ZVEI Discussion Paper

DPP 4.0: An Architecture Proposal for a DPP-System to implement the EU Digital Product Passport for Industrial Products

Kai Garrels (ABB STOTZ-KONTAKT GmbH), Sten Grüner (ABB AG), Andreas Orzelski (Phoenix Contact GmbH & Co. KG), Jochen Reinschmidt (ZVEI)

Version 1.1 (December 2023)

Abstract

The upcoming EU regulation “Ecodesign for Sustainable Products” (ESPR) introduces the concept of the digital product passport, a set of information which accompanies the product throughout its lifecycle.

A large community of manufacturers and users of industrial products has developed an implementation of the digital product passport system, applying the concept of the “asset administration shell” – a universal system for implementing the exchange of asset-related information in the value chain of industrial companies.

The implementation is called “Digital Product Passport 4.0” (DPP4.0) in reference to “Industrie 4.0”.

This paper describes the architecture of this implementation.

For change proposals, please contact
<Jochen.Reinschmidt@zvei.org>

Inhaltsverzeichnis

1 Introduction and Goals

This document describes the basic architecture for the DPP4.0 proposed by ZVEI as a possible implementation for the EU DPP regarding industrial products.

The underlying goals are:

- Enable the collection, and composition of all required regulative data into one information package (the “DPP”).
- Structure the DPP in a modular way to be flexible regarding new requirements for the DPP data (the “DPP Data”).
- Ensure that each data element in the DPP bears a clear semantic definition.
- Ensure data sovereignty of participating industrial companies regarding identifiers, DPP data (data inside a DPP), and holding and providing of the DPP to other actors.
- Enable a decentralized soft infrastructure that is mainly operated by the participating stakeholders, using interoperable implementations that are based on a set of agreements among all actors.
- Ensure a level playing field for data sharing and exchange, reducing dominance of central players, also ensuring low entry barriers for actors.

Essential features of DPP4.0 include:

- product identification based on IEC 61406-x, using company administrated identifiers based on internet URLs
- a modular information modelling approach based on the asset administration shell (AAS) described in IEC 63278-x, allowing a flexible modelling of a DPP based on actual requirements from delegated acts and other industry requirements
- decentralized DPP repositories offering http-REST interfaces (APIs) for DPP access that allow easy integration into existing IT landscapes.

Major quality goals of the DPP4.0 system include:

- an architecture that is easy to understand, to implement, and to maintain
- DPP data that is packed in a clearly defined format, including semantic identification of each information element included
- a flexible approach to access control of DPP data, covering the protection of intellectual property and trade secrets of actors.

1.1 Scenario 1: “Unmodifiable” Product, DPP always at manufacturer

In this document, we are focusing on a DPP scenario for a product which is not modified during the lifecycle of product. The product is either never modified, or only modified (updates originated by the manufacturer, repaired, refurbished to its original state) by the manufacturer.

Consequently, the product manufacturer can always be used as a reliable source for DPP data during the entire product’s lifecycle. Only the product manufacturer will be able to make changes to the DPP data, and the authenticity of his changes can be verified by a digital signature.

For a simplification of language, we will describe the architecture from the perspective of an actor *manufacturer* (see stakeholders):

- the manufacturer, producing a specific product, preparing DPP data for this product and make it accessible
- a user having a need to access the DPP data for the product.

From this perspective, the systems operated by the manufacturer are called *internal systems*, other systems are *external*.

1.2 Scenario 2: “Modifiable” Product, DPP moves with product ownership

An alternative scenario can be implemented for products that can be upgraded, refurbished, repaired, equipped with spare parts etc.

For this scenario, the DPP needs to be “passed on” from one actor to the next, e.g. from a user to a refurbisher. It is necessary to track the ownership of the product to find the current DPP at the current owner of the product, e.g. by updating resolvers.

Only the actor that currently uses, modifies, upgrades the product is allowed to make changes to the actual DPP for the product – the DPP will change “holdership” with changes in ownership of the product.

Alternatively, the actor that modifies the product could affix a new product identifier to the product (e.g. with a new QR code as data carrier), which then links to the updated data.

All the principles and the overall architecture are the same for this scenario, except for the tracking of the current owner of the product.

As the ESPR does not detail requirements on this use case, we are not describing it further.

1.3 Requirements Overview and Quality Goals

The requirements in bold are taken from [1], where you can also find the driving forces for these requirements.

Additional requirements have been identified during the development of the DPP4.0 system (not in bold letters).

	Legal obligations	**L1. Ensure compliance with the Proposal for the new Ecodesign for Sustainable Products Regulation (ESPR)** **L2. Ensure compliance with Extended Producer Responsibility (EPR) and EU government legislation “right to repair”** **L3. Ensure compliance with the General Data Protection Regulation (GDPR)**
	Functional suitability	**F1. Need to fit the respective sector, industry, and use case** **F2. Allow actors to make statements exclusively for the information for which they are responsible** **F3. Allow decentralized data storage locations for the DPP information** **F4. Enable the decentralized collection of the information required for a DPP**
	Security, confidentiality, and IP protection	**S1. Ensure nonrepudiation** **S2. Enable data verification** (confirm the authenticity of the data) **S3. Ensure data sovereignty** (responsible DPP actor controls access to the data) **S4. Ensure secure data storage for a specified time** (data securely stored and protected from unauthorized access) **S5.** Logging and audit trail must be implemented to identify cybersecurity incidents. S6. Ensure no intellectual property or trade secrets are exposed by the DPP data. DPP data for a product only includes the aggregate information for the delivered product. DPP data does not contain detailed information for all its components unless this is mutually agreed between actors. S7. Access to DPP data shall be provided on a “need to know” basis and on request of the actor, so no DPPs will be placed in central repositories for good. S8. DPP data must only be modifiable by the actor currently responsible for the DPP data, and not by any other party.

