

Expertennetzwerk-Treffen zur Verwaltungsschule am 28. Oktober 2025

AAS in der Praxis erfolgreich umsetzen! Ein Erfahrungsaustausch

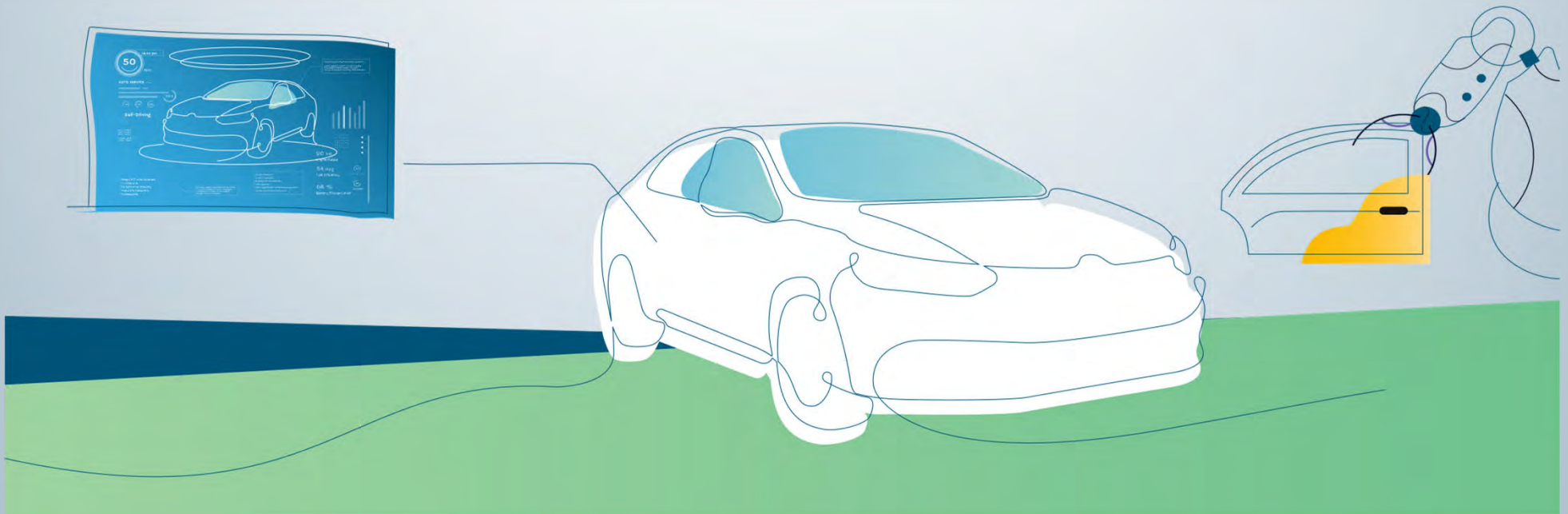
Olaf Poenicke & Florian Warschewske
Fraunhofer IFF, Magdeburg

Workshop 2

Projekt „DigiFlexMont“

Unser Usecase für die Anwendung und Erprobung von AAS

- **Digitale flexible Montageassistenz- und Prüftechnologien auf Basis des digitalen Zwillings von Produkt und Produktionsmitteln**



Projekt „DigiFlexMont“

Montage und Prüfung von PKW-Türen

Fokus der Entwicklungen

- ❖ Entwicklung **flexibler Inline-Montageprüfsysteme** zur Erkennung von Bauteilvollständigkeit und korrekter Lage

Prüfverfahren

- ❖ Berührungslose Prüfung mittels **robotergetragener optischer Sensorik**
- ❖ **Automatische Anpassung** der Prüfabläufe an Produktvarianten durch den **digitalen Zwilling**

Mensch-Maschine-Interaktion

- ❖ **Augmented Reality** zeigt Produktdaten und Prüfergebnisse direkt im Arbeitsbereich an



