GitHub Bootcamp

Michael Kaufmann

Introduction to GitHub

Microsoft

June 2018 **$7.5 billion**

Where the world builds software

73M+

Developers

200M+

Repositories

1,000s

Open-Source Communities

2.6B+

Contributions / Year

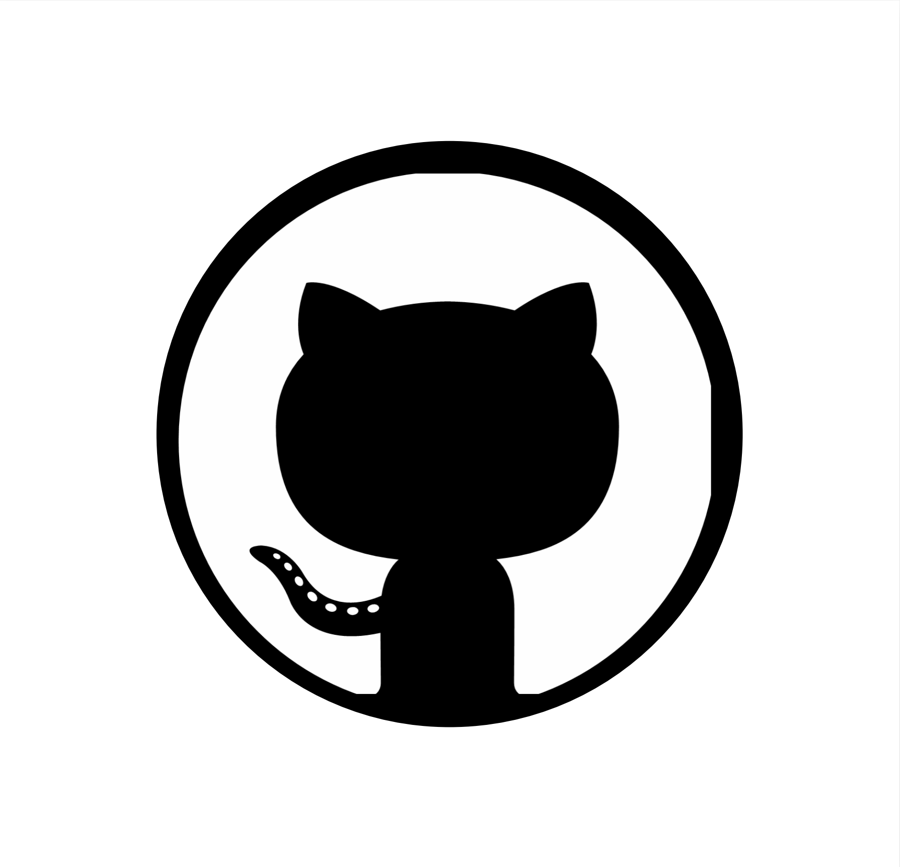
4M+

Organizations

84%

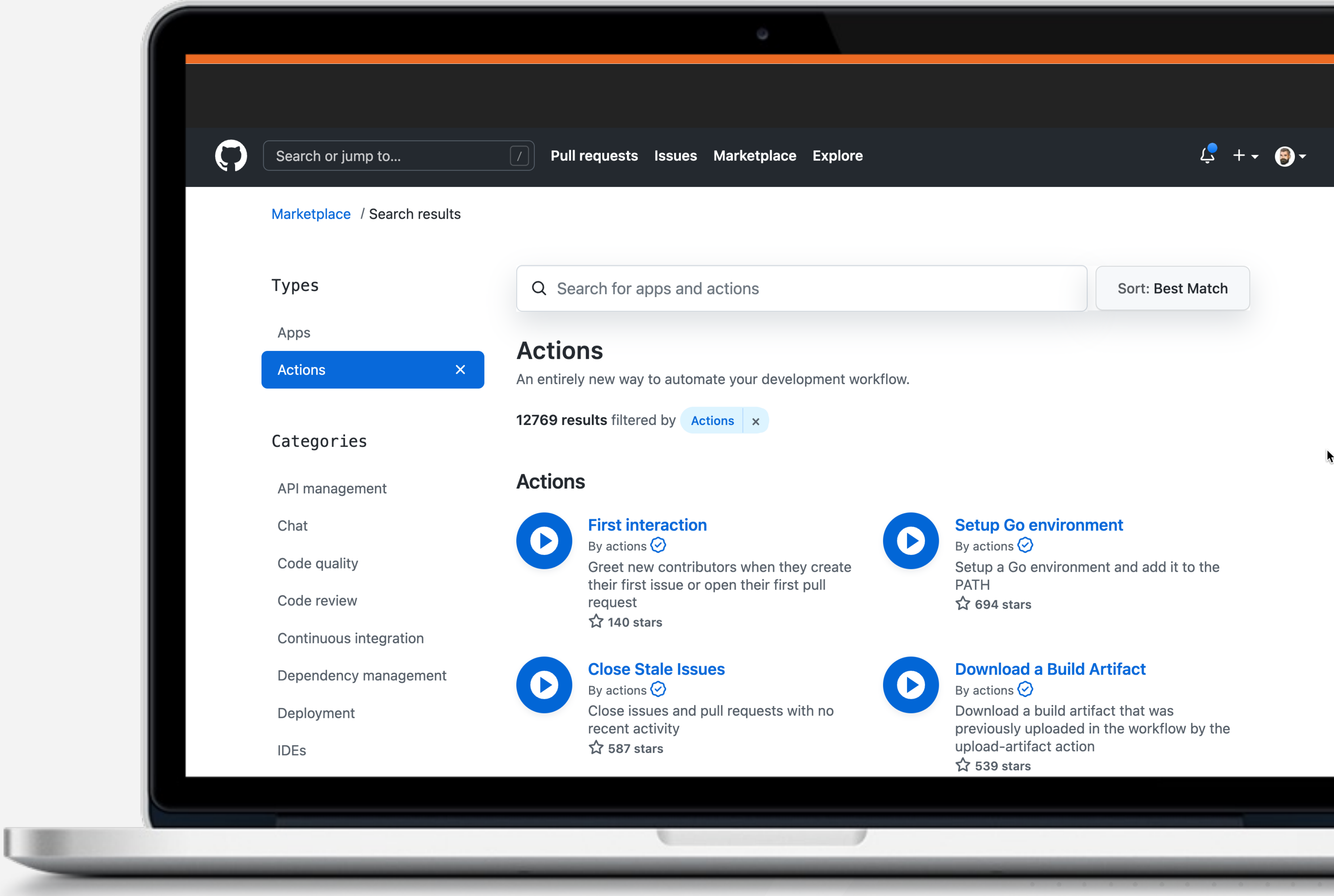
Fortune 500 companies

A different approach…



- Leverage the power of the open-source community
- Platform
- flexible

The GitHub marketplace



Search or jump to... /

Pull requests Issues Marketplace Explore

Marketplace / Search results

Types

Apps

Actions ×

Categories

API management

Chat

Code quality

Code review

Continuous integration

Dependency management

Deployment

IDEs

Search for apps and actions

Sort: Best Match

Actions

An entirely new way to automate your development workflow.

12769 results filtered by Actions ×

Actions

First interaction
By actions
Greet new contributors when they create their first issue or open their first pull request
☆ 140 stars

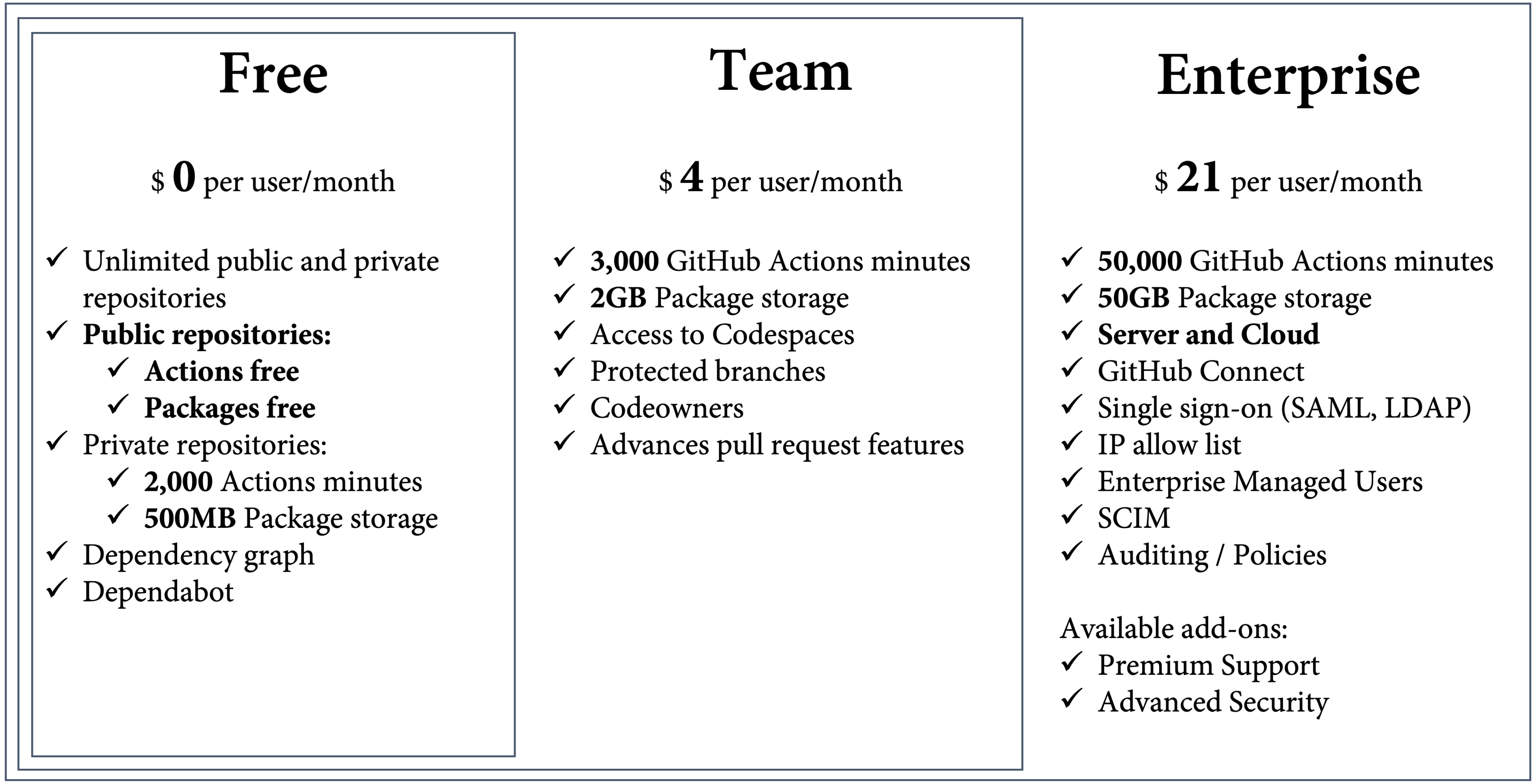
Setup Go environment
By actions
Setup a Go environment and add it to the PATH
☆ 694 stars

Close Stale Issues
By actions
Close issues and pull requests with no recent activity
☆ 587 stars

Download a Build Artifact
By actions
Download a build artifact that was previously uploaded in the workflow by the upload-artifact action
☆ 539 stars

Xpirit

Different pricing tiers



Free

$ **0** per user/month

- ✓ Unlimited public and private repositories
- ✓ **Public repositories:**
  - ✓ **Actions free**
  - ✓ **Packages free**
- ✓ Private repositories:
  - ✓ **2,000** Actions minutes
  - ✓ **500MB** Package storage
- ✓ Dependency graph
- ✓ Dependabot

Team

$ **4** per user/month

- ✓ **3,000** GitHub Actions minutes
- ✓ **2GB** Package storage
- ✓ Access to Codespaces
- ✓ Protected branches
- ✓ Codeowners
- ✓ Advances pull request features

Enterprise

$ **21** per user/month

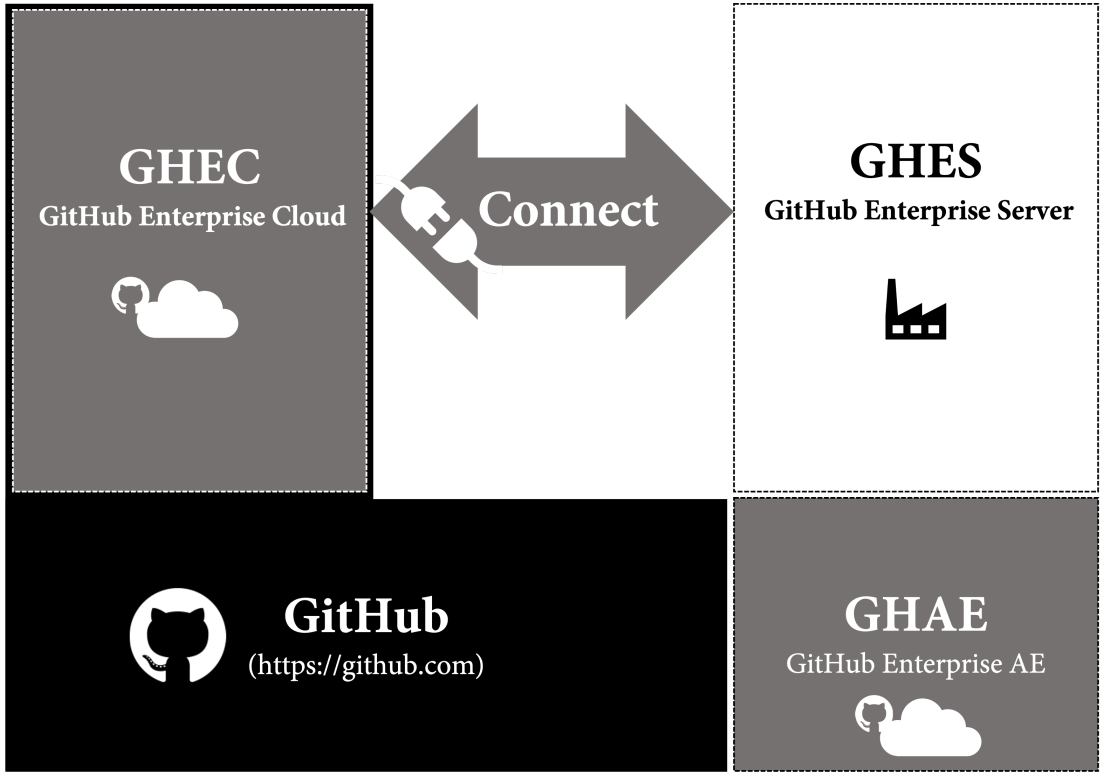
- ✓ **50,000** GitHub Actions minutes
- ✓ **50GB** Package storage
- ✓ **Server and Cloud**
- ✓ GitHub Connect
- ✓ Single sign-on (SAML, LDAP)
- ✓ IP allow list
- ✓ Enterprise Managed Users
- ✓ SCIM
- ✓ Auditing / Policies

Available add-ons:

- ✓ Premium Support
- ✓ Advanced Security

- Free
- Team
- Enterprise

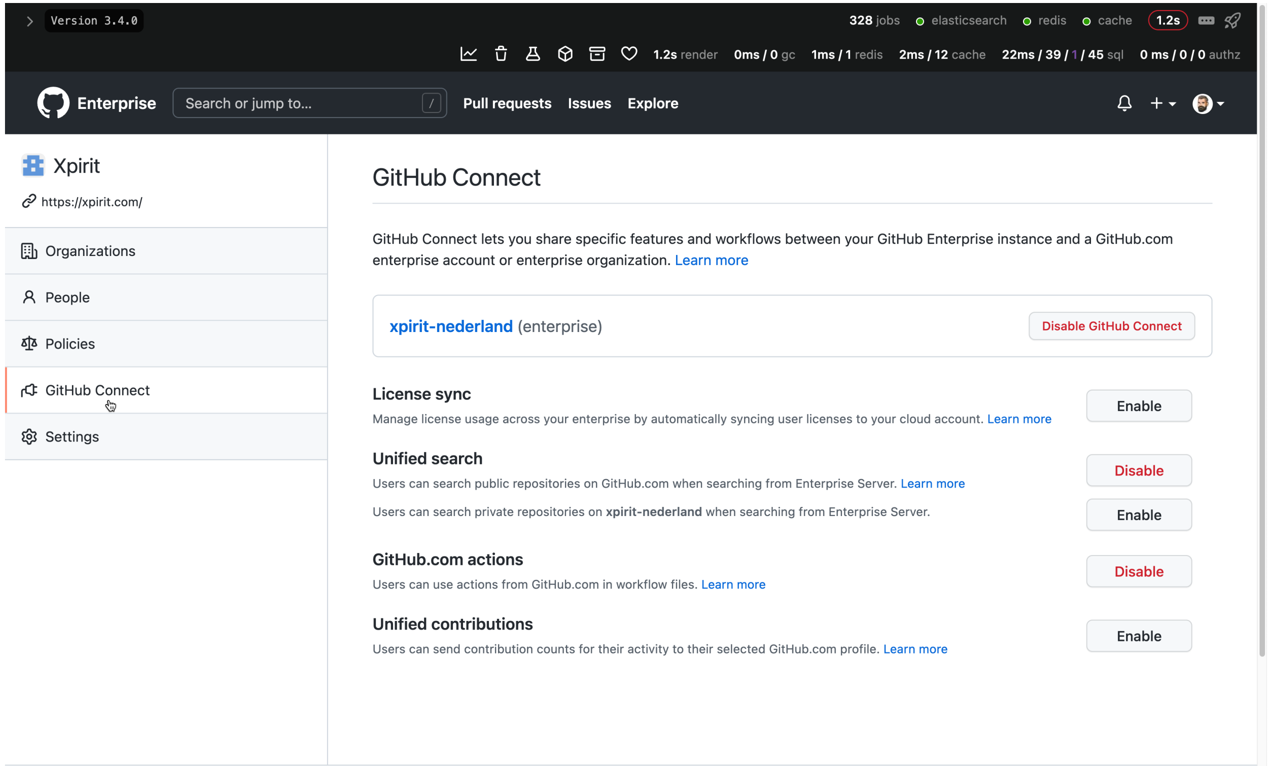
Different hosting options



- GitHub Enterprise Cloud
- GitHub Enterprise Server
- GitHub AE (private beta)

Xpirit

GitHub Connect



GitHub Connect

GitHub Connect lets you share specific features and workflows between your GitHub Enterprise instance and a GitHub.com enterprise account or enterprise organization. Learn more

xpirit-nederland (enterprise) — Disable GitHub Connect

License sync — Enable

Manage license usage across your enterprise by automatically syncing user licenses to your cloud account. Learn more

Unified search — Disable

Users can search public repositories on GitHub.com when searching from Enterprise Server. Learn more

Users can search private repositories on **xpirit-nederland** when searching from Enterprise Server. — Enable

GitHub.com actions — Disable

Users can use actions from GitHub.com in workflow files. Learn more

Unified contributions — Enable

Users can send contribution counts for their activity to their selected GitHub.com profile. Learn more

- License sync
- Unified search
- GitHub Actions and Dependencies
- Unified contributions

Xpirit

Collaborate using GitHub

Collaborate – using GitHub

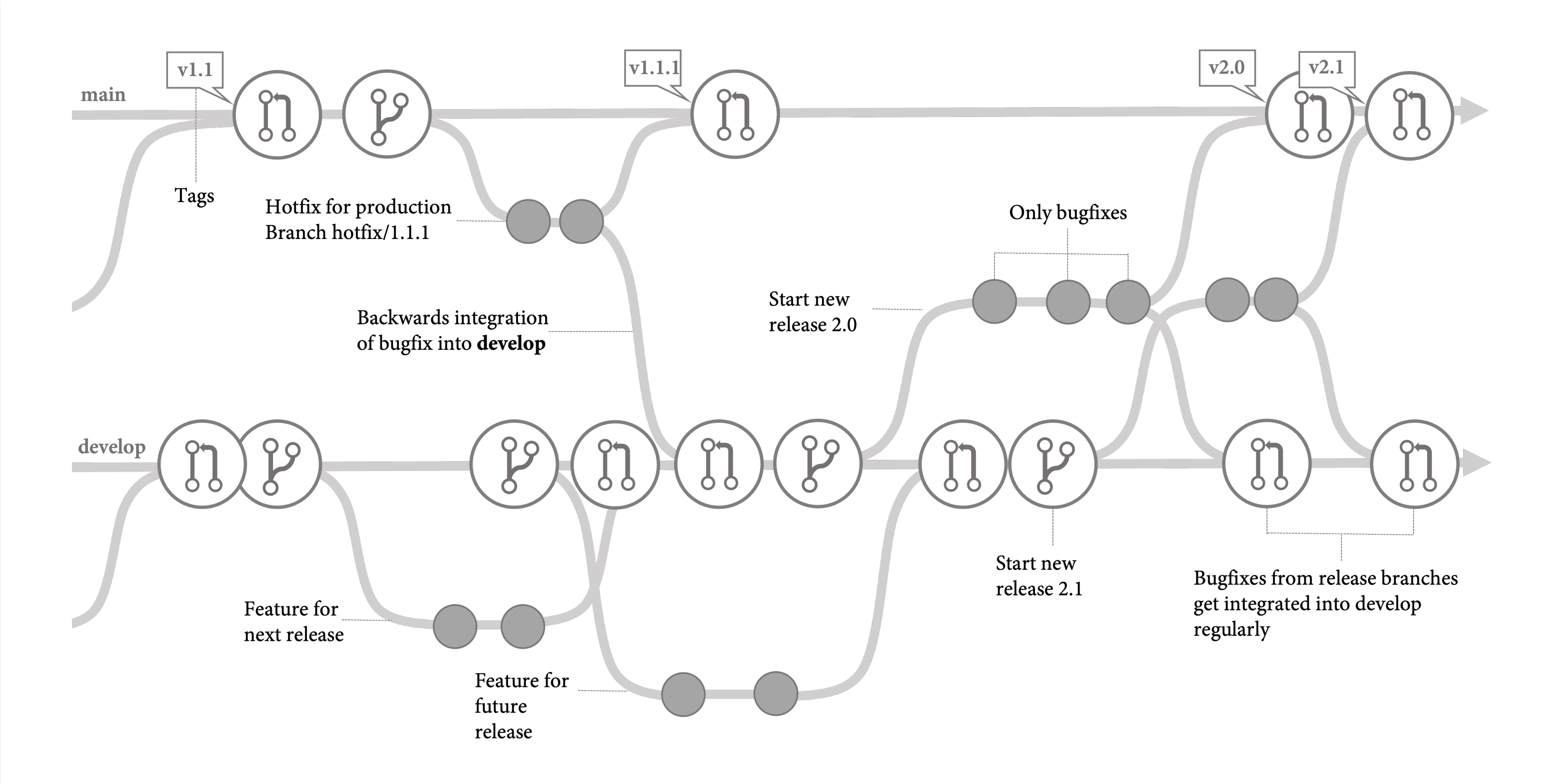
- Markdown
- Issues
- Discussions
- Wikis
- Pages

GitHub Projects (demo & HOL)

Git Workflows

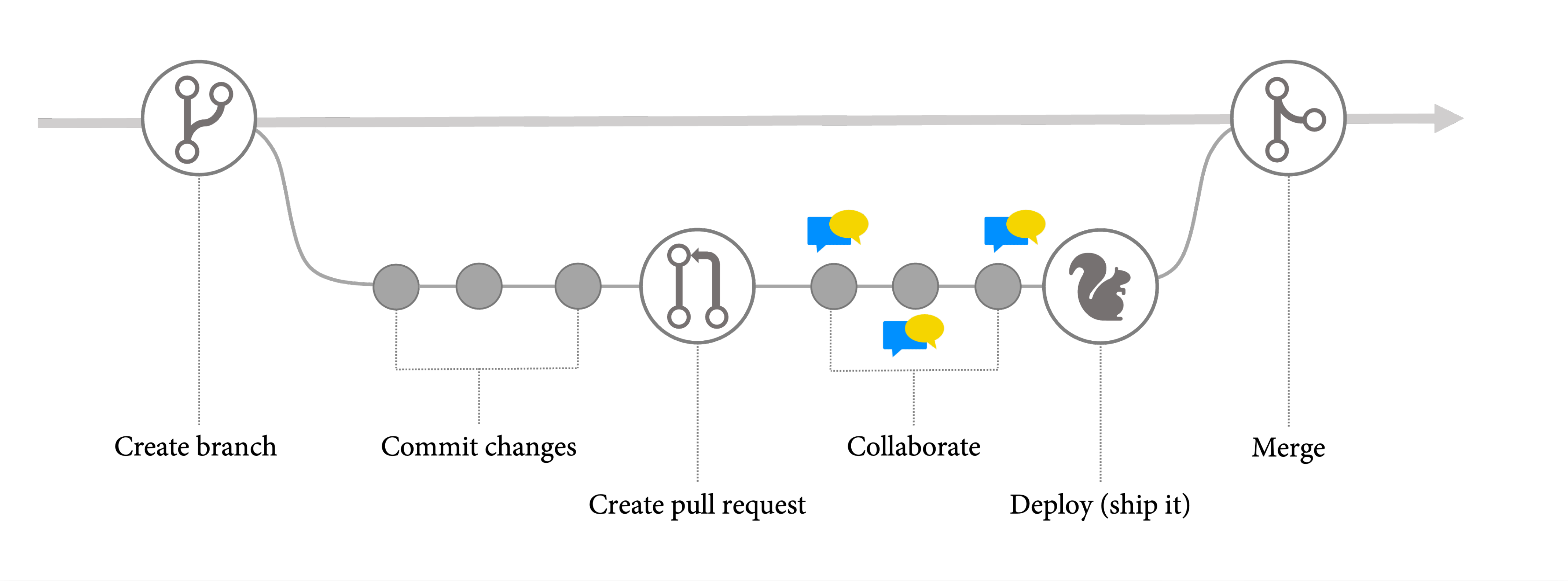
Branching workflows

- **Git flow** (https://nvie.com/posts/a-successful-git-branching-model)



