CATENA-X

STANDARD

CX - 0002 DIGITAL TWINS IN CATENA-X v.1.0.2

Contact: standardisierung@catena-x.net

Table of Contents

ABOUT THIS DOCUMENT & MOTIVATION

Catena-X is the first open and collaborative data ecosystem. The goal is to provide an environment for the creation, operation, and joint use of end-to-end data chains along the entire automotive value chain. All partners are on an equal ground, have sovereign control over their data and no lock-in effects occur. This situation provides a sustainable solution for the digitalization

of supply chains, especially for medium-sized and small companies, and supports the cooperation and collaboration of market participants and competitors.

The ever-growing Catena-X ecosystem will enable enormous amounts of data to be integrated and collaboratively harnessed. To ensure that these complex data volumes can be sent, received, and processed smoothly across all stages of the value chain, one language for all players: common standards. The standards of the Catena-X data ecosystem define how the exchange of data and information in our network works. They are the basis for ensuring that the technologies, components, and processes used are developed and operated according to uniform rules.

Common standards create added value for all partners: Within our network, data flows more smoothly through interfaces. In addition, we avoid cumbersome individual IT solutions for sharing data with other partners. In the field of international standardization, Catena-X follows the proven international standardization institutions: ISO/IEC/ITU and CEN-CENELC/ETSI.

For users and data providers, implementation of standards will reduce the costs that would arise from adapting different systems. In addition, no important data is lost. On the contrary, it even becomes easier to collect data across companies. For operators and developers, standards will create a framework that provides reliable orientation and planning security.

The following document describes one of the standards used in the Catena-X ecosystem and the requirements needed to implement it. Here, it serves as main resource to illustrate the following data model. It contains information starting from the format of the model, up to the conceptual and physical model. The standardisation of the data model will enable faster information sharing and homogeneity throughout the entire Catena-X ecosystem.

DISCLAIMER & LIABILITY

REVISIONS & UPDATE

The present document may be subject to revision or change of status. Catena-X reserves the right to adopt any changes or updates to the present document as it deems necessary or appropriate.

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be copied or modified without the prior written authorization of Catena-X. In case of any existing or perceived difference in contents between any versions and/or in print, the prevailing version of the present document is the one made publicly available by Catena-X in PDF format.

If you find any errors in the present document, please send your comments to: standardisierung@catena-x.net

COPYRIGHT & TRADEMARKS

ABSTRACT

The Catena-X network is about accessing/sharing/providing/using data, formulated in the different and varied Use Cases. This standardization scenario is about how the data, and the data models should or have to look like and how the modelling has to be done, so that data between ecosystem partners can be shared, lossless and in a machine-readable way. This document focuses on Digital Twins and their application and administration within Catena-X.

The purpose of this standardization is to provide concepts and specifications in order to allow proper digital twin modelling in Catena-X

1. Introduction

1.1 Audience & Scope

> *This section is non-normative*

This standard is relevant for

- Business Application Provider
- Data Provider / Consumer
- Core Service Provider
- Enablement Service Provider
- Onboarding Service Provider
- Consulting Service Provider

The standard needs to be applied in the following cases for the following roles:

- all data providers who need to provide information via digital twins
- all data consumers and business application provider who need access to data provided via digital twins
- solution providers of a Digital Twin Registry
- onboarding service providers who need to offer core service of a Digital Twin Registry to their customers
- enabling service providers who need access to data provided via digital twins
- consulting service providers who need to explain how digital twins are implemented and/or used

1.2 Context

> *This section is non-normative*

Catena-X creates a central, uniform, and consistent solution for the daily challenges of the automotive industry. In this context, the provision or exchange of data is an essential requirement for the success of our network. For this purpose, Catena-X provides various methods, tools, and standards to ensure semantic interoperability. Digital Twins have established themselves here as a central element for structuring and accessing data. With the help of defined semantics, both data provision and app development are simplified and encouraged.