Projektziel

Nachweis des Potenzials durchgängiger digitaler Zwillinge mit standardisierten Schnittstellen

Projekt „DigiFlexMont“

Systemkomponenten des Demonstrators

Prüfobjekt

Linearachse

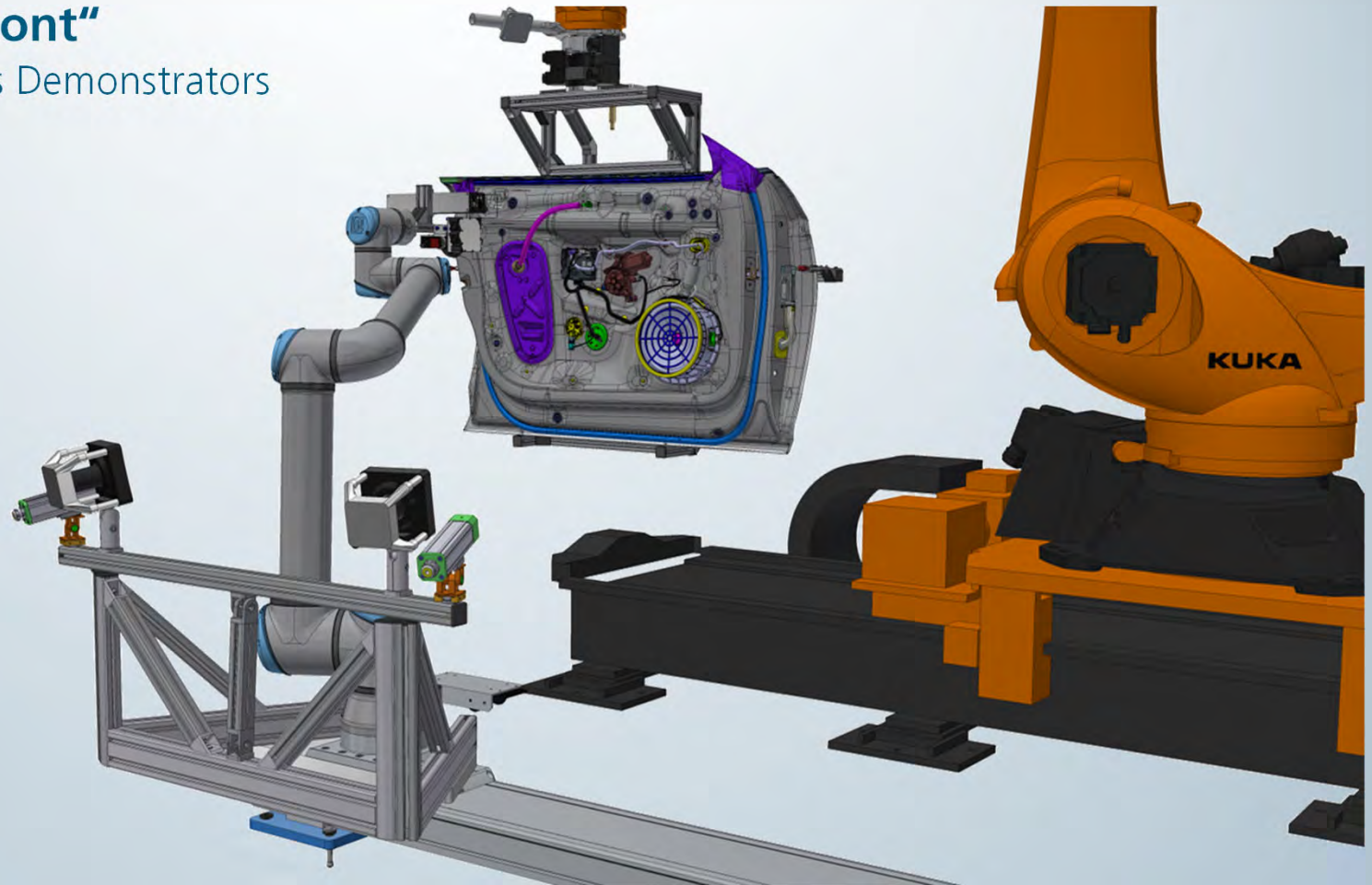
Roboter UR10

Sensoren

Trackingsystem

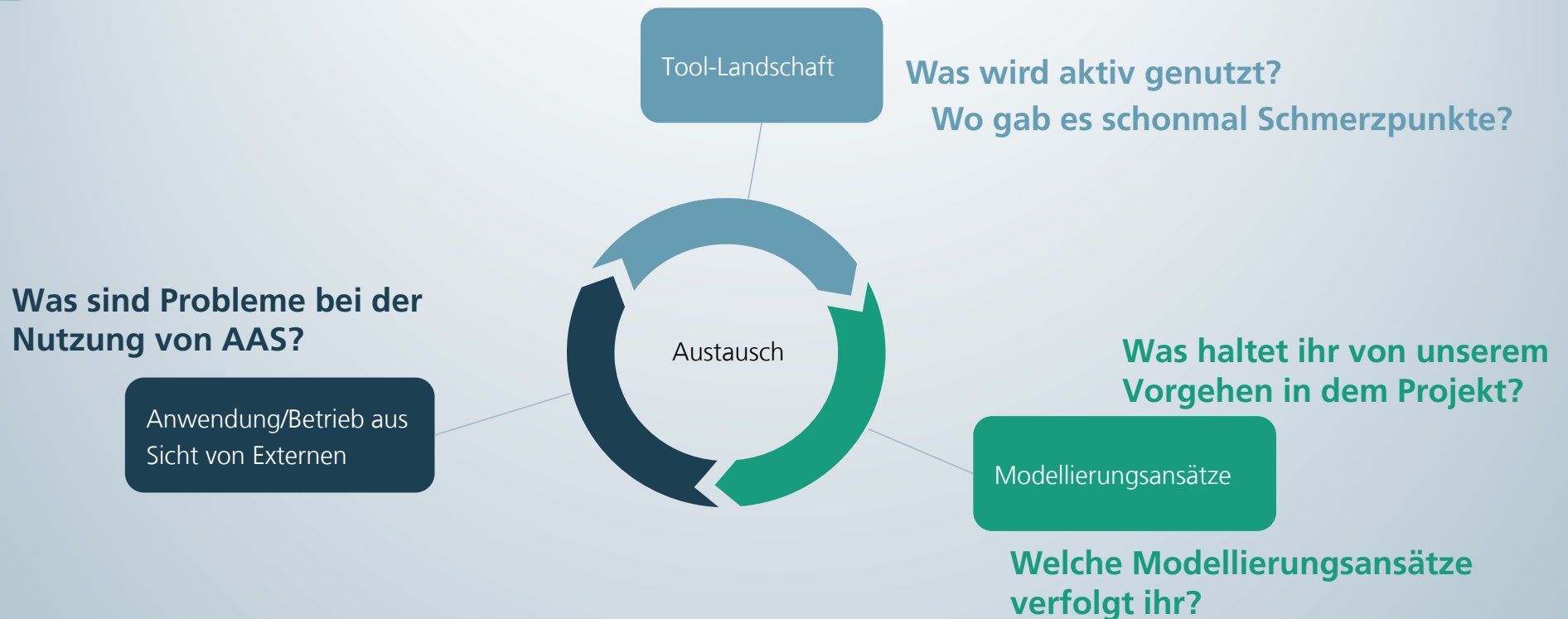
Transportsystem

Nacharbeitsstation



Diskussionsrunde & Austausch

Was sind die Themen?



