Continuous Integration Using Github Action

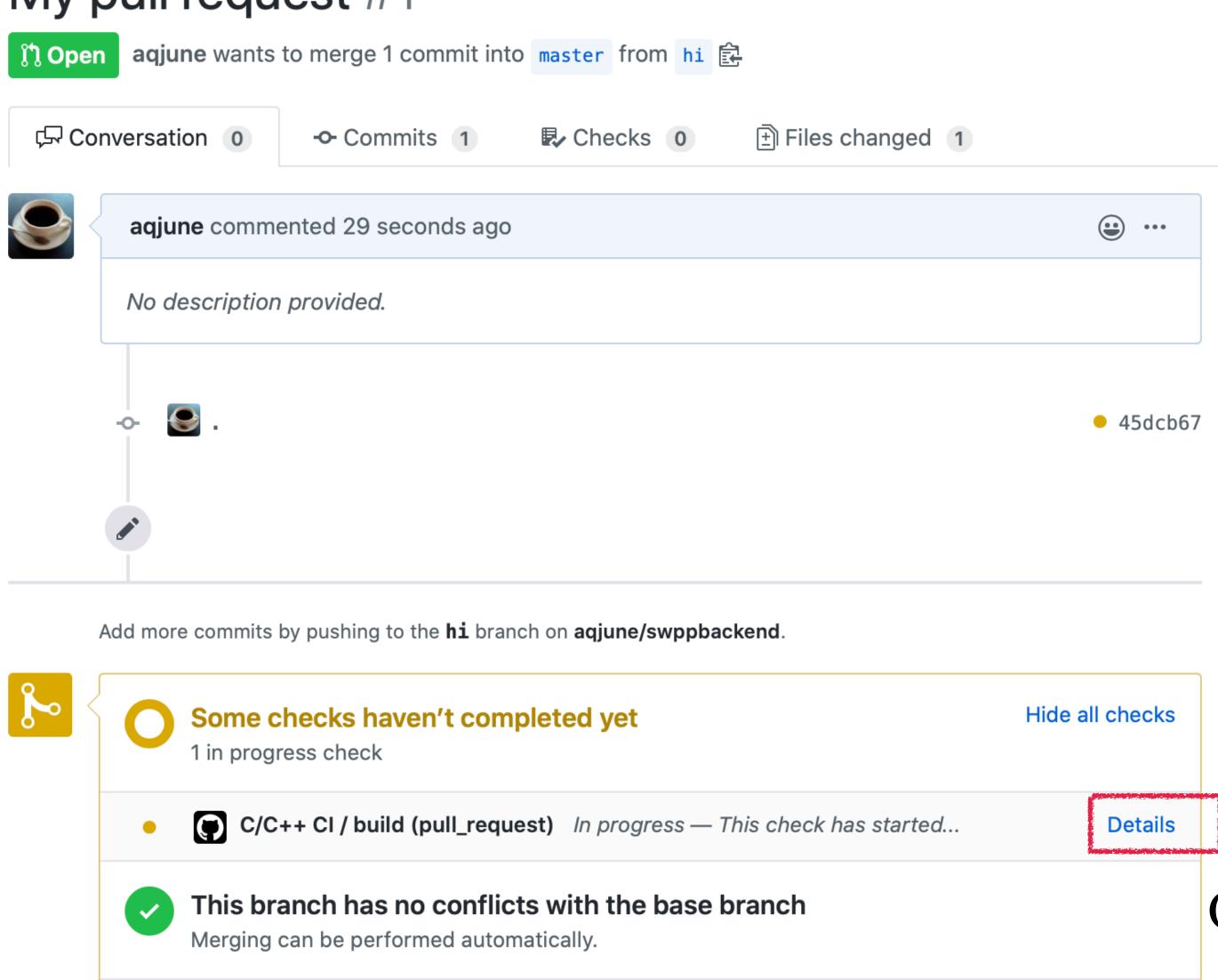
Juneyoung Lee May. 26

Continuous Integration?

- Build your project & run tests whenever...
 - A new commit is created
 - A pull request is made
- Helps developers to be aware of validity of a patch + the master branch

My pull request #1

Merge pull request



You can also open this in GitHub Desktop or view command line instructions.

GitHub is checking validity of 'hi' by cloning and building it.



