Sam Caldwell

Asymmetric Effort, LLC.

Monorepo Tools

mONOREPO COMMAND

Table of Contents

[Monorepo Command 2](#_Toc144925394)

[General Commands 3](#_Toc144925395)

[Global Options 4](#_Toc144925396)

[General Command Structure 5](#_Toc144925397)

[Monorepo Configuration 6](#_Toc144925398)

[Build-Host Runtime Configuration 7](#_Toc144925399)

[Container-Runtime Configuration 9](#_Toc144925400)

[Hypervisor Configuration 11](#_Toc144925401)

[Language Configuration 13](#_Toc144925402)

[Platform Configuration 15](#_Toc144925403)

[Project Configuration 17](#_Toc144925404)

[Project-Action Commands 19](#_Toc144925405)

[Project Build Command 20](#_Toc144925406)

[Project Linter Command 21](#_Toc144925407)

[Project Test Runner Command 22](#_Toc144925408)

[General Task Shell Command 23](#_Toc144925409)

# Monorepo Command

This document describes a ‘monorepo’ tool for interacting with and managing a git monorepo for all Asymmetric Effort projects. Specifically, this command will allow the user to—

* Create, delete, list and configure projects.
* Configure the monorepo tool.
* Build, Lint, Scan and Test projects.
* Distribute Build, Lint, Scan and Test tasks across one or more computer systems.

## General Commands

The following general commands will terminate execution if their corresponding options are encountered.

| Command | Description |
| --- | --- |
| monorepo -h | --help | Show the command usage. |
| monorepo -v | --version | Show the command's current version. |

## Global Options

All non-general commands support the following options:

| Option | Description |
| --- | --- |
| --color | Display all output using ANSI color |
| --debug | Print verbose logging messages |
| --noop | Execute command without performing any changes. |

## General Command Structure

The monorepo command follows this general syntax:

monorepo [command] [object | scope] [parameters]

## Monorepo Configuration

The monorepo configuration is a local configuration persisted as part of the monorepo and ‘.gitconfig,’ managed by the monorepo command using the syntax—

monorepo config [object] [parameters]

This includes configuration of—

* Build hosts which will run tasks
* The git repo itself
* Projects within the monorepo
* The supported languages, container runtimes, hypervisors, operating systems, CPU architectures and interconnected build hosts.

The “monorepo config” commands generally create YAML configuration files within the monorepo as well as storing some configuration state in the git config (key-value) store. Configuration files are used to store parameters

### Build-Host Runtime Configuration

The monorepo can execute tasks across one or more diverse build hosts, interconnected via key-based SSH. The following commands allow the monorepo to configure build host configuration:

| Command | Description |
| --- | --- |
| monorepo config host list | List the currently configured build hosts. |
| monorepo config host show <name> | Show the RESOLVED configuration for a given build host, where the resolved configuration is the configuration state after all configuration is loaded, validated and resolved in order of precedence. |
| monorepo config host delete <name> | Delete the named host configuration. |
| monorepo config host create <name> [Options] | Create a new host configuration for the monorepo using the given name and options. |
| monorepo config host enable <name> | Enable a given build host. |
| monorepo config host disable <name> | Disable a given build host. |
| monorepo config host check <name> | Execute a test command on a given build host to verify connectivity. |

#### Build-Host Create Options

| Option | Description |
| --- | --- |
| --host <string> | The IP address or FQDN and port for the target host.  If localhost is used, no port is required since local execution will be used. |
| --enabled | Sets the host to enabled (default is disabled). |
| --username | Declare the username used for connectivity. If not specified the current user is used. Assumes all authentication is key-based. |
| --platform <name> | Specify the target host's platform. |
| --hypervisor | Define what hypervisor is expected on remote host. |
| --cri | Define what container runtime. is expected on remote host. |

### Container-Runtime Configuration

The following commands allow the user to manage the container runtimes used by supported platforms:

| Command | Description |
| --- | --- |
| monorepo config cri list | List the supported container runtimes (e.g., docker, Kubernetes). |
| monorepo config cri create <name> [Options] | Create a new container runtime interface (CRI) configuration. |
| monorepo config cri delete <name> | Delete a container runtime interface (CRI) configuration. |
| monorepo config cri show <name> | Show the current container runtime interface configuration. |

Given that any platform (operating system/CPU combination) could have an installed container runtime (e.g., docker), this command allows that container runtime to be defined as a supported CRI which can be mapped to such a platform.

#### Create CRI Options

| Option | Description |
| --- | --- |
| --enabled | Enable the resulting config. (Default is disabled) |
| --platform <name> | Specify the target host's platform. |

### Hypervisor Configuration

The following commands allow the user to manage the supported hypervisors:

| Command | Description |
| --- | --- |
| monorepo config hypervisor list | List the supported hypervisors |
| monorepo config hypervisor create <name> [Options] | Create a new hypervisor configuration. |
| monorepo config hypervisor delete <name> | Delete a hypervisor configuration. |
| monorepo config hypervisor show <name> | Show the current hypervisor configuration. |
| monorepo config hypervisor enable <name> | enable the given hypervisor |
| monorepo config hypervisor disable <name> | disable the given hypervisor |

Given that any platform (operating system/CPU combination) could have an installed hypervisor (e.g., kvm or VMware), this command allows that hypervisor to be defined as a supported CRI which can be mapped to such a platform.