		S9. DPP data changes need to be traced by the actor responsible for the changes in DPP data.
	Interoperability	**I1. Provide clear semantics** **I2.** Semantics are implemented with IEC 61360 (IEC CDD or ECLASS) concept descriptions and dictionaries **I3. Standardize data schemas describing the products** **I4. Provide an application interface (API) for data provision and data request**
	Modularity and modifiability	**M1. Ensure flexibility to add/edit/remove actors, products, or product attributes** **M2. Ensure readiness for broader, international use**
	Accessibility	**A1. Allow the determination and implementation of access rules based on the 'need-to-know' principle** **A2. Ensure participation opportunities for actors who do not have their own information system**
	Availability and time behavior	**A3. Ensure appropriate availability of the DPP information (**data is available when needed, depends on use case, 2.5s for a web representation should be sufficient)
	Identifiers and Portability	**P1. Ensure that product identifiers and the DPP information** (DPP data/DPP data) **are transferable from one software system to another,** that means interoperability between systems. **P2. Ensure that product identifiers are referenceable and harmonizable through-out the entire EU** P3. Identifiers should be unique, specific to a product (including intermediates) P4. Identifiers should be persistent (the info should remain available even if the company change name, web address, goes bankrupt etc.) P5. Identifiers shall also be administrated by a decentral system, avoid a breakdown of the uniqueness of all identifiers if a registrar goes out of business.

1.4 Stakeholders

1.4.1 Actors

Stakeholders are actively participating in the data exchange around the DPP are called *actors.*

- manufacturers: producing industrial products, and providing DPP data
- suppliers: delivering components to manufacturers, and DPP relevant data for these components
- users: users of industrial products, also consuming DPP information
- sales channels, like distributors or wholesalers
- recyclers: using DPP information to optimize the recycling of products
- regulator: national authorities (e.g. market surveillance, customs) having the right to access DPP data

Table of Actors:

Role/Name	Expectations
Manufacturer	• clear description how to set up the product passport and how to make it available • protection of trade secrets and intellectual property regarding his product • data sovereignty about their DPP data
Supplier	• has an agreement with manufacturer how to deliver component information

Role/Name	Expectations
	• protection of trade secrets and intellectual property regarding his component
User	• easy identification of product by reading the data carrier • easy access to DPP data • access to DPP data before they are bound by a contract
Sales channel	• easy access for DPP data *without physical access* to the product, e.g. by using a list of product identifiers • right to pass on DPP data to other actors, e.g. users
Recycler	• easy identification of product by reading the data carrier • easy access to DPP data • relevant DPP data regarding the recycling of the product
Regulator	• access to DPP data • access to central registry of product IDs • right and possibility to verify DPP data with registry data

1.4.2 Community

An additional group of stakeholders is the group of people and organizations involved in defining the game rules for the DPP system and DPP data. They will not necessarily be participating in the data exchange later.

We call this group *community*, examples would be the EU legislator, involved industry associations like ORGALIM or the ZVEI, or the CIRPASS project.

In the end, the *community* defines the setup and rules for the DPP system, and the *actors* will be using it.

2 System Scope and Context

The scope of the DPP4.0 system includes:

- the method used to identify the products (data carrier and identifier)
- the actors that are involved in the data exchange
- the identity and access management system allowing actors to trustfully identify each other, restricting, or allowing access to DPP data, and allowing to issue cryptographic certificates for signing or encrypting DPP data
- the data model that is used for structuring DPP data
- the data storage and data provisioning by DPP repositories operated by the actors (or on their behalf by a 3rd party)
- application programming interfaces (API), protocols and data formats to access the DPP repositories
- resolvers to find the endpoint to access the DPP data
- the EU central registry of identifiers as required the ESPR
- the interactions and information flows that are used to exchange DPP data.

2.1 Business Context

Scenario 1: “Unmodifiable” Product, DPP always at manufacturer

If a valid DPP data is available at the manufacturer, the information flow is the following:

- The manufacturer receives DPP-relevant data for all DPP-relevant components of his product from his suppliers. These may or may not be in DPP format.
- From this information, and own internal information, the manufacturer sets up the DPP data, and makes it available to authorized actors.
- The manufacturer uploads the product identifiers and the required mandatory attributes into the central EU central registry - or updates them in case of modifications.
- All other actors can access the publicly available DPP data at the manufacturer. It is their choice whether they want to keep a copy of the DPP data, or keep a reference, or only display/process DPP data for a certain use.

Scenario 2: “Modifiable” Product, DPP moves with product ownership

If the product is expected to be modified by any of the stakeholders which is in possession of the product, the DPP cannot be accessed at the manufacturer, but needs to be accessed at the stakeholder which currently has the product in possession, and possibly has modified the product and DPP.

In this case, the product identifier on the product cannot directly be used to access the DPP but is still valid to uniquely identify the product.

The resolver at the manufacturer (see 3.5), when queried for the unique product ID, needs to answer that the product is modifiable, not in his possession anymore, and the DPP cannot be accessed here.