1.2.1 Digital Twins in Catena-X

The term Digital Twin (DT) describes a virtual representation of assets. Any asset -- it can be an actual physical asset like a drilling machine but also something virtual like a web service -- has a digital representation with consistent semantics. Hence, Digital Twins adhere to the following characteristics:

- The DT has at least one Catena-X-wide unique ID
- An asset can have more than one DT
- DTs are organized by a set of Aspects. The set can be extended over lifetime
- An Aspect of a DT includes both structural as well as behavioral data and models, including operations and simulation models
- The semantics of an Aspect can be described via semantic models
- A single Aspect can be connected to different heterogenous data sources, including behavioral models
- The DT can represent asset types (e.g., virtual prototype of a car) and asset instances (e.g., real car)

A DT can cover the whole asset lifecycle including, e.g., the planning, production, sales, use and decommissioning phases. However, in practice there may be more than one twin with different IDs representing different lifecycle phases, e.g., a twin for types and multiple twins for instances

The DT represents current available information about an asset, synchronized at a specified frequency and fidelity, which can be leveraged for simulation and business process integration

By using aspects, the DT can reference other DT to express "part of" or "consists of" relations

1.2.2 Digital Twin Registry

The Digital Twin Registry (DTR) is a solution which lists all digital twins and references their Aspects including information about the underlying asset, asset manufacturer, and access options, like aspect endpoints. Moreover, the DTR is used to register and find data related to DTs.

In general, every partner must proactively register their DTs / Aspects to this service in order to reveal their "offer" to share respective data sets. The data offered by DT is semantically described via a mapping to the data models stored in the Semantic Hub. Up to Release 2 Catena-X uses a central digital twin registry. DTs operating on an item level (e.g., vehicle, parts) are going to be created for Circular Economy and other use cases. And as the architecture will support decentralized digital registries, no central digital twin registry will be required any longer. Here, any DT registry can then be used as long as it's fully interoperable with DT registries across the Catena-X dataspace. Further documentation can be found under GitHub (publicly available: https://github.com/eclipse-tractusx/sldt-digital-twin-registry), both containing the official developer documentation of the DT registry.

Note: Currently, you also need to register data (i.e., each submodel) at the EDC Connector as an EDC asset because of access policies. The endpoint in the DT registry is an EDC asset endpoint and is later mapped to the "real" data endpoint.

1.2.3 Asset Administration Shell

The Asset Administration Shell (AAS) is a key concept of Industry 4.0 (or "Industrie 4.0" in German), maintained by the Industrial Digital Twin Association (IDTA), and is used to describe an asset electronically in a standardized manner. Moreover, the AAS includes a concept of sub models, each of which can characterize the asset by describing its aspects in different domains. Hence, the AAS is a set of API Methods and Resources to access data of a digital twin. In other words: AAS describes how to access which data of a digital twin.

It doesn't describe the semantics of the submodel, instead this can be done by using e.g., the SAMM standard (see CX-0003). In general, the AAS has proven to be suitable also for the following use cases:

how to represent data exchanged in a standardized way between companies (API Payload)

how to access data exchanged in a standardized way between companies (API operations)

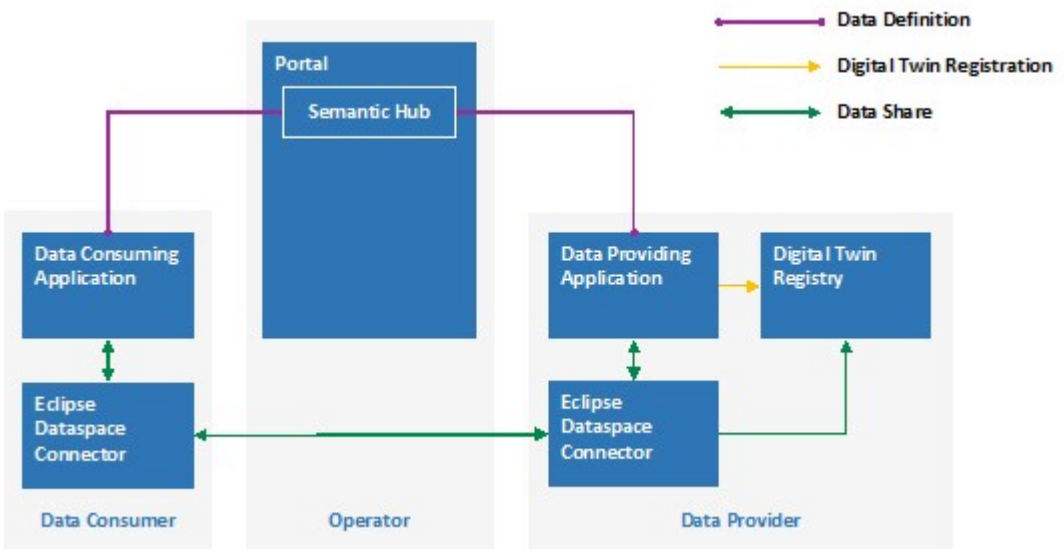
how to find data available for the asset under consideration for data exchange between companies (digital twin registry) in a standardized way

