

Execution Path

[Go up to Terms and Definitions](#)

Discussion

An Execution Path is the actual progression through [Sequence Steps](#) during [Test Execution](#).

A [Test Sequence](#) defines the permitted progression through its Sequence Steps. An Execution Path identifies the progression that actually occurs during a particular [Test Run](#).

An Execution Path can identify:

- Its initial Sequence Steps
- Each performed Sequence Step
- The order in which Sequence Steps occurred
- Concurrent Sequence Steps
- Repeated Sequence Steps
- Skipped Sequence Steps
- Conditional branches taken
- Conditional branches not taken
- Error-handling branches taken
- The [Test Argument Values](#) supplied to each occurrence
- The [Starting Conditions](#) applicable to each occurrence
- The [Timing Conditions](#) applicable to each occurrence
- The start and completion time of each occurrence
- The event or condition that caused each progression
- The applicable [Test Result](#) for each occurrence
- The applicable [Verdict](#) for each occurrence
- The point and cause of termination
- Its [Provenance](#)
- Its [Traceability](#)

An Execution Path can contain:

- Sequential progression
- Parallel progression
- Conditional progression
- Iterative progression
- Event-driven progression
- Error-handling progression
- A combination of progression types

Definition

actual progression through [Sequence Steps](#) during [Test Execution](#)

Source

Adapted from:

- [OMG Test Information Interchange Format \(TestIF\), Version 1.0 Beta 3, Sections 1.3, 7.3.4, and 7.3.14](#)
- [Test Sequence](#)
- [Sequence Step](#)
- [Test Execution](#)
- [Test Run](#)
- DIDO Reference Implementation Conceptual Model
- DIDO-TE draft Requirements Register

OMG TestIF defines TestSequence as a directed progression of SequenceSteps and permits progression to depend on the applicable Sequence Step relationships and conditions.

OMG TestIF discusses test paths but does not establish Execution Path as a separate TestIF model element. This definition introduces Execution Path to identify the Test Sequence progression that actually occurred during a particular Test Run.

Note

Use the singular glossary term Execution Path. Link plural usage to the same controlling page:

```
[[dido:99_annexes:annex-b-terms-and-definitions:e:execution_path|Execution Paths]]
```

An Execution Path differs from a [Test Sequence](#):

- A Test Sequence defines permitted progression
- An Execution Path identifies the progression that actually occurred

A Test Sequence can permit multiple Execution Paths.

An Execution Path differs from a [Sequence Step](#):

- A Sequence Step identifies one position and occurrence within a Test Sequence
- An Execution Path identifies the actual progression through the performed Sequence Steps

An Execution Path differs from a [Test Run](#):

- An Execution Path identifies the actual progression
- A Test Run records the Execution Path, execution context, Test Results, Evidence, and related information for a particular occurrence of Test Execution

An Execution Path differs from a Test Result:

- An Execution Path identifies what progression occurred

- A Test Result records an outcome associated with a Sequenced Test Object or Sequence Step

A repeated Test Run does not necessarily follow the same Execution Path. Differences can result from:

- Different Test Argument Values
- Different Starting Conditions
- Different Test Results
- Different Verdicts
- Different Timing Conditions
- Timeouts
- Errors
- Concurrent behavior
- External events
- Nondeterministic behavior

The applicable Acceptance Criteria determine whether reproducible Test Execution requires:

- The same Execution Path
- An equivalent Execution Path
- Any permitted Execution Path
- A particular set of required Sequence Steps
- Exclusion of prohibited Execution Paths

Parallel progression does not always establish a single total ordering. An Execution Path can preserve a partial ordering that identifies concurrency and the required temporal relationships among Sequence Steps.

An Execution Path should preserve enough information to reconstruct:

- Which Sequence Steps occurred
- Which Sequence Steps did not occur
- The order or partial order of occurrence
- The Test Argument Values used
- The conditions that controlled progression
- The Test Results produced
- The reason Test Execution completed or terminated

Example

A Test Sequence permits the following progression:

- Sequence Step SS-01 establishes a Node Set
- Sequence Step SS-02 submits a transaction
- Sequence Steps SS-03A, SS-03B, and SS-03C observe three Nodes concurrently
- Sequence Step SS-04 evaluates the Test Results when all three Nodes respond
- Sequence Step SS-05 handles a timeout when one or more Nodes fail to respond

During one Test Run, all three Nodes respond. The Execution Path is:

- SS-01

- SS-02
- Concurrent occurrence of SS-03A, SS-03B, and SS-03C
- SS-04

During a repeated Test Run, Node C fails to respond. The Execution Path is:

- SS-01
- SS-02
- Concurrent occurrence of SS-03A, SS-03B, and SS-03C
- SS-05

The two Test Runs use the same Test Sequence but follow different Execution Paths. The Acceptance Criteria determine whether both Execution Paths satisfy the applicable Reproducibility requirement.

© 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_path

Last update: **2026/08/05 02:04**

