Towards Integrated Data Control for Digital Twins in Industry 4.0

Sebastian R. Bader[1][0000−0003−1328−704X] and Maria Maleshkova[2][0000−0003−3458−4748]

[1] Fraunhofer IAIS, Schloss Birlinghoven, 53757 Sankt Augustin, Germany
[2] University of Bonn, Endenicher Allee 19a, 53115 Bonn, Germany

Abstract. The Digital Twin is currently a widely discussed topic for presenting and exchanging information on physical assets through virtual networks. Especially the digitisation of the manufacturing industry, Industry 4.0, drives the implementation of Digital Twins as part of integrated production processes. Hence, the protection of sensitive data becomes an evident challenge. We contribute by determining the requirements of current scenarios, the descriptive expressiveness of necessary vocabularies, and by formalising the involved operations. The Asset Administration Shell is used as a concrete metamodel for Digital Twins and is combined with the data sovereignty and enforcement concepts of the International Data Spaces to illustrate how these concepts can be implemented into a comprehensive Industry 4.0 scenario.

Keywords: digital twin · asset administration shell · usage control · industry 4.0

1 Introduction

The excessive use of the term 'Digital Twin' in many different contexts makes it nearly impossible to find a proper definition that covers all intended use cases. While this vagueness certainly is one reason for the success of the terminology itself, it significantly complicates its technical implementation. As a consequence, several influential initiatives and standardisation consortia have recently published detailed technical specifications and requirement lists for respective Digital Twin realisations. This technical grounding and consolidation process states what a Digital Twin must provide and how it needs to behave. The resulting specifications can be seen as the backbone for every practical roll-out of Digital Twins in productive settings.

Especially in the industrial domain faces a rising the demand for standardised, interoperable and semantically described physical assets. The heterogeneous machine parks – and the thereby created inefficiencies through elaborate integration steps, difficult maintenance procedures and the various information flows across companies – motivate Digital Twins as the core information carrier in a digital integration layer. The interoperability of assets is required due to the distributed nature of industrial production and the fact that the regarded digitisation processes usually happen in brownfield environments. Different to greenfield scenarios, brownfield settings contain a legacy systems and facilities, which cannot be replaced easily. The semantically unambiguous descriptions are necessary due to an unmanageable amount of different terms, definitions and data models, even within the same company.

One of the most critical concerns are certainly the ensuring of data security and privacy. Part of that is the protection of business-critical information and know-how. The desired opening of interfaces, systems and devices contains the risk of losing control of these resources. This paper outlines ways and processes how digital data can be exchanged through the Asset Administration Shell as one implementation of Digital Twin but at the same time ensuring that the containing information stays protected. We propose mechanisms relying on current standards, the concept of data sovereignty as proposed by the International Data Spaces (IDS) and incorporating the conventions of the Semantic Web.

We outline our vision of the AAS Digital Twin at the intersection of Web technologies, industrial requirements and privacy protection (1), explain how the Industry 4.0 requirements can be faced with Usage Control Contracts (2) and provide discussion points required to be solved for further progress in the domain (3). The following section gives a brief overview of the current situation, with a short discussion on related research works in Section 3. We provide a reference example in Section 4 to better illustrate how the AAS acts as a Digital Twin (Section 5) and which role Usage Contracts play (Section 6). We conclude with a discussion on the outlined points in Section 7.

2 Background

The proposed approach in this paper relies on the latest specifications for Digital Twins in Industry 4.0, the concept of Data Sovereignty and Data Usage Policy, and Access and Usage Control languages. These topics are briefly summarised in the following.

Digital Twins in Industry 4.0 Originally introduced as a simulation-driven virtual representation of space and aircraft vehicles, the focus of Digital Twins has shifted to data interoperability concerns and to serve as a foundation for exchanging comprehensive data models of a nearly any kind of physical assets. As this idea has gained a lot of attention, uncountable approaches, models and variation appeared in the recent years. Worth mentioning are however the proposals of the W3C working group Web of Things [4], the Digital Twin as described by the Industrial Internet Consortium [5], and the Asset Administration Shell specified by the Plattform Industrie 4.0 [2]. All three approaches are promoted by large communities and therefore comprise a sufficiently wide-spread consent.

We have selected the Asset Administration Shell as the Digital Twin specification because the data model has now, with version 2.0, reached a sufficient maturity level, the specification contains clear implementation instructions and the security model is defined as an integral part of the metamodel. In particular, the data constraint statements enable the expressing of many required constructs and therefore form a suitable foundation for the following concepts.