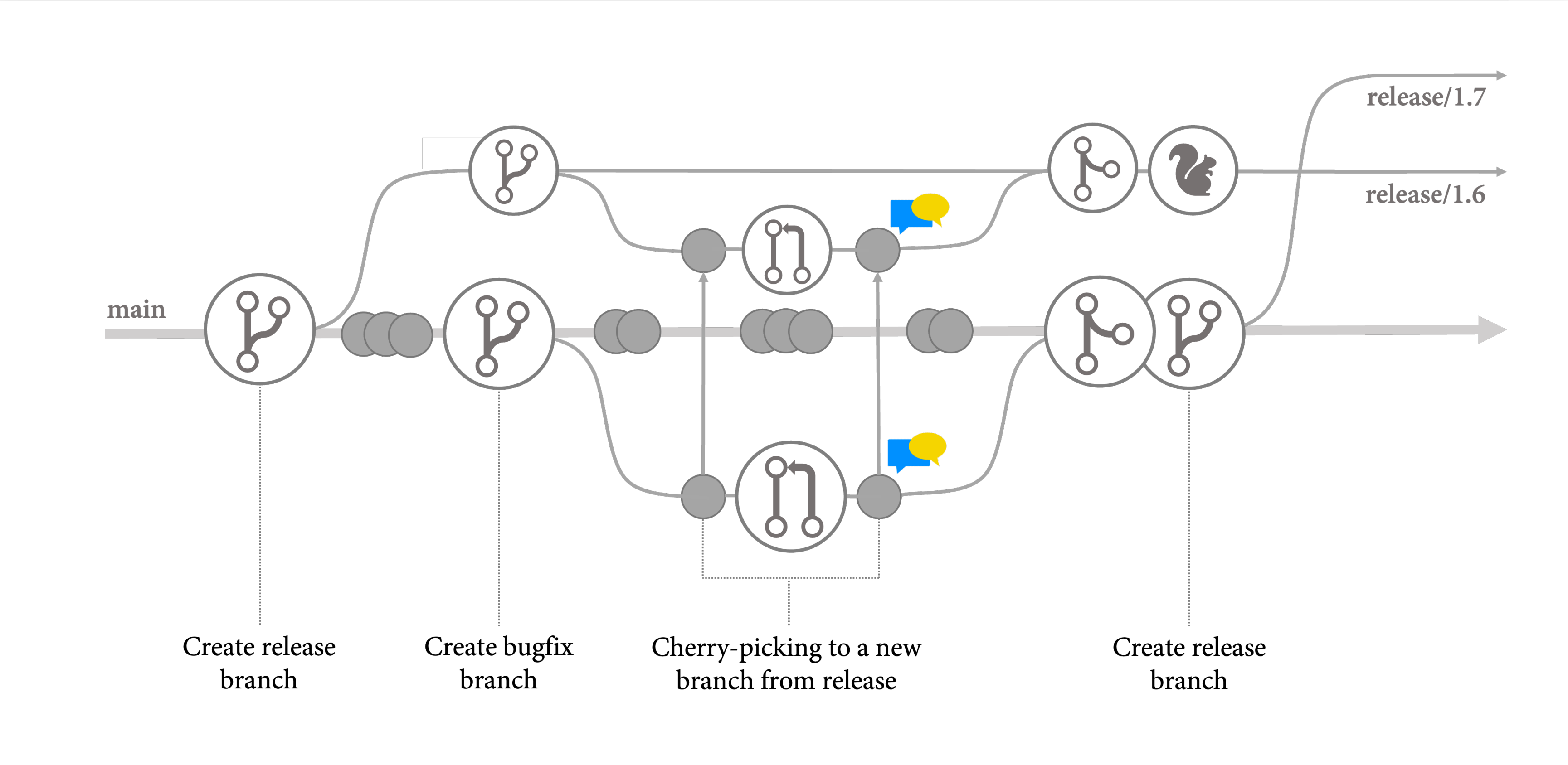
Branching workflows

- **GitHub flow** (trunk-based workflow, https://docs.github.com/en/get-started/quickstart/github-flow)



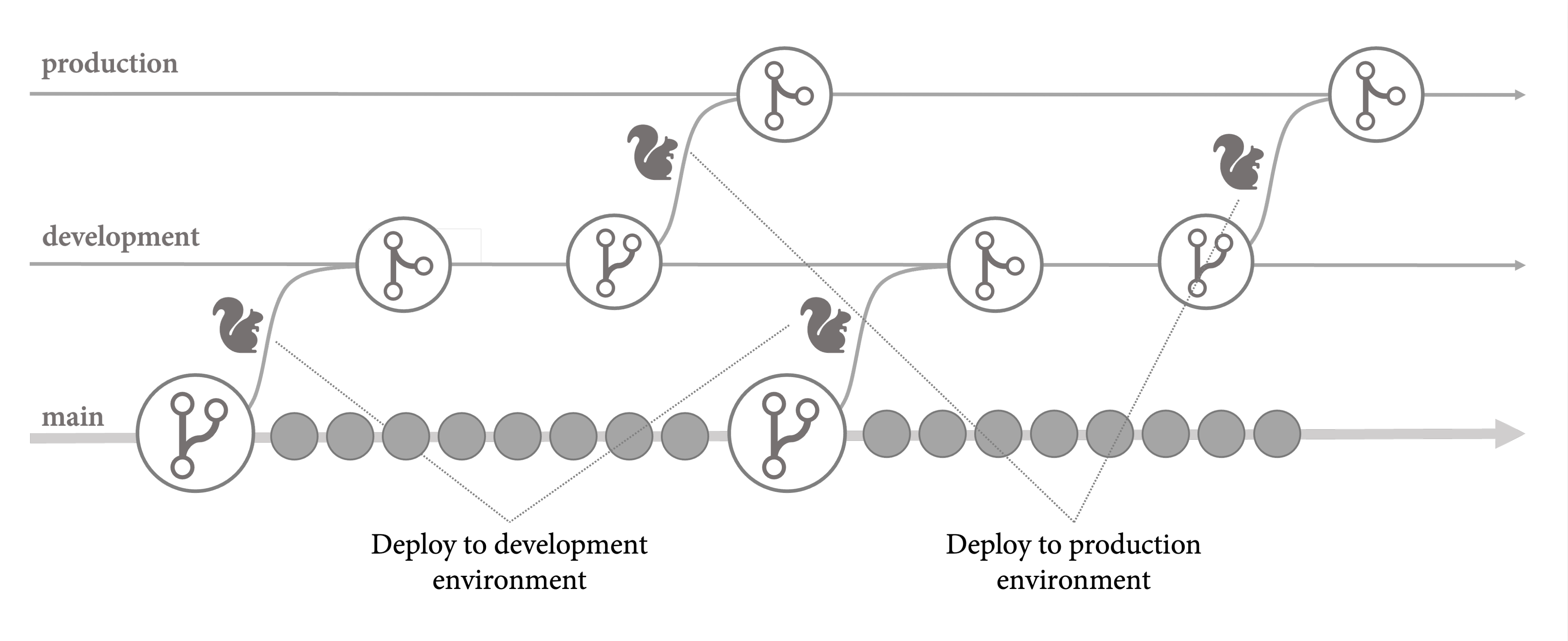
Branching workflows

- **Release flow** (https://devblogs.microsoft.com/devops/release-flow-how-we-do-branching-on-the-vsts-team/)



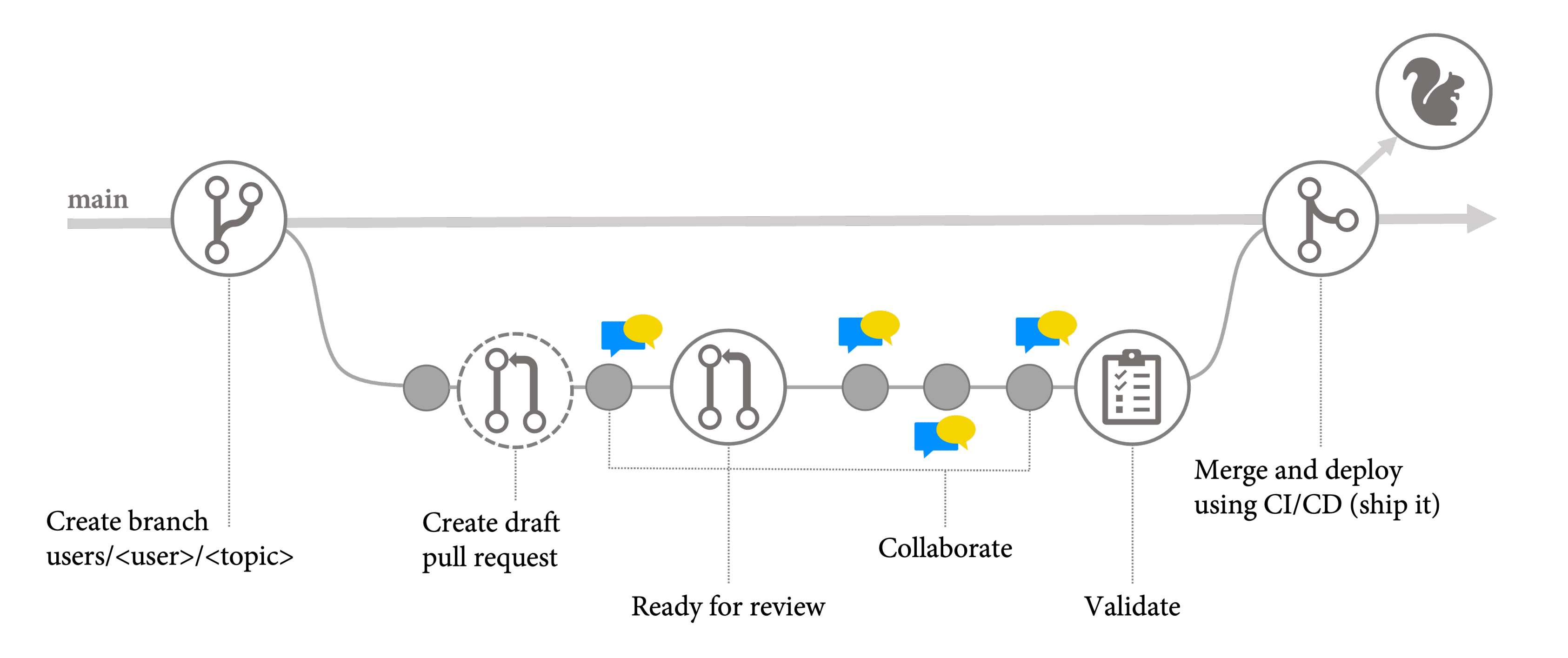
Branching workflows

- **GitLab flow** (https://docs.gitlab.com/ee/topics/gitlab_flow.html)


production
development
main
Deploy to development environment
Deploy to production environment

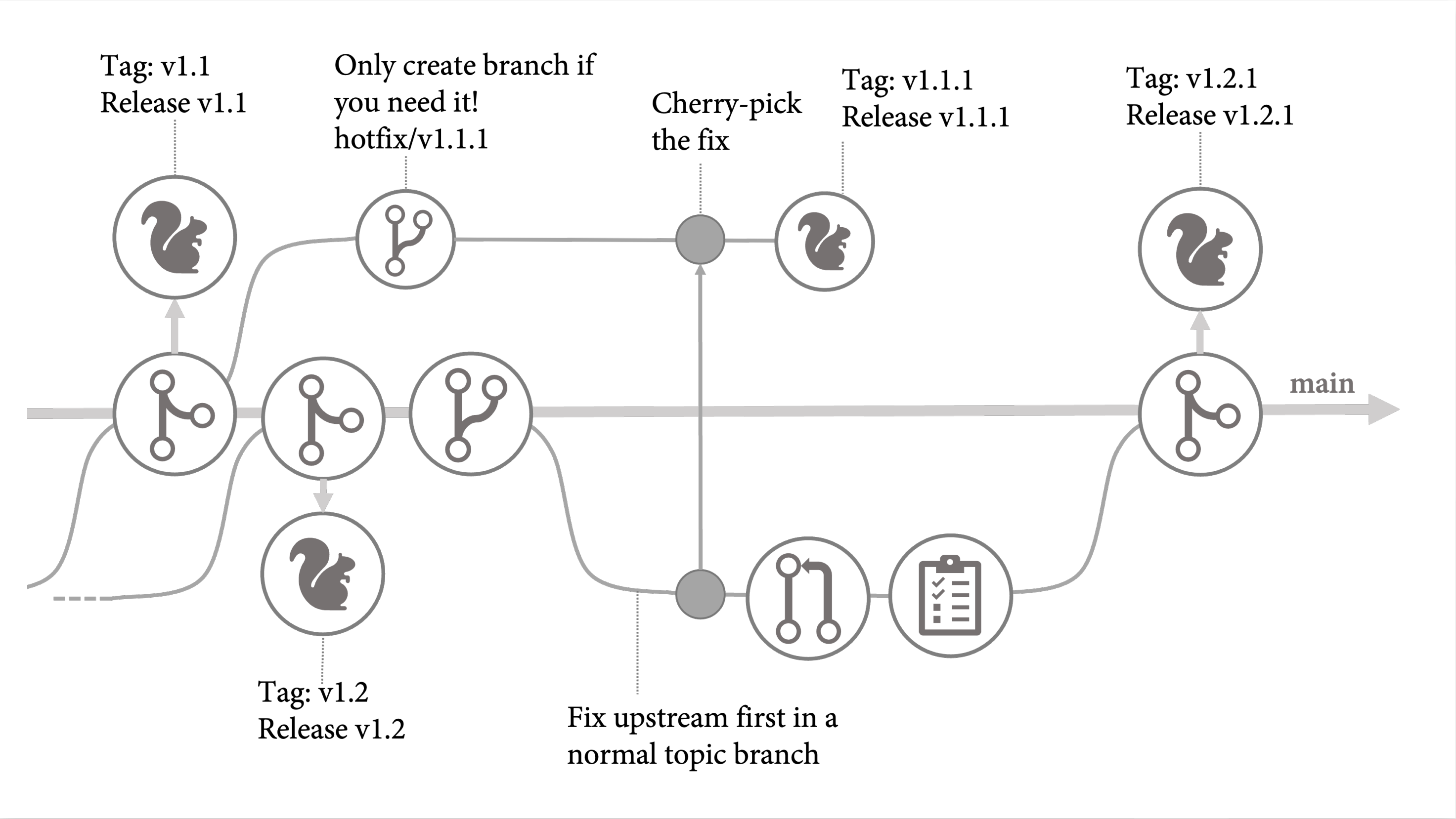
Branching workflows

- **MyFlow** (https://wulfland.github.io/MyFlow/)
  - Trunk-based (main, branch protection, CODEOWNERS)
  - Private topic branches (`users/<username>/<id>_<topic>`)
  - (Draft) pull requests, auto merge, `git push -f` / `git push origin +<branch-name>`



Branching workflows

- **MyFlow** (https://wulfland.github.io/MyFlow/)
  - Releases / Tags / cherry-pick

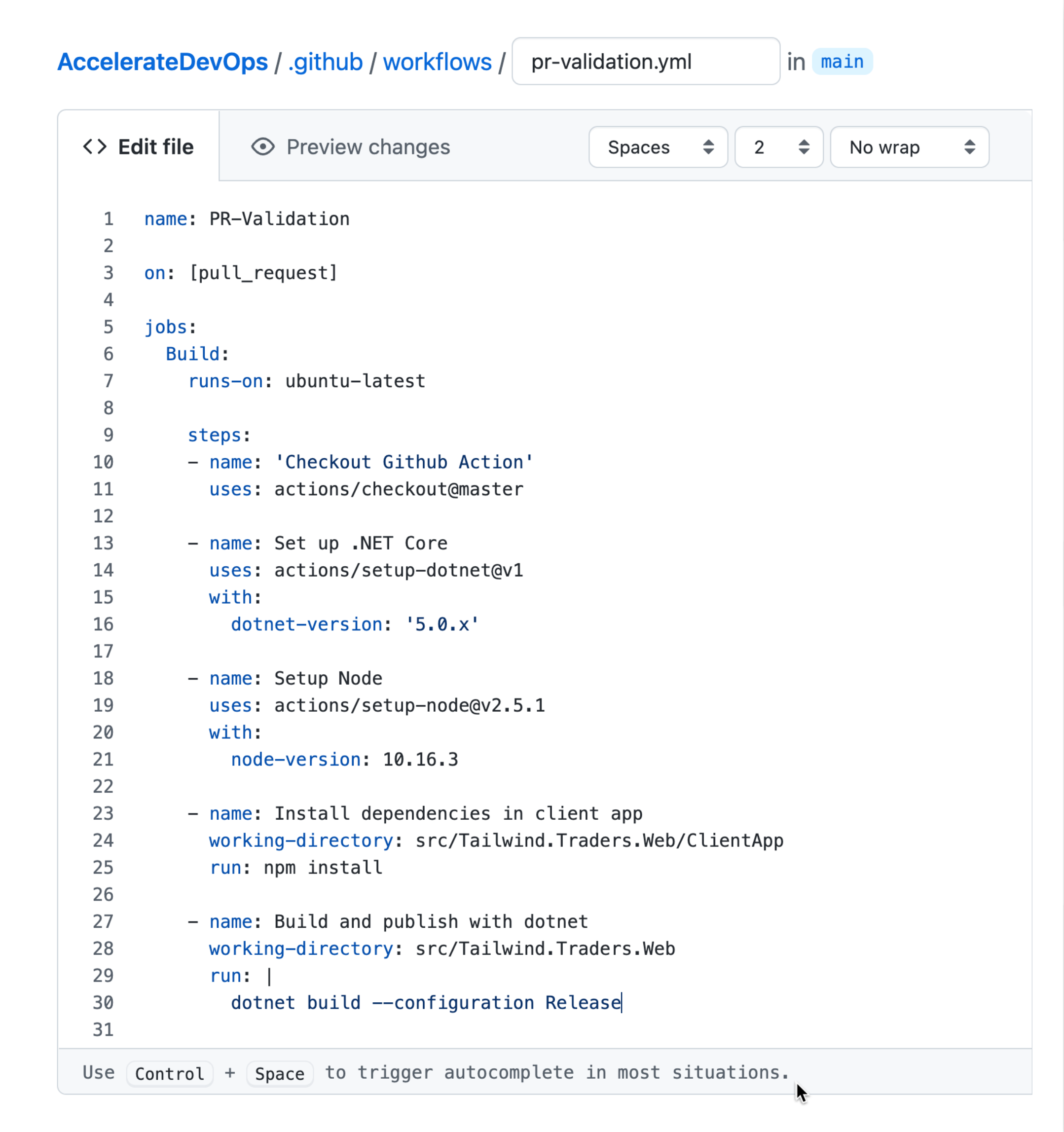


GitHub Actions - More then CI/CD

- Automate everything with workflows
- 35 events can trigger a workflow
- GitHub Token and Workflow Permissions
- Community-powered workflows
- Any platform, any language, any cloud

Workflows

- A text file in your repository (.github/workflows)
- YAML Ain't Markup Language **(YAML)**
- Events trigger workflows **(on:)**
- One or multiple **jobs**
- Executed on a **runner**
- Contains **steps**
- A reusable step is **action**


AccelerateDevOps / .github / workflows / pr-validation.yml in main
Edit file
Preview changes
Spaces
2
No wrap

```yaml
name: PR-Validation

on: [pull_request]

jobs:
  Build:
    runs-on: ubuntu-latest

    steps:
    - name: 'Checkout Github Action'
      uses: actions/checkout@master

    - name: Set up .NET Core
      uses: actions/setup-dotnet@v1
      with:
        dotnet-version: '5.0.x'

    - name: Setup Node
      uses: actions/setup-node@v2.5.1
      with:
        node-version: 10.16.3

    - name: Install dependencies in client app
      working-directory: src/Tailwind.Traders.Web/ClientApp
      run: npm install

    - name: Build and publish with dotnet
      working-directory: src/Tailwind.Traders.Web
      run: |
        dotnet build --configuration Release

```

Use Control + Space to trigger autocomplete in most situations.

Xpirit

YAML

YAML basics

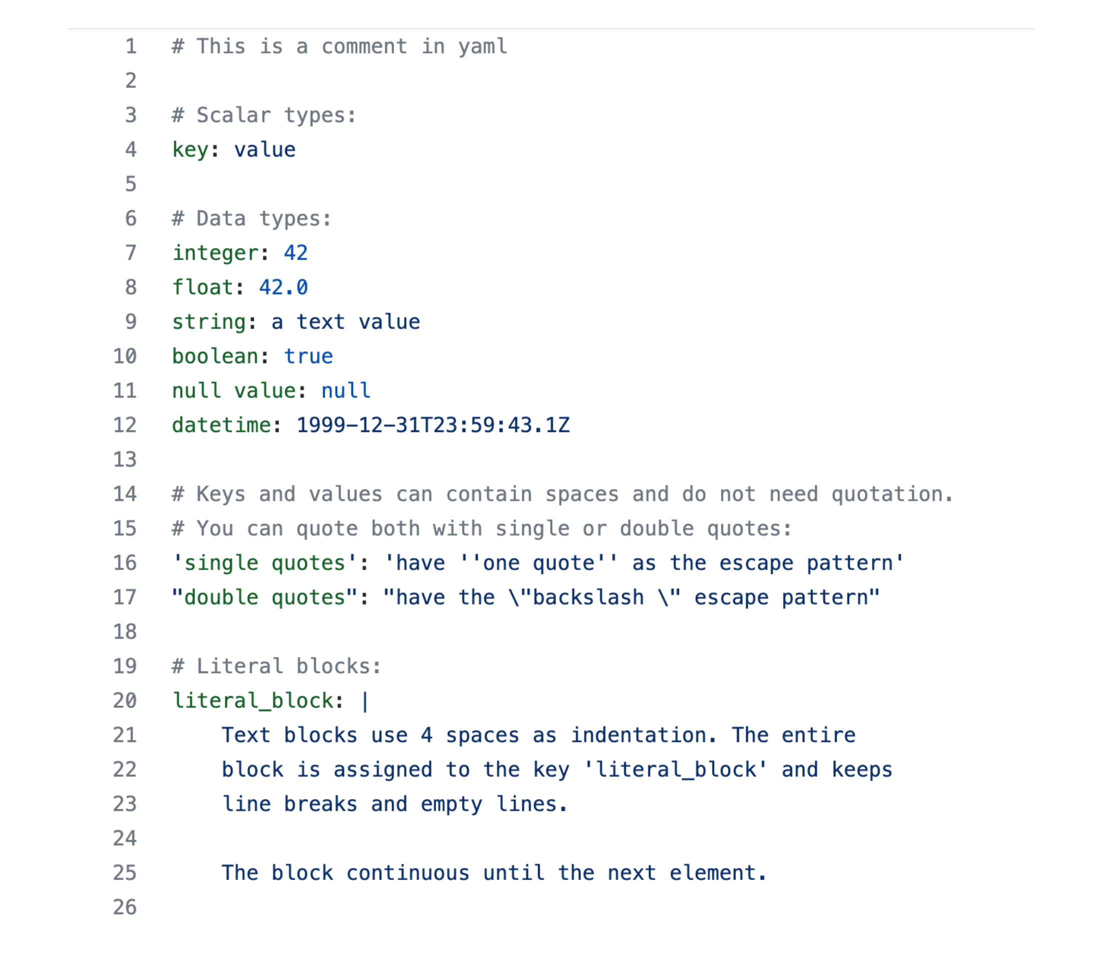
Extension: .yml or .yaml

A strict superset of **JSON**

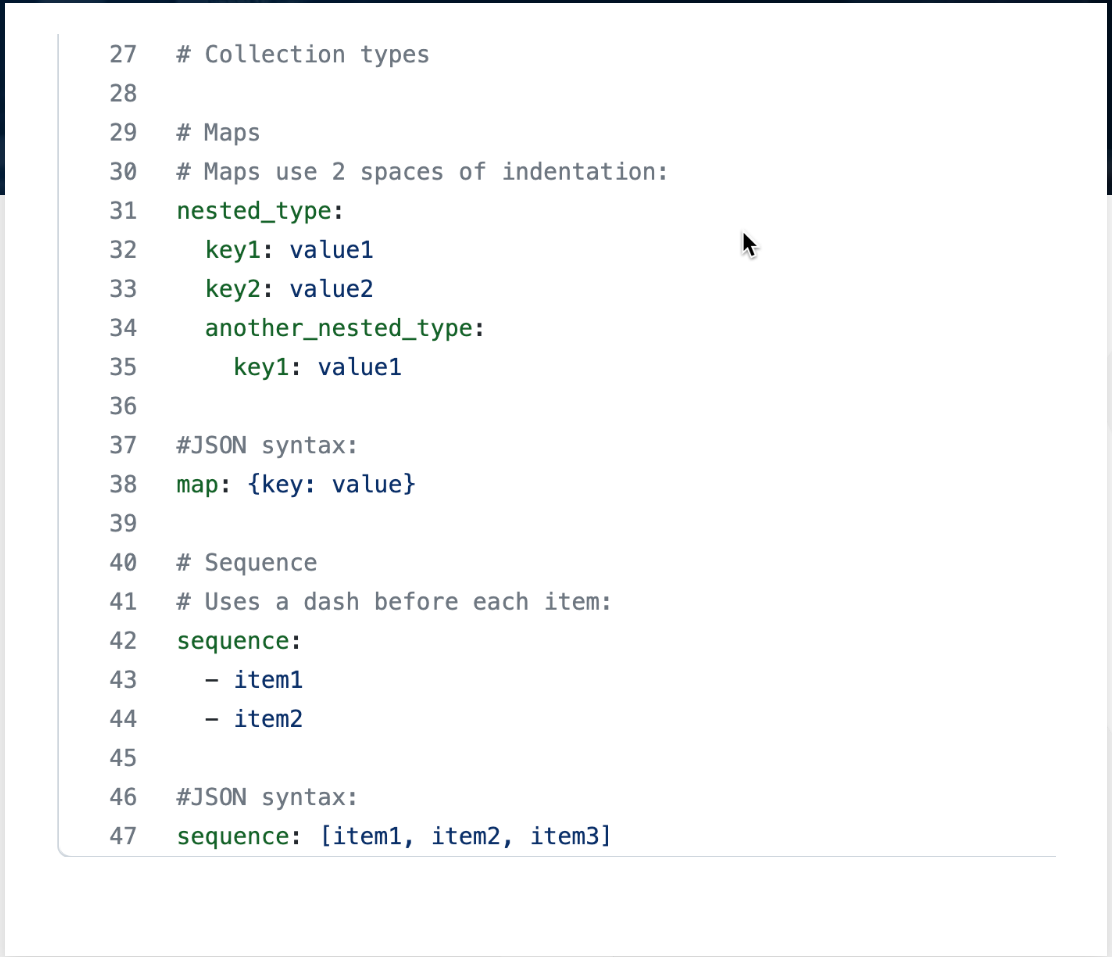
Data-serialization language writable and readable by humans

Contains syntactically relevant **newlines** and **indentation** instead of braces

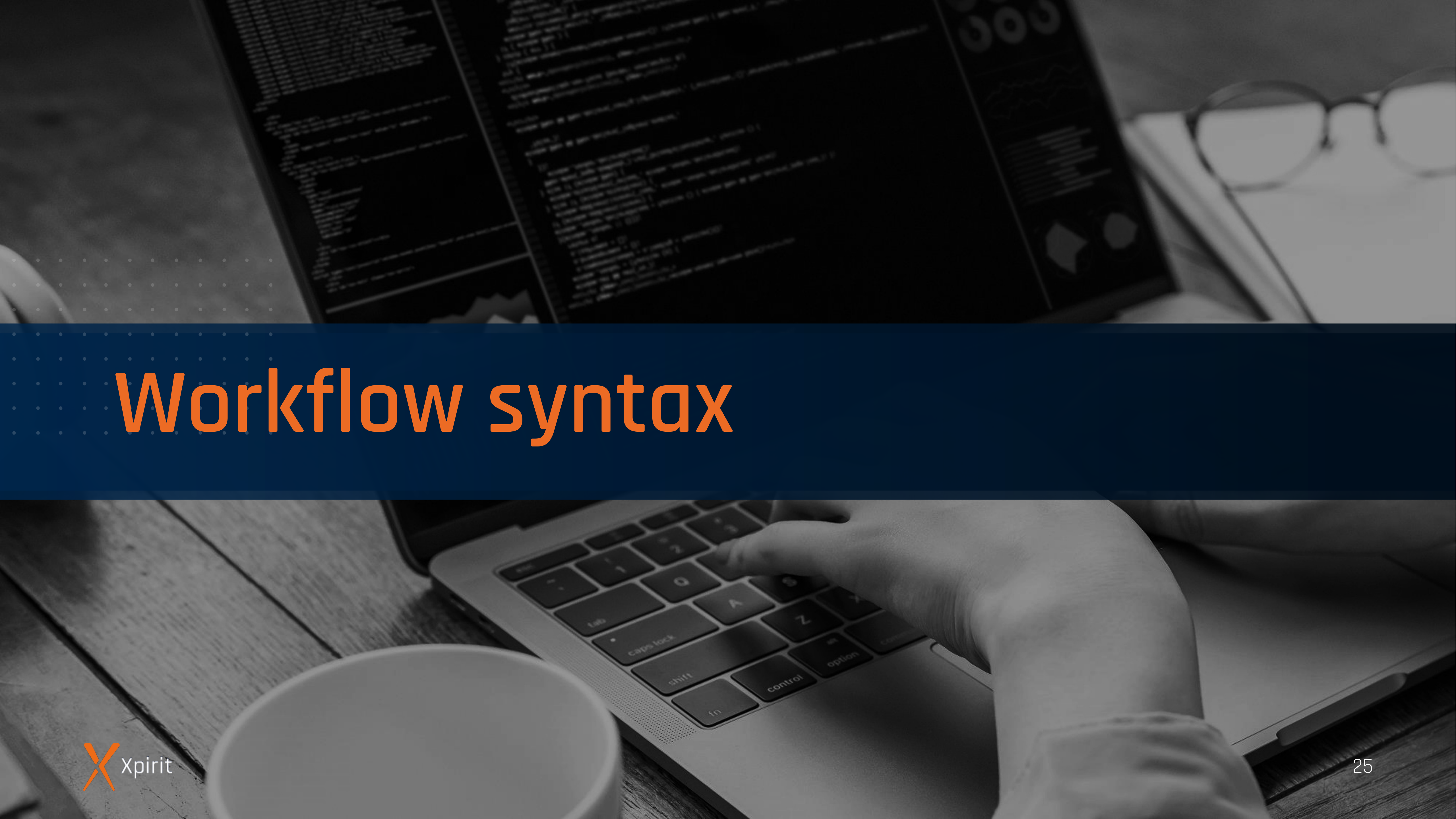
YAML basics



```yaml
# This is a comment in yaml

# Scalar types:
key: value

# Data types:
integer: 42
float: 42.0
string: a text value
boolean: true
null value: null
datetime: 1999-12-31T23:59:43.1Z

# Keys and values can contain spaces and do not need quotation.
# You can quote both with single or double quotes:
'single quotes': 'have ''one quote'' as the escape pattern'
"double quotes": "have the \"backslash \" escape pattern"

# Literal blocks:
literal_block: |
    Text blocks use 4 spaces as indentation. The entire
    block is assigned to the key 'literal_block' and keeps
    line breaks and empty lines.

    The block continuous until the next element.