➤ Alle Betrachtungen unter der Maßgabe „Industry ready“

Diskussionspunkt I: Tool-Landschaft & Metamodell

Was noch nicht rund läuft – Erfahrungen aus dem Projektalltag

1

Submodel-Templates

„Offizielle Submodel-Templates sind nicht standardkonform bzw. schaffen es nicht durch die Validierung.“

35 von 52 sind valide (67%)

Quelle (Stand 27.10.25)

<https://github.com/admin-shell-io/submodel-templates/tree/main/published>

2

AAS-Environments

„Das System akzeptiert Environments mit mehrfach vergebenen AAS-IDs, was nicht dem erwarteten Verhalten entspricht.“

- Beispiel: Anlegen von Prüfobjekten als AAS. In der CAD-Welt existieren u.U. doppelte IDs, was zu Problemen führt

Quelle (Stand 24.10.25)

Eigene Tests mit aas-core3.0-python

3

Spezifikation

„Die Spezifikation scheint an einigen Stellen noch Klärungsbedarf zu haben.“

- Constraint AASd-128 vs. Definition von Key/value
- statement vs. statements in der Definition „Entity“

Quelle (Stand 24.10.25)

IDTA Document Number: IDTA-01001-3-1
(Part 1: Metamodell)

4

Serialisierungen

„In der Praxis zeigen bestimmte Tools bei der Serialisierung und Deserialisierung einfacher Listen Abweichungen.“

- Beispiel: Anlegen von Transformationsmatrizen für unsere Prüfmerkmale als SubmodelElementList

Quelle (Stand 24.10.25)

Eigene Tests mit Eclipse Basyx

Diskussionspunkt I: Tool-Landschaft & Metamodell

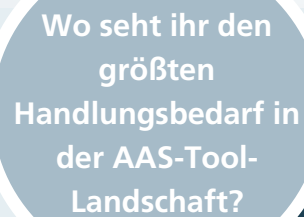
Bewertung der aktuellen Tool-Landschaft



- durch Teilnehmende genutzte und bewertete Tools
- blau → Tool wird genutzt
- grün → Tool wird positiv bewertet
- rot → Kritikpunkte am Tool

→ Negativaussagen zum Package Explorer beziehen sich darauf, dass dieser nicht ausreichend maintained wird

