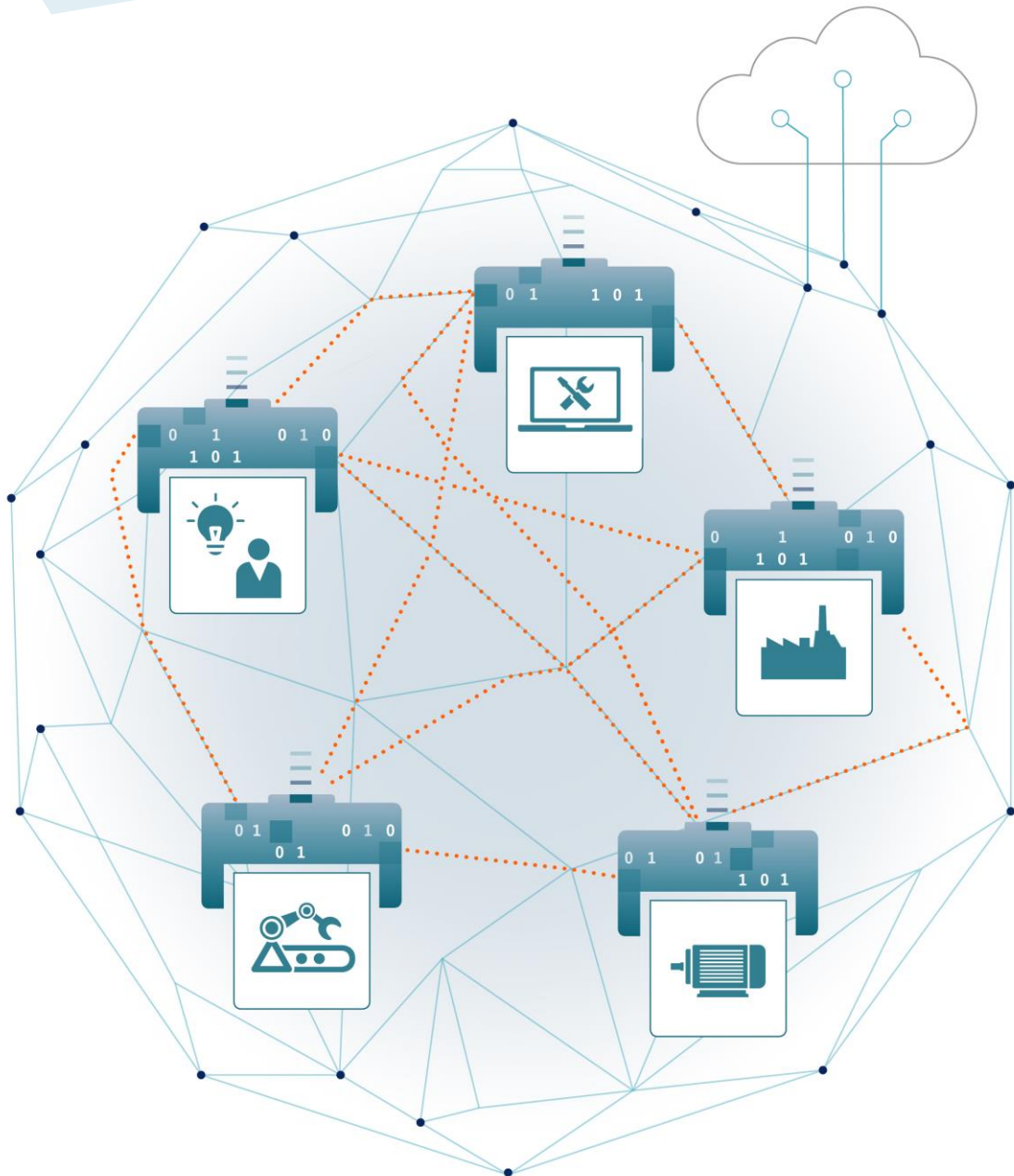


Diskussionspapier



VWS-Referenzmodellierung

Exemplarische Modellierung einer fertigungstechnischen Anlage mit AASX Package Explorer auf Basis des VWS-Metamodells

Impressum

Herausgeber

Plattform Industrie 4.0
Bülowstraße 78
10783 Berlin
geschaeftsstelle@plattform-i40.de

Redaktion, Layout und Fotos/Grafiken

Plattform Industrie 4.0
Bülowstraße 78
10783 Berlin

Stand

April 2021

Diese Publikation wird von der Plattform Industrie 4.0 herausgegeben. Die Publikation wird kostenlos abgegeben und ist nicht zum Verkauf bestimmt.

Inhalt

(1) Ziel und Vorgehen des Dokumentes	5
(2) Gegenstand der Modellierung.....	6
(3) Modellierungsmethode.....	8
VWS-Interaktionspartner- Rollenbeschreibung	8
VWS-Interaktionspartner-Rolle “Anlagenplanungswerkzeug”	9
VWS-Interaktionspartner-Rolle “Inbetriebnahmewerkzeug”	9
VWS-Interaktionspartner-Rolle “Service Requester und Service Provider”	10
Festlegungen zum allgemeinen Vorgehen bei der Modellierung	10
Nutzung des AASX-Package-Explorers zur Veranschaulichung	13
(4) Modellierung der ausgewählten Komponenten der P&P-Station.....	14
P&P-Station aus Sicht der Planung	14
P&P-Station aus Sicht der Inbetriebnahme	15
Motortyp aus Sicht der Planungs- und Inbetriebnahme	15
Motorinstanz aus Sicht der Planung- und Inbetriebnahme	16
VWS der Transporteinheit: Teilmodelle für SP/SR	18
Workstation-Typ aus Sicht der Planung.....	19
Workstation-Instanz aus Sicht der Planung	22
Workstation-Instanz: Teilmodelle für SP/SR.....	23
(5) VWS Modellierungskonzepte.....	25
Grundstruktur der VWS und ausgewählte wichtige Attribute	25
Identifizierung von Elementen.....	26
Ziel des Kapitels	26
Darstellung im Package Explorer.....	26
Erläuterungen.....	27
Semantische Zuordnung zu Modellelementen - die “semanticId”	27
Ziel des Kapitels	27
Verwendete Eingangsinformationen für das Modell.....	27
Verwendete Modellelemente des VWS-Metamodells	28
Darstellung im Package Explorer.....	28
Erläuterungen.....	28
Beziehung zwischen VWS, Asset und Asset Identification Teilmodell	29
Ziel des Kapitels	29

Verwendete Modellelemente des VWS-Metamodells	29
Erläuterungen und Darstellung im Package Explorer	30
Bill of material	30
Ziel des Kapitels	30
Verwendete Modellelemente des VWS-Metamodells	30
Erläuterungen und Darstellung im Package Explorer	30
Das Kind-Attribute – Typen/Templates und Instanzen	35
Ziel des Kapitels	35
Verwendete Modellelemente des VWS-Metamodells	35
Erläuterungen und Darstellung im Klassendiagramm und im Package Explorer	35
14.0 Verbundkomponente	37
Ziel des Kapitels	37
Verwendete Eingangsinformationen für das Modell	37
Verwendete Modellelemente des VWS-Metamodells	37
Erläuterungen	37
Concept Descriptions and Data Specifications	39
Ziel des Kapitels	39
Verwendete Modellelemente des VWS-Metamodells	39
Erläuterungen	39
Teilmodell-Templates	41
(6) Metamodell Referenzen	42
(7) Zusammenfassung und Ausblick	44
Literaturverzeichnis	45
Anhang	46
Identifikatoren	46
Instanzen	47
Package Explorer und AASX-Dateien	48
Autoren	48

(1) Ziel und Vorgehen des Dokumentes

Ziel ist es, die Modellierungsmöglichkeiten des Verwaltungsschalen (VWS)-Metamodells [PLA20] systematisch zu verwenden, sodass die Aussagekraft der VWS-Modellierungselemente erprobt werden kann. Das Dokument möchte sowohl ein Leitfaden bieten, wie die Daten der Assets in der VWS einzusortieren sind (Kochbuch für VWS Konfigurationen) als auch eine Anleitung für Nutzer von VWS sein, die die Daten in den Anwendungen verwenden. Außerdem soll ein Nachschlagewerk entstehen, mit dem beispielhafte Anwendungen der Elemente des VWS-Metamodells zu finden sind.

Das Dokument kann nur Orientierungshilfe geben, da es sehr viele Möglichkeiten gibt, die Modellelemente zu verwenden und es schwierig ist, eine Konsistenz mit der sich weiterentwickelnden VWS-Spezifikation aufrecht zu erhalten.

Das Dokument zeigt Best Practice Muster für die Umsetzung von VWS von konkreten Assets und spricht deshalb folgende Zielgruppen an:

- Produktmanager
- Entwickler, die die Details bei der Konfiguration der VWS umsetzen müssen
- Maschinenbauer, die mit den Informationen der Zulieferer-Assets und deren Dokumentation umgehen müssen
- Entwickler der technischen Dokumentation

Das Dokument ist wie folgt aufgebaut:

Kapitel 2: Kurze Einführung des Demonstrators, dessen Komponenten als VWS beschrieben werden. Die Anwendung des VWS-Metamodells wird in weiteren Kapiteln anhand dieser beispielhaften VWS erläutert.

Kapitel 3: VWS sollen Informationen über ein Asset über den gesamten Asset-Lebenszyklus hinweg bereitstellen können. Es kann durchaus sein, dass unterschiedliche VWS-Nutzer bzw. VWS-Interaktionspartner

unterschiedliche Sichten auf ein Asset haben und unterschiedliche Darstellungen des Assets in seiner VWS erwarten. Um Vielfalt und starke Ausdrucksmöglichkeiten des VWS-Metamodells zu veranschaulichen werden in diesem Kapitel beispielhaft drei VWS-Interaktionspartner-Rollen eingeführt. Die definierten VWS-Interaktionspartner-Rollen erheben nicht den Anspruch auf Allgemeingültigkeit und Vollständigkeit. Sie dienen im Kontext dieses Dokument der besseren Veranschaulichung von VWS-Modellierungskonzepten.

Kapitel 4: Modellierung von unterschiedlichen VWS-Interaktionspartner-bezogenen Aspekten (s. Kapitel 3) der ausgewählten Assets (s. Kapitel 2) als VWS-Teilmodelle. Dabei erfolgt keine Erläuterung der Elemente des VWS-Metamodells. Es wird nur gezeigt, welche VWS-Metamodell-Elemente in den jeweiligen Teilmodellen verwendet wurden. Dabei wird auf die VWS-Metamodell-Erläuterungen im Kapitel 5 verwiesen.

Kapitel 5: Beinhaltet Erläuterungen des VWS-Metamodells und deren Nutzung bei der Abbildung der Sachverhalte der physischen Welt in die Informationswelt

Kapitel 6: Beinhaltet eine alphabetische Liste wichtiger Modellelemente, die in den vorherigen Kapiteln verwendet wurden. Es werden jeweils die Referenzen auf die entsprechenden Unterkapitel in Kapitel 4 und 5 bereitgestellt.

Kapitel 7: Zusammenfassung und Ausblick

Die Version 3.0 RC01 des VWS-Metamodells ist die Grundlage der hier verwendeten Elemente des Metamodells in den textuellen Beschreibungen [PLA20].

Es werden alle VWS in dem AASX-Austauschformat des Open Source Werkzeugs AASX Package Explorer bereitgestellt. Zum Zeitpunkt der Erstellung der Beispiele hat der Package Explorer noch nicht alle Details des Metamodells der V3.0 RC01 korrekt

umgesetzt. Deshalb kann es Abweichungen zu den textuellen Beschreibungen geben. Da es hier um ein prinzipielles Verständnis geht, sollten eventuelle Inkonsistenzen toleriert werden.

(2) Gegenstand der Modellierung

Für die Modellierung wird exemplarisch die Pick and Place Station (P&P-Station) der Otto-von-Guericke-Universität Magdeburg (OvGU) (Abbildung 2-1) verwendet.

In Abbildung 2-2 stellt die steuerungstechnische Struktur der P&P-Station vor. Der Demonstrator besteht aus den drei Bearbeitungsstationen (Workstation) sowie den

Steuerungen für den Deltaroboter und den Bearbeitungsstationen. Zum Deltaroboter gehören die drei Motoren mit ihren jeweiligen Motor-Controllern. Von der P&P-Station können mehrere Produkte bearbeitet werden. Jede Steuerung steuert die jeweilige Station beziehungsweise die Motor-Controller. Sie sind alle mit einer VWS verbunden. Auch die Produkte haben jeweils eine VWS. Die VWS interagieren untereinander mit der I4.0 Sprache und organisieren den Ablauf in der P&P-Station. Zusätzlich gibt es die Möglichkeit in die VWS über ein Browser-Schnittstelle hineinzuschauen (zurzeit nur statische Daten).

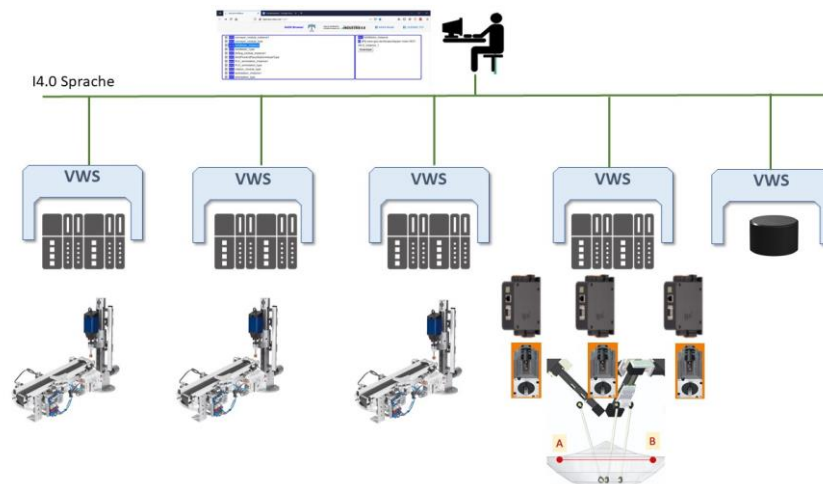


Abbildung 2-2: Automatisierungsarchitektur der Pick and Place Station



Abbildung 2-1: Pick and Place Demonstrator der OvGU

Zunächst wird der Demonstrator als ein Ganzes mit einer eigenen VWS versehen. Da die P&P-Station durchaus komplex ist und aus Komponenten besteht, die unter Umständen als eigenständige I4.0-Komponenten betrachtet werden können, erhalten einige ausgewählte Komponenten eigene VWS. Die VWS der P&P-Station beinhaltet die Referenzen auf die einzelnen VWS der Komponenten. Damit soll die Anwendung des Prinzips der geschachtelten/zusammengesetzten Verwaltungsschale (Verbundkomponente) demonstriert werden.

Für die exemplarische Modellierung wurden als eigenständige Teilkomponenten der P&P-Station die Transporteinheit und die Bearbeitungsstation ausgewählt. Die Trans-

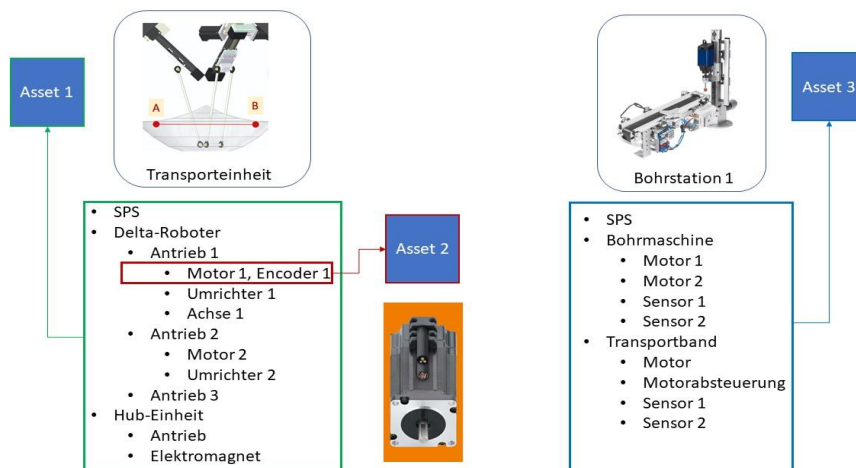


Abbildung 2-3: Auswahl von Assets, die mit eigenen Verwaltungsschalen in diesem Dokument versehen werden (Verfeinerung siehe Kapitel 3.2)

In diesem Dokument soll gezeigt werden, wie ausgewählte Komponenten (Assets) dieses Demonstrators in I4.0-Komponenten überführt werden, in dem diese eigene VWS erhalten. Durch die entsprechenden VWS werden die erwähnten Assets in einem I4.0-System repräsentiert. Dabei werden relevante Informationen dieser Assets als Teilmodelle gemäß dem VWS-Metamodell in der Informationswelt abgebildet und in den entsprechenden VWS den potentiellen VWS-Nutzern zur Verfügung gestellt.

porteinheit besteht aus einem Delta-Roboter, einem elektromechanischen Linearaktuator und zugehörigen Steuerungseinheiten und wird bei der Modellierung als ein Asset betrachtet. Ebenso wird eine Workstation als ein Asset betrachtet, die aus einem Transportband und einer Bohrmaschine besteht.

An den Beispielen dieser Assets soll die Ausdruckstärke der Modellierungsmöglichkeiten des VWS-Metamodells, diverse Detaillierungsgrade und Modellierungstechniken bei der Erstellung der Teilmodelle und Verwaltungsschalen zur Geltung kommen. Dies soll

durch die Einführung unterschiedlicher VWS-Interaktionspartner-Rollen, welche auf diese Rollen zugeschnittene Teilmodell-Inhalte erfordern, verdeutlicht werden. Ebenso soll damit die Flexibilität des VWS-Metamodells hinsichtlich der Einteilung der VWS-Inhalte für unterschiedliche Stakeholder dargestellt werden, z. B. durch die Verwendung des Konzeptes View.

Zusätzlich wird das Konzept der technologie-neutralen, durchgängigen und interoperablen Beschreibung und des Zugriffs auf die Informationen am Beispiel der Modellierung eines elektrischen Motors gezeigt.

Insgesamt werden für die Modellierung in diesem Dokument folgende Assets ausgewählt:

- Asset 1: Transporteinheit insgesamt
- Asset 2: Motor mit Encoder des Delta-Roboters
- Asset 3: Workstation insgesamt
- Asset 4: P&P-Station als Ganzes

Diese Assets werden mit eigenen VWS ausgestattet. Die bei der Modellierung der VWS entstandenen Teilmodelle haben keinen Anspruch auf die fachliche Korrektheit und Vollständigkeit. Die vollständigen und widerspruchsfreien Teilmodelle für die Abbildung von diversen Aspekten von in diesem Dokument betrachteten Assets entstehen derzeit in Arbeitsgruppen und Gremien der jeweiligen Expertenorganisationen. Nach Möglichkeit und dem Kenntnisstand der Autoren wird im Dokument auf die jeweiligen Arbeiten verwiesen.

(3) Modellierungsmethode

In diesem Kapitel werden Festlegungen zum allgemeinen Vorgehen bei der Modellierung und Erstellung von VWS für die exemplarisch ausgewählten Assets des Demonstrators getroffen. Es sei noch einmal darauf hingewiesen, dass es sich hier um eine beispielhafte Anwendung der Modellierungsmöglichkeiten der VWS handelt. Es besteht kein

normativer Anspruch. Trotzdem ist beabsichtigt, gute praktische Beispiellösungen anzubieten.

VWS-Interaktionspartner-Rollenbeschreibung

VWS sollen Informationen über ein Asset über den gesamten Asset-Lebenszyklus hinweg bereitstellen. Es kann durchaus sein, dass unterschiedliche Interaktionspartner von VWS unterschiedliche Sichten auf ein Asset haben und unterschiedliche Darstellungen des Assets in seiner VWS erwarten. Je nachdem können unterschiedliche Merkmale und Teilmodelle relevant sein. Um die Vielfalt und Stärken der Ausdrucksmöglichkeiten des VWS-Metamodells zu veranschaulichen, werden in diesem Kapitel drei Rollen der Interaktionspartner von VWS eingeführt. Als mögliche Interaktionspartner werden zunächst Werkzeuge für Planung und Inbetriebnahme einer Anlage gesehen, in der das jeweilige Asset verwendet werden soll. Als dritte Interaktionspartner-Rolle wurden Service Requester und Service Provider eingeführt. Da es vorstellbar ist, dass einige Komponenten der P&P-Station in bestimmten I4.0-Anwendungsszenarien als autonome Dienstleister betrachtet werden können, wie z. B. Transporteinheit und Workstation als Anbieter von Transport- oder Bohrdienstleistungen. Um diese VWS-Interaktionspartner-Rollen zu charakterisieren, werden die eventuellen Anforderungen an die VWS-Inhalte aus Sicht dieser Rollen spezifiziert. Die Definition von VWS-Interaktionspartner-Rollen, die Auswahl von Teilmodellen und ihren Inhalten haben keinen allgemeingültigen Charakter und gelten nur im Kontext dieses Dokument.

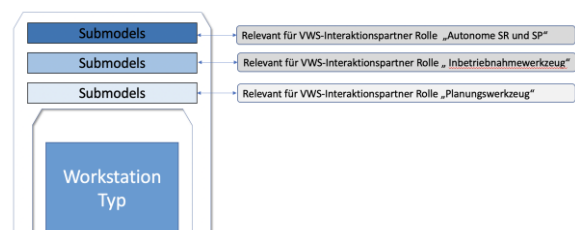


Abbildung 3-1: VWS-Interaktionspartner-Rollen

VWS-Interaktionspartner-Rolle “Anlagenplanungswerkzeug”

Die Planung einer komplexen Anlage stellt eine herausragende Rolle innerhalb der Lebensphasen einer Anlage dar. Der Planer einer Anlage bzw. der Anlagenkonstrukteur erstellt ein Pflichtenheft, in dem prinzipielle Anforderungen an die Eigenschaften der geplanten Anlage definiert werden.

Hersteller von Komponenten erstellen eine datentechnische Beschreibung ihrer Komponenten. Diese Beschreibung soll im Sinne von Konzepten der Industrie 4.0 in Form einer VWS mit entsprechenden Teilmodellen zur Verfügung gestellt werden. Diese VWS beschreibt den Asset-Typen.

