

4.1 Repository Design Principles

[Go To Top](#)

[Return to Repository Layout](#)

The Phase 0 [Repository](#) should make the implementation easy to understand, reproduce, review, and extend. The team should organise the Repository around clear [Artifact](#) types and responsibilities rather than around individual developer preferences.

The Repository layout should support the following principles:

- Separate handwritten source files from generated Artifacts.
- Separate architecture-facing documentation from developer-facing documentation.
- Separate source definitions from build outputs and runtime outputs.
- Separate reusable implementation code from demonstration-specific [Configuration](#).
- Keep scripts, Configuration files, [IDL](#) definitions, tests, and documentation in predictable locations.
- Ensure that every top-level directory has a clear purpose.
- Avoid hidden [Dependencies](#) on local machine paths, shell profiles, IDE settings, or personal tooling.

The Repository should help the team answer basic questions quickly: where does the source code live, where do IDL definitions live, where does the generated code go, where do scripts live, where does the Configuration live, where do logs go, and which Artifacts should [Git](#) track?

© 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/04-repository-layout/04-1-repository-design-principles/start?rev=1786242135>

Last update: 2026/08/08 19:22

