IDTA 02027-1-0 Asset Interfaces Mapping Configuration

June 2024

SPECIFICATION

Submodel Template of the Asset Administration Shell

Imprint

Publisher
Industrial Digital Twin Association
Lyoner Strasse 18
60528 Frankfurt am Main
Germany
https://www.industrialdigitaltwin.org/

Version history

Date	Version	Comments
2024-06-14	1.0	Release of the official Submodel template published by IDTA.

Contents

Figures

Tables

1 General

1.1 About this document

This document is a part of a specification series. Each part specifies the contents of a Submodel template for the Asset Administration Shell (AAS). The AAS is described in [1], [2], [3] and [6]. First exemplary Submodel contents were described in [4], while the actual format of this document was derived by the "Administration Shell in Practice" [5]. The format aims to be very concise, giving only minimal necessary information for applying a Submodel template, while leaving deeper descriptions and specification of concepts, structures and mapping to the respective documents [1] to [6].

The target group of the specification are developers and editors of technical documentation and manufacturer information, which are describing assets in smart manufacturing by means of the Asset Administration Shell (AAS) and therefore need to create a Submodel instance with a hierarchy of SubmodelElements. This document especially details on the question, which SubmodelElements with which semantic identification shall be used for this purpose.

1.2 Scope of the Submodel

This Asset Interfaces Mapping Configuration (AIMC) Submodel specifies an information model and a common representation for describing the mapping of interface(s) of an asset service or asset-related service already described in an Asset Interfaces Description (AID) Submodel. It can be understood as a configuration Submodel for south-bound communication between AAS and asset. Based on this information in the AIMC Submodel, it is possible to configure and initiate a connection to an asset service and map payloads to intended locations in an AAS automatically, and vice versa.

As a complement to the AID, an AIMC Submodel is specified to map the received data from asset services to a specific place within an AAS (e.g. an application specific Submodel to monitor or log data). The principle scope and use of the AID and AIMC Submodels is explained in the following Figure 1.

Even though AID and AIMC can be closely used together, they are separated Submodels. AID describes available interfaces of an asset and is usually provided by the asset provider. On the other hand, AIMC configures a (bidirectional) mapping between an asset and an AAS and is usually provided by the asset user. Furthermore, an AIMC in an AAS A might reference an AID in an AAS B.

- Example 1: An AAS shall integrate data from the asset it represents uniquely. Consequently, AID and AIMC are both integrated into the only one AAS of the asset.
- Example 2: An AAS of a product shall integrate product specific production data from a machine that manufactures the product. In this case, the AIMC in the AAS of the product references an AID in the AAS of the machine.

The AIMC 1.0 adapts to the structure and protocols supported by AID 1.0 standard: MQTT, HTTP and Modbus. In this version only a unidirectional mapping from asset to AAS is considered. Subsequent releases of AID versions may trigger the release of adapted AIMC versions.