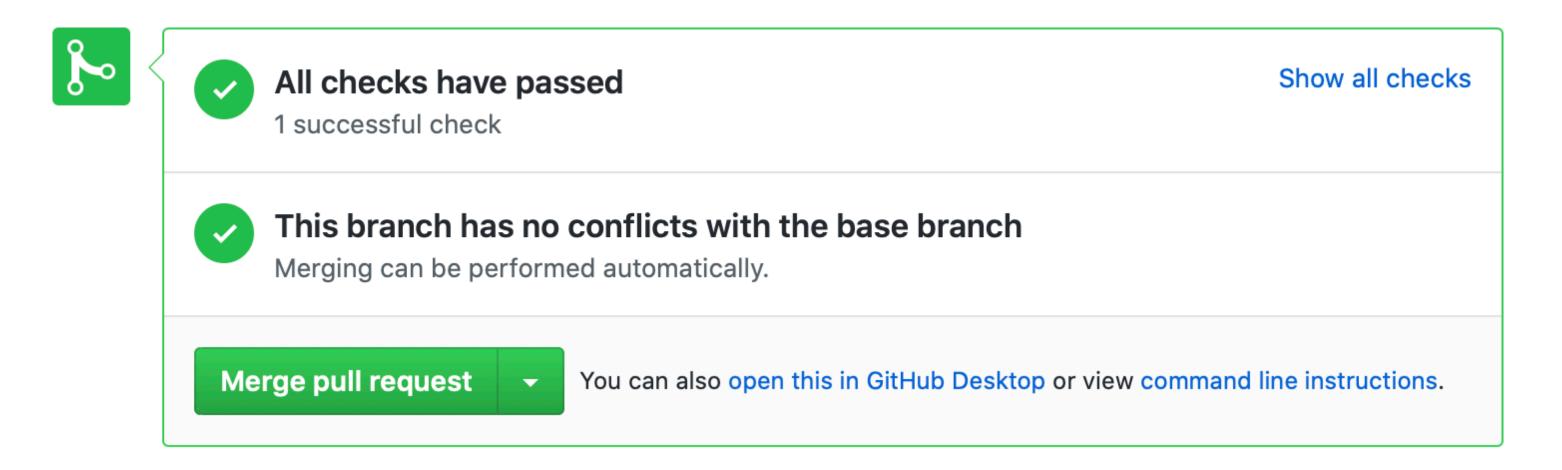
on: pull_request ✓ build

C/C++ CI / build

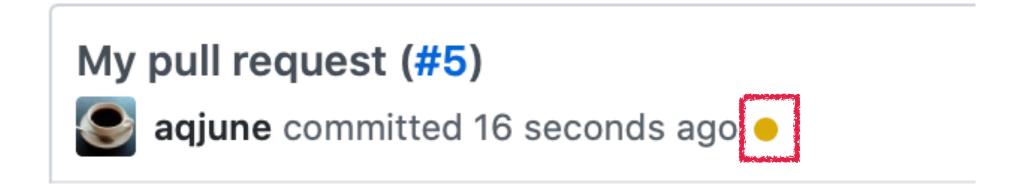
succeeded now in 1m 51s

Complete job

```
▶ ✓ Set up job
  ✓ Initialize containers
  ✓ Run actions/checkout@v2
 configure
▶ ✓ make
▼ ✓ make test
    1 ▶ Run make test
    4 [=======] Running 1 test from 1 test case.
       [-----] Global test environment set-up.
    6 [----] 1 test from TestDemo
      [ RUN ] TestDemo.CheckMain
      [ OK ] TestDemo.CheckMain (0 ms)
       [-----] 1 test from TestDemo (1 ms total)
   10
   11 [----] Global test environment tear-down
   12 [=======] 1 test from 1 test case ran. (1 ms total)
   13 [ PASSED ] 1 test.
   14 --- Start FileCheck.. ---
     Stop containers
      Post Run actions/checkout@v2
```