The Asset Administration Shell Reading Guide gives a good overview of different stakeholders. Detailed technical documentation can be found in the Content Hub of the IDTA and on GitHub: https://github.com/admin-shell-io/aas-specs.

1.3 Architecture Overview

> *This section is non-normative*

In the following the target architecture is sketched with decentralized Digital Twin Registries.



1.4 Conformance

As well as sections marked as non-normative, all authoring guidelines, diagrams, examples, and notes in this specification are non-normative. Everything else in this specification is normative.

The key words MAY, MUST, MUST NOT, OPTIONAL, RECOMMENDED, REQUIRED, SHOULD and SHOULD NOT in this document are to be interpreted as described in [BCP 14], [RFC2119], [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.5 Proof of conformity

> *This section is non-normative*

All participants and their solutions will need to prove that they conform to the Catena-X standards. To validate that the standards are applied correctly, Catena-X employs Conformity Assessment Bodies (CABs).

1.5.1 Proof of Conformity for Digital Twin Registry Solutions

A Digital Twin Registry Solution MUST provide http/REST APIs conformant to the openAPI specification as defined in this document.

Digital Twin Registry solution providers MUST proof their conformity by providing:

- An openAPI specification of the endpoints
- API Response of the implementation MUST match to the response structure of the API specification in this document

1.5.2 Proof of Conformity for Data Providers

A data provider MUST create and register digital twins using the http/REST APIs conformant to the openAPI specification as defined in this document.

A data provider MUST register the required aspect endpoints for each digital twin depending on the use case the data provider is involved in using the http/REST APIs conformant to the openAPI specification as defined in this document. Additionally, usage and access policies conformant to CX-0018 MUST be defined and access of the single aspects is done via the EDC.

A data provider MUST use the unique identifier of the standardized aspect model conformant to CX-0003 when registering a new aspect endpoint to a digital twin.

A data provider MUST add the required specific asset IDs for each digital twin depending on the use case the data provider is involved in.

As long as the Digital Twin Registry is centralized digital twins MUST be registered in this central Digital Twin Registry. As soon as the Digital Twin Registries are decentralized the data provider MUST use a Digital Twin Registry service with a certified Digital Twin Registry solution.

1.5.3 Proof of Conformity for Data Consumers

A data consumer, business application provider or enabling service provider MUST lookup the endpoints of the aspects relevant for the use case using the http/REST APIs conformant to the openAPI specification as defined in this document.

As soon as the Digital Twin Registries are decentralized a data consumer, business application provider or enabling service provider MUST first lookup the available Digital Twin Registry endpoint of the relevant EDCs using the corresponding standardized lookup and discovery services (see CX-0001).

Additionally, data consumers MAY use standardized discovery services – if available -, e.g., to find the EDC for a specific company via its BPN (see CX-0053).

