

P1-REQ-13-10-006

[Return to Part 1 Requirements](#)

Statement

A later part that introduces a new term SHALL identify whether the term is conceptual, logical, implementation-specific, deployment-specific, or evidence-specific.

Source

Part 1, Section 13.10: Profile and Layering Requirements.

Rationale

New terms introduced in later parts need clear placement within the architecture layering model. Identifying a term as conceptual, logical, implementation-specific, deployment-specific, or evidence-specific prevents ambiguity about the term's role and authority.

This requirement supports controlled vocabulary growth and prevents later parts from introducing terms that blur architectural layers.

Applies To

This requirement applies to new terms introduced in later parts of the document set.

It applies specifically to terms classified as:

- Conceptual
- Logical
- Implementation-specific
- Deployment-specific
- Evidence-specific

Verification

Verification SHALL confirm that each new term introduced in a later part identifies whether it is conceptual, logical, implementation-specific, deployment-specific, or evidence-specific.

Verification activities include review checks confirming that:

- New terms identify their architectural layer
- New terms do not conflict with existing conceptual definitions
- New terms include appropriate traceability to related terms
- New terms preserve separation among conceptual, logical, implementation, deployment, and evidence concerns

Traceability

Related source section:

- [13.10 Profile and Layering Requirements](#)

Related requirement identifiers:

- [P1-REQ-13-10-001](#)
- [P1-REQ-13-10-002](#)
- [P1-REQ-13-10-003](#)
- [P1-REQ-13-10-007](#)
- [P1-REQ-13-2-001](#)

Status

Draft

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

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

Permanent link:
https://wiki.didosolutions.com/dido/99_annexes/annex-d-requirements/part-01/p1-req-13-10-006/start

Last update: **2026/07/11 12:57**