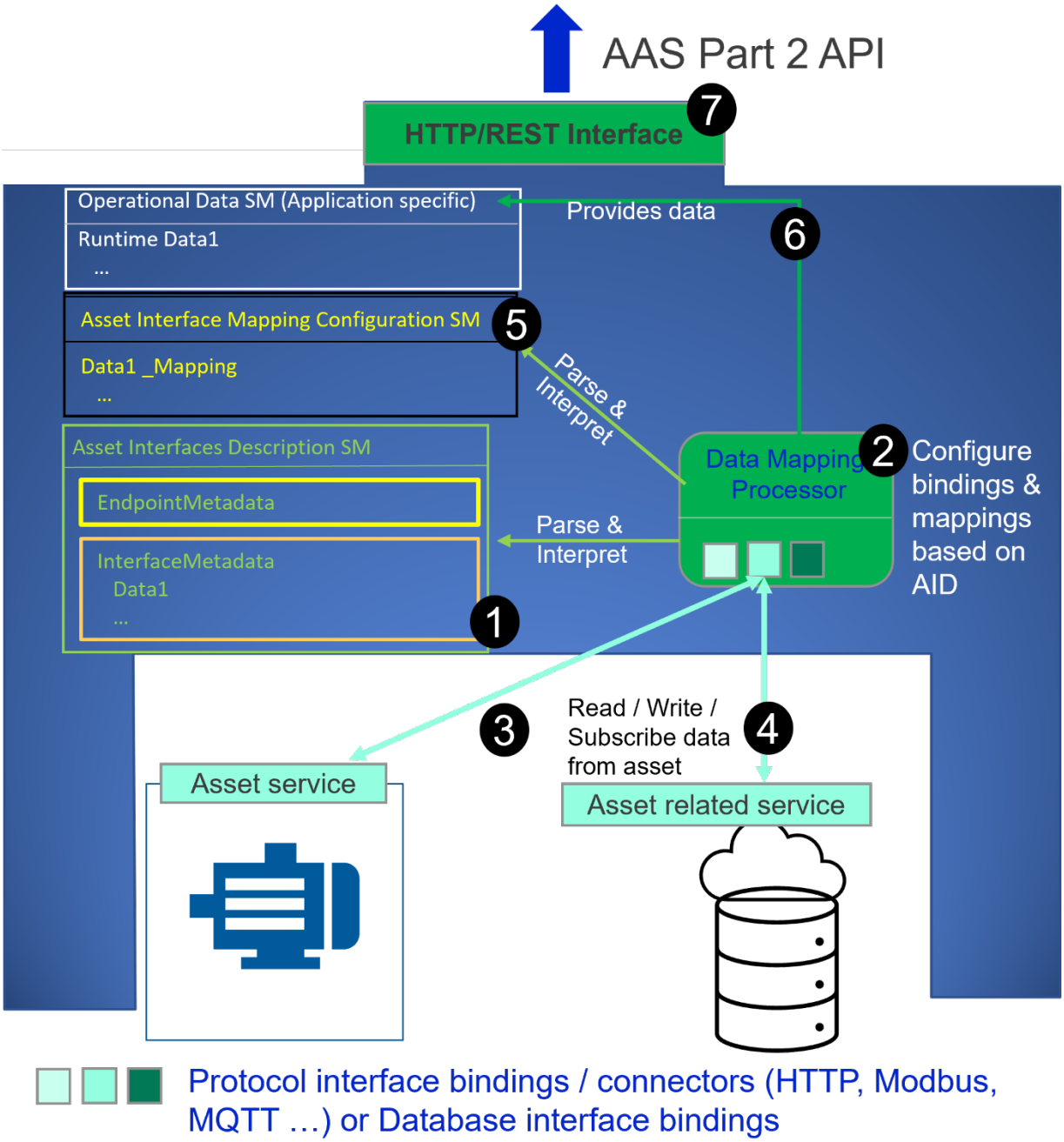

Figure 1: AIMC usage in conjunction with AID Submodel

The number legends in **Figure 1** are described as follows:

(1) Asset Interfaces Description Submodel: it holds the description model of the asset service (or asset related service) interfaces and its datapoint.

(2) Data Mapping Processor (DMP): This is a software component that provides connection to the asset and exchanges data as defined within the AID Submodel. It also manages the mapping of retrieved data to a desired SM according to AIMC SM definition.

(3) Data transmission channel between Data Mapping Processor and asset service. Depending on the underlying protocol (e.g. Modbus, MQTT) used by the asset service (and as described by the AID), the specific data point can be requested/subscribed.

(4) Data transmission channel between Data Mapping Processor and asset related service. Depending on the underlying protocol (e.g. HTTP) used by the asset related service (and as described by the AID), the specific data point can be requested/subscribed.

(5) AIMC Submodel: it provides the necessary information about the mapping of the datapoints described by the AID to elements in a desired submodel.

(6) Operational Data Submodel: it is a target submodel where the (runtime) data is being stored. The details about this location are in the AIMC. With AIMC's information, the Data Mapping Processor can correctly map the asset's data to the right parts of the target submodel.

(7) HTTP/REST AAS Interface [11] used to enable communication between AAS server and external applications.

1.3 Not in Scope of the Submodel

The interface reference and datapoint mappings described in AIMC are parsed and interpreted by an application (e.g. Data Mapping Processor). This application is beyond the scope of this document, but its main functionality is to map received payloads from asset to a specification in the AAS. It can be packaged with several communication protocol adapters and database integration drivers.

The creation of the target submodel is out of scope for this submodel. It is assumed that a target submodel exists before an AIMC references it.