Diskussionsrunde

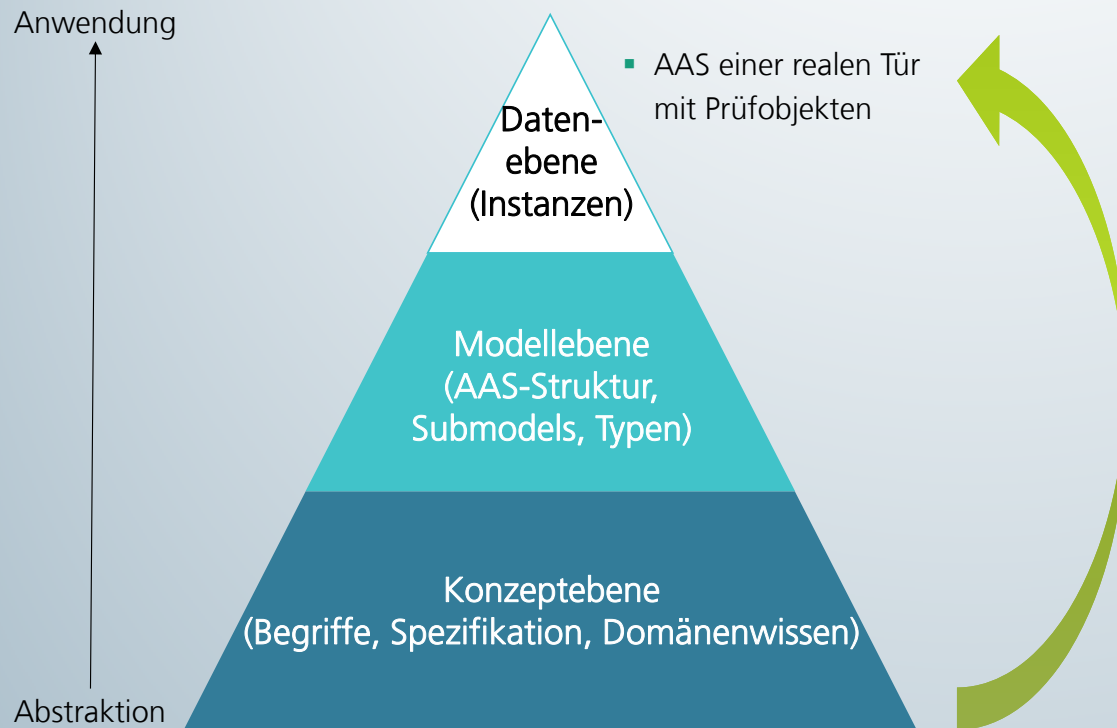


Welche Features sind noch wünschenswert?

1. Es fehlen Tools zur automatisierten Validierung – auch schon auf Ebene der Definition von Submodels
2. Bedarf an einer zertifizierenden Stelle, die Validität von Standards und Modellen absichert
3. Quick-Win, wenn in Submodel-Gruppen auch Modellierungsexperten mitwirken

Diskussionspunkt II: Modellierungsansätze

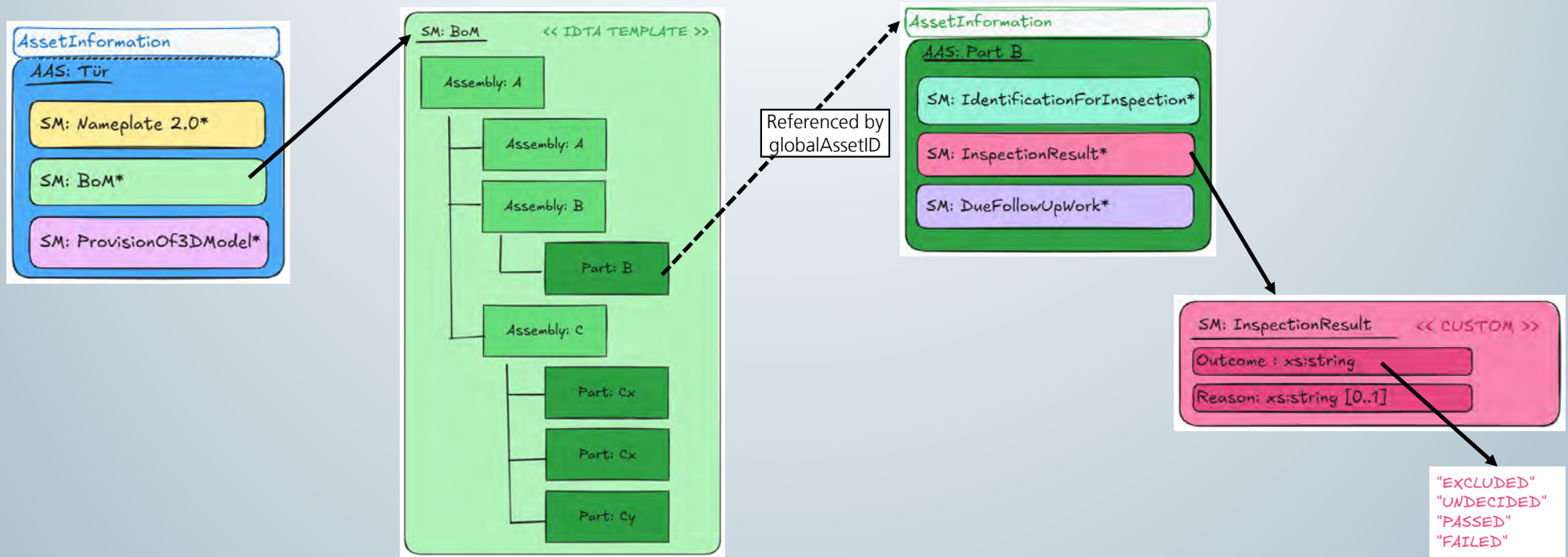
Was meinen wir mit „Modellierung“ in diesem Kontext?



