



GitHub Actions Overview

Tuesday April 11, 2023

What are GitHub Actions?

- Custom workflows triggered on any GitHub event
- Workflows stored as `yaml` files
- Live logs and visualized workflow execution
- Community powered workflows
- Linux, macOS, and Windows hosted runners
- Self hosted runners
- Built-in secret store
- Reusable workflows
- Parallelized job execution



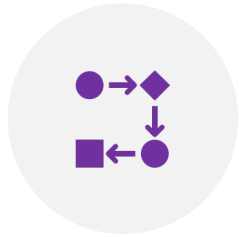
#1

CI on GitHub

10,000+

Actions in GitHub
Marketplace

Anatomy of a GitHub Workflow



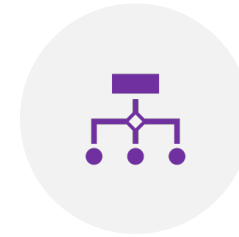
WORKFLOW



TRIGGERS



JOB

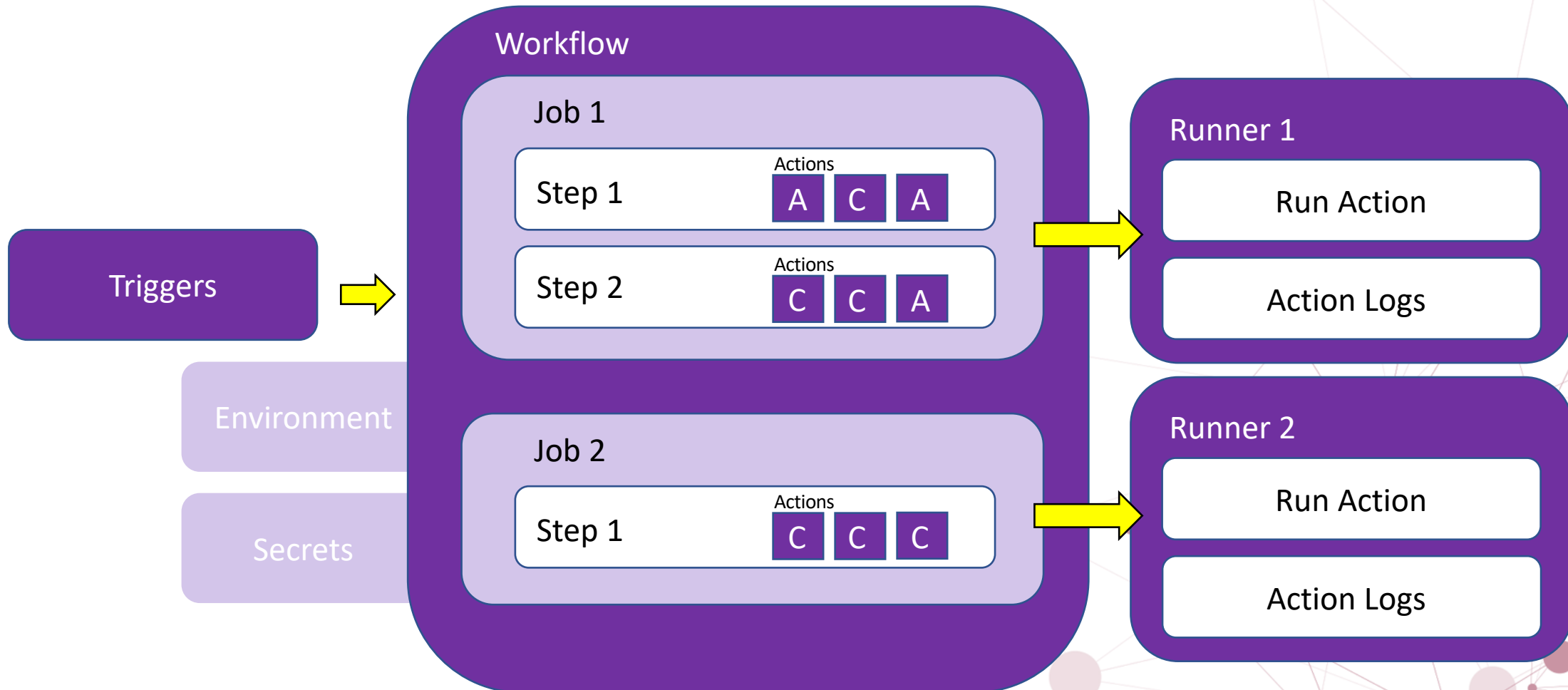


STEPS



ACTIONS

Anatomy of a GitHub Workflow



Quick Summary

- Triggers “trigger” workflows, e.g. a push to a branch
- Workflows contain one or more jobs, which contains one or more steps
- These steps can reference actions or execute commands
- The term “***GitHub Actions***” include all components, not just the Actions themselves

GitHub Marketplace

- Discover open-source Actions across multiple domains
- > 10,000 Actions (and counting)
- Verified creators
- Reference these Actions directly in your workflow
- Integrated into the GitHub editor

The screenshot displays the GitHub Marketplace interface. At the top, there's a navigation bar with 'Marketplace' and 'Explore'. Below it, a header section for 'd GitHub' includes the text 's to help you build and grow' and a 'lore apps' button. To the right, there's an illustration of two cartoon characters, one holding a briefcase and the other holding a laptop. Below the header, there's a search bar with the placeholder text 'Search for apps and actions'. The main content area is divided into two sections: 'Recommended for you' and 'Trending'. The 'Recommended for you' section features four action cards: 'Codecov | Code Coverage' (with a pink umbrella icon), 'Stale' (with a blue robot icon), 'Imgbot' (with a green robot icon), and another card with a blue diamond icon. The 'Trending' section features two cards: 'Hound' (with a purple dog icon) and 'Moesif API Insights' (with an orange 'M' icon). The 'Hound' card shows 'By houndci', 'Automated code reviews', and '3.8k installs'. The 'Moesif API Insights' card shows 'By Moesif', 'Understand API usage and take action with user-centric API observability', and '698 installs'. On the left side of the screenshot, there's a vertical navigation menu with categories like 'API management', 'Chat', 'Code quality', 'Code review', 'Continuous integration', 'Dependency management', 'Deployment', 'Es', 'aining', 'al zation', 'e', 'ri 1', and 'a gement'.

Reusable Workflows

- Workflows can be made reusable to avoid duplication and make it easier to maintain them
- Reusing workflows can help create new workflows more quickly and promote best practices
- A "caller" workflow uses a "called" workflow, and a single caller workflow can use multiple called workflows
- Reusing a workflow means using the entire called workflow as if it was part of the caller workflow
- If a workflow is reused from a different repository, actions in the called workflow run as if they were part of the caller workflow

GitHub Action Drawbacks

- Secret management is at the repository level, this means the same secret would have to be copied to each repository. This can be hard to manage without automation
- No native ARM support, would need to do some sort of virtualized build via qemu for ARM images
- Learning curve may be higher for developers not used to CI/CD
- Reusable workflows can become complicated and hard to manage for the developer community
- Less support from Linux Foundation
- Signing Git Tags may be difficult given the Sigul setup