1.4 Relevant standards for the Submodel template

- W3C Web of Things Thing Description [7]
- Modbus [8]
- MQTT [9]
- HTTP [10]

1.5 Use cases, requirements and design decisions

1.1.1 Use Cases

Table 1: AIMC Use Cases

Use Case	Explanation
Asset Data Integration	1. The target user (TU) connects to the AAS of the asset of interest 2. The TU connects to the Asset Interfaces Description Submodel describing which data (semanticId) in which quality (semanticId) the asset provides through which interfaces with which security requirements. 3. The TU selects relevant data points from the relevant asset interfaces. 4. The TU configures one client per relevant asset interface fulfilling the security requirements and configuring a mapping (via an Asset Interfaces Mapping Configuration Submodel) of the selected asset's data into a data sink such as a separate monitoring or logging Submodel.

	5. The configured client is deployed, connects to the asset's specified interfaces and integrates data into the specified data sinks. Example: • An application-specific OperationData Submodel is intended to provide measurement data of a sensor asset. • The AID Submodel describes the sensor asset's interface (it describes which measurement datapoints can be read/subscribed, which protocol is used and which security requirements must be met) • An AIMC Submodel configures the mapping where the read runtime measurement datapoints of the sensor asset must be mapped within the application-specific OperationData Submodel

1.1.2 Requirements

- Provide the client application an understanding of what asset datapoint is of interest with regards its interface and/or related interface.
- Provide a logical mapping of interested datapoints to intended location in the AAS.
- Provide required information that runtime data can be mapped to a specific place (e.g., to an AAS SME Property) within a SM.

1.1.3 Design Decisions

- Define a reusable mapping representation of interested asset datapoints to specific locations in AAS Submodels.

2 Submodel AIMC

2.1 Approach

The Submodel consists of an AIMC core part (see Figure 2) that specifies the basic structure of the AIMC Submodel.

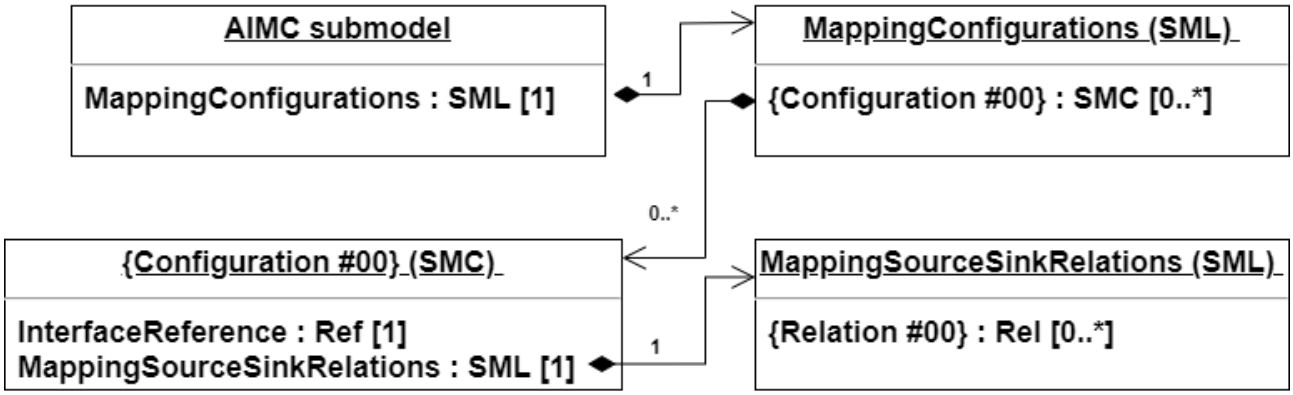

Figure 2: AIMC core structure

2.2 Overview of the AIMC Core Structure

The AIMC Submodel configures the mapping from one or more interfaces in form of SMCs provided in the MappingConfigurations SML. In each interface mapping configuration SMC, information about an interface reference and source sink relations are provided as a reference element and a Submodel element list respectively. The interface reference provides a link to an AID interface SMC of interest and the source sink relations defines the mapping of datapoints described in an AID Submodel to an element of the AAS.

