Automating migration with GitHub Actions Importer

In this article

Use GitHub Actions Importer to plan and automate your migration to GitHub Actions.

Legal notice

About GitHub Actions Importer

You can use GitHub Actions Importer to plan and automatically migrate your CI/CD supported pipelines to GitHub Actions.

GitHub Actions Importer is distributed as a Docker container, and uses a GitHub CLI extension to interact with the container.

Any workflow that is converted by the GitHub Actions Importer should be inspected for correctness before using it as a production workload. The goal is to achieve an 80% conversion rate for every workflow, however, the actual conversion rate will depend on the makeup of each individual pipeline that is converted.

Supported CI platforms

You can use GitHub Actions Importer to migrate from the following platforms:

- Azure DevOps
- Bamboo
- Bitbucket Pipelines
- CircleCI
- GitLab
- Jenkins
- Travis CI

Prerequisites

GitHub Actions Importer has the following requirements:

- An environment where you can run Linux-based containers, and can install the necessary tools.
    - Docker is installed and running.
    - GitHub CLI is installed.

> **Note**: The GitHub Actions Importer container and CLI do not need to be installed on the same server as your CI platform.

Installing the GitHub Actions Importer CLI extension

1. Install the GitHub Actions Importer CLI extension:

    Bash

    ```
    gh extension install github/gh-actions-importer
    ```

2. Verify that the extension is installed:

    ```
    $ gh actions-importer -h
    Options:
      -?, -h, --help  Show help and usage information

    Commands:
      update     Update to the latest version of GitHub Actions Importer.
      version    Display the version of GitHub Actions Importer.
      configure  Start an interactive prompt to configure credentials used to
    authenticate with your CI server(s).
      audit      Plan your CI/CD migration by analyzing your current CI/CD
    footprint.
      forecast   Forecast GitHub Actions usage from historical pipeline
    utilization.
      dry-run    Convert a pipeline to a GitHub Actions workflow and output its
    yaml file.
      migrate    Convert a pipeline to a GitHub Actions workflow and open a pull
    request with the changes.
    ```

Updating the GitHub Actions Importer CLI

To ensure you're running the latest version of GitHub Actions Importer, you should regularly run the `update` command:

```
gh actions-importer update
```

Authenticating at the command line

You must configure credentials that allow GitHub Actions Importer to communicate with GitHub and your current CI server. You can configure these credentials using environment variables or a `.env.local` file. The environment variables can be configured in an interactive prompt, by running the following command:

```
gh actions-importer configure
```

Using the GitHub Actions Importer CLI

Use the subcommands of `gh actions-importer` to begin your migration to GitHub Actions, including `audit`, `forecast`, `dry-run`, and `migrate`.

Auditing your existing CI pipelines

The `audit` subcommand can be used to plan your CI/CD migration by analyzing your current CI/CD footprint. This analysis can be used to plan a timeline for migrating to GitHub Actions.

To run an audit, use the following command to determine your available options:

```
$ gh actions-importer audit -h
Description:
  Plan your CI/CD migration by analyzing your current CI/CD footprint.

[...]

Commands:
  azure-devops  An audit will output a list of data used in an Azure DevOps
instance.
  bamboo        An audit will output a list of data used in a Bamboo instance.
  circle-ci     An audit will output a list of data used in a CircleCI instance.
  gitlab        An audit will output a list of data used in a GitLab instance.
  jenkins       An audit will output a list of data used in a Jenkins instance.
  travis-ci     An audit will output a list of data used in a Travis CI instance.
```

Forecasting usage

The `forecast` subcommand reviews historical pipeline usage to create a forecast of GitHub Actions usage.

To run a forecast, use the following command to determine your available options:

```
$ gh actions-importer forecast -h
Description:
  Forecasts GitHub Actions usage from historical pipeline utilization.

[...]

Commands:
  azure-devops  Forecasts GitHub Actions usage from historical Azure DevOps
pipeline utilization.
  bamboo        Forecasts GitHub Actions usage from historical Bamboo pipeline
utilization.
  jenkins       Forecasts GitHub Actions usage from historical Jenkins pipeline
utilization.
  gitlab        Forecasts GitHub Actions usage from historical GitLab pipeline
utilization.
  circle-ci     Forecasts GitHub Actions usage from historical CircleCI pipeline
utilization.
  travis-ci     Forecasts GitHub Actions usage from historical Travis CI pipeline
utilization.
  github        Forecasts GitHub Actions usage from historical GitHub pipeline
utilization.
```

Testing the migration process

The `dry-run` subcommand can be used to convert a pipeline to its GitHub Actions equivalent, and then write the workflow to your local filesystem.

To perform a dry run, use the following command to determine your available options:

```
$ gh actions-importer dry-run -h
Description:
  Convert a pipeline to a GitHub Actions workflow and output its yaml file.

[...]

Commands:
  azure-devops  Convert an Azure DevOps pipeline to a GitHub Actions workflow and
output its yaml file.
  bamboo        Convert a Bamboo pipeline to GitHub Actions workflows and output
its yaml file.
  circle-ci     Convert a CircleCI pipeline to GitHub Actions workflows and
output the yaml file(s).
  gitlab        Convert a GitLab pipeline to a GitHub Actions workflow and output
the yaml file.
  jenkins       Convert a Jenkins job to a GitHub Actions workflow and output its
yaml file.
  travis-ci     Convert a Travis CI pipeline to a GitHub Actions workflow and
output its yaml file.
```

Migrating a pipeline to GitHub Actions

The `migrate` subcommand can be used to convert a pipeline to its GitHub Actions equivalent and then create a pull request with the contents.

To run a migration, use the following command to determine your available options:

```
$ gh actions-importer migrate -h
Description:
  Convert a pipeline to a GitHub Actions workflow and open a pull request with
the changes.

[...]

Commands:
  azure-devops  Convert an Azure DevOps pipeline to a GitHub Actions workflow and
open a pull request with the changes.
  bamboo        Convert a Bamboo pipeline to GitHub Actions workflows and open a
pull request with the changes.
  circle-ci     Convert a CircleCI pipeline to GitHub Actions workflows and open
a pull request with the changes.
  gitlab        Convert a GitLab pipeline to a GitHub Actions workflow and open a
pull request with the changes.
  jenkins       Convert a Jenkins job to a GitHub Actions workflow and open a
pull request with the changes.
  travis-ci     Convert a Travis CI pipeline to a GitHub Actions workflow and and
open a pull request with the changes.
```

Performing self-serve migrations using IssueOps

You can use GitHub Actions and GitHub Issues to run CLI commands for GitHub Actions Importer. This allows you to migrate your CI/CD workflows without installing software on your local machine. This approach is especially useful for organizations that want to enable self-service migrations to GitHub Actions. Once IssueOps is configured, users can open an issue with the relevant template to migrate pipelines to GitHub Actions.

For more information about setting up self-serve migrations with IssueOps, see the `actions/importer-issue-ops` template repository.

Using the GitHub Actions Importer labs repository

The GitHub Actions Importer labs repository contains platform-specific learning paths that teach you how to use GitHub Actions Importer and how to approach migrations to GitHub Actions. You can use this repository to learn how to use GitHub Actions Importer to help plan, forecast, and automate your migration to GitHub Actions.

To learn more, see the GitHub Actions Importer labs repository.

Legal notice

Portions have been adapted from https://github.com/github/gh-actions-importer/ under the MIT license: