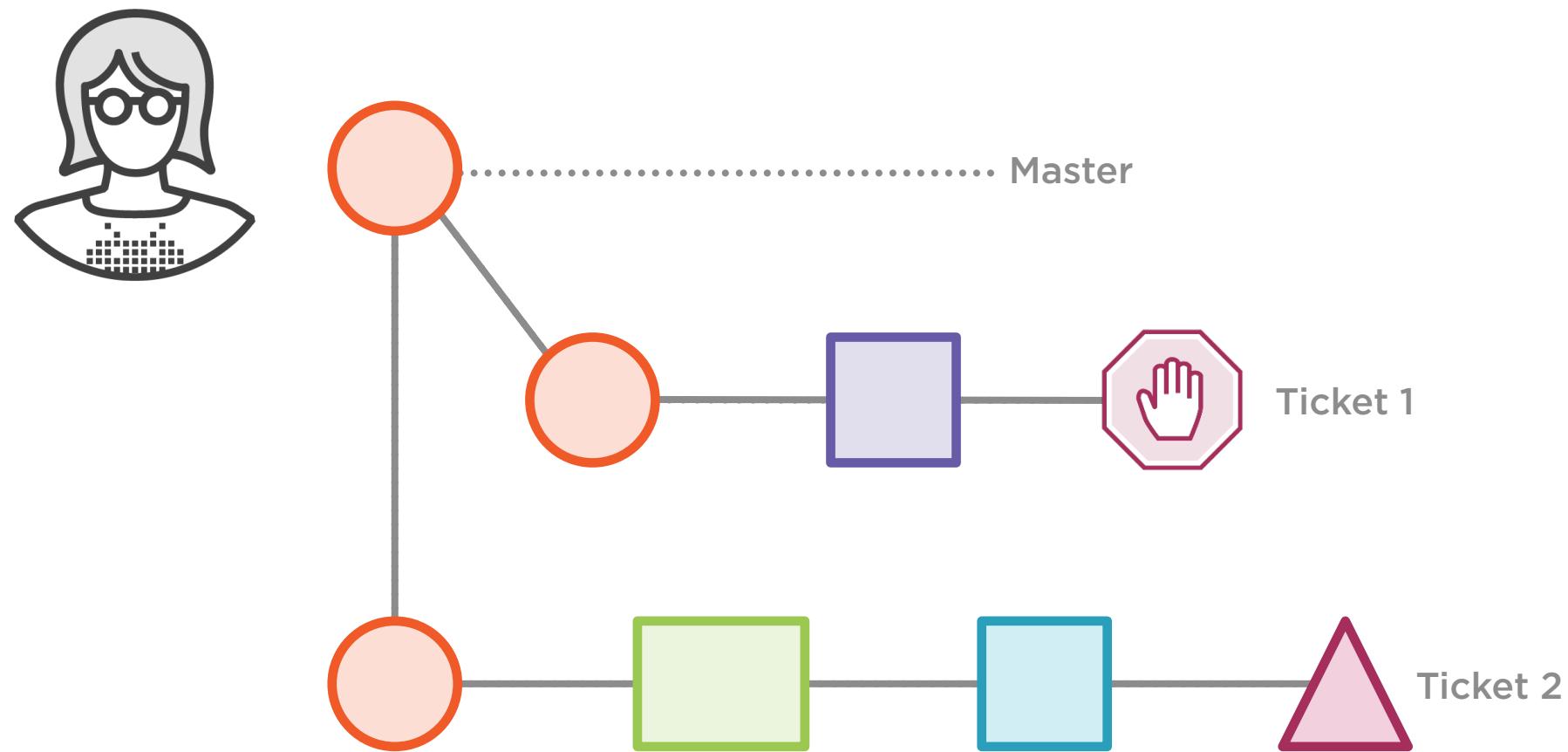
◀ Switch to work in a different branch

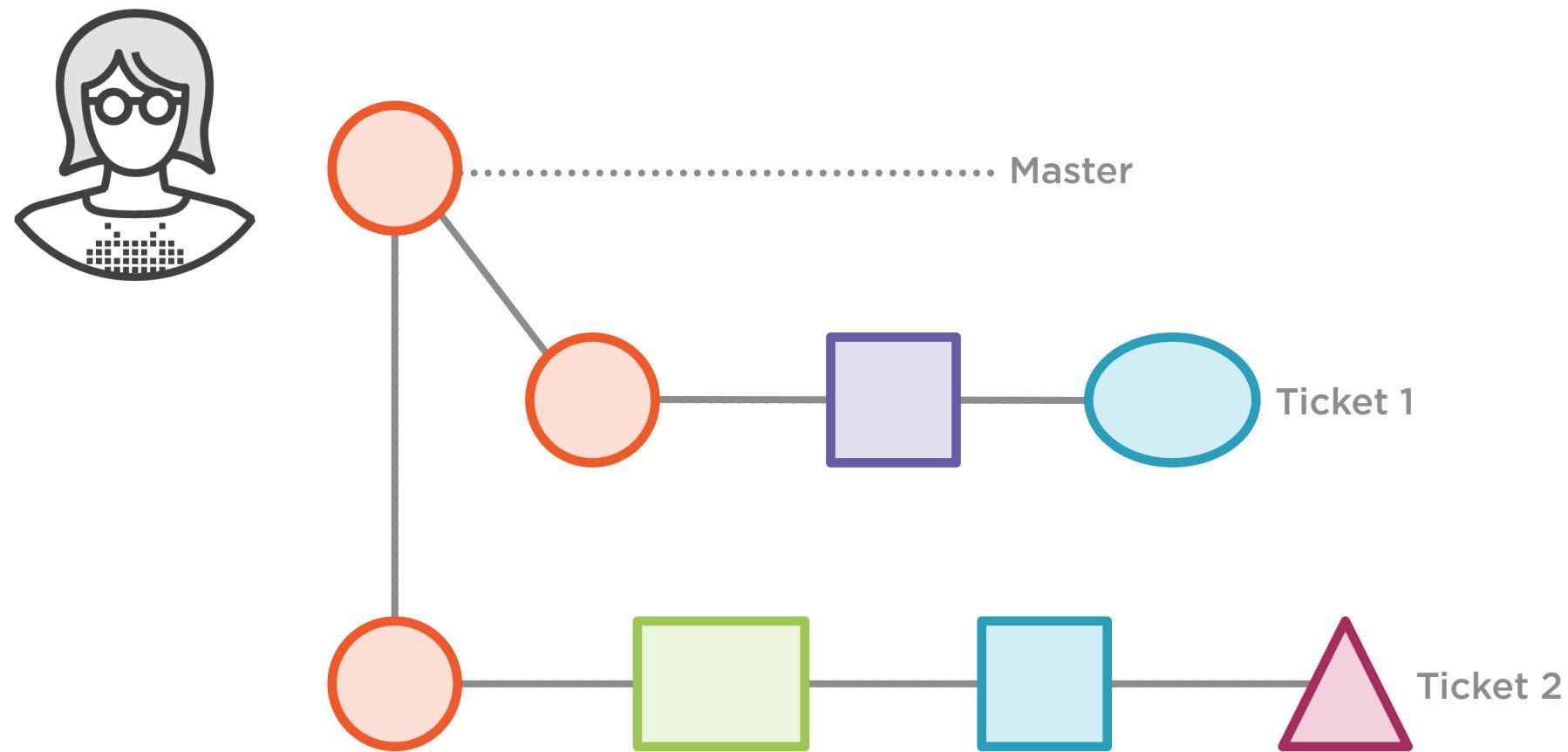














**Branches help you organize your work**

**Changes are isolated in each branch**

**File system automatically updates when switch branches**

**Cleaner and less error-prone than trying to do it yourself without branches**



“There are 2 hard problems in computer science: cache invalidation, naming things, and off-by-1 errors.”

**Phil Karlton and Leon Bambrick**



```
git branch -m <oldbranch> <newbranch>
```

```
git branch -m quickfix longfix
```

## Rename a Branch





```
git branch -d <branchname>
```

```
git branch -d longfix
```

```
git branch -D longfix
```

## Delete a Branch

**Error message if unmerged commits**

**Can force delete if sure**



# Summary



**Create, rename, delete**

**Move from branch to branch**

**How branching can help you**

- Oops case
- Iterating to solve a problem
- When you are blocked



Up Next:  
Merging Made Easy

---