2.3 Elements of the Submodel

For the Submodel, these important attributes need to be set:

Table 2: Elements of the Submodel

idShort:	AssetInterfacesMappingConfiguration		
Class:	Submodel (SM)		
semanticId:	[IRI] https://admin-shell.io/idta/AssetInterfacesMappingConfiguration/1/0/Submodel		
Parent:	An AAS		
Explanation:	Definition of the Asset Interfaces Mapping Configuration Submodel identified by its semanticId. The idShort can be picked freely.		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	
[SML] MappingConfigurations	[IRI] https://admin-shell.io/idta/AssetInterfacesMappingConfiguration/1/0/MappingConfigurations A list of collections each referencing an interface in an AID Submodel and configuring source sink relations.	N/A	1

2.4 Elements of SML “MappingConfigurations”

Table 3: Elements of SML "MappingConfigurations"

idShort:	MappingConfigurations		
Class:	SubmodelElementList (SML)		
semanticId:	[IRI] https://admin-shell.io/idta/AssetInterfacesMappingConfiguration/1/0/MappingConfigurations		
Parent:	AIMC Submodel with respective semanticId.		
Explanation:	A Submodel element list listing collections each referencing an interface in an AID Submodel and configuring source sink relations.		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	
[SMC] {Configuration #00}	[IRI] https://admin-shell.io/idta/AssetInterfacesMappingConfiguration/1/0/MappingConfiguration A collection referencing an interface in an AID Submodel and configuring source sink relations.	N/A	0..*

2.5 Elements of SMC “{Configuration #00}”

Table 4: Elements of SMC “{Configuration #00}”

idShort:	{Configuration #00} Note: The idShort will not be used according to the AAS rule, as no idShort is allowed for children of an SML.		
Class:	SubmodelElementCollection (SMC)		
semanticId:	[IRI] https://admin-shell.io/idta/AssetInterfacesMappingConfiguration/1/0/MappingConfiguration		
Parent:	SML with idShort = MappingConfigurations and a respective semanticId.		
Explanation:	A configuration SMC references an interface in an AID Submodel and configures source sink relations.		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	
[Ref] InterfaceReference	[IRI] https://admin-shell.io/idta/AssetInterfacesMappingConfiguration/1/0/InterfaceReference A reference referencing an interface of interest in an AID.	(Submodel) https://asset.com/assetinterfacesdescription (SubmodelElementCollection) InterfaceHTTP	1
[SML] MappingSourceSinkRelations	[IRI] https://admin-shell.io/idta/AssetInterfacesMappingConfiguration/1/0/MappingSourceSinkRelations A list of collections each configuring mappings from source to sink.	N/A	1

2.6 Element of SML “MappingSourceSinkRelations”

Table 5: Element of SML "MappingSourceSinkRelations"

idShort:	MappingSourceSinkRelations		
Class:	SubmodelElementList (SML)		
semanticId:	[IRI] https://admin-shell.io/idta/AssetInterfacesMappingConfiguration/1/0/ MappingSourceSinkRelations		
Parent:	SMC with idShort = {Configuration #00} and respective semanticId.		
Explanation:	The Submodel element list lists relations each configuring mappings from source to sink.		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	
[Rel] {Relation #00}	[IRI] https://admin-shell.io/idta/AssetInterfacesMappingConfiguration/1/0/ MappingSourceSinkRelation Configuration of a mapping from source to sink. The first reference references the source and the second the sink. Both references reference SMEs in Submodels, one in an AID Submodel.	First Reference: (Submodel) https:/asset.com/assetinterfacesdescription (SubmodelElementCollection) InterfaceHTTP (SubmodelElementCollection) InterfaceMetadata (SubmodelElementCollection) Properties (SubmodelElementCollection) Voltage Second Reference: (Submodel) https://asset.com/status/ (Property) voltage	0..*

Annex A. Explanations on used table formats

1. General

The used tables in this document try to outline information as concise as possible. They do not convey all information on Submodels and SubmodelElements. For this purpose, the definitive definitions are given by a separate file in form of an AASX file of the Submodel template and its elements.

2. Tables on Submodels and SubmodelElements

For clarity and brevity, a set of rules is used for the tables for describing Submodels and SubmodelElements.

