

TWIN-005g — Detect an Invalid or Unknown Twin Realization State

[Go to TWIN-005 — Access Twin Realization State](#)

Statement

The [DIDO-TE](#) SHALL detect when the state of a [Twin Realization](#) is invalid or cannot be determined.

Source

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

Rationale

The state of a [Twin Realization](#) may become unusable for Test Execution when required state values are unavailable, contradictory, stale, malformed, outside defined constraints, or otherwise insufficient to determine the represented condition.

Failure to distinguish such a condition from a valid determined state can compromise synchronization, simulation, comparison, [Validation](#), and interpretation of Test Execution results.

This requirement therefore distinguishes a valid determined state from a condition in which the state is invalid or cannot be determined.

The criteria used to determine validity derive from the governing [Configuration](#), [Test Definition](#), [Twin Data Mapping](#), or other controlled requirement.

Applies To

- [DIDO-TE](#)
- [Twin Realization](#)
- [Twin Relationship](#)
- [Twin Data Mapping](#)
- [Data Source](#)
- [Configuration](#)
- [Test Definition](#)

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

Verification

Verification confirms that:

1. The DIDO-TE distinguishes a valid determined state from a state that is invalid or cannot be determined.
2. Missing required state values are detected when those values prevent determination of state.
3. Contradictory or malformed state values are detected when they violate the governing Configuration, Test Definition, or Twin Data Mapping.
4. State values outside defined temporal or validity constraints are detected when those constraints govern state determination.
5. An invalid or indeterminate state is not represented as a valid determined state.
6. The detected condition remains associated with the Twin Realization for which state determination failed.

Verification includes at least one case in which the state of a Twin Realization cannot be validly determined because required state information is missing, contradictory, stale, or invalid.

Traceability

Plugin Backlinks: Nothing was found.

ConOps Relationship

This requirement supports detection of conditions that prevent reliable use of the state of a Twin Realization during Test Execution, synchronization, simulation, comparison, Validation, monitoring, and evidence generation.

Delivery Phase

TBD

Requirement Status

Draft

Statement Reference

{{section>dido:03-dido-te:99-annexes:annex-c-requirements:03-functional-

```
requirements:03-03-twin-node-requirements:twin-005-acquire-twin-  
state:twin-005g-detect-an-invalid-or-unknown-twin-  
state#Statement&noheader&nofooter&noeditbtn}}
```

© 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/twin-005g-detect-an-invalid-or-unknown-twin-state>

Last update: 2026/08/20 17:32

