
Lessons Learnt Using Github actions in an Apache Project

Daan Hoogland && David Jumani



David Jumani

Daan Hoogland



ShapeBlue

Committer && PMC @ Apache CloudStack

Employee Owned

Apache CloudStack

- Support
- Implementation
- Development
- Consultancy
- Community Support



Agenda :

- **Github Actions**
- **Motivation**
- **Past**
- **Present**
- **Future**
- **Learnings**



**Don't try this ~~at home~~
in Prod**

We're innocent !!

TBD

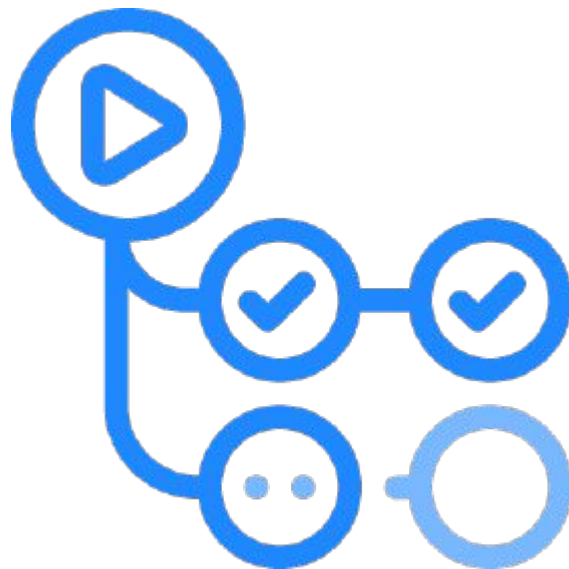


Github Actions ??

Explain

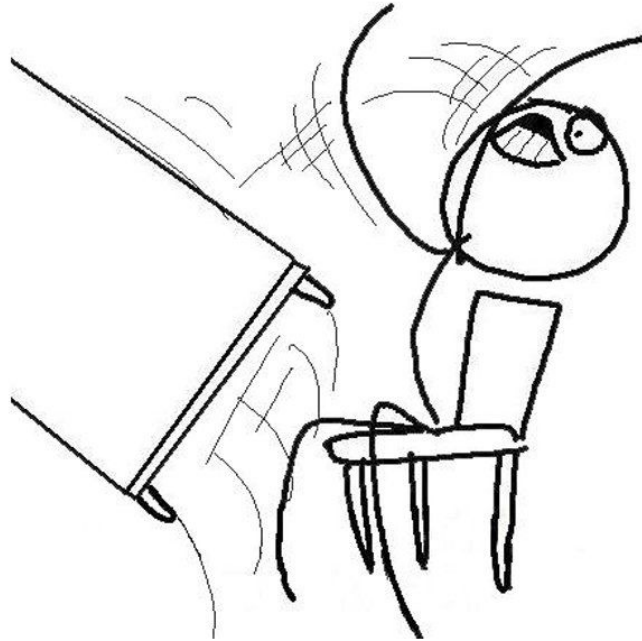
Github Actions ??

- “CI / CD” Platform
- Automate workflow
- Events
- Jobs
- Runner



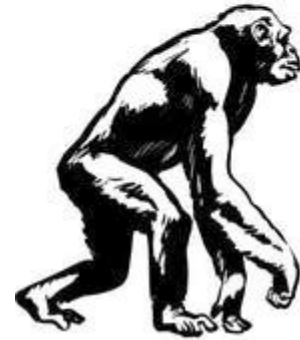
Why are we here ?

- Short staffed
- Manual labour
- Hardware reliant
- We just love automation !