```



```yaml
# Collection types

# Maps
# Maps use 2 spaces of indentation:
nested_type:
  key1: value1
  key2: value2
  another_nested_type:
    key1: value1

#JSON syntax:
map: {key: value}

# Sequence
# Uses a dash before each item:
sequence:
  - item1
  - item2

#JSON syntax:
sequence: [item1, item2, item3]
```

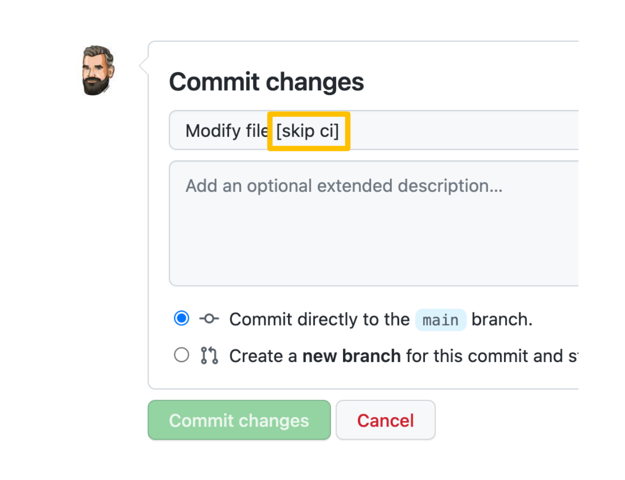


Workflow syntax

Name and Triggers

- Name of the workflow

- Trigger:
  - Webhook events
  - Scheduled events
  - Manual events



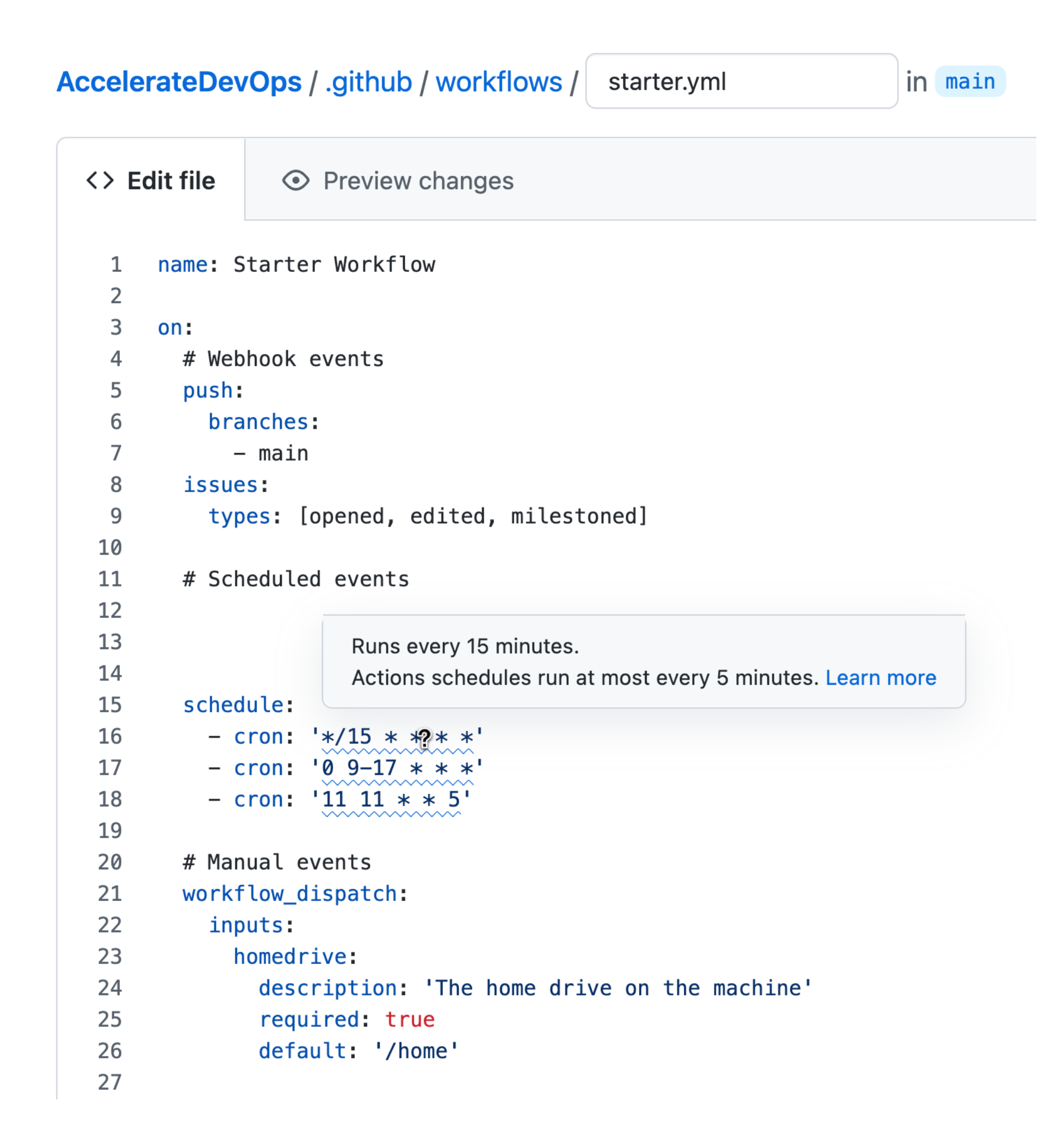
Commit changes

Modify file [skip ci]

Add an optional extended description...

● Commit directly to the main branch.

○ Create a new branch for this commit and s



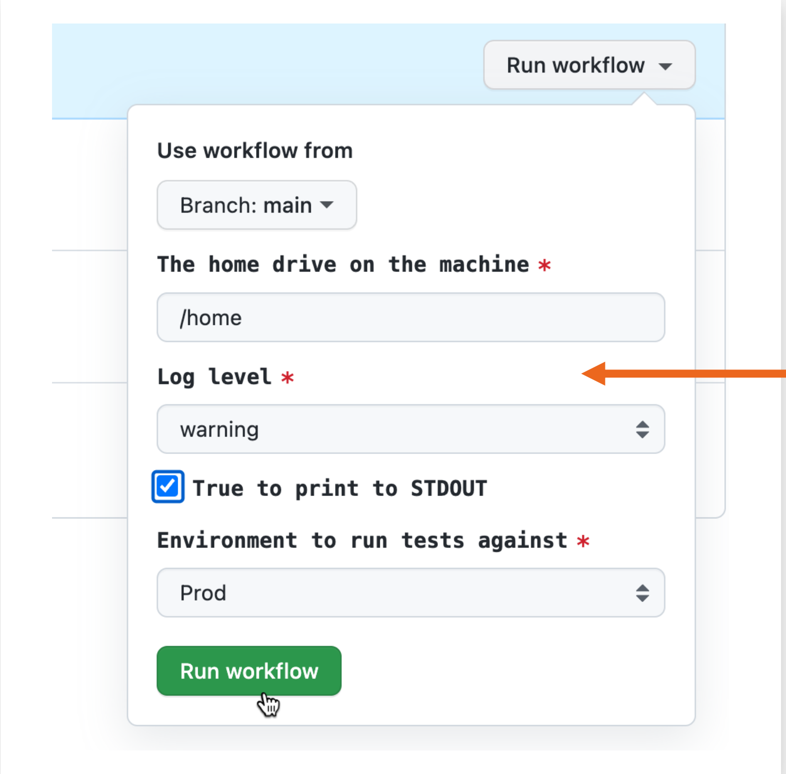
AccelerateDevOps / .github / workflows / starter.yml in main

```yaml
name: Starter Workflow

on:
  # Webhook events
  push:
    branches:
      - main
  issues:
    types: [opened, edited, milestoned]

  # Scheduled events



  schedule:
    - cron: '*/15 * * * *'
    - cron: '0 9-17 * * *'
    - cron: '11 11 * * 5'

  # Manual events
  workflow_dispatch:
    inputs:
      homedrive:
        description: 'The home drive on the machine'
        required: true
        default: '/home'

```

Runs every 15 minutes.
Actions schedules run at most every 5 minutes. Learn more

Name and Triggers

Manual events



Run workflow ▾

Use workflow from

Branch: main ▾

The home drive on the machine *

/home

Log level *

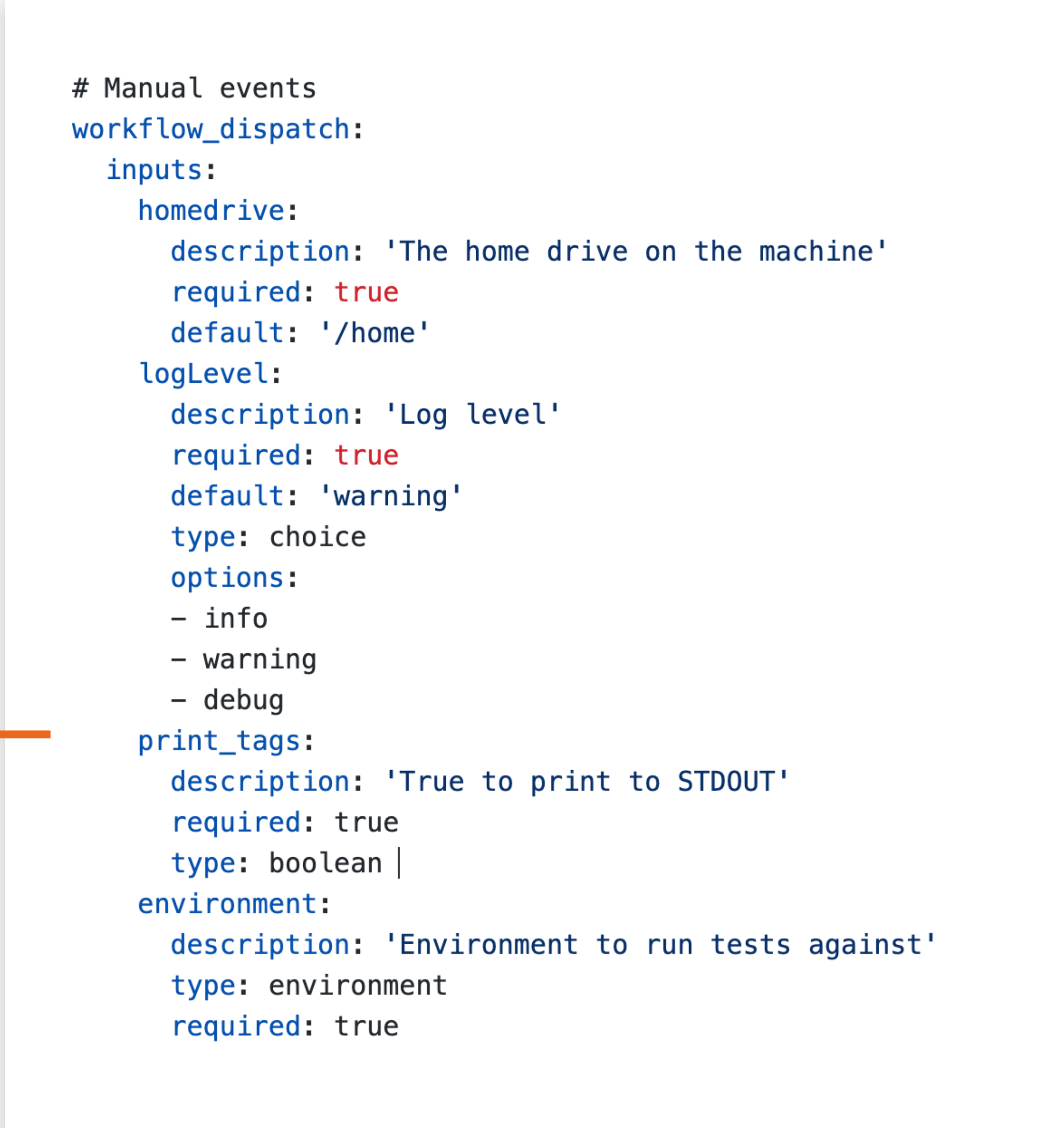
warning

☑ True to print to STDOUT

Environment to run tests against *

Prod

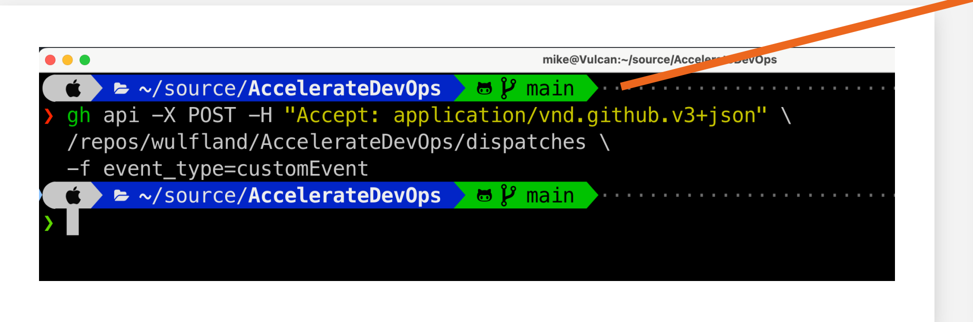
Run workflow



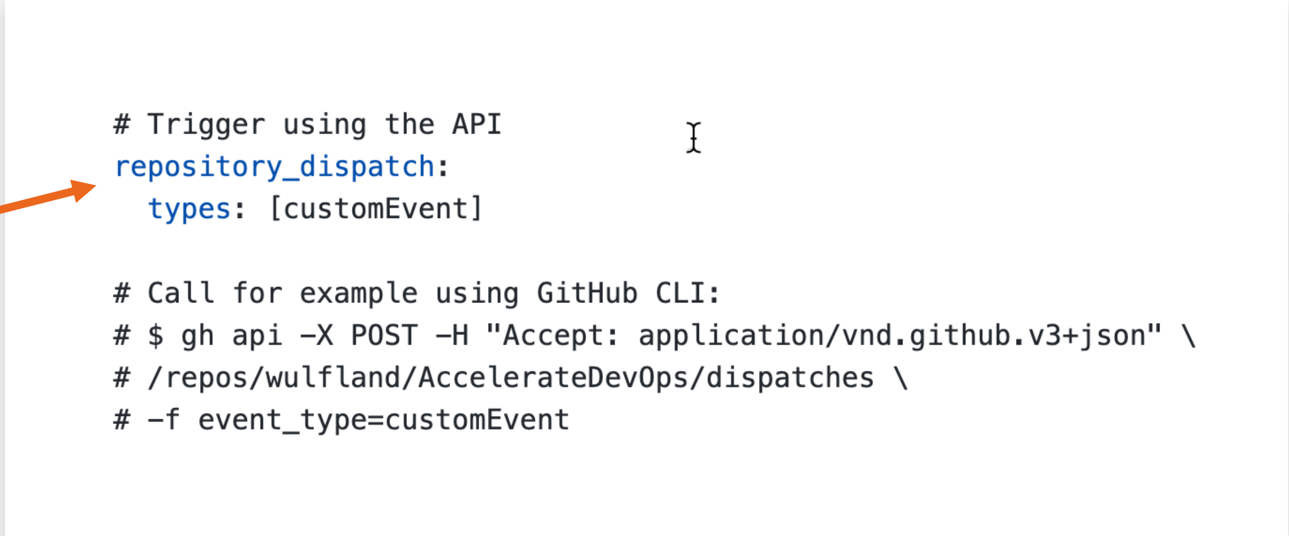
```
# Manual events
workflow_dispatch:
  inputs:
    homedrive:
      description: 'The home drive on the machine'
      required: true
      default: '/home'
    logLevel:
      description: 'Log level'
      required: true
      default: 'warning'
      type: choice
      options:
      - info
      - warning
      - debug
    print_tags:
      description: 'True to print to STDOUT'
      required: true
      type: boolean |
    environment:
      description: 'Environment to run tests against'
      type: environment
      required: true
```

Name and Triggers

Manual events: trigger using the API
(curl, octokit, GitHub CLI)



```
 ~/source/AccelerateDevOps  main
❯ gh api -X POST -H "Accept: application/vnd.github.v3+json" \
  /repos/wulfland/AccelerateDevOps/dispatches \
  -f event_type=customEvent
 ~/source/AccelerateDevOps  main
❯
```

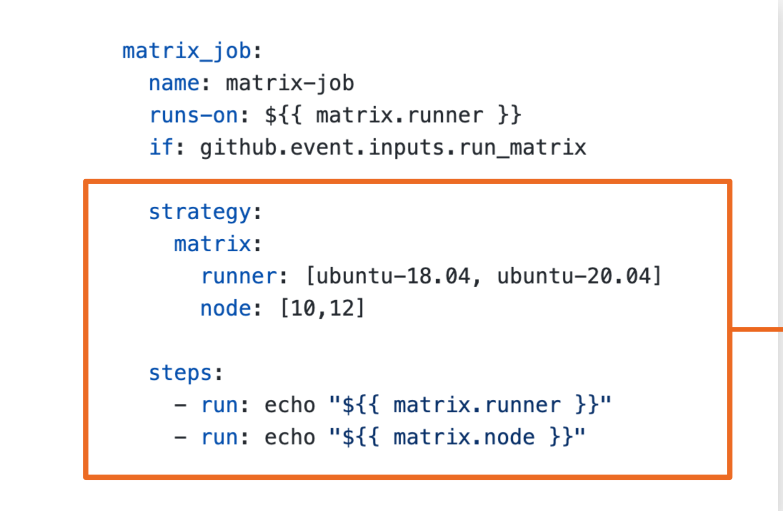


```
# Trigger using the API
repository_dispatch:
  types: [customEvent]

# Call for example using GitHub CLI:
# $ gh api -X POST -H "Accept: application/vnd.github.v3+json" \
# /repos/wulfland/AccelerateDevOps/dispatches \
# -f event_type=customEvent
```

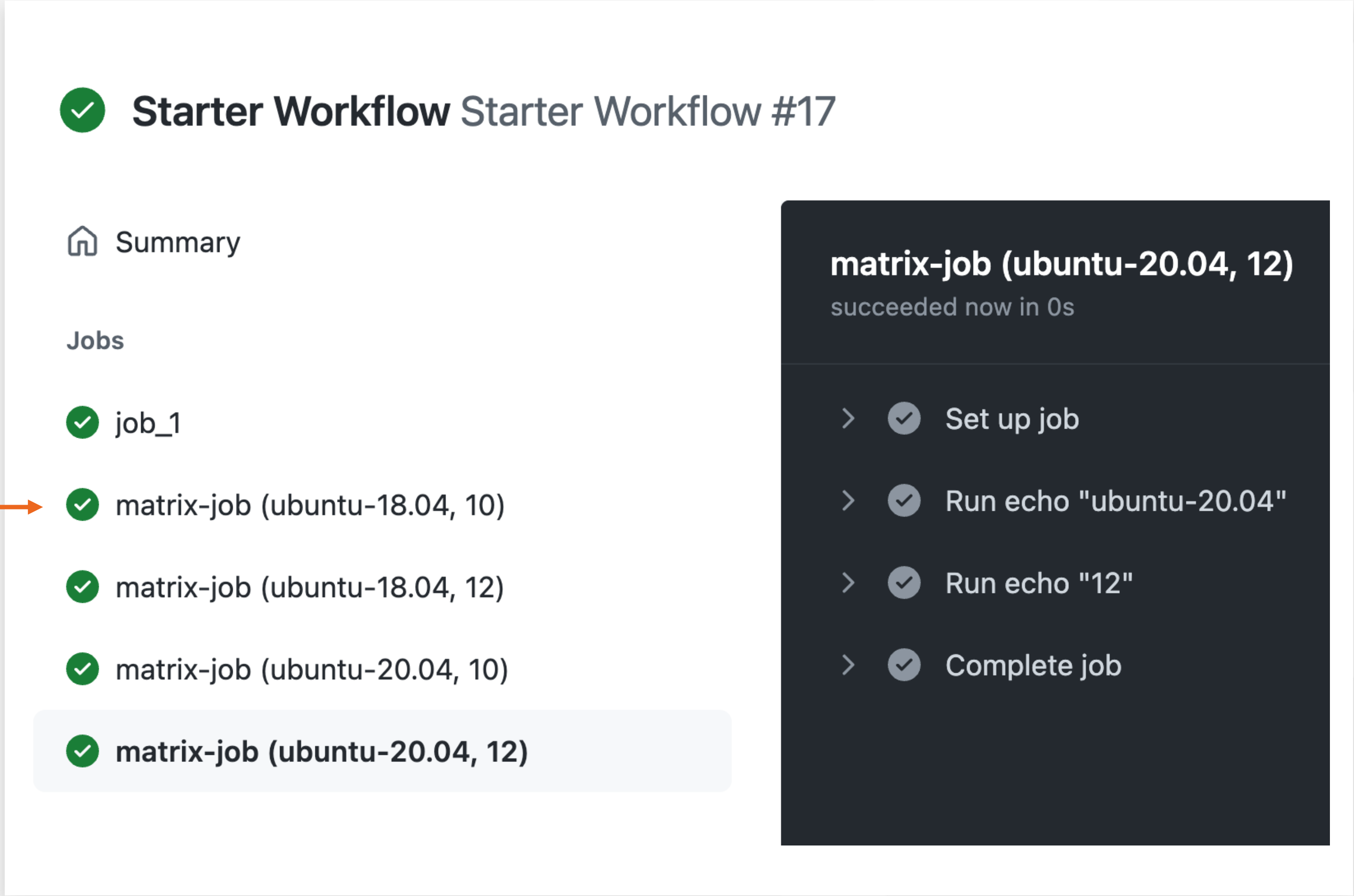
Xpirit

Strategy

- For-loop – array
- Nested for-loops: multidimensional array
- Runs for all combinations in all dimensions
- Fail-fast (yes/no)
- May parallel jobs



```yaml
matrix_job:
  name: matrix-job
  runs-on: ${{ matrix.runner }}
  if: github.event.inputs.run_matrix

  strategy:
    matrix:
      runner: [ubuntu-18.04, ubuntu-20.04]
      node: [10,12]

  steps:
    - run: echo "${{ matrix.runner }}"
    - run: echo "${{ matrix.node }}"
```



Starter Workflow Starter Workflow #17

Summary

Jobs

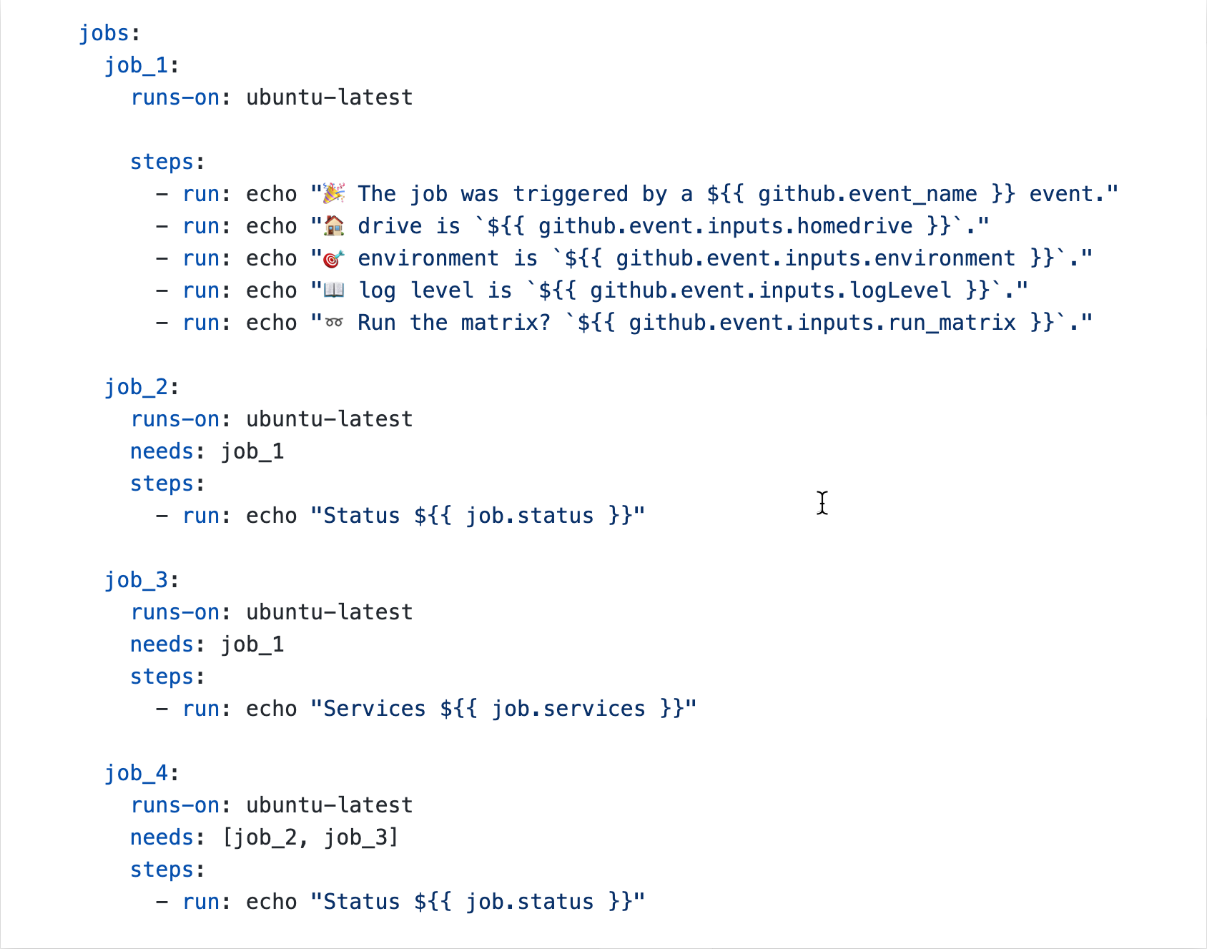
- ✓ job_1
- ✓ matrix-job (ubuntu-18.04, 10)
- ✓ matrix-job (ubuntu-18.04, 12)
- ✓ matrix-job (ubuntu-20.04, 10)
- ✓ **matrix-job (ubuntu-20.04, 12)**

matrix-job (ubuntu-20.04, 12)
succeeded now in 0s

- ✓ Set up job
- ✓ Run echo "ubuntu-20.04"
- ✓ Run echo "12"
- ✓ Complete job

Workflow jobs

- Map - run in parallel by default
- Can be chained using needs keyword
- Runs on a runner in one process
- Contains a sequence of steps

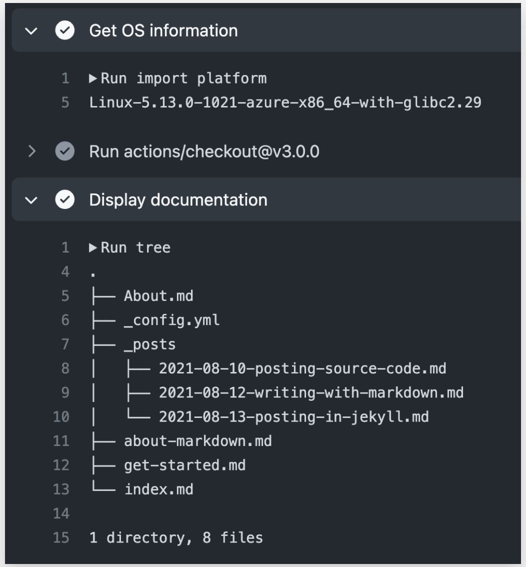


```yaml
jobs:
  job_1:
    runs-on: ubuntu-latest

    steps:
      - run: echo "🎉 The job was triggered by a ${{ github.event_name }} event."
      - run: echo "🏠 drive is `${{ github.event.inputs.homedrive }}`."
      - run: echo "🎯 environment is `${{ github.event.inputs.environment }}`."
      - run: echo "📖 log level is `${{ github.event.inputs.logLevel }}`."
      - run: echo "👓 Run the matrix? `${{ github.event.inputs.run_matrix }}`."

  job_2:
    runs-on: ubuntu-latest
    needs: job_1
    steps:
      - run: echo "Status ${{ job.status }}"

  job_3:
    runs-on: ubuntu-latest
    needs: job_1
    steps:
      - run: echo "Services ${{ job.services }}"

  job_4:
    runs-on: ubuntu-latest
    needs: [job_2, job_3]
    steps:
      - run: echo "Status ${{ job.status }}"
```

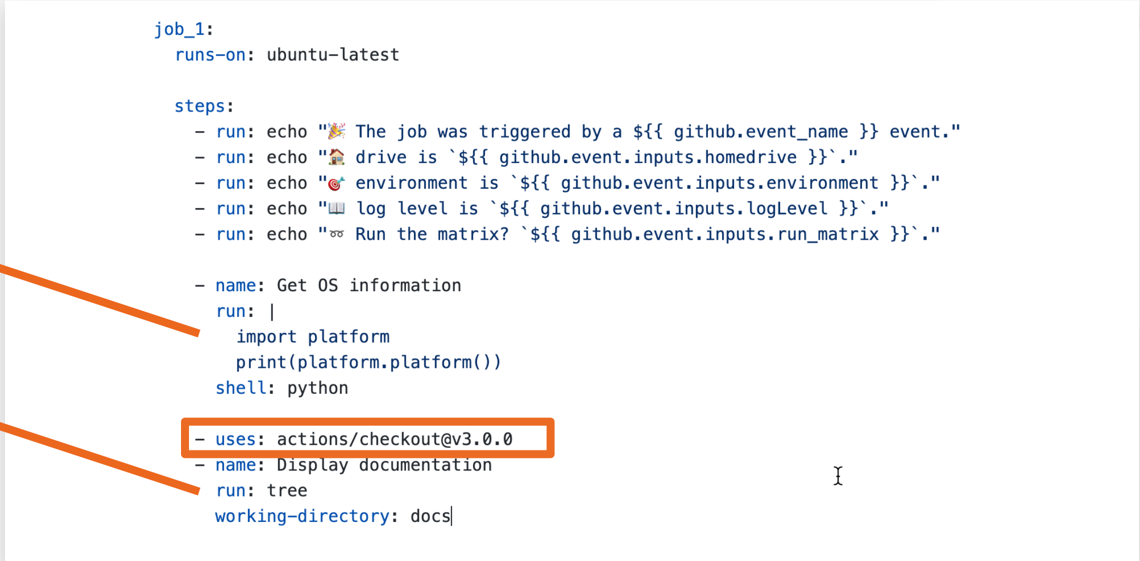
Workflow steps

- Sequence in a job
- Run in the same process / same directory
- Run in a shell

Parameter	Description
bash	Bash shell. The default shell on all non-Windows platforms with a fallback to sh . When specified on Windows, the bash shell included with Git is used.
pwsh	PowerShell Core. Default on the Windows platform.
python	The python shell. Allows you to run python scripts
cmd	Windows only! The windows command prompt.
powershell	Windows only! The classical Windows PowerShell.



```
⌄ ✓ Get OS information

  ▶Run import platform
  Linux-5.13.0-1021-azure-x86_64-with-glibc2.29

› ✓ Run actions/checkout@v3.0.0

⌄ ✓ Display documentation

  ▶Run tree
  .
  ├── About.md
  ├── _config.yml
  ├── _posts
  │   ├── 2021-08-10-posting-source-code.md
  │   ├── 2021-08-12-writing-with-markdown.md
  │   └── 2021-08-13-posting-in-jekyll.md
  ├── about-markdown.md
  ├── get-started.md
  └── index.md

  1 directory, 8 files
```

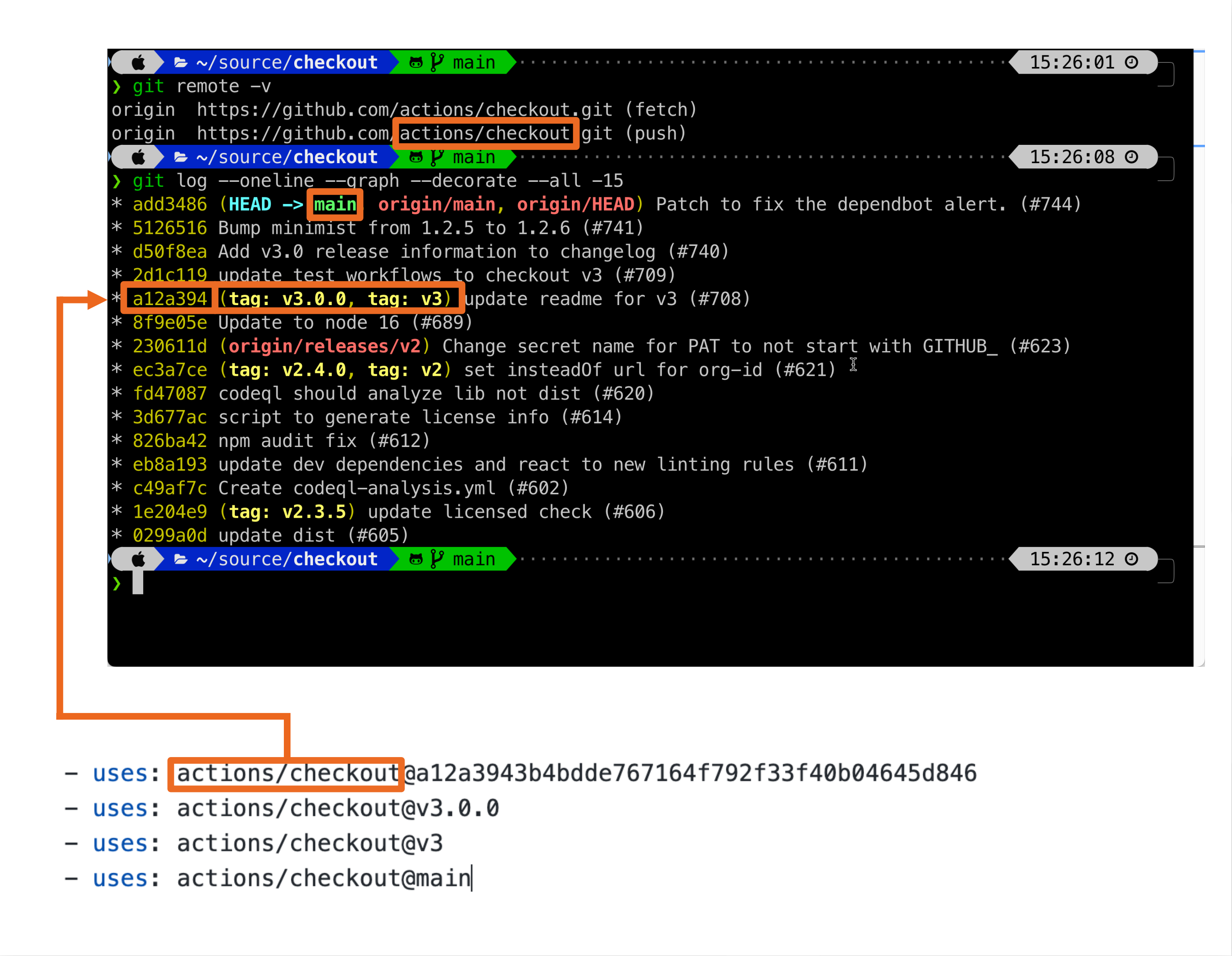


```yaml
job_1:
  runs-on: ubuntu-latest

  steps:
    - run: echo "🎉 The job was triggered by a ${{ github.event_name }} event."
    - run: echo "🏠 drive is `${{ github.event.inputs.homedrive }}`."
    - run: echo "🎯 environment is `${{ github.event.inputs.environment }}`."
    - run: echo "📖 log level is `${{ github.event.inputs.logLevel }}`."
    - run: echo "👓 Run the matrix? `${{ github.event.inputs.run_matrix }}`."

    - name: Get OS information
      run: |
        import platform
        print(platform.platform())
      shell: python

    - uses: actions/checkout@v3.0.0
    - name: Display documentation
      run: tree
      working-directory: docs
```

Xpirit

Actions

- A reusable step
- Lives in a git repo
- Syntax:
  - {owner}/{repo}@{ref}
  - {owner}/{repo}/{path}@{ref}
  - ./.github/actions/my-action
- References:
  - SHA
  - Tag
  - Branch



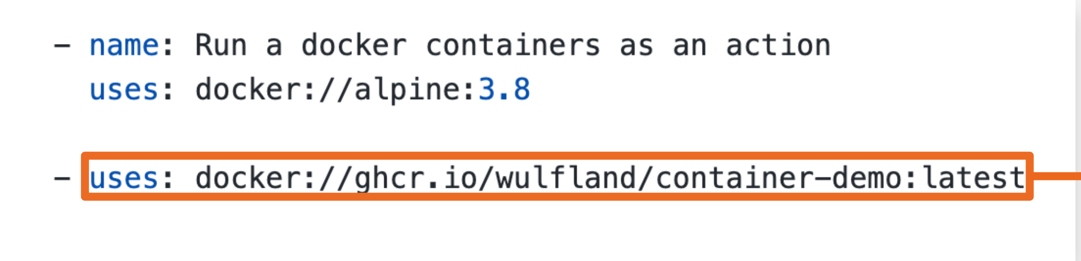
```
~/source/checkout   main ······························ 15:26:01
❯ git remote -v
origin  https://github.com/actions/checkout.git (fetch)
origin  https://github.com/actions/checkout.git (push)
~/source/checkout   main ······························ 15:26:08
❯ git log --oneline --graph --decorate --all -15
* add3486 (HEAD -> main, origin/main, origin/HEAD) Patch to fix the dependbot alert. (#744)
* 5126516 Bump minimist from 1.2.5 to 1.2.6 (#741)
* d50f8ea Add v3.0 release information to changelog (#740)
* 2d1c119 update test workflows to checkout v3 (#709)
* a12a394 (tag: v3.0.0, tag: v3) update readme for v3 (#708)
* 8f9e05e Update to node 16 (#689)
* 230611d (origin/releases/v2) Change secret name for PAT to not start with GITHUB_ (#623)
* ec3a7ce (tag: v2.4.0, tag: v2) set insteadOf url for org-id (#621)
* fd47087 codeql should analyze lib not dist (#620)
* 3d677ac script to generate license info (#614)
* 826ba42 npm audit fix (#612)
* eb8a193 update dev dependencies and react to new linting rules (#611)
* c49af7c Create codeql-analysis.yml (#602)
* 1e204e9 (tag: v2.3.5) update licensed check (#606)
* 0299a0d update dist (#605)
~/source/checkout   main ······························ 15:26:12
❯
```

```
- uses: actions/checkout@a12a3943b4bdde767164f792f33f40b04645d846
- uses: actions/checkout@v3.0.0
- uses: actions/checkout@v3
- uses: actions/checkout@main
```

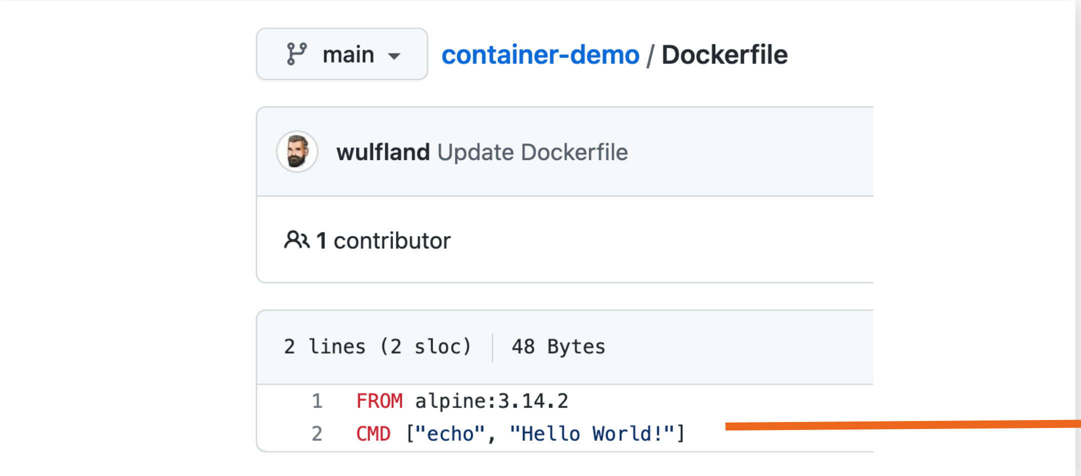
Xpirit

Actions

User docker images as actions



```yaml
- name: Run a docker containers as an action
  uses: docker://alpine:3.8

- uses: docker://ghcr.io/wulfland/container-demo:latest
```



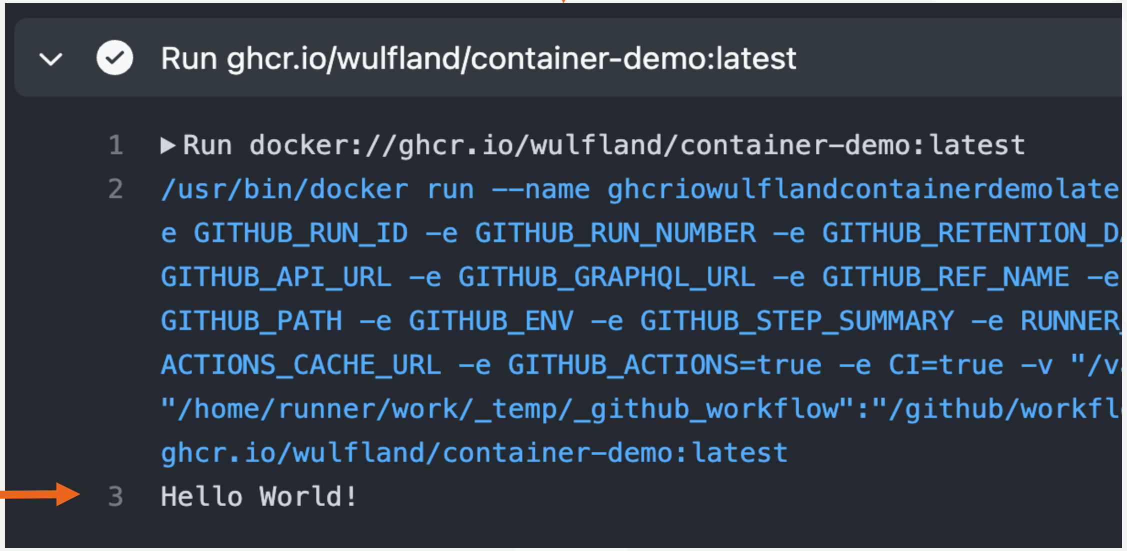
main · container-demo / Dockerfile

wulfland Update Dockerfile

1 contributor

2 lines (2 sloc) | 48 Bytes

```dockerfile
FROM alpine:3.14.2
CMD ["echo", "Hello World!"]
```



Run ghcr.io/wulfland/container-demo:latest

```
▶Run docker://ghcr.io/wulfland/container-demo:latest
/usr/bin/docker run --name ghcriowulflandcontainerdemolate
e GITHUB_RUN_ID -e GITHUB_RUN_NUMBER -e GITHUB_RETENTION_D
GITHUB_API_URL -e GITHUB_GRAPHQL_URL -e GITHUB_REF_NAME -e
GITHUB_PATH -e GITHUB_ENV -e GITHUB_STEP_SUMMARY -e RUNNER
ACTIONS_CACHE_URL -e GITHUB_ACTIONS=true -e CI=true -v "/v
"/home/runner/work/_temp/_github_workflow":"/github/workfl
ghcr.io/wulfland/container-demo:latest
Hello World!
```

Xpirit

Contexts and expressions syntax

Contexts and expressions syntax

${{ <expression> }}

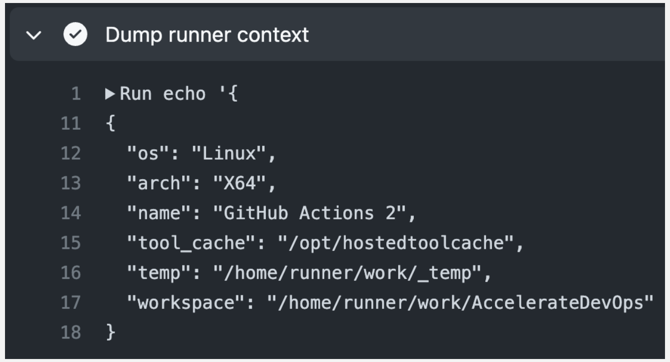
context syntax:

- context['key'] (if key starts with number or contains special characters)
- context.key

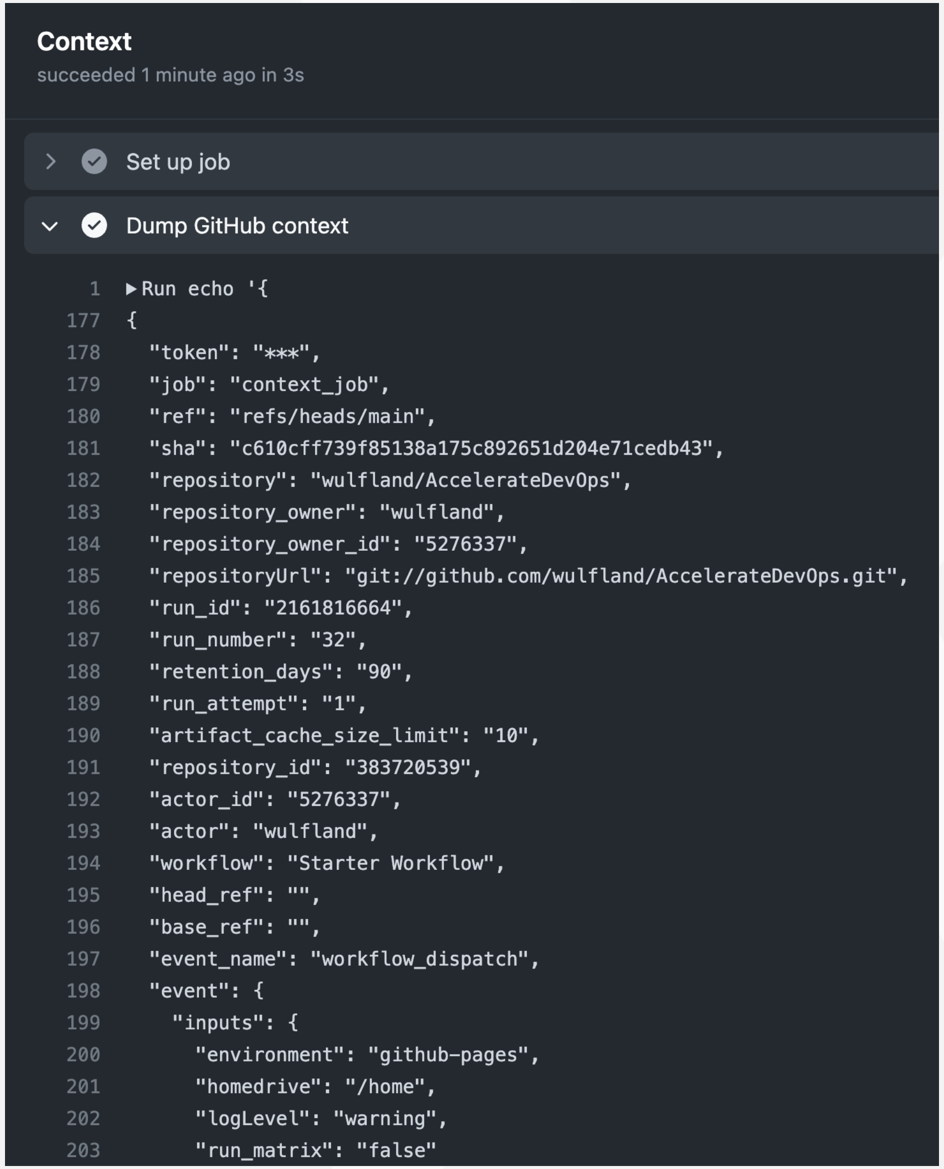
Context:

- matrix
- github
- env
- runner

```yaml
steps:
  - name: Dump runner context
    run: echo '${{ toJSON(runner) }}'
  - name: Dump GitHub context
    run: echo '${{ toJSON(github) }}'
```

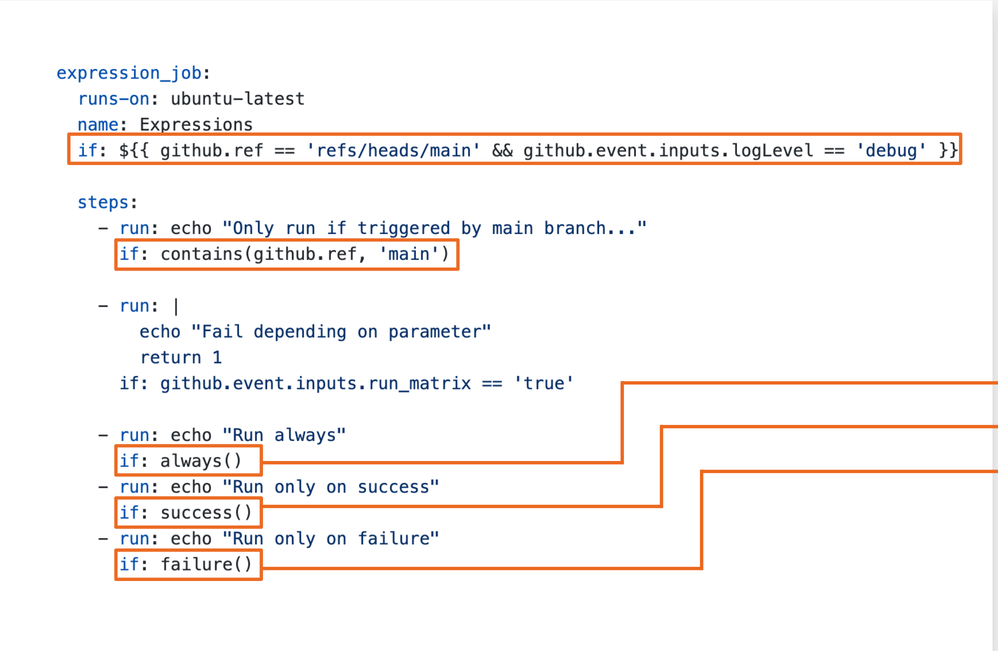


```
▶Run echo '{
{
  "os": "Linux",
  "arch": "X64",
  "name": "GitHub Actions 2",
  "tool_cache": "/opt/hostedtoolcache",
  "temp": "/home/runner/work/_temp",
  "workspace": "/home/runner/work/AccelerateDevOps"
}
```

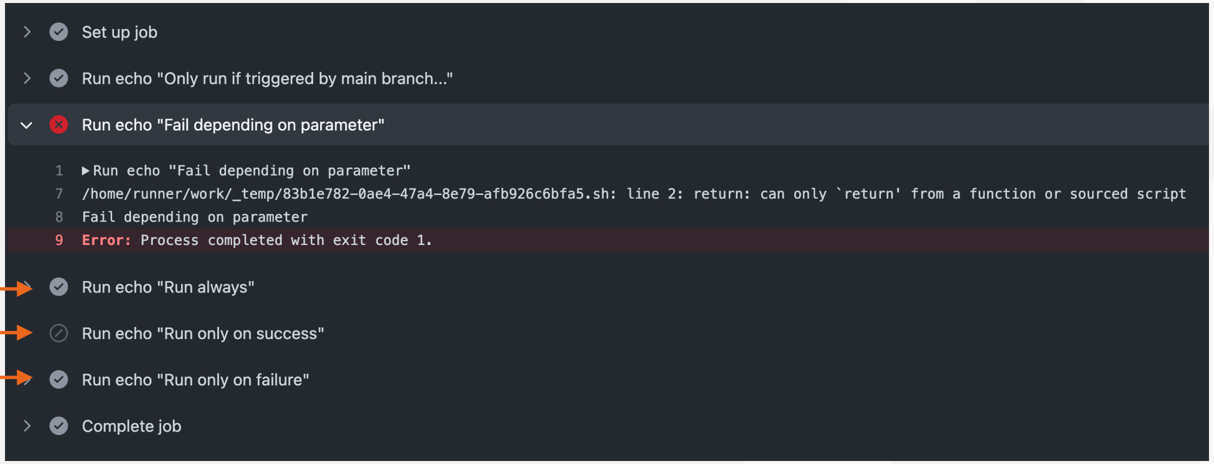


```
▶Run echo '{
{
  "token": "***",
  "job": "context_job",
  "ref": "refs/heads/main",
  "sha": "c610cff739f85138a175c892651d204e71cedb43",
  "repository": "wulfland/AccelerateDevOps",
  "repository_owner": "wulfland",
  "repository_owner_id": "5276337",
  "repositoryUrl": "git://github.com/wulfland/AccelerateDevOps.git",
  "run_id": "2161816664",
  "run_number": "32",
  "retention_days": "90",
  "run_attempt": "1",
  "artifact_cache_size_limit": "10",
  "repository_id": "383720539",
  "actor_id": "5276337",
  "actor": "wulfland",
  "workflow": "Starter Workflow",
  "head_ref": "",
  "base_ref": "",
  "event_name": "workflow_dispatch",
  "event": {
    "inputs": {
      "environment": "github-pages",
      "homedrive": "/home",
      "logLevel": "warning",
      "run_matrix": "false"
```

Contexts and expressions syntax



```
expression_job:
  runs-on: ubuntu-latest
  name: Expressions
  if: ${{ github.ref == 'refs/heads/main' && github.event.inputs.logLevel == 'debug' }}

  steps:
    - run: echo "Only run if triggered by main branch..."
      if: contains(github.ref, 'main')

    - run: |
        echo "Fail depending on parameter"
        return 1
      if: github.event.inputs.run_matrix == 'true'

    - run: echo "Run always"
      if: always()
    - run: echo "Run only on success"
      if: success()
    - run: echo "Run only on failure"
      if: failure()
```



```
Set up job
Run echo "Only run if triggered by main branch..."
Run echo "Fail depending on parameter"
  ▶Run echo "Fail depending on parameter"
  /home/runner/work/_temp/83b1e782-0ae4-47a4-8e79-afb926c6bfa5.sh: line 2: return: can only `return' from a function or sourced script
  Fail depending on parameter
  Error: Process completed with exit code 1.
Run echo "Run always"
Run echo "Run only on success"
Run echo "Run only on failure"
Complete job
```

Contexts and expressions syntax

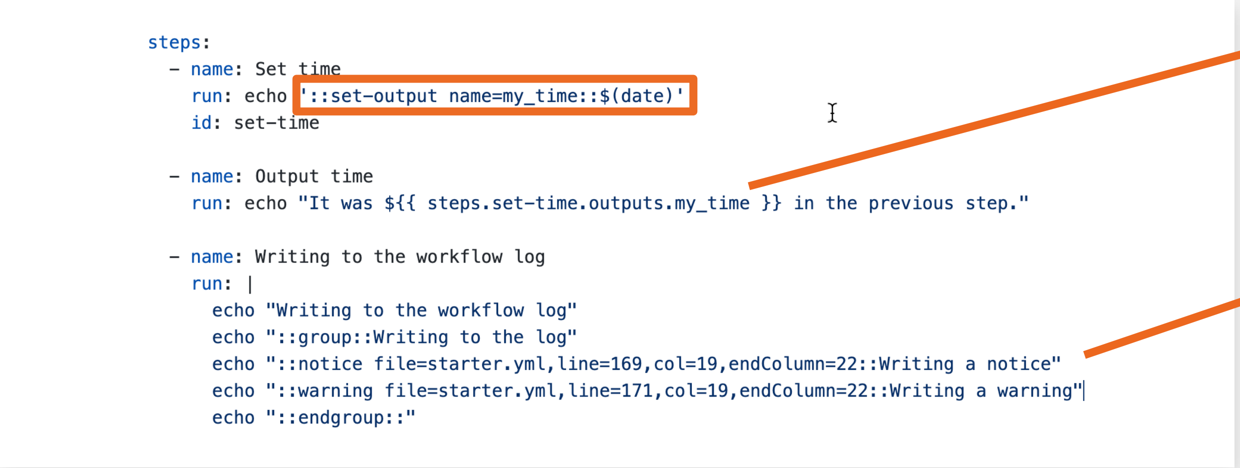
Function	Description
success()	Returns true if none of the previous steps have failed or been cancelled.
always()	Returns true even if a previous step was cancelled and causes the step to always get executed anyway.
cancelled()	Returns only true if the workflow was canceled.
failure()	Returns true if a previous step of the job had failed.

Operator	Description
( )	Logical group
!	Not
< , <=	Less than, less than or equal
> , >=	Greater than, greater than or equal
==	Equal
!=	Not equal
&&	And
||	Or

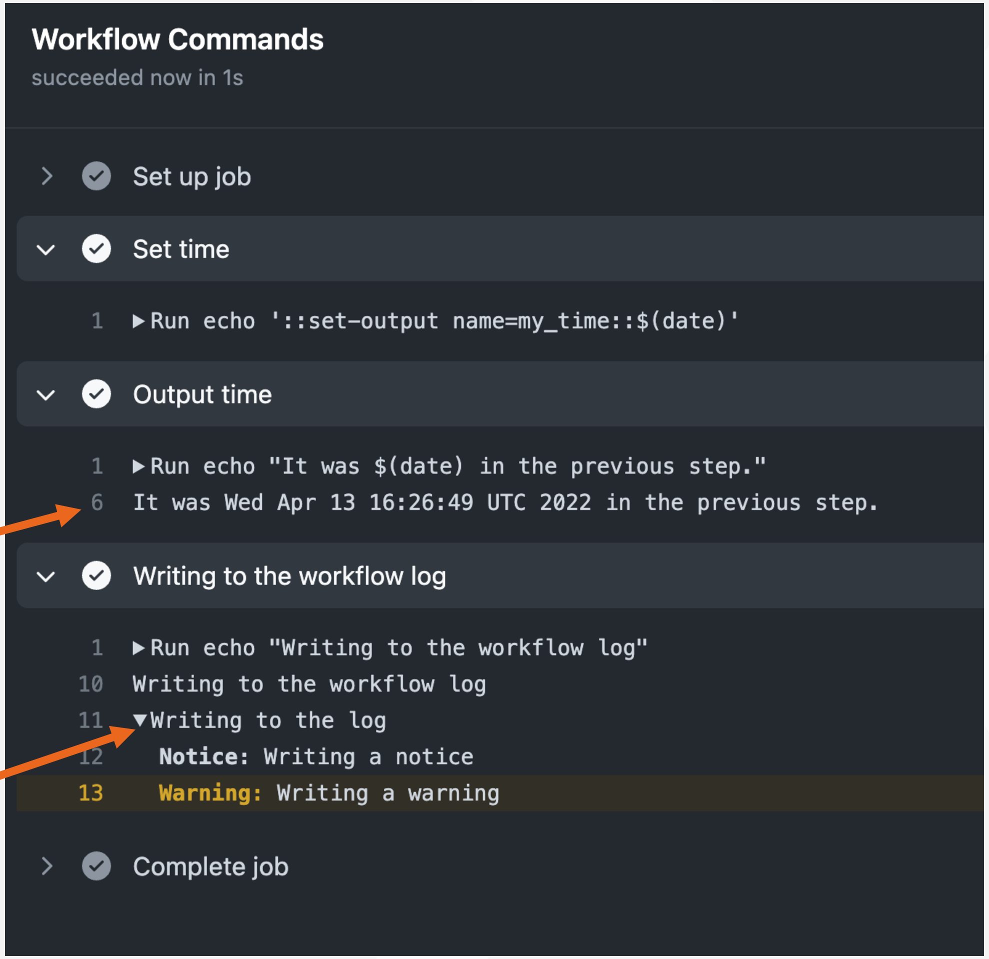
Function	Description
Contains (search, item)	Returns true if search contains item.
startsWith (search, item)	Returns true if search start with item.
endsWith (search, item)	Returns true if search ends with item
format(' {0} ', item)	Replaces placeholders in a string.
join(array, seperator)	All values in array are concatenated into a string.
toJSON (value)	Returns a pretty-print JSON representation of value.
fromJSON (value)	Returns a JSON object or JSON data type for value.

Workflow commands

- Interact with the workflow from within your steps
- Write command to output (normally using echo)
- Examples:
  - Set-output
  - Error



```
steps:
  - name: Set time
    run: echo '::set-output name=my_time::$(date)'
    id: set-time

  - name: Output time
    run: echo "It was ${{ steps.set-time.outputs.my_time }} in the previous step."

  - name: Writing to the workflow log
    run: |
      echo "Writing to the workflow log"
      echo "::group::Writing to the log"
      echo "::notice file=starter.yml,line=169,col=19,endColumn=22::Writing a notice"
      echo "::warning file=starter.yml,line=171,col=19,endColumn=22::Writing a warning"
      echo "::endgroup::"
```



Workflow Commands
succeeded now in 1s

```
▶Run echo '::set-output name=my_time::$(date)'
```

```
▶Run echo "It was $(date) in the previous step."
It was Wed Apr 13 16:26:49 UTC 2022 in the previous step.
```

```
▶Run echo "Writing to the workflow log"
Writing to the workflow log
▼Writing to the log
  Notice: Writing a notice
  Warning: Writing a warning
```

Workflow commands

Toolkit function	Equivalent workflow command
core.addPath	Accessible using environment file GITHUB_PATH
core.debug	debug
core.notice	notice
core.error	error
core.endGroup	endgroup
core.exportVariable	Accessible using environment file GITHUB_ENV
core.getInput	Accessible using environment variable INPUT_{NAME}
core.getState	Accessible using environment variable STATE_{NAME}

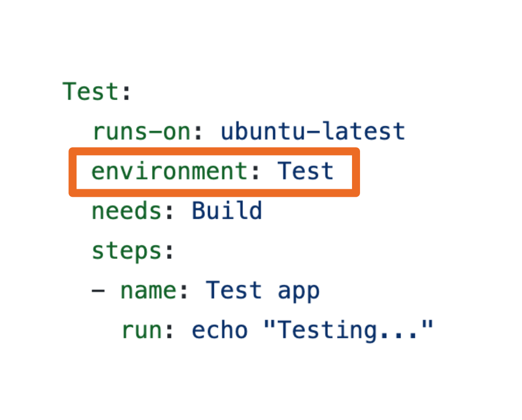
Toolkit function	Equivalent workflow command
core.isDebug	Accessible using environment variable RUNNER_DEBUG
core.saveState	save-state
core.setCommandEcho	echo
core.setFailed	Used as a shortcut for ::error and exit 1
core.setOutput	set-output
core.setSecret	add-mask
core.startGroup	group
core.warning	warning

Environments

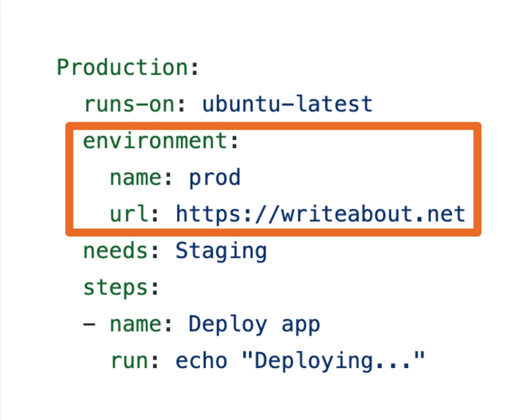
Environments

Environments:

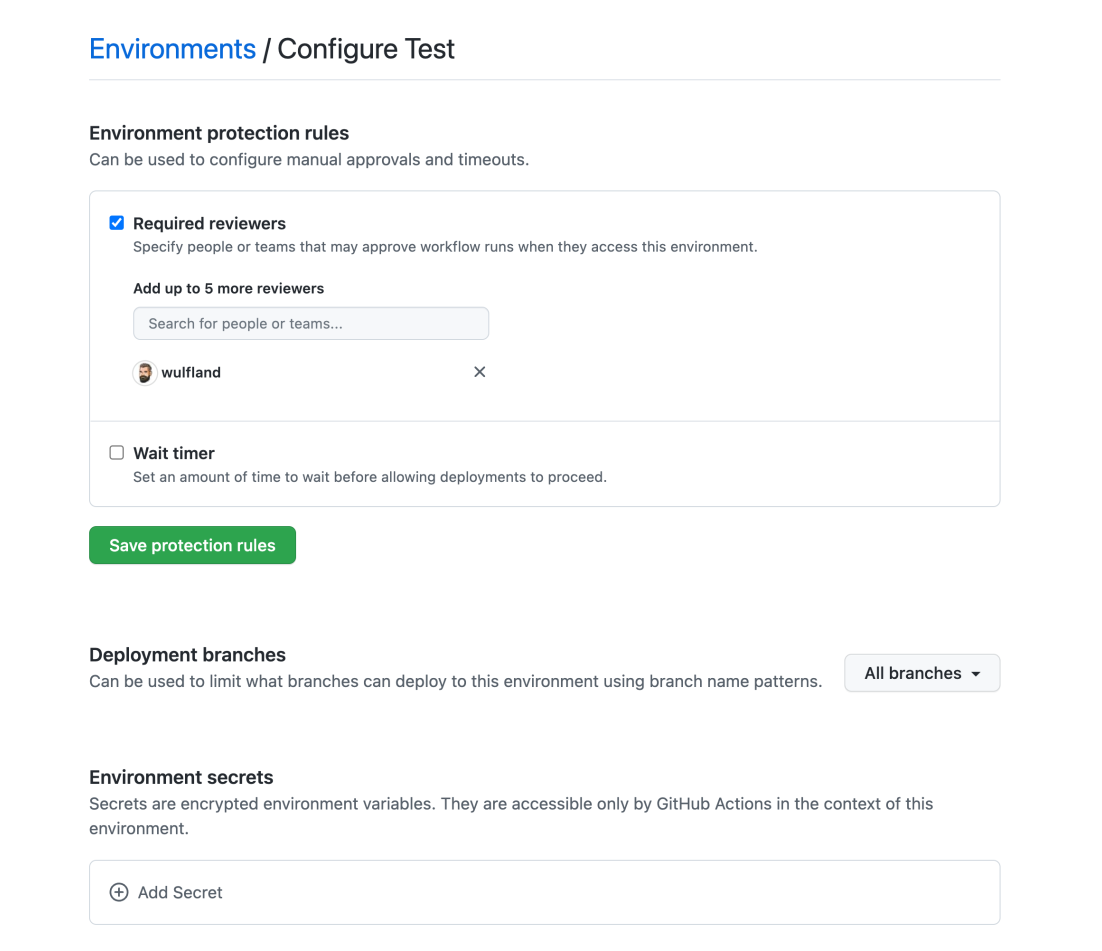
- Reviewers / Approvers
- Wait timer (until 30 days)
- Branches (→ branch protection!)
- Deployment branches
- Secrets



```
Test:
  runs-on: ubuntu-latest
  environment: Test
  needs: Build
  steps:
  - name: Test app
    run: echo "Testing..."
```



```
Production:
  runs-on: ubuntu-latest
  environment:
    name: prod
    url: https://writeabout.net
  needs: Staging
  steps:
  - name: Deploy app
    run: echo "Deploying..."
```



Environments / Configure Test

Environment protection rules

Can be used to configure manual approvals and timeouts.

☑ **Required reviewers**

Specify people or teams that may approve workflow runs when they access this environment.

Add up to 5 more reviewers

Search for people or teams...

wulfland ✕

☐ **Wait timer**

Set an amount of time to wait before allowing deployments to proceed.

Deployment branches

Can be used to limit what branches can deploy to this environment using branch name patterns.

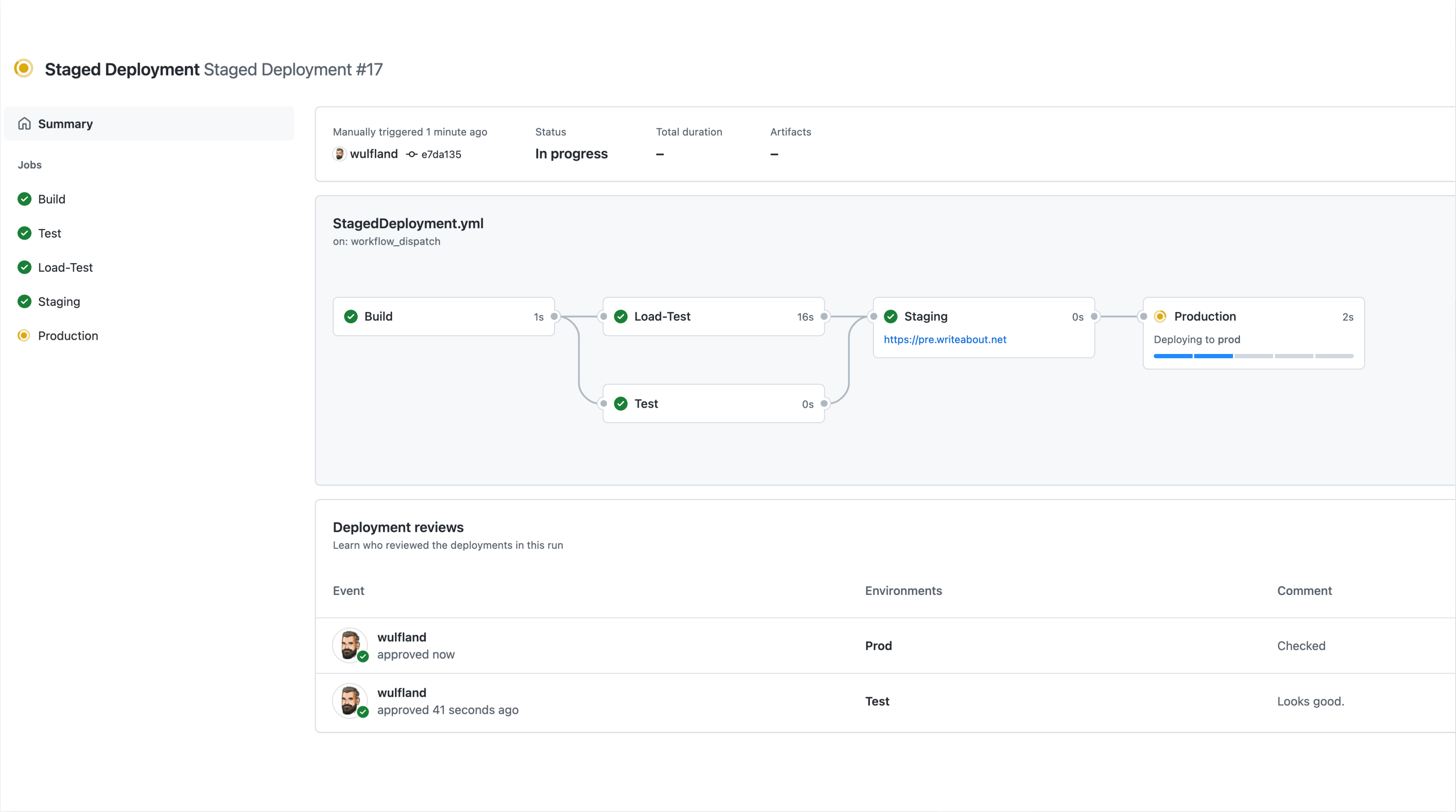
All branches ▾

Environment secrets

Secrets are encrypted environment variables. They are accessible only by GitHub Actions in the context of this environment.

Environments

- Approvals
- Secrets after approval
- Set URL from output of other job/step
- Progress



Staged Deployment Staged Deployment #17

Summary

Jobs

- ✓ Build
- ✓ Test
- ✓ Load-Test
- ✓ Staging
- ● Production

Manually triggered 1 minute ago	Status	Total duration	Artifacts
wulfland e7da135	In progress	–	–

StagedDeployment.yml
on: workflow_dispatch

Deployment reviews
Learn who reviewed the deployments in this run

Event	Environments	Comment
wulfland approved now	**Prod**	Checked
wulfland approved 41 seconds ago	**Test**	Looks good.

Secrets

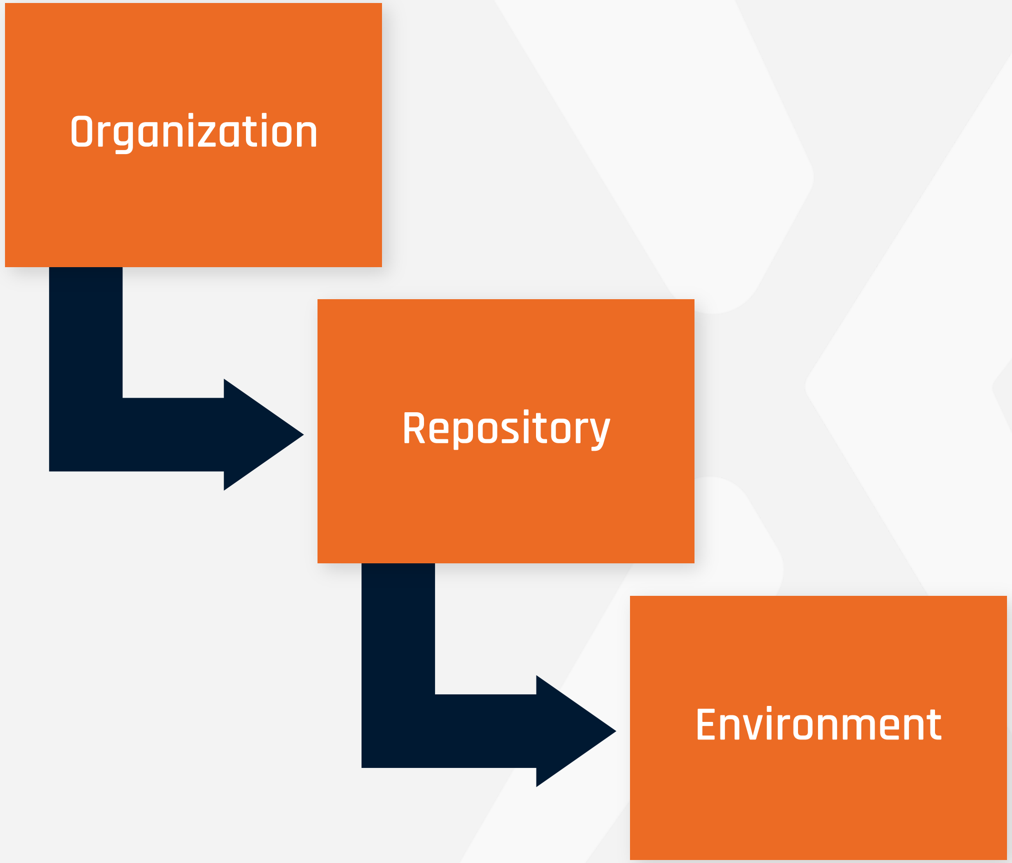
Secrets

- Defined on org, repo, or environment level

- Secret context
  - ${{ secrets.MY_SECRET }}
  - Set as input (**with:**) or environment (**env:**) for actions

- Set in UI or CLI
  - $ gh secret set MY_SECRET --body P4ssw0rd
  - $ gh secret set MY_SECRET --env Production
  - $ gh secret set MY_SECRET --org my-org

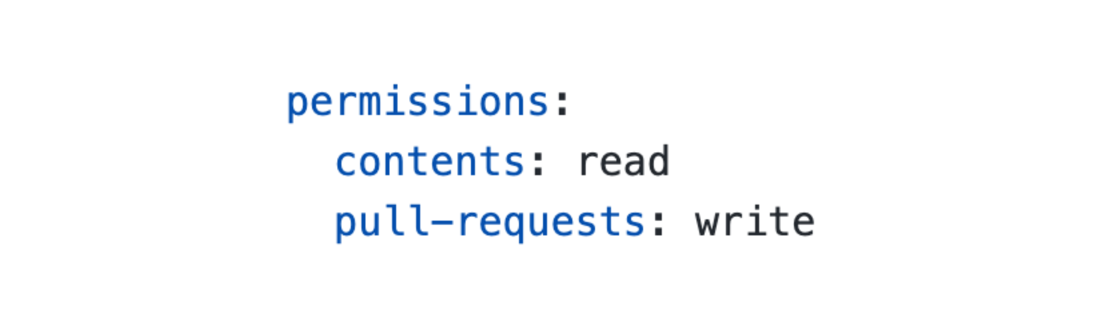
- Masked in log



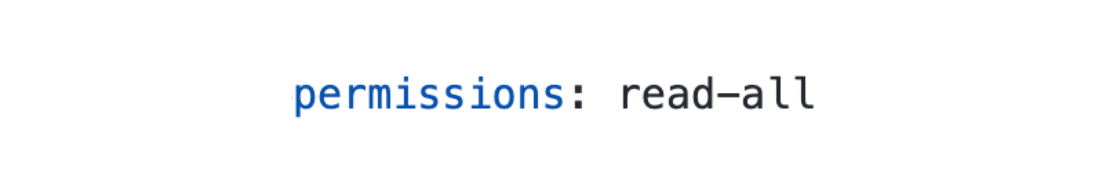
Xpirit

The GITHUB_TOKEN

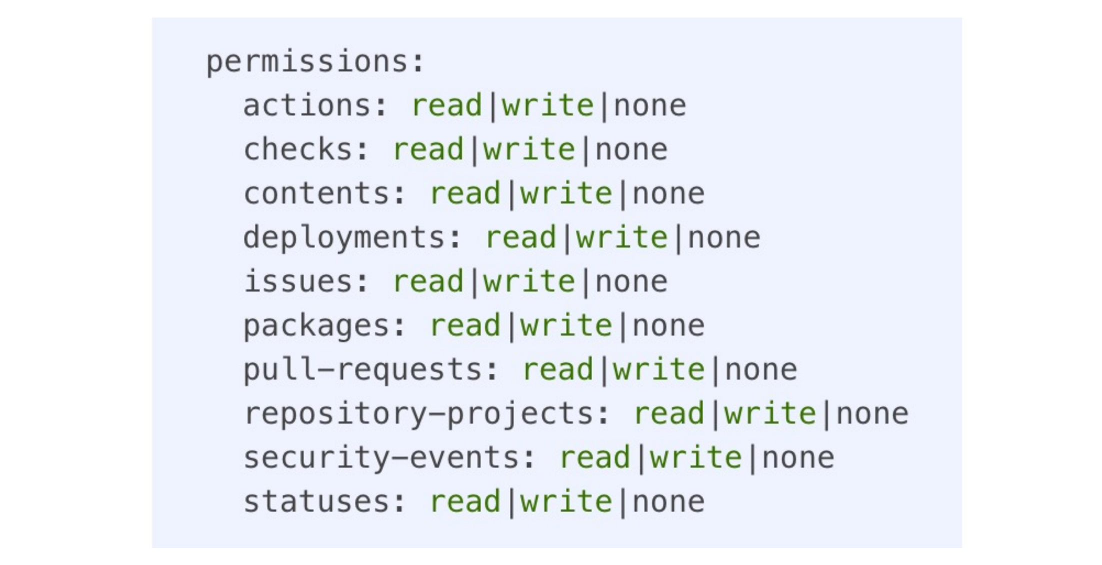
- ${{ secrets.GITHUB_TOKEN }} or ${{ github.token }}
- Authenticate to GitHub to perform automation
- Default permission read/write for all scopes (current default) or read repo



```yaml
permissions:
  contents: read
  pull-requests: write
```



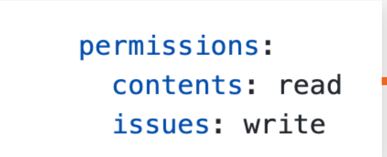
```yaml
permissions: read-all
```



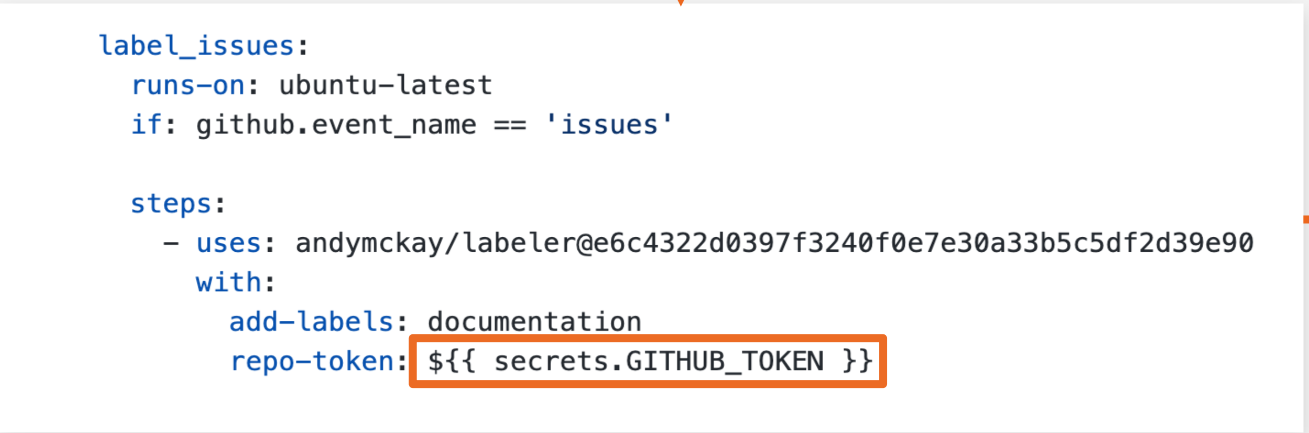
```yaml
permissions:
  actions: read|write|none
  checks: read|write|none
  contents: read|write|none
  deployments: read|write|none
  issues: read|write|none
  packages: read|write|none
  pull-requests: read|write|none
  repository-projects: read|write|none
  security-events: read|write|none
  statuses: read|write|none
```

The GITHUB_TOKEN

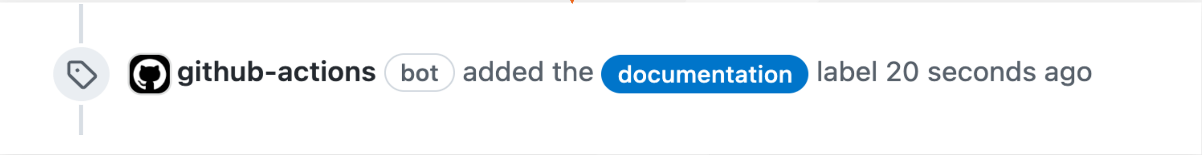
Perform actions as github-actions:



```yaml
permissions:
  contents: read
  issues: write
```



```yaml
label_issues:
  runs-on: ubuntu-latest
  if: github.event_name == 'issues'

  steps:
    - uses: andymckay/labeler@e6c4322d0397f3240f0e7e30a33b5c5df2d39e90
      with:
        add-labels: documentation
        repo-token: ${{ secrets.GITHUB_TOKEN }}
```



github-actions bot added the documentation label 20 seconds ago

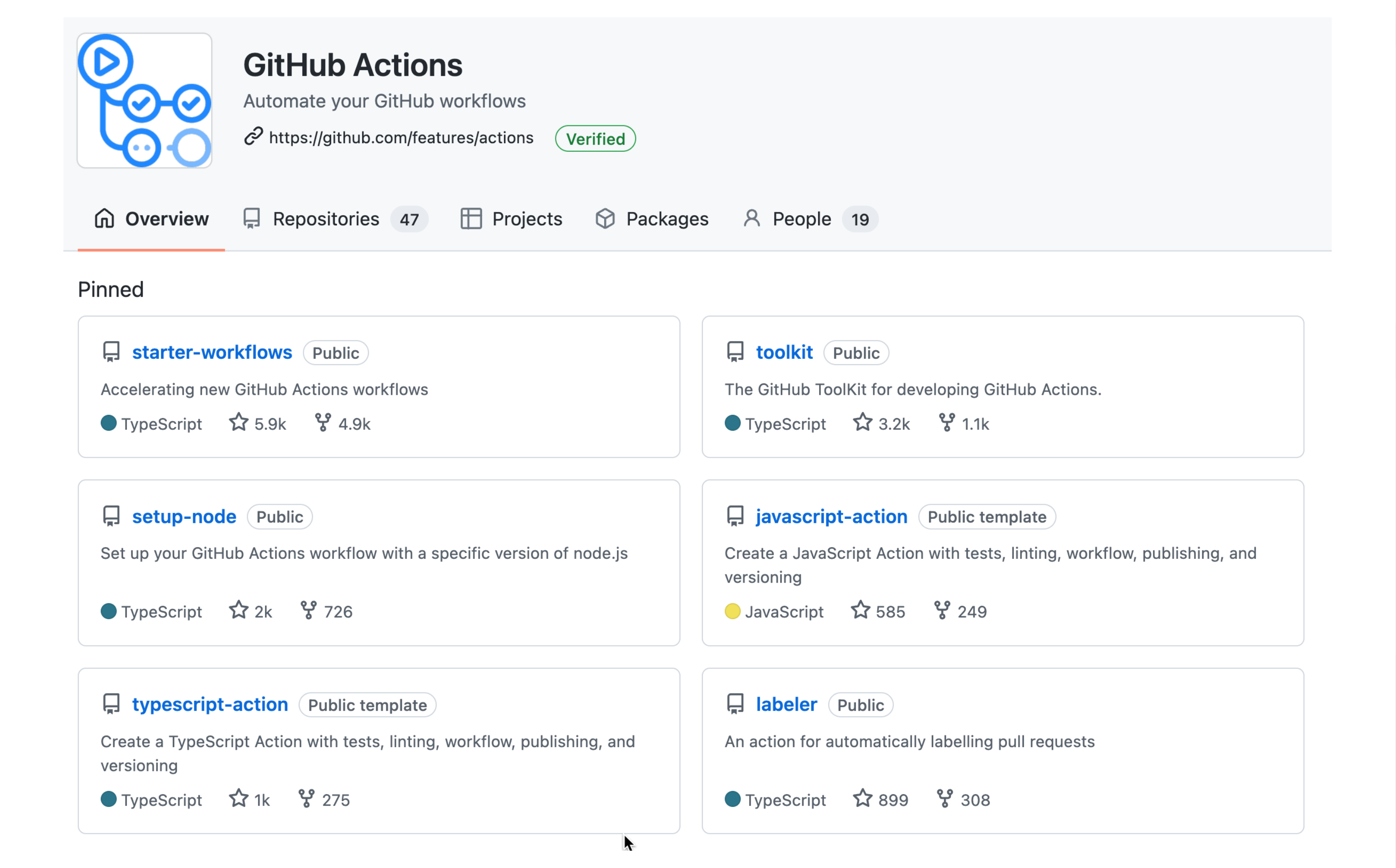
Hands-on

Actions

GitHub Actions

- Actions are reusable

- 3 kind of Actions
  - Container
  - JavaScript / Typescript
  - Composite Actions



GitHub Actions

Automate your GitHub workflows

https://github.com/features/actions Verified

Pinned

starter-workflows Public
Accelerating new GitHub Actions workflows
● TypeScript ☆ 5.9k 4.9k

toolkit Public
The GitHub ToolKit for developing GitHub Actions.
● TypeScript ☆ 3.2k 1.1k

setup-node Public
Set up your GitHub Actions workflow with a specific version of node.js
● TypeScript ☆ 2k 726

javascript-action Public template
Create a JavaScript Action with tests, linting, workflow, publishing, and versioning
● JavaScript ☆ 585 249

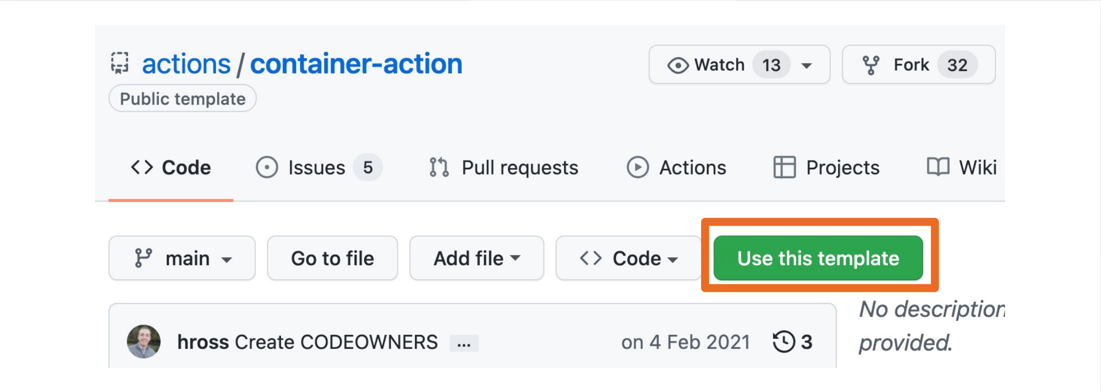
typescript-action Public template
Create a TypeScript Action with tests, linting, workflow, publishing, and versioning
● TypeScript ☆ 1k 275

labeler Public
An action for automatically labelling pull requests
● TypeScript ☆ 899 308

Xpirit

Container Actions

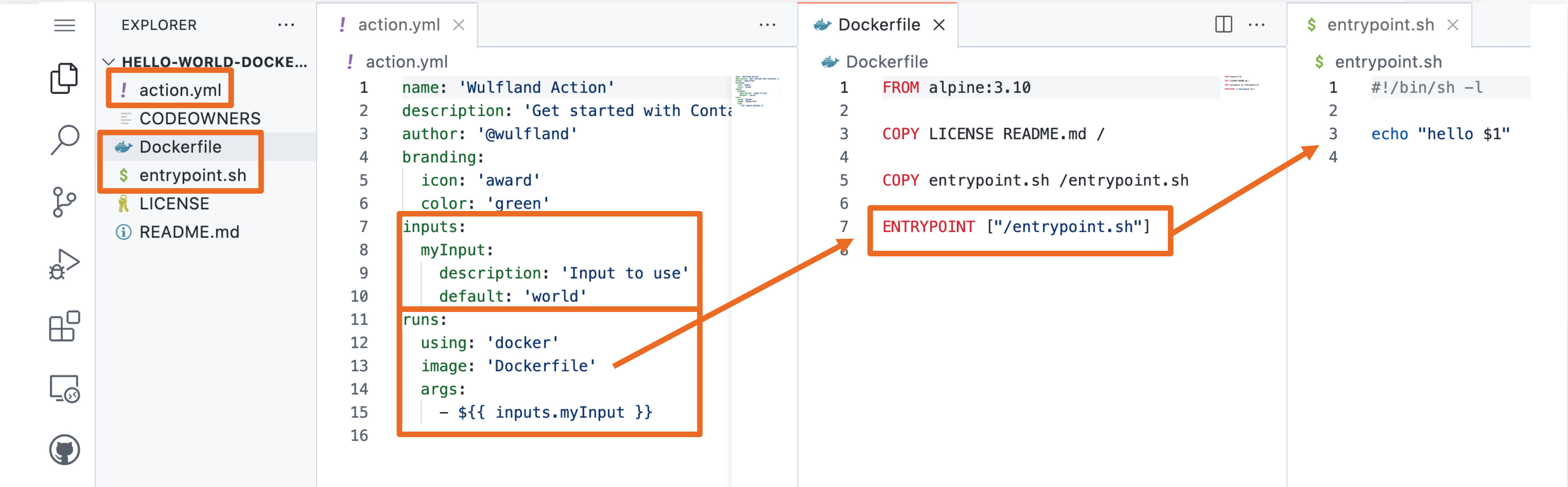
- Dockerfile or existing image
- inputs


actions / container-action

Public template

hross Create CODEOWNERS ... on 4 Feb 2021 3

No description provided.



action.yml

```yaml
name: 'Wulfland Action'
description: 'Get started with Conta
author: '@wulfland'
branding:
  icon: 'award'
  color: 'green'
inputs:
  myInput:
    description: 'Input to use'
    default: 'world'
runs:
  using: 'docker'
  image: 'Dockerfile'
  args:
    - ${{ inputs.myInput }}

```

Dockerfile

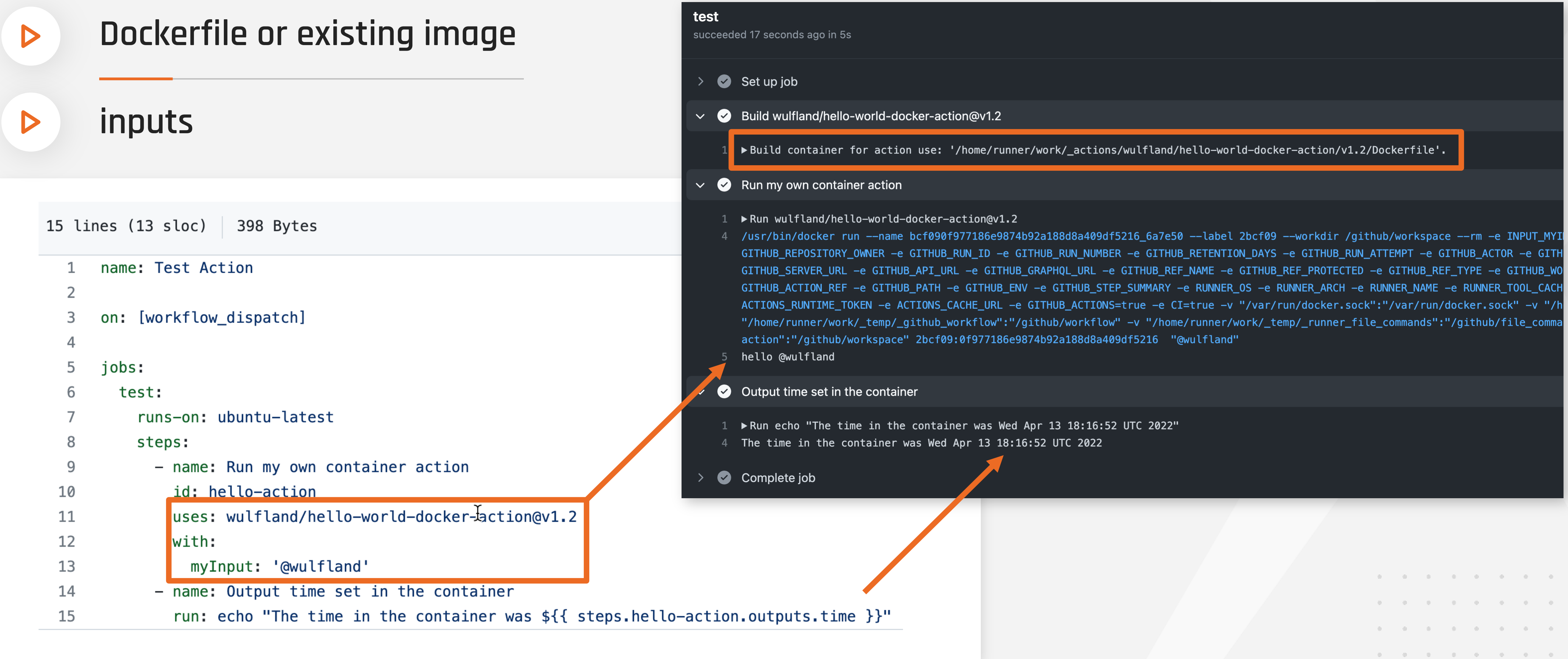
```dockerfile
FROM alpine:3.10

COPY LICENSE README.md /

COPY entrypoint.sh /entrypoint.sh

ENTRYPOINT ["/entrypoint.sh"]
```

entrypoint.sh

```sh
#!/bin/sh -l

echo "hello $1"

```

Container Actions

- Dockerfile or existing image
- inputs



15 lines (13 sloc) | 398 Bytes

```yaml
name: Test Action

on: [workflow_dispatch]

jobs:
  test:
    runs-on: ubuntu-latest
    steps:
      - name: Run my own container action
        id: hello-action
        uses: wulfland/hello-world-docker-action@v1.2
        with:
          myInput: '@wulfland'
      - name: Output time set in the container
        run: echo "The time in the container was ${{ steps.hello-action.outputs.time }}"
```

test

succeeded 17 seconds ago in 5s

Build wulfland/hello-world-docker-action@v1.2

```
▶Build container for action use: '/home/runner/work/_actions/wulfland/hello-world-docker-action/v1.2/Dockerfile'.
```

Run my own container action

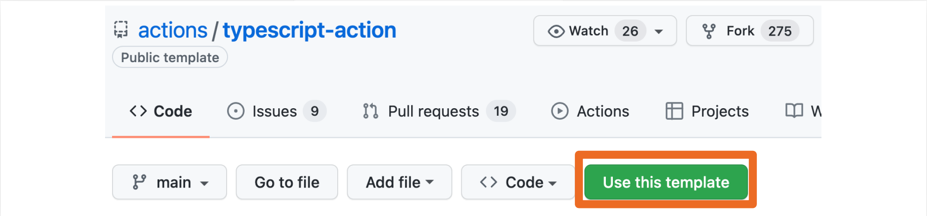
```
▶Run wulfland/hello-world-docker-action@v1.2
/usr/bin/docker run --name bcf090f977186e9874b92a188d8a409df5216_6a7e50 --label 2bcf09 --workdir /github/workspace --rm -e INPUT_MYI
GITHUB_REPOSITORY_OWNER -e GITHUB_RUN_ID -e GITHUB_RUN_NUMBER -e GITHUB_RETENTION_DAYS -e GITHUB_RUN_ATTEMPT -e GITHUB_ACTOR -e GITH
GITHUB_SERVER_URL -e GITHUB_API_URL -e GITHUB_GRAPHQL_URL -e GITHUB_REF_NAME -e GITHUB_REF_PROTECTED -e GITHUB_REF_TYPE -e GITHUB_WO
GITHUB_ACTION_REF -e GITHUB_PATH -e GITHUB_ENV -e GITHUB_STEP_SUMMARY -e RUNNER_OS -e RUNNER_ARCH -e RUNNER_NAME -e RUNNER_TOOL_CACH
ACTIONS_RUNTIME_TOKEN -e ACTIONS_CACHE_URL -e GITHUB_ACTIONS=true -e CI=true -v "/var/run/docker.sock":"/var/run/docker.sock" -v "/h
"/home/runner/work/_temp/_github_workflow":"/github/workflow" -v "/home/runner/work/_temp/_runner_file_commands":"/github/file_comma
action":"/github/workspace" 2bcf09:0f977186e9874b92a188d8a409df5216  "@wulfland"
hello @wulfland
```

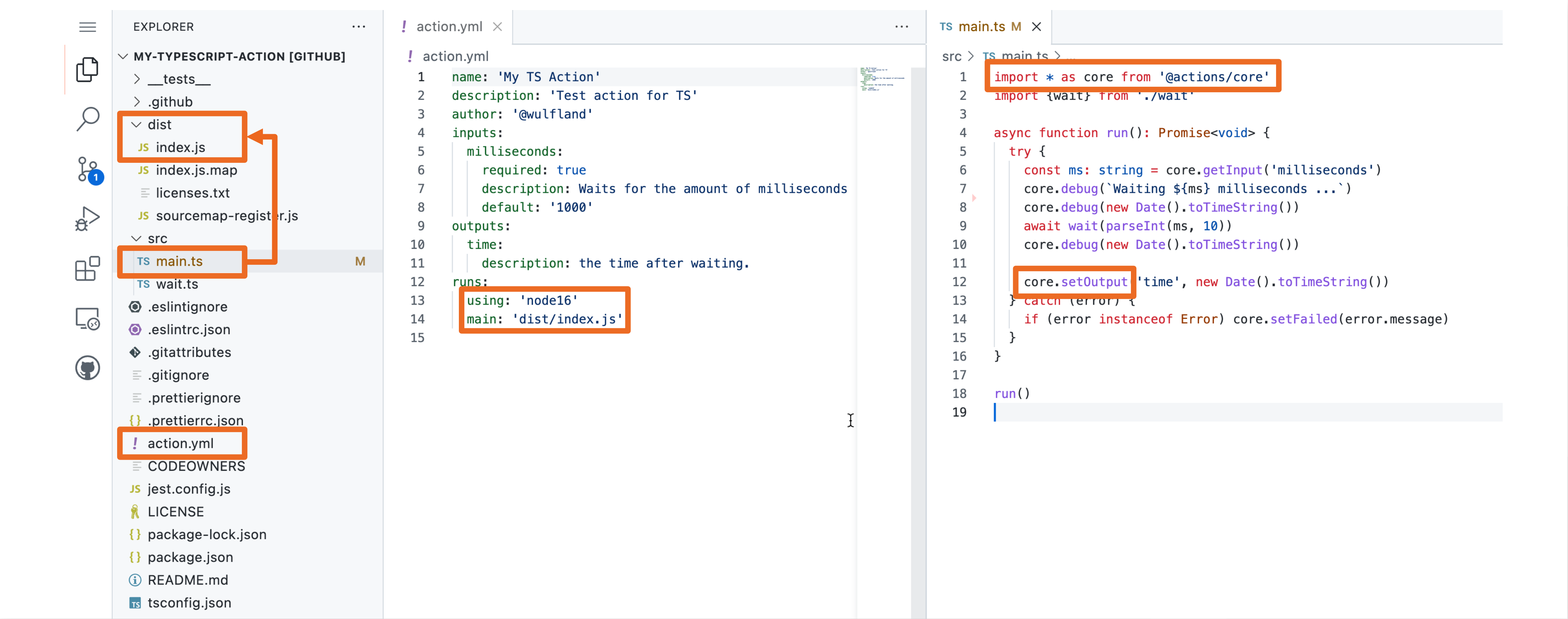
Output time set in the container

```
▶Run echo "The time in the container was Wed Apr 13 18:16:52 UTC 2022"
The time in the container was Wed Apr 13 18:16:52 UTC 2022
```

Xpirit

JavaScript Actions



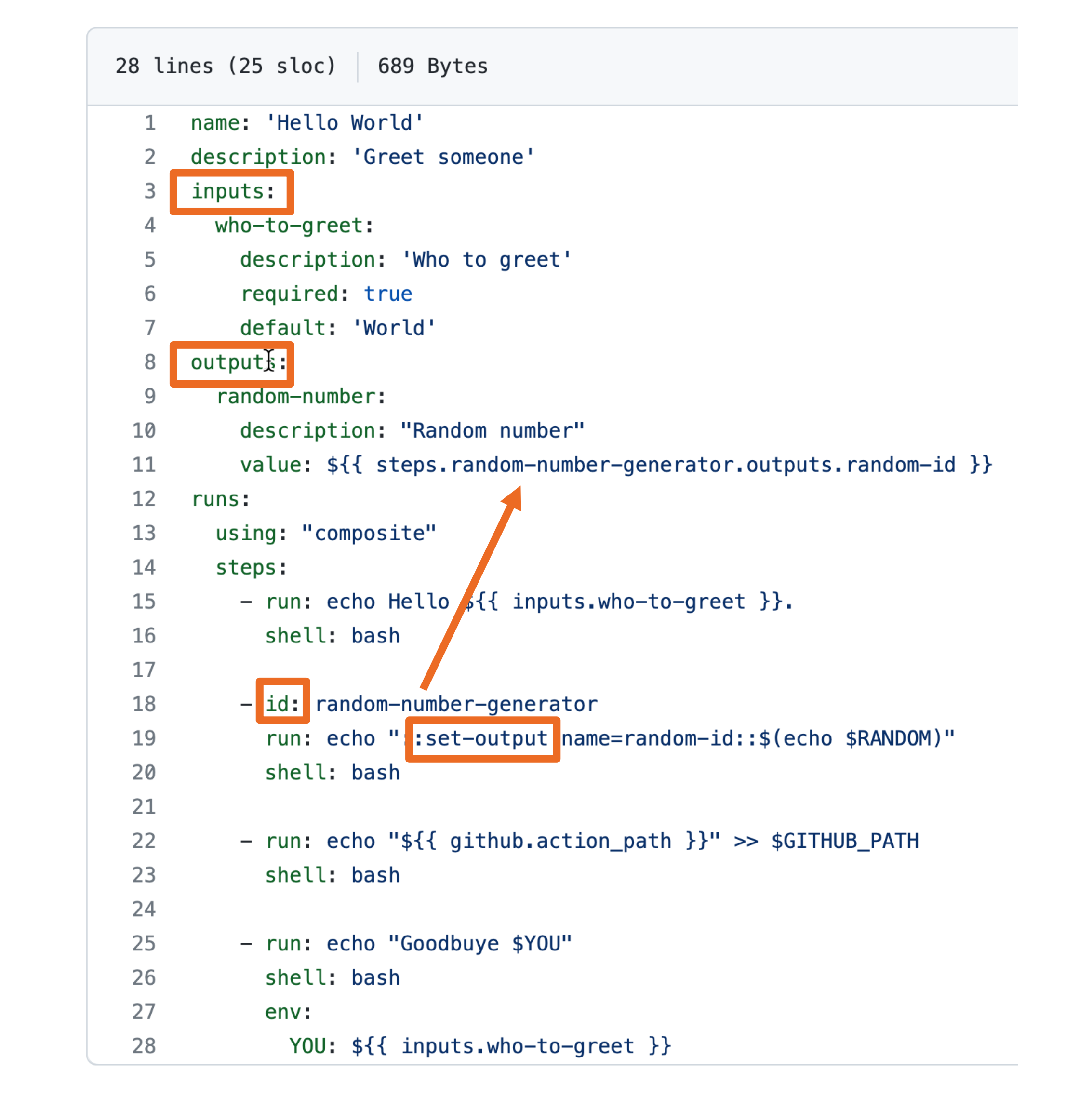


```yaml
name: 'My TS Action'
description: 'Test action for TS'
author: '@wulfland'
inputs:
  milliseconds:
    required: true
    description: Waits for the amount of milliseconds
    default: '1000'
outputs:
  time:
    description: the time after waiting.
runs:
  using: 'node16'
  main: 'dist/index.js'
```

```typescript
import * as core from '@actions/core'
import {wait} from './wait'

async function run(): Promise<void> {
  try {
    const ms: string = core.getInput('milliseconds')
    core.debug(`Waiting ${ms} milliseconds ...`)
    core.debug(new Date().toTimeString())
    await wait(parseInt(ms, 10))
    core.debug(new Date().toTimeString())

    core.setOutput('time', new Date().toTimeString())
  } catch (error) {
    if (error instanceof Error) core.setFailed(error.message)
  }
}

run()
```

Composite Actions

- Just a action.yml file
- Inputs
- Outputs
- Runs

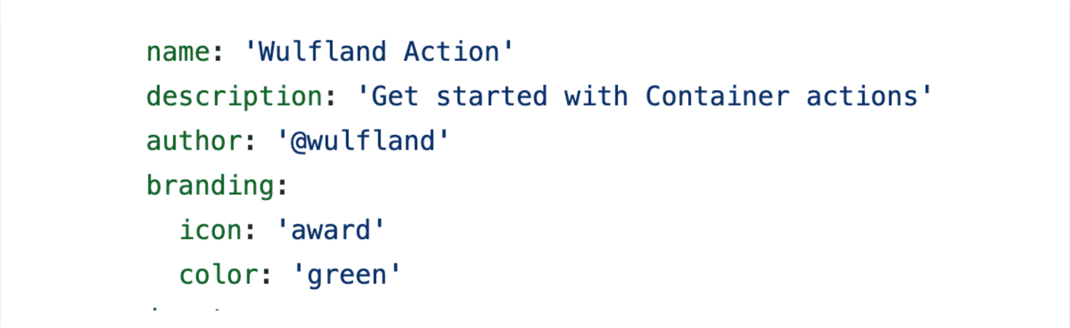


28 lines (25 sloc) | 689 Bytes

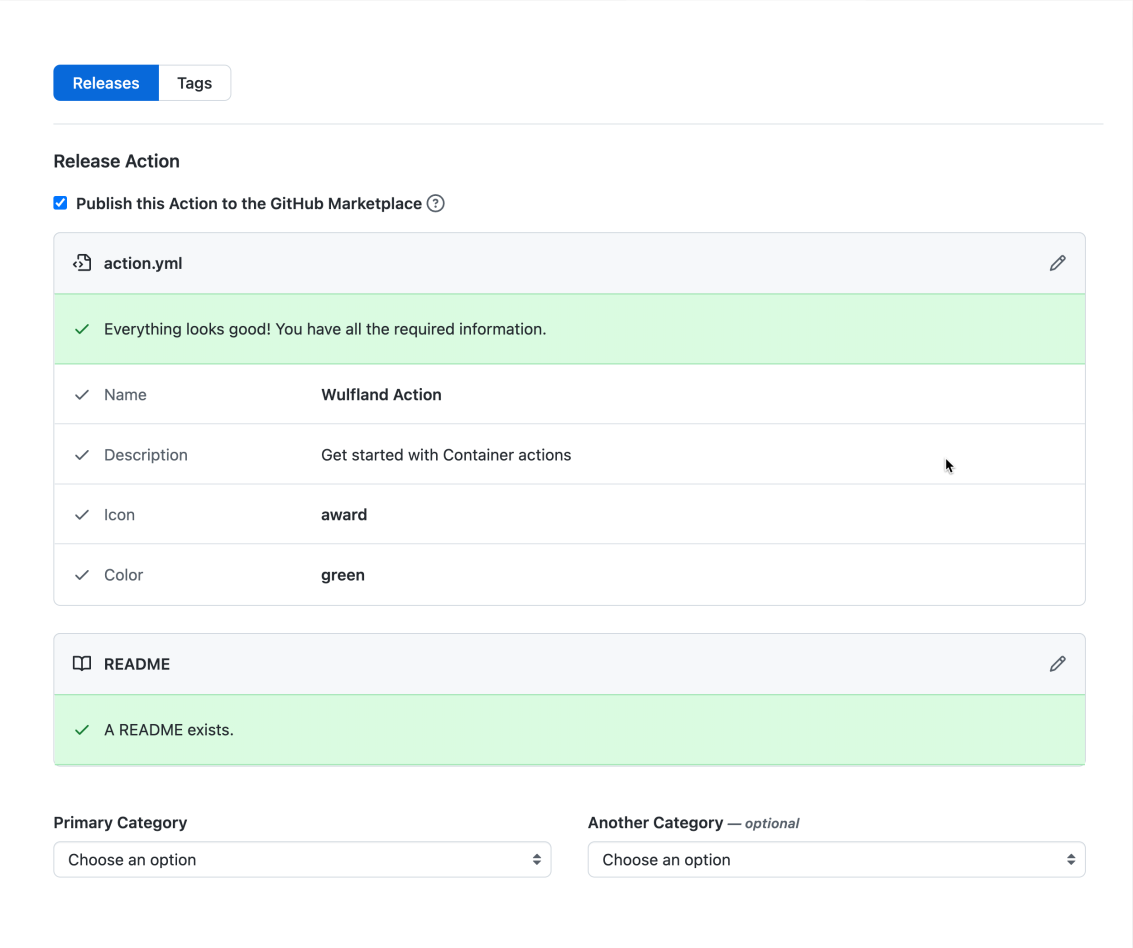
```yaml
name: 'Hello World'
description: 'Greet someone'
inputs:
  who-to-greet:
    description: 'Who to greet'
    required: true
    default: 'World'
outputs:
  random-number:
    description: "Random number"
    value: ${{ steps.random-number-generator.outputs.random-id }}
runs:
  using: "composite"
  steps:
    - run: echo Hello ${{ inputs.who-to-greet }}.
      shell: bash

    - id: random-number-generator
      run: echo "::set-output name=random-id::$(echo $RANDOM)"
      shell: bash

    - run: echo "${{ github.action_path }}" >> $GITHUB_PATH
      shell: bash

    - run: echo "Goodbuye $YOU"
      shell: bash
      env:
        YOU: ${{ inputs.who-to-greet }}
```

Share your action in the marketplace

- Really easy to share
- Draft a release
- Unique name
- Check for README, Icon, Color



```yaml
name: 'Wulfland Action'
description: 'Get started with Container actions'
author: '@wulfland'
branding:
  icon: 'award'
  color: 'green'
```



Release Action

☑ Publish this Action to the GitHub Marketplace

action.yml

✓ Everything looks good! You have all the required information.

✓	Name	Wulfland Action
✓	Description	Get started with Container actions
✓	Icon	award
✓	Color	green

README

✓ A README exists.

Primary Category: Choose an option

Another Category — *optional*: Choose an option

Hands-on

Running your workflows

GitHub hosted runners

- Linux
- Windows

Hardware:

- Standard_DS2_v2 virtual machines in Microsoft Azure
- 2-core CPU
- 7 GB of RAM
- 14 GB of SSD disk space

Passwordless sudo / UAC disabled

- MacOS

Hardware:

- 3-core CPU
- 14 GB of RAM
- 14 GB of SSD disk space

Passwordless sudo

Virtual Environments

Environment	YAML Label	Included Software
Ubuntu 20.04	**ubuntu-latest** or ubuntu-20.04	ubuntu-20.04
Ubuntu 18.04	ubuntu-18.04	ubuntu-18.04
macOS 11	**macos-latest** or macos-11	macOS-11
macOS 10.15	macos-10.15	macOS-10.15
Windows Server 2022	**windows-latest** or windows-2022	windows-2022
Windows Server 2019	windows-2019	windows-2019
Windows Server 2016	windows-2016	windows-2016

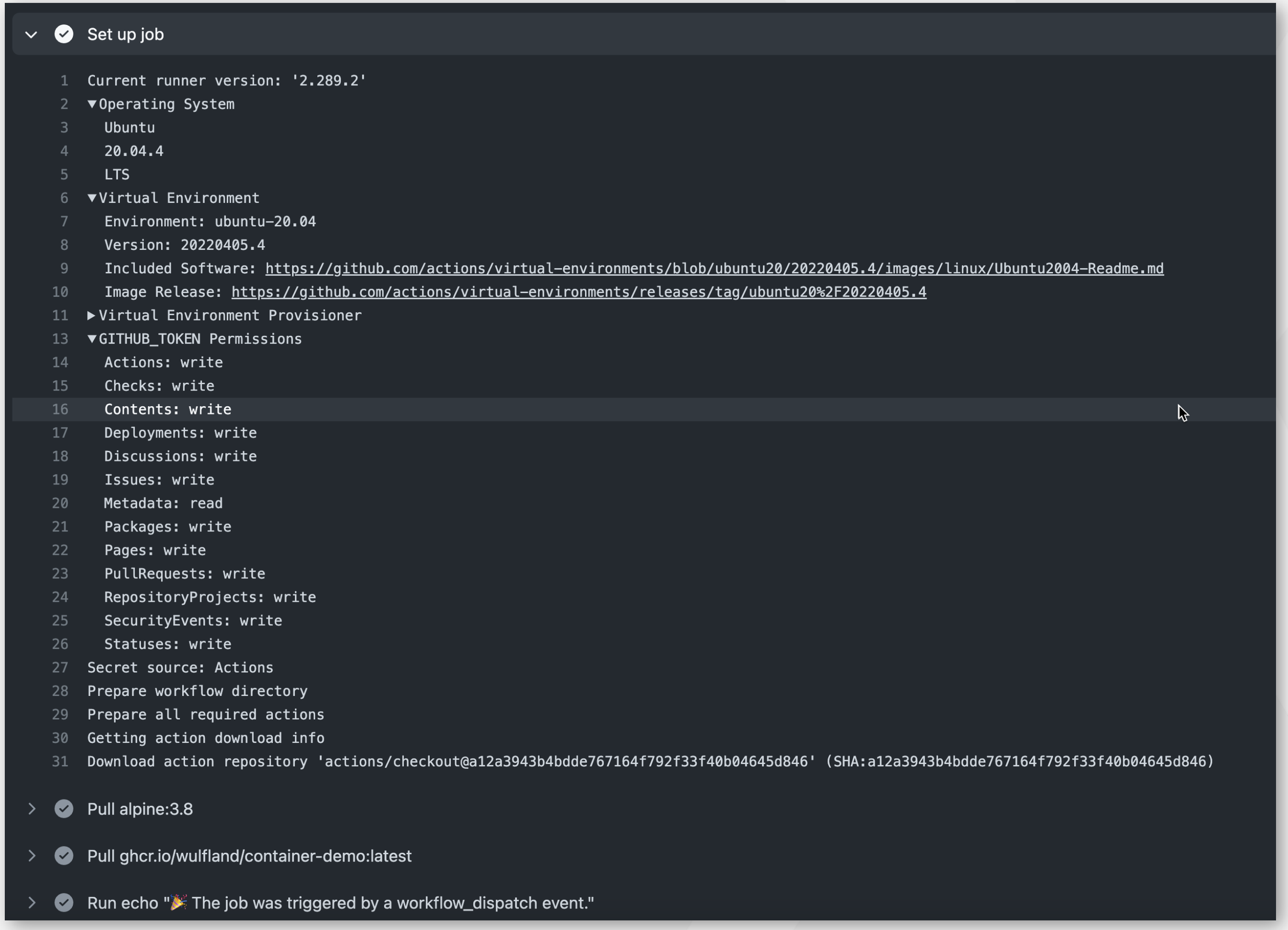
https://github.com/actions/virtual-environments

GitHub hosted runners - pricing

Build minutes:

- On Linux: $0.008
- On macOS: $0.08
- On Windows: $0.016

GitHub edition	Storage	Minutes	Max concurrent jobs
GitHub Free	500 MB	2,000	20 (5 for macOS)
GitHub Pro	1 GB	3,000	40 (5 for macOS)
GitHub Free for organizations	500 MB	2,000	20 (5 for macOS)
GitHub Team	2 GB	3,000	60 (5 for macOS)
GitHub Enterprise Cloud	50 GB	50,000	180 (50 for macOS)



```
Current runner version: '2.289.2'
▼Operating System
  Ubuntu
  20.04.4
  LTS
▼Virtual Environment
  Environment: ubuntu-20.04
  Version: 20220405.4
  Included Software: https://github.com/actions/virtual-environments/blob/ubuntu20/20220405.4/images/linux/Ubuntu2004-Readme.md
  Image Release: https://github.com/actions/virtual-environments/releases/tag/ubuntu20%2F20220405.4
▶Virtual Environment Provisioner
▼GITHUB_TOKEN Permissions
  Actions: write
  Checks: write
  Contents: write
  Deployments: write
  Discussions: write
  Issues: write
  Metadata: read
  Packages: write
  Pages: write
  PullRequests: write
  RepositoryProjects: write
  SecurityEvents: write
  Statuses: write
Secret source: Actions
Prepare workflow directory
Prepare all required actions
Getting action download info
Download action repository 'actions/checkout@a12a3943b4bdde767164f792f33f40b04645d846' (SHA:a12a3943b4bdde767164f792f33f40b04645d846)
```

Pull alpine:3.8
Pull ghcr.io/wulfland/container-demo:latest
Run echo "🎉 The job was triggered by a workflow_dispatch event."

Xpirit

Self-hosted runners

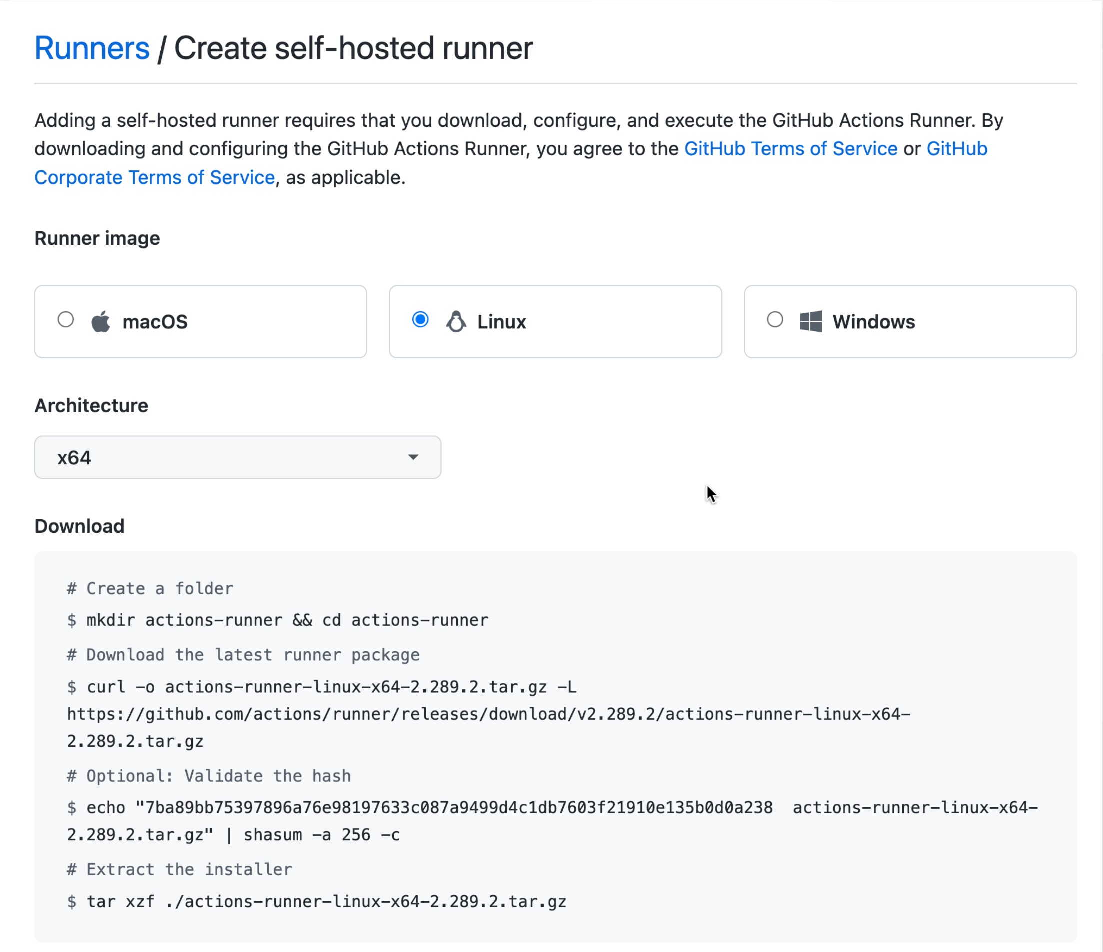
- **Free**
- **Any platform** (x64: Linux, macOS, Window. ARM64 and ARM32 on Linux)
- **HTTPS long polling** port 443 - 50 seconds
- Can be used to **deploy to local resources**
- Can be added at **Enterprise, Organization, and Repository level**

```
$ ./config.sh --url <url> --token PAT
```

```
$ sudo ./svc.sh start
```

```
$ sudo ./svc.sh install
```

```
$ ./run.sh
```



Runners / Create self-hosted runner

Adding a self-hosted runner requires that you download, configure, and execute the GitHub Actions Runner. By downloading and configuring the GitHub Actions Runner, you agree to the GitHub Terms of Service or GitHub Corporate Terms of Service, as applicable.

Runner image

○ macOS ● Linux ○ Windows

Architecture

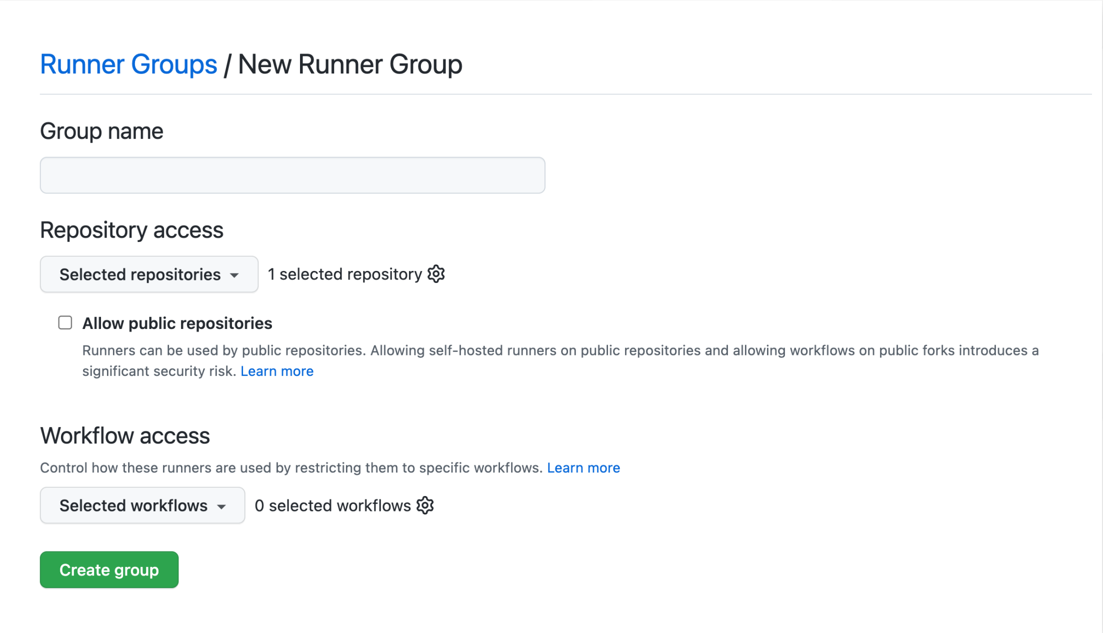
x64

Download

```
# Create a folder
$ mkdir actions-runner && cd actions-runner
# Download the latest runner package
$ curl -o actions-runner-linux-x64-2.289.2.tar.gz -L
https://github.com/actions/runner/releases/download/v2.289.2/actions-runner-linux-x64-
2.289.2.tar.gz
# Optional: Validate the hash
$ echo "7ba89bb75397896a76e98197633c087a9499d4c1db7603f21910e135b0d0a238  actions-runner-linux-x64-
2.289.2.tar.gz" | shasum -a 256 -c
# Extract the installer
$ tar xzf ./actions-runner-linux-x64-2.289.2.tar.gz
```

Self-hosted runners

- Access: Runner Groups
- A runner can only be in 1 group
- Apply labels
  - $ ./config.sh --labels self-hosted,x64,linux
  - runs-on: [self-hosted, linux, X64]



Runner Groups / New Runner Group

Group name

Repository access

Selected repositories ▾ 1 selected repository

□ **Allow public repositories**

Runners can be used by public repositories. Allowing self-hosted runners on public repositories and allowing workflows on public forks introduces a significant security risk. Learn more

Workflow access

Control how these runners are used by restricting them to specific workflows. Learn more

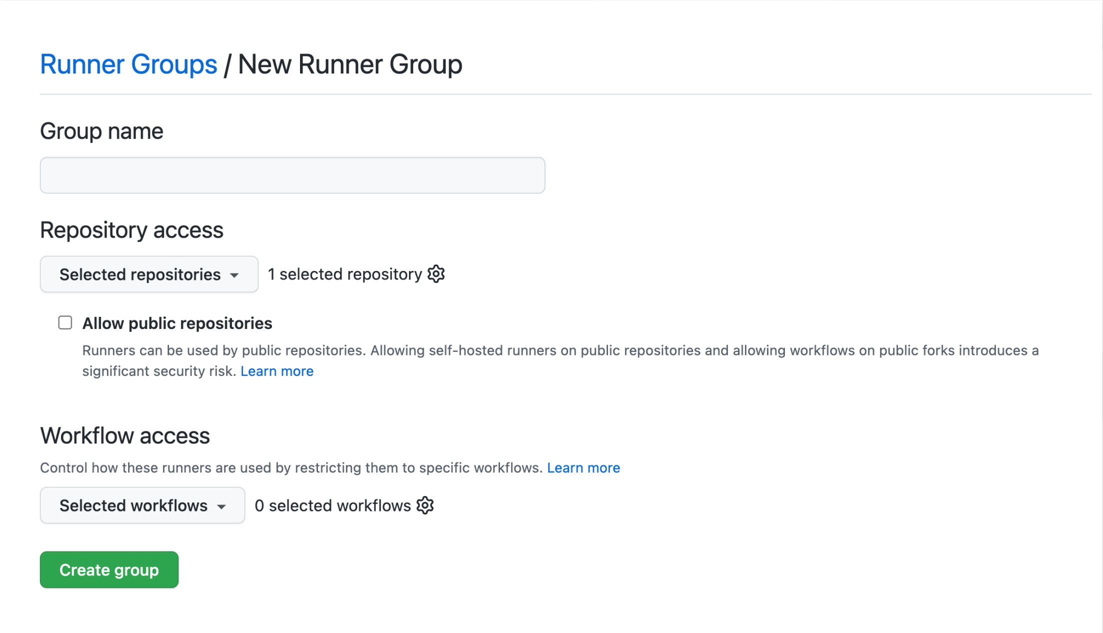
Selected workflows ▾ 0 selected workflows

Create group

Xpirit

Self-hosted runners

- Access: Runner Groups
- A runner can only be in 1 group
- Apply labels
  - $ ./config.sh --labels self-hosted,x64,linux,**matlab**
  - runs-on: [self-hosted, linux, X64, **matlab**]



Runner Groups / New Runner Group

Group name

Repository access

Selected repositories ▾ 1 selected repository

□ **Allow public repositories**

Runners can be used by public repositories. Allowing self-hosted runners on public repositories and allowing workflows on public forks introduces a significant security risk. Learn more

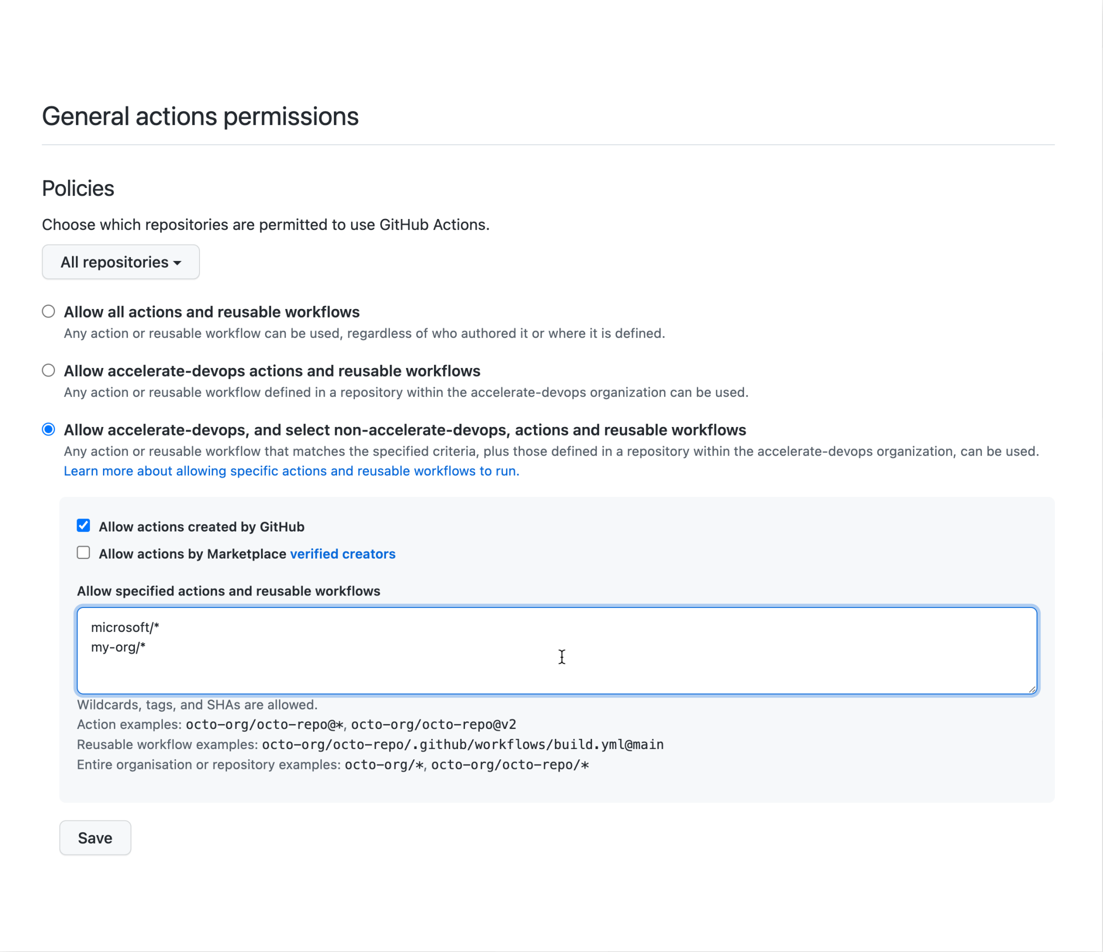
Workflow access

Control how these runners are used by restricting them to specific workflows. Learn more

Selected workflows ▾ 0 selected workflows

Self-hosted runners - gotchas

- Runners are not ephemeral per default - you have to clean up after a build yourself
  - $ ./config.sh --ephemeral
- Use web hooks to auto scale (https://github.com/jonico/awesome-runners)
- Do not allow public repositories!
- Limit Actions and use SHA or fork
- Create a company marketplace (https://github.com/rajbos/actions-marketplace)



General actions permissions

Policies

Choose which repositories are permitted to use GitHub Actions.

All repositories ▾

○ **Allow all actions and reusable workflows**
Any action or reusable workflow can be used, regardless of who authored it or where it is defined.

○ **Allow accelerate-devops actions and reusable workflows**
Any action or reusable workflow defined in a repository within the accelerate-devops organization can be used.