Diskussionspunkt II: Modellierungsansätze

Beispiel: Prüfergebnisse für Anbauteile

Vorschlag für ein mögliches Zielbild:

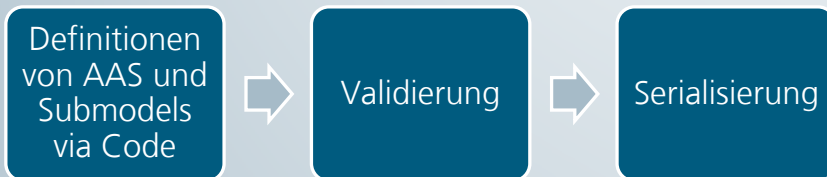


Diskussionspunkt II: Modellierungsansätze

aas-core-works (<https://github.com/aas-core-works/aas-core3.0-python>)

Definition

aas-core-works stellt Typen und Funktionalitäten bereit, um AAS-konforme Datenstrukturen zu **erzeugen**, zu **validieren** und zu **serialisieren**.



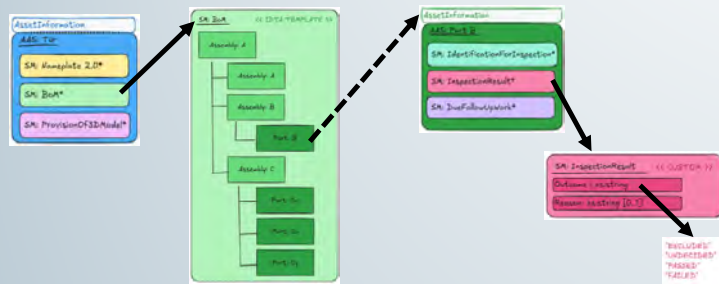
Beispiel

```
some_element = aas_types.Property(  
    id_short="some_property",  
    value_type=aas_types.DataTypeDefXSD.INT,  
    value="1984"  
)
```

```
submodel = aas_types.Submodel(  
    id="some-unique-global-identifizier",  
    submodel_elements=[  
        some_element,  
        another_element  
    ]  
)
```

Diskussionspunkt II: Modellierungsansätze

Wie sieht das für unser Beispiel aus?



