

10.2 Traceability Across Layers

[Go to Top](#)

[Traceability](#) across layers preserves architectural meaning as later parts specialize, realize, deploy, and provide [Evidence](#) for the conceptual architecture.

Each layer has a distinct [Traceability](#) responsibility:

Table 10.2-1: Traceability responsibilities across architectural layers

Layer	Traceability responsibility
Conceptual Architecture	Defines stable concepts and conceptual boundaries.
Logical Architecture / Platform Independent Model (PIM)	Specializes conceptual elements into platform-independent logical structures, relationships, and constraints.
Implementation Profile / Platform Specific Model (PSM)	Maps logical elements to selected implementation technologies and artifacts.
Deployment Profile	Describes how implementation artifacts run in a concrete environment.
Evidence	Supports claims about architecture, implementation, deployment, execution, Interpretation , Governance and Authority , or review.

[Traceability](#) keeps these layers connected without allowing one layer to redefine another. A logical element may specialize a conceptual element. An implementation artifact may realize a logical element. A deployment artifact may execute or configure an implementation artifact. [Evidence](#) may support a claim about any of these layers.

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

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

Permanent link:
<https://wiki.didosolutions.com/fxdemo/01-part/10-conceptual-traceability-model/10-2-traceability-across-layers>

Last update: **2026/07/18 12:33**