Der Abgleich der angeforderten und zugesicherten Eigenschaften von Komponenten, beziehungsweise die Prüfung auf die Erfüllung der geforderten Eigenschaften soll im Sinne von I4.0 mit Hilfe eines Planungswerkzeuges möglichst automatisiert bzw. teilautomatisiert erfolgen. Dies erfordert eine durchgängige und interoperable Beschreibung der Eigenschaften in der VWS des Asset-Types in Form von entsprechenden Teilmodellen der Komponenten. Diese Beschreibung soll von dem Planungswerkzeug eingelesen und korrekt interpretiert werden (Abbildung 3-2).

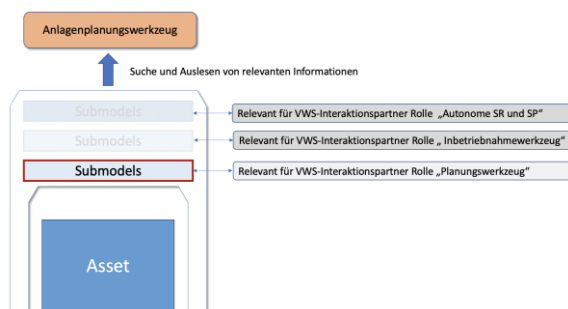


Abbildung 3-2: Auswahl von Teilmodellen für das Anlagenplanungswerkzeug

Teilmodelle für die Rolle Planer:

- Technical Data für die Planer einer Anlage

- Elektrische, pneumatische, hydraulische und mechanische Schnittstellen
- Einsatzbedingungen (z. B. klimatisch, Betriebstemperaturen)
- Funktionale Ein- und Ausgänge
- Montagevorschriften
- Wartungsmanagement

VWS-Interaktionspartner-Rolle “Inbetriebnahmewerkzeug”

Eine VWS kann auch eine technologieunabhängige interoperable Schnittstelle eines Assets bereitstellen, die eine Kopplung des Assets mit Engineering- und Steuerungssoftware für das Testen und die Inbetriebnahme ermöglicht.

Dabei soll die VWS als eine Asset-Schnittstelle dem Inbetriebnehmer ermöglichen, das Asset zu parametrieren und einen Funktionstest des Assets durchzuführen. Dafür sollen die Funktionalitäten des Assets aufgerufen und gesteuert werden können (Abbildung 3-3).

Zum Nachweis der angeforderten Prozessfähigkeit der Anlage soll es dem Inbetriebnehmer möglich sein, und Asset-Funktionalitäten zu überprüfen und die relevanten Sensordaten auszulesen. Diese sollen z. B. für die Erstellung der Kennlinien verwendet werden können, die weitere Aufschlüsse bezüglich der Funktionstüchtigkeit der Anlage ermöglichen sollen.

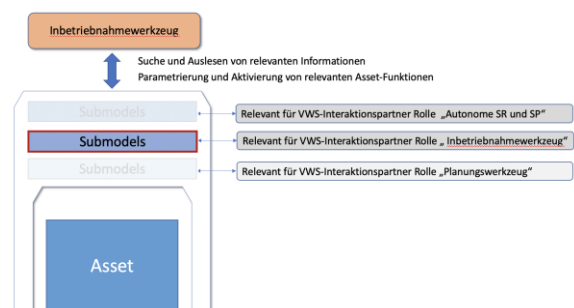


Abbildung 3-3: Auswahl von Teilmodellen für das Inbetriebnahmewerkzeug

Teilmodelle für die Rolle Inbetriebnehmer:

- Typenschild
- Dokumentationen
- Technical Data für Inbetriebnehmer
 - Mess- und Stellbereiche
 - Elektrische und kommunikationstechnische Schnittstellen
- Teilmodelle für funktionale Daten
 - Aktualdaten von Mess- und Stellwerten, z.B. "Motorstrom", "Position"
 - Einstellparameter der Geräte und Komponenten
 - Status- und Diagnosewerte über die Anlage und über die Geräte und Komponente

VWS-Interaktionspartner-Rolle "Service Requester und Service Provider"

Eine semantisch eindeutige maschinenlesbare Beschreibung der Eigenschaften und Fähigkeiten von Assets, so wie diese die Teilmodelle für Planer und Inbetriebnehmer bieten, ist ein wichtiger Schritt zur Erhöhung der Interoperabilität und Integration von Komponenten der Automatisierungstechnik. Anwendungsbereich ist die Produktionsstätte des Betreibers, welcher die eigentliche Produktionsfähigkeit des Assets in seine Produktions- und Arbeitspläne einbinden möchte und somit eine einfache Integration in ERP- oder MES-Systemen anstrebt. Jedoch erfordern bestimmte I4.0-Anwendungsszenarien eine gewisse Autonomie der Produktionssysteme und ihrer Komponenten, z. B. dynamische Optimierung der Produktionsauslastung, Vermeidung von Stillstandszeiten durch Maschinenausfälle, auftragsgesteuerte Produktion sowie die dynamische Orchestrierung von Produktionsressourcen für eine kostenoptimierte Produktion in Losgröße Eins. Zu diesem Zweck sollen die Teilmodelle der VWS die Informationen über die Asset-Fähigkeiten oder die von Assets angebotenen Dienstleistungen

den Interaktionspartnern bereitstellen können.

Die Assets werden in solchen Szenarien zu autonomen Service Provider oder Service Requester (Abbildung 3-4). Die VWS ermöglichen eine Teilhabe an horizontalen protokollbasierten Interaktionen, wie sie beispielsweise in der VDI/VDE 2193 - Richtlinie für das Ausschreibungsverfahren definiert sind.

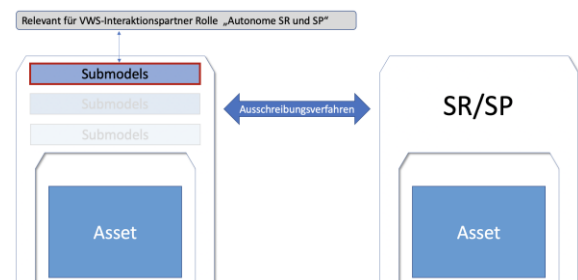


Abbildung 3-4: Auswahl von Teilmodellen für die autonome VWS

Die Teilmodelle sollen folgende Informationen bereitstellen können:

- Welchen Zweck erfüllt das Asset?
- Was sind die Fähigkeiten des Assets und welche Dienstleistungen kann das Asset/ I4.0-Komponente anbieten?
- Technische Daten für die detaillierte Überprüfung der Anforderungen (siehe Planer)

Festlegungen zum allgemeinen Vorgehen bei der Modellierung

In diesem Dokument wird beschrieben, wie eine P&P-Station (der prinzipielle Aufbau dargestellt in Abbildung 2-2) in einem I40-System digital repräsentiert und mit eigenen VWS ausgestattet werden können. Die folgenden Annahmen und Anforderungen liegen der Struktur der VWS der P&P Station zu Grunde:

- I4.0-Komponenten bestehen aus einem Asset und seiner VWS. Assets können durch eigene VWS oder in

Teilmodellen einer VWS beschrieben werden, in der dieses Asset ein Bestandteil ist. Dies wird durch SelfManagedEntity und CoManagedEntity beschrieben. Dies wird hier beispielhaft in einer Bill of Material (BoM) gezeigt.

- Einige VWS eines „self-managed“-Assets haben eine Stückliste (Teilmodell mit semanticId BOM)
- Das Gerüst mit seinen mechanischen Teilen wird als Teilmodell der P&P-Station-VWS beschrieben und besitzt keine eigene VWS.
- Als Komponenten werden in diesem Dokument jeweils die Bestandteile der P&P-Station, bzw. allgemein die Bestandteile von Assets bezeichnet.
- Die Assets werden mit einer oder mehreren VWS versehen, die als eigenständige Komponente betrachtet werden¹. Dahinter steht die Idee, dass die eigenständige Komponente auch eine in sich konsistente Beschreibung hat, die z. B. vom Zulieferer bereitgestellt wird. Diese Assets werden als „Self-ManagedEntity“ (5.5) bezeichnet.
- P&P-Station als Ganzes wird als ein Asset betrachtet und erhält eine eigene VWS. Die VWS der P&P-Station beinhaltet als sogenannte Verbundkomponente in einem speziellen Teilmodell Referenzen/Verweise auf die VWS der anderen modellierten Assets
- Als Asset werden die tatsächlich in die P&P-Station eingebauten Automatisierungskomponenten verstanden, sozusagen die Instanzen. Dementsprechend werden die VWS dieser Assets als die VWS von Asset-Instanzen verstanden.
- Zusätzlich wird es davon ausgegangen, dass es Asset-Typen existieren. Vollständigkeitshalber werden in diesem Dokument in Ergänzung zu VWS

der Asset-Instanzen ausgewählte VWS der entsprechenden Asset-Typen modelliert. VWS ist eine digitale Abbildung eines Assets. In unterschiedlichen Lebenszyklusphasen wird diese digitale Abbildung von unterschiedlichen Interaktionspartnern benutzt.

- Damit die unterschiedlichen Interaktionspartnern die für sie relevanten Teilmodelle identifizieren können, wird das Konzept „View“ für die Strukturierung von VWS-Inhalten beispielhaft eingeführt. Views können auch verwendet werden, um Informationen zu gruppieren, die für den Asset Type und die Asset Instanz von Bedeutung sind. Dann können beide Aspekte in einer VWS enthalten sein.
- Jede VWS hat je ein Teilmodell für Nameplate, Dokumentation und TechnicalData

Abbildung 3-5 zeigt auszugsweise eine mögliche Abbildung der ausgewählten Komponenten der P&P-Station in VWS. Es wird prinzipiell zwischen dem Asset-Typ und der Asset-Instanz unterschieden. Der Asset-Typ wird durch die Katalogdaten und weiteren Planungsunterlagen beschrieben, die noch keine spezifischen Angaben zum individuellen Gerät oder zur individuellen Komponente haben (z. B. die mögliche Spannungsversorgungsvarianten).

¹ In dieser Beschreibung wird nur ein Subset der möglichen VWS tatsächlich aufgenommen. Eine vollständige

Beschreibung aller VWS würde den Rahmen des Dokumentes sprengen.

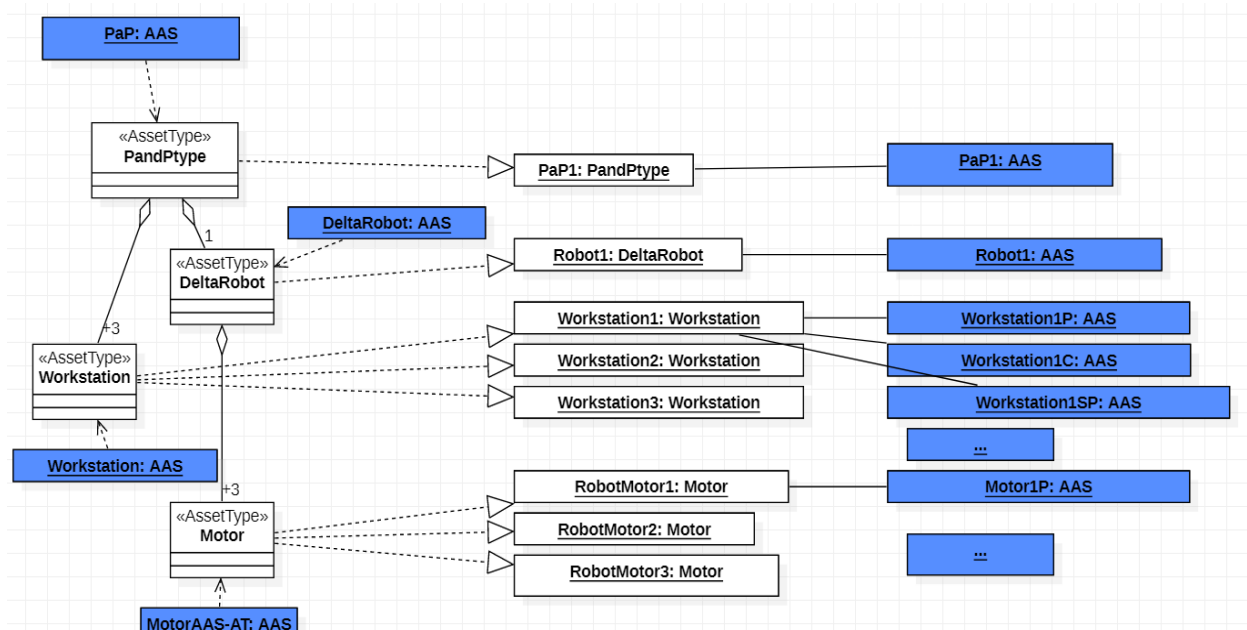


Abbildung 3-5: Übersicht über die VWS Modellierung der P&P-Station (Ausschnitt)

In Abbildung 3-2 sind auszugsweise nur die P&P-Station selbst (PandPtype)², der Delta-Roboter (DeltaRobot), die Workstation und der Motor dargestellt. Jeder Asset-Typ ist durch eine VWS repräsentiert (PaP: AAS), die z. B. beim Verkauf mitgeliefert wird. Betrachtete Asset-Typen sind u. a. die P&P-Station selbst (PaP1: PandPtype), ein Delta-Roboter (Robot1: DeltaRobot), eine Workstation (Workstation1: Workstation, ...) und ein Motor (Motor1: Motor, ...). Zwischen Asset-Instanz und VWS von einer Asset-Instanz und VWS eines Asset-Types und gibt es Beziehungen, die im Kapitel 5.6 näher beschrieben werden. Diese sind hier als Links (zwischen den Asset-Instanzen und der VWS – durchgezogene Linie) und als Dependency (zwischen den Asset-Typen und VWS – durchgezogene Linie) beschrieben.

In den Abbildungen 3-6 bis 3-8 wird die Auswahl der in diesem Dokument beispielhaft beschriebenen VWS dargestellt.

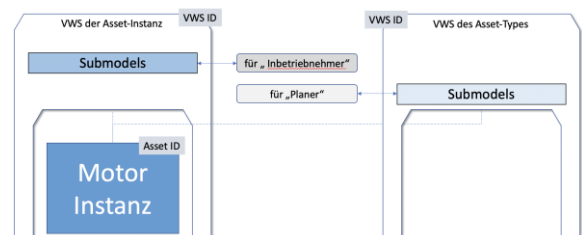


Abbildung 3-6: Die in Kapitel 4 für die Abbildung des Motors erstellten Teilmodelle

² Die Angaben in den Klammern dieses Abschnitts sind die Bezeichner der Klassen und Objekte in Abbildung 3-2

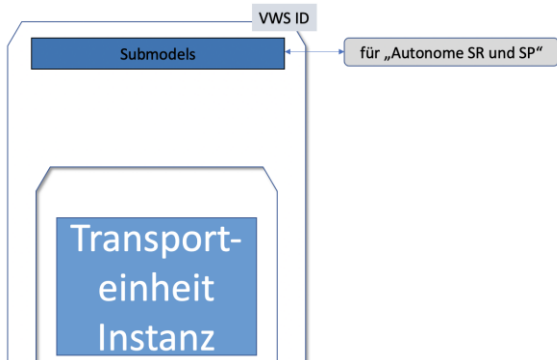


Abbildung 3-7: Die in Kapitel 4 für die Abbildung der Transporteinheit erstellten Teilmodelle

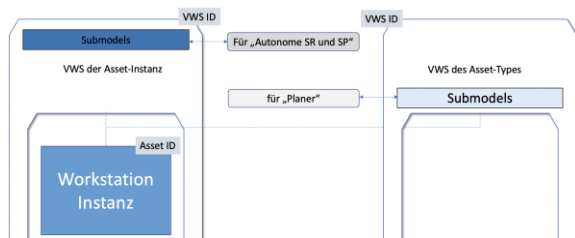


Abbildung 3-8: Die in Kapitel 4 für die Abbildung der Workstation erstellten Teilmodelle

Für die Modellierung zusammengesetzter Assets werden hier zwei verschiedene Möglichkeiten verwendet. Dies soll den Gestaltungsraum beim Entwurf verdeutlichen.

Abbildung 3-9 zeigt, wie weitere Informationen zu einzelnen Bauteilen des Motors (hier der Encoder) in einem Teilmodell des zusammengesetzten Assets beschrieben werden können. Dies ist dann sinnvoll, wenn die Einzelteile dauerhaft miteinander verbunden sind und die Assets in der Regel nur zusammengesetzt erworben werden. Auch kann der Motorhersteller durch diesen Modellierungsansatz Informationen zum Encoder einbinden (bspw. die Positioniergenauigkeit) ohne dessen VWS an den nächsten Wertschöpfungspartner weiterzureichen. Die unterschiedlichen Teilkomponenten werden in separierten Teilmodellen beschrieben. Dies ist sinnvoll, wenn die Komponente nur als Ganzes erworben werden kann. Die Beschreibung befindet sich in Kapitel 4.3 und 4.4.

Abbildung 3-10 zeigt einen anderen Weg. Hier werden die Teilkomponenten mit eigenen VWS ausgestattet. Dies ist sinnvoll, wenn die Teilkomponenten separat erworben und beim Nutzer zusammengesetzt werden. Die Beschreibung befindet sich im Kapitel 4.6 und 4.7.

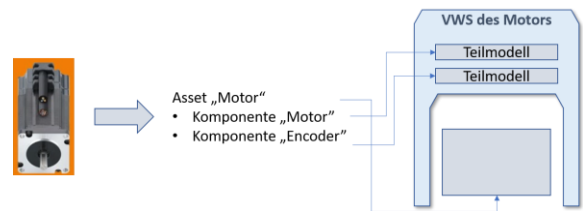


Abbildung 3-9: Built-in Assets der I4.0-Komponente „Motor with integrated encoder“ werden als Teilmodelle in der VWS modelliert

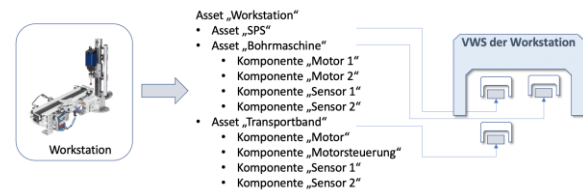


Abbildung 3-10: VWS der Workstation verweist auf die VWS von Built-in Assets der Workstation

Nutzung des AASX-Package-Explorers zur Veranschaulichung

Die dargestellten Modellierungen sind generell als Beispiele zu verstehen, die die Modellierungselemente und -möglichkeiten veranschaulichen. Sie sind keine Vorgaben. Deshalb werden auch manchmal verschiedene Modellierungsmöglichkeiten in verschiedenen Beispielen verwendet.

Zur Veranschaulichung werden alle Beispiele mit dem AASX-Package-Explorer in der zurzeit verfügbaren Version verwendet. Die AASX-Dateien sind unter <http://liabroker.ddns.net:51001/> einsehbar.

Nicht alle beispielhaften Umsetzungen im Package Explorer entsprechen zurzeit der Version 3.0 RC01 der VWS-Spezifikation. Trotzdem sind die Autoren der Meinung, dass die AASX-Modelle und die daraus entstandenen Screenshot das Verständnis unterstützen.

(4) Modellierung der ausgewählten Komponenten der P&P-Station

Dieses Kapitel beschreibt die exemplarische Modellierung der in Kapitel 2 ausgewählten Assets in Form von VWS und Teilmodellen. Der dargestellten Modellierung liegt die im Kapitel 3 beschriebene Modellierungsmethode zugrunde.

P&P-Station aus Sicht der Planung

Der Demonstrator ist eine P&P-Station, in der drei Workstations verschiedene Fähigkeiten anbieten. Ein Deltaroboter übernimmt

die Transportaufgaben zwischen den Workstations. Ein Planer hat zunächst eine grobe Sicht und betrachtet den Deltaroboter und die Workstations als jeweils eine Komponente. In diesem Sinne ist die P&P-Station eine I4.0 Verbundkomponente (Abbildung 4-1). Eine Stückliste (Bill of Material - BoM) listet die Komponenten auf. Dies wird näher in Kapitel 5.6 erläutert. Wie für diese Modellierung festgelegt (Kapitel 3.2), hat die VWS der P&P-Station die Teilmodelle Nameplate, Documentation und TechnicalData. Die Teilmodelle Nameplate und Documentation beruhen auf den entsprechenden Teilmodell-Templates. Nähere Informationen sind in [DNI20] und [DD20] enthalten. In diesen Teilmodellen wird die Station als Ganzes beschrieben, wie z. B. die elektrischen Anschlusswerte und die geometrischen Abmessungen. Der Planer braucht jedoch auch die technischen Details der Komponenten. Deshalb hat die VWS der P&P-Station Beziehungen (Relationship) zu den VWS der Komponenten. Diese sind in einem eigenen Teilmodell "CompositeRelationship" hinterlegt (nähere Beschreibungen dazu in Kapitel 5.8).

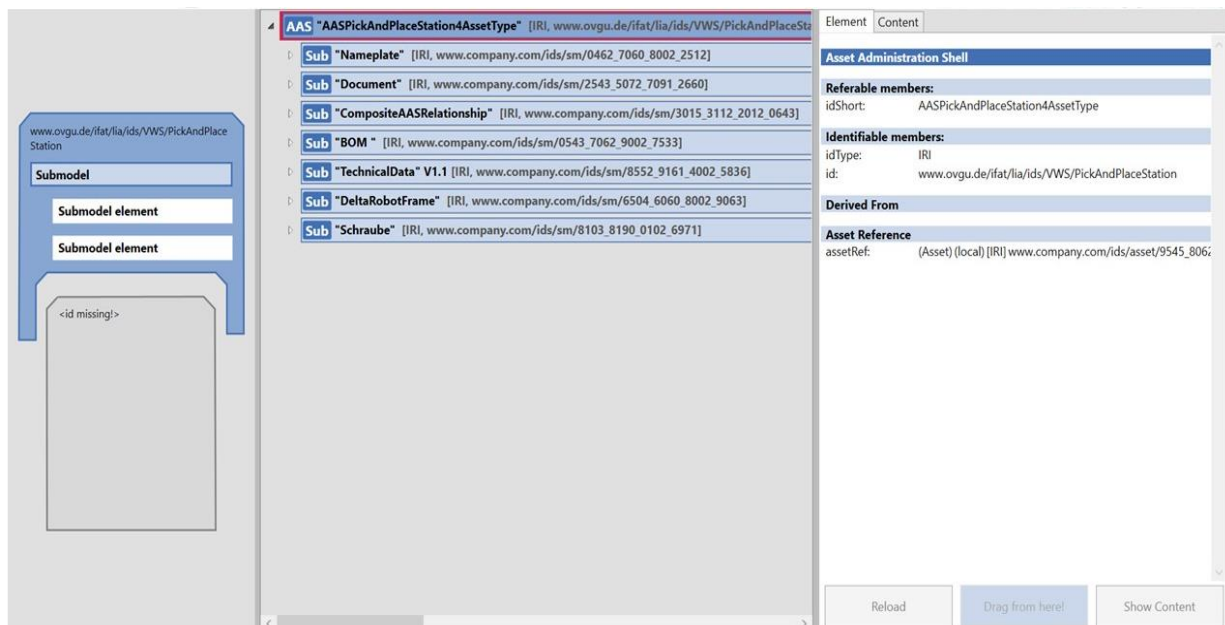


Abbildung 4-1: Teilmodell-Übersicht P&P-Station Typ

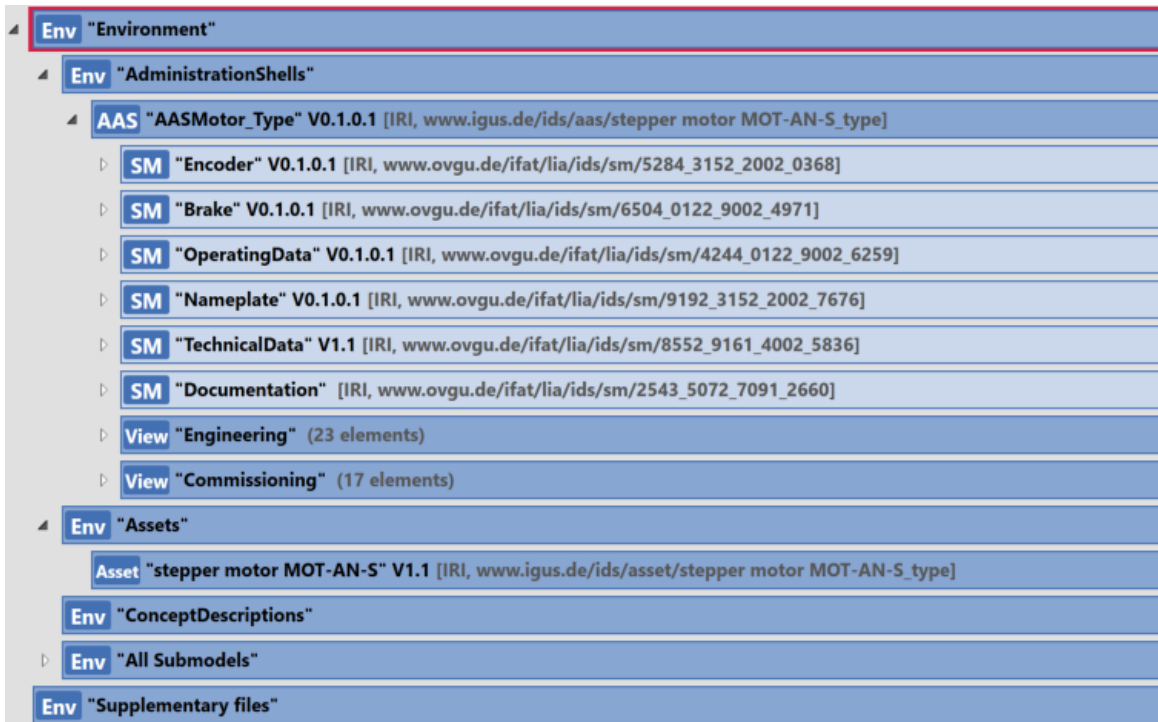


Abbildung 4-2: Ansicht der VWS des Motortyps

Über den Einstiegspunkt der P&P-Station VWS werden damit sowohl alle Komponenten, die zur Station gehören als auch ihre Daten auffindbar. In ihnen sind die typbezogenen Daten der Komponenten enthalten.

P&P-Station aus Sicht der Inbetriebnahme

Sind die konstruktiven Arbeiten der Planung beendet und die Komponenten bestellt und geliefert, so muss die Montage und Inbetriebnahme durchgeführt werden. Die VWS der P&P-Station Instanz verweist nicht wie ihr Typ auf jeweils eine Asset-Typ bezogene Komponenten-VWS, sondern auf alle instanzbezogenen VWS, in diesem Beispiel auf drei Workstations und einen Deltaroboter. Wie in den folgenden Kapiteln beschrieben, unterscheiden sich die Teilmodell-Einträge der I4.0-Komponenten zwischen denen die den Asset-Typ und die Asset-Instanz beschreiben. Die VWS der Instanz der P&P-Station verwendet das VWS-Attribut “derivedFrom”, in dem die AASId der VWS des P&P-Station Types eingetragen wird.

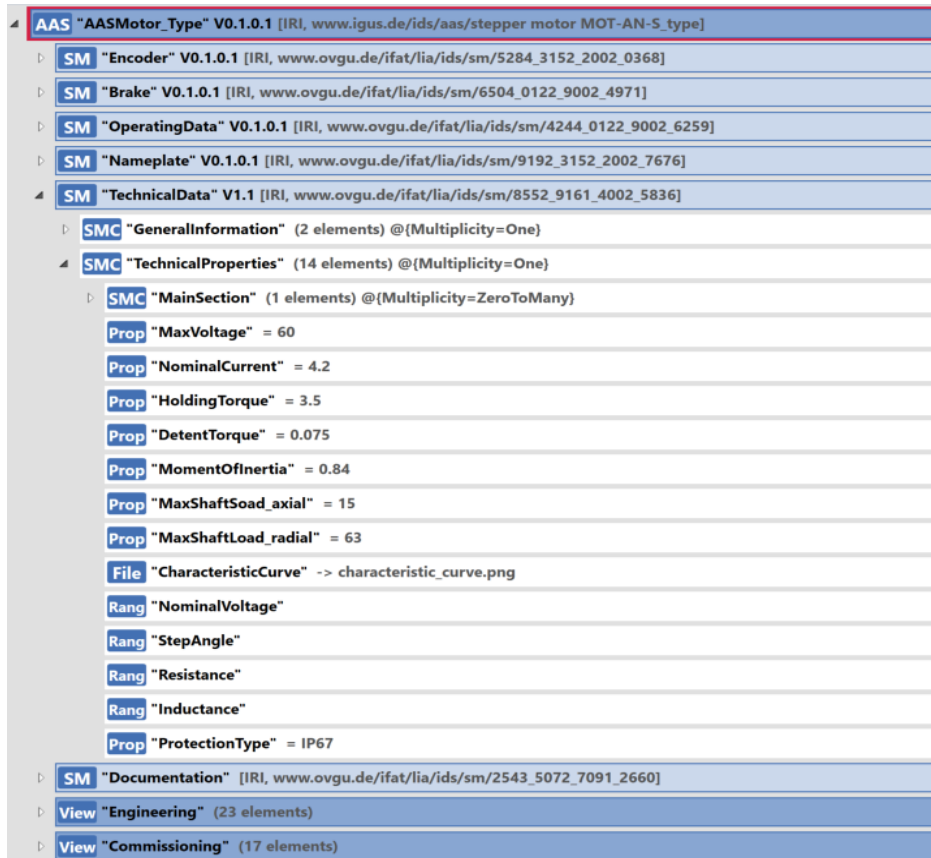
“derivedFrom” beschreibt keine klassische objektorientierte Typ/Instanz-Beziehung.

Hier wird lediglich ausgedrückt, dass die Asset-Instanz-VWS einmalig bei der Instanziierung Elemente aus dem Asset-Typ-VWS übernimmt oder durch Referenz auf die Daten der anderen VWS diese zugreifbar macht. Dabei können die Elemente der VWS verändert, gekürzt oder ergänzt werden.

Motortyp aus Sicht der Planungs- und Inbetriebnahme

Der Demonstrator beinhaltet drei Schrittmotoren zur Ansteuerung des Delta-Roboters. Alle drei Motoren sind Ausprägungen desselben Produkttyps. Daher wird zunächst eine VWS für diesen Typ modelliert. Abbildung 4-2 zeigt die Ansicht der VWS des Motortyps im AASX Package Explorer.

Angelegt sind ein Asset sowie eine zugehörige VWS. Das Asset ist vom AssetKind



▲	AAS "AASMotor_Type" V0.1.0.1 [IRI, www.igus.de/ids/aas/stepper motor MOT-AN-S_type]
▷	SM "Encoder" V0.1.0.1 [IRI, www.ovgu.de/ifat/lia/ids/sm/5284_3152_2002_0368]
▷	SM "Brake" V0.1.0.1 [IRI, www.ovgu.de/ifat/lia/ids/sm/6504_0122_9002_4971]
▷	SM "OperatingData" V0.1.0.1 [IRI, www.ovgu.de/ifat/lia/ids/sm/4244_0122_9002_6259]
▷	SM "Nameplate" V0.1.0.1 [IRI, www.ovgu.de/ifat/lia/ids/sm/9192_3152_2002_7676]
▲	SM "TechnicalData" V1.1 [IRI, www.ovgu.de/ifat/lia/ids/sm/8552_9161_4002_5836]
▷	SMC "GeneralInformation" (2 elements) @({Multiplicity=One})
▲	SMC "TechnicalProperties" (14 elements) @({Multiplicity=One})
▷	SMC "MainSection" (1 elements) @({Multiplicity=ZeroToMany})
	Prop "MaxVoltage" = 60
	Prop "NominalCurrent" = 4.2
	Prop "HoldingTorque" = 3.5
	Prop "DetentTorque" = 0.075
	Prop "MomentOfInertia" = 0.84
	Prop "MaxShaftSoad_axial" = 15
	Prop "MaxShaftLoad_radial" = 63
	File "CharacteristicCurve" -> characteristic_curve.png
	Rang "NominalVoltage"
	Rang "StepAngle"
	Rang "Resistance"
	Rang "Inductance"
	Prop "ProtectionType" = IP67
▷	SM "Documentation" [IRI, www.ovgu.de/ifat/lia/ids/sm/2543_5072_7091_2660]
▷	View "Engineering" (23 elements)
▷	View "Commissioning" (17 elements)

Abbildung 4-3: Teilmodell *TechnicalData* mit verschiedenen *SubmodelElement*-Typen

Type, da es den Typen des Motors repräsentiert. Die VWS enthält sechs Teilmodelle, deren Wahl sich an der Gliederung der Dokumentationsunterlagen [IGUS03] des Motors orientiert. Sämtliche IDs der Teilmodelle fallen unter die Authority www.ovgu.de.

Alle Teilmodelle sind hier vom *ModelingKind Instance*. Die *SubmodelElements* der Teilmodelle vom *Kind Instance* sind ebenfalls vom *Kind Instance*. Die des einen Teilmodells vom *Kind Template* sind ebenfalls vom *Kind Template*. Alle *SubmodelElements* tragen *semanticIds*, die als *GlobalReference* vom Typ *IRDI* oder *IRI* auf einen Standard wie z. B. ECLASS oder IEC61360-CDD verweisen.

Abbildung 4-3 zeigt exemplarisch für das Teilmodell *TechnicalData*, dass verschiedene Subtypen von *SubmodelElements* Verwendung finden. So enthält dieses Teilmodell neben dem häufig verwendeten Typ *Property* auch die Typen *Range* und *File*, um

dem *value*-Attribut einer *Property* einen Bereich zuzuweisen bzw. auf eine Datei zu verweisen. Die *Property* mit dem Bezeichner *protection type* verwendet zudem eine *ValueID*, um für deren Ausprägung auf einen standardisierten Wert aus einer *ValueList* zu verweisen (bspw. IP65, IP67). Einzelne *SubmodelElements* sind einem der beiden *Views* „Engineering“ oder „Commissioning“ zugewiesen. Verschiedene Nutzerrollen können damit gezielt auf die für sie als relevant erachteten *SubmodelElements* zugreifen.

Motorinstanz aus Sicht der Planungs- und Inbetriebnahme

Abbildung 4-4 zeigt die Ansicht der VWS einer Motorinstanz im AASX Package Explorer. Hiervon liegen insgesamt drei vor. Angelegt sind ein Asset sowie eine zugehörige VWS. Das Asset ist vom *ModelingKind Instance*, da es eine Instanz des Motors

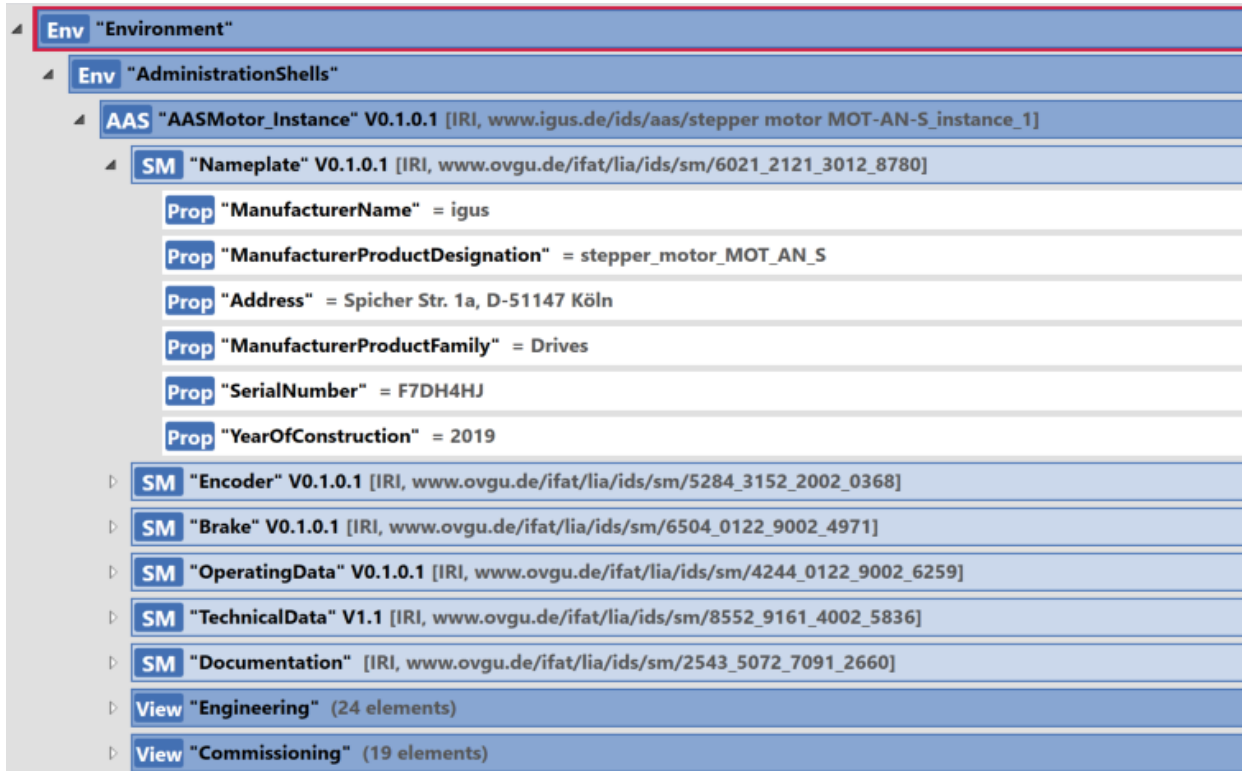


Abbildung 4-4: Ansicht der VWS einer Motorinstanz

repräsentiert. Die VWS enthält nun nur noch fünf Teilmodelle, da bezüglich des Teilmodells *Nameplate* nun keines vom *Modeling-Kind Template* mehr benötigt wird. Stattdessen haben nun alle *SubmodelElements* von *Nameplate* konkrete Werte und sind daher in einem einzigen Teilmodell vom *Modeling-Kind Instance* zusammengefasst.

Auch die Instanz-VWS enthält wieder die beiden *Views* „Engineering“ und „Commissioning“. In Abbildung 4-5 ist exemplarisch die *View* „Commissioning“ aufgeklappt. Zu sehen sind die Verweise auf *SubmodelElements* der verschiedenen Teilmodelle. Die Zuordnung zu den *Views* ist als beispielhaft anzusehen und wird je nach Verwendungszweck des Assets und dessen VWS unterschiedlich ausfallen. Das Metamodell selbst kann dafür keine Kriterien bieten. In den übrigen Aspekten unterscheidet sich die Instanz-VWS nicht von der zuvor beschriebenen Typ-VWS.

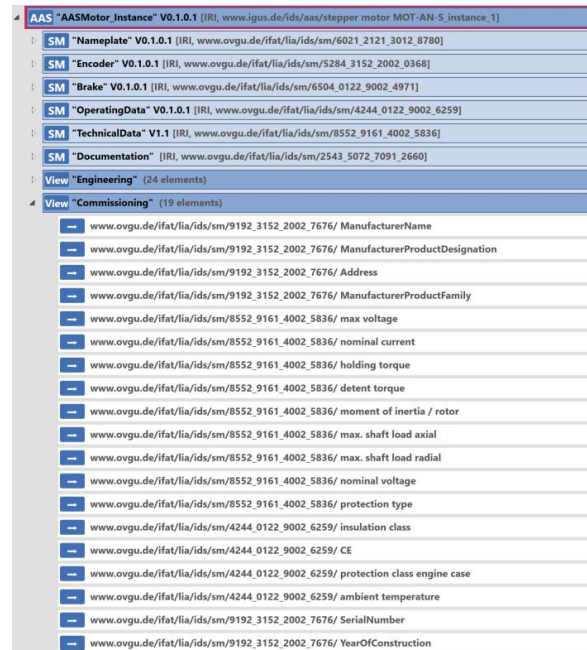


Abbildung 4-5: Ansicht der VWS einer Motorinstanz mit aufgeklappter *View* „Commissioning“

VWS der Transporteinheit: Teilmodelle für SP/SR

In den vorherigen Kapiteln wurde erklärt, wie die Eigenschaften, Parameter und Zustände von Assets mit Merkmalen als Teilmodelle modelliert werden können. Die nächste Herausforderung ist es zu beschreiben, wozu die Assets geeignet sind, um ihre Verwendung exakt definieren zu können. Damit ein Asset in einem CPS eingebunden werden kann, muss dieser Gegenstand relevante Informationen und die Beschreibung seiner Fähigkeiten in seiner VWS bereitstellen. In I4.0-Systemen können manche I4.0-Komponenten vertragliche Verhandlungen zur Einbringung von Produktionsleistungen unterstützen. In einem Dienstsysteem, das in Industrie 4.0 zur Anwendung kommt, können manche I4.0-Komponenten als Dienstleistungserbringer oder –nutzer aufgefasst werden. Die Informationen über die verfügbaren Dienstleistungen und bereitgestellten Funktionen müssen in einer Weise zur Verfügung gestellt werden, dass alle Interaktionspartner diese auch gleich verstehen. Das bedeutet, dass die Semantik der Dienst- und der Funktionsbeschreibung eindeutig sein muss.

In der heutigen Diskussion der I4.0-Community spielen die Begriffe „Funktion“, „Funktionalität“, „Fähigkeit“, „Skill“, „Operation“, „Capability“, „Prozess“, „Service“, „Dienst“ und „Dienstleistung“ zur Beschreibung einer Wirkung eines Assets eine zentrale Rolle. Jedoch ist ein Konzept zu einer sauberen Abgrenzung dieser Begriffe den Autoren nicht bekannt.

In diesem Dokument werden Begriffe „Funktion“, „Funktionalität“, „Fähigkeit“ als Synonym gesehen und mit dem VWS-Metamodell Element „Capability“ modelliert. Die Begriffe „Service“, „Dienstleistung“ und „Dienst“ sind im Kontext dieses Beitrages auch als Synonyme gesehen. Soll zum Beispiel eine Dienstleistung mit Merkmalen modelliert werden, so kann dafür ein Capability-Teilmodellelement verwendet werden. Die Merkmale können sowohl in dem Teilmodell

liegen in dem das Capability-Teilmodellelement eingetragen ist oder die Merkmale sind in einem anderen Teilmodell, das durch Referenz vom Teilmodell mit dem Capability-Teilmodellelement verbunden ist.

Ein Capability-Teilmodellelement beschreibt die Fähigkeit eines Assets oder einer Komponente, eine bestimmte Funktion zu erfüllen. Die Merkmalsätze für die Beschreibung von unterschiedlichen Capability-Teilmodellelementen sind funktionspezifisch und können für jede technische Funktion in Produktbeschreibungsstandards (z. B. ECLASS) zur Verfügung gestellt werden.

Wie auf der Abbildung 4-6 dargestellt, stellt das in diesem Kapitel betrachtete Asset eine Transporteinheit dar. Die Aufgabe bzw. erwartete Wirkung dieses Assets ist es, ein Werkstück innerhalb der P&P-Station von A nach B zu transportieren.

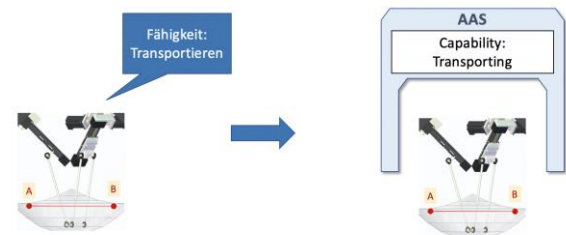


Abbildung 4-6: Agiert die Transporteinheit im I4.0-System als ein autonomer Dienstleister, werden ihre Fähigkeiten und angebotene Dienstleistungen in der VWS beschrieben

Dementsprechend ist die Fähigkeit der Transporteinheit Dinge von A (Startposition) nach B (Zielposition) mit dem Capability-Konzept [PLA20] zu modellieren.