1.6 Examples

Examples can be found in https://github.com/eclipse-tractusx/sldt-digital-twin-registry/blob/v0.2.0-M4-multi-tenancy/doc/documentation.md

1.7 Terminology

> *This section is non-normative*

Aspect

> *a domain-specific view on information and functionality associated with a specific Digital Twin with a reference to a concrete Aspect Model.*

Note 1 to entry: An Aspect is a software service to retrieve the actual runtime data of a Digital Twin (current or aggregated) from a data source or to trigger operations. Thus, an aspect is built with an implementation that ensures that the exchanged data is compliant to the specification of the referenced Aspect Model via a defined interface.

Note 2 to entry: Aspects are registered (incl. their "API endpoint" information) with the Digital Twin to which they belong in the Digital Twin Registry.

Note 3 to entry: an aspect corresponds to a *Submodel* in the *Asset Administration Shell*

Aspect Model

> *a formal, machine-readable semantic description (expressed with RDF/turtle) of data accessible from an Aspect.*

Note 1 to entry: An Aspect Model must adhere to the Semantic Aspect Meta Model (SAMM), i.e., it utilizes elements and relations defined in the Semantic Aspect Meta Model and is compliant with the validity rules defined by the Semantic Aspect Meta Model.

Note 2 to entry: Aspect models are logical data models which can be used to detail a conceptual model in order to describe the semantics of runtime data related to a concept. Further, elements of an Aspect model can/should refer to terms of a standardized Business Glossary (if existing).

Note 3 to entry: An Aspect Model describes the semantics of a *Submodel*.

Asset Administration Shell

> *standardized digital representation of an asset*

Note 1 to entry: Asset Administration Shell and Administration Shell are used synonymously.

[SOURCE: IEC 63278-1, note added]

Digital Twin

> *digital representation, sufficient to meet the requirements of a set of use cases*

Note 1 to entry: in this context, the entity in the definition of digital representation is typically an asset.

[SOURCE: IIC Vocabulary IIC:IIVOC:V2.3:20201025, adapted (an asset, process, or system was changed to an asset)]

Digital representation

> *information and services representing an entity from a given*

EXAMPLE 1: examples of information are properties (e.g., maximum temperature), actual parameters (e.g., actual velocity), events (e.g., notification of status change), schematics (electrical), and visualization information (2D and 3D drawings).

EXAMPLE 2: examples of services are providing the history of the configuration data, providing the actual velocity, and providing a simulation.

EXAMPLE 3: examples of viewpoints are mechanical, electrical, or commercial characteristics.

[SOURCE: IEC 63278-1, editorial changes]

Submodel

> *container of SubmodelElements defining a hierarchical structure consisting of SubmodelElements*

[SOURCE: IEC 63278-1]

SubmodelElement

> *elements in a Submodel*

[SOURCE: IEC 63278-1]

Additional terminology used in this standard can be looked up in the glossary on the association homepage.

2. Digital Twin Registry API [NORMATIVE]

2.1 Preconditions and dependencies

The submodel endpoints registered for a digital twin MUST be published towards the network using a Data Asset in terms of the IDS Protocol following the Catena-X standard CX-0018.

2.2 API Specification

2.2.1 API Endpoints & resources

The API MUST be implemented as specified in the openAPI specification.

This API specification follows the API specification for "Registry-and-Discovery" as defined in Details of the Asset Administration Shell - Part 2. Interoperability at Runtime – Exchanging Information via Application Programming Interfaces. V1.0RC01 with minor deviations and adjustments. A documentation of the deviations can be found in https://github.com/eclipse-tractusx/sldt-digital-twin-registry/blob/v0.2.0-M4-multi-tenancy/doc/documentation.md.

The following tables just give an overview.

