

9.2 Command-Line Interface

[Go to 9. Crucible Interfaces](#)

The [Command-Line Interface \(CLI\)](#) provides a text-based interface for invoking supported [Crucible](#) operations.

The CLI exposes the Shared Interface Operations described in [9.1 Shared Interface Operations](#) through commands, options, arguments, and referenced input files.

The CLI does not establish separate lifecycle semantics or independent Crucible capabilities. Each CLI command invokes an underlying operation governed by its applicable requirements.

Command Invocation

An authorized user invokes a supported Crucible operation by entering the applicable CLI command.

A command can identify:

- The requested operation
- The subject of the operation
- Required inputs
- Applicable options
- Applicable parameters
- Referenced configuration or input files
- The requested output form or location

The applicable command definition determines which arguments and options the operation accepts.

Command Inputs

The CLI accepts the information required by the underlying Crucible operation.

Command inputs can include:

- Positional arguments
- Named options
- Parameter values
- Identifiers
- File paths
- Configuration files
- References to controlled artifacts
- References to Baselines, Images, Deployment Targets, or assessment subjects

The CLI presentation of an input does not change the meaning of the value supplied to the underlying operation.

Input Validation

The CLI validates command syntax and required inputs before invoking the applicable operation.

Input validation can identify:

- An unknown command
- A missing required argument
- An unsupported option
- An invalid parameter value
- An unreadable referenced file
- An unresolved identifier
- An incompatible combination of inputs

Successful CLI input validation establishes that the request can be submitted to the underlying Crucible operation. It does not establish that the operation is authorized or that the operation will complete successfully.

Authorization

CLI operations remain subject to the same identity, access, Security Domain, and information-handling controls that govern the underlying Crucible capability.

The CLI does not permit a user to:

- Invoke an unauthorized operation
- Access an unauthorized resource
- Bypass a Security Domain restriction
- Override an information-handling rule
- Avoid an applicable validation or approval activity

Access to the CLI does not imply access to every Crucible operation.

Command Status

The CLI presents the status of the invoked operation through text output or another interface-appropriate representation.

The status can indicate whether the operation:

- Was accepted
- Is in progress
- Completed successfully
- Completed with an incomplete result
- Failed
- Could not proceed

- Requires corrected or additional input

The CLI preserves the distinction between a successfully parsed command and a successfully completed Crucible operation.

Command Results

The CLI provides access to the result produced by the underlying Crucible operation.

A result can include:

- A completion status
- An assigned identifier
- A produced artifact
- A report
- An assessment or validation result
- A deployment result
- A reference to stored output
- Error or failure information

The CLI can present a result directly as terminal output or identify the location of a separately produced artifact.

The CLI does not change the meaning of the result produced by the underlying operation.

Error Information

When a command or operation cannot complete, the CLI presents information sufficient to identify the observed condition.

Error information can identify:

- The command that failed
- The invalid or missing input
- The operation that could not proceed
- The observed failure
- The affected subject or resource
- Available corrective information

The CLI does not report a failed or incomplete operation as successful.

Relationship to Automation

The CLI provides a command-based means of invoking Crucible operations.

FR-UI-002 establishes the Command-Line Interface. It does not, by itself, establish requirements for:

- Unattended execution
- Shell scripting
- Pipeline integration
- Scheduled execution
- Machine-to-machine invocation
- Stable exit-code conventions
- A machine-readable output format

Those capabilities require separate approved requirements.

CLI Operation Result

A CLI invocation preserves the relationship among:

- The command entered
- The requested Crucible operation
- The supplied arguments and options
- The authorization result
- The operation status
- The resulting output
- Any error or incomplete condition

The underlying Crucible capability determines the operational result. The CLI provides the text-based means of requesting and presenting that result.

Requirements Addressed

Requirement	Statement
FR-UI-002 — Command-Line Interface	Crucible SHALL provide a Command-Line Interface (CLI) for invoking Crucible operations.

The linked leaf requirement page remains the canonical source.

© 2026 Dido Solutions, Inc. and Jackrabbit Consulting, Inc.

From:
<https://wiki.didosolutions.com/> - **Dido Solutions, Inc Wiki**

Permanent link:
<https://wiki.didosolutions.com/dido/02-crusible/09-crucible-interfaces/09-02-command-line-interface>

Last update: **2026/08/01 07:42**