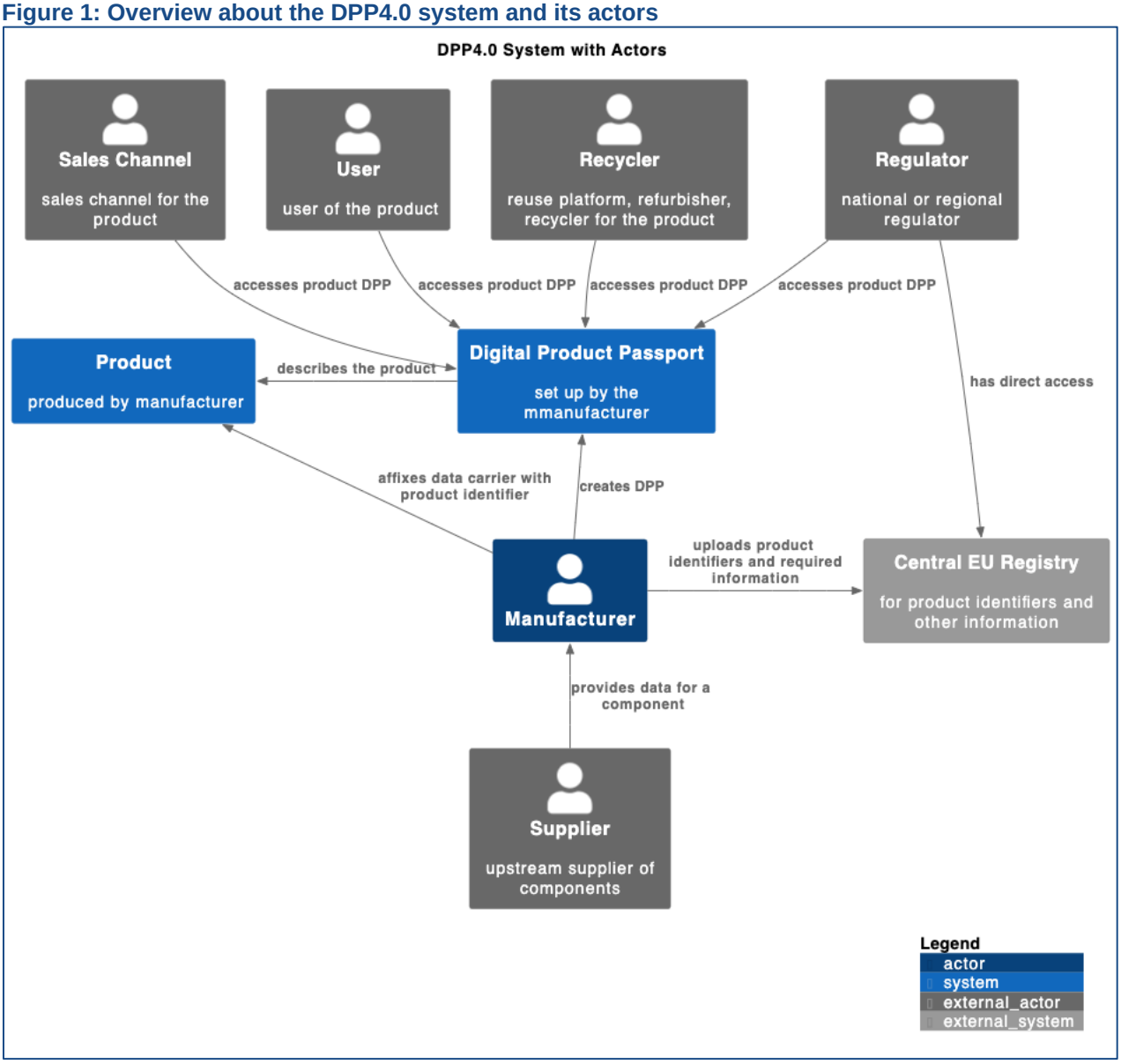
The new stakeholder must not be revealed to unauthorized parties, so by default, the request ends here.

Parties with a justified interest to access the DPP at the new owner (e.g. a regulator) can ask the manufacturer for the new owner, and the new owner will be able to provide the updated DPP for the modified product.

These requests may have to be repeated until the current owner of the product, holding the last updated DPP is found.

Summarizing the two scenarios: the original DPP from the original manufacturer describes the product in its “as-delivered state”; if a stakeholder modifies the product later on, he has to update the DPP and make it available to parties that need the information.

Figure 1: Overview about the DPP4.0 system and its actors

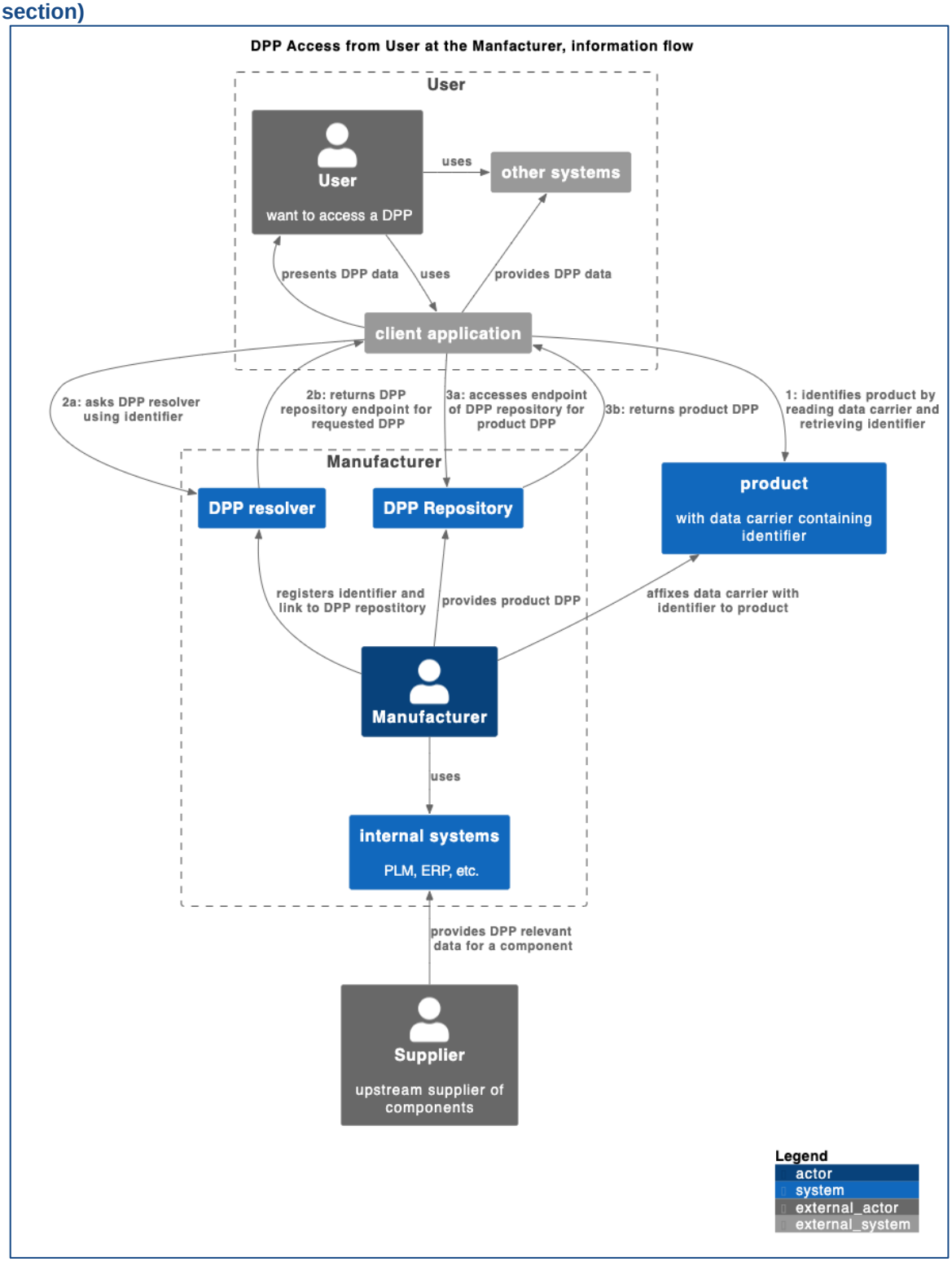


2.2 Technical Context

To access DPP data at the manufacturer, the user must follow these steps:

- Identify the product by reading its data carrier and thus receiving the product identifier. IEC 61406- describes 2D-Codes (e.g. QR codes, data matrix codes) and RFID as data carriers.
- Authenticate with the authentication service to receive an access token that must be used to get access to DPP resolver and DPP repository. This step can be skipped if anonymous access to public data is sufficient.
- The product identifier is used to contact a resolver at the manufacturer, which is returning the final address (endpoint) of the DPP in the DPP repository.
  De facto, using IEC 61406-x identifiers, the product identifier contains the address of the resolver in the host part of the identifier.
- The DPP repository finally delivers the DPP data of the requested product.