Die Abbildung 4-7 zeigt einen beispielhaften Modellierungsansatz. In diesem Beispiel beinhaltet das Teilmodell „Capability“ alle Capabilities über die ein Asset verfügt. In diesem Fall handelt es sich um eine Capability, nämlich „Transportieren“. SemanticId der Capability „Transportieren“ verweist auf die entsprechende semantische Beschreibung im ECLASS-Standard. Das Capability-Modellierungskonzept, beschrieben im relevanten

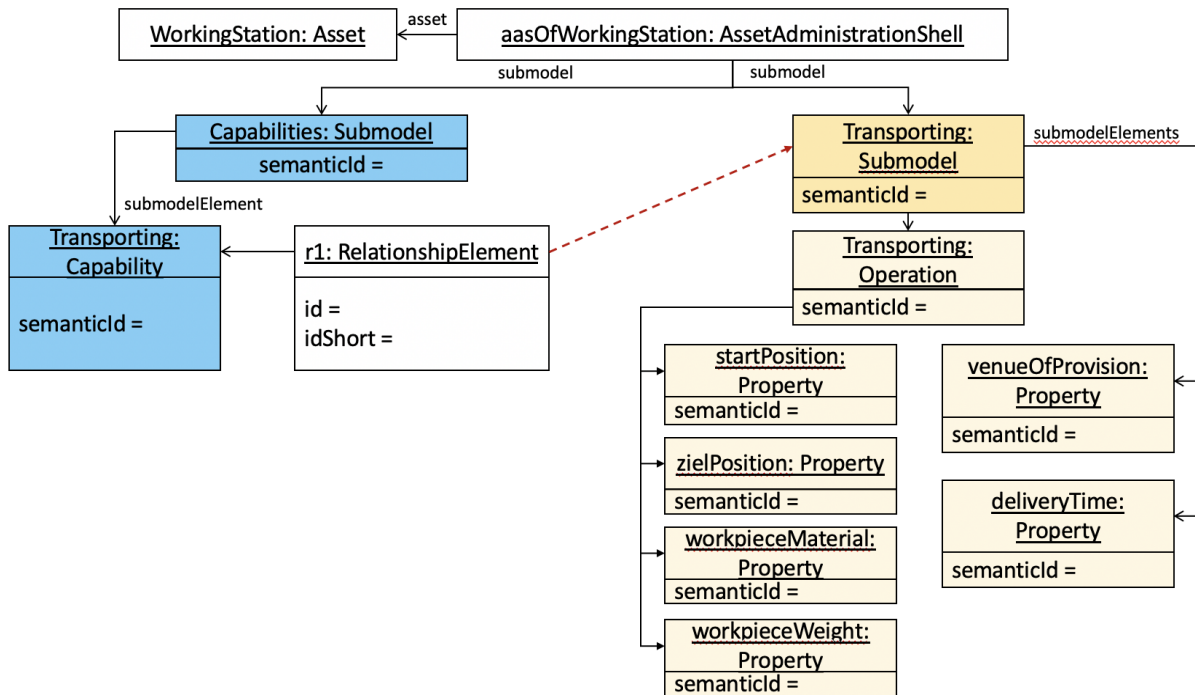


Abbildung 4-7: Beschreibung der Capability “Transportieren” angelehnt an das Konzept der Capability-Beschreibung im Capability-Whitepaper [DCI20]

Whitepaper der Plattform I4.0 [DCI20], sieht vor, dass es ein zweites Teilmodell gibt, das die Merkmale für die Beschreibung der Asset-Fähigkeit beinhaltet. Dabei sieht das Modellierungskonzept eine Strukturierung einiger Merkmale zu den einzelnen Operationen vor, um die Ausführung der Capability zu spezifizieren. In diesem Beispiel gehören dazu Start- und Zielposition, Material und Gewicht des transportierenden Werkstückes. Soll eine Dienstleistung modelliert werden, so können dem Teilmodell zusätzlich zu den Funktions-beschreibenden (technischen) Merkmalen die Dienstleistung-beschreibenden (kaufmännischen) Merkmalen hinzugefügt werden (z. B. Preis, Ort und Zeit der Erbringung einer Dienstleistung).

Die in dem Capability-Whitepaper [DCI20] dargestellten Konzepte der VWS-Capability und VWS-Operation lassen sich auch anders interpretieren, so dass eine alternative vereinfachte Variante der Capability-Beschreibung vorstellbar wäre. Die in der Abbildung 4-8 dargestellte Capability-Beschreibung

kommt ohne Verwendung der Operationen aus. Die in dem vorigen Beispiel der Operation zugeordneten Properties werden in dem Alternativvorschlag dem Teilmodell “Transporting” direkt zugeordnet.

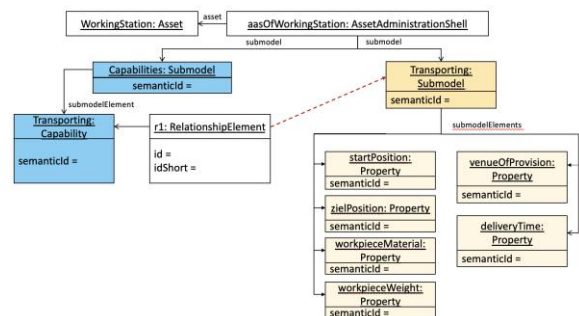


Abbildung 4-8: Alternative Möglichkeit, Capability “Transportieren” zu beschreiben

Workstation-Typ aus Sicht der Planung

Das VWS-Konzept unterstützt die Aggregation von I4.0-Komponenten zu einer neuen,

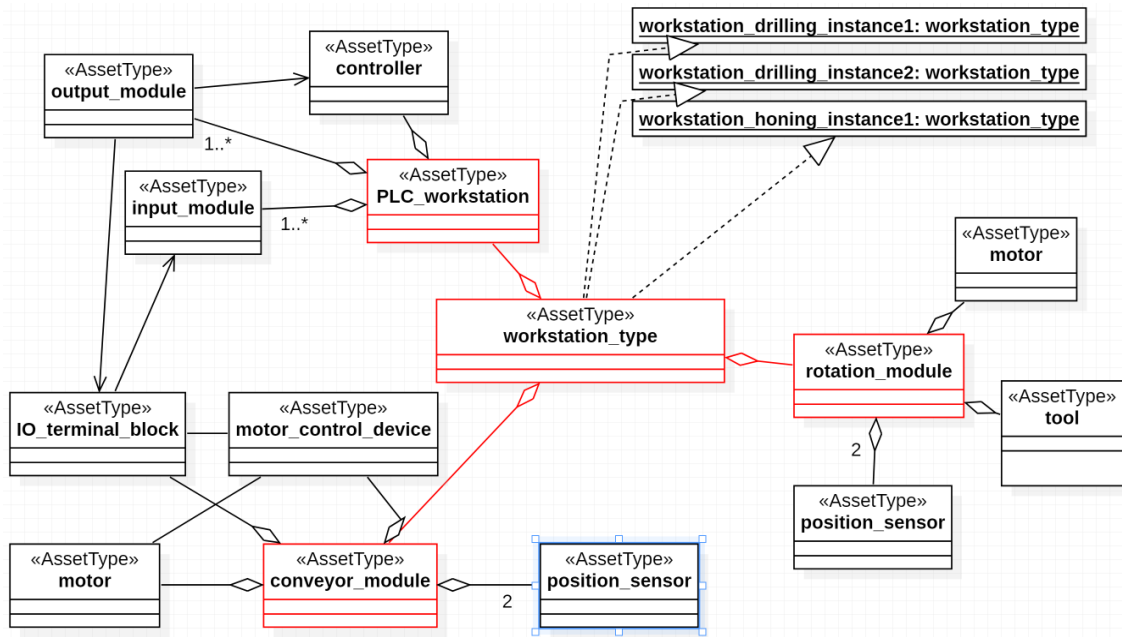


Abbildung 4-9 - Klassendiagramm Workstation

übergeordneten I4.0-Komponente [PLA17]. Daraus ergibt sich, dass unterschiedliche Möglichkeiten existieren, die Asset- und VWS-Struktur der Bearbeitungsstation (Workstation) zu modellieren. In diesem Abschnitt werden die Komponenten, aus denen sich die Workstation zusammensetzt, als jeweils eigene VWS modelliert (Vgl. Abbildung 4-9). Die Workstation ist dann ein Asset, welches sich aus verschiedenen Sub-Assets, wie Steuerung, Förderband und Bohrmodul, zusammensetzt.

Dies ist in der Praxis durchaus vorstellbar, weil die Workstation aus einer Menge verschiedener Zuliefererkomponenten konstruiert wird. In einem möglichen Zukunftsszenario können die VWS der entsprechenden Komponenten von den Zulieferern mitgeliefert und in die VWS der Workstation von dessen Hersteller (bzw. Systemintegrator), in dem Fall die OVGU, integriert werden.

In der P&P-Station sind drei Instanzen des AssetType `workstation_type`³ verbaut. Gemäß RAMI 4.0 ist es sinnvoll, den Produktlebenszyklus von Assets nach Typ- und Instanzinformationen zu separieren (siehe Abschnitt 5.6). Dieser Abschnitt zeigt die Modellierung einer VWS der Workstation als AssetType. Ein Vorschlag, wie die VWS einer AssetInstance aus der VWS eines AssetType abgeleitet werden kann, wird in Abschnitt 4.6 beschrieben.

Neben der SPS (PLC_workstation) besteht `workstation_type` weiterhin aus einem Förderband (`conveyor_module`) und einem Rotationsmodul (`rotation_module`). Diese Assets bestehen wiederum aus weiteren Sub-Assets, wie Aktoren und Sensoren. Aufgrund des Aufwands werden nur die Sub-Assets der ersten Ebene modelliert (Vgl. rot markierte Klassen und Aggregationsbeziehungen in Abb. 4-9).

³ zur Schreibweise in diesem Kapitel: Die Stereotypen der Klassen sind die Modellelemente der VWS und haben die gleiche Schreibweise (CamelCase) wie in der VWS-Spezifikation. Die Klassen- und Objektnamen

nutzen “_” für die Trennung der einzelnen Benennungselemente.

Die Zuordnung der Beziehungen erfolgt für diese I4.0-Komponente in einem eigenen Teilmodell. Dieses Teilmodell enthält Relationships zwischen den Assets, die in dem umgesetzten Beispiel (rot in Abbildung 4-10 gekennzeichnet) `workstation_type` zusammengehören. Die Beziehungen werden auf Asset-Ebene eingetragen (siehe Abbildung 4-10) und nicht auf der Ebene der VWS. Der Grund hierfür ist, dass ein Asset prinzipiell mehrere VWS besitzen kann und ein Austausch eingepflegt werden müsste. Vermutlich ist ein Austausch der Assets seltener. Eine speziell für diese Aufgabe vorgesehene andere Variante ist die Modellierung als Verbundkomponente (siehe Kapitel 5.7) in dem das Stücklisten-Teilmodell (Bill of Material - BoM, Kapitel 5.5) Verwendung findet.

Jede VWS besteht aus den Teilmodellen *Nameplate*, *Document* (Sammlung relevanter Unterlagen), *Service* (Informationen bzgl. Support/Maintenance), *Identification* (identifizierende Merkmale) und *TechnicalData* (technische Merkmale). Die Auswahl der Merkmale und die Zuordnung von Werten ergibt sich aus den verfügbaren

Informationen der Hersteller. Diese wurde von den Autoren dieses Dokuments festgelegt oder von den Typenschildern der Assets abgelesen (siehe Tabelle 4-1). Ein Sonderfall wird bei dem AssetType der Steuerung `PLC_workstation_type` betrachtet. Die Typ-Informationen werden hierbei nicht von einem konkreten Hersteller übernommen. Der Systemintegrator der Workstation hat entschieden, je nach Bestand, SPSen unterschiedlicher Typen einzubauen. Diese können prinzipiell auch von unterschiedlichen Herstellern stammen. Allerdings versichert der Systemintegrator in dem Teilmodell *TechnicalData*, dass die genannten Merkmale eingehalten werden.

The screenshot displays a software interface for defining relationships between sub-assets. On the left, a tree view lists various sub-assets, including 'SM Identification', 'SM TechnicalData', and 'AAS PLC_workstation_instance1 V1.0'. The 'Relationships' sub-asset is highlighted in red. On the right, a detailed view of the 'Relationships' element is shown, containing the following information:

- Element Content:** Submodel Element (RelationshipElement)
- Referable:** idShort: Rel_workstation_type_PLC_workstation_type
- Kind:** kind: Instance
- Semantic ID:** semanticId: (ConceptDescription) (local) [IRD] 0173-1#02-ZZZ998#002
- Qualifiable:**
- HasDataSpecification (Reference):** ConceptDescription
- Referable:** idShort: isPartOf
- Identifiable:** idType: IRDI; id: 0173-1#02-ZZZ998#002
- isCaseOf:**
- HasDataSpecification:** HasDataSpecification (Reference); reference[0]: (GlobalReference) (no-local) [IRI] http://admin-shell.io/DataSpecificationTemplates/DataSpecificationIEC61360/2/0
- Data Specification Content IEC61360:** preferredName: [en] Is part of; [de] Teil von; shortName: [en] isPartOf; unit: ; dataType:
- RelationshipElement:** first: (Asset) (local) [IRI] www.ovgu.de/ifat/ia/ids/asset/workstation_type; second: (Asset) (local) [IRI] www.ovgu.de/ifat/ia/ids/asset/PLC_workstation_type

Abbildung 4-10 - Definition der Beziehungen zwischen den Sub-Assets der Workstation VWS im Teilmodell Relationships

Der gemeinsame Einsatz von *Nameplate* und *Identification* erscheint redundant, da zum Teil dieselben SubmodelElements enthalten sind. Dies ist auf den Einsatz in verschiedenen Use-Cases zurückzuführen. Eine Fehlerquelle, die zu Inkonsistenz führen kann, besteht darin, gleichnamigen SubmodelElements unterschiedliche Werte zuzuordnen.

Insbesondere das Teilmodell TechnicalData erscheint interessant für einen Planer, der nach Komponenten sucht, welche die Anforderungen einer geplanten Anlage befriedigen. In diesem Sinne stellen die Merkmale dieses Teilmodells hier Zusicherungen dar, dass der Typ des entsprechenden Assets diese Merkmale erfüllen wird.

Asset	AssetKind	Quelle
Workstation_type	Type	Eigene Festlegung
Rotation_module_type	Type	[FES01]
Conveyor_module_type	Type	[FES02]
PLC_workstation_type	Type	Eigene Festlegung
Workstation_instance1	Instance	Eigene Festlegung
Drilling_module_instance1	Instance	[FES01] + Typenschildinformationen
Conveyor_module_instance1	Instance	[FES02] + Typenschildinformationen
PLC_workstation_instance1	Instance	[WAG01] + Typenschildinformationen

Tabelle 4-1 - Informationsquellen für Teilmodelle der Workstation

Workstation-Instanz aus Sicht der Planung

Im Folgenden soll gezeigt werden, wie die VWS der AssetInstance der Workstation modelliert wurde. Ein Asset vom Kind Instance ist ein konkretes Asset, welches nur einmalig existiert. Im P&P-Demonstrator sind drei Instanzen des Typs workstation_type vorhanden. Allerdings wird exemplarisch nur die Instanz drilling_workstation_Instance1 modelliert. Die "Instantiierung" erfolgte im AASX Package Explorer durch das rekursive Kopieren der bestehenden VWS workstation_type. Rekursiv bedeutet in diesem Kontext, dass alle Teilmodelle und ihre Inhalte aus der Quell-VWS übernommen werden. idShort und id der neuen VWS, ihrer Teilmodelle und der Assets müssen umbenannt werden. Die Ableitung von der VWS des

AssetType kann durch die DerivedFrom-Beziehung kenntlich gemacht werden (siehe Abb. 4-11).

Bei der Ableitung einer VWS aus der VWS eines AssetType können sich bezüglich der SubmodelElements verschiedene Konsequenzen ergeben:

- die Art des SubmodelElements kann sich ändern
 - Beispiel: die Länge des Förderbands vom workstation_type kann vom Kunden bei der Bestellung in bestimmten Grenzen konfiguriert werden und wird daher als Range modelliert. Das Förderband einer Instanz hat schließlich eine feste Länge und wird zum Property

- die Bedeutung des Merkmals kann sich ändern und erhält eine andere semantische ID
 - Beispiel: der Planer legt fest, dass der SPS-Typ mind. 8 digitale Eingänge erfordert (MinNumberOfInputs: 0173-1#02-AAP328#001). Die Instanz der SPS hat 8 Eingänge (NumberOfInputs: 0173-1#02-AAP508#003)

Es werden in einer VWS immer nur Merkmale modelliert, welche für die Beschreibung des jeweiligen AssetKinds relevant sind. Merkmale, die auf Instanz-Ebene existieren, aber für die Beschreibung des AssetType nicht relevant sind, müssen in einem geeigneten Submodell in der VWS der AssetInstance hinzugefügt werden. Ein Beispiel ist die Seriennummer eines Assets. Dieses Merkmal wird nur in der VWS der AssetInstance modelliert.

Workstation-Instanz: Teilmoleküle für SP/SR

Das Konzept der Modellierung der Fähigkeiten und Dienstleistungen eines Assets mit Capability-Teilmodellelementen wird im Kapitel 4.6 eingeführt. Um Wiederholungen zu vermeiden, wird für allgemeine Erläuterungen an der Stelle auf das entsprechende Kapitel verwiesen.

Dieses Kapitel bezieht sich auf eine beispielhafte Modellierung der Fähigkeit "Bohren" der Workstation. Eine abstrakte Beschreibung der Capability der Workstation, in diesem Fall "Bohren", angelehnt an das Beispiel in dem Capability-Whitepaper [DCI20], ist in der Abbildung 4-12 und Abbildung 4-13 dargestellt.

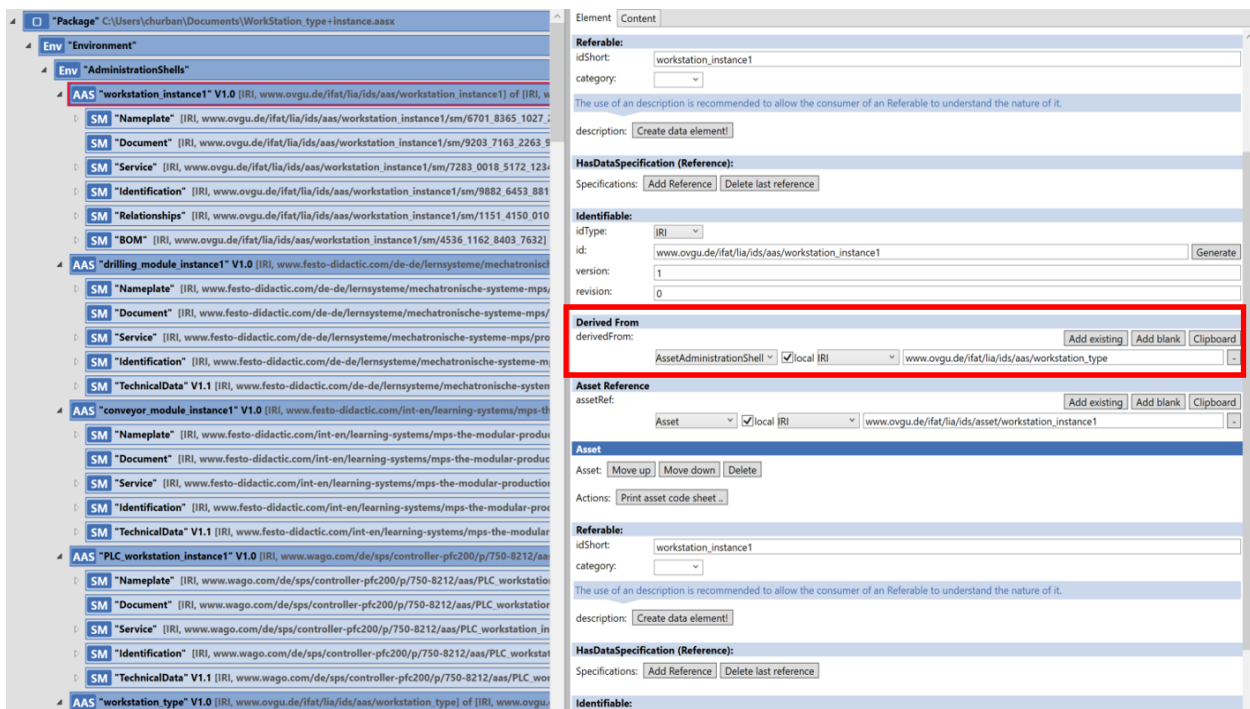


Abbildung 4-11: Derived From - Ableitung einer VWS aus Asset-Kind:Type

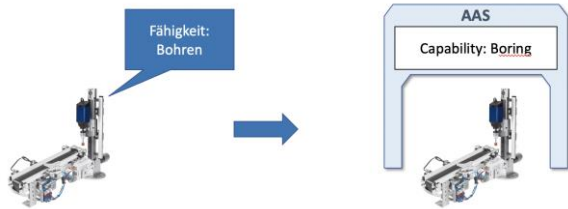


Abbildung 4-12: Agiert die Workstation im I4.0-System als ein autonomer Dienstleister, werden ihre Fähigkeiten und angebotene Dienstleistungen in der VWS beschrieben

Die VWS-Capability- und VWS-Operation-Konzepte lassen sich auch anders interpretieren, so dass eine alternative Variante der Capability-Beschreibung vorstellbar wäre. Die auf der Abbildung 4-15 dargestellte Capability-Beschreibung kommt ohne Verwendung der Operationen aus. Die in dem vorigen Beispiel der Operation zugeordneten Properties werden in dem Alternativvorschlag dem Teilmodell "Boring" direkt zugeordnet.

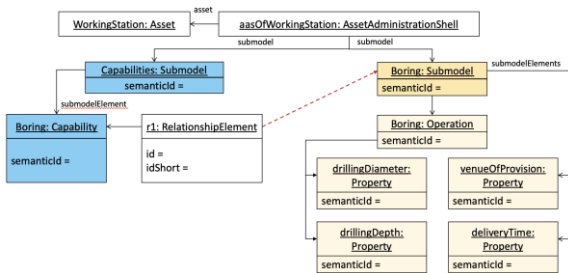


Abbildung 4-13: Beschreibung der Capability "Bohren" angelehnt an das Capability-Whitepaper [DCI20]

Die Abbildung 4-16 zeigt wie die Beschreibung der Capability "Bohren" ohne Verwendung der Operationen im AASX Package Explorer umgesetzt wird.