- The tables follow in principle the same conventions as in [5].
- The table heads abbreviate 'cardinality' with 'card'.
- The tables often place two information in different rows of the same table cell. In this case, the first information is marked out by sharp brackets [] form the second information. A special case are the semanticIds, which are marked out by the format: (type)(local)[idType]value.
- The types of SubmodelElements are abbreviated:

SME type	SubmodelElement type
Property	Property
MLP	MultiLanguageProperty
Range	Range
File	File
Blob	Blob
Ref	ReferenceElement
Rel	RelationshipElement
SMC	SubmodelElementCollection

- If an idShort ends with '{00}', this indicates a suffix of the respective length (here: 2) of decimal digits, in order to make the idShort unique. A different idShort might be chosen, as long as it is unique in the parent's context.
- The Keys of semanticId in the main section feature only idType and value, such as: [IRI]https://admin-shell.io/vdi/2770/1/0/DocumentId/Id. The attributes "type" and "local" (typically "ConceptDescription" and "(local)" or "GlobalReference" and (no-local)") need to be set accordingly; see [6].
- If a table does not contain a column with "parent" heading, all represented attributes share the same parent. This parent is denoted in the head of the table.
- Multi-language strings are represented by the text value, followed by '@'-character and the ISO 639 language code: example@EN.
- The [valueType] is only given for Properties.

Annex B. AIMC in AASX Package Explorer

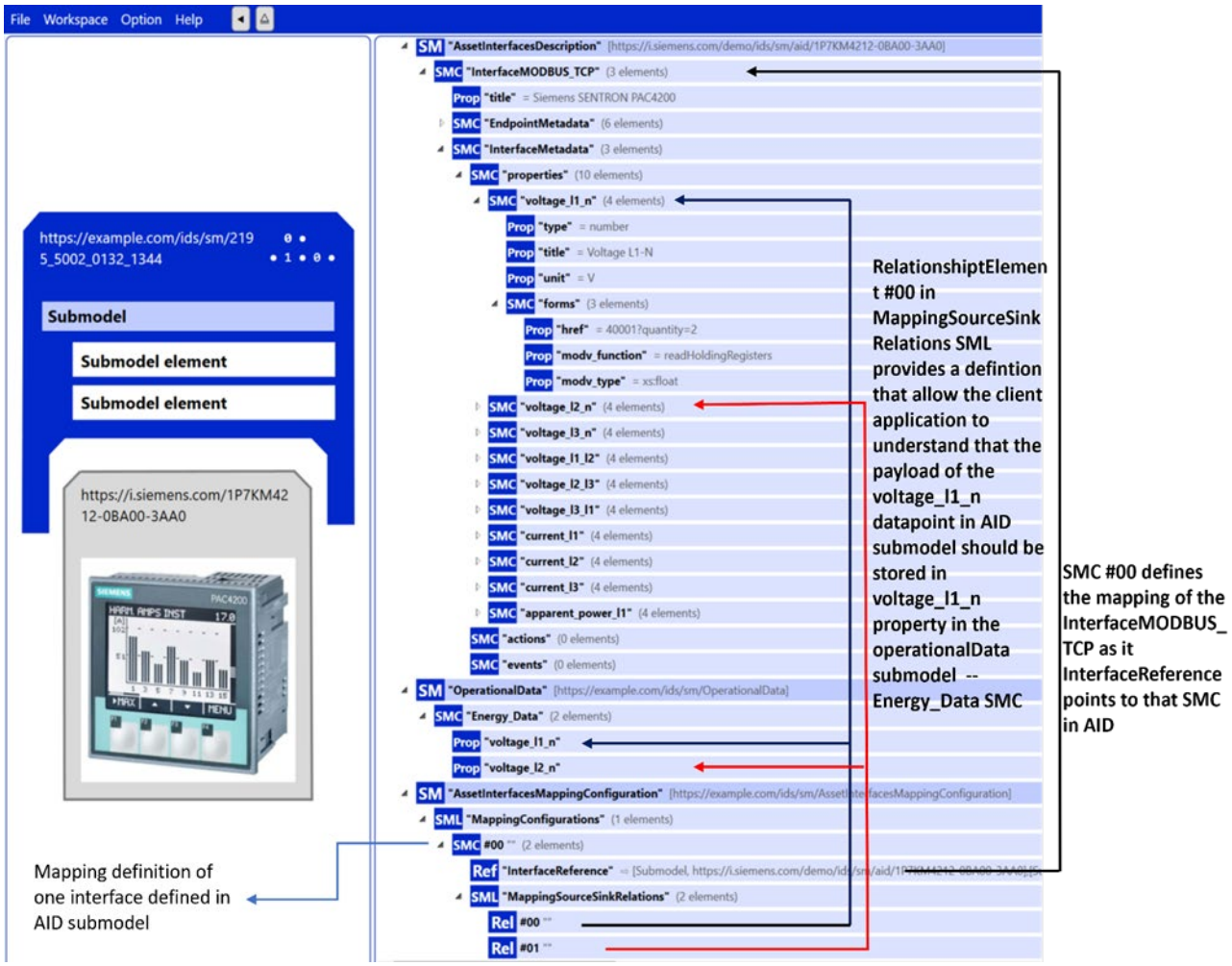

Figure 3: Example depiction of how AID, AIMC and OperationalData (or any other submodel) are used to integrate asset data into AAS