Figure 2: Data flow for DPP access by a user at the manufacturer (authorization explained in a later section)



3 Solution Strategy

3.1 Actors that are involved in the data exchange

The actors that are active in a DPP4.0 system have already been outlined in chapter stakeholders.

To meet the requirements on data security, authentication of actors is required before they access system components of the DPP4.0 system.

It may be useful to assign an “actor role” to actors, for example to identify recyclers or regulators as special actors. This can be implemented by adding an *actor role* attribute to an actor.

3.2 Method used to identify the products: data carrier and product identifier

The DPP4.0 system uses URIs (universal resource identifiers, RFC 3986) to identify products, more precisely it uses URLs (RFC 1738), a specific form of URIs that simplifies the resolution of the identifier to the final source of DPP data.

Manufacturers use a domain name that they own, and codify product identification (on *model, batch,* or *item* level) under their own responsibility.

This choice has several advantages:

- the identifiers can be assigned without a central registration authority; except for the manufacturer domain name which must be registered in the DNS system
- no incremental cost for each created identifier is incurred.

Identifiers can be implemented on a granularity level that is useful for the product under consideration: either on item level for products that bear an individual serial number, or on product model or batch level where serialization is not applied.

As data carrier to be affixed on the product, DPP4.0 uses 2D codes (QR or data matrix), alternatively RFID of NFC can be used. These data carriers can be read with common reading devices, and with modern smart phones.

IEC 61406-x specify the relevant principles and restrictions for the identifier and the data carrier.

3.3 Client-Server Model

The DPP4.0 system is using a client-server model as distributed application structure.

Actors who provide access to data operate servers:
- DPP repositories act as servers for DPP data
- DPP resolvers act as servers for finding DPP repository endpoints, e.g. based on a product identifier.

Actors who want to access *DPP data* are using *client applications*.

3.4 DPP repositories operated by the actors

The DPP4.0 system implements storage of the DPP data in DPP repositories operated by the actors (e.g., the manufacturer of a product). The repositories can also be operated by a 3rd party (e.g., a cloud service provider) on behalf of the actor.

The repository holds (stores) the DPP data for each defined product, depending on the defined granularity level of the identifier. Each individual product identifier is linked to exactly one DPP data set.

The repository offers an *application programming interface* (API) that can be accessed from other actors in the DPP4.0 system after authentication. The API can be accessed via a defined *endpoint* belonging to the DPP

repository. The DPP data related to one product identifier can be accessed as a complete DPP data set (including all DPP data), or partially on submodel or submodel element level (see section Data Model).

The repository also verifies actor's authentication and enforces access authorization (see below).

The repository is filled with DPP data for each DPP data set from internal systems holding the relevant data at the actor. It can also be connected to internal systems (e.g., PLM systems, ERP systems) for automatic data provisioning on demand.

3.5 DPP Resolvers to find DPP repositories

To find the endpoint for a given identifier, a *DPP resolver* is used. When asked for a product identifier, the resolver returns the DPP corresponding repository endpoint. After that, accessing the repository endpoint with the product identifier as a parameter, the DPP data can be accessed.

The DPP resolver offers an application programming interface (API) that can be accessed from other actors in the DPP4.0 system after authentication.

In the simplest case, the URL identifying the product is directly accessed via the https protocol, adding an **Accept** header in the http request indicating that DPP data is requested.

If that **Accept** header is omitted, or a general HTML header is provided, the repository may respond with a human-readable web page related to the identifier.

3.6 Decentral Resolvers operated by Product Manufacturers

In the DPP4.0 system the unique product identifier is a unique URL in a web domain of the manufacturer. This has a couple of advantages:

- The manufacturer can stay responsible for maintaining the correct assignments between the unique product identifier, the corresponding DPP repository, and finally the DPP belonging to the product. No 3rd party needs to be involved or informed in this process.
- Queries to find a DPP for a unique product ID are handled by the manufacturers resolver; as no 3rd party resolvers are involved, nobody can capture statistics on DPP queries and thus nobody can derive information on market share etc. indirectly by monitoring DPP queries.
- IT resources to operate resolvers are managed by the manufacturers, if a product creates a lot of load for resolvers, these are correctly burdened to the responsible manufacturer.

As described in IEC 61406-x and RFC, the unique product identifier ("identification link" in IEC 61406-x) is a URL, consisting of

- an optional schema "https://"
- an internet host name: "id.example.com"
- and the rest of the URL (path, queries and fragments): "/model=123nn2&SN=54123214"

The resulting unique identification URL is "https://id.example.com/model=123nn2&SN=54123214".

To resolve this unique product ID, to a zipped DPP, the following steps need to be followed

- send a https request for "https://id.example.com/model=123nn2&SN=54123214" to the host "id.example.com" (the resolver)
- the resolver answers with the final URL of the DPP repository, e.g. "dpp.example.com/model=123nn2&SN=54123214".
- accessing this URL with "Accept: application/dpp+zip" will finally return the DPP in a zipped format.

3.7 Shared Resolvers

Product identifiers can also be resolved by other registries, if the relation between product identifier and DPP repository endpoint is registered there.

For example, in the ZVEI showcase PCF@Control Cabinet, only one shared DPP resolver ("ZVEI DPP AAS registry") is used in order to reduce the implementation effort for participating companies by using one shared resolver.

3.8 Central EU registry of identifiers as required by the ESPR

As the ESPR states, actors must upload identifiers and other mandatory information into the central registry operated by the European Union.

Details for this are not yet specified in the ESPR.

From the perspective of the DPP4.0 architecture, it would be sufficient to register manufacturer's resolvers in the EU registry and leave the registration of each individual identifier to the manufacturers.

3.9 Identity and access management

3.9.1 Actor Authentication

To ensure a trusted data exchange between actors, they must agree on a common way of authentication.

In the DPP4.0 system, all actors agree to trust one common trust anchor, e.g. a trust service provider listed by EU in the scope of the eIDAS regulation.

This trust service provider issues X.509 certificates to all actors, which they can use as credentials to authenticate.

Authentication in the DPP4.0 system is implemented by using *OpenID Connect*. Actors need to authenticate with an OpenID Connect server, using their X.509 certificate[1].

In return, they receive an *access token* that they will present when accessing the DPP resolver or DPP repository. DPP4.0 uses *JSON web tokens (JWT)* as access tokens.

OpenID authentication services are available on the market today. A list of accepted authentication providers needs to be maintained, these cross-certify each other.

In the DPP4.0 system, access is also possible without authentication – in this case only publicly available DPP data is accessible.

As most of the information is expected to be publicly available information (e.g. manufacturer address, instructions and safety information, RoHS and REACH information etc.), an anonymous access without authentication is foreseen.

3.9.2 Access Rights and Access Authorization

Users may have different information requirements; for example recyclers need access to the recycling instructions. Therefore, different parts of the DPP data might be restricted to certain actors (or actor roles).

The ESPR article 10 refers to the access permissions of an actor as *access rights*.

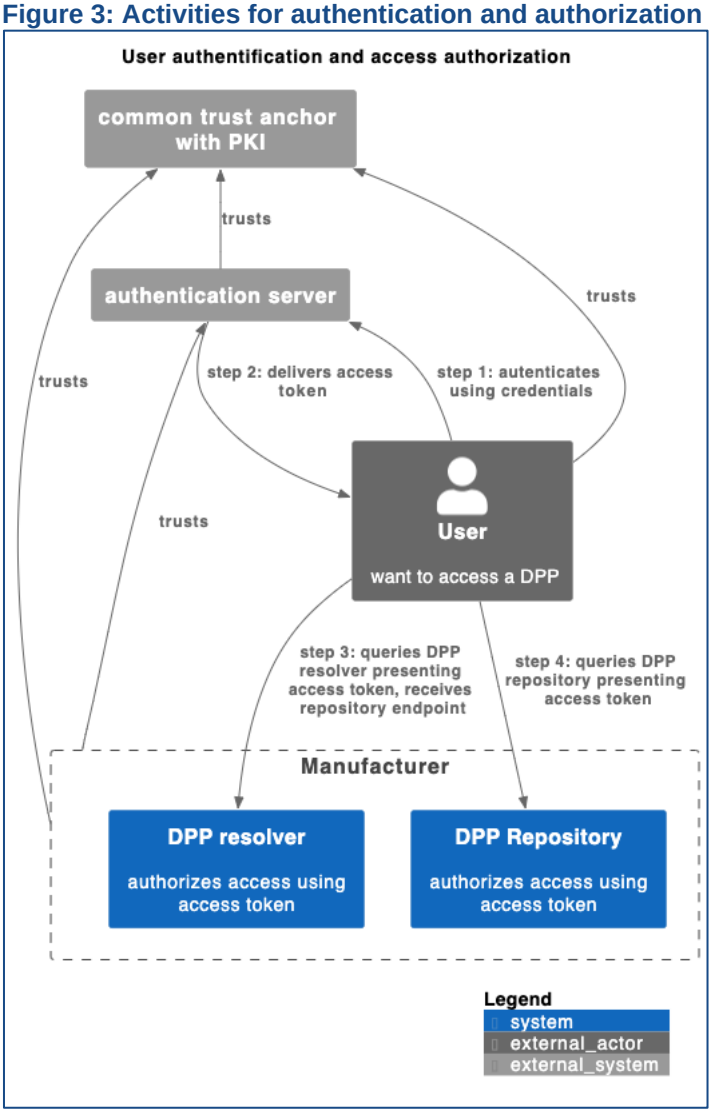
By verifying the presented access token, both DPP resolver and DPP repository can decide which parts of the DPP data is visible or can be accessed by the authenticated actor.

The enforcement of access authorization is completely in the hands of the actor that holds the DPP, thus allowing data sovereignty for the DPP holder.

Figure 3 show the activity flow for authorization, the same flow applies for other accesses, e.g. when the manufacturer registers product identifiers in the central EU registry.

[1] In the DPP4.0 showcase, we also confirmed the suitability of verifiable credentials; as X.509 certificates are widely used today for other applications, we decided to use them.

Figure 3: Activities for authentication and authorization



3.9.3 Certificates for signing DPP data

Actors have received X.509 certificates for each actor that allow signing of DPP data.