Die Abbildung 4-14 zeigt, wie die Beschreibung der Capability "Bohren" im AASX Package Explorer umgesetzt wird.

based on specifications of Plattform Industrie 4.0

Element	Content
SubmodelReference	submodelRef: (Submodel) (local) [IRI] www.company.com/ids/sm/5004_4190_2102_4338
Submodel	
Referable members:	idShort: BoringSubmodel
Identifiable members:	idType: IRI id: www.company.com/ids/sm/5004_4190_2102_4338
Semantic ID	semanticId: (GlobalReference) (no-local) [IRD] 0173-1#01-AKG243014
Kind:	

Abbildung 4-14: Umsetzung der Beschreibung der Capability "Bohren" im AASX Package Explorer

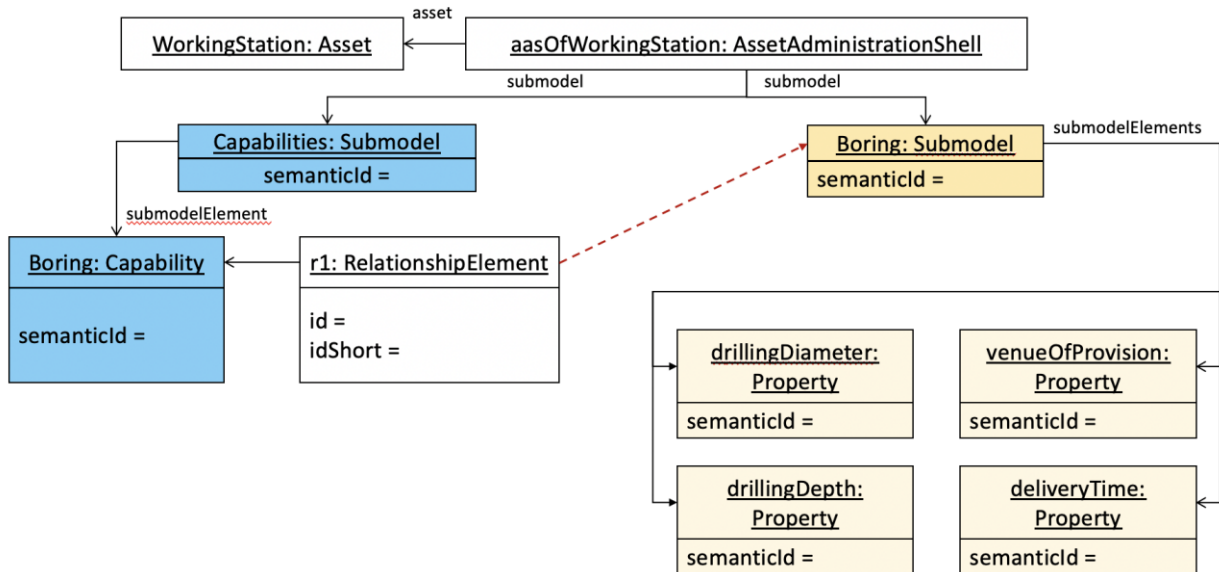


Abbildung 4-15: Alternativer Vorschlag zur Beschreibung der Capability "Bohren"

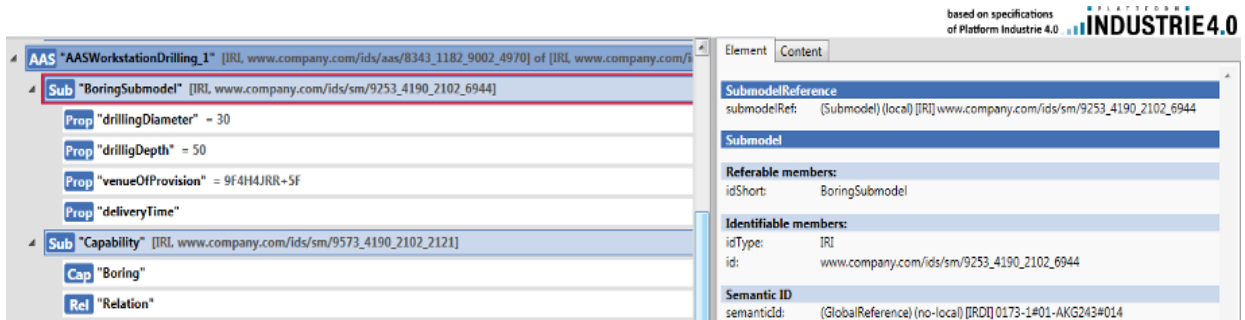


Abbildung 4-16: Umsetzung der Capability-Beschreibung (Option 2) im AASX Package Explorer

(5) VWS Modellierungs-konzepte

Dieses Kapitel greift wesentliche Konzepte der VWS-Modellierung heraus und stellt diese beispielhaft vor. Ziel ist es, das Zusammenwirken verschiedener Modellelemente für eine konkrete Aufgabe zu beschreiben.

Die Beispiele im Kapitel 5 beziehen sich auf den Delta Roboter. Deshalb wird durchgängig dessen technische Dokumentation [I-GUS01] als Asset-bezogene Informationsquelle verwendet.

Grundstruktur der VWS und ausgewählte wichtige Attribute

Dieses Kapitel beschreibt die grobe Struktur der VWS und stellt diese beispielhaft vor. Ziel ist es, die Einstiegspunkte für die Modellierung zu beschreiben.

Die Asset Administration Shell (VWS) besteht aus der AssetInformation sowie den Teilmodellen und den Views. Alle vier Informationseinheiten haben Attribute, die diese näher beschreiben. Eine Auswahl von den Attributen mit ihren Kardinalitäten ist in Abbildung 5-1 eingetragen (weiße Schrift). Security-Attribute und Funktionalitäten sind durchgängig in allen Elementen verankert.

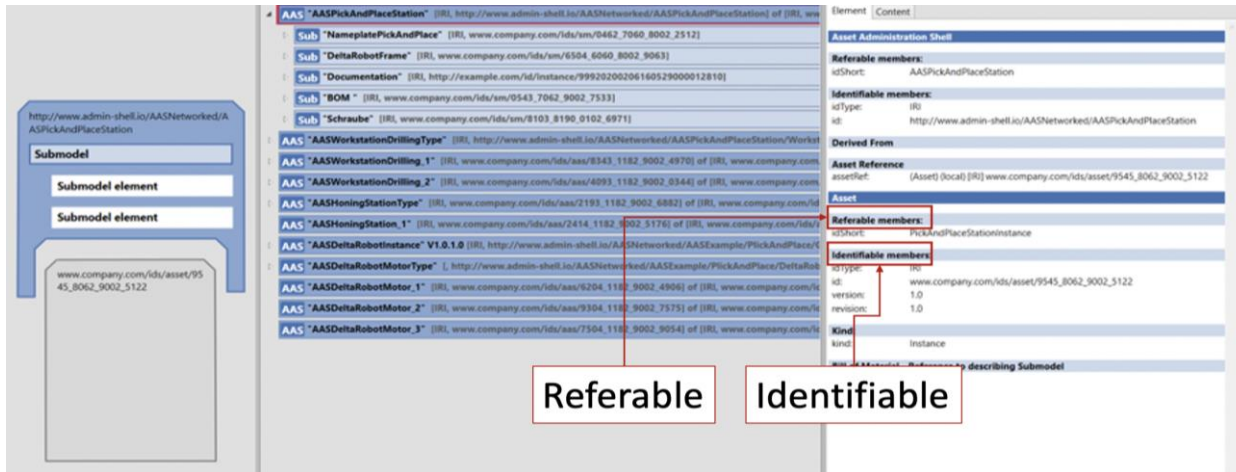


Abbildung 5-2 – Referable- und Identifiable-Angaben für VWS und Asset der P&P-Station

Erläuterungen

Jedes Element des VWS-Modells benötigt einen Identifizierer, um eindeutig maschinenlesbar benannt zu werden. Diese Benennung wird für die verschiedenen Use Cases, in denen die Modellelemente benötigt werden, verwendet.

Es wird zwischen global eindeutiger Identifizierung, bezeichnet mit „Identifiable“ und lokal eindeutiger Identifizierung, bezeichnet mit „Referable“ unterschieden. Diese Unterscheidung macht es möglich, dass Benennungen in mehreren VWS identisch verwendet werden können (nur für idShort möglich). Dies trägt wesentlich zur Wiederverwendung von Modellteilen bei.

Es sind fast alle Modellelemente Referable. VWS Identifiable Modellelemente sind Referable und haben zusätzlich die Attribute:

- AdministrationInformation (version und revision) sowie den Identifier bestehend aus idType (IRI, IRDI und Custom) und id Identifiable sind:
- VWS, Asset, Teilmodell, ConceptDescription

Semantische Zuordnung zu Modellelementen - die „semanticId“

Ziel des Kapitels

Beschreibung der Aufgabe des Attributs „semanticId“.

Verwendete Eingangsinformationen für das Modell

Relevante Eigenschaften, Parameter, Zustände der modellierten Assets sollen in die Merkmale nach IEC 61360 überführt werden. Die eindeutigen Identifikatoren für Merkmale anderer Modellelemente sind den standardisierten Vokabularen zu entnehmen.

- Ein von der Plattform Industrie 4.0 favorisiertes Vokabular der semantischen Definitionen ist der ECLASS-Standard.
- Andere Vokabulare: Firmenspezifische Geräteklassifikationen und deren Merkmalfestlegungen
 - Eine nutzerspezifische semanticId kann in Form eines IRI definiert werden
 - Der PackageExplorer hat die Möglichkeit, die IRI automatisch zu erzeugen

Verwendete Modellelemente des VWS-Metamodells

- Kapitel 4.7.2.7 “Semantic Reference Attribute”

Darstellung im Package Explorer

Am Beispiel der Property “ManufacturerTypNameVWS” des Nameplate-Teilmodells ist in Abbildung 5-3 eine semanticId dargestellt. Daraus lässt sich der Verweis auf die entsprechende ConceptDescription, in diesem Fall eine ECLASS IRDI, erkennen.

Erläuterungen

Zusätzlich zur Identifizierung der Modellelemente ist vorgesehen, dass jedes Modellelement eine ergänzende semantische Beschreibung besitzt. Deshalb haben die Modellelemente das Attribut “semanticId”. Das ist eine Referenz auf die maschinenlesbare

Merkmale (Property) entsprechend der Norm IEC 61360 beschrieben sind. Diese Norm legt die Attribute fest, die für die Beschreibung zu verwenden sind.

Nahezu alle Modellelemente (außer Asset, ConceptDescription und Asset Administration Shell) haben eine semantische Referenz, d. h. die semanticId. Die Modellelemente der VWS haben auch eine eigene semanticId, die im Namenraum [www.admin-shell.io/...](http://www.admin-shell.io/) angelegt ist.

Die semanticId ist kein Identifier, der für die Benennung und damit für das Ansprechen in einem Use Case verwendet wird. Dazu sind die Identifizierer von Identifiable und Referable zuständig. Die semanticId ist ein zusätzliches Attribut, das für die inhaltlich fachliche Erläuterung des Modellelements vorhanden ist. Sie weist auf die ergänzende Beschreibung hin und trägt damit zur Zuordnung der Bedeutung des Modellelementes bei. Es wird damit geklärt, was sich hinter dem Modellelement verbirgt. Mit anderen

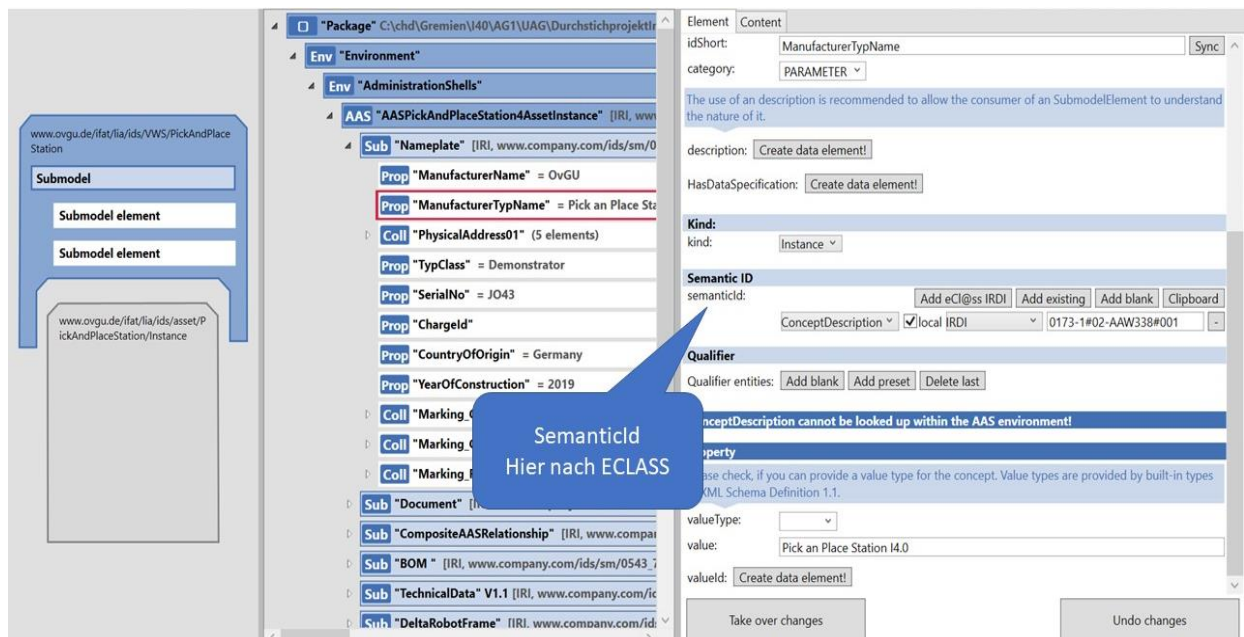


Abbildung 5-3 – SemanticId am Beispiel einer Property

Beschreibung des Modellelements, d. h. sie ist eine Referenz auf einen Eintrag in einem Dictionary. Der bevorzugte Fall eines verwendeten Dictionaries ist eines, in dem die

Worten, die Attribute eines Modellelements sind nicht lokal hinterlegt, sondern global festgelegt. Dadurch können in unterschiedlichen Engineeringphasen gleiche Merkmale

mit den gleichen Attributen verwendet werden. Die Abbildung von Merkmalwerten beim Übergang von einer Technologie in eine andere oder von einem Werkzeug zu einem anderen entfällt.

Das Metamodell der VWS definiert Einträge in einem Dictionary auf der Basis von IEC 61360. Diese Einträge werden mit Concept-Description (siehe 5.8) bezeichnet. Es wird stets nur auf einzelne Einträge in einem Dictionary von einem Modellelement der VWS verwiesen. Deshalb gibt es auch kein Dictionary selbst in der VWS-Spezifikation, sondern nur die Einträge, die ConceptDescriptions.

Beziehung zwischen VWS, Asset und Asset Identification Teilmodell

Ziel des Kapitels

- Beschreibung der Beziehung zwischen Asset und VWS in einem Teilmodell
- Modellierung des Typenschildes/Nameplate des Assets und dessen Referenzierung in der VWS

Verwendete Modellelemente des VWS-Metamodells

- Table P. 58 – attribute „asset“ (this is a reference)
- Table P. 59 – attribute „assetIdentificationModel“

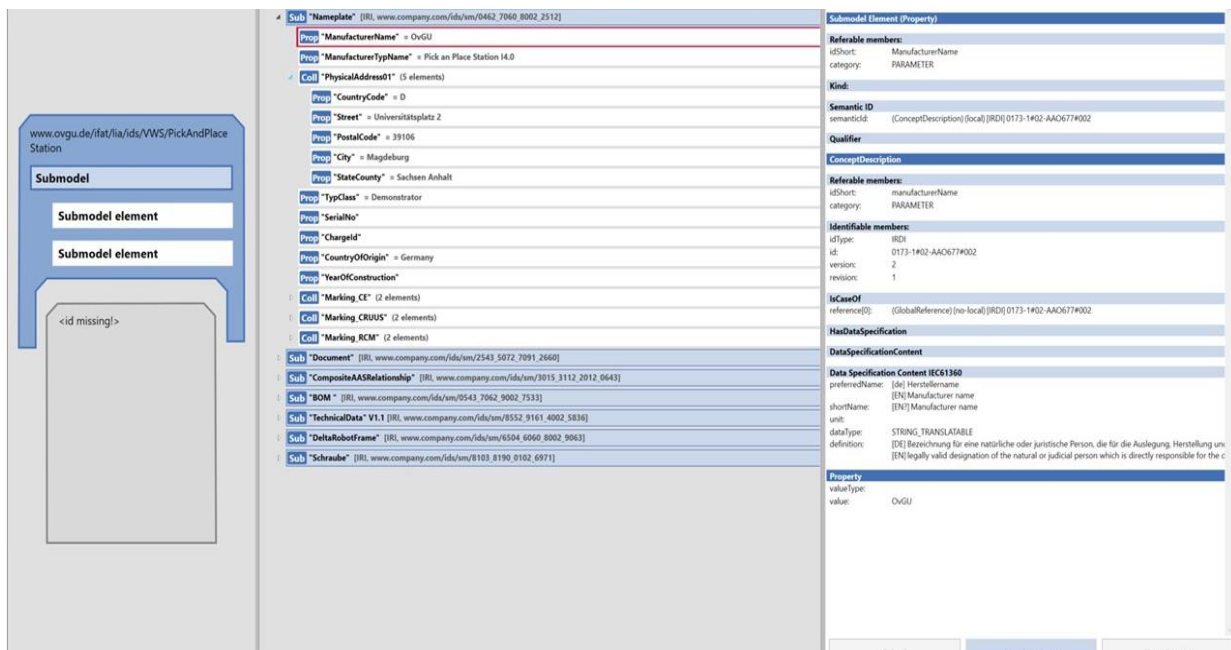


Abbildung 5-4 - Referenz zwischen VWS and Asset

Erläuterungen und Darstellung im Package Explorer

Für die Darstellung der Beziehungen zwischen VWS und Asset ist ab Version 3.0 der VWS Spezifikation das Objekt "AssetInformation" vorhanden. In diesem sind unter anderem die Asset-Identifikation und der Asset-Kind anzugeben. Die Asset-Identifikation wird als globalAssetId angegeben, die eine weltweit eindeutige Identifikation ermöglichen soll. Eine gute Möglichkeit ist es, diese entsprechend DIN SPEC 91406 zu vergeben. Firmen haben in der Regel zunächst eigene interne Bezeichner, z.B. Seriennummer und Bestellcode. Diese ergänzende ID sind in der externalAssetId einzutragen, von der es mehrere geben kann. Außerdem wird in der VWS ein Typenschild-ähnliches Teilmodell angelegt, das durch Referenz aus dem AssetInformation-Objekt eindeutig zu finden ist. Dieses Teilmodell sollte aus dem Teilmodell-Template "Digital Nameplate" [DNI20] abgeleitet werden. Hier sind für diese Aufgabe typischen Informationen hinterlegt (Abbildung 5-4).

Das Asset ist nicht Bestandteil der VWS, die I4.0-Komponente besteht aus Asset und VWS. Dennoch wird das Asset im Package Explorer modelliert. Dafür ist das Environment (Env) zuständig.

Da es die AssetInformation erst ab der V3.0 der VWS-Spezifikation gibt, kann sie im Beispiel mit dem Package Explorer noch nicht verwendet werden. Die entsprechenden Festlegungen zeigt deshalb Abbildung 5-5 mit Seitenangabe der VWS-Spezifikation, in denen die Definitionen enthalten sind.

Sind Assets aus mehreren Komponenten zusammengesetzt, so sind deren Beziehungen in der Stückliste (Bill of Material - BoM) zu beschreiben (siehe 5.5).

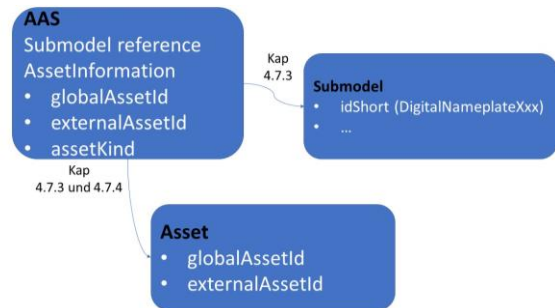


Abbildung 5-5 - Beziehung zwischen VWS, Asset and Asset Identification Teilmodell

Bill of material

Ziel des Kapitels

Darstellung der Struktur eines Assets als Teilmodell „Bill of Material“ am Beispiel von DeltaRobot 360.

Verwendete Modellelemente des VWS-Metamodells

- Kapitel 4.3: Composite I4.0 Components
- Kapitel 4.7.5, S. 58 und S. 59: Definition

Erläuterungen und Darstellung im Package Explorer

Die Stückliste (BillOfMaterial – BOM) ist ein Teilmodell. Dieses ist in der VWS des Assets enthalten, zu der die Komponenten in der Stückliste gehören. In der AssetInformation befindet sich das Attribut, in dem die Referenz auf das BoM-Teilmodell enthalten ist.

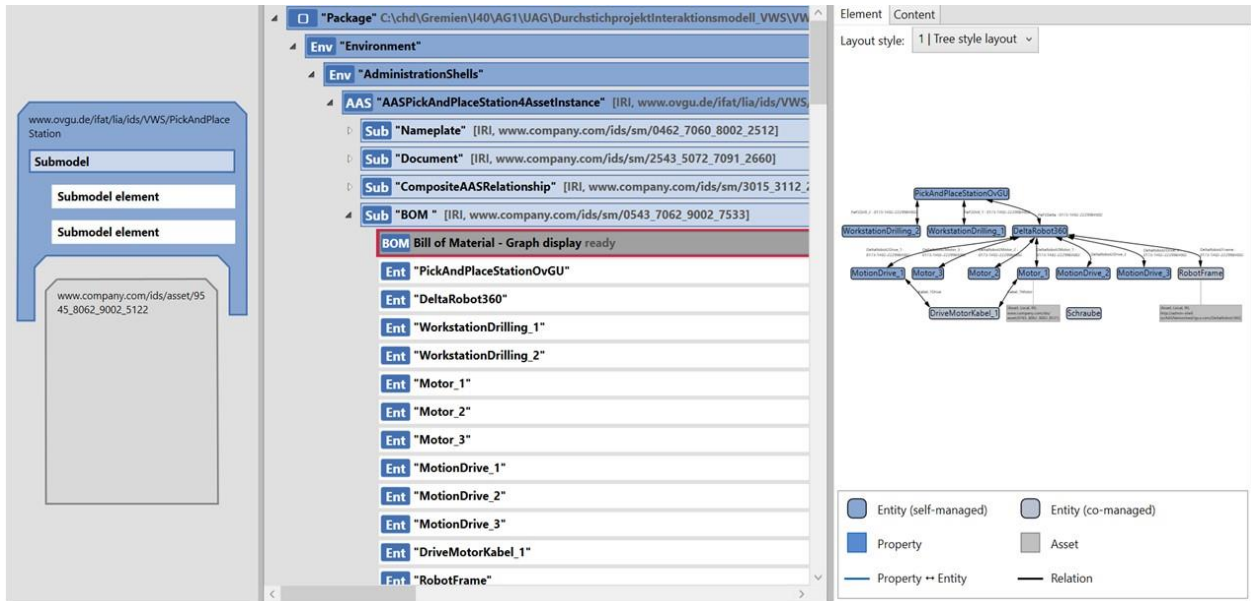


Abbildung 5-6: Übersicht BillOfMaterial P&P-Station Instance

Sind Komponenten enthalten, die aus weiteren Komponenten zusammengesetzt sind, so ist diese Verfeinerung der entsprechenden VWS dem aus weiteren Komponenten bestehenden Asset zugeordnet.

