

AUTOMATED CI/CD WITH GITHUB ACTIONS

STARTING FROM SCRATCH – A WALKTHROUGH

Dr William Haese-Hill: William.Haese-Hill@glasgow.ac.uk

Research Software Engineer

University of Glasgow

9th September 2025



University
of Glasgow



What is CI/CD?

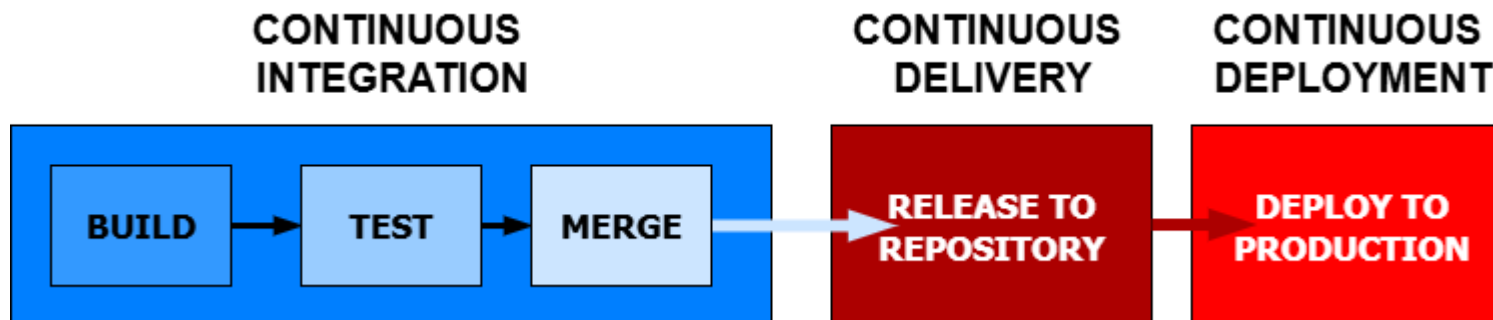
Automating boring, repeatable tasks that ensure code is robust and reusable

Continuous Integration

- Usually triggered when code is updated in PR or main
- Running pre-defined tests
- Pass / Fail - Checking that change doesn't break anything

Continuous Delivery / Deployment

- Often triggered when code is merged into main, or tagged for release
- CI workflow passed -> containerise and deploy to server / hub



Why use CI/CD?

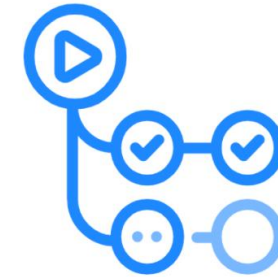
- Runners are containerised – a fresh environment to verify everything installs / runs
- Can ensure code works on a variety of operating systems and versions
 - E.g. Python 3.8 <-> 3.13
 - Windows, Linux, MacOS
- Manually repeating processes can be tedious / forgotten about (-> bugs slipping in)
- Fully customisable triggers
- Makes your software **FAIR**

—_ (ツ) _—
IT WORKS
on my machine

Why use GitHub Actions?



