

14.1 Purpose of Containerisation in Phase 0

[Go To Top](#)

[Return to Containerisation](#)

Phase 0 uses containerisation to make the executable [Baseline](#) easier to reproduce across developer workstations, test environments, CI runners, and demonstration environments. A container should package a defined runtime participant or support service with the dependencies it needs to run.

Containerisation should support the following goals:

1. Reduce differences between developer machines.
2. Package Nodes and support services consistently.
3. Make build and run workflows repeatable.
4. Keep runtime Configuration explicit.
5. Support controlled startup and shutdown.
6. Preserve logs and [Evidence](#).
7. Improve troubleshooting.
8. Prepare the project for later deployment refinement.

Containers should not hide architectural meaning. A container packages an implementation [Artifact](#). It does not define the logical architecture, Node Role, topic model, [Lifecycle](#) semantics, or data-governance meaning. The Node Catalogue, Topic Catalogue, [IDL](#) definitions, [Configuration](#) files, and architecture documents remain the authoritative sources for those concerns.

© 2026 Dido Solutions, Inc. and Jackrabbit Consulting, Inc.

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

Permanent link:
<https://wiki.didosolutions.com/fxdemo/05-part/14-containerisation/14-1-purpose-of-containerisation-in-phase-0/start>

Last update: **2026/08/10 15:28**

