

Executable Instruction

[Go up to Terms and Definitions](#)

Discussion

An Executable Instruction specifies an operation in a form that an [Execution Facility](#) can interpret or perform.

An Executable Instruction can implement:

- An action
- A calculation
- A [Validation](#)
- A communication
- A state transition
- A measurement
- An observation
- A decision
- A control

An Executable Instruction can occur within:

- Source code
- Compiled code
- A script
- A command
- Declarative automation
- A tool-specific procedure
- A test-framework procedure
- An executable model
- A service request
- A hardware-control procedure
- Another form that an Execution Facility can interpret or perform

An Executable Instruction can identify or reference:

- Its identity
- Its name
- Its description
- Its [Version](#)
- Its instruction type
- Its language or format
- Its operation
- Its applicable [Test Arguments](#)
- Its applicable [Test Argument Values](#)
- Its required inputs
- Its produced outputs

- Its required dependencies
- Its required [Execution Facilities](#)
- Its applicable [Test Items](#)
- Its applicable [Test Environments](#)
- Its preconditions
- Its completion condition
- Its failure condition
- Its timeout
- Its applicable [Governance Policies](#)
- Its [Provenance](#)
- Its [Traceability](#)

A [Test Executable](#) contains or references one or more Executable Instructions that implement a test action or [Validation](#).

A Test Executable can contain the Executable Instructions directly or reference an independently maintained [Executable Artifact](#) that contains them.

An [Execution Facility](#) interprets or performs the Executable Instructions during [Test Execution](#).

Definition

instruction expressed in a form that an [Execution Facility](#) can interpret or perform

Source

Adapted from:

- [OMG Test Information Interchange Format \(TestIF\), Version 1.0 Beta 3, Section 7.3.7, TestExecutable](#)
- [Test Executable](#)
- [Test Step](#)
- [Test Execution](#)
- DIDO Reference Implementation Conceptual Model
- DIDO-TE draft Requirements Register

OMG TestIF describes a TestExecutable as a low-level procedural definition that supports automated testing through code, scripts, or other executable information.

OMG TestIF does not define Executable Instruction as a separate TestIF model element. This definition introduces Executable Instruction to distinguish an instruction from:

- The [Test Step](#) that specifies the required test action or Validation
- The [Test Executable](#) that contains or references the instruction
- The [Executable Artifact](#) that stores or packages the instruction
- The [Execution Facility](#) that interprets or performs the instruction

Note

Every Executable Instruction expresses an operation in a form that an identified Execution Facility can interpret or perform.

An Executable Instruction differs from a [Test Step](#):

- A Test Step specifies a test action or Validation independently of a particular execution mechanism
- An Executable Instruction provides part or all of the executable realization of that action or Validation

An Executable Instruction differs from a [Test Executable](#):

- An Executable Instruction specifies an operation
- A Test Executable contains, organizes, or references one or more Executable Instructions

An Executable Instruction differs from an [Executable Artifact](#):

- An Executable Instruction specifies an operation
- An Executable Artifact stores, packages, or provides one or more Executable Instructions

An Executable Instruction differs from an [Execution Facility](#):

- An Executable Instruction specifies the operation
- An Execution Facility interprets or performs the operation

An Executable Instruction does not require direct execution by a processor. An Execution Facility can include:

- An interpreter
- A command processor
- An automation engine
- A test framework
- A model-execution engine
- A service
- A device controller
- Another facility capable of interpreting or performing the instruction

A natural-language direction intended only for human interpretation is not an Executable Instruction under this definition. A direction qualifies as an Executable Instruction when an identified Execution Facility can interpret or perform it according to defined execution rules.

An Executable Instruction is not necessarily:

- Source code
- Compiled code
- Independently deployable
- Independently reusable
- Specific to testing
- Contained within a Test Executable
- Stored in a particular file

- Associated with a particular programming language

A change to an Executable Instruction's operation, inputs, outputs, parameters, control flow, dependencies, expected behavior, or execution requirements can create a new [Version](#) under the applicable versioning policy.

A reproducible [Test Run](#) should identify the exact Versions of the Executable Instructions and referenced Executable Artifacts used during [Test Execution](#).

Example

A [Test Step](#) requires a [Node](#) to submit a proposed transaction through a specified [Endpoint](#).

A [Test Executable](#) references a versioned script containing Executable Instructions that:

- Read the applicable [Test Argument Values](#)
- Establish a connection to the Endpoint
- Submit the proposed transaction
- Capture the response
- Record the transaction identifier
- Return the submission status
- Preserve the resulting [Evidence](#)

During Test Execution, the applicable [Execution Facility](#) interprets or performs the Executable Instructions.

The [Test Run](#) records:

- The Test Executable identity
- The Test Executable Version
- The referenced Executable Artifact
- The Executable Artifact Version
- The Executable Artifact digest
- The applicable Test Argument Values
- The applicable [Test Environment](#)
- The resulting [Test Result](#)
- The supporting Evidence

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

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

Permanent link:
https://wiki.didosolutions.com/dido/99_annexes/annex-b-terms-and-definitions/e/execution_facility

Last update: **2026/08/04 13:20**

