Matrix targets

- Release and debug variants
- Multiple axes
- Multiple values per matrix target

A matrix strategy lets you fork a single target into multiple different variants, based on parameters that you specify. This works in a similar way to Matrix strategies for GitHub Actions. You can use this to reduce duplication in your Bake definition.

The matrix attribute is a map of parameter names to lists of values. Bake builds each possible combination of values as a separate target.

Each generated target must have a unique name. To specify how target names should resolve, use the name attribute.

The following example resolves the app target to `app-foo` and `app-bar`. It also uses the matrix value to define the target build stage.

```
target "app" {
  name = "app-${tgt}"
  matrix = {
    tgt = ["foo", "bar"]
  }
  target = tgt
}
```

```
$ docker buildx bake --print app
[+] Building 0.0s (0/0)
{
  "group": {
    "app": {
      "targets": [
        "app-foo",
        "app-bar"
```

```
      ]
    },
    "default": {
      "targets": [
        "app"
      ]
    }
  },
  "target": {
    "app-bar": {
      "context": ".",
      "dockerfile": "Dockerfile",
      "target": "bar"
    },
    "app-foo": {
      "context": ".",
      "dockerfile": "Dockerfile",
      "target": "foo"
    }
  }
}
```

Release and debug variants

Use a matrix value in build arguments and tags to produce distinct image variants. For a Dockerfile that accepts a `BUILD_MODE` argument:

```
target "app" {
  name = "app-${mode}"
  matrix = {
    mode = ["release", "debug"]
  }
  args = {
    BUILD_MODE = mode
  }
  tags = ["docker.io/username/app:${mode}"]
}
```

For file exports, also give each variant a separate output directory to avoid overwriting files from another variant. Using the `bin` target and Dockerfile from Export build artifacts, add:

```
target "bin-all" {
  inherits = ["bin"]
  name = "bin-${mode}"
  matrix = {
    mode = ["release", "debug"]
  }
  args = {
    BUILD_MODE = mode
  }
  output = ["type=local,dest=build/bin/${mode}"]
}
```

```
$ docker buildx bake --print app
$ docker buildx bake bin-all
```

The `app-release` and `app-debug` targets use the tags `docker.io/username/app:release` and `docker.io/username/app:debug`, respectively. The artifact targets write `build/bin/release/app` and `build/bin/debug/app`.

Multiple axes

You can specify multiple keys in your matrix to fork a target on multiple axes. When using multiple matrix keys, Bake builds every possible variant.

The following example builds four targets:

- `app-foo-1-0`
- `app-foo-2-0`
- `app-bar-1-0`
- `app-bar-2-0`

```
target "app" {
  name = "app-${tgt}-${replace(version, ".", "-")}"
  matrix = {
    tgt = ["foo", "bar"]
    version = ["1.0", "2.0"]
```

```
  }
  target = tgt
  args = {
    VERSION = version
  }
}
```

Multiple values per matrix target

If you want to differentiate the matrix on more than just a single value, you can use maps as matrix values. Bake creates a target for each map, and you can access the nested values using dot notation.

The following example builds two targets:

- `app-foo-1-0`
- `app-bar-2-0`

```
target "app" {
  name = "app-${item.tgt}-${replace(item.version, ".", "-")}"
  matrix = {
    item = [
      {
        tgt = "foo"
        version = "1.0"
      },
      {
        tgt = "bar"
        version = "2.0"
      }
    ]
  }
  target = item.tgt
  args = {
    VERSION = item.version
  }
}
```

□□□ #2
□ □□□ □□□ 2026-10-31 19:03:28 UTC
□ □□□ □□□ 2026-10-31 19:03:30 UTC