

NOD-001c — Define Node Interfaces

[Go to NOD-001 — Define a Node](#)

Statement

The [DIDO-TE](#) SHALL define each interface through which a [Node](#) interacts with another element of the applicable [Test Environment](#).

Derived From

- [NOD-001 — Define a Node](#)
- [\[DTE5\] 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

A Node participates in a Test Environment through defined interaction points. These interfaces carry data, commands, events, status information, observations, control signals, or resource access between the Node and other Nodes, Test Resources, operators, or external systems.

An interface definition identifies the interaction contract independently of a particular runtime connection. It establishes the information required to configure compatible endpoints, validate exchanged information, control access, observe interactions, and determine whether the Node behaves as specified.

Undefined or ambiguous interfaces prevent reliable integration and weaken the interpretation of Test Results because the expected direction, content, timing, protocol, constraints, and failure behaviour of an interaction remain unknown.

Applies To

- [Node](#)
- [Test Environment](#)
- [Test Definition](#)
- [Test Resource](#)
- [Configuration](#)
- [Constraint](#)
- [Evidence](#)

Verification

Verification confirms that:

- Each Node interface has a unique identity.
- The Node Definition identifies the purpose of each interface.
- The Node Definition identifies each interface as provided, required, or bidirectional.
- The Node Definition identifies the elements permitted or required to interact through each interface.
- The Node Definition identifies the data, command, event, status, control, observation, or resource interaction carried by each interface.
- The Node Definition identifies the applicable information model, data type, schema, message structure, operation signature, or topic definition.
- The Node Definition identifies the applicable protocol, transport, middleware, or invocation mechanism.
- The Node Definition identifies the direction, cardinality, ordering, timing, frequency, capacity, and delivery characteristics applicable to each interface.
- The Node Definition identifies initialisation, connection, disconnection, retry, timeout, error, and recovery behaviour.
- The Node Definition identifies the authentication, authorisation, confidentiality, integrity, and auditing controls applicable to each interface.
- The Node Definition identifies constraints governing values, sequences, rates, sizes, states, and permitted interactions.
- The Node Definition identifies the version or revision of each interface specification.
- The Node Definition identifies compatibility rules governing different interface versions.
- The Node Definition identifies any interface condition that depends on the Node's role, type, configuration, or lifecycle state.
- The Node Definition identifies the observations and records required to assess interface behaviour.
- The DIDO-TE verifies compatibility between connected Node interfaces before Test Execution.
- The DIDO-TE detects a missing, additional, incompatible, incorrectly configured, or unavailable mandatory interface.
- The DIDO-TE records the interface versions and endpoint bindings used during each applicable Test Execution.
- The DIDO-TE maintains [Traceability](#) among each interface, its specification, Node Definition, endpoint binding, configuration, Test Definition, Test Execution, and resulting Evidence.
- An undefined, ambiguous, incompatible, incorrectly bound, or untraceable mandatory Node interface constitutes nonconformance with this requirement.

Verification includes:

- Inspection of each Node interface definition
- Inspection of interface identities, directions, endpoints, information structures, protocols, constraints, and security controls
- Comparison of provided and required interfaces
- Validation of the information exchanged through each interface
- Confirmation that interface versions and compatibility rules are identified
- Testing of connection, normal interaction, disconnection, timeout, error, and recovery behaviour
- A negative assessment involving incompatible provided and required interfaces
- A negative assessment involving an absent or incorrectly configured mandatory interface

- Inspection of runtime endpoint bindings, interface records, Evidence, and Traceability

Referenced By

Plugin Backlinks: Nothing was found.

Related Architecture Sections

- Add links to the architecture sections governing Node interaction, information exchange, communication, interface contracts, security, configuration, and observability.

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-001-define-a-node:nod-001c-define-node-interfaces#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-001-define-a-node/nod-001c-define-node-interfaces>

Last update: 2026/08/17 14:20

