

# **Semantische Allianz 4.0: Semantische Inhalte für Industrie 4.0**

Constantin **Hildebrandt**, Helmut-Schmidt-Universität Hamburg

André **Scholz**, Helmut-Schmidt-Universität Hamburg

Alexander **Fay**, Helmut-Schmidt-Universität Hamburg

Tizian **Schröder**, Otto-v.-Guericke Universität Magdeburg

Thomas **Hadlich**, Otto-v.-Guericke Universität Magdeburg

Christian **Diedrich**, Otto-v.-Guericke Universität Magdeburg

Martin **Dubovy**, Rösberg Engineering, Karlsruhe

Christian **Eck**, eCI@ss e.V., Köln

Ralf **Wiegand**, eCI@ss e.V., Köln

## **1. Kurzfassung**

Kooperation und Kollaboration technischer Systeme ist in verschiedenen Anwendungskontexten eines der wesentlichen Ziele der Vision der Industrie 4.0. Hierzu ist es jedoch notwendig, eine geeignete semantische Basis für den Informationsaustausch der technischen Systeme zu definieren. Dies ist das Ziel des Projektes SemAnz4.0. Im Rahmen des Projektes wird, basierend auf internationalen Standards, die maßgeblich von deutscher Seite vorangetrieben werden, ein Konzept entwickelt, welches die Informationsmodellierung und den Informationsaustausch technischer Systeme im Kontext von Industrie-4.0-Anwendungsszenarien ermöglichen soll. Die Normen und Standards bzw. die sie unterstützenden Organisationen bilden damit perspektivisch eine „Semantische Allianz für Industrie 4.0“ – daher das Akronym „SemAnz4.0“. Der vorliegende Beitrag stellt ein Konzept zur Systemmodellierung dar, das auf der Nutzung einer modellübergreifenden Semantik basiert. Auf diesem Konzept aufbauende Systemmodelle bilden die für den Anwendungskontext des Systems notwendigen Informationen ab, sodass sie durch andere Systeme interpretierbar sind, und stellen somit die Basis für die Kooperation bzw. Kollaboration dieser Systeme dar. Das SemAnz4.0-Konzept wird in diesem Beitrag sowohl technologieunabhängig dargestellt als auch für ein Beispiel der verfahrenstechnischen Domäne mit Hilfe von AutomationML und eCI@ss implementiert.

## **2. Einleitung**

Das Konzept „Industrie 4.0“ (I40), welches entsprechend [1] auf die Kollaboration und Kooperation (KnK) technischer Systeme setzt, wird sowohl in Deutschland als auch international als Schlüsselinnovation für die Zukunft verstanden und mit Hochdruck in Forschung und Industrie verfolgt [2]. Im Kontext der I40 bezieht sich die KnK technischer

Systeme sowohl auf eine horizontale Dimension, in der ein möglichst hoher Grad an Automatisierung in den Wertschöpfungsnetzwerken erreicht werden soll, als auch auf eine vertikale Dimension, in der Systeme von der Sensor-Aktor-Ebene bis zur Unternehmensleitebene in einer durchgängigen Lösung integriert sind [3]. Zur Realisierung einer derartigen Interaktion technischer Systeme ist es unerlässlich, die Semantik ausgetauschter Informationen entsprechend zu beschreiben, sodass von allen beteiligten Systemen der Informationsgehalt ausgetauschter Nachrichten gleichermaßen interpretiert werden kann [4]. Betrachtet man die Anwendungsszenarien der Plattform Industrie 4.0 (AG2) [5], so ist das Thema Semantik essentieller Bestandteil für acht der zehn beschriebenen Szenarien. Wie in [3] dargestellt, besteht jedoch noch erheblicher Handlungsbedarf hinsichtlich Methoden und Prozessen zur Erstellung, Qualifizierung und Prüfung einer während des gesamten Lebenszyklus verwendbaren Semantik.

Der vorliegende Beitrag fokussiert auf die Modellierung mechatronischer Systeme und stellt dar, wie die Informationen im Engineering gewonnen und so modelliert werden können, dass sie anschließend in der Betriebsphase für bestimmte Anwendungen automatisiert genutzt werden können. Kapitel 2 stellt hierzu drei wesentliche Anforderungen vor und stellt das SemAnz4.0-Konzept vor. Kapitel 3 beinhaltet das Systemmetamodell, und Kapitel 4 beschreibt das Eigenschaftsmetamodell des Konzepts. Kapitel 5 stellt an einem Beispiel dar, wie das SemAnz4.0-Konzept in Use-Cases der I40 genutzt werden kann, während Kapitel 6 die Ergebnisse des Beitrags zusammenfasst.

### **3. Konzept zur Modellierung semantischer Inhalte**

Die Arbeitsgruppe 2 (Forschung und Innovation) der Plattform Industrie 4.0 entwickelt Anwendungsszenarien, um mögliche Potentiale der Digitalisierung in der produzierenden Industrie aufzuzeigen [5]. Im Rahmen des Projektes SemAnz4.0 wurden vier dieser Szenarien ausgewählt, die in Summe sowohl horizontale als auch vertikale Integration sowie das Engineering der Systeme beinhalten. Bei den ausgewählten Szenarien handelt es sich um „auftragsgesteuerte Produktion“ (AGP), „smarte Produktentwicklung für smarte Produktion“ (SP2), „Value-based Services“ (VBS), sowie das Szenario der „wandlungsfähigen Fabrik“ (WFF). Diese Szenarien wurden über die Langfassungen, auf die in [5] hingewiesen wird, hinaus weiter konkretisiert und spezialisiert, und anhand realer Fallbeispiele wurde untersucht, welche Informationen die beteiligten Systeme in ihrem jeweiligen Anwendungskontext austauschen müssen. Aus der Analyse der so spezialisierten Anwendungsszenarien wurden Anforderungen an das Konzept zur semantischen Modellierung von Systeminformationen in

der produktionstechnischen Domäne erhoben. Diese wurden zu drei Kernanforderungen (A1 bis A3) zusammengefasst, welche nachfolgend dargestellt werden.

**A1: Sicherstellung einer modellübergreifend eindeutigen Semantik**

Alle ausgewählten Szenarien stellen die KnK technischer Systeme in den Vordergrund. Während im Szenario AGP Produktionssysteme miteinander kooperieren, um gemeinsam eine Auftragsbearbeitung zu realisieren, sind es im Szenario WFF das Produktionssystem und das Leitsystem, welche zum Zweck der Integration in die Systemumgebung miteinander kollaborieren müssen. Die eindeutige Interpretation der ausgetauschten Informationen durch alle Systeme, wie in [4] beschrieben, ist durch das Konzept sicherzustellen.

**A2: Berücksichtigung aller anwendungsspezifisch relevanten Informationsklassen**

In den Szenarien sind unterschiedliche Arten von Informationen von Interesse. Während im Szenario VBS Informationen über den aktuellen Zustand einer Anlage im Rahmen des „Plant Asset Managements“ entsprechend [6] von Interesse sein können, um z.B. eine Ausfallprognose zu erstellen, werden im Szenario WFF unter anderem Informationen über die Struktur des Systems zum Zweck der Rekonfiguration benötigt. Im Szenario AGP hingