

Specification 3.0: A big milestone for the growing AAS community

Birgit Boss, Andreas Orzelski, Sebastian Bader

As the Industrial Digital Twin Association just announced, Version 3.0 of the Specification of the Asset Administration Shell is released now. It gives a stable and mature basis for the implementation of interoperable Digital Twins.

The specifications are embedded into the general concepts of the Asset Administration Shell (AAS) and, therefore, presume familiarity with the building blocks of Industry 4.0. The main stakeholders addressed are architects and software developers aiming to implement Digital Twins conformant to the specification of the AAS.

The additionally published Reading Guide (see content hub at the website of IDTA: <https://industrialdigitaltwin.org/content-hub>) enables a fast entry also for non-experienced readers and provides a comprehensive overview of the most important information sources on the AAS. It also gives advice which documents should be read depending on the role of the reader.

The release 3.0 covers the following depicted main aspects – all can be downloaded at the IDTA content hub: <https://industrialdigitaltwin.org/content-hub>

Part 1: Metamodel

The first part of the specification series defines the metamodel of the AAS. The main concepts consist of

- » How to describe assets being represented by the AAS

- » How to describe Submodels with their structures and elements: The Submodels are the subject of data exchange specific to individual use cases.

As part of the specification the following formats are supported for data exchange: XML, JSON and RDF. XML is a good basis for file exchange of Asset Administration Shells (type 1). JSON is mainly motivated through its usage in the HTTP/REST APIs (type 2) as specified in Part 2. RDF enables to build more enhanced querying applications and the presentation in distributed graphs.

The metamodel is also the basis for the specification of so-called Submodel Templates. Submodel Templates define standardized semantics of the data being exchanged between partners in an ecosystem. The Industrial Digital Twin Association (IDTA) provides a growing library of Submodel Templates: <https://industrialdigitaltwin.org/en/content-hub/submodels>.

Part 1 includes both the specification and schema definitions for XML, JSON and RDF (see GitHub: <https://github.com/admin-shell-io/aas-specs>).

Part 2: Application Programming Interfaces

The second part of the specification series specifies generic interfaces as well as the APIs in selected technologies for the AAS and their Submodels. In version 3.0, HTTP/REST APIs are described as the frontrunner. In addition, several profiles further define subsets of features and thereby allow a good balance between expressiveness and complexity.

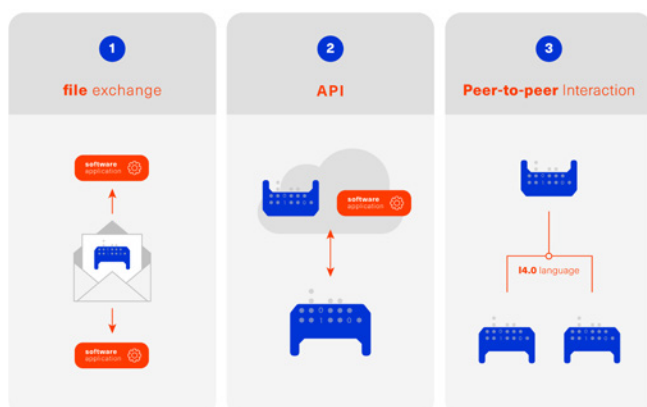


Figure 1: Types of Information Exchange via Asset Administration Shells.

“Interoperability goes operational with version 3 of the AAS. All implementations will be using the same models and the same API.”

Andreas Orzelski, Phoenix Contact

“The specification of the Asset Administration Shell is the basis for any implementation of digital twins aiming for secure cross-company data exchange – in each life cycle of a product.”

Dr. Birgit Boss, Bosch

Besides the basic definitions for:

- » **Asset Administration Shell:** getting the information about an Asset Administration Shell given its ID
- » **Submodel:** getting the whole Submodel data given its ID or getting a part of a Submodel by a nested idShort-path
- » Several other **profiles** cover certain infrastructure **services**
- » Asset Administration Shell **Registry** and Submodel Registry: how to find an Asset Administration Shell and the endpoints of its Submodels via the identifier of an AAS
- » **Discovery:** how to find the ID of an AAS based on other identifying information like the global or the company-specific asset identifiers, for example the serial number
- » Asset Administration Shell **Repository** and Submodel Repository: how to query items in large sets of AAS and Submodels
- » **Concept Description Repository:** how to query for concept descriptions which contain the semantic meaning of Submodels and their elements

Part 2 includes both the specification and OpenAPI definitions for services and profiles (see https://app.swaggerhub.com/organizations/Plattform_i40).

Part 3a: Data Specification – IEC 61360

Part 3 comprises a sub-series of specifications for data specification templates. Part 3a is the first part of this series, specifying data specification templates conformant to IEC 61360.

IEC 61360 explains how to define the semantics of single properties or values. The value range of a property can be defined as a value list – an enumeration –, while each of the (coded) values of the value list are treated as single concepts. They are thus suited to be used as data specifications for concept descriptions. Additional documents will follow in the future.

Part 5: Package File Format (AASX)

This part of the specification series focuses on the exchange file format (type 1) for the transport of information from one partner in the value chain to the next, and further on. The AASX package format leverages the Open Packaging Conventions on how to place content and folders in a ZIP-compressed file and allows the shipment of both the AAS and additional content, e.g., PDFs or pictures.

Note: Part 4 will specify security mechanisms for the Asset Administration Shell and will be published at a later point in time. Part 3, Part 4 and Part 5 were chapters of Part 1 in previous release candidates.

The whole AAS specification is available as an open standard. The schema files (XSD, JSON and RDF) for the defined serializations and OpenAPI specifications are published under open source licenses.

The Asset Administration Shell is also subject of IEC standardization: look for the IEC standard IEC 63278 series.

The YouTube channel (<https://www.youtube.com/@industrial-digitaltwin>) of the IDTA offers many videos and tutorials for the different aspects of the AAS. Just watch out for new tutorials on the different specification parts of V3.0 that guide you step-by-step and explain changes compared to previous releases.

A specification series like the AAS is only possible with a strong and committed community. The home of this community is the IDTA. The second pillar of the AAS is its active open source community giving feedback from practical implementations and collaborating together on SDKs and generic libraries.

Thanks are due to all contributors – either as editors, as OpenAPI developers, programmers of generators for XML, JSON and RDF Schemata, test case creators, reviewers of this version but also to all who already worked with the previous release candidates and gave us feedback how to meet the business requirements. Fast feedback for the Version 3.0 from the community is very welcome.

The Hannover Fair 2023 with already more than 40 exhibits showed that IDTA and its community are on the right way!

Birgit Boss

Robert Bosch GmbH
70469 Stuttgart
birgit.boss@de.bosch.com

Andreas Orzelski

Phoenix Contact
32825 Blomberg
aorzelski@phoenixcontact.com

Sebastian Bader

SAP
85399 München
sebastian.bader@sap.com