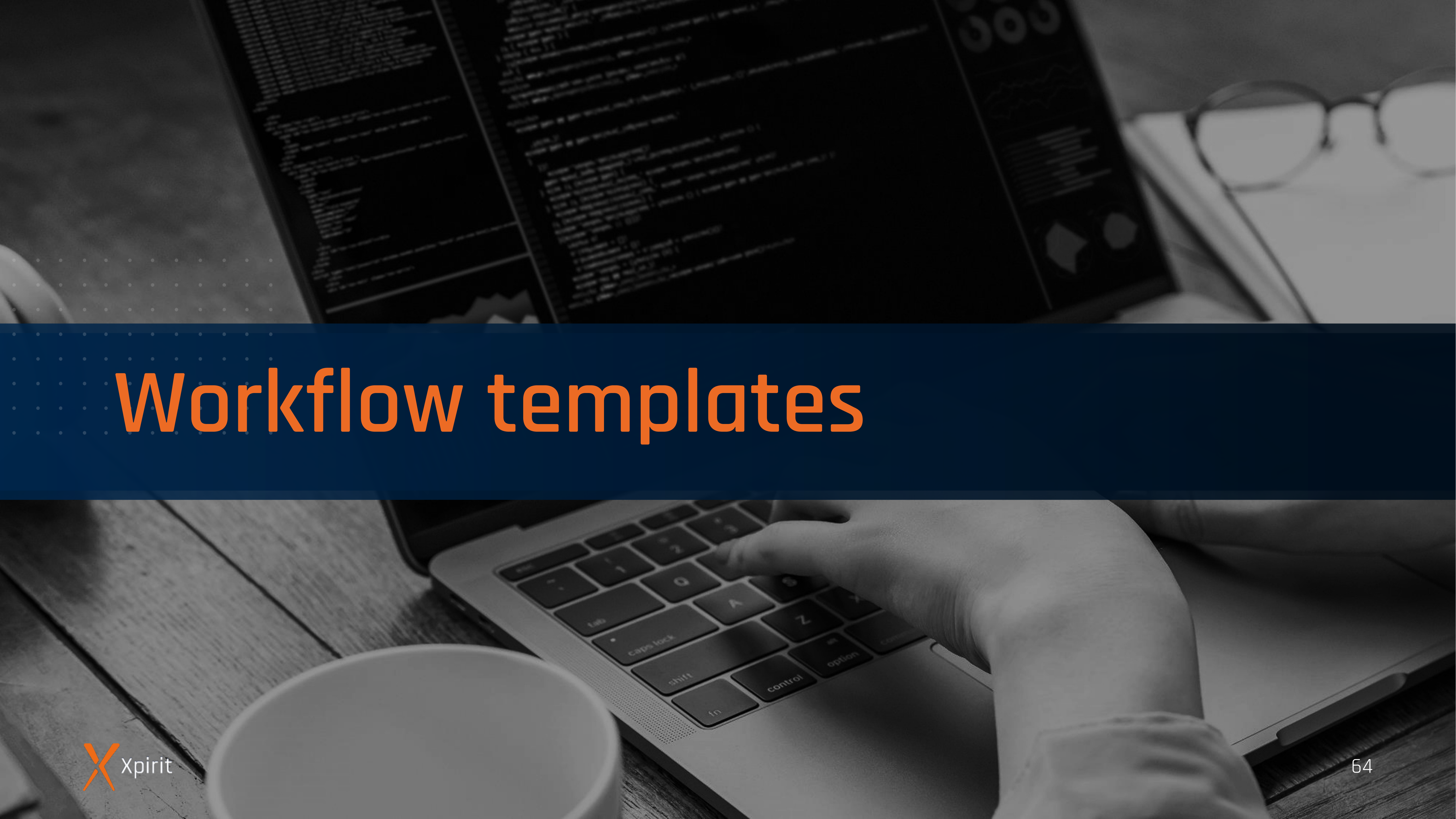
● **Allow accelerate-devops, and select non-accelerate-devops, actions and reusable workflows**
Any action or reusable workflow that matches the specified criteria, plus those defined in a repository within the accelerate-devops organization, can be used.
Learn more about allowing specific actions and reusable workflows to run.

☑ Allow actions created by GitHub
☐ Allow actions by Marketplace verified creators

Allow specified actions and reusable workflows

```
microsoft/*
my-org/*
```

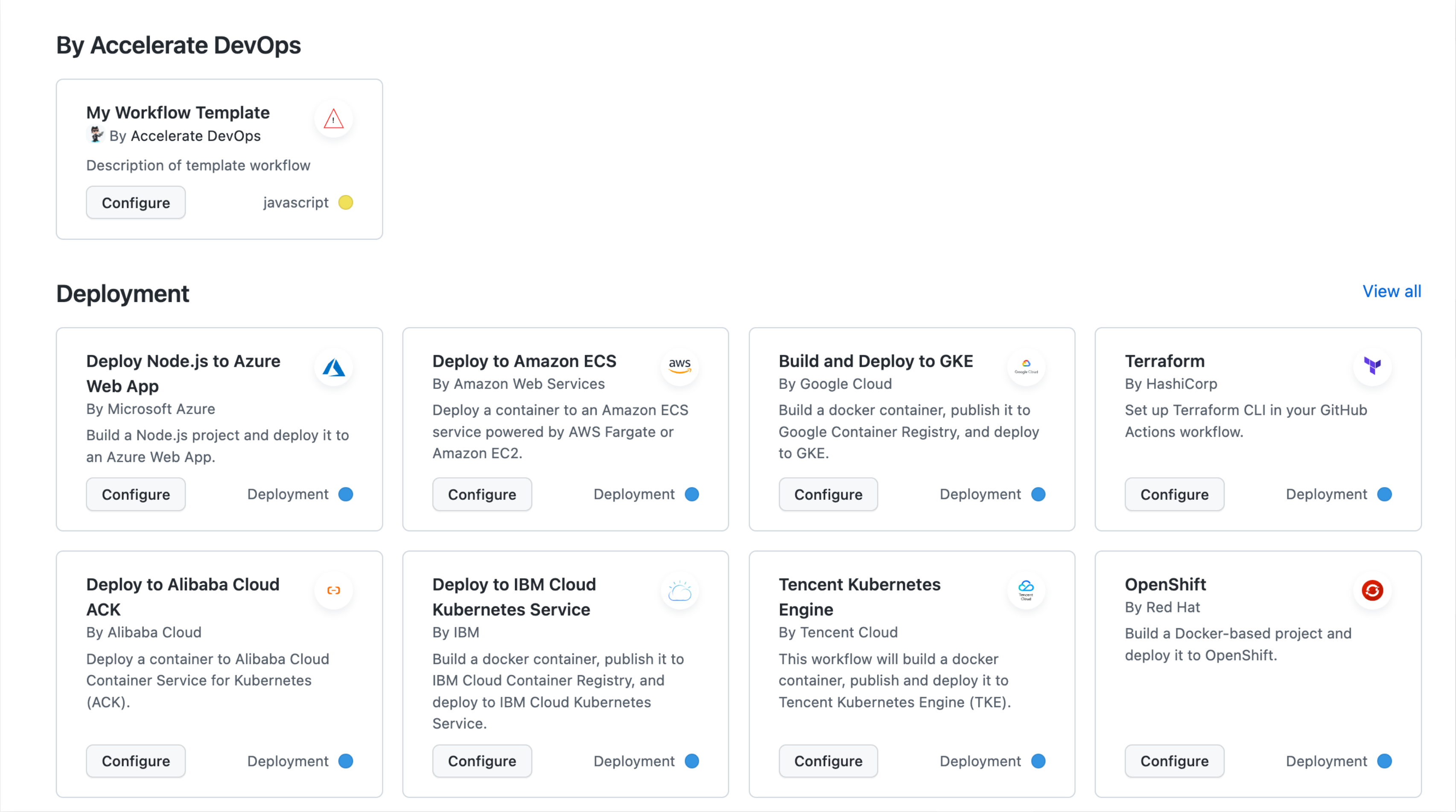
Wildcards, tags, and SHAs are allowed.
Action examples: `octo-org/octo-repo@*`, `octo-org/octo-repo@v2`
Reusable workflow examples: `octo-org/octo-repo/.github/workflows/build.yml@main`
Entire organisation or repository examples: `octo-org/*`, `octo-org/octo-repo/*`



Workflow templates

Workflow templates

- Available in Actions / New workflow
- Get copied one time
- Starter workflows



By Accelerate DevOps

My Workflow Template
By Accelerate DevOps
Description of template workflow
Configure — javascript ●

Deployment

View all

Deploy Node.js to Azure Web App
By Microsoft Azure
Build a Node.js project and deploy it to an Azure Web App.
Configure — Deployment ●

Deploy to Amazon ECS
By Amazon Web Services
Deploy a container to an Amazon ECS service powered by AWS Fargate or Amazon EC2.
Configure — Deployment ●

Build and Deploy to GKE
By Google Cloud
Build a docker container, publish it to Google Container Registry, and deploy to GKE.
Configure — Deployment ●

Terraform
By HashiCorp
Set up Terraform CLI in your GitHub Actions workflow.
Configure — Deployment ●

Deploy to Alibaba Cloud ACK
By Alibaba Cloud
Deploy a container to Alibaba Cloud Container Service for Kubernetes (ACK).
Configure — Deployment ●

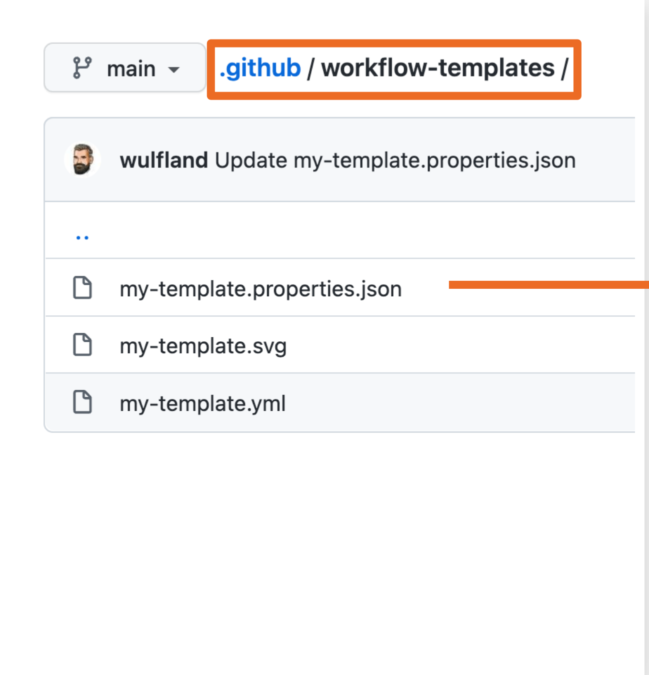
Deploy to IBM Cloud Kubernetes Service
By IBM
Build a docker container, publish it to IBM Cloud Container Registry, and deploy to IBM Cloud Kubernetes Service.
Configure — Deployment ●

Tencent Kubernetes Engine
By Tencent Cloud
This workflow will build a docker container, publish and deploy it to Tencent Kubernetes Engine (TKE).
Configure — Deployment ●

OpenShift
By Red Hat
Build a Docker-based project and deploy it to OpenShift.
Configure — Deployment ●

Workflow templates

<org>/.github/workflow-templates



main .github / workflow-templates /

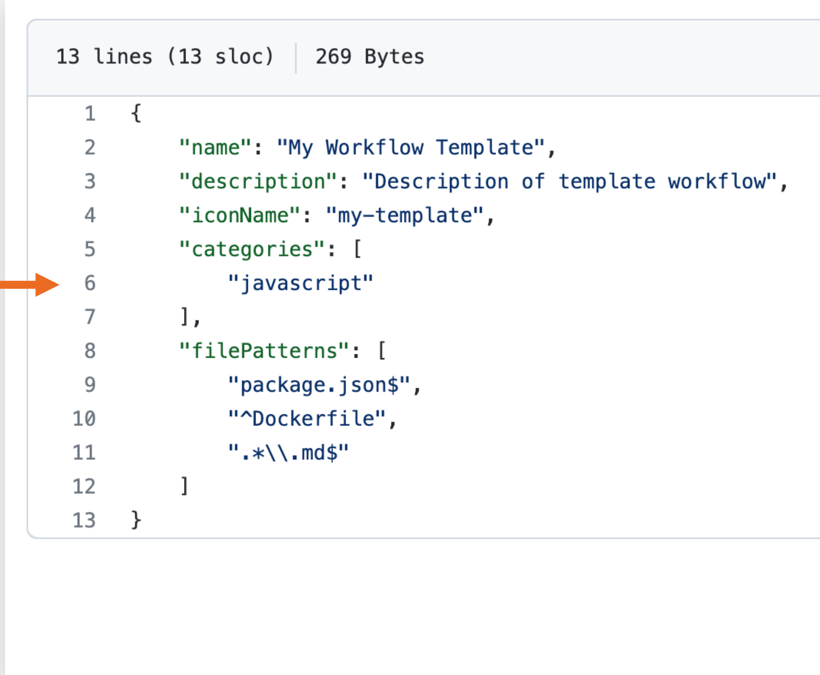
wulfland Update my-template.properties.json

..

my-template.properties.json

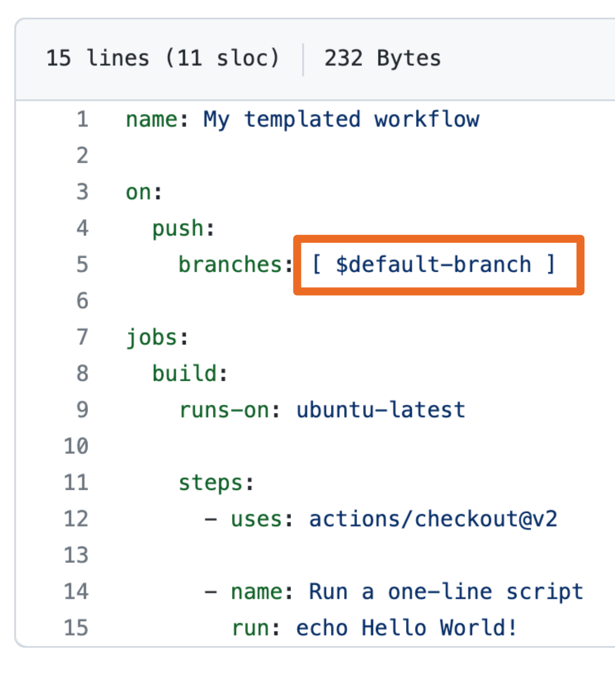
my-template.svg

my-template.yml



13 lines (13 sloc) | 269 Bytes

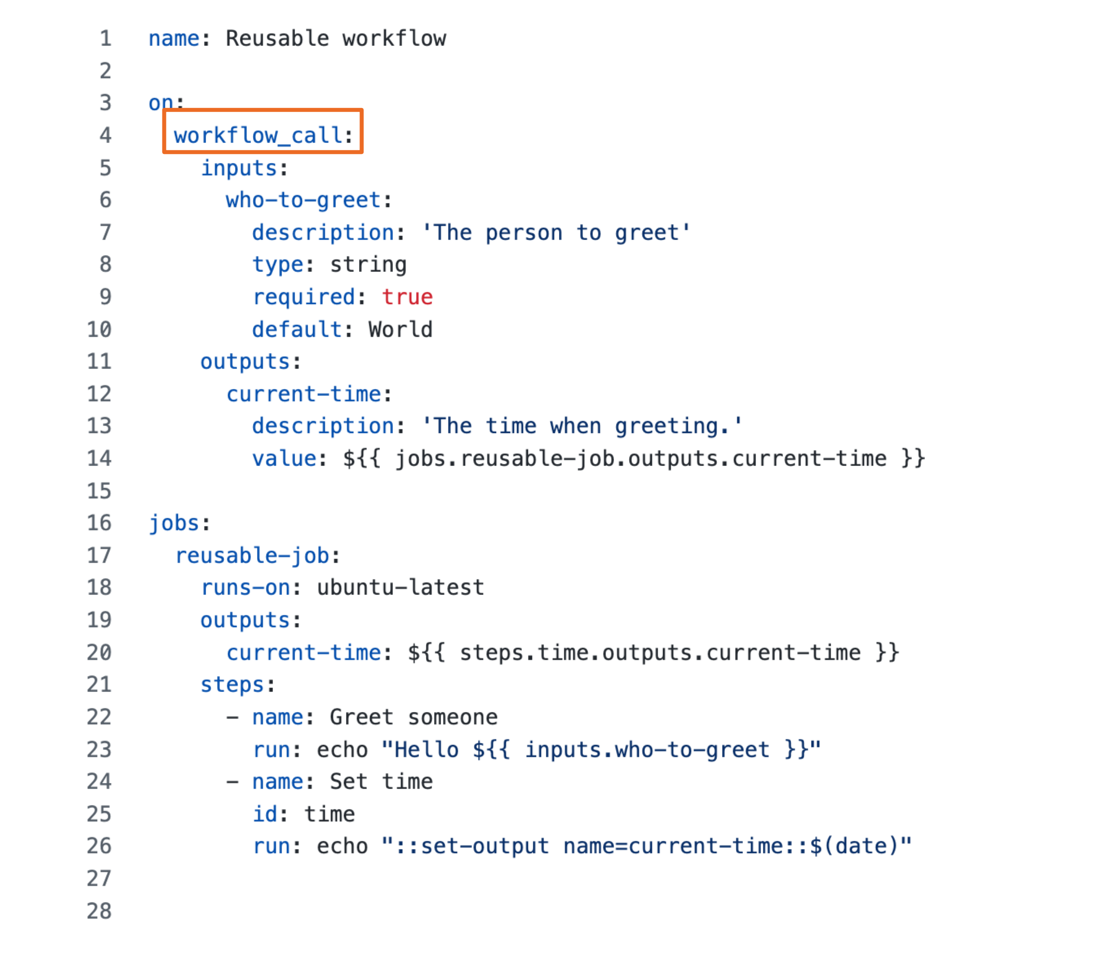
```
{
    "name": "My Workflow Template",
    "description": "Description of template workflow",
    "iconName": "my-template",
    "categories": [
        "javascript"
    ],
    "filePatterns": [
        "package.json$",
        "^Dockerfile",
        ".*\\.md$"
    ]
}
```



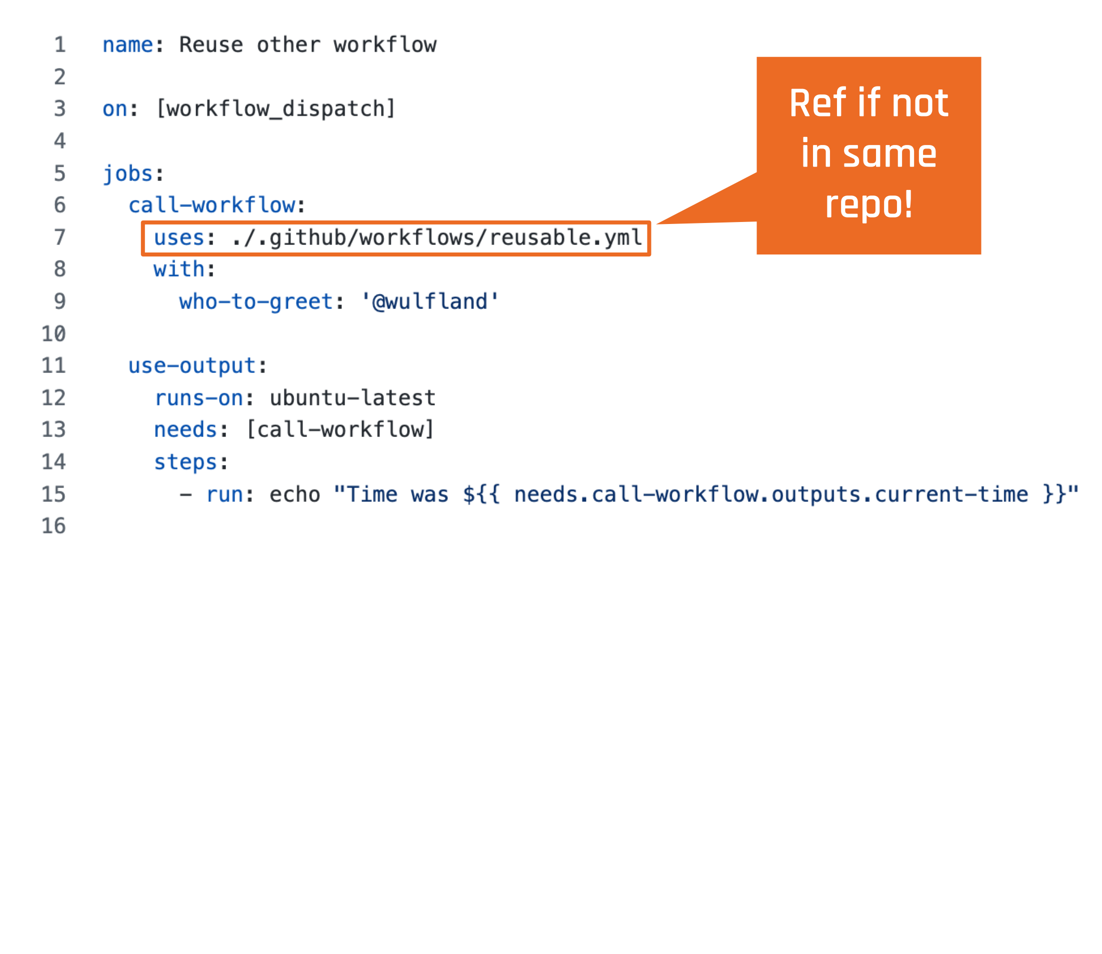
15 lines (11 sloc) | 232 Bytes

```
name: My templated workflow

on:
  push:
    branches: [ $default-branch ]

jobs:
  build:
    runs-on: ubuntu-latest

    steps:
      - uses: actions/checkout@v2

      - name: Run a one-line script
        run: echo Hello World!
```

Reusable workflows



```yaml
name: Reusable workflow

on:
  workflow_call:
    inputs:
      who-to-greet:
        description: 'The person to greet'
        type: string
        required: true
        default: World
    outputs:
      current-time:
        description: 'The time when greeting.'
        value: ${{ jobs.reusable-job.outputs.current-time }}

jobs:
  reusable-job:
    runs-on: ubuntu-latest
    outputs:
      current-time: ${{ steps.time.outputs.current-time }}
    steps:
      - name: Greet someone
        run: echo "Hello ${{ inputs.who-to-greet }}"
      - name: Set time
        id: time
        run: echo "::set-output name=current-time::$(date)"


```

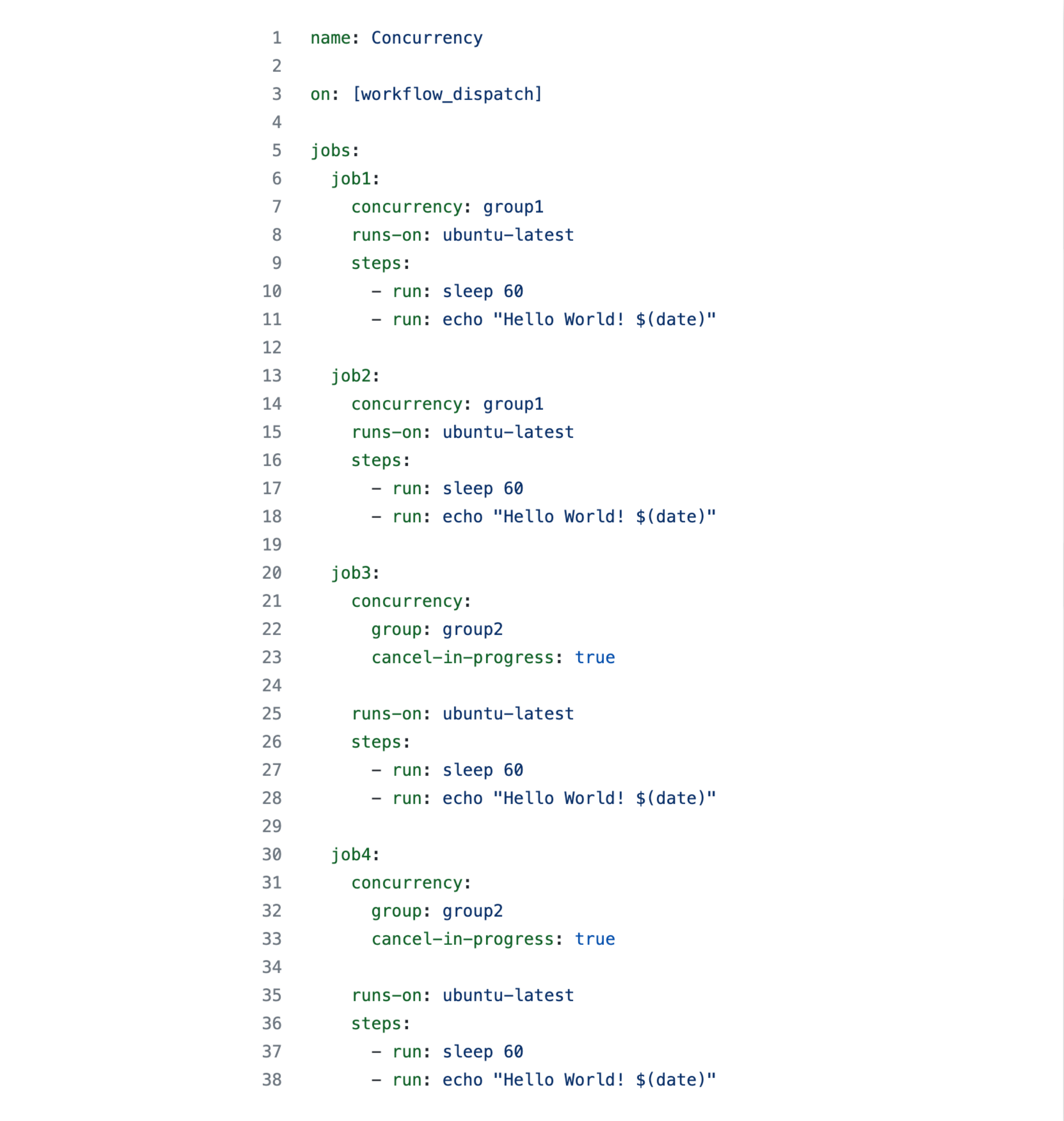


```yaml
name: Reuse other workflow

on: [workflow_dispatch]

jobs:
  call-workflow:
    uses: ./.github/workflows/reusable.yml
    with:
      who-to-greet: '@wulfland'

  use-output:
    runs-on: ubuntu-latest
    needs: [call-workflow]
    steps:
      - run: echo "Time was ${{ needs.call-workflow.outputs.current-time }}"

```

Ref if not in same repo!

Concurrency

- Workflow or job
- Optional: cancel in-progress jobs
- Use cases:
  - Wait job/workflow until deployment completed
  - Cancel deployment and deploy newer version instead



```yaml
name: Concurrency

on: [workflow_dispatch]

jobs:
  job1:
    concurrency: group1
    runs-on: ubuntu-latest
    steps:
      - run: sleep 60
      - run: echo "Hello World! $(date)"

  job2:
    concurrency: group1
    runs-on: ubuntu-latest
    steps:
      - run: sleep 60
      - run: echo "Hello World! $(date)"

  job3:
    concurrency:
      group: group2
      cancel-in-progress: true

    runs-on: ubuntu-latest
    steps:
      - run: sleep 60
      - run: echo "Hello World! $(date)"

  job4:
    concurrency:
      group: group2
      cancel-in-progress: true

    runs-on: ubuntu-latest
    steps:
      - run: sleep 60
      - run: echo "Hello World! $(date)"
```

Let‘s connect

@mike_kaufmann

@wulfland

https://writeabout.net

Let's connect!

in

Michael Kaufmann

Founder/MD Xpirit Germany
Microsoft Regional Director, MVP
Linkedin.com/in/mikaufmann
mkaufmann@xpirit.com