Pre-Alpha Build

- Terrible Manual Test workflow
- Bad / Unavailable Infra
- Better than nothing



Alpha Build

- Github Bot
- Respond to comments
- Cron Job
- Kick packages & Tests
- Report Success / Failure (sometimes)



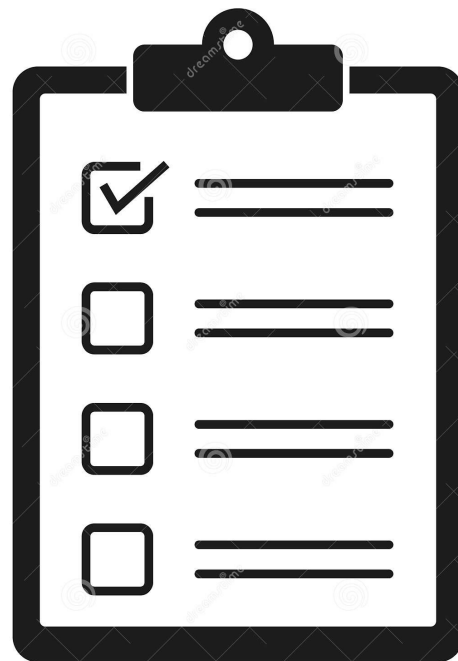
Beta Build

- Github actions
- Boring Cyborg
- Automatically Comments



What we have

- Label Assignment - Triage
- Linter
- Code Coverage
- Code Quality
- Unit & E2E Tests
- Automatically trigger builds
- Notify users of merge conflicts
- Tests on push



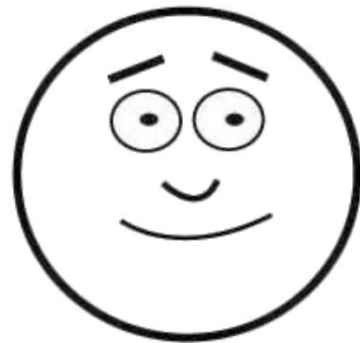
It works (Sorta)

- Triaging
 - Label, Reviewer
 - Ready for review
- Code coverage
- Code quality



It doesn't work (Sorta)

- Test Coverage
 - workflow works, provider doesn't
- All packages and tests



1 - 2
hurts just
a little bit

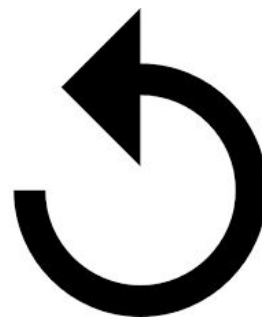
What we've Learnt - The good

- They work !
- Simple
- Super flexible
- Prioritize this



What we've Learnt - The bad

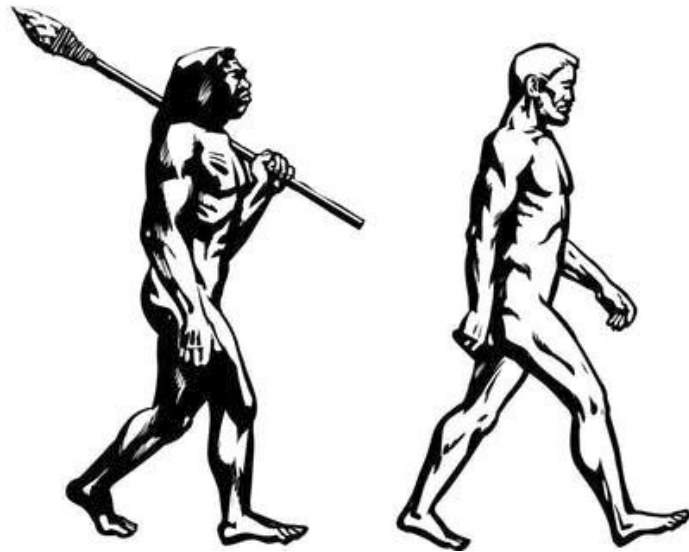
- We have custom infra
- We're human
- Shoulda Woulda Coulda



Ctrl + Z

The future is bright-ish

- Load balancing
- Triggered by tags
- Periodic Healthchecks
- Explore options



Thank You !