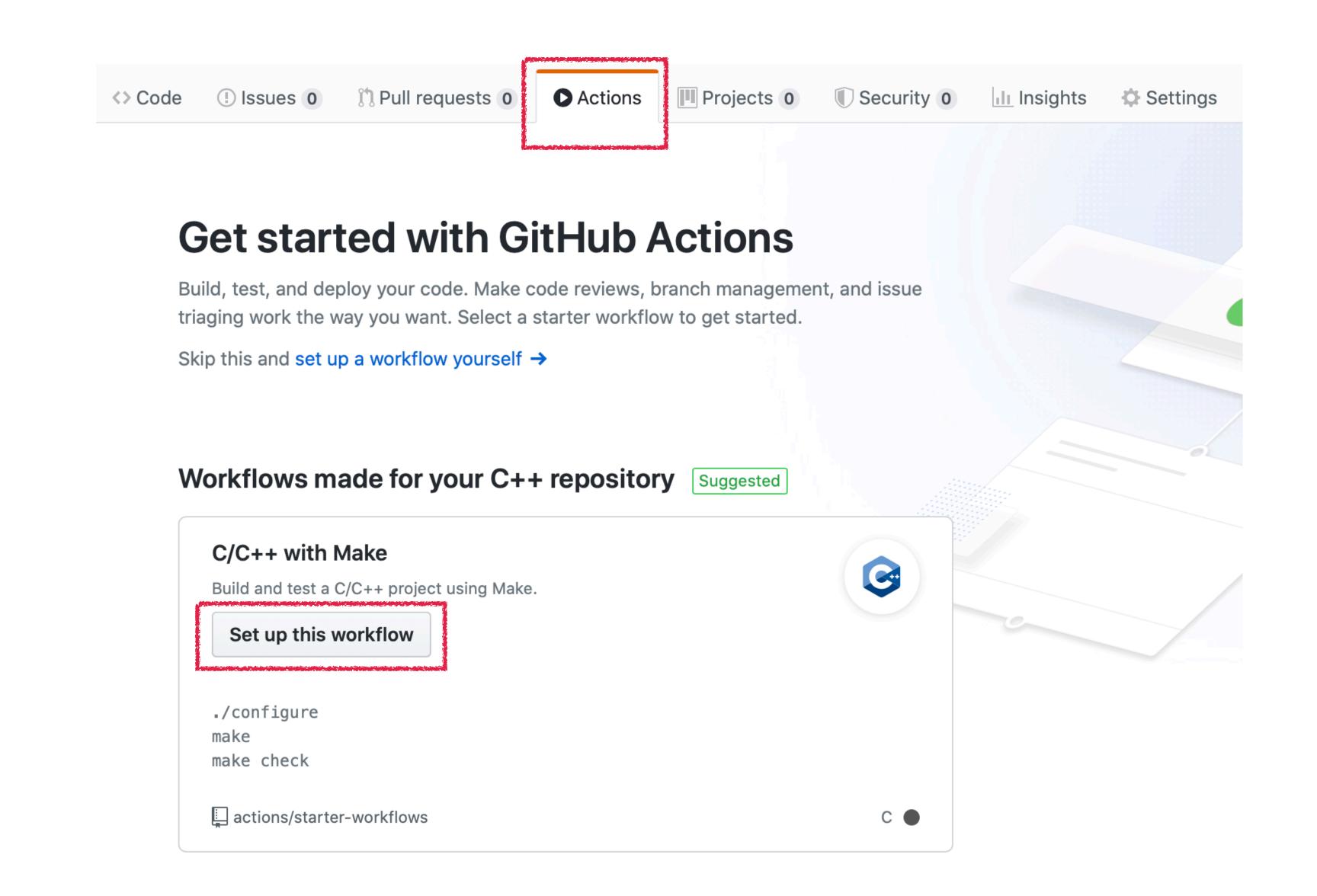


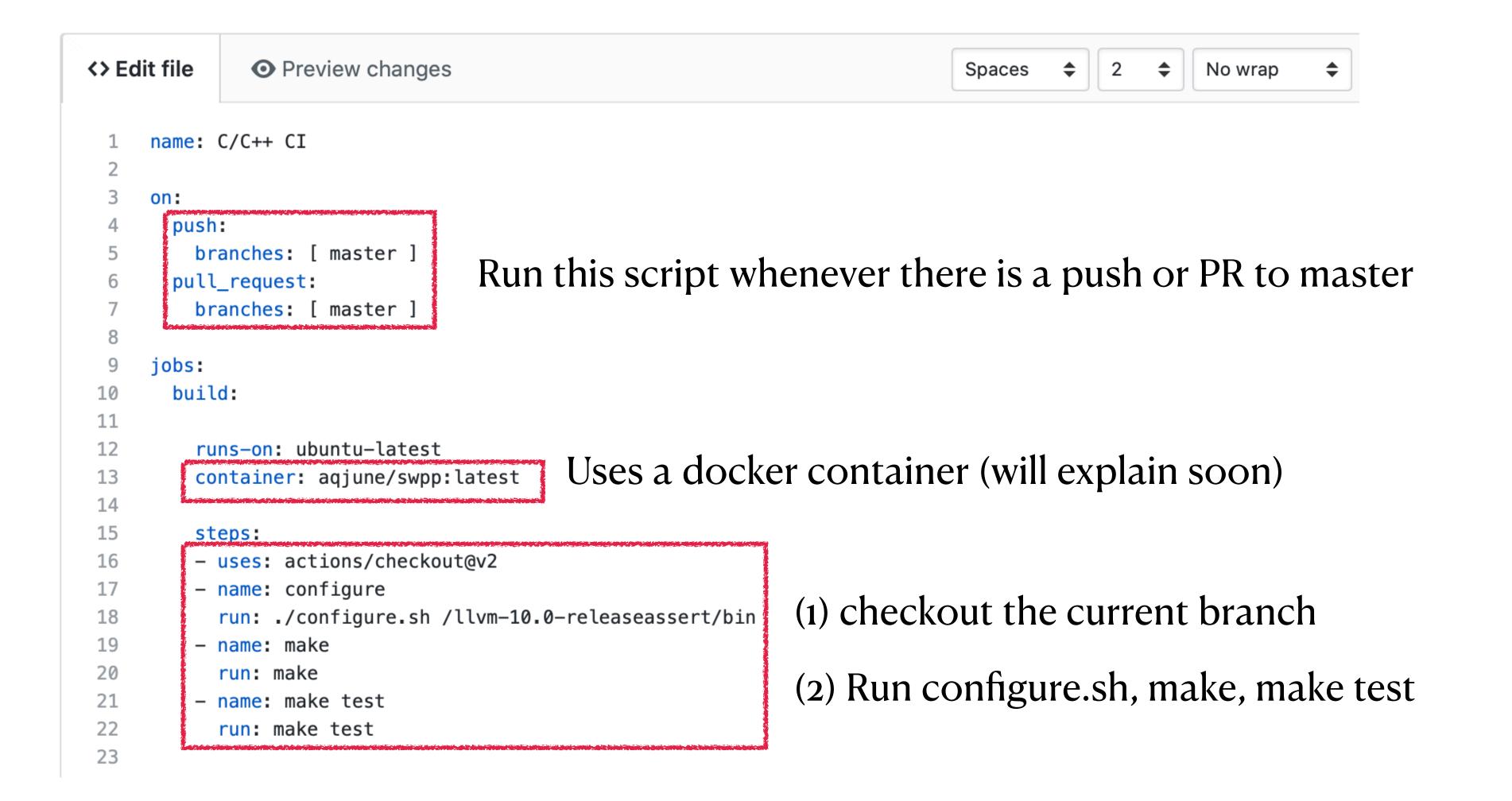
(Don't forget to do 'squash and merge')