#### Create Hypervisor Options

| Option | Description |
| --- | --- |
| --enabled | Enable the resulting config. (Default is disabled) |
| --platform <name> | Specify the target host's platform. |

### Language Configuration

The following commands allow the user to manage the supported languages:

| Command | Description |
| --- | --- |
| monorepo config language list | List the currently supported languages |
| monorepo config language show <name> | Show the language configuration for the named language |
| monorepo config language create <name> [Options] | Create a language directory structure and configuration |
|  | file. |
| monorepo config language delete <name> | Delete a given language configuration file (source code |
|  | will remain). |
| monorepo config language enable <name> | enable the language |
| monorepo config language disable <name> | disable the language |

#### Create Language Options

| Option | Description |
| --- | --- |
| --directory <relative path> | The child directory/path where the language source code will exist. |
| --enabled | Enable the language config (disabled by default) |
| --platform <name> | Specify the platform supported by the given language. |

### Platform Configuration

The following commands allow the user to manage the supported platforms (opsys, arch):

| Command | Description |
| --- | --- |
| monorepo config platform list | List the currently supported platforms |
| monorepo config platform create <name> [Options] | Create a new supported platform configuration file. |
| monorepo config platform show <name> | Show the named platform configuration |
| monorepo config platform delete <name> | Delete the named platform |
| monorepo config platform enable <name> | Enable the given platform |
| monorepo config platform disable <name> | Disable the given platform |

A “platform” in the context of this command is a combination of operating system, CPU architecture and optional software. This combination creates an environment in which tasks can be supported to run without problems.

#### Create Platform Options

| Option | Description |
| --- | --- |
| --description <string> | A simple description of the platform. |
| --connect local | ssh | Connection method (local execution or key-based SSH) |
| --enabled | Enable the configuration |
| --opsys\_family <regex> | A regex for a single supported operating system families |
| --opsys\_version <regex> | A regex for a single supported operating system version. |
| --cpu\_arch <regex> | A regex for a single supported CPU Architecture |

### Project Configuration

The following commands allow the user to configure/manage the monorepo itself. (The changes made by this command can be reviewed and committed to git separately.)

| Command | Description |
| --- | --- |
| monorepo config project list [--language <lang>] | Print a list of all projects in the monorepo.  The “—language” flag can be used to filter the list to |
| monorepo config project show <name> | Show the RESOLVED configuration for a given project, where the resolved configuration is the final configuration state after starting with /monorepo.yaml, then applying the language-specific monorepo.yaml and finally after applying the project-specific configuration. |
| monorepo config project delete <name> | Delete a given project from the monorepo |
| monorepo config project create <name> [options] | Create a new project in the monorepo using the given name and options. Any options not specified will result in the defaults being used. |
| monorepo config project enable <feature> <name> | Enable the given project and feature |
| monorepo config project disable <feature> <name> | Disable the given project and feature |

#### Project Options

| Option | Description |
| --- | --- |
| --description <string> | Project description |
| --author <string> | Project author (used in copyright strings) |
| --output <executable | source> | Project output (executable or reusable source code |
| --enable <feature> | Enable a feature flag (linter, build, tests) |
| --addPlatform <opsys string> | Add a supported platform |

## Project-Action Commands

The following commands execute actions on one or more target build hosts. These commands follow this general syntax:

Monorepo <action command> <all | project ${name}> [parameters]

Command actions include actions to—

* Build project artifacts
* Lint project source code
* Run project tests

### Project Build Command

The following commands will facilitate building projects for the monorepo:

| Command | Description |
| --- | --- |
| monorepo build all | Build all projects for all languages |
| monorepo build project <name> | Build the named project |

### Project Linter Command

The following commands will facilitate linting projects for the monorepo:

| Command | Description |
| --- | --- |
| monorepo lint all | Lint all projects for all languages |
| monorepo lint project <name> | Lint the named project |

### Project Test Runner Command

The following commands will facilitate linting projects for the monorepo:

| Command | Description |
| --- | --- |
| monorepo test all | Test all projects for all languages |
| monorepo test project <name> | Test the named project |

### General Task Shell Command

The following will execute a command string and abstract the user from platform-specific issues:

| Command | Description |
| --- | --- |
| monorepo exec <string> [options] | Run the given string with its options. |

#### Task Shell Options

| Option | Description |
| --- | --- |
| --working-dir <string> | Specify the working directory where the command will run. |
| --using <shell> | Specify the command shell (bash, zsh, PowerShell) to run the command string. |