

NOD-010d — Inject a Node Fault

[Go to NOD-010 — Inject Node Faults](#)

Statement

The [DIDO-TE](#) SHALL introduce each authorized [Fault](#) into the specified [Node](#) or condition governing its operation using the governing [Test Procedure](#).

Derived From

- [NOD-010 — Inject Node Faults](#)
- [NOD-010a — Specify a Node Fault](#)
- [NOD-010b — Authorize Node Fault Injection](#)
- [NOD-010c — Prepare Node Fault Injection](#)
- [\[DTE5\] 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

Introducing the specified Fault performs the authorized [Fault Injection](#) after completion of the required specification, authorization, and preparation activities.

The governing [Test Definition](#) identifies the Fault, target Node, injection point, initiating condition, timing, duration, intensity, scope, expected effects, prohibited effects, termination conditions, and recovery conditions. The governing Test Procedure specifies the actions and sequence used to introduce the Fault.

The DIDO-TE verifies the identity and state of the target Node immediately before introducing the Fault. It also verifies the readiness of the injection mechanism, observation sources, containment controls, termination controls, recovery mechanisms, and required Test Resources.

The DIDO-TE preserves the distinction between the fault-injection action, the introduced Fault, and the observed Node response. Successful introduction of the Fault does not establish whether the Node produces the expected Test Outcome.

An incorrectly targeted, mistimed, excessive, incomplete, ineffective, or unauthorized Fault Injection may invalidate Test Results or affect Nodes, Test Resources, Test Environments, or Test Executions outside the authorized scope.

Applies To

- [DIDO-TE](#)
- [Fault](#)
- [Fault Injection](#)
- [Node](#)
- [Node Configuration](#)
- [Node Binding](#)
- [Dependency](#)
- [Test Definition](#)
- [Test Procedure](#)
- [Test Environment](#)
- [Test Execution](#)
- [Test Resource](#)
- [Test Outcome](#)
- [Test Result](#)
- [Authorizing Authority](#)
- [Evidence](#)
- [Traceability](#)

Verification

Verification confirms that:

1. The Fault Injection has current authorization from the identified Authorizing Authority.
2. The authorization covers the specified Fault, target Node, injection point, timing, duration, intensity, and scope.
3. Every preparation action specified by the governing Test Procedure is complete.
4. The DIDO-TE uniquely identifies the target Node before introducing the Fault.
5. The DIDO-TE verifies the Node Configuration, Node Bindings, lifecycle state, and readiness state before introducing the Fault.
6. The DIDO-TE verifies the readiness of the injection mechanism.
7. The DIDO-TE verifies the readiness of required observation sources, containment controls, termination controls, recovery mechanisms, and Test Resources.
8. The DIDO-TE introduces the Fault into the target specified by the governing Test Definition.
9. The DIDO-TE introduces the Fault at the specified injection point.
10. The DIDO-TE introduces the Fault when the specified initiating condition occurs.
11. The DIDO-TE introduces the Fault at the specified time, intensity, duration, and scope.
12. The DIDO-TE prevents introduction of the Fault before the specified initiating condition occurs.
13. The DIDO-TE prevents introduction of the Fault into an unauthorized Node or operating condition.
14. The DIDO-TE prevents the Fault from extending beyond its authorized scope.
15. The DIDO-TE observes whether the specified Fault was introduced.
16. The DIDO-TE distinguishes the introduction of the Fault from the Node response.
17. The DIDO-TE detects an unsuccessful, incomplete, ineffective, delayed, premature, mistimed, excessive, misdirected, or unauthorized Fault Injection.
18. The DIDO-TE detects an effect prohibited by the governing Test Definition.
19. The DIDO-TE initiates exception handling when the Fault Injection differs from its governing

specification or authorization.

20. The DIDO-TE maintains Traceability among the authorization, Fault, Fault Injection, target Node, governing Test Definition, governing Test Procedure, injection action, observations, Test Outcomes, Test Results, and resulting Evidence.
21. An unsuccessful, incomplete, ineffective, delayed, premature, mistimed, excessive, misdirected, unauthorized, or untraceable Fault Injection constitutes nonconformance with this requirement.

Referenced By

The following pages reference this requirement:

- [NOD-010e — Control a Node Fault](#)
- [NOD-010f — Recover from a Node Fault](#)
- [NOD-010g — Record Node Fault Injection](#)

Delivery Phase

- ☐ Assign the delivery phase.

Implementation Status

- ☐ Assign the implementation status.

Requirement Status

- ☐ The proposed derived requirement requires review and acceptance.

Issues

- ☐ Assign the source requirement identifier and obligation from the DIDO-TE Requirements Register.

Notes for Editors

```
{{section>dido:03-dido-te:99-annexes:annex-c-requirements:03-node-requirements:nod-010-inject-node-faults:nod-010d-inject-a-node-fault#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-010-inject-node-faults/nod-010d-inject-a-node-fault>

Last update: 2026/08/17 14:20

