

NOD-009d — Handle a Node Resource-Control Exception

[Go to NOD-009 — Control Node Resources](#)

Statement

The [DIDO-TE](#) SHALL handle each [Test Execution Exception](#) arising from the control of a [Node](#) resource using the governing [Test Procedure](#).

Derived From

- [NOD-009 — Control Node Resources](#)
- [NOD-009a — Establish Node Resource Controls](#)
- [NOD-009b — Allocate Node Resources](#)
- [NOD-009c — Control Node Resource Use](#)
- [\[DTE5\] 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 resource-control exception arises when resource allocation, access, use, modification, restriction, suspension, restoration, retention, or release differs from a condition specified by the governing [Test Definition](#).

Such conditions include an unavailable resource, unsuccessful allocation, unauthorized access, excessive consumption, exhaustion, contention, leakage, unsuccessful restriction, ineffective control action, unintended effect, or failure to release a resource.

The governing Test Procedure specifies the actions used to detect, contain, classify, assess, correct, escalate, and disposition the exception. The disposition identifies whether the DIDO-TE continues, restricts, suspends, restores, retries, terminates, or invalidates the affected activity.

Exception handling prevents a resource-control failure from changing test conditions without detection, affecting unrelated Nodes, exhausting shared resources, or allowing invalid Test Results to be treated as valid.

The DIDO-TE preserves the distinction between the exception, its cause, its effects, and its disposition. A detected exception remains unresolved until the DIDO-TE verifies the required disposition and resulting resource and Node states.

Applies To

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

Verification

Verification confirms that:

1. The governing Test Definition identifies the resource-control conditions whose violation produces a Test Execution Exception.
2. The governing Test Procedure specifies the actions used to handle each resource-control exception.
3. The DIDO-TE uniquely identifies each resource-control exception.
4. The DIDO-TE identifies the affected Node and resource.
5. The DIDO-TE identifies the expected condition and the observed condition.
6. The DIDO-TE assigns a Test Execution Exception Type to each exception.
7. The DIDO-TE identifies the time, sequence, source, scope, severity, and potential effect of each exception.
8. The DIDO-TE identifies affected Nodes, resources, Test Environments, Test Executions, and Test Results.
9. The DIDO-TE contains or restricts the affected resource use when the governing Test Procedure requires containment or restriction.
10. The DIDO-TE prevents the exception from unnecessarily affecting unrelated Nodes, resources, or Test Executions.
11. The DIDO-TE retries a resource-control action only when the governing Test Procedure specifies a retry.
12. The DIDO-TE prevents unlimited, duplicate, or uncontrolled retries.
13. The DIDO-TE assigns a disposition to each resource-control exception.
14. Each disposition identifies the responsible Authorizing Authority, rationale, scope, required actions, and governing conditions.
15. The DIDO-TE verifies completion of each containment, corrective, recovery, termination, or escalation action.
16. The DIDO-TE determines the effect of the exception on affected Test Executions and Test

Results.

17. The DIDO-TE identifies the resulting Node and resource states.
18. The DIDO-TE prevents closure of the exception without a disposition and supporting Evidence.
19. The DIDO-TE maintains Traceability among the Node, resource, governing Test Definition, governing Test Procedure, resource control, exception, disposition, corrective actions, Test Executions, Test Results, and resulting Evidence.
20. An undetected, unhandled, unauthorized, unresolved, incorrectly dispositioned, or untraceable resource-control exception constitutes nonconformance with this requirement.

Referenced By

The following pages reference this requirement:

- [NOD-009e — Release Node Resources](#)
- [NOD-009f — Record Node Resource Control](#)

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-009-control-node-resources:nod-009d-handle-a-node-resource-control-exception#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-009-control-node-resources/nod-009d-handle-a-node-resource-control-exception?rev=1787001647>

Last update: 2026/08/17 14:20

