

15.4 Logical Data Structures Remain Distinct from Serialisation Formats

[Go To Top](#)

A [Logical Data Structure Definition](#) describes the structure, meaning, and constraints of an information class at the platform-independent logical level. It does not represent an IDL structure, a Protocol Buffers message, a JSON Schema definition, an XML Schema definition, an Avro schema, an OpenAPI schema, a database table, an ontology class, a generated type, a file, or a wire-format payload.

Implementation profiles select data representation mechanisms that realise [Logical Data Structure Definitions](#) and [Logical Data Structure Instances](#). The selected representation supports implementation, exchange, storage, validation, or processing. It does not redefine the logical information structure.

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

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

Permanent link:
<https://wiki.didosolutions.com/fixdemo/02-part/15-logical-rules-and-constraints/15-4-logical-data-structures-remain-distinct-from-serialisation-formats/start?rev=1783793243>

Last update: 2026/07/11 11:07