4	AAS	"AAS_DOOR_FOR_WORKSHOP"	[urn:x-dfm-v1:assetaudi-door-for-workshop] of [urn:x-dfm-v1:assetaudi-door-for-workshop]
	Asset	AssetInformation	urn:x-dfm-v1:assetaudi-door-for-workshop
	SM	"NAMEPLATE_FOR_WORKSHOP"	[https://admin-shell.io/idta/SubmodelTemplate/DigitalNameplate/2/0]
	DN2	Digital Nameplate	ready
	Prop	"URIOfTheProduct"	= www.iff.fraunhofer.de @ (Multiplicity=One)
	MLP	"ManufacturerName"	→ Audi @ (Multiplicity=One)
	MLP	"ManufacturerProductDesignation"	→ Door-1 @ (Multiplicity=One)
	SMC	"ContactInformation"	(4 elements) @ (Multiplicity=One)
	Prop	"YearOfConstruction"	= 2022 @ (Multiplicity=One)
	SM	"BOM_FOR_WORKSHOP"	[https://admin-shell.io/idta/SubmodelTemplate/HierarchicalStructuresIdM/1/1]
	Prop	"Archetype"	= Full
	Ent	"EntryNode"	
	Ent	"Node_0"	
	Rel	"HasPart_Node_0"	@ (SMT/Cardinality=ZeroToMany) @ (EditShort=True)
4	AAS	"Node_0"	[urn:x-dfm-v1:asset1935c4d6_b68e_46a4_8dca_4355b236e3c4] of [urn:x-dfm-v1:asset1935c4d6_b68e_46a4_8dca_4355b236e3c4]
	Asset	AssetInformation	urn:x-dfm-v1:asset1935c4d6_b68e_46a4_8dca_4355b236e3c4
	SM	"SM_INSPECTION_RESULT"	[urn:x-dfm-v1:de.fraunhofer.iff.fma.submodelinspection-result:1935c4d6_b68e_46a4_8dca_4355b236e3c4]
	Prop	"Outcome"	@ (Select=[EXCLUDED, UNDECIDED, PASSED, FAILED]) @ (Cardinality=One)
	Prop	"Reason"	@ (Cardinality=ZeroToOne)
1	AAS	"Node_1"	[urn:x-dfm-v1:asset8191328_a136_4e0d_b416_312778ccdd63] of [urn:x-dfm-v1:asset8191328_a136_4e0d_b416_312778ccdd63]
1	AAS	"Node_2"	[urn:x-dfm-v1:asset3a8ef6a4_d0ed_49df_a438_a0d9db6a483] of [urn:x-dfm-v1:asset3a8ef6a4_d0ed_49df_a438_a0d9db6a483]
1	AAS	"Node_3"	[urn:x-dfm-v1:asset45c05549_d974_459b_ac53_e0446f4e90086] of [urn:x-dfm-v1:asset45c05549_d974_459b_ac53_e0446f4e90086]
1	AAS	"Node_10"	[urn:x-dfm-v1:asset14bed3_5f54_4ad3_a471_1114ef014f9] of [urn:x-dfm-v1:asset14bed3_5f54_4ad3_a471_1114ef014f9]
1	AAS	"Node_37"	[urn:x-dfm-v1:asset8ca1fe0a_8119_4aa7_9cad_964db60e74cd] of [urn:x-dfm-v1:asset8ca1fe0a_8119_4aa7_9cad_964db60e74cd]
1	AAS	"Node_40"	[urn:x-dfm-v1:asset27346d8b_865d_4e2d_b613_abc0246024e] of [urn:x-dfm-v1:asset27346d8b_865d_4e2d_b613_abc0246024e]
1	AAS	"Node_53"	[urn:x-dfm-v1:asset83b6984_739e_445a_bcb5_626396c20403] of [urn:x-dfm-v1:asset83b6984_739e_445a_bcb5_626396c20403]

Element	Content
AssetAdministrationShell (according IEC 63278)	
Referable:	
idShort:	AAS_DOOR_FOR_WORKSHOP
displayName:	[en] Audi Door Instance
description:	[de] Audi Tür Instance
	[en] Instance-AAS for DigiFlexMont-Door at IFF
	[de] Instanz-AAS für DigiFlexMont-Autotür am IFF
Known extensions « experimental »:	
HasExtension:	
Identifiable:	
id:	urn:x-dfm-v1:assetaudi-door-for-workshop
id (Base64):	dXJuOnRlZGZlLXVxOmFnc3phdWRpLWLRvb3h2Zm9yLXZvcmtzaG9w
HasDataSpecification (Reference):	
AssetInformation	
Kind (of AssetInformation):	
kind:	Instance
AssetInformation	
globalAssetId:	
globalAssetId:	urn:x-dfm-v1:assetaudi-door-for-workshop
assetType:	
assetType:	AutomotivePart
specificAssetId:	
Element 1: PartName	
name:	PartName
value:	Audi Door
Element 2: InternalPartName	
name:	InternalPartName
value:	1001V63013_001_VPM_AU49X
DefaultThumbnail: Resource element	

Subtopics

- AAS für die Tür
- Submodel BoM (IDTA Template)
- Submodel „InspectionResult“
- Validierung/Serialisierung
- Visitor Pattern (bei Interesse)

Diskussionspunkt II: Modellierungsansätze

Diskussionsrunde

```
import aas_core3.types as aas_types

# Create a Property
aas_types.Property(
    id_short="CoolProperty",
    value_type=aas_types.DataTypeDefXSD.STRING,
    value="This is a cool property",
    description=[
        aas_types.LangStringTextType(language="en",
            text="A very cool property"),
        aas_types.LangStringTextType(language="de",
            text="Eine sehr kühle Eigenschaft"),
    ],
)
```

Wer hat die aas-core-works Bibliothek schon benutzt?

