

Logical Node Lifecycle State

[Return to L Terms](#)

Discussion

Logical Node Lifecycle State describes the logical operational condition of a Logical Node as seen by the architecture.

Logical lifecycle states support operational coordination, monitoring, recovery, replay, evidence, and review. The Logical Architecture uses those states to describe the Node condition without prescribing the implementation mechanism for reporting or storing the state.

Logical Node Lifecycle State uses conditions such as Starting, Running, Degraded, Recovering, Stopping, Stopped, Failed, and Unknown. Implementation profiles map these logical states to concrete runtime mechanisms, logs, status topics, API responses, monitoring metrics, or orchestration status, depending on the selected implementation approach.

Definition

Logical operational condition associated with a Logical Node.

Source

Specialization of Node, Runtime Plane, Evidence, and Traceability from Part 1, Sections 7.2, 7.5, 7.9, and 7.10; generalises control-plane and health-monitoring material from the original FX Demo Reference Architecture.

Note

Logical Node Lifecycle State does not redefine the Logical Node's architectural role. It describes the operational condition of the Logical Node at a point in time or over an interval.

Example

A health-monitoring interaction reports that a Logical Node moved from Starting to Running, or from Running to Degraded, without prescribing whether the implementation reports that state through DDS, REST, RPC, logs, metrics, or another mechanism.

[Return to Top](#)

From:
<https://wiki.didosolutions.com/> - **Dido Solutions, Inc Wiki**

Permanent link:
https://wiki.didosolutions.com/dido/99_annexes/annex-b-terms-and-definitions/l/logical_node_lifecycle_state

Last update: **2026/07/18 12:33**