- Paywalls
- Require additional integrations



GitHub Actions

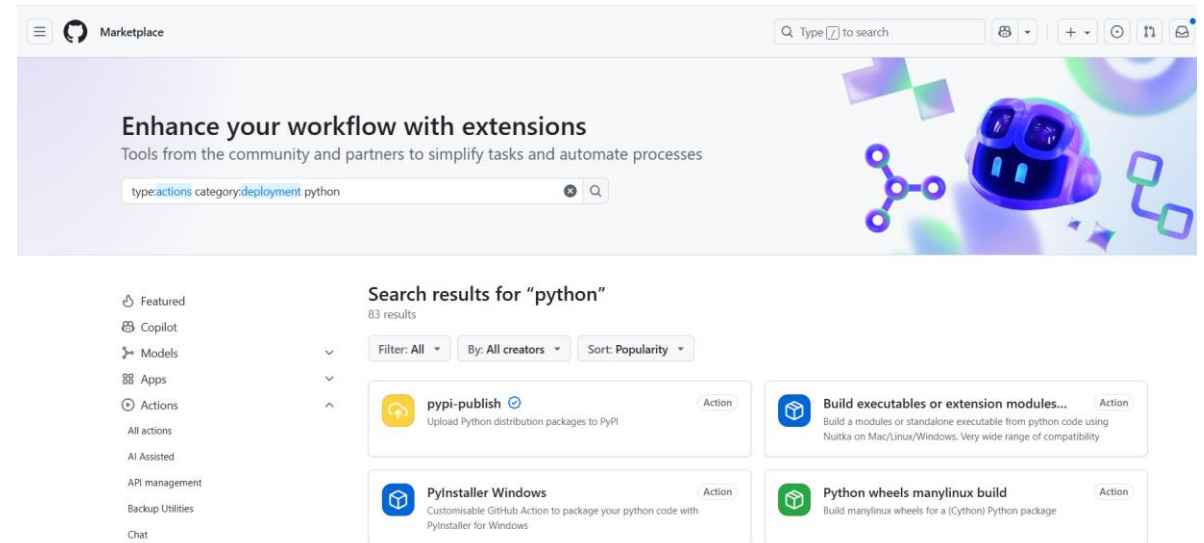
- Simple
- Ubiquitous
- "Free" to use
- Code is already there



* Other options are available

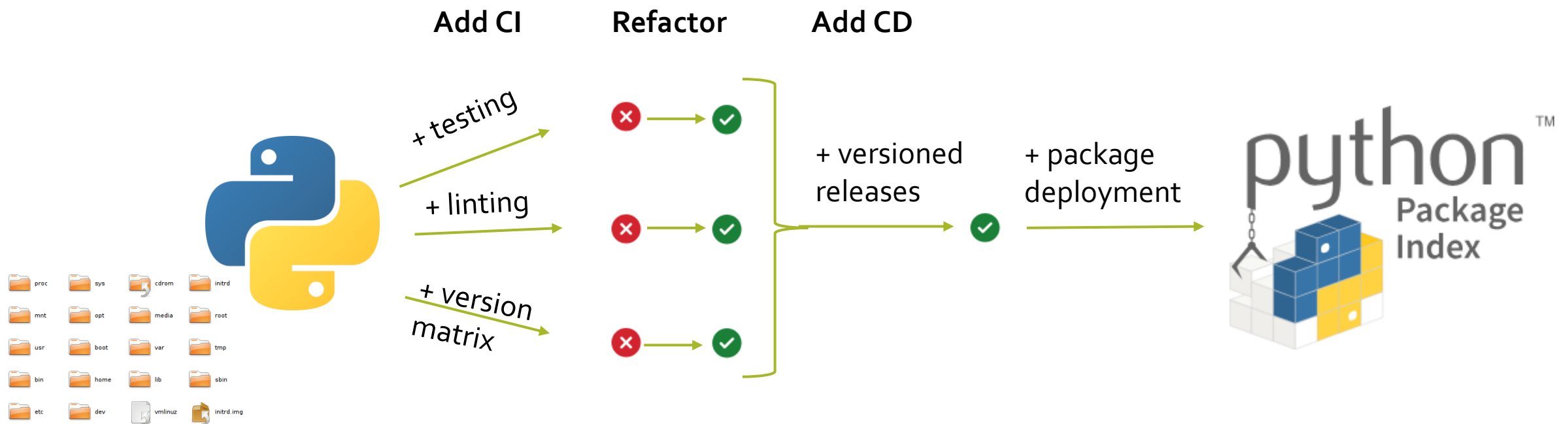
Is it right for your software?

- Not just for big companies
- Templated YAML workflows for common use cases
- Reusable actions from the Marketplace to plug-and-play
- Start simple; build in complexity only when you need it
 - Focus on automating tests as an initial win
 - CD (deployments / semantic versioning) more relevant when production-ready.



Structure of Walkthrough

- Template repo – simple installable Python package with tests
- Will use GitHub Codespaces -> VSCode in the browser
- Will “live code” incremental steps / contrived examples to demonstrate value / common gotchas



Walkthrough: Link to repository template



<https://github.com/haessar/rsecon25-gha-walkthrough>