Choosing the runner for a job

Define the type of machine that will process a job in your workflow.

Overview

Use `jobs.<job_id>.runs-on` to define the type of machine to run the job on.

- The destination machine can be either a GitHub-hosted runner, larger runner, or a self-hosted runner.
- You can target runners based on the labels assigned to them, or their group membership, or a combination of these.
- You can provide `runs-on` as:
  - a single string
  - a single variable containing a string
  - an array of strings, variables containing strings, or a combination of both
  - a `key: value` pair using the `group` or `label` keys
- If you specify an array of strings or variables, your workflow will execute on any runner that matches all of the specified `runs-on` values. For example, here the job will only run on a self-hosted runner that has the labels `linux`, `x64`, and `gpu`:

  ```
  runs-on: [self-hosted, linux, x64, gpu]
  ```

  For more information, see "Choosing self-hosted runners."
- You can mix strings and variables in an array. For example:

  ```
  on:
    workflow_dispatch:
      inputs:
        chosen-os:
          required: true
          type: choice
          options:
          - Ubuntu
          - macOS

  jobs:
    test:
      runs-on: [self-hosted, "${{ inputs.chosen-os }}"]
      steps:
      - run: echo Hello world!
  ```
- If you would like to run your workflow on multiple machines, use `jobs.<job_id>.strategy`.

Choosing GitHub-hosted runners

If you use a GitHub-hosted runner, each job runs in a fresh instance of a runner image specified by `runs-on`.

Available GitHub-hosted runner types are:

Virtual Machine	Processor (CPU)	Memory (RAM)	Storage (SSD)	OS (YAML workflow label)	Notes
Linux	2	7 GB	14 GB	`ubuntu-latest`, `ubuntu-22.04`, `ubuntu-20.04`	The `ubuntu-latest` label currently uses the Ubuntu 22.04 runner image.
Windows	2	7 GB	14 GB	`windows-latest`, `windows-2022`, `windows-2019`	The `windows-latest` label currently uses the Windows 2022 runner image.
macOS	3	14 GB	14 GB	`macos-latest`, `macos-12`, `macos-11`	The `macos-latest` workflow label currently uses the macOS 12 runner image.
macOS	4	14 GB	14 GB	`macos-13` [Beta]	N/A

Note: The `-latest` runner images are the latest stable images that GitHub provides, and might not be the most recent version of the operating system available from the operating system vendor.

Warning: Beta and Deprecated Images are provided "as-is", "with all faults" and "as available" and are excluded from the service level agreement and warranty. Beta Images may not be covered by customer support.

Example: Specifying an operating system

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

For more information, see "Using GitHub-hosted runners."

Choosing self-hosted runners

To specify a self-hosted runner for your job, configure `runs-on` in your workflow file with self-hosted runner labels.

All self-hosted runners have the `self-hosted` label. Using only this label will select any self-hosted runner. To select runners that meet certain criteria, such as operating system or architecture, we recommend providing an array of labels that begins with `self-hosted` (this must be listed first) and then includes additional labels as needed. When you specify an array of labels, jobs will be queued on runners that have all the labels that you specify.

Although the `self-hosted` label is not required, we strongly recommend specifying it when using self-hosted runners to ensure that your job does not unintentionally specify any current or future GitHub-hosted runners.

Example: Using labels for runner selection

```yaml
runs-on: [self-hosted, linux]
```

For more information, see "About self-hosted runners" and "Using self-hosted runners in a workflow."

Choosing runners in a group

You can use `runs-on` to target runner groups, so that the job will execute on any runner that is a member of that group. For more granular control, you can also combine runner groups with labels.

Runner groups can only have larger runners or self-hosted runners as members.

Example: Using groups to control where jobs are run

In this example, Ubuntu runners have been added to a group called `ubuntu-runners`. The `runs-on` key sends the job to any available runner in the `ubuntu-runners` group:

```yaml
name: learn-github-actions
on: [push]
jobs:
  check-bats-version:
    runs-on:
      group: ubuntu-runners
    steps:
      - uses: actions/checkout@v4
      - uses: actions/setup-node@v3
        with:
          node-version: '14'
      - run: npm install -g bats
      - run: bats -v
```

Example: Combining groups and labels

When you combine groups and labels, the runner must meet both requirements to be eligible to run the job.

In this example, a runner group called `ubuntu-runners` is populated with Ubuntu runners, which have also been assigned the label `ubuntu-20.04-16core`. The `runs-on` key combines `group` and `labels` so that the job is routed to any available runner within the group that also has a matching label:

```yaml
name: learn-github-actions
on: [push]
jobs:
  check-bats-version:
    runs-on:
      group: ubuntu-runners
      labels: ubuntu-20.04-16core
    steps:
      - uses: actions/checkout@v4
      - uses: actions/setup-node@v3
        with:
          node-version: '14'
      - run: npm install -g bats
      - run: bats -v
```