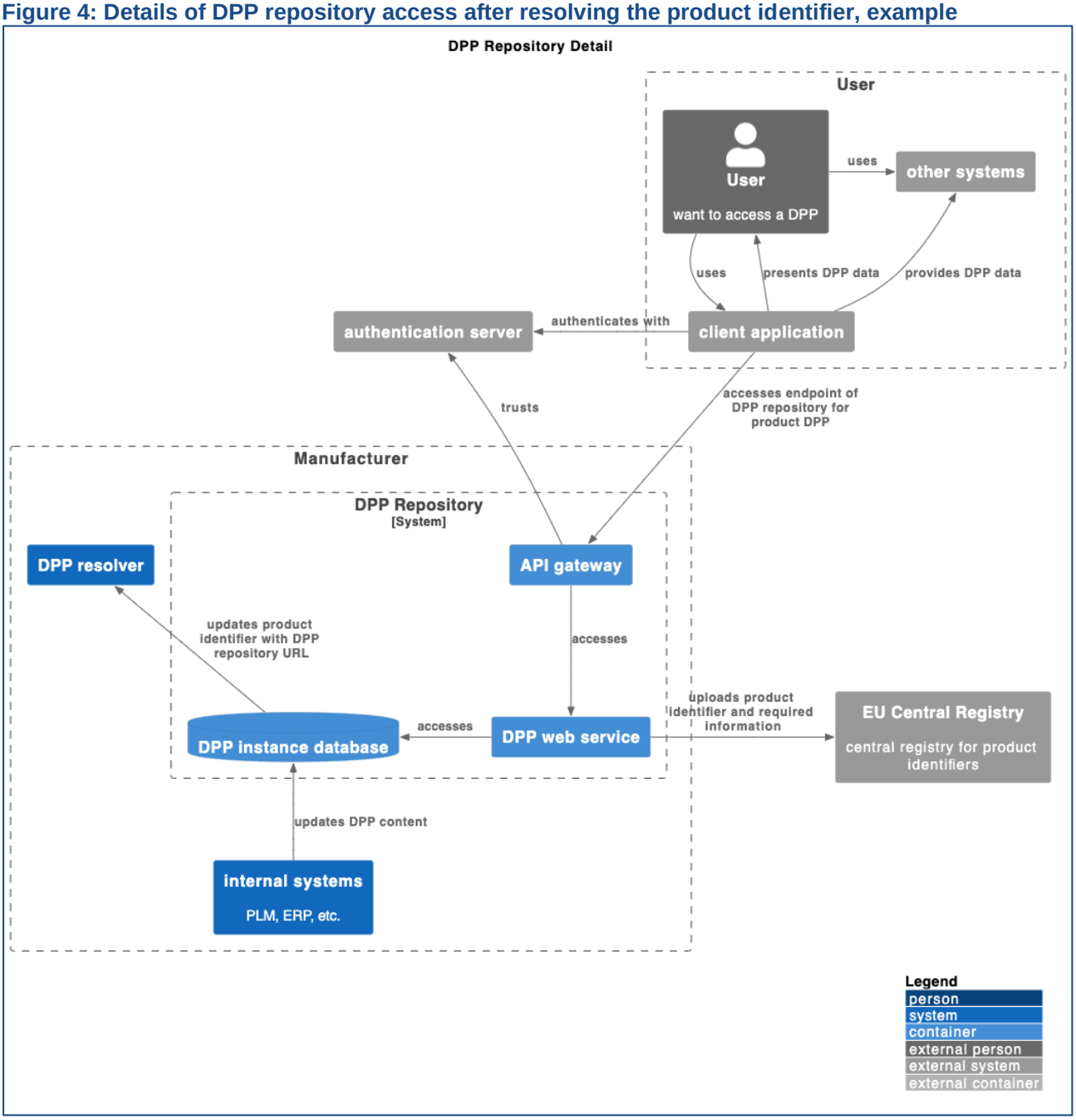
Signing allows the verification of integrity and authenticity of DPP data; thus it can be proven that the DPP data was originally provided by the actor who signed, and that it has not been modified.

4 Building Block View

Figure 1 has already shown the setup of actors in the DPP4.0 system, and Figure 2 gave an overview about the main involved system components.

Figure 4 give an overview about the main building block at the user and the manufacturer as a simplified example.

Figure 4: Details of DPP repository access after resolving the product identifier, example



4.1 Authentication Server

The authentication server identifies the user (e.g. by asking for registered credentials) and grants an access token that user's client application uses to access DPP resolver and DPP repository.

It is implemented using the OpenID Connect standard.

4.2 Building Blocks at the User

The user wants to access DPP data based on product identifiers, either for viewing it, or for further processing in other systems that he is using.

The client application must at least implement the following functions:

- read the data carrier of one or more products and extract their identifiers,
  alternatively: accept a list of product identifiers from the user
- authenticate the user at the authentication service, in return receive an access token. If only public DPP data shall be accessed, this step can be skipped.
- use the DPP resolver to resolve the product identifiers to endpoints of DPP repositories, presenting the access token to the DPP resolver
- access the DPP repositories with the identifier as parameter to access the DPP data, presenting the access token to the DPP repository
- display the DPP data to the user, or forward it to other systems the user may have for further processing.

4.3 Building Blocks at the Manufacturer

Name	**Responsibility**
internal systems	• Gather DPP relevant data from suppliers or internal processes, allowing the complete DPP data to be provided to the DPP repository
DPP repository	• handle all tasks related to processing, preparing, storing, and providing access to DPP data • implement logging and audit trail for the DPP repository
API gateway	• maintain a trust relationship with the authentication service • accept client access • authenticate the client and authorize access of the client application by verifying the validity of the access token presented by client with DPP4.0 authentication server • forward request to DPP web service
DPP web service	• enforce access authorization to different parts of the DPP data based on presented access token • deliver DPP data when http request header is “Accept: application/DPP40”, deliver web page otherwise • synchronize DPP instance data with internal systems • upload or update the EU central registry with required information • updates the DPP resolver to link product identifier with DPP repository URL
DPP instance database	• database holding DPP data for each product identifier
DPP resolver	• Provide endpoint of DPP repository for requested product identifier • implement logging and audit trail for the DPP repository

4.4 Central EU Registry

The central EU registry must be updated with the product identifier and further required information as regulated in the ESPR.

In the DPP4.0 system, the registry is also used to ensure persistence of product identifiers is the responsible actor ceases operation.

Actors must register their product identifiers at the registry together with at least the *economic operator identifier*, plus other required information.

Any access to the registry must be authenticated and authorized to avoid incorrect registrations.

As the registry must "[…] allow for the verification of the authenticity of the product passport", it must verify that a product identifier is either not yet registered, or updated by the economic operator that originally registered it.

To prove the authenticity of registered product identifiers, the identification of the manufacturer must also be stored in the registry, e.g. in the form of its company address, VAT-identification, or similar.

If an actor ceases to operate and gives up the internet domain it has used for IEC 61406-x identification links, the product identifier is still registered in the central registry and cannot be re-registered by a potential new owner of that internet domain.

The authenticity of DPP data belonging to a product identifier can be verified by checking the manufacturer identification in the registry with the one in the DPP data.

4.5 Required Services

From the interactions between different components of the DPP4.0 system, required services can be derived. The following table lists the necessary services and maps them to the existing AAS-API operations[2], which need to be standardized in IEC 63278-5 in the future.

Described are operations that are external to the manufacturer, e.g. the registration of a DPP endpoint into the manufacturers DPP resolver is not described and may vary with different software integrations inside the manufacturer.

