

Concurrency Management in GitHub Actions

Concurrency management in GitHub Actions allows you to control and limit the number of jobs or workflows that can run simultaneously. This is useful to prevent job conflicts and resource contention, especially in multi-job workflows.

- [Overview](#)
- [Testing Concurrency Management](#)

Overview

Concurrency management in GitHub Actions is controlled using the `concurrency` keyword in your workflow YAML file. With this feature, you can specify which jobs or workflows should be allowed to run concurrently and which should be queued.

By defining a concurrency group, which can be any string or expression, you ensure that only one job or workflow using the same concurrency group will run at a time. If a job or workflow is queued and another with the same concurrency group is in progress, the queued one will be pending. It's possible to cancel any currently running job or workflow in the same concurrency group using the `cancel-in-progress: true` option.

Notes:

- Concurrency group names are case-insensitive.
- Ordering is not guaranteed for jobs or runs using concurrency groups; they are handled in the order they are processed.

Testing Concurrency Management

1. Concurrency Configuration (24-01-Concurrency):

```
name: 24-01-Concurrency

on:
  workflow_dispatch:

concurrency:
  group: ${{ github.workflow }}-${{ github.ref }}

jobs:
  ping:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
      - name: Ping URL
        id: ping-url
        uses: ../.github/actions/docker-ping-url
        with:
          url: https://www.invalid-ubfadufhbaudhfua.com
```

```
max_trials: 20
delay: 5
```

- This workflow defines a concurrency group based on the workflow and reference (branch/tag).
- The concurrency group prevents concurrent executions of workflows triggered by the same workflow and reference combination.

2. Workflow with Concurrency (24-02-Concurrency):

```
name: 24-02-Concurrency

on:
  workflow_dispatch:

jobs:
  ping-with-concurrency:
    runs-on: ubuntu-latest
    concurrency:
      group: ${{ github.workflow }}-${{ github.ref }}
    steps:
      - uses: actions/checkout@v4
      - name: Ping URL
        id: ping-url
        uses: ../github/actions/docker-ping-url
        with:
          url: https://www.invalid-ubfadufhbaudhfua.com
          max_trials: 20
          delay: 5
  ping-without-concurrency:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
      - name: Ping URL
        id: ping-url
        uses: ../github/actions/docker-ping-url
        with:
          url: https://www.invalid-ubfadufhbaudhfua.com
          max_trials: 20
          delay: 5
```

- This workflow includes two jobs: `ping-with-concurrency` and `ping-without-concurrency`.
- The `ping-with-concurrency` job is configured with concurrency control, ensuring that only one instance of this job runs at a time within the concurrency group defined in the workflow.
- The `ping-without-concurrency` job does not have concurrency control, meaning multiple instances of this job can run concurrently.

3. Workflow Testing Steps:

- Commit Workflows:

- Commit the `24-01-Concurrency.yml` and `24-02-Concurrency.yml` files to the `.github/workflows` directory in your repository.
- Trigger Workflows:
 - Manually trigger both workflows (`24-01-Concurrency` and `24-02-Concurrency`) from the GitHub Actions UI.
- Review Logs:
 - Check the logs of both workflows to ensure that the jobs are executed as expected.
 - For `24-02-Concurrency`, observe how the job with concurrency control (`ping-with-concurrency`) behaves compared to the job without concurrency control (`ping-without-concurrency`).
- Concurrency Testing:
 - Manually trigger the workflows multiple times to simulate concurrent executions.
 - Verify that the jobs within the same concurrency group do not execute concurrently for `24-02-Concurrency`.
 - Observe any differences in execution behavior between jobs with and without concurrency control.