

14.2 Container Design Principles

[Go To Top](#)

[Return to Containerisation](#)

The team should design containers around clear runtime responsibilities. In Phase 0, a container may package one [Node](#), a support service, an observer, or a tooling function. The container name, image name, Configuration, logs, and documentation should make that responsibility clear.

Container design should follow these principles:

1. Keep container responsibilities clear and narrow.
2. Avoid bundling unrelated Nodes or tools into one container unless the phase Baseline requires it.
3. Keep runtime Configuration outside the image where practical.
4. Avoid embedding secrets, credentials, personal paths, or machine-specific settings in images.
5. Use reproducible build inputs.
6. Use approved generated [Artifacts](#) and source files.
7. Keep image names and tags consistent with the naming conventions.
8. Make logs available outside the container where acceptance Evidence requires them.
9. Allow controlled startup and shutdown.
10. Avoid requiring manual steps inside a running container.

A container should run the same Artifact that the team documents in the Node Catalogue or support-service documentation. If a container image contains generated DDS types, Node code, scripts, or Configuration defaults, the build process should trace those inputs to Repository-controlled sources.

© 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/14-2-container-design-principles/start>

Last update: **2026/08/10 15:29**

