

5.1 Information Model Overview

[Go to 5. DIDO-TE Information Model](#)

The DIDO-TE Information Model provides the structural view of the information used to define, configure, execute, observe, and evaluate tests within the [DIDO-TE](#).

The model is derived initially from the Testing Package of the DIDO Reference Data Model and is refined where necessary to support the DIDO-TE architecture and requirements.

Model Scope

The Information Model includes the information required to describe:

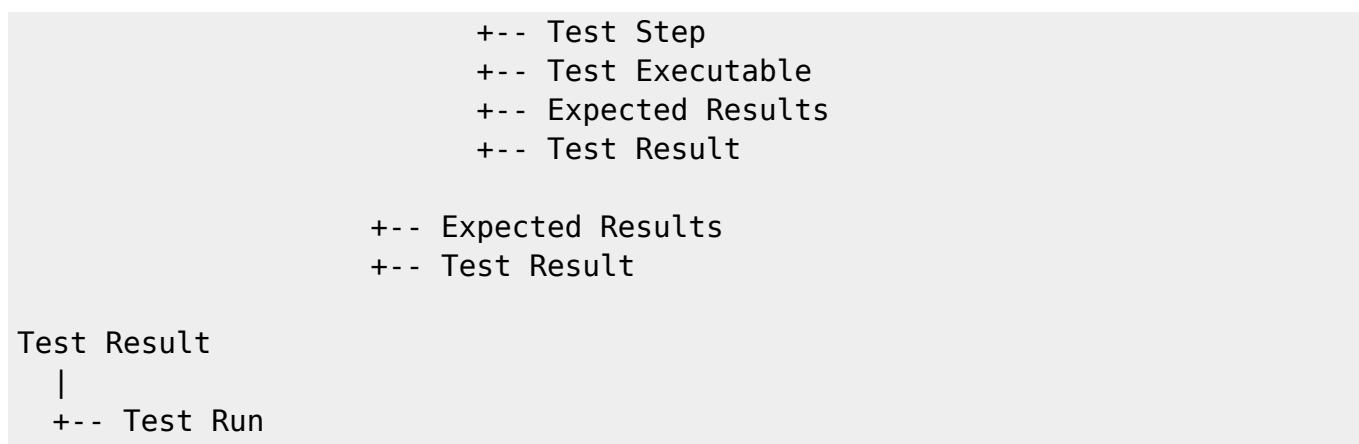
- the organization of a test;
- the [Test Sets](#), [Test Cases](#), and [Test Steps](#) that compose the test structure;
- the [Test Executables](#) and [Test Arguments](#) used to perform test activities;
- the [Expected Results](#) used to describe expected test behavior;
- the [Test Environment](#) in which testing occurs;
- the [Nodes](#), [Twin Nodes](#), and other [Resources](#) participating in testing;
- the [Configuration](#), communication, state, and time conditions governing testing;
- the [Test Results](#) produced by testing;
- the [Test Runs](#) associated with those Test Results; and
- the information required to support [Validation](#), [Evidence](#), and [Traceability](#).

Existing Testing Structure

The Testing Package of the DIDO Reference Data Model provides the initial structural foundation for the DIDO-TE Information Model.

At a high level, the existing model organizes testing using the following hierarchy:

```
Test
|
+-- Test Plan
    |
    +-- Test Set
        |
        +-- Test Set
            |
            +-- Test Case
                |
                +-- Test Executable
                    |
                    +-- Test Step
                        |
```



The existing model also associates a Test Set with the environment in which that Test Set executes.

The hierarchy shown above represents the source-model structure. DIDO-TE may refine individual relationships as the model elements are reviewed in the following sections.

Model Element Review

Each principal model element is reviewed independently before the integrated DIDO-TE Information Model is established.

The review proceeds from the existing DIDO Reference Data Model rather than beginning with a newly imposed structure.

For each model element, the review identifies:

- the element represented by the source model;
- the attributes represented by the source model;
- associations with other model elements;
- association multiplicities;
- semantic or structural issues discovered in the source model;
- DIDO-TE architectural interpretation;
- relationships to existing DIDO-TE glossary concepts;
- relationships to DIDO-TE requirements; and
- any deliberate DIDO-TE extension or refinement.

This approach allows the DIDO-TE Information Model to preserve useful source-model structures while distinguishing them from additions made specifically for DIDO-TE.

Source Model and DIDO-TE Extensions

The Information Model distinguishes between **source-model content** and **DIDO-TE extensions**.

Source-model content consists of information elements, attributes, associations, and multiplicities represented by the DIDO Reference Data Model.

DIDO-TE extensions consist of information or relationships required by the DIDO-TE architecture that are not represented explicitly in the source model.

