

Formal Semantics

[Go up to Terms and Definitions](#)

Discussion

Formal semantics defines the meaning of architectural constructs precisely and unambiguously.

Formal semantics separates meaning from implementation syntax, storage technology, programming language, exchange mechanism, deployment environment, or runtime platform.

Formal semantics supports consistent interpretation across implementations, tools, organizations, and governance contexts. It provides a basis for traceability, conformance evaluation, controlled interpretation, and semantic consistency within the [reference_architecture](#).

Definition

precise and unambiguous definition of meaning associated with architectural constructs, independent of implementation

Source

Structured Information Processing [reference_architecture](#)

Note

Formal semantics provides the basis for consistent interpretation and conformance evaluation across implementation technologies.

Example

A [reference_architecture](#) defines the meaning of an [attachment_point](#), [conformance_point](#), lifecycle state, or governed artifact independently of the software mechanism that implements it.

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

Permanent link:
https://wiki.didosolutions.com/public/99_annexes/annex-b-terms-and-definitions/f/formal_semantics?rev=1789260147

Last update: **2026/09/12 17:42**