Interaction	Description	Involved Components	interface operations
User authentication	User's client application authenticates with the authentication server and receives an access token in return	• User's client application • Authentication server	t.b.d, Identity and Access Management details not yet defined in ESPR
Resolve product's unique identifier to endpoint for DPP access	Client application requests DPP resolver at the manufacturer to find the endpoints for the related DPP. The client needs to present its access token to the resolver, otherwise anonymous access is assumed. Resolving is done in two steps inside the resolver: 1. Find AAS-ID for given asset-ID 2. Find endpoint of AAS-ID	• Users' client application • Manufacturer's DPP resolver	AAS-API: GetAllAssetAdministrationShellIdsByAssetLink GetAssetAdministrationShellDescriptorById
Retrieve DPP data	Client application requests DPP data from endpoint, presenting its access token.	• Users' client application • Manufacturer's DPP repository	AAS-API: GetAssetAdministrationShellById
Register product-ID at EU Central Registry	Manufacturer's DPP web service registers or updates a product-ID with the EU Central Registry, and provides required information about the product	• Manufacturer's DPP web service	t.b.d. services of EU Central Registry are not yet defined in ESPR

4.6 Dataspace Integration

It is possible to integrate the DPP4.0 system into other dataspaces, for example Catena-X.

[2] AAS-API operations as described in:
https://industrialdigitaltwin.org/content-hub/aasspecifications/idta_01002-3-0_application_programming_interfaces

The following requirements must be met by dataspace connectors that want to access DPP4.0 data:

- they need to present an access token which is accepted by the DPP4.0 resolvers and repositories; it does not matter if the authentication process requires different steps as for example in Catena-X
- they need to be able to handle the AAS API for resolvers and repositories
- they need to transform the DPP4.0 data model (AAS with submodels, community-agreed submodel templates) into whatever data model they are using; Catena-X is already using the AAS data model.

5 Deployment and Operation View

The following principles for deployment should be applied in the DPP4.0 system:

- Actors operate their own resolvers and repositories or ask a 3rd party to operate them on their behalf.
- Authentication services are operated by authentication service providers. Which are already operating on the market. All accepted authentication service providers need to cross-certify each other.
- The EU central registry is operated by the European Union.
- Additional resolvers may be operated on behalf of individual actors or consortia of actors. It must then be clarified how these resolvers and registries are filled with product identifiers and other needed information. Usage of additional resolvers should be restricted to a minimum for specified use cases to avoid network traffic.

6 Cross Cutting Concepts

6.1 DPP4.0 Data Model (Information Meta-Model)

The DPP data in the DPP4.0 system is structured along the principles of the *asset administration shell (AAS)*, described in IEC 63278-x[3].

AAS submodels are used to implement different DPP sections, (e.g. about product durability and reliability, reparability, maintenance, presence of substances of concern etc.), and AAS submodel elements are used to implement the individual data elements (e.g. properties) inside each DPP section.

The community will agree on fixed content for the different sections, DPP4.0 uses *submodel templates* to document these agreements.

AAS submodels can by cryptographically signed to ensure data authenticity (confirm the source of the data), reliability and integrity (detection of incidental or intentional modifications).

An appendix below introduces the concepts of the asset administration shell.

6.2 Semantics for Submodels and Submodel Elements

Concept dictionaries in line with IEC 61360 (IEC CDD or ECLASS) are being used to clarify the semantic meaning of submodels and submodel elements.

Each submodel and submodel element is tagged with the IRDI referring to the corresponding CLASS concept according to IEC 61360 (IEC CDD, or also ECLASS).

6.3 Exchange Protocols and Formats

For the data exchange between actors, a client-server architecture is used.

Client applications query DPP resolvers and DPP repositories, these act as servers. The client applications take the active role in this architecture.

The APIs for both DPP repositories and DPP resolvers are described in https://app.swaggerhub.com/organizations/Plattform_i40.

All communication transactions shall are stateless.

HTTPS is being used as a secure communication protocol.

JSON is being used for HTTP payloads in queries and responses.

XML may be used for exchanging DPP data sets for one or more specific DPPs, using the AASX format to package the information for one or several DPPs into a single file.

6.4 Cybersecurity

The DPP4.0 system should be implemented along IEC 62443 Series and other relevant standards.

As most information in DPP data is public and does not contain trade secrets or IP to be protected, the target security level shall be “Security Level 2: Protection against intentional misuse by simple means with few resources, general skills and low motivation.”

[3] The current detailed description of the AAS information meta-model can be found in the specification “Details of the Asset Administration Shell - The exchange of information between partners in the value chain of Industrie 4.0 ” published by the *IDTA e.V.* and *Plattform Industrie 4.0.*

7 Risks and Technical Debt

7.1 Performance

- ESPR article 10 specifies reliability as a requirement; as each actor is responsible for the reliability of his own systems, a total DPP4.0 system reliability cannot be assured.

7.2 Cybersecurity Risks

- Setting up access permissions for thousands of actors will be extremely difficult, especially when actors have different roles (e.g. a switch gear cubicle manufacturer might also want to act as a switchgear recycler, he then has two roles).
  Only using “actor roles” as identities (e.g. one identity for all recyclers or all regulatory authorities) is not possible, as this will leads to hundreds of actors sharing one identity.
  Regulatory authorities as actors include hundreds of customs offices.
  We recommend limiting the DPP data to data that can be public – then an anonymous access is sufficient.
- DPP resolvers and repositories will be exposed to the public internet with many actors accessing it; it seems that many companies will prefer to separate them from systems with business-critical content like customer data etc.
- Life cycle management of signature systems (along decades) is a considerable challenge in praxis.
- The usage of open-source SDKs and reference implementations is recommended; these can be reviewed to confirm cybersecurity requirements.

7.3 Central EU Registry

Concerns around the central EU registry need to be addressed when details are known:

- system performance with respect to registering product IDs
- prevent to draw conclusions on market share and no of items sold based on registry access
- efforts and cost of storing information for millions of registered products, and must cover the whole product life cycle along the circular economy, which may last several decades.

7.4 EU Backup Storage

A central backup storage is still under discussion in the ESPR trilogues – and therefore at the moment not addressed n this document.

However, this implies that massive amounts of data need to be stored. As an example, we know from an industrial company in our sector that they operate a repository with data sets for >120Mio products.

7.5 Expansion of regulated Data

In principle the data model used in the DPP4.0 is flexible enough to handle new requirements from delegated acts.

On the other hand, assuming many DPP data sets created, it will produce massive amounts of data to be handled.

8 Glossary

Multiple entries in the “Term” column are synonyms.