Data Sovereignty In current business interactions, permissions and obligations are declared through textual contracts. Wherever a formal, legally binding agreement needs to be documented, lawyers are required to express the intended meaning in written form. This is not suitable for the data-driven Industry 4.0, as interactions between organisations form on the fly, dynamically and evolve and dissolve quickly.

In addition, the relation between the involved organisations needs to be reflected by the technical implementations. Currently, arrangements between business partners rely on responsible human actors, which are trusted by the opposite party to treat their assets according to the previously concluded agreements. In case of violations, the textual contracts and the legislative system serve as escalation and disciplining stages. The interaction speed and data volume of the regarded scenarios, however, limit the monitoring capability of the involved humans, therefore restricting their options to control the data processing processes. The solution is to demand the implementation of data restrictions directly as part of the respective systems and, as far as this paper is concerned, into the Digital Twins.

The outlined situation is slightly different for person-related data, especially in member countries of the EU. The General Data Protection Regulation (GDPR) defines certain rights and obligations for digital information related to humans. Still, the relevant data sets of Industry 4.0 do usually refer to machine to machine communication and, therefore, are not in the scope of GDPR laws. As no other legislative option exists to enforce control on digital information, respective agreements have to be established on a case-by-case basis.

In the following, we refer to *contracts* when talking about legally binding agreements, which usually appear in written, natural language form. In contrast to these human-readable contracts, *policies*, are understood as formally modelled descriptions:

Definition 1. *A* ***Data Usage Policy*** *is machine-readable representation of an agreement between two legal entities, exactly one assigner and one assignee, defining the permissions on a defined data artefact. Policies are serialised in a common data format (e.g. XML or JSON) and use shared vocabularies to unambiguously outline their intentions.*

Note that in general more than two parties can participate in a policy. We intentionally leave this and other extensions open for future work. It is also important to understand that Data Usage Policies, as defined here, have currently no legally binding power. A binding agreement still requires textual contracts. Policies however omit the intended blurriness of legal clauses and need to be objectively decidable. Contracts on the other hand must be interpreted based on their context and the prevailing legal understanding of the contained clauses. The challenge is therefore to translate the established patterns of textual contracts into the digital world, and to create an equally recognised level of trust.

3 Related Work

As stated, several frameworks and guidelines have targeted the formulation, exchange and implementation of Digital Twins but also data restriction policies. The terminology of permissions, obligations, and conditions introduced by the $\mathrm{UCON}_{\mathrm{ABC}}$ [8] usage control model has been adopted by nearly all later models. In combination with RFC 2904 [9] and the introduction of the different *policy points*, the theoretical foundation of usage control systems and architectures are defined. However, none of them proposes a vocabulary to specify distinct permissions or prohibitions, their focus is rather on classifications and conceptualization. This issue is, to some degree, covered by XACML [6]. Still, XACML only focuses on *access* control, not on the more holistic usage control.

XACML applies very well in Attribute-based Access Control (ABAC) scenarios. The *object* refers to the attribute of the resource under consideration, which a *subject* wants to access. ABAC rules can be stated in plain text and interpreted by the hosting system, as intended by the AAS metamodel. Yu et al. [10] also show an approach to encode the access permission in cryptographic access key structures, hiding the content also from the hosting service. This is especially relevant if a third party is used as a cloud provider. Nevertheless, while this supports the downstream usage of ABAC rules, this approach is bound to the syntax of the data instead of its meaning.

The Data Privacy Vocabulary[3] (DPV) provides terms to annotate and categorise instances of legally compliant personal data handling according to the GDPR, including the notions *data categories*, *data controllers*, *purposes of processing data*, etc. The focus is on the description and annotation of privacy constraints. The Open Digital Rights Language (ODRL) [3] is equally able to express usage concepts through its RDF vocabulary. The specification allows expressive statements but the implications of many supported constructs are not yet sufficiently understood. The challenge is not the description of the policies but their interpretation in an enforcing system. The proposed terms lack a sufficient definition of their context, side-effects and implicit dependencies.

The IDS regards the secure and trustworthy data exchange on a data-centric, domain-agnostic level. The Reference Architecture Model [7] consists of layers to establish interoperability and crosscutting perspectives for reaching its main target, to ensure end-to-end data sovereignty. The syntactic interoperability is accomplished by the IDS Connectors as a core gateway to the IDS, with standardised interfaces and exchange protocols. An IDS Connector is a hosting system for any kind of data, for instance also for AAS, and ensuring the interests of Data Owners as expressed in formal Contracts. The IDS specification is thereby defining a data protecting infrastructure and requirements for the interacting systems, while the AAS further outlines the endpoints and data model of the specific Digital Twins living in such an infrastructure. The Policy Language itself has already been presented for general data resources [1] but went recently through major rework in order to sharpen the Usage Control clauses.

[3] `https://www.w3.org/ns/dpv`

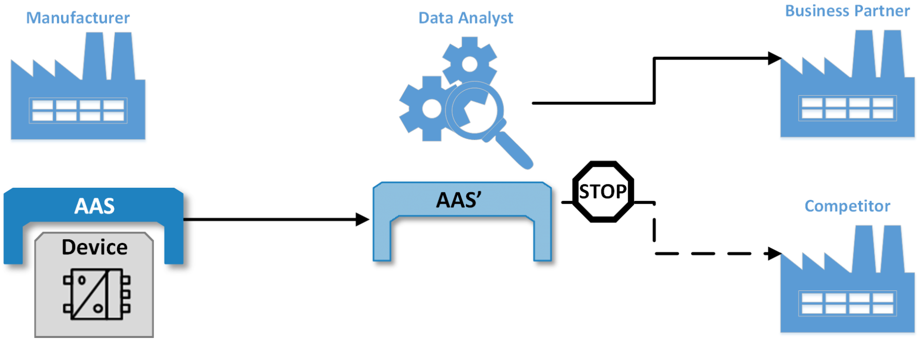

Fig. 1. Example collaboration network. The AAS is forwarded from left to right.

4 Scenario of Digital Twins in Industry 4.0

The scenario in this section serves as a running example to outline the ideas and approaches for an integrated Usage Control of a Digital Twin in an industrial setting (cf. Fig. 1). A *Manufacturer* collects static master data together with sensor observations of an operating device inside an AAS. It mandates a *Data Analyst* to access the AAS instance and come up with performance KPIs and failure predictions in order to optimise its production processes. The Manufacturer also is willing to share these insights contained in the Data Analyst's copy (AAS') with its *Business Partner*, as early information on potential breakdowns directly affects their just-in-time supply chain. It is however crucial for the Manufacturer to prohibit any access of other customers of the Data Analyst, as those might be *Competitors*.

Figure 1 outlines the information flow in this simplified setting. The asset, enriched with the AAS to a Digital Twin, provides static master data and dynamic sensor streams. This interaction is managed by an access control engine deployed directly at the AAS host. The Data Analyst as an intermediary requests the data and enriches it with the result of its prediction algorithm. Note that this step results in a newly created AAS', which is not directly located at the original asset but in the Data Analyst's server. The uniform representation conforming to the Asset Shell metadata allows the Manufacturer to automatically request and understand the AAS', independently of the data model used internally by the Data Analyst.

However – as the AAS' is now stored at the Data Analyst – the contained information is not under control of the Manufacturer anymore but still contains its critical data. As such, the Data Analyst needs to evaluate the access requests from both the Manufacturer's Business Partner and the Competitor accordingly. The required instructions and descriptions must be contained in the AAS, and also in the AAS'.

5 Asset Administration Shell as the Digital Twin

The Asset Administration Shell concept is a collection of several specification modules. The data model part specifies the core structure and different elements of the AAS, mainly administrative information about the AAS itself, the asset, and use case-specific data collected and sorted in so-called Submodels. The serialised data can be transmitted in several data formats, for instance JSON, XML or any RDF serialisation. Another part of the AAS specification targets the interaction in a remote environment. API calls for OPC UA, MQTT and REST (based on HTTP) are currently under development.

Another specification targets the infrastructure components in an AAS-driven network. Distinct systems for search and discovery but also identity provision are required. For this challenge, two different aspects need to be distinguished. The identificators of actual assets and their AAS will always be in the authority of their owners, following their company-specific schemes. As the Digital Twin of the asset is intended as a product itself, no manufacturer can be expected to rely on a third party for naming its products. The attributes, properties and values contained in the AAS, however, need to be known to the intended users, otherwise no interoperability can be obtained. Rich and extensive vocabularies such as eCl@ss[4] or the Common Data Dictionary (IEC CDD[5]) shall provide the necessary concepts. In our example, the Manufacturer can annotate the sensor streams using the eCl@ss property '0173-1#02-BAJ001#006'. Thereby, the value is fixed to integer and the Data Analyst directly knows that the values have the unit litres per minute (l/min).