Die Komponenten des Assets in der BoM sind als „Entity“ in diesem Teilmodell zu modellieren (Abbildung 5-6). Beziehungen zwischen den Komponenten (z. B. ist Teil von - isPartOf) werden als „RelationshipElement“ modelliert. Das Teilmodell, welches die BOM

enthält, muss mit der semanticId „http://example.com/id/type/submodel/BOM/1/1“ versehen werden, um dieses Teilmodell als BOM identifizieren zu können.

Die Entitäten entsprechen in einer 1:1 Beziehung den Assets, die zu modellieren sind. Für die Entities sind idShort, entityType und die Referenz (Identifiable) auf das Asset anzugeben (Abbildung 5-7). Das Attribut „entityType“ kann „CoManagedEntity“ oder

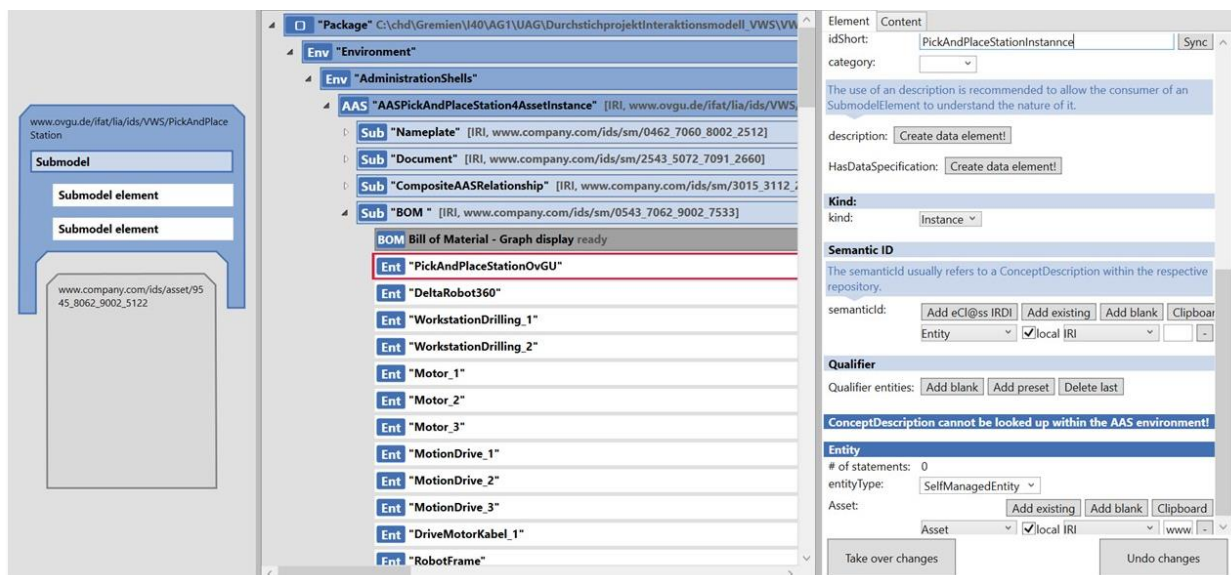


Abbildung 5-7: Bill of Material-Modellierung

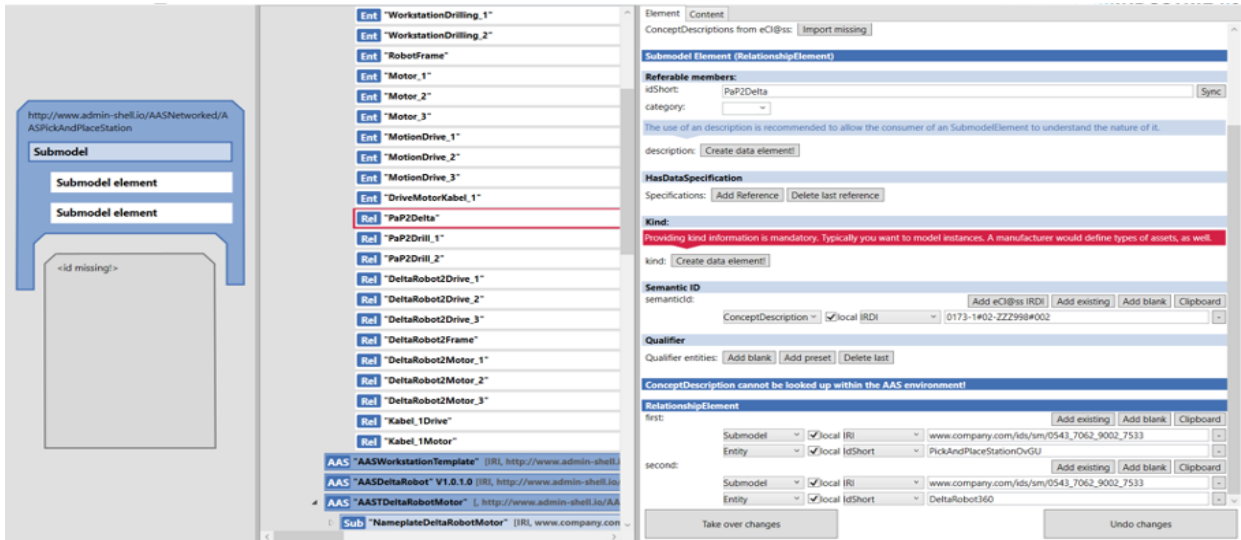


Abbildung 5-8: RelationshipElement Beispiel

„SelfManagedEntity“ sein. CoManagedEntity sagt aus, dass das Asset dieser Entity keine VWS besitzt und in der VWS verwaltet wird, in der die BOM ist. SelfManagedEntity sagt aus, dass das zugehörige Asset eine eigene VWS besitzt.

Mit dem RelationshipElement werden die Beziehungen zwischen den Entities der BOM modelliert. Die Beziehung ist gerichtet und hat einen Ausgangspunkt (first) und einen Endpunkt (second).

Die Identifikation erfolgt jeweils mittels idShort und dem Identifier (IRI) der dazugehörigen Entity (Abbildung 5-8). In der semanticId des RelationshipElements wird hinterlegt, welche Bedeutung die Relationship hat. Hier im Beispiel „isPartOf“ gekennzeichnet durch „0173-1#02-ZZZ998#002“. Dafür gibt es dann einen Eintrag in der ConceptDescription (Abbildung 5-9).

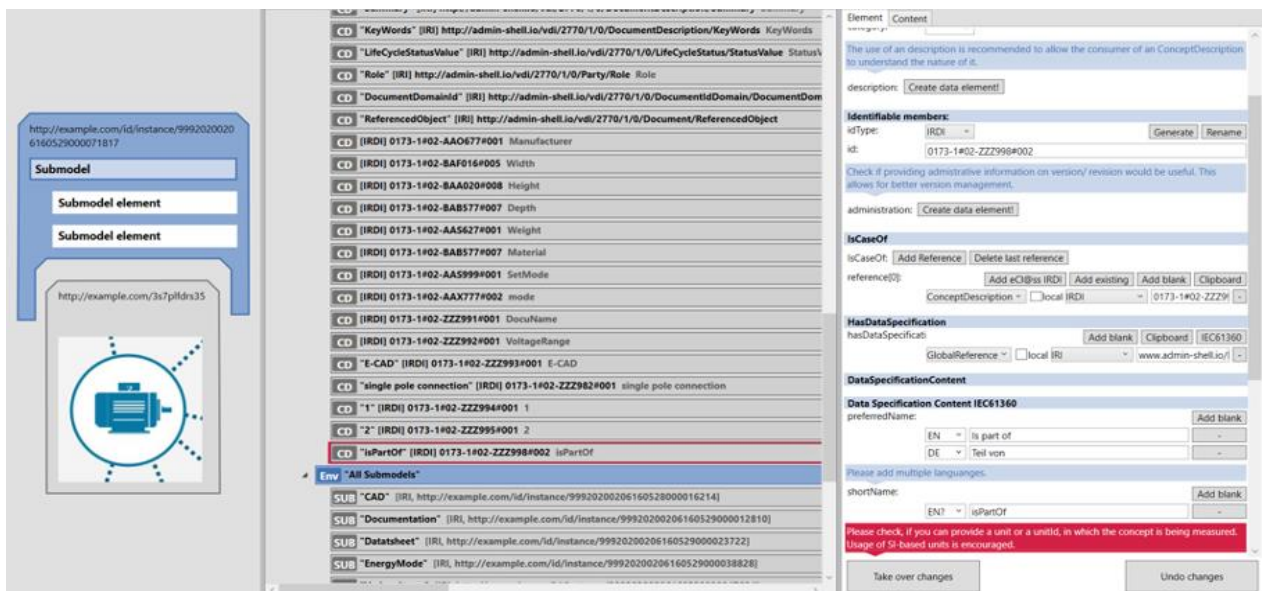


Abbildung 5-9: isPartOf - Property in der ConceptDescription

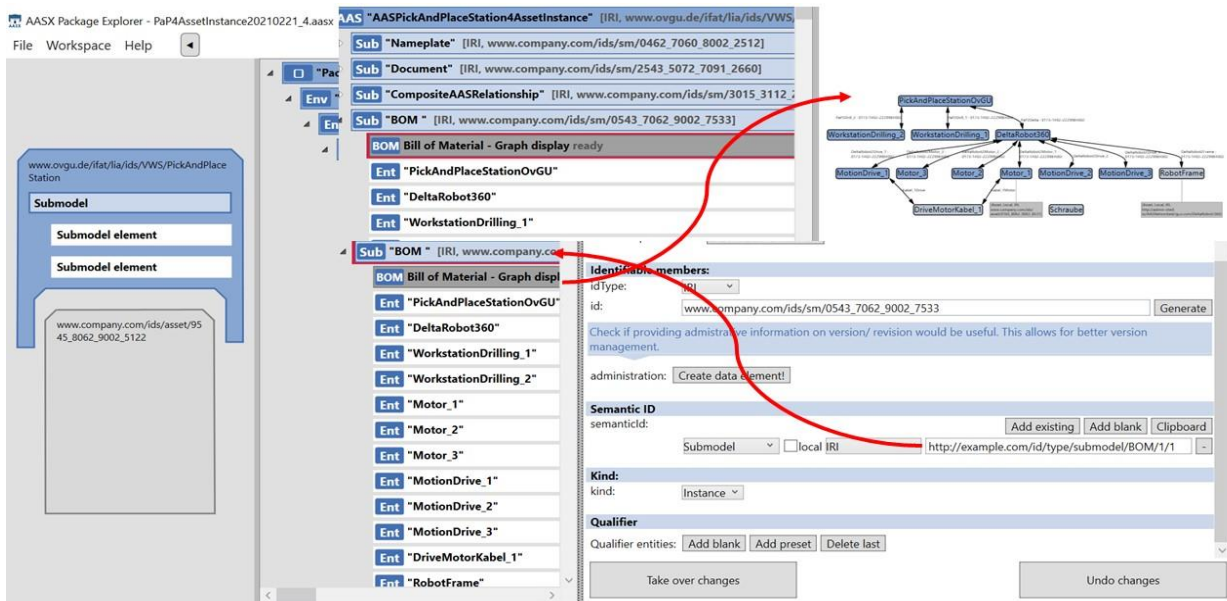


Abbildung 5-10: Beziehung des Assets zur BOM im Package Explorer - Beispiel

Ein besonderes Feature des Package Explorers ist es, dass bei semanticId = „http://example.com/id/type/submodel/BOM/1/1“ eine graphische Darstellung der BOM erscheint, wenn man das BOM-Element des Teilmodells aktiviert. Dieses erscheint bei richtiger Eingabe der semanticId.

Außerdem können bei einer Entity weitere SubmodelElements zur detaillierteren Beschreibung der Entity, d. h. des Assets, angegeben werden. Z. B. eine Referenz auf die VWS eines self-managed Assets oder Properties für ein co-managed Asset. Dies soll noch etwas ausführlicher erläutert werden.

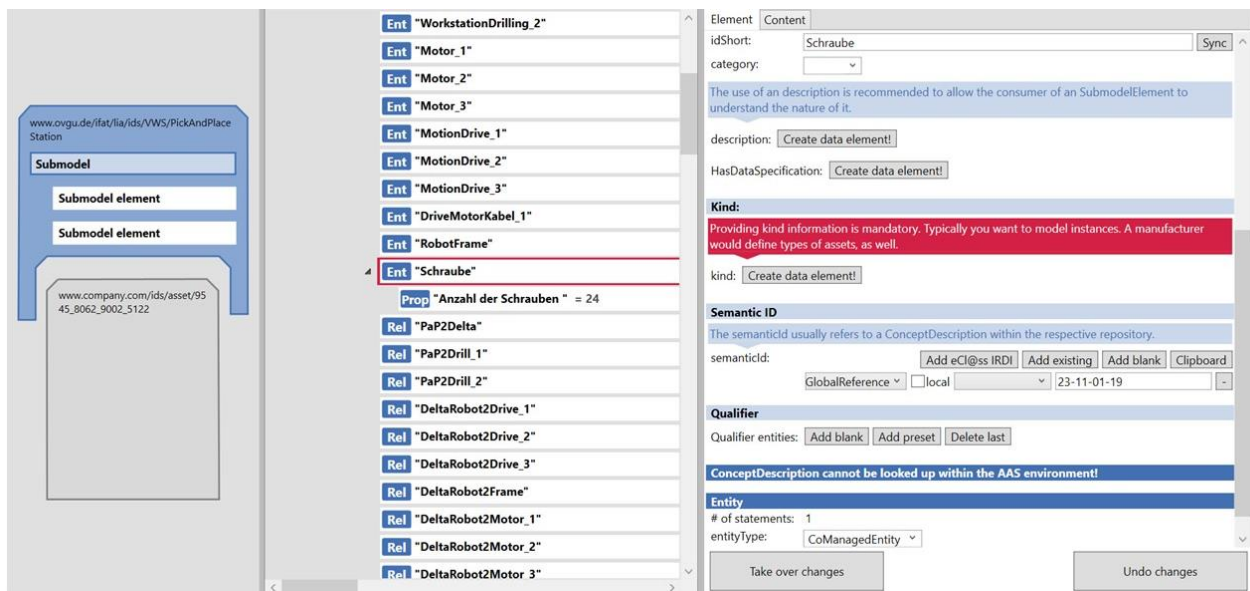


Abbildung 5-11: Entity Übersicht aus der BOM im Package Explorer

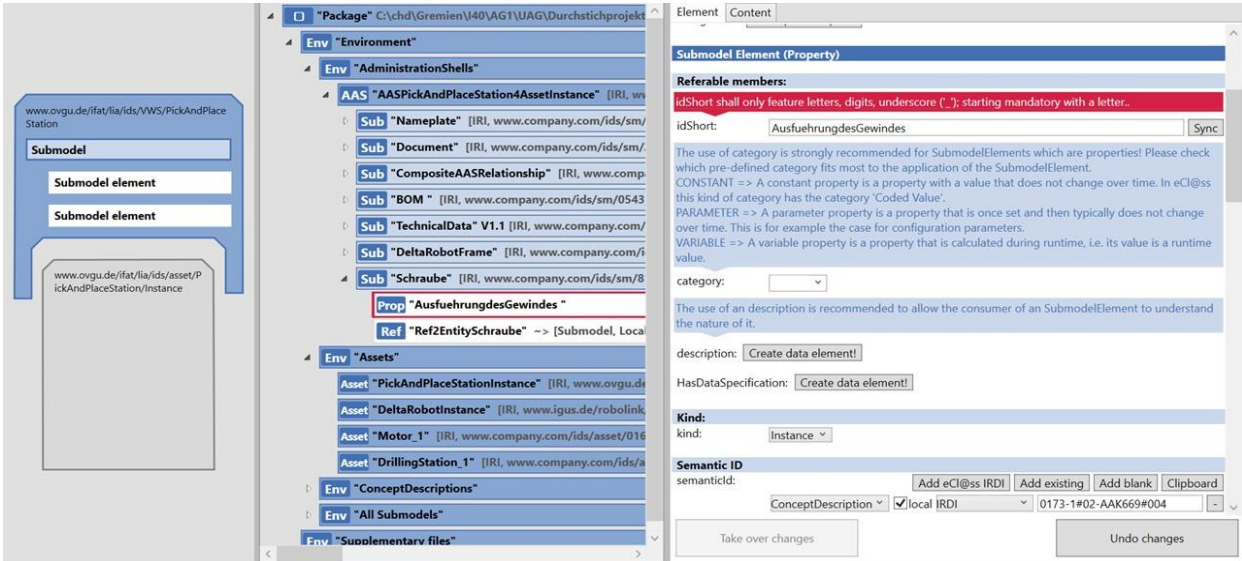


Abbildung 5-12: Verfeinertes Merkmal (Property) der CoManagedEntity Schrauben - hier Beispiel Anzahl der Schrauben

Es gibt Komponenten in Maschinen und Anlagen, die entweder keine eigene Kommunikationsfähigkeit besitzen (z. B. Schrauben, Gestänge, Dichtungen), in größeren Stückzahlen auftreten oder deren individueller Wert (im Sinne von Einkaufskosten) gering ist. Für diese werden aber trotzdem zur Begleitung des Lebenszyklus Informationen benötigt. Der entityType „CoManagedEntity“ trifft für diese Assets zu. Im Beispiel sind Schrauben solche „CoManagedEntity“ und werden entsprechend in der BOM modelliert (Abbildung 5-11). Die konkrete Anzahl sollte beispielsweise auch modelliert werden und erscheint als Property der Entity, hier im Beispiel „Anzahl der Schrauben“ (0173-1#01-RAA001#001), einem Merkmal, das es bei ECLASS gibt (Abbildung 5-11). Das Merkmal „Anzahl der Schrauben“ ist ein ECLASS-Merkmal und wird zusätzlich in die ConceptDescription aufgenommen (Abbildung 5-13).

In Abbildung 5-12 ist dargestellt, dass es ein Teilmodell geben kann, in dem die Schraube näher beschrieben wird, hier exemplarisch nur mit der Property „AusfuehrungdesGewindes“. Dieses

Teilmodell ist in der VWS, in der die BOM beheimatet ist, untergebracht. Außerdem bekommt dieses Teilmodell eine Referenz auf die dazugehörige CoManagedEntity. Diese Referenz benötigt eine maschineninterpretierbare semanticId, um die Tatsache, dass es sich um die Verbindung zwischen der in der BOM enthaltenen Entity und ihrer Beschreibung in einem Teilmodell handelt, eindeutig auszudrücken. Es ist jedoch auch umgekehrt möglich, dass die Entity auf das sie beschreibende Teilmodell zeigt. Die umfangliche Liste der Properties der Schrauben kann unter der beiliegenden Referenz eingesehen werden.

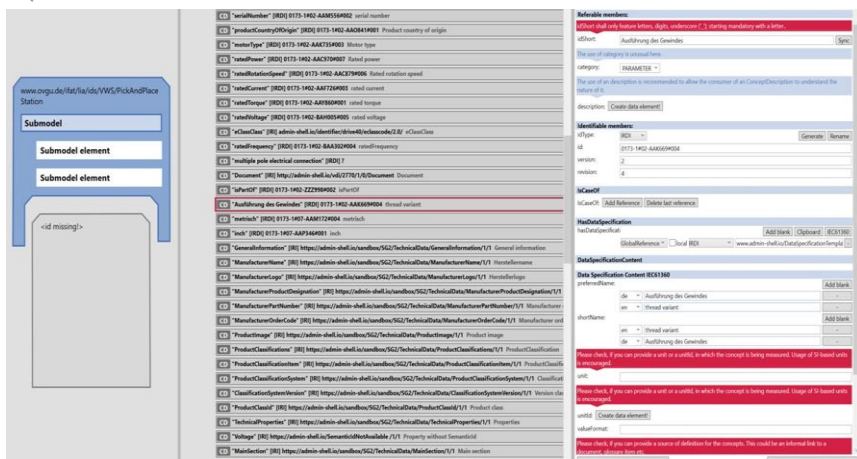


Abbildung 5-13: Eintrag des Merkmals "Anzahl der Schrauben" in die Concept Description nach IEC 61360/ECLASS

https://www.eclasscontent.com/index.php?id=23110119&version=11_1&language=de&action=det.

Abbildung 5-13 zeigt einen Ausschnitt der ConceptDescriptions die in der P&P-Station modelliert werden.

Das Kind-Attribute – Typen/Templates und Instanzen

Ziel des Kapitels

Erläuterung der Anwendung des Attributes „Kind“ für Assets (assetKind) und Teilmodelle (modelingKind) und Zusammenhänge zwischen diesen Konzepten.

Verwendete Modellelemente des VWS-Metamodells

- Kapitel 4.2 (allgemeine Erläuterungen und Beispiele)
- Kapitel 4.7.5, S. 66 und S. 67: Asset-Kind
- Kapitel 4.7.2.5, S. 57: ModelingKind

Erläuterungen und Darstellung im Klassendiagramm und im Package Explorer

In diesem Kapitel wird anhand des Beispiels der P&P-Station die Nutzung der unterschiedlichen durch das Kind-Attribute definierten Ausprägungen der Modellelemente erklärt. Es wird zwischen AssetKind und ModelingKind unterschieden. Wie der Name AssetKind sagt, bezieht sich dieses Attribut nur auf die Unterscheidung bei Assets. ModelingKind wird zur Unterscheidung von VWS-MetamodelElementen verwendet.

Die Modellierung könnte mit den Assets beginnen, die vom AssetKind=Type und

AssetKind=Instance sein können. Die Modellierung (Abbildung 5-14) sieht prinzipiell vor, dass für wiederkehrende Komponenten der P&P-Station (z. B. die Arbeitsstationen für das Bohren) die Teilmodelle jeweils für den Asset-Type (WorkstationType1 und WorkstationType2 jeweils mit Kind = Type) und die Asset-Instance (WorkstationType1_1 und WorkstationType1_2 jeweils mit Kind = Instance) zu modellieren sind.