Term	Definition
digital product passport, DPP	a set of data specific to a product that includes the information specified in the ESPR or related delegated acts
DPP system	the technical system that is used to implement the DPP concepts described in the ESPR
DPP data	The information which is contained in a digital product passport and that is exchanged between actors.
DPP data set	One specific set of DPP data related to one specific product bearing one specific product identifier
DPP4.0 system	A system concept for implementing the DPP system by means of the asset administration shell concept
product	any physical good that is placed on the market or put into service
component	a product intended to be incorporated into another product
data carrier	a linear bar code symbol, a two-dimensional symbol or other automatic identification data capture medium that can be read by a device
(unique) product identifier	means a unique string of characters for the identification of products that also enables a web link to the product passport. Can be applied on model, batch or item level.
actor	a person or organization that needs to exchange DPP data
user	an actor (person or company) that is using a product and wants access to the DPP
manufacturer	An actor (most likely a company) that produces a product and is obliged to provide a DPP
actor group, actor role	A group of actors having the same need to access DPP data, e.g. all recyclers
DPP community	All persons or organizations agreeing on the game rules for the DPP system
Resolver	A software service to find the repository endpoints for a given product identifier
Repository	A software service that makes DPP data accessible
central EU Registry	“The registry” as described in the ESPR text
application programming interface, API	An application programming interface (API) is a way for two or more computer programs to communicate with each other. It is a type of software interface, offering a service to other pieces of software. (Source: Wikipedia)
endpoint, communication endpoint	The URL of a DPP repository, offering AAS API
access token	an **access token** contains the security credentials for a login session and identifies a user and potentially his client application
Asset administration shell, AAS	Concept for exchanging asset related information, standardized in IEC 63278 series
(AAS) submodel	A collection of data elements the functionally belong together
(AAS) submodel element	An individual data element in a submodel
(AAS) submodel template	A community-agreed collection of submodel elements

9 Appendix: Normative References

Topic	Applied standard	Definition
Identity and Access Management	OpenID Connect	OpenID Connect 1.0 is a simple identity layer on top of the OAuth 2.0 protocol. It allows Clients to verify the identity of the End-User based on the authentication performed by an Authorization Server, as well as to obtain basic profile information about the End-User in an interoperable and REST-like manner. Source: https://openid.net/connect/
	JSON web token	JSON Web Tokens are an open, industry standard RFC 7519 method for representing claims securely between two parties. https://jwt.io/
	X.509	Public Key Infrastructure and authorization credentials
Data Models	IEC 63278 Asset administration shell	Part 1: Administration Shell Structure Part 2: Information Meta-Model Part 3: Security Provisions for Asset Administration Shells Part 4: Use cases and modelling examples Part 5: Interfaces
Semantics	IEC 61360, IEC CDD, ECLASS	Semantic concept dictionaries for submodels and AAS submodel elements
Data Carriers and unique identifiers	IEC 61406 Identification Link	Part 1: General Requirements Part 2: Types/Models, Lots/Batches, Items and Characteristics
Protocols and Data Formats	https with JSON payloads	Online access through REST APIs
	Open Packaging Conventions ECMA-376, ISO/IEC 29500-2	AASX file format for file transfer of one or more AAS instance data sets
Cybersecurity	IEC 62443	Industrial communication networks – Network and system security – Part 3-3: System security requirements and security levels

10 Appendix: An Introduction to the Asset Administration Shell

10.1 Asset Administration Shell

The asset administration shell (AAS) information model allows describing any information that is related to one specific asset, plus

- the serializations needed for exchanging the asset related data
- a package format to pack data from several products into one package
- an application programmer interface to allow online access to AAS data via REST interfaces
- rules for authentication of actors and enforcement of access control.

This document only describes the basic concepts, detailed specifications are referenced in this text.

10.2 Submodels

The AAS organizes information into groups called *submodels*. Submodels collect data elements that belong together for a functional reason, e.g. for one specific use case. They could be seen as “sections” of a digital product passport.

To define a semantically unambiguous submodel that everybody can understand, the community standardizes *submodel templates* which specify exactly which data elements must be present in an agreed submodel, e.g. for a product nameplate, or a declaration of regulated substances.

Relevant submodels can be included based on the requirements of the ESPR (or its delegated acts) in an AAS, allowing the required modularity if AAS information.

The ESPR does not yet outline in detail which “sections” a DPP needs to have, and which data elements must be present, thus the DPP4.0 system is using AAS information model as a flexible and generic approach.

10.3 Submodel Elements

Submodels contain *submodel elements*, which are the primary data elements in the AAS. A submodel element could be a property, a string, a reference to a file etc.

Submodel elements are described by their attributes, a few we want to highlight here:

- a *reference* that allows it to be referenced in an AAS
- a *semantic identifier* that specifies its meaning using IEC 61360 (IEC CDD or ECLASS) ECLASS dictionary
- a *data type* that specifies the format of its value
- and the value itself.

Submodel elements can be grouped into *submodel element collections*, allowing data to stay together that must stay together, e.g. for an address the addressee, city, ZIP-code, street address and so on.

10.4 The AASX package

All data in an AAS instance describing one specific product item (or model, batch) can be serialized into standardized JSON objects (for online data exchange) or XML.

The serializations of one or more AAS instances can be packed into one file based on the Open Packaging Conventions (ISO…), the file bears the extension *.aasx.

The AASX file is the simplest form of exchanging AAS data, it can for example be easily sent by email.

10.5 Cryptographic Signing of Submodels

AAS submodels can be cryptographically signed using the X.509 certificate of an actor, thus allowing to verify authenticity and integrity of the submodel data.

All actors need to trust at least one common trust anchor to allow the verification of certificates and thus the authenticity of the data source; multiple trust anchors need to be considered and possible. Chosen trust anchor solutions need to be robust and scalable for global markets.

10.6 AAS access via the AAS repositories

The data exchange between active runtime system is described in "Details of the Asset Administration Shell - Part 2 Interoperability at Runtime – Exchanging Information via Application Programming Interfaces".

That document describes all that is needed to find the AAS repository by querying a registry (resolver), access an AAS repository, which operations are available at the API, and the responses to be expected by the repository.

Contact
Jochen Reinschmidt • Director • Digitalisation & Law •
Tel.: +49 30 30696023 • Mobile: +49 174 9414 164 • E-Mail: Jochen.Reinschmidt@zvei.org

ZVEI e. V. • Electro and Digital Industry Association • Charlottenstrasse 35/36 • 10117 Berlin • Germany
Lobbying Register ID.: R002101 • EU Transparency Register ID: 94770746469-09 • www.zvei.org

Date: December 20, 2023