These approaches, however, only marginally reflect the potential of a comprehensively web-oriented approach. The catalogues only provide basic identifiers reflected in a taxonomy structure. Hypermedia links or machine-readable annotations are not provided out of the box. Identifiers have to be dereferenced using catalogue-specific methods. Following the assumption that the Web stack (cf. Ass. 1) is a uniting technology and in place in every Industry 4.0 scenario at some point. Note that this does not mean that non-Web interactions shall be neglected. For instance, in machine-to-machine interactions, HTTP or Web browsers are usually not an appropriate choice. However, all developers, operators or any other involved stakeholders is capable to access and exchange Web-based information using the *currently* available tools, different to any other available technology set. This convenience advantage can not be underestimated since the acceptance by the target community is absolutely crucial for the success of any Industry 4.0 proposal.

Assumption 1 *Web technologies are the common denominator known to every Industry 4.0 stakeholder. All involved parties are familiar with URIs, HTTP, DNS, and Web Browsers and can use this technology set to establish Industry 4.0 use cases based on it.*

[4] http://www.eclasscontent.com/

[5] https://cdd.iec.ch/cdd/iec61360/iec61360.nsf

Linked Data and the Linked Data vocabularies can provide terms and concepts with rich annotations and machine-readable relations. Nevertheless, several reasons still severely hamper the wider adoption of Linked Data-based approaches for Industry 4.0 vocabularies. First, the requirements of industrial applications are **long-term** and for **formal and guaranteed maturity**. As legal liabilities arise from the implementation of concepts in productive facilities, a community-based approach such as DBpedia is indefensible. Established and well-reputated agencies need to back-up such a vocabulary, with clearly defined process and decision-criteria. This is in contrast to the less formal procedures currently in place in the Semantic Web community, which is still mostly driven from an academic background.

Statement 1 *Long-term support and formal liability are crucial factors for the adoption of any digital resource though the manufacturing industry. Preferred are established standardisation agencies such as ISO, IEC, NIST or DIN.*

The second reason is certainly the separation of communities. Industrial developments are, by their nature, mostly driven and managed by people with an engineering background. Web developers on the other hand do not understand the specific requirements or speak the used language appropriately. For instance, using the running example, a software application ran at the Data Analyst can quite easily be updated, relocated or completely replaced. This is a common development for Web services. A physical device or facility needs to stick to safety regulations and certification processes as otherwise people could get harmed. This leads to significantly longer and more complex design and decision processes. Combining digital services with physical assets leads to clashes between both approaches and communities. Fortunately, a steady progress can be identified. This is certainly driven by the identified business potential and the wide-spread conviction that neither non-digital devices nor plain software presents a future-proven product.

Statement 2 *The current solutions proposed by Industry 4.0 consortia hardly reflect the potential of Web technologies. The (Semantic) Web community, on the other hand, needs to reflect the formal requirements of industrial applications and potential harmful consequences.*

An entity, as understood in this paper, aggregates physical or tangible objects, information resources or digital data objects, software applications and any other identifiable asset. It becomes a Digital Twin when it is extended with a digital form, representing it in digital networks. As we want to examine the AAS as the Digital Twin for the Industry 4.0, we use the following definition:

Definition 2. *A* ***Digital Twin*** *is the combination of one distinct entity, called* ***Asset****, and its digital counterpart. This counterpart consists of the* ***Asset Administration Shell*** *and several Submodels containing data about the Asset filtered according to relevant use cases.*

Note that the relation between an Asset and an representing AAS is not necessarily a one-to-one connection. For instance, the device in our example has an assigned AAS with all information for the Data Analyst and another one with relevant data for the Manufacturer's engineering department. They have different content and different identifiers. That means that not one single Digital Twin exist but rather overlapping sets.

Statement 3 *As Digital Twins go through their own, but also reflect their Assets' life-cycles across companies and phases, one single, universal Digital Twin instance is usually not possible. Therefore, the AAS must be regarded as a fragmented set of information of the actual Asset.*

In particular, this implies that the Open World Assumption holds for the AAS. Potential reasoning or other examinations of their content always takes place with incomplete information. This fact is partly regarded by the efforts to standardise Submodels as use case-dependent information carriers, which fix the set of possible attributes and features.