GitHub is running scripts after the pull request is merged; It runs on master branch too.

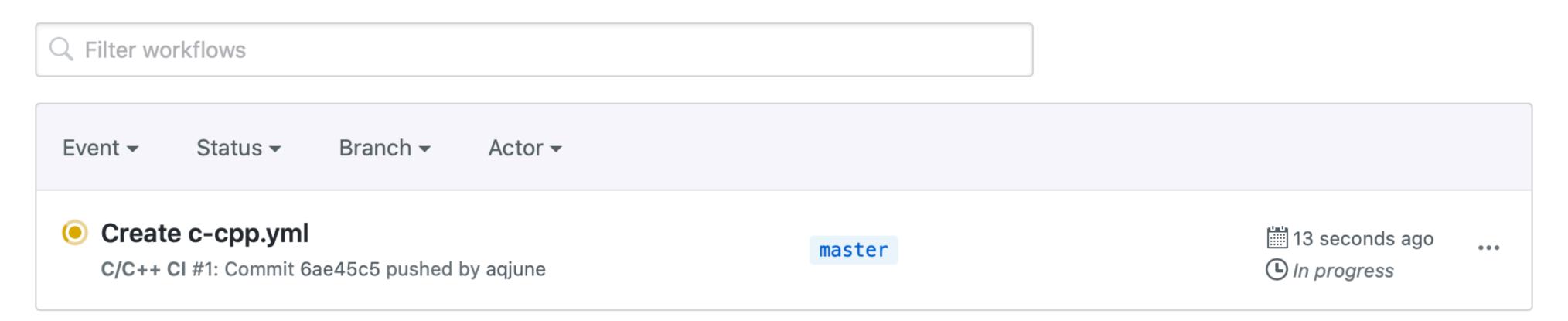
How to use CI for your project?