Examples of concepts introduced or substantially expanded by DIDO-TE include:

- [Test Definition](#);
- [Test Environment](#) configuration and controlled environmental conditions;
- [Twin Node](#) participation;
- [Virtual Network](#) and communication conditions;
- controlled test state;
- controlled test time;
- Validation Criteria;
- Evidence requirements; and
- governed Traceability relationships.

The inclusion of a DIDO-TE extension does not imply that the corresponding concept existed in the original DIDO Reference Data Model.

Test Definition as an Integrating Concept

The [Test Definition](#) is an important integrating concept within the DIDO-TE Information Model.

The existing DIDO Reference Data Model does not contain a model element named Test Definition. Instead, information relevant to defining a test is distributed across multiple testing elements and their relationships.

DIDO-TE therefore derives the Test Definition structure from:

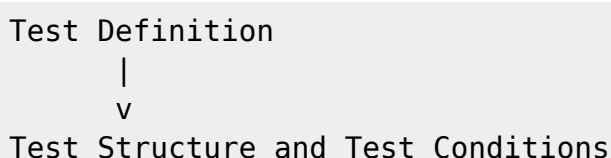
- the testing structure represented by the existing model;
- the conditions required by the DIDO-TE functional requirements;
- the information required by [Test Execution](#);
- the information required to interpret Test Results; and
- the criteria and Evidence required for Validation and Traceability.

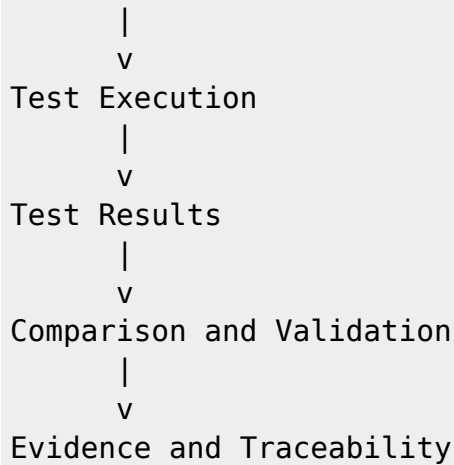
The Test Definition is therefore developed after the constituent model elements have been reviewed.

This ordering avoids imposing a Test Definition structure that conflicts with or unnecessarily duplicates the existing testing model.

Information Model Relationships

The Information Model supports a general architectural flow in which governed test information is defined before execution and is preserved through evaluation:





This flow represents an architectural relationship among information elements. It does not prescribe a sequential software implementation, workflow engine, storage mechanism, or execution platform.

Relationship to Requirements

The Information Model provides architectural structure for concepts referenced by the DIDO-TE [requirements](#).

The requirements remain the normative source of DIDO-TE obligations.

For example:

- Test Environment requirements identify information that must be represented or referenced by the test definition and environment model;
- Twin Node requirements identify information concerning Twin Identities, Twin Realizations, and Twin Relationships;
- Virtual Network and Communication requirements identify governed communication conditions;
- State and Time Control requirements identify governed initial state and temporal conditions; and
- [C.3.6 Test Definition Requirements](#) collects and formalizes the information that the governing Test Definition is required to identify, specify, or reference.

As the model is refined, individual model elements should maintain explicit Traceability to the requirements that establish the need for those elements or relationships.

Contents

The following sections review the principal information-model elements:

- [5.2 Test](#)
- [5.3 Test Plan](#)
- [5.4 Test Set](#)
- [5.5 Test Case](#)
- [5.6 Test Step](#)

- [5.7 Test Executable](#)
- [5.8 Expected Results](#)
- [5.9 Test Result](#)
- [5.10 Test Run](#)
- [5.11 Test Tag](#)
- [5.12 Test Environment Relationships](#)
- [5.13 Test Definition](#)
- [5.14 Integrated DIDO-TE Information Model](#)

Source

The initial structural content of this section is derived from the Testing Package of the DIDO Reference Data Model.

DIDO-TE-specific refinements are derived from the DIDO-TE architecture and requirements.

Notes for Editors

- Preserve the distinction between source-model content and DIDO-TE extensions.
- Do not silently correct source-model anomalies.
- Review each source-model element independently before establishing the integrated model.
- Use focused [UML](#) class diagrams for the individual model-element sections.
- Preserve source-model multiplicities unless a deliberate DIDO-TE refinement is documented.
- Do not interpret UML classes as prescribed software implementation classes.
- Maintain explicit Traceability between model refinements and the DIDO-TE requirements that require those refinements.
- Develop the Test Definition after reviewing its constituent information elements.

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

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

Permanent link:
<https://wiki.didosolutions.com/dido/03-dido-te/06-information-model/06-01-overview/start?rev=1787530520>

Last update: **2026/08/23 17:15**

