

Annex C: Requirements

[Go to Annexes](#)

Annex C contains the canonical requirements register for the Distributed Immutable Data Object Test Environment (DIDO-TE).

The requirements are organized by requirement class. Each normative requirement receives a stable identifier and resides on a separate Wiki page. Architecture pages, implementation artifacts, tests, Evidence, verification records, and other records can reference each requirement through its stable identifier and URI.

The draft Requirements Register provides source requirements and supporting information for this annex. The identifiers REQ-0001 through REQ-0091 identify requirements in the source register. Annex C preserves those identifiers for traceability but assigns new canonical identifiers to the DIDO-TE requirements.

DIDO-TE requirements conform to the:

- Common Specification Weakness Enumeration (CWE)
- OMG Specification Discipline and Authoring (SDA)
- DIDO Solutions shared [Terms and Definitions](#)

Annex C distinguishes among:

- The canonical DIDO-TE requirement
- The source requirement
- The requirement class
- The governance status of the requirement
- The delivery phase associated with the requirement
- The implementation status of DIDO-TE relative to the requirement
- The verification activities used to determine whether DIDO-TE satisfies the requirement
- The Evidence required to support the verification result
- The architecture, ConOps, and operational concepts that support the requirement

Requirement Categories

- [C.1 Mission Objectives](#)
- [C.2 Operational Requirements](#)
- [C.3 Functional Requirements](#)
 - [C.3.1 Test Environment Configuration](#)
 - [C.3.2 Nodes and Deployment Targets](#)
 - [C.3.3 Twin Nodes](#)
 - [C.3.4 Virtual Networks and Communication](#)
 - [C.3.5 State and Time Control](#)
 - [C.3.6 Test Definition](#)
 - [C.3.7 Test Execution](#)
 - [C.3.8 Record and Playback](#)
 - [C.3.9 Baseline and Comparison](#)

- [C.3.10 Validation](#)
 - [C.3.11 Monitoring and Evidence](#)
 - [C.3.12 Catalogs and Reuse](#)
 - [C.3.13 Governance and Communities of Interest](#)
 - [C.3.14 DIDO-TE Operations](#)
4. [C.4 Security Requirements](#)
 5. [C.5 Logging and Auditing Requirements](#)
 6. [C.6 Reliability Requirements](#)
 7. [C.7 Performance Requirements](#)
 8. [C.8 Maintainability Requirements](#)
 9. [C.9 Data Management Requirements](#)
 10. [C.10 Interoperability Requirements](#)
 11. [C.11 Future Capability Requirements](#)
 12. [C.12 Key Value Proposition Requirements](#)

Requirement Identifiers

DIDO-TE uses the following requirement-class identifiers:

Requirement Class	Prefix
Mission Objective	MO
Operational Requirement	OR
Functional Requirement	FR
Security Requirement	SEC
Logging and Auditing Requirement	LOG
Reliability Requirement	REL
Performance Requirement	PER
Maintainability Requirement	MNT
Data Management Requirement	DAT
Interoperability Requirement	INT
Future Capability Requirement	FUT
Key Value Proposition Requirement	KVP

Each requirement identifier consists of the requirement-class prefix followed by a three-digit number.

Examples include:

- MO-001
- OR-001
- FR-001
- SEC-001
- PER-001

A requirement identifier remains stable when the requirement moves to another section or undergoes an editorial revision that does not substantively change the requirement.

A retired, withdrawn, deprecated, or superseded identifier is not reassigned to another

requirement.

Requirement Decomposition

Each normative requirement expresses one independently interpretable and independently verifiable obligation.

When a source requirement contains multiple normative obligations, Annex C preserves the source requirement on a non-leaf parent page and assigns a lowercase suffix to each derived requirement.

For example:

- FR-040 identifies the source requirement represented by the parent page
- FR-040a identifies the first derived requirement
- FR-040b identifies the second derived requirement

The non-leaf parent page retains a trailing `:start` in its namespace. Each derived leaf requirement omits the trailing `:start`.

A derived requirement includes a **Derived From** section that identifies:

- The parent requirement
- The Original Requirement
- The obligation preserved by the derived requirement
- Other requirements derived from the same parent requirement

A requirement that does not require decomposition uses a **Source** section rather than a **Derived From** section.

Requirement Page Structure

Each leaf requirement page uses the following sections where applicable:

- Statement
- Source
- Derived From
- Rationale
- Applies To
- Verification
- Referenced By
- Delivery Phase
- Implementation Status
- Requirement Status
- Issues
- Notes for Editors

The **Source** and **Derived From** sections are mutually exclusive.

A requirement page uses **Source** when the requirement does not result from the decomposition of another requirement.

A requirement page uses **Derived From** only when the requirement results from the decomposition of a parent requirement.

Normative Language

Requirement Statements use the following normative terms:

- **SHALL** identifies a mandatory requirement
- **SHALL NOT** identifies a mandatory prohibition
- **MAY** identifies a permitted capability or action
- **SHOULD** identifies an advisory recommendation

The Statement section contains only the normative requirement.

Rationale, examples, implementation information, verification information, source information, and editorial guidance remain outside the Statement section.

Requirement Quality

Each DIDO-TE requirement conforms to the applicable CWE and SDA rules.

Each requirement:

- Has a unique and stable identifier
- Expresses a single, independently interpretable obligation
- Identifies the responsible actor
- Identifies the required behavior, outcome, permission, or prohibition
- Uses defined terminology consistently
- Uses explicit normative language
- Avoids vague, weak, open-ended, and logically ambiguous expressions
- Avoids implementation-specific mechanisms unless the mechanism constitutes an approved constraint
- Supports objective verification
- Preserves traceability to the source material
- Preserves traceability to architecture, implementation, test, Evidence, and conformance artifacts

Requirement Status

Requirement Status identifies the governance state of the requirement.

Possible states include:

- Draft
- Proposed
- Approved
- Deprecated
- Superseded
- Withdrawn

Implementation Status

Implementation Status identifies the state of DIDO-TE relative to the requirement.

Possible states include:

- Implemented
- Partially Implemented
- Demonstrated
- Planned
- Deferred
- Strategic Unscheduled
- Unsupported
- Not Assessed

Requirement Status and Implementation Status remain separate.

An approved requirement can remain planned, partially implemented, deferred, unsupported, or not assessed.

Contents

- [C.1 Mission Objectives](#)
 - [C.2 Operational Requirements](#)
 - [C.3 Functional Requirements](#)
 - [C.11 Conformance Requirements](#)
-

Notes for Editors

Annex C is the authoritative source for DIDO-TE normative requirements.

The draft Requirements Register remains source material. Its REQ-0001 through REQ-0091 identifiers remain source-reference identifiers and do not become canonical DIDO-TE requirement identifiers.

Do not change an assigned requirement identifier because the requirement moves to another section.

Do not reuse an identifier assigned to a retired, withdrawn, deprecated, or superseded requirement.

Apply SDA pattern detection before performing CWE evaluation.

Preserve traceability from each DIDO-TE requirement to:

- The source requirement
- Supporting source material
- Applicable ConOps scenarios
- Applicable architecture sections
- Implementation artifacts
- Verification activities
- Verification results
- Evidence
- Identified issues

Use a non-leaf parent page only when requirement decomposition requires one.

Leaf requirement pages omit a trailing :start from their namespaces.

Non-leaf requirement pages retain a trailing :start in their namespaces.

Do not rename a requirement page after an external citation unless a redirect or move plan is in place.

© 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/99-annexes/annex-c-requirements/start?rev=1785835832>

Last update: **2026/08/04 02:30**