Das bedeutet, dass es zunächst eine VWS für den Asset-Typ (je für WorkstationType1_Type und WorkstationType2_Type) und je eine VWS für die Asset-Instanzen gibt (Workstation_1 und Workstation_2, jeweils von WorkstationType1_Type abgeleitet (derivedFrom), WorkstationType2_Type hat in Abbildung 5-14 keine Instanz). VWS besitzen das „Kind“-Attribut nicht, es ergibt sich aus der Zuordnung zu dem Asset. Für die Asset-Instanz gibt es typischerweise eine VWS, die bei Auslieferung mitgegeben wird und es kann weitere VWS mit zusätzlichen Teilmodellen geben (WorkstationStatusActual_2), die z. B. Aktualdaten von Properties beschreiben. Diese können in der Typ-Beschreibung eine Range haben, zur Laufzeit dann aber Werte aus dem Asset nutzen, die nach Umbauten verändert oder ergänzt werden oder die neue Features freischalten (z. B. Auswertungsfunktion). Die Teilmodelle, die erst zur Laufzeit mit Assetdaten gefüllt werden, können auch schon in der Auslieferung-VWS enthalten sind.

Teilmodelle können von Kind=Template und Kind=Instance sein. Teilmodelle mit Kind=Template sind vor allem für standardisierte Teilmodelle gedacht. So sind z. B. das Nameplate oder die Dokumentation (in Abbildung 5-14 nicht dargestellt) solche Teilmodelle. In den Templates werden alle SubmodelElements (Properties, Ranges, Files, Collection ...) angelegt. Die entsprechenden Attribute dürfen Default-Werte besitzen. Zusätzlich zu den SubmodelElementen gehören die ConceptDescription-Einträge zur Teilmodell-Template-Festlegung dazu, da hier die Definitionsgrundlagen hinterlegt sind. Alle Teilmodelle einer VWS, sowohl für

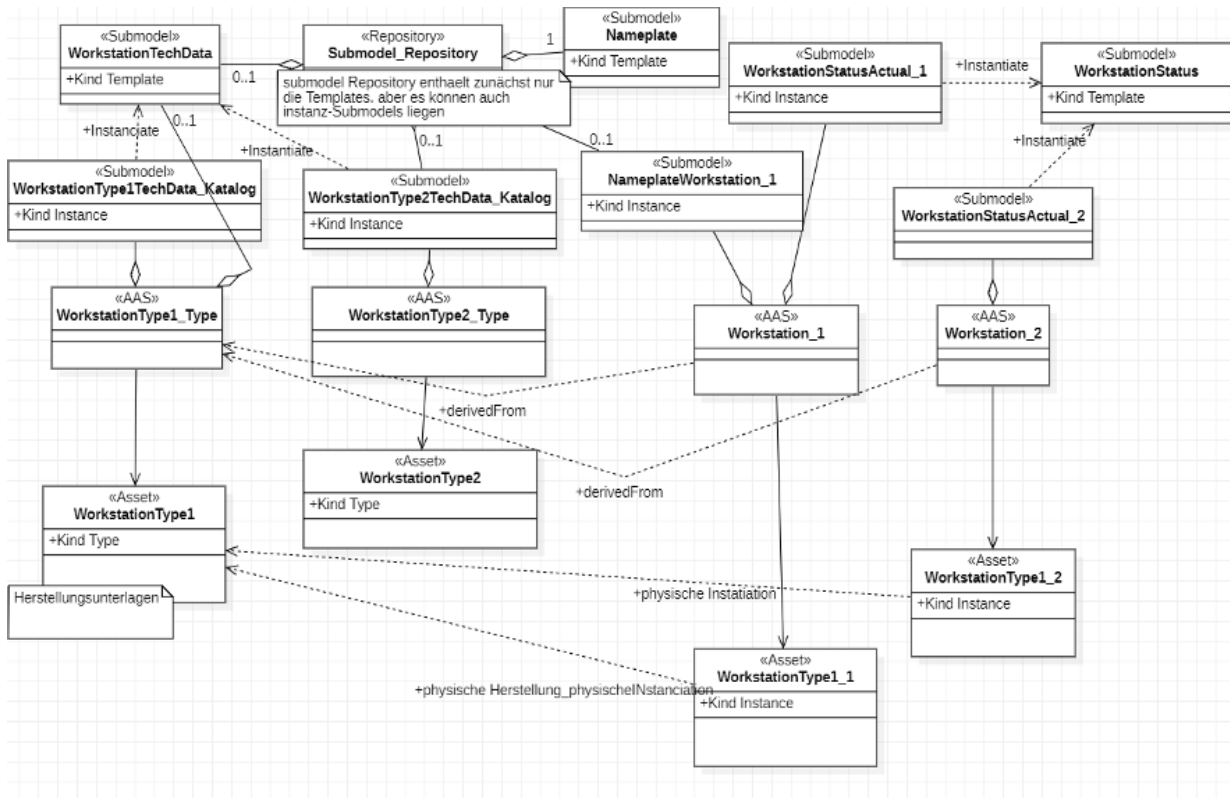


Abbildung 5-14: Modellierungsbeispiel von Asset, VWS und Teilmodelle hinsichtlich Type, Template und Instance

AssetKind=Type oder AssetKind = Instance, sind mit dem Teilmodell-Attribute Kind=Instance definiert. Aus Teilmodell-Template abgeleitete Teilmodelle können zusätzliche SubmodelElemente haben. Deshalb sind sie formal gesehen keine aus einem Typ abgeleitete Instanz.

Die im Klassendiagramm entworfene Struktur wurde als Composite-VWS im Package Explorer umgesetzt (Abbildung 5-15). Darin sind die P&P-Station (VWSPickAndPlaceStation) sowie die Typen für die Bohr- und Schleifmaschinen (VWSWorkstationDrillingType und VWSWorkstationHoningType) und dem Motortyp (VWSDeltaRobotMotorType) mit den dazugehörigen Instanzen *Drilling_1, *Drilling_2, *Honing_1, *Motor_1, *Motor_2 und *Motor_3 umgesetzt. Dazu gehören die entsprechenden Assets.

Teilmodell-Templates, die als Standard dienen und von den entsprechenden Arbeitskreisen verabschiedet wurden, haben eine semanticId, die diese als solches kennzeichnen. Dadurch kann sich der Nutzer auf einen bestimmten Informationsinhalt, d. h. SubmodelElemente, verlassen. Diese semanticId beginnen mit <http://admin-shell.io/> ... z. B. <http://admin-shell.io/vdi/2770/1/0/Documentation>

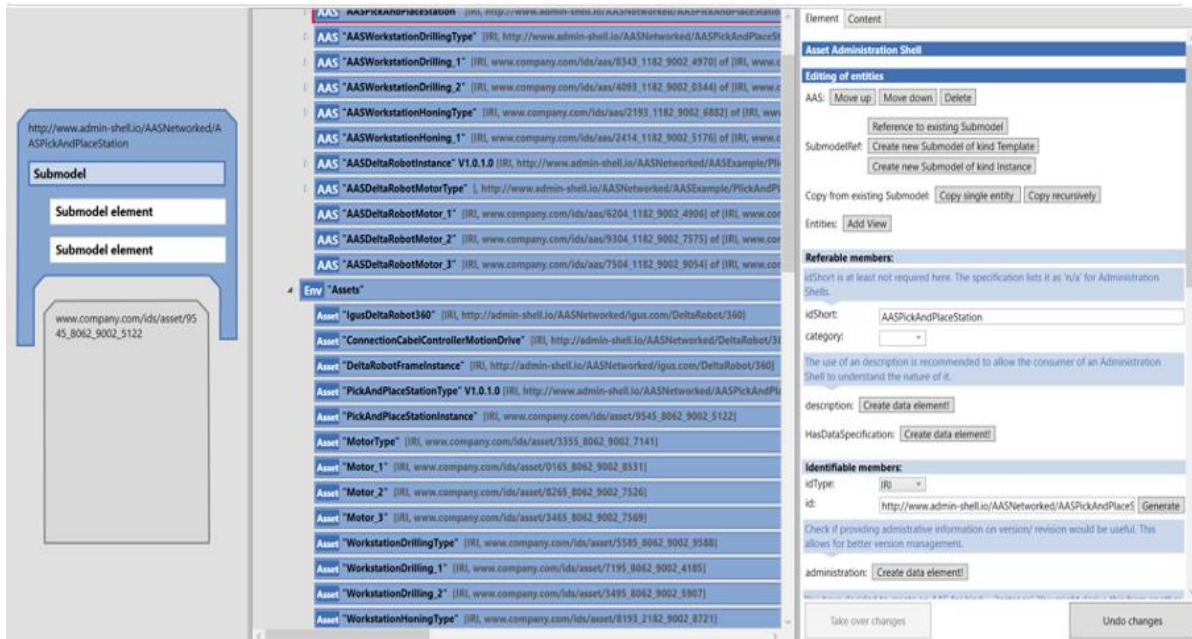


Abbildung 5-15: VWS-Übersicht im Package Explorer

14.0 Verbundkomponente

Ziel des Kapitels

Aufgabe und Aufbau der Komposition von VWS zu erläutern.

Verwendete Eingangsinformationen für das Modell

- Alle Beschreibungen der P&P-Station

Verwendete Modellelemente des VWS-Metamodells

- Kapitel 4.3, S. 33-34: Composite I4.0 Components

Erläuterungen

Assets sind in der Regel aus mehreren Komponenten zusammengesetzt. Wie in der BoM in Kapitel 5.6 dargestellt, können Assets durch SelfManagedEntity und CoManagedEntity beschrieben werden. CoManagedEntity-Assets werden in Teilmodellen beschrieben, die in der VWS sind, in der sich auch das BoM-Teilmodell befindet. SelfManagedEntity-Assets haben ihre eigene VWS.

Daraus ergibt es sich, dass mehrere VWS der Teilkomponenten und die der übergeordneten VWS zusammengehören. Dies wird als Composite I4.0 Component und letztlich als Composite VWS bezeichnet. Die VWS stehen untereinander in Aggregationsbeziehungen, die auch aus mehr als einer Hierarchiestufe bestehen kann.

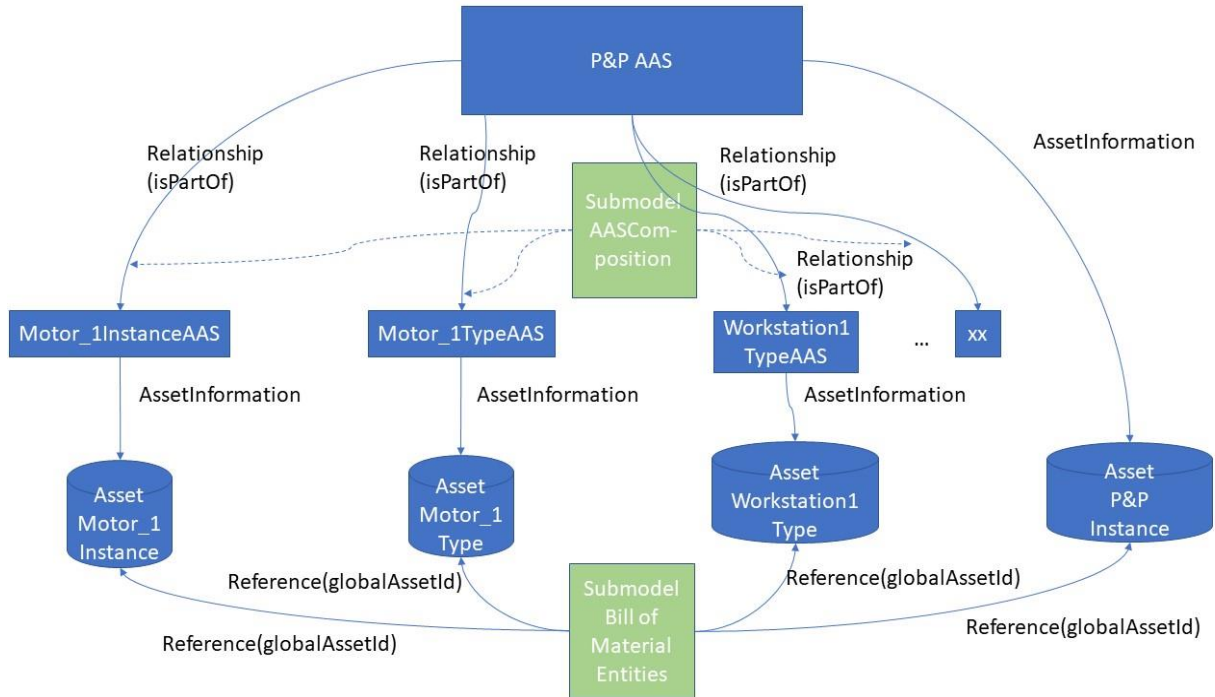


Abbildung 5-16: Composition von VWS als Übersicht

Dafür wird eine Beschreibung in der jeweils zusammengesetzten VWS benötigt. Es wird explizit darauf hingewiesen, dass sich die Komponenten-VWS nicht in der übergeordneten VWS befinden. Die VWS bleiben eigenständig. Die Aggregationsbeziehung wird durch das SubmodelElement "Relationship" beschrieben.

Hier wird als Best-Practice-Lösung ein Teilmodell vorgeschlagen, in dem die Relationships für je eine "isPartOf"-Beziehung eingetragen werden (siehe Abbildung 5-16, grüne Teilmodell-Box oben). Dieses Teilmodell ist jeweils in der VWS, in der sich das BoM-Teilmodell mit den SelfManagedEntities befindet. In diesem CompositionVWS-Teilmodell sind die Beziehungen jeweils als Relationship eingetragen, z. B. als Relationship zwischen P&PVWS und Drive_1InstanceVWS. Die durchgezogene Linie ist die Aggregationsbeziehung (isPartOf). Die gestrichelte Linie verdeutlicht, dass sich die

Beschreibung als Relationship im Teilmodell befindet. Es ist der gleiche RelationshipElement-Typ, der bereits bei der Beschreibung der Beziehung der Entities in der BoM verwendet wird. Natürlich sind hier die Ids der Subjekte und Objekte der entsprechenden VWS zu verwenden. Der untere Teil der Abbildung 5-16 entspricht der aus der Beschreibung der BoM aus Kapitel 5.5.

Im Package Explorer sehen die Relationships wie in Abbildung 5-17 dargestellt aus.

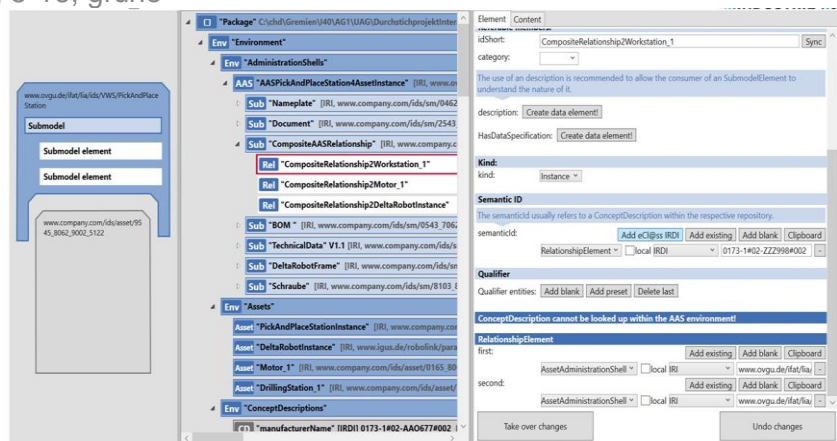


Abbildung 5-17: Relationship beschreibt die Beziehungen zu den VWS der Anlagenkomponenten

Concept Descriptions and Data Specifications

Ziel des Kapitels

Die Aufgabe und Aufbau der Concept-Description zu erklären und die Wirkungsweise der DataSpecification zu erläutern.

Verwendete Modellelemente des VWS-Metamodells

- Kapitel 4.8, S. 94 ff: Predefined Data Specification Templates
- Kapitel 6.2.4 S. 124: Embedded Data Specification

Erläuterungen

Das VWS-Informationsmodell bietet vielfältige Typen an Modellelementen, die entsprechend der verschiedensten Use Cases entlang des Lebenszyklus die Assets beschreiben. Diese Vielfalt spiegelt sich vor allem in den SubmodelElements wider, die die Teilmodelle ausfüllen. Jedes Modellelement wird durch eine Reihe von Attributen detaillierter beschrieben, die die Modellelemente charakterisieren. Die Attribute sind abhängig vom Typ des Modellelements.

In der industriellen Praxis werden bereits viele der in der VWS aufgenommenen Modellelemente verwendet. Sie sind für die unterschiedlichen Einsatzfälle mit zugeschnittenen Attributen versehen. Eines der SubmodelElements ist die Property. Sie beschreibt die Variablen, Parameter und Konstanten der Assets. Nun hätte man erwarten können, dass für diese Properties im VWS-Metamodell ein Satz von Attributen definiert wird. Dieser Weg wurde nicht beschritten. Wie in Kapitel 5.3 und 5.4 beschrieben, wird eine Trennung von Identifikation von Modellelementen und semantischer Definition vorgenommen. Deshalb wird für die SubmodelElements nur ein Minimalsatz an

Attributen festgelegt und durch Nutzung der semanticId auf die detaillierte Beschreibung verwiesen. Als Default-Spezifikation für die Attribute wird die Definition nach IEC 61360 verwendet. Diese entspricht der Embedded-DataSpecification im Metamodel. Das hat mehrere Gründe und Effekte:

- Es wird ausdrücklich empfohlen, Properties zu verwenden, die bereits in Vokabularen enthalten sind. Von der Plattform Industrie 4.0 werden industrielle Standards wie ECLASS oder Companion Specifications der OPC Foundation favorisiert. Diese Properties sind bereits Ergebnis eines Konsensverfahrens und haben deshalb einen größeren Gültigkeitsraum als herstellereigene Definitionen.
- Bei der Definition der Teilmodelle wird sichtbar, welche Properties noch nicht durch einen Standardisierungsprozess gelaufen sind. Es kann dann entschieden werden, ob sich eine Standardisierung lohnt oder diese nicht notwendig ist.
- Die Normen und Standards enthalten auch Festlegungen zu ausgewählten Werten, die z. B. Maßeinheiten, Farben oder andere zum einem Asset-Typ gehörende Enumerations. Auch diese meist durch Strings oder individuell festgelegte Nummerierungen repräsentierten Werte sind dann eindeutig identifizierbar.
- Der eindeutig definierte Satz an Attributen ermöglicht es den Anwendungen, diese Festlegungen als das "Normal" zu implementieren. Das schränkt die Vielfalt der Auswertelgorithmen wesentlich ein und vereinfacht die Anwendungsentwicklung.

Das Metamodel der VWS legt zugeschnitten für jeden SubmodelElement-Type einen empfohlenen bzw. verpflichtenden Subset der IEC 61360 Attribute fest (Kapitel 4.8.2 in [BMW 2020]).

Das Prinzip sieht vor, dass VWS-Modellelemente, die von HasDataSpecification erben bei der Instanziierung einen

DataSpecificationContent umsetzen, der zwar variabel sein kann, in der jetzigen Spezifikation aber nach IEC 61360 umgesetzt wird. Die Attribute von DataSpecificationContent werden also nur bei einer Instanziierung eines Modellelements den entsprechenden Instanzen hinzugefügt (Kapitel 6.2.4 in [BMW 2020]).

In diesem Dokument wird die Möglichkeit, andere Attribute als die im Metamodell vordefinierten und die von IEC 61360 vorgesehenen zu nutzen, nicht verwendet. Für das Verständnis der Details dieses Mechanismus ist die Spezifikation (Metamodell 4.8 und 6.4.2) zu verwenden.

Die hier verwendete ConceptDescription ist genau die detaillierte Attribute-Auswahl für die Modellelemente, die mit IEC 61360 vorgegeben ist. Eine Übersicht gibt Abbildung 5-18.

Die Menge der ConceptDescriptions, die von einer VWS durch eine semanticId referenziert werden, richtet sich nach der Anzahl der VWS-Modellelemente, die die semanticId ausgewiesen haben. Die ConceptDescription ist zwar nicht Bestandteil einer VWS (sie beinhaltet übergreifend standardisierte

Beschreibungen von Modellelementen, z. B. Properties), wird jedoch in dem PackageExplorer im Environment-Abschnitt (env) unter ConceptDescription mit aufgenommen. Es ist nicht verpflichtend, alle ConceptDescriptions in die AASX-Beschreibung aufzunehmen. Concept-Descriptions, die aus einer Norm oder einem Standard entnommen sind (ECLASS oder IEC CDD), sind damit durch ihre semanticId, d. h. die IRDI, eindeutig identifiziert. Verbirgt sich dahinter eine korrekte und vollständige Beschreibung, kann die Anwendung die Attribute auch eindeutig interpretieren.

Das prinzipielle Konzept, das hier wirkt, ist die sogenannte Data Specification. Diese kommt zunächst nur für die Integration von IEC 61360-Attributen zum Tragen. Obwohl das Data Spezifikation Konzept wegen der Option auf verschiedene Attributdefinitionen recht kompliziert erscheint, ist die Umsetzung sehr einfach. Die VWS-Modellelemente erhalten eine semanticId auf die ConceptDescription und können deshalb die IEC 61360 Attribute in dem VWS-Modell verwenden. Die entsprechende IRDI oder IRI der ConceptDescription ist dann der Wert der semanticId des VWS-Modellelements. Der

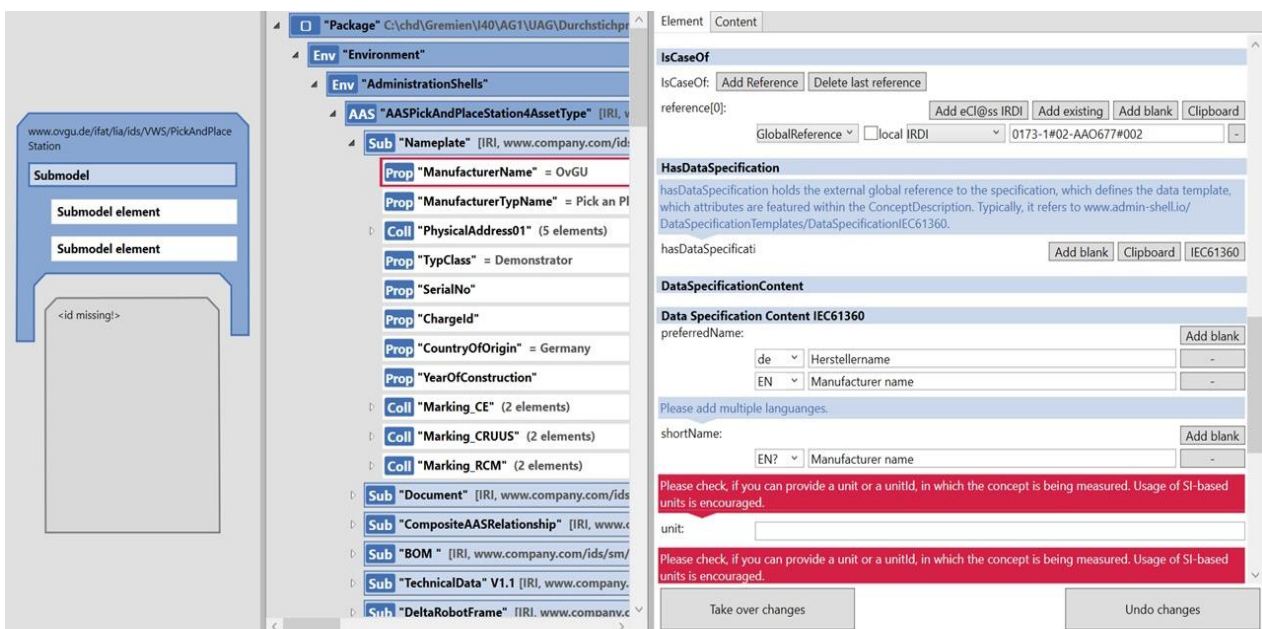


Abbildung 5-18: Attribute der DataSpecification nach IEC 61360 einer Property am Beispiel "manufacturerName"

Effekt für die Anwenderprogrammierer ist es, dass bei standardisierter ConceptDescription mit deren Attribute-Definitionen und Werten programmiert werden kann.