Registry:

Interface-Operation(s): API-Operation(s)	Parameter	Comment
Interface: Asset Administration Shell Registry (GetAllAssetAdministrationShellDescriptors, GetAssetAdministrationShellDescriptorById, PostAssetAdministrationShellDescriptor, PutAssetAdministrationShellDescriptorById, DeleteAssetAdministratiionShellDescriptorById): POST PUT DELETE GET /registry/*	BASE64-URL-encoded identifier	
Interface: Submodel Registry (GetAllSubmodelDescriptors, GetSubmodelDescriptorById, PostSubmodelDescriptor, PutSubmodelDescriptorById, DeleteSubmodelDescriptorById): POST PUT DELETE GET /registry/shelldescriptors/{aasIdentifier}/submodelDescriptors/*	BASE64-URL-encoded identifier	
-: POST /registry/shell-descriptors/batch		For data provider

Lookup:

Interface-Operation(s): API-Operation(s)	Parameter	Comment
GetAllAssetAdministrationShellsByAssetLink: GET /lookup/shells	BASE64-URL-encoded JSON-serialized key-value-pairs ?assetids=...	For data consumer
GetAllAssetLinksById: GET /lookup/shells/{aasIdentifier}	BASE64-URL-encoded identifier	For data consumer
PostAllAssetLinksById: POST /lookup/shells/{aasIdentifier}	BASE64-URL-encoded identifier	For data provider
DeleteAllAssetLinksById: DELETE /lookup/shells/{aasIdentifier}	BASE64-URL-encoded identifier	For data provider
-: POST /lookup/shells/fetch		For data consumer

A swagger implementation is also available: https://semantics.int.demo.catena-x.net/registry/swagger-ui/index.html

This API is independent from the architecture, i.e., the same API MUST be provided for the centralized and decentralized approach of Digital Twin Registries.

2.2.2 Available Data Types

The API MUST use JSON as the payload transported via HTTP.

For explanation of data types see Details of the Asset Administration Shell - Part 2. Interoperability at Runtime – Exchanging Information via Application Programming Interfaces. V1.0RC01.

2.2.3 EDC Data Asset Structure

For the central Digital Twin Registry, no EDC asset needs to be registered.

2.2.4 Error Handling

Error response **501 Not Implemented** MUST be used for parameter values not yet supported.

For error handling see Details of the Asset Administration Shell - Part 2. Interoperability at Runtime – Exchanging Information via Application Programming Interfaces. V1.0RC01.

3. Submodel API [NORMATIVE]

3.1 Preconditions and dependencies

Data Provider: The submodel endpoints registered for a digital twin MUST be published towards the network using a Data Asset in terms of the IDS Protocol following the Catena-X standard CX-0018.

3.2 API Specification

3.2.1 API Endpoints & resources

The API for consuming submodels/aspects MUST be conformant to the following Catena-X profile defining a subset of API, API operations and parameters as specified in Details of the Asset Administration Shell - Part 2. Interoperability at Runtime – Exchanging Information via Application Programming Interfaces. V1.0RC01:

API-Operation	Parameter	Comment
GetSubmodel GET /submodel	level=deep content=value extent=WithBLOBvalue	
InvokeOperationSync POST /submodel/submodel-elements//invoke		So far it has only been used for organizational twin and VIN/VAN conversion.

3.2.2 Available Data Types

The API MUST use JSON as the payload transported via HTTP. The payload is defined in CX-0003.

3.2.3 Error Handling

Error response **501 Not Implemented** MUST be used for parameter values not yet supported.

For error handling see Details of the Asset Administration Shell - Part 2. Interoperability at Runtime – Exchanging Information via Application Programming Interfaces. V1.0RC01.

4. References

4.1 Normative References

none

4.2 Non-Normative References

> *This section is non-normative*

Asset Administration Shell Reading Guide

CX – 0003 SEMANTIC ASPECT META MODEL. In Catena-X Standard Library

CX – 0018 ECLIPSE DATA SPACE CONNECTOR (EDC). In Catena-X Standard Library

CX – 0001 EDC DISCOVERY API. In Catena-X Standard Library

Details of the Asset Administration Shell - Part 1. The exchange of information between partners in the value chain of Industrie 4.0. R3.0RC02

Details of the Asset Administration Shell - Part 2. Interoperability at Runtime – Exchanging Information via Application Programming Interfaces. V1.0RC01

4.3 Reference Implementations

> *This section is non-normative*

The following open-source project implements a Digital Twin Registry solution conformant to this standard:

https://github.com/eclipse-tractusx/sldt-digital-twin-registry