16.2 Node Exceptions

Go To Top

Return to Exception Handling

Nodes should handle exceptions in a way that preserves runtime visibility and Lifecycle discipline. A Node should not crash silently, continue after a critical failure, or report Running when it cannot perform its required Phase 0 role.

Node exception handling should identify:

1. Configuration errors.
2. Startup errors.
3. DDS participant initialization errors.
4. Publisher or subscriber creation errors.
5. Generated type or binding errors.
6. Topic publication or subscription errors.
7. Command-handling errors.
8. Data-processing errors, where applicable.
9. Logging failures.
10. Shutdown errors.
11. Unexpected runtime exceptions.

When a Node detects an exception, it should log the error with relevant context. Where possible, it should publish an appropriate Control Plane status such as Degraded, Recovering, or Failed. If the Node cannot continue safely, it should move toward controlled shutdown and exit with a meaningful exit code.

A Node may continue running after a recoverable error if the Node profile defines the recovery behavior. For example, a Node may reject an invalid command, log the rejection, publish current status, and continue Running. A Node should not continue Running after a failure that prevents required startup, DDS participation, status reporting, or safe shutdown.