Teilmodell-Templates

Teilmodell-Templates sind Schablonen, die für häufig wiederkehrende Aspekte der Assets oder von Anwendungsfällen Teilmodelle mit den dafür benötigten Modellelemente vorgefertigt bereitstehen. Im Rahmen der I4.0-Aktivitäten werden eine Reihe von Teilmodell-Templates standardisiert. Beispiel sind Nameplate, Documentation, TechnicalData, Simulation und viele andere mehr. Sie dienen als Kopiervorlage. Die standardisierten Teilmodell-Templates werden sowohl für die Entwickler der VWS als auch für deren Nutzer zugänglich sein.

Das Vorgehen bei der Nutzung der Templates ist nicht so streng, wie man es bei der in der Objektorientierung üblichen Typ/Instanzbeziehung kennt. Einerseits sind optionale Elemente in den Templates, die bei der Nutzung weggelassen werden können und andererseits kann der Nutzer weitere Elemente hinzufügen oder Teile mehrmals instanziiieren. Außerdem können sich Hersteller von Assets aus den standardisierten Templates eigene zugeschnittene Templates erstellen und dann herstellerspezifisch umsetzen (Abbildung 5-19). Die in den VWS instanziierten Teilmodelle sollen durch die Nutzung der semanticId des zugehörigen Teilmodell-Templates anzeigen, dass sie aus diesem Template entstanden sind. Dies ermöglicht es, den verpflichtenden Teil des Teilmodells bei der Anwendung voraussetzen zu können. Ein Beispiel dafür ist die BoM (siehe Kapitel 5.5).

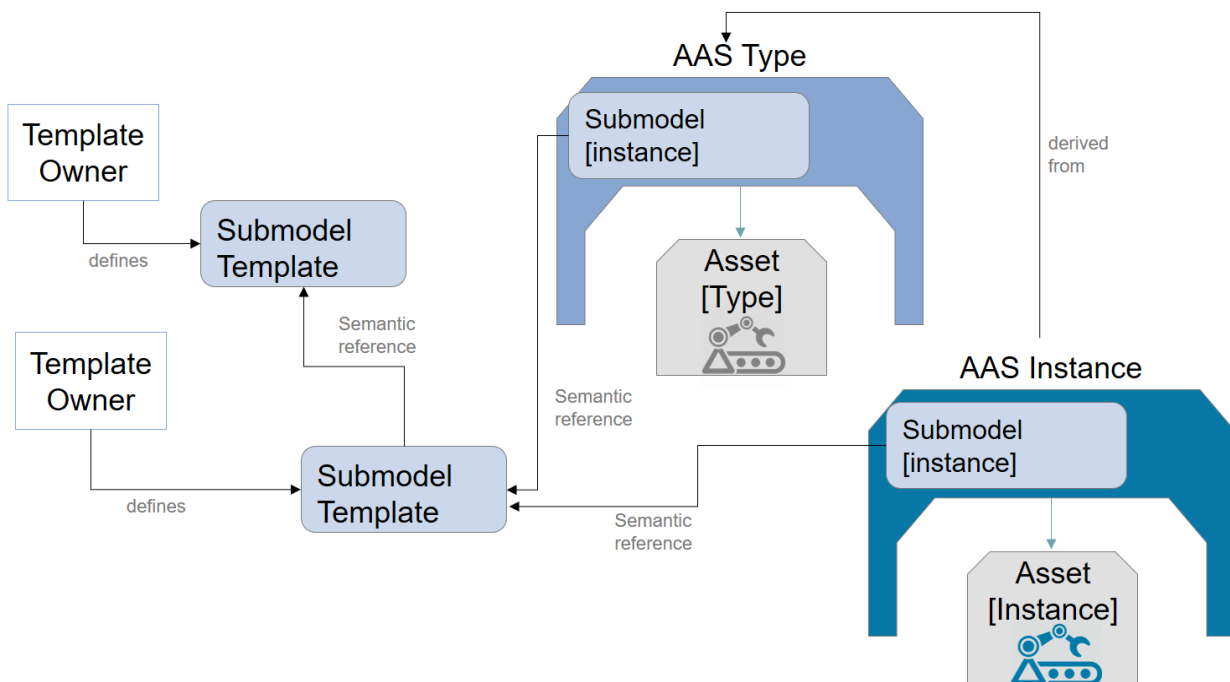


Abbildung 5-19: Zusammenhang zwischen Teilmodell-Templates (Submodel-Templates) und den Teilmodellen in den VWS

(6) Metamodell Referenzen

Die folgende Tabelle enthält die wesentlichen VWS-Modellelemente in alphabetischer Reihenfolge. Den Elementen ist jeweils eine Referenz innerhalb dieses Dokuments zugeordnet, das in ein Kapitel verweist, in dem das Element und seine Verwendung beschrieben ist. Dies soll bei der Suche nach einzelnen Elementen helfen, ohne lange in den Kapiteln suchen zu müssen.

Modellelemente	Erläuterungen und Referenzen
Administrative Information	Administrative Information sind Versions- und Revisionsattribute der Identifier Verwendung in Kapitel 5.4
Asset	Verwendung in Kapitel 5.4
AssetAdministrationShell	Verwendung in Kapitel 5.4
AssetInformation	Verwendung in Kapitel 5.2
AssetKind	Verwendung in Kapitel 5.6
Capability/Fähigkeit	Verwendung in Kapitel 4.5, 4.8
ConceptDescription	Verwendung in Kapitel 5.8
Entity	Verwendung in Kapitel 5.5
Fähigkeit/Capability	Verwendung in Kapitel 4.5, 4.8
File	Verwendung in Teilmodell "Documentation"
hasDataSpecification	Verwendung in Kapitel 5.8
HasKind	Dieses Attribut beschreibt, ob ein Modellelement als Instanz betrachtet wird oder nicht. Assets können auch als Type modelliert werden, andere Modellelemente werden, wenn sie keine Instanzen sein sollen als Template gekennzeichnet. Dies sind dann Vorlagen, aus denen Instanzen abgeleitet werden können, ohne die strengen Type-Instanz-Ableitungsregeln zu beachten. Verwendung in 5.3
HasSemantics	Wird als semanticId umgesetzt und Verwendung in Kapitel 5.3 sowie den Kapiteln 4.3, 4.5, 4.7
Identifiable	Verwendung in Kapitel 5.2
Identifier	IRDI, IRI, Custom Identifier sind Werte des globalen Identifier Typs "idType"

	Verwendung in Kapitel 5.2
Instance	Verwendung in Kapitel 5.6
ModellingKind	Verwendung in Kapitel 5.6 sowie in den Kapitel 4.3 und 4.4
Operation	Verwendung in Kapitel 4.5, 4.8
OperationVariables	Verwendung in Kapitel 4.5, 4.8
Property	Verwendung in Kapitel 4.3 und 4.7
Range	Verwendung in Kapitel 4.3, 4.7 und 5.6
Referable	Verwendung in Kapitel 5.2
ReferenceElement	Verwendung in Kapitel 5.5
RelationshipElement	Verwendung in Kapitel 5.5 und 5.7
Submodel/Teilmodell	in fast allen Kapiteln
SubmodelElementCollection	z.B. im Teilmodell "TechnicalData" Kapitel 4.3
Teilmodell/Submodel	in fast allen Kapiteln
Template	Verwendung in Kapitel 5.9 und 5.6
Type	Verwendung in Kapitel 5.6
View	Kapitel 4.3 und 4.4

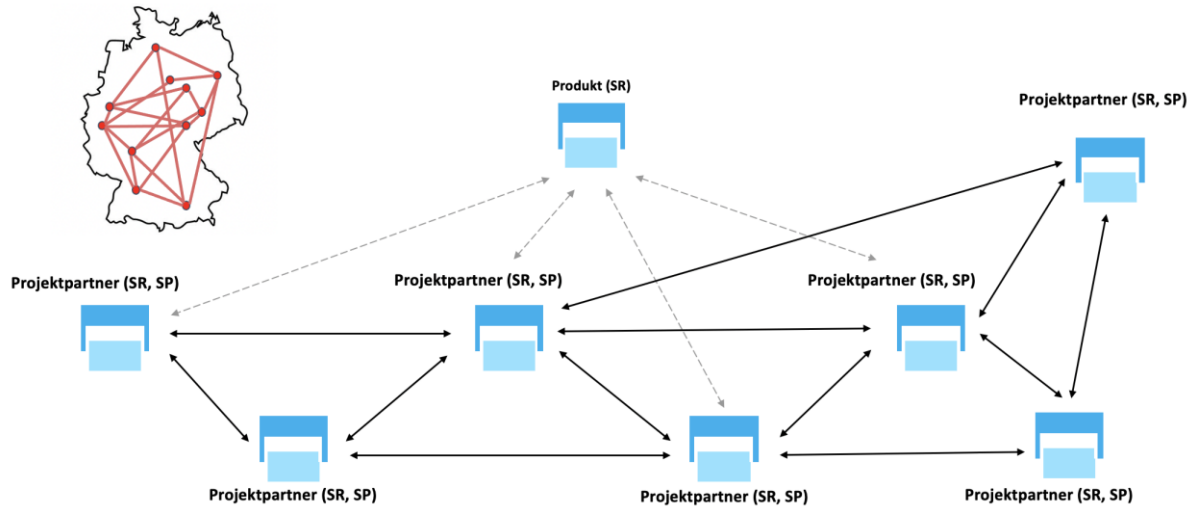


Abbildung 7-1: Der in diesem Dokument beschriebene Demonstrator ist Teil eines entstehenden vernetzten deutschlandweiten I4.0-Demonstrators

(7) Zusammenfassung und Ausblick

Dieses Dokument zeigt an einer beispielhaften Anlage, wie die Modellierungselemente der VWS verwendet werden können. Es möchte damit unterstützen, typische Komponenten von Produktionsanlagen zu beschreiben. Diese Anlage ist ein Demonstrator, der in einem deutschlandweit entstehenden Netz von Demonstratoren (Abbildung 7-1), Anwendungsszenarien für die VWS und deren Interoperabilität aufzeigt.

Nach einer einführenden Darstellung der Modellierungsprinzipien (Kapitel 3) wird in zwei getrennten Kapiteln die konkreten Modellierungsvorschläge (Kapitel 4) ausgewählter Komponenten der Anlage und wesentliche Modellierungskonzepte (Kapitel 5) beschrieben.

Kapitel 6 dient als Register, wenn man sich über ein einzelnes Modellelement informieren möchte.

Die Modellierungsvorschläge sind alle mit dem AASX Package Explorer umgesetzt und stehen als AASX-File unter <http://liabroker.ddns.net:51001/> abrufbar zur Verfügung.

Während der Arbeiten an diesem Dokument traten eine Reihe von Fragen auf, die mit den entsprechenden UAGs der AG1 der Plattform I4.0 diskutiert wurden. Einige wurden in die Q&A - Liste aufgenommen.

Literaturverzeichnis

[PLA17]	Plattform Industrie 4.0: Beziehungen zwischen I4.0-Komponenten – Verbundkomponenten und intelligente Produktion: Fortentwicklung des Referenzmodells für die Industrie 4.0–Komponente SG Modelle und Standards, BMWi Public Relations, Juni 2017, URL: https://www.zvei.org/fileadmin/user_upload/Presse_und_Medien/Publikationen/2017/Juni/Beziehungen_zwischen_I4.0_Komponenten/Beziehungen-zwischen-I4.0-Komponenten-zvei.pdf (Stand: 20.02.2021).
[PLA20]	Bundesministerium für Wirtschaft und Energie (BMWi) (Herausgeber): Specification Details of the Administration Shell – Part 1: The exchange of information between partners in the value chain of Industrie 4.0; Version 3.0 RC01 Federal Ministry for Economic Affairs and Energy (BMWi). Plattform Industrie 4.0, Berlin 2020. URL: https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/Details-of-the-Asset-Administration-Shell-Part1.html
[I40Sp18]	Bundesministerium für Wirtschaft und Energie (BMWi) (Herausgeber): I4.0 Sprache. Diskussionspapier Plattform I4.0. April 2018.
[FES01]	Festo Didactic SE: Modul Bohren, URL: https://www.festo-didactic.com/de-de/lernsysteme/mechatronische-systeme-mps/projektbaukasten/komponenten-module-projektbaukasten/modul-bohren.htm?fbid=ZGUuZGUuNTQ0LiEzLjE4LjcxMC4zOTcx (Stand: 22.02.2021).
[FES02]	Festo Didactic SE: Modul Band, URL: https://www.festo-didactic.com/de-de/lernsysteme/mechatronische-systeme-mps/projektbaukasten/komponenten-module-projektbaukasten/modul-band.htm#prd_det_accessories (Stand: 22.02.2021).
[WAG01]	WAGO Kontakttechnik GmbH & Co. KG: Controller PFC200, URL: https://www.wago.com/de/sps/controller-pfc200/p/750-8212 (Stand: 22.02.2021).
[IGUS01]	IGUS: Technische D-± DC-, EC/BLDC-Motorsteuerung Handbuch V2.3 (Handbuch dryve D1 DE.pdf)
[IGUS02]	IGUS: dryve D1, ST- DC- EC/BLDC-Motor Control System Manual V2.3 (IGUS, DOM Mai 2020))
[IGUS03]	IGUS: stepper motor Manual, MOT-AN-S_EN_202003#1, Mai 2020
[DCI20]	Bundesministerium für Wirtschaft und Energie (BMWi) (Herausgeber): Describing Capabilities of Industrie 4.0 Components. Diskussionspapier Plattform I4.0. 2020
[DNI20]	Bundesministerium für Wirtschaft und Energie (BMWi) (Herausgeber): ZVEI Digital Nameplate for industrial equipment. Diskussionspapier Plattform I4.0. 2020, URL: https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/Submodel_Templates-Asset_Administration_Shell-digital_nameplate.pdf?__blob=publicationFile&v=2
[DTD20]	Bundesministerium für Wirtschaft und Energie (BMWi) (Herausgeber): ZVEI Generic Frame for Technical Data for Industrial Equipment in Manufacturing (Version 1.1). Diskussionspapier Plattform I4.0. 2020, URL:
[DD20]	Bundesministerium für Wirtschaft und Energie (BMWi) (Herausgeber): ZVEI Minimum requirements for the Handover documentation from the manufacturer to the operator based on the VDI 2770 specification. Diskussionspapier Plattform I4.0. 2020, URL:

Anhang

Identifikatoren

Tabelle A-1 illustriert Vergabe der Identifier für Teilmodelle, Assets und VWS. Instanzen: generieren und die IRI-Domäne des Asset-Herstellers bei AssetId und VWSId Type: IRI der Asset-Domäne/ evt Substrukturen /ids/VWS oder asset/Produkttypenname

Komponente	Asset ID	VWS ID
P&P Type	www.ovgu.de/ifat/lia/ids/asset/PickAndPlaceStation/Type	www.ovgu.de/ifat/lia/ids/VWS/PickAndPlaceStation/Type
P&P Instance	www.ovgu.de/ifat/lia/ids/asset/PickAndPlaceStation/Instance	www.ovgu.de/ifat/lia/ids/VWS/PickAndPlaceStation/Instance
Motor_Type	www.igus.de/ids/asset/stepper motor MOT-AN-S_type	www.igus.de/ids/VWS/stepper motor MOT-AN-S_type
Motor_Instance_1	www.igus.de/ids/asset/stepper motor MOT-AN-S_instance_1	www.igus.de/ids/VWS/stepper motor MOT-AN-S_instance_1
Motor_Instance_2	www.igus.de/ids/asset/stepper motor MOT-AN-S_instance_2	www.igus.de/ids/VWS/stepper motor MOT-AN-S_instance_2
Motor_Instance_3	www.igus.de/ids/asset/stepper motor MOT-AN-S_instance_3	www.igus.de/ids/VWS/stepper motor MOT-AN-S_instance_3
Workstation_type	www.ovgu.de/ifat/lia/ids/asset/workstation_type	www.ovgu.de/ifat/lia/ids/VWS/workstation_type
rotation_module_type	www.festo-didactic.com/de-de/lernsysteme/mechatronische-systeme-mps/projektbau-kaesten/komponenten-module-projektbaukasten/asset/rotation_module_type	www.festo-didactic.com/de-de/lernsysteme/mechatronische-systeme-mps/projektbau-kaesten/komponenten-module-projektbaukasten/VWS/rotation_module_type
conveyor_module_type	www.festo-didactic.com/inten/learning-systems/mps-the-modular-production-system/mps-modules/asset/conveyor_module_type	www.festo-didactic.com/inten/learning-systems/mps-the-modular-production-system/mps-modules/VWS/conveyor_module_type
PLC_workstation_type	www.ovgu.de/ifat/lia/ids/asset/PLC_workstation_type	www.ovgu.de/ifat/lia/ids/VWS/PLC_workstation_type
workstation_instance1	www.ovgu.de/ifat/lia/ids/asset/workstation_instance1	www.ovgu.de/ifat/lia/ids/VWS/workstation_instance1
drilling_module_instance1	www.festo-didactic.com/de-de/lernsysteme/mechatronische-systeme-mps/projektbau-kaesten/komponenten-module-projektbaukasten/asset/drilling_module_instance1	www.festo-didactic.com/de-de/lernsysteme/mechatronische-systeme-mps/projektbau-kaesten/komponenten-module-projektbaukasten/VWS/drilling_module_instance1

conveyor_module_instance1	www.festo-didactic.com/int-en/learning-systems/mps-the-modular-production-system/mps-modules/asset/conveyor_module_instance1	www.festo-didactic.com/int-en/learning-systems/mps-the-modular-production-system/mps-modules/VWS/conveyor_module_instance1
PLC_workstation_instance1	www.wago.com/de/sps/controller-pfc200/p/750-8212/asset/PLC_workstation_instance1	www.wago.com/de/sps/controller-pfc200/p/750-8212/VWS/PLC_workstation_instance1
Transport	www.igus.de/roboLink/parallel-kinematik/instance1	www.igus.de/roboLink/parallel-kinematik/VWS

Tabelle A-1: Vergabe der Identifizier für Teilmodelle, Assets und VWS

Instanzen

Generieren und die IRI-Domäne des Asset-Herstellers bei AssetId und VWSId

Teilmodell	idShort	Beispielhafte id	semanticId
Namensschild	Nameplate	www.ovgu.de/ifat/lia/ids/ass/PickAndPlaceStation/sm/4343_5072_7091_3242	http://admin-shell.io/zvei/nameplate/1/0/nameplate
	Documentation	dito	http://admin-shell.io/vdi/2770/1/0/Documentation
	TechnicalData	dito	http://admin-shell.io/sandbox/SG2/TechnicalData/Submodel/1/1
Encoder	Encoder	www.ovgu.de/ifat/lia/ids/sm/5284_3152_2002_0368	http://admin-shell.io/Encoder
Brake	Brake	www.ovgu.de/ifat/lia/ids/sm/6504_0122_9002_4971	http://admin-shell.io/Brake
OperatingData	OperatingData	www.ovgu.de/ifat/lia/ids/sm/4244_0122_9002_6259	http://admin-shell.io/OperatingData
BoM	BoM	www.ovgu.de/ifat/lia/ids/ass/PickAndPlaceStation/sm/BoM/4343_5072_7091_3242	http://example.com/id/type/submodel/BOM/1/1
alle anderen	individuell	individuell	

Package Explorer und AASX-Dateien

Als Modellierungswerkzeug wird der Package Explorer 1.9.8.1 Build date 18.5.20 verwendet.

Der AASX Package Explorer dient der einfachen und schnellen Ansicht und Erzeugung von VWS, die aktuell zur Version 2.0.1 des VWS-Metamodell konform sind. Das Tool ermöglicht, VWS in den Formaten XML und JSON zu serialisieren und zur Editierung einzulesen. So werden Konzeptbeschreibungen mit ECLASS-IRDIs automatisch angelegt und referenziert. Mit Import- und Export-Funktionen für z. B. BMEcat, AutomationML oder OPC UA können schnell andere Datenformate und relevante Firmendaten integriert werden. Der AASX Package Explorer ist kostenfrei herunterladbar unter github.com/admin-shell. Das Tool ist eine Open Source Implementierung, die unter der Eclipse Public License 2.0 (EPL-2.0) steht.

Installation Package Explorer: <https://github.com/admin-shell-io/AASX-package-explorer>

Screencasts: <http://admin-shell-io.com/screencasts/>

VWS-Beispiele: <http://liabroker.ddns.net:51001/>

Autoren

Alexander Belyaev, Otto von Guericke Universität

Christian Diedrich, Otto von Guericke Universität

Daniel Espen, Fraunhofer IESE

Benno Lüdicke, Expleo Germany GmbH

Torben Miny, RWTH Aachen

Anja Mrosowski, Mitsubishi Electric Europe B.V.

Matthias Müller, Mitsubishi Electric Europe B.V.

Jörg Neidig, Siemens AG

Stefan Pollmeier, ESR Pollmeier GmbH

Johannes Reich, SAP SE

Pascal Rübel, Technologie-Initiative SmartFactory KL e.V.

Manuel Sauer, SAP SE

Tizian Schröder, Otto von Guericke Universität

Olaf Ulrich, Siemens AG

Chris Urban, Otto von Guericke Universität

Bernd Vojanec, Wittenstein SE

Michael Wiczorek, Siemens AG