All these models and data provisioning needs to be based on a well-defined and consistent identification method. Currently, an incomprehensible set of legacy patterns is used in the domain, challenging a unifying approach. Furthermore, the ability and authority to select own identifiers is an essential characteristic for the public appearance of companies. However, proprietary schemes – sometimes implicitly encoding product families or versions – can lead to misunderstandings and errors at downstream services. Following the recommendations of the (Semantic) Web – using URIs – is a proven and technically easily manageable convention. This is formulated by the following assumption:

Assumption 2 *Every relevant actor in Industry 4.0 has its own internet domain. A direct mapping between the domain and the related company is always possible, also vice versa.*

Due to the consideration that every company nowadays needs its own website, the validity of this claim seems reasonable. We furthermore regard an internet domain as a valuable, well-protected resource. Even though one can construct use cases where companies lose control over their domain, this will only happen in very few cases. Consequently, a sufficiently durable namespace for identifiers is directly available for every company. Even though this assumption might be convincing at a first glance, the legal obligations of long-lasting industrial identifiers are not necessarily met yet. This gap however might be resolved through new service providers, guaranteeing their sustainability as a new business model.

Assumption 3 *Internet domains are maintained over the whole life-cycle of a Digital Twin. A company will not lose control over its domain at any time.*

Statement 4 *Constructing identifiers by concatenating a company's internet authority with its dedicated product-naming scheme automatically creates globally unique identifiers. The expressiveness of URIs is suitable for transporting this information in a syntactically feasible manner.*

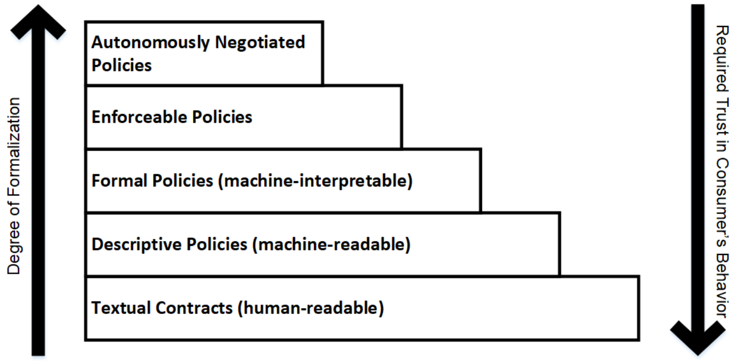
Autonomously Negotiated Policies
Enforceable Policies
Formal Policies (machine-interpretable)
Descriptive Policies (machine-readable)
Textual Contracts (human-readable)
Degree of Formalization
Required Trust in Consumer's Behavior

Fig. 2. Stairway of Usage Policy Categories by required formalisation.

Statement 5 *URIs are the most commonly used identifiers throughout digital applications. Their established dissemination alone makes them superior to any other scheme.*

Note that the convenience of the target group is again the dominant argument in favour of URIs. While this appears as a non-functional argument, we want to stress the fact that missing dissemination and adoption are the key obstacles of any Digital Twin concept published so far.

6 Policies and Machine-readable Restrictions

The expressiveness of the policies does, in general, not depend on the area they are evaluated but on the targeted position in Figure 2. A human operator can formulate and understand complex constructs, easily creating insolvable issues for any AI. An control language stretching an unrestricted space is therefore not beneficial, as it requires direct human intervention each time a policy is regarded. However, in order to cope with most of the currently relevant use cases, we are confident that a small number of formalised dimensions (counting, time, space, membership cf. [1]) is sufficient.

Those dimensions need to be unambiguously defined, limit the variety of interpretations and specify implementable logic for their evaluation. While for instance XACML or, as regarded in this paper, the Security module of the AAS and the IDS Usage Contracts provide a structure for data protection rules, their implications for evaluations is not yet sufficiently defined. The assumption reflects the fact that there should be not only a shared understanding on the syntax of contracts but also their meaning. For our example, the Data Analyst must not only be able to request and receive the AAS but also understand and evaluate the clauses itself later on. While this is certainly solvable in point-to-point scenarios by simply agreeing on the allowed semantics, a truly federated framework for Industry 4.0 is not reached yet. This brings us to our next postulation, as the Usage Control of the AAS can only be as reliable as the embedding systems:

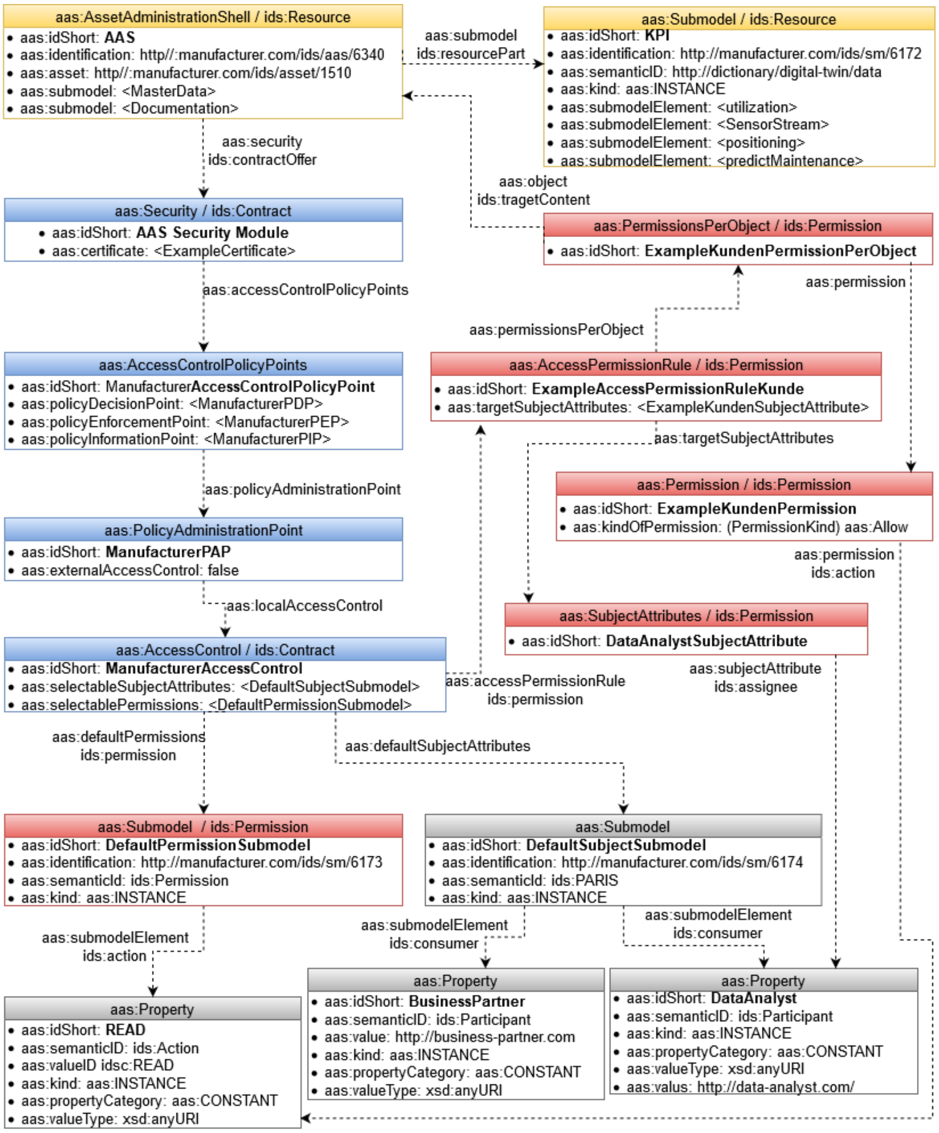
aas:AssetAdministrationShell / ids:Resource
• aas:idShort: AAS
• aas:identification: http//:manufacturer.com/ids/aas/6340
• aas:asset: http//:manufacturer.com/ids/asset/1510
• aas:submodel: <MasterData>
• aas:submodel: <Documentation>
aas:submodel
ids:resourcePart
aas:Submodel / ids:Resource
• aas:idShort: KPI
• aas:identification: http://manufacturer.com/ids/sm/6172
• aas:semanticID: http://dictionary.de/digital-twin/data
• aas:kind: aas:INSTANCE
• aas:submodelElement: <utilization>
• aas:submodelElement: <SensorStream>
• aas:submodelElement: <positioning>
• aas:submodelElement: <predictMaintenance>
aas:security
ids:contractOffer
aas:object
ids:tragetContent
aas:Security / ids:Contract
• aas:idShort: AAS Security Module
• aas:certificate: <ExampleCertificate>
aas:PermissionsPerObject / ids:Permission
• aas:idShort: ExampleKundenPermissionPerObject
aas:permission
aas:accessControlPolicyPoints
aas:permissionsPerObject
aas:AccessControlPolicyPoints
• aas:idShort: ManufacturerAccessControlPolicyPoint
• aas:policyDecisionPoint: <ManufacturerPDP>
• aas:policyEnforcementPoint: <ManufacturerPEP>
• aas:policyInformationPoint: <ManufacturerPIP>
aas:AccessPermissionRule / ids:Permission
• aas:idShort: ExampleAccessPermissionRuleKunde
• aas:targetSubjectAttributes: <ExampleKundenSubjectAttribute>
aas:targetSubjectAttributes
aas:policyAdministrationPoint
aas:Permission / ids:Permission
• aas:idShort: ExampleKundenPermission
• aas:kindOfPermission: (PermissionKind) aas:Allow
aas:PolicyAdministrationPoint
