

ENV-005e — Emulate Resource Conditions

[Go to ENV-005 — Emulate Distributed Systems](#)

Statement

The [DIDO-TE](#) SHALL emulate resource conditions in accordance with the applicable [Test Definition](#).

Derived From

- [ENV-005 — Emulate Distributed Systems](#)
- [ENV-005b — Emulate Participating Nodes](#)
- [ENV-005c — Emulate Communication Behaviour](#)
- [\[DTE5\] 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

The behaviour of a distributed system depends on the resources available to its participating [Nodes](#) and supporting [Test Resources](#).

Relevant resource conditions include processor capacity, memory, persistent storage, temporary storage, network bandwidth, connection capacity, file descriptors, threads, processes, queues, accelerators, and external service capacity.

Resource availability, allocation, contention, exhaustion, and recovery affect throughput, latency, scheduling, message handling, persistence, error handling, and system stability. Controlled resource emulation permits the DIDO-TE to reproduce nominal, constrained, contended, and exhausted resource conditions.

The applicable Test Definition identifies each resource, its allocation, applicable limits, scheduled changes, permitted variation, and acceptance criteria.

Applies To

- [Test Environment](#)
- [Test Definition](#)
- [Test Execution](#)
- [Test Object](#)
- [Node](#)
- [Test Resource](#)

- [Configuration](#)
- [Constraint](#)
- [Evidence](#)

Verification

Verification confirms that:

- The applicable Test Definition identifies each resource subject to emulation.
- Each resource has a unique identity and specified owner or consumer.
- The Test Definition specifies the resource type, capacity, allocation, availability, limit, and permitted variation.
- The Test Definition specifies each required contention, restriction, exhaustion, release, or recovery condition.
- Each specified resource condition identifies its required value, state, range, sequence, duration, or tolerance.
- The DIDO-TE establishes the specified initial resource conditions before Test Execution.
- The DIDO-TE applies each scheduled resource-condition change during Test Execution.
- The DIDO-TE measures actual resource allocation, availability, consumption, contention, and exhaustion during their applicable periods.
- The DIDO-TE detects a resource condition outside its specified criteria or tolerance.
- The DIDO-TE restores a resource condition when the Test Definition specifies recovery.
- The DIDO-TE records resource identities, allocations, limits, measurements, scheduled changes, actual changes, exhaustion events, recovery events, and detected deviations as [Evidence](#).
- The DIDO-TE maintains [Traceability](#) among each resource condition, affected Node or Test Resource, Test Definition, Test Execution, Evidence, and [Test Results](#).
- A missing, incorrect, uncontrolled, unmeasured, unsupported, or untraceable mandatory resource condition constitutes nonconformance with this requirement.

Verification includes:

- Inspection of the resources and resource conditions specified by the Test Definition
- Inspection of resource identities, consumers, capacities, allocations, limits, sequences, durations, and tolerances
- Comparison of specified and emulated resource conditions
- Observation of resource measurements during Test Execution
- Testing under nominal and constrained resource conditions
- Testing with resource contention or exhaustion
- Testing with a scheduled change in resource availability or allocation
- Testing resource release and recovery when specified
- Confirmation that the DIDO-TE detects and records each unpermitted resource deviation
- Inspection of resource-condition Evidence and Traceability records
- Confirmation that each mandatory resource-condition deviation produces a nonconformance result

Referenced By

- [ENV-005f — Emulate Node Failures](#)

- [ENV-005l — Record Distributed-System Emulation](#)
- [ENV-005m — Validate Emulation Fidelity](#)

Related Architecture Sections

- Add links to the architecture sections governing resource allocation, resource constraints, resource observation, distributed execution, 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:02-test-environment-requirements:env-005-emulate-distributed-systems:env-005e-emulate-resource-conditions#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/03-01-test-environment-requirements/env-005-emulate-distributed-systems/env-005e-emulate-resource-conditions>

Last update: 2026/08/18 12:34