Bibliography

[1] acatech, “Recommendations for implementing the strategic initiative INDUSTRIE 4.0,“ Platform Industrie 4.0, Frankfurt am main, April, 2013. [Online]. Available https://www.acatech.de/Publikation/recommendations-for-implementing-the-strategic-initiative-industrie-4-0-final-report-of-the-industrie-4-0-working-group/

[2] bitkom;VDMA;ZVEI, “Implementation Strategy Industrie 4.0: Report on the results of the Industrie 4.0 Platform,“ VDMA e.V., Frankfurt am Main, January, 2016. [Online]. Available:https://www.bitkom.org/noindex/Publikationen/2016/Sonstiges/Implementation-Strategy-Industrie-40/2016-01-Implementation-Strategy-Industrie40.pdf

[3] Plattform Industrie 4.0;PIANO Industria 4.0, “The Structure of the Administration Shell: TRILATERAL PERSPECTIVES from France, Italy and Germany,“ Plattform Industrie 4.0, Berlin, March, 2018. [Online]. Available: https://www.plattform-i40.de/I40/Redaktion/EN/Downloads/Publikation/hm-2018-trilaterale-coop.html

[4] Industrial Digital Twin Association, “Specification of the Asset Administration Shell Part 1: Metamodel,“ Industrial Digital Twin Association, Frankfurt am Main, March,2023. [Online]. Available: https://industrialdigitaltwin.org/wp-content/uploads/2023/06/IDTA-01001-3-0_SpecificationAssetAdministrationShell_Part1_Metamodel.pdf

[5] ZVEI, “Beispiele zur Verwaltungsschale der Industrie 4.0-Komponente – Basisteil (German),“ ZVEI, Frankfurt am Main, November, 2016. [Online]. Available: https://www.zvei.org/presse-medien/publikationen/beispiele-zur-verwaltungsschale-der-industrie-40-komponente-basisteil/

[6] Plattform Industrie 4.0, “Verwaltungsschale in der Praxis. Wie definiere ich Teilmodelle, beispielhafte Teilmodelle und Interaktion zwischen Verwaltungsschalen (in German),“ Plattform Industrie 4.0, Berlin, Juli,2020. Plattform Industrie 4.0 in Kooperation mit VDE GMA Fachausschuss 7.20, Federal Ministry for Economic Affairs and Energy (BMWi), [Online]. Available: https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/2019-verwaltungsschale-in-der-praxis.html

[7] S. Kaebisch, M. McCool und E. Korkan, “Web of Things(WoT) Thing Description 1.1,“ World Wide Web Consortium, Cambridge, 2023. [Online]. Available: https://www.w3.org/TR/wot-thing-description11/

[8] Modbus Organisation, “MODBUS Messaging on TCP/IP Implementation Guide V1.0b,“ Modbus Organisation, 2006. [Online]. Available: https://www.modbus.org/docs/Modbus_Messaging_Implementation_Guide_V1_0b.pdf

[9] OASIS, “MQTT Version 3.1.1 Plus Errata 01,“ OASIS, December 2015. [Online]. Available: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html

[10] H. Nielsen, J. Mogul, L. M. Masinter, R. T. Fielding, J. Gettys, P. J. Leach und T. Berners-Lee, „RFC 2616 - Hypertext Transfer Protocol -- HTTP/1.1,“ 1999. [Online]. Available:https://datatracker.ietf.org/doc/html/rfc2616

[11] Specification of the Asset Administration Shell; Part 2: Application Programming Interfaces, June 2023, Industrial Digital Twin Association, Available :

https://industrialdigitaltwin.org/wp-content/uploads/2023/06/IDTA-01002-3-0_SpecificationAssetAdministrationShell_Part2_API_.pdf

www.industrialdigitaltwin.org