• aas:idShort: ManufacturerPAP
• aas:externalAccessControl: false
aas:permission
ids:action
aas:localAccessControl
aas:SubjectAttributes / ids:Permission
• aas:idShort: DataAnalystSubjectAttribute
aas:AccessControl / ids:Contract
• aas:idShort: ManufacturerAccessControl
• aas:selectableSubjectAttributes: <DefaultSubjectSubmodel>
• aas:selectablePermissions: <DefaultPermissionSubmodel>
aas:accessPermissionRule
ids:permission
aas:subjectAttribute
ids:assignee
aas:defaultPermissions
ids:permission
aas:defaultSubjectAttributes
aas:Submodel / ids:Permission
• aas:idShort: DefaultPermissionSubmodel
• aas:identification: http://manufacturer.com/ids/sm/6173
• aas:semanticId: ids:Permission
• aas:kind: aas:INSTANCE
aas:Submodel
• aas:idShort: DefaultSubjectSubmodel
• aas:identification: http://manufacturer.com/ids/sm/6174
• aas:semanticId: ids:PARIS
• aas:kind: aas:INSTANCE
aas:submodelElement
ids:action
aas:submodelElement
ids:consumer
aas:submodelElement
ids:consumer
aas:Property
• aas:idShort: READ
• aas:semanticID: ids:Action
• aas:valueID idsc:READ
• aas:kind: aas:INSTANCE
• aas:propertyCategory: aas:CONSTANT
• aas:valueType: xsd:anyURI
aas:Property
• aas:idShort: BusinessPartner
• aas:semanticID: ids:Participant
• aas:value: http://business-partner.com
• aas:kind: aas:INSTANCE
• aas:propertyCategory: aas:CONSTANT
• aas:valueType: xsd:anyURI
aas:Property
• aas:idShort: DataAnalyst
• aas:semanticID: ids:Participant
• aas:kind: aas:INSTANCE
• aas:propertyCategory: aas:CONSTANT
• aas:valueType: xsd:anyURI
• aas:value: http://data-analyst.com/