Start commit **▼**

All workflows

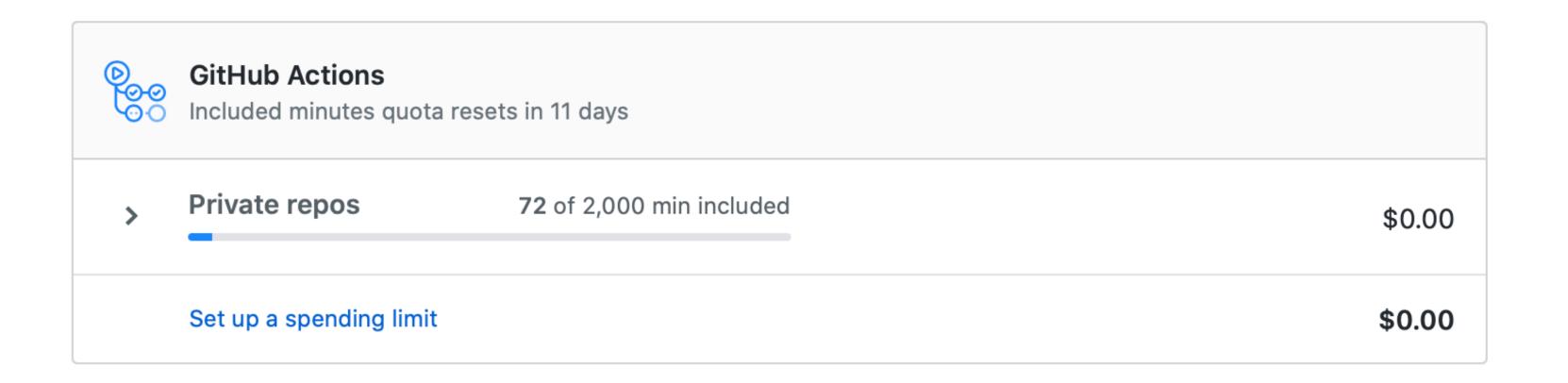


Now you're done.:)

TAs won't count the commits needed to set CIs.

Should I pay to run CI?

- You'll be given a free-tier quota per month: 2000 min, 500 MB *
- https://github.com/settings/billing shows the usage
- In our case, space isn't used (it is for packaging); time is the constraint



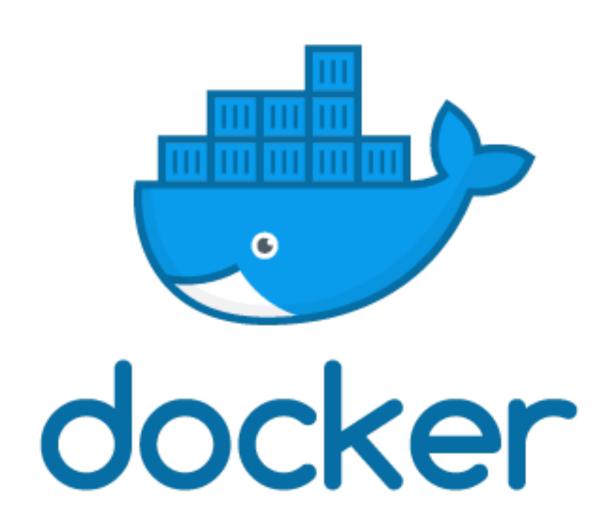
^{*} https://help.github.com/en/github/setting-up-and-managing-billing-and-payments-on-github/about-billing-for-github-actions

Isn't compiling LLVM needed?

- Note that you didn't have to compile LLVM to run CI
- The secret is behind 'container' option
- It is given a lightweight virtual machine that has LLVM already built in
- It is called a docker container, and published online through DockerHub

Docker

- A lightweight virtual machine maintainer
- Has two important concepts:
 - Image: a read-only disk (analogous to .iso file)
 - Container: a virtual machine created from an image
- Dockerfile specifies how to create an image
 - Start from Ubuntu 18.08, apt-get install abc, clone LLVM, ...



DockerHub

hub.docker.com

- Git: GitHub = Docker: DockerHub
- As GitHub does, a single account has several repositories
- A repo contains several tags (think of branches in Git)
- Each tag corresponds to an image
- https://hub.docker.com/repository/docker/aqjune/swpp/general