Wie stark müsste ein Tool abstrahieren, damit auch Nicht-Programmierer modellieren können?

Wie könnte dieses Vorgehen in eurem Umfeld helfen?

Wo seht ihr aktuell Hürden beim Modellieren?


Was haltet ihr vom gezeigten Vorgehen um AAS und Submodels zu erzeugen?

Bedarfe / Quick Wins

1. Herausforderungen in der Modellierung sind sehr anwendungsspezifisch → für Domänenexperten darf die AAS keine separaten Anforderungen mit sich bringen → AAS-Funktionen (z.B. zur Modellierung einer AAS) müssen in etablierten Tools für den jeweiligen Anwendungsfokus integriert werden → z.B. Modellierung einer AAS als Resultat einer CAD-Modellierung

Diskussionspunkt III: Nutzung durch Externe

Modelle und Server stehen bereit, die Anwendung bleibt herausfordernd



„Ich möchte die Konzepte verstehen und anwenden, aber bitte ohne 300 Seiten Spezifikation!“

„Das IFF hat uns da Container mit AAS-Typen und Templates geliefert, ich habe keine Ahnung was ich damit machen soll!?“

„Wie traversiert man jetzt eigentlich die Baumstruktur der Prüfmerkmale für unsere Prüfroutine?“

Wir haben im Projektverlauf festgestellt:

- Uns fiel es schwer den Mehraufwand zu rechtfertigen.
- Die Erstellung von Anleitungen oder Vorgehensweisen, die den Einstieg erleichtern, gestaltete sich für uns schwierig.
- Die Bausteine der AAS sind solide, aber die Entwicklung einer komplexeren Anwendung erfordert nach wie vor sorgfältige Abstimmungen und durchdachte Designentscheidungen.
- ~~KMU~~ Menschen lieben ihre vorhandenen Tools und Infrastrukturen, Neuerungen werden sehr kritisch aufgefasst.

Diskussionspunkt III: Nutzung durch Externe

Diskussionsrunde



Fehlt aktuell eine
niedrigschwellige
„Einstiegstür“ in die
AAS-Welt?

Wie leicht oder schwer ist
es aktuell, eine AAS zu
verstehen oder zu nutzen,
wenn man kein Entwickler
bzw. „fachfremd“ ist?

Wie sind eure
Erfahrungen mit
„Fachfremden“?



Wie kann man zeigen,
dass AAS keine
zusätzliche Arbeit,
sondern Entlastung
bringt?

Wie viel Komplexität
ist für diese
Zielgruppen zumutbar
und wo beginnt
Überforderung?

Bedarfe / Quick Wins

1. Selbe Erkenntnis wie unter Modellierung → AAS-Funktionen müssen als Module in andere etablierte Tools integriert werden
→ Herausforderung hierbei ist, dass Entwickler solcher Tools in der AAS bisher keinen klaren Business Case haben

Vielen Dank für die Teilnahme an unserem Workshop!

Kontakt

M.Sc. Florian Warschewske

Fertigungsmesstechnik und digitale Assistenzsysteme

Tel.: +49 391 4090 269

Email: florian.warschewske@iff.fraunhofer.de

Fraunhofer-Institut für Fabrikbetrieb und -automatisierung

Sandtorstraße 22

39106 Magdeburg

www.iff.fraunhofer.de

Dr.-Ing. Olaf Poenicke

Fertigungsmesstechnik und digitale Assistenzsysteme

Tel.: +49 391 4090 337

Email: olaf.poenicke@iff.fraunhofer.de

Fraunhofer-Institut für Fabrikbetrieb und -automatisierung

Sandtorstraße 22

39106 Magdeburg

www.iff.fraunhofer.de

Besucht uns in der Elbfabrik in Magdeburg!