Fig. 3. Basic model of the discussed AAS. Color-coding and namespace annotations refer to the respective IDS concepts.

Statement 6 *A reliable end-to-end Usage Control scenario for Digital Twins requires trustworthy systems. Their state needs to be verifiable through certification processes, cryptographic signatures and controlled environments.*

Figure 3 shows the example expressed as an Asset Administration Shell snippet with focus on the description of the usage restrictions. The yellow, top classes represent the actual Digital Twin and the features of interest. According to the metadata model, those are represented by the AssetAdministrationShell and Submodel classes. The core data control sections are outlined, starting with the Security to the AccessControl class (blue), followed by the definition of relevant entities and interactions (grey). The red Permission-related classes close the circle and link back to the actual protected features. Note that the same pattern can be expressed through an Usage Contract. The respective classes and properties are directly added but in another namespace (aas vs. ids).

The AAS is driven by the understanding of the host as the authority to define security rules. The focus is twofold, (a) to describe the requesting parties (called 'subjects') and their relations to certain attributes ('objects'). Furthermore, the relevant subcomponents of the control system are explicitly modelled (b).

The IDS aims to provide such a controlled, trustworthy ecosystem. While making no difference in terms of the nature of the exchanged data, the IDS Connectors ensure a certain degree of control through certified execution environments, partly independent of the operating organisation. The IDS therefore locates the definition of those Policy Points in the responsibility of the hosting system, without any mentioning in the Policy at all. While this behaviour allows for the shipment of both the AAS and the Policies, the interpretation of the described Policy is harder and requires higher degrees of formalisation. For instance, as soon as the AAS' is evaluated by the Data Analyst, its enforcement engine needs to interpret the Policy in accordance with its own local environment. This implies the need to be able to independently guess appropriate Policy Points. This very challenging task is further complicated by the prohibition of direct manual configuration, since the hosting system must ensure the Data Producer's interests and resist these kinds of manipulations.

Statement 7 *Derived Digital Twins must contain the original usage constraints. Downstream activities must be compliant to all previously stated restrictions.*

This is of course difficult to maintain, as at some point in any workflow, the input components merge to a new product, with the assembler (previously consumer) as the new owner and provider. A simple example are the manuals shipped with each component of a machine. The Manufacturer in our example combines not only the device components into its product but also the manuals and safety documentation. Its customers of course do not get a single document set for each and every component. The resulting device's documentation, even though reusing content from several suppliers, is the property of the Manufacturer. Defining at which stage a new product appears – and who has the authority over its further use – is obviously a significant challenge. This is especially true for digital data in general and Digital Twins in particular.

7 Conclusion

We have outlined our vision how the merging of Usage Control and Data Sovereignty with the Asset Administration Shell can create an interoperable but protected Digital Twin for industrial purposes. The provided assumptions and statements are intended as starting points for further discussions. We believe that a consolidation of the thereby affected approaches and concepts is necessary, and a trade-off between formalisation and expressed details on the one hand and adoption on the other is indispensable.

Still, the demand for more and more autonomously acting systems enforces overhead in terms of data models and implementations. The Usage Policies explained in the AAS show how the different specifications can be combined to a comprehensive interpretation. We have observed that similar ideas and pattern appear in the different communities. The integrated approach as outlined in this paper can therefore also serve as a bridge between the communities. The combination of identification and interaction patterns with the standardisation efforts and domain requirements of manufacturers requires efforts from all parties. The result however has the potential to disrupt the way we regard assets and data in the intersection of the physical and virtual world.

References

1. Bader, S., Maleshkova, M.: Towards enforceable usage policies for industry 4.0. In: Proceedings of the 1st Workshop on Large Scale RDF Analytics (2019)
2. Barnstedt, E., Bedenbender, H., Billmann, M., Boss, B., Clauer, E., Fritsche, M., Garrels, K., Hankel, M., Hillermeier, O., Hoffmeister, M., et al.: Details of the Asset Administration Shell: Part 1. Tech. rep., BMWi (2019), Version 2.0
3. Ianella, R., Villata, S.: ODRL Information Model 2.2 (2018), https://www.w3.org/TR/odrl-model/, W3C Recommendation
4. Kaebisch, S., Kamiya, T., McCool, M., Charpenay, V., Kovatsch, M.: Web of Things (WoT) Thing Description (Jan 2020), https://www.w3.org/TR/2020/PR-wot-thing-description-20200130/, W3C Proposed Recommendation
5. Malakuti, S., van Schalkwyk, P., Boss, B., Sastry, C., Runkana, V., other: Digital Twins for Industrial Applications (2020), https://www.iiconsortium.org/white-papers.htm, IIC White Paper
6. Moses, T., Anderson, A., Nadalin, A., et al.: eXtensible Access Control Markup Language (XACML) (Dec 2004), http://docs.oasis-open.org/xacml/access_control-xacml-2_0-core-spec-cd-04.pdf
7. Otto, B., Lohmann, S., Auer, S., Brost, G., et al.: IDS Reference Architecture Model. IDSA (2019), Version 3.0, available at https://www.internationaldataspaces.org/ressource-hub/publications-ids/
8. Park, J., Sandhu, R.: The UCON ABC usage control model. ACM Transactions on Information and System Security (TISSEC) **7**(1), 128–174 (2004)
9. Vollbrecht, J., Calhoun, P., Farrell, S., Gommans, L., Gross, G., et al.: RFC 2904: AAA Authorization Framework. Network Working Group (2000)
10. Yu, S., Wang, C., Ren, K., Lou, W.: Achieving secure, scalable, and fine-grained data access control in cloud computing. In: 2010 Proceedings IEEE INFOCOM. pp. 1–9. Ieee (2010)