

TWIN-005 — Access Twin Realization State

[Go to C.3.3 Twin Node Requirements](#)

TWIN-005 is a non-leaf requirement group governing establishment, determination, access, and contextualization of the state of a [Twin Realization](#) within a [Twin Relationship](#).

The state of a Twin Realization describes the condition of the characteristics represented by that Twin Realization at a specified time. That state provides a controlled basis for synchronization, simulation, comparison, [Validation](#), and [Test Execution](#).

Access to the state of a Twin Realization includes establishment of an initial state, determination of its current state, access to state values, identification of the originating [Data Source](#), association of state with time, preservation of provenance, and detection of an invalid or unknown state.

The communication mechanism used to obtain state information does not, by itself, establish the semantic meaning, validity, source eligibility, authority, or temporal context of that state.

Conformance is assessed against the individual leaf requirements listed below. TWIN-005 does not establish a separate conformance obligation.

Requirements

Requirement ID	Requirement Title	Purpose
TWIN-005a	Establish the Initial State of a Twin Realization	Establish a controlled starting state for a Twin Realization before its use in Test Execution.
TWIN-005b	Determine the Current State of a Twin Realization	Determine the current condition of the represented characteristics of a Twin Realization.
TWIN-005c	Access Twin Realization State Values	Provide access to the values that constitute the determined state of a Twin Realization.
TWIN-005d	Identify the State Data Source	Identify the Data Source from which each accessed state value originated.
TWIN-005e	Associate Twin Realization State with Time	Associate the state of a Twin Realization with the temporal information required to interpret that state.
TWIN-005f	Preserve Twin Realization State Provenance	Preserve provenance associated with the state of a Twin Realization and its constituent values.
TWIN-005g	Detect an Invalid or Unknown Twin Realization State	Detect when the state of a Twin Realization cannot be established as valid and known.

Contents

- [TWIN-005a — Establish the Initial State of a Twin Realization](#)
- [TWIN-005b — Determine the Current State of a Twin Realization](#)
- [TWIN-005c — Access Twin Realization State Values](#)
- [TWIN-005d — Identify the State Data Source](#)
- [TWIN-005e — Associate Twin Realization State with Time](#)
- [TWIN-005f — Preserve Twin Realization State Provenance](#)

- [TWIN-005g — Detect an Invalid or Unknown Twin Realization State](#)

Source

- [\[DTE1\] U.S. Patent Application US20220237111A1](#), virtual and physical Twin Node concepts, baseline test results, modification of Twin Nodes, comparison of results, and validation
- [\[DTE2\] Non-Traditional BAA Submission](#), Digital Twin concepts, real-time monitoring, scenario testing, and Twin Nodes Selection
- [\[DTE9\] Distributed Immutable Data Object Reference Architecture \(DIDO-RA\)](#), Digital Twin and distributed data concepts

Rationale

A [Twin Relationship](#) represents selected characteristics of a logical twin through one or more [Twin Realizations](#). Testing requires more than access to isolated values. The DIDO-TE requires a controlled understanding of each participating Twin Realization's state.

The [Twin Data Mapping](#) establishes the semantic correspondence between represented characteristics and their data representations. The state of a Twin Realization describes the condition represented by those characteristics at a specified time.

Establishing the initial state of a Twin Realization provides a controlled starting point for Test Execution. Determining its current state provides the basis for observing subsequent changes and comparing actual state with expected or previously established state.

Access to state remains distinct from Data Source eligibility and [Source Selection](#). A technically available value does not establish the state of a Twin Realization solely because the value is reachable or received through a communication mechanism.

Interpretation of state also depends on the origin, temporal context, and provenance of the values from which that state is determined. The architecture therefore preserves those relationships independently of the communication technology used to obtain the data.

An invalid or unknown state requires detection because synchronization, simulation, comparison, Validation, or Test Execution performed from an indeterminate state can compromise reproducibility and interpretation of results.

Applies To

- [DIDO-TE](#)
- [Twin Identity](#)
- [Twin Relationship](#)
- [Twin Realization](#)
- [Twin Data Mapping](#)
- [Data Source](#)
- [Source Selection](#)
- [Configuration](#)

- [Test Definition](#)
- [Test Execution](#)
- [Validation](#)

Traceability

- [TWIN-005a — Establish the Initial State of a Twin Realization](#)
- [TWIN-005b — Determine the Current State of a Twin Realization](#)
- [TWIN-005c — Access Twin Realization State Values](#)
- [TWIN-005d — Identify the State Data Source](#)
- [TWIN-005e — Associate Twin Realization State with Time](#)
- [TWIN-005f — Preserve Twin Realization State Provenance](#)
- [TWIN-005g — Detect an Invalid or Unknown Twin Realization State](#)

ConOps Relationship

This requirement group supports establishment and determination of the state of a Twin Realization before and during Test Execution.

The initial state establishes the controlled starting condition for a Twin Realization. Subsequent determination of its state supports observation of state changes, synchronization among Twin Realizations, simulation, comparison, Validation, monitoring, and evidence-producing activities.

The Data Source, temporal context, and provenance associated with the state of a Twin Realization support reproducibility and interpretation of Test Execution results.

Delivery Phase

TBD

Requirement Status

Draft

Notes for Editors

TWIN-005 is a non-leaf requirement group and does not contain an independent normative Statement or Verification section.

The explicit child links under Contents remain while the subordinate requirement pages are being created and reviewed. After all child pages exist, you may remove the explicit links because the

indexmenu automatically discovers them.

The displayed title is *Access Twin Realization State*. The existing namespace `twin-005-acquire-twin-state` remains unchanged to preserve the stable URI.

The architecture does not define *Twin State* as a separate controlled term. State is treated as a characteristic of a Twin Realization, consistent with how the Node requirements treat state.

The decomposition intentionally distinguishes:

- establishment of the initial state of a Twin Realization;
- determination of its current state;
- access to the values constituting that state;
- identification of the originating Data Source;
- temporal association;
- provenance; and
- detection of an invalid or unknown state.

Do not treat successful receipt of data as sufficient evidence that the data establishes the state of a Twin Realization. Twin Data Mapping, Data Source eligibility, Source Selection, temporal context, provenance, and governing Configuration remain separate architectural concerns.

© 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/03-functional-requirements/03-03-twin-node-requirements/twin-005-acquire-twin-state/start>

Last update: 2026/08/20 17:32

