

Running Steps on the Host

Overview

When you need to execute your step directly on the host node, and not in a container, you can control this through the Pipelines DSL.

Running steps in a container on the build node ensures that a pipeline produces the same result every time it is run. The host node does not guarantee this repeatability, since the step will execute in whatever state the host was left in when it was last used. For this reason, the default for Pipelines steps is to execute in a container.

However, there may be cases where you need to execute your step directly on the host node. For example, you may need to execute Docker commands and don't want to do so from within a Docker container.

In these cases, you can configure the Pipelines DSL to execute your step or all steps of a pipeline directly on the host node.

Page Contents

- [Overview](#)
- [Configuring a Step](#)
- [Configuring a Pipeline](#)

Configuring a Step

You can configure a single step to execute directly on the host node in the `runtime` section of your step by specifying the `type` as `host`:

```
pipelines:
- name: pipeline_example
  steps:
  - name: step_1
    type: Bash                # may be any step type
    configuration:
      runtime:
        type: host
    execution:
      ...
```

You should not add any `image` section; that will cause pipeline source sync to fail.

Configuring a Pipeline

You can configure a pipeline to execute all steps by default directly on the host node in the `runtime` section of your pipeline by specifying the `type` as `host`:

```
pipelines:
- name: pipeline_host
  configuration:
    runtime:
      type: host
```

You should not add any `image` section; that will cause pipeline source sync to fail.

You may override this pipeline's default in any step with a different `runtime` specification.