

2.6 Relationship to Generated Code and Runtime Artefacts

[Go To Top](#)

[Return to Relationship to the Architecture Documents](#)

Generated code and runtime [Artifacts](#) project the architecture into implementation form. They do not become the architecture itself. The team should treat [IDL](#)-derived files, generated language bindings, [Container Images](#), logs, [Configuration](#) files, and runtime outputs as Artifacts produced from [Governed](#) source material and repeatable processes.

This handbook defines how the team separates generated Artifacts from handwritten source files. Generation processes should produce reproducible, identifiable, and [Traceable](#) Artifacts. They should not overwrite handwritten source files. Runtime Artifacts, such as logs and container outputs, should provide [Evidence](#) of execution. They should not define architectural meaning.

This distinction protects the architectural discipline. The Phase 0 [Baseline](#) should show that the team can generate, build, run, observe, and review implementation Artifacts without allowing generated code or runtime convenience to become the source of architectural truth.

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

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

Permanent link:
<https://wiki.didosolutions.com/fixdemo/05-part/02-relationship-to-the-architecture-documents/02-6-relationship-to-generated-code-and-runtime-artifacts/start>

Last update: 2026/08/08 18:58

