

NOD-002h — Validate the Provisioned Node

[Go to NOD-002 — Provision a Node](#)

Statement

The [DIDO-TE](#) SHALL validate each provisioned [Node](#) before using it in [Test Execution](#).

Derived From

- [NOD-002 — Provision a Node](#)
- [NOD-001h — Validate the Node Definition](#)
- [NOD-002a — Select the Node Definition](#)
- [NOD-002b — Select the Node Implementation](#)
- [NOD-002c — Resolve Node Dependencies](#)
- [NOD-002d — Allocate Node Resources](#)
- [NOD-002e — Instantiate the Node](#)
- [NOD-002f — Establish the Node Initial State](#)
- [NOD-002g — Record Node Provisioning](#)
- [\[DTE5\] DIDO-TE Requirements Register](#), source requirement identifier and obligation to be assigned
- [\[DTE6\] Structured Information Processing Reference Architecture \(SIP-RA\)](#)
- [\[DTE7\] Federated Data Interpretation Systems Reference Architecture \(FDIS-RA\)](#)

Rationale

Provisioning validation determines whether the resulting Node instance corresponds to its approved Node Definition, selected Node Implementation, resolved Dependencies, allocated Test Resources, required initial state, and applicable Test Environment constraints.

The validation assesses the actual provisioned instance rather than the suitability of its definition or implementation in isolation. It identifies incomplete, inconsistent, altered, incompatible, or unauthorised provisioning before the Node participates in Test Execution.

A recorded validation result establishes the criteria applied, checks performed, findings produced, exceptions accepted, and determination reached for the specific Node instance.

Applies To

- [DIDO-TE](#)
- [Node](#)
- Node Definition
- Node Implementation

- [Provisioning](#)
- [Dependency](#)
- [Test Resource](#)
- [Test Environment](#)
- [Test Definition](#)
- [Test Execution](#)
- [Evidence](#)

Verification

Verification confirms that:

- Each provisioning validation activity has a unique identity.
- The validation identifies the Node instance and its governing Node Definition.
- The validation identifies the applicable provisioning validation criteria.
- The validation criteria identify the required composition, identity, integrity, compatibility, resource, dependency, state, security, and readiness conditions.
- The DIDO-TE verifies the identity of the provisioned Node instance.
- The DIDO-TE verifies that the Node instance has the required Node role and Node type.
- The DIDO-TE verifies that the Node instance corresponds to the selected Node Definition and version.
- The DIDO-TE verifies that the Node instance uses the selected Node Implementation, version, build, and variant.
- The DIDO-TE verifies the identity and integrity of the implementation artifacts present in the Node instance.
- The DIDO-TE verifies that each mandatory Dependency has been resolved and remains available, accessible, compatible, and correctly associated with the Node.
- The DIDO-TE verifies that each mandatory Test Resource has been allocated and remains available, sufficient, compatible, and correctly associated with the Node.
- The DIDO-TE verifies that the Node satisfies applicable placement, isolation, security, jurisdictional, residency, and access-control constraints.
- The DIDO-TE verifies that each mandatory interface exists and satisfies its applicable identity, version, protocol, binding, and compatibility requirements.
- The DIDO-TE verifies that the Node supports its required lifecycle states and transitions.
- The DIDO-TE verifies that the Node occupies its specified initial state.
- The DIDO-TE verifies that the Node provides the observations required for subsequent configuration, execution, monitoring, and assessment.
- The DIDO-TE verifies that the Node remains distinguishable from other Node instances.
- The DIDO-TE verifies that the provisioning record remains complete, internally consistent, and traceable to the actual Node instance.
- The DIDO-TE compares the actual Node composition with the intended provisioning composition.
- The DIDO-TE identifies each missing, altered, unexpected, incompatible, unresolved, incorrectly configured, unauthorised, or obsolete component.
- The DIDO-TE identifies each residual artifact, process, allocation, connection, or state that affects Node validity.
- The DIDO-TE assigns a uniquely identified validation result to the assessed Node instance.
- The validation result identifies the criteria applied, checks performed, observations collected, findings produced, exceptions accepted, and resulting determination.

- The DIDO-TE rejects a provisioned Node that fails a mandatory validation criterion.
- The DIDO-TE prevents a rejected or unvalidated Node from participating in Test Execution.
- An approved exception identifies its authority, rationale, scope, duration, and effect on Test Execution and Test Results.
- A material change to the Node instance, Node Definition, Node Implementation, Dependencies, Test Resources, initial state, Test Environment, or validation criteria triggers revalidation.
- The DIDO-TE records the provisioning validation result as [Evidence](#).
- The DIDO-TE maintains [Traceability](#) among the Node Definition, Node Implementation, provisioned Node instance, Dependencies, Test Resources, Test Environment, provisioning record, validation criteria, validation activities, findings, exceptions, determination, Test Execution, and resulting Evidence.
- A missing, incomplete, inconsistent, incompatible, unauthorised, unvalidated, rejected, or untraceable provisioned Node constitutes nonconformance with this requirement.

Verification includes:

- Inspection of the provisioned Node identity and composition
- Inspection of the applicable provisioning validation criteria
- Comparison of the provisioned Node with its Node Definition and Node Implementation
- Verification of implementation artifact identities and integrity
- Verification of resolved Dependencies and allocated Test Resources
- Inspection and exercise of mandatory interfaces
- Confirmation of required lifecycle behaviour and initial state
- Confirmation of required observations
- Confirmation of placement, isolation, security, jurisdictional, residency, and access-control constraints
- Comparison of the actual Node composition with the provisioning record
- A negative assessment involving an altered implementation artifact
- A negative assessment involving a missing or incompatible Dependency
- A negative assessment involving an insufficient or incorrectly allocated Test Resource
- A negative assessment involving an incorrect initial state
- A negative assessment involving a missing mandatory interface or observation
- Confirmation that a rejected or unvalidated Node cannot participate in Test Execution
- Modification of a material provisioning element to confirm revalidation
- Inspection of validation findings, exceptions, determinations, Evidence, and Traceability

Referenced By

Plugin Backlinks: Nothing was found.

Related Architecture Sections

- Add links to the architecture sections governing Node Provisioning, provisioning validation, Node readiness, Test Execution preconditions, configuration control, Evidence, and Traceability.

Delivery Phase

Assign the applicable delivery phase.

Requirement Status

Draft

Statement Reference

Use the following syntax to reference this requirement's Statement section from another DokuWiki page:

```
{{section>dido:03-dido-te:99-annexes:annex-c-requirements:03-node-requirements:nod-002-provision-a-node:nod-002h-validate-the-provisioned-node#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/03-02-node-requirements/nod-002-provision-a-node/nod-002h-validate-the-provisioned-node>

Last update: 2026/08/17 14:20

