

12.1 Overview

Go To Top

The Logical Traceability Model defines how the distributed, node-based Logical Architecture / PIM preserves relationships among Part 1 conceptual elements, Part 2 logical elements, domain-profile specialisations, implementation mappings, deployment artifacts, and [Evidence](#) expectations.

Logical [Traceability](#) allows authors and reviewers to follow a concern from conceptual meaning to logical structure. It also prepares later parts to trace logical elements to implementation artifacts, deployment artifacts, test scenarios, runtime observations, and [Evidence](#) records.

The Logical Traceability Model does not prescribe a traceability tool, repository structure, modeling notation, requirements-management system, issue tracker, metadata format, or evidence store. Implementation and governance profiles select mechanisms that realize logical [Traceability](#) for a specific implementation context.

Figure 12-1 summarises the logical traceability chain.

```
Part 1 Conceptual Element
└─ Part 2 Logical Element / PIM
    └─ Domain Logical Profile Element
        └─ Implementation Artifact / PSM
            └─ Deployment Artifact
                └─ Evidence
```

Figure 12-1: Logical traceability chain from concept to evidence

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

From:

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

Permanent link:

<https://wiki.didosolutions.com/fxdemo/02-part/12-logical-traceability-model/12-1-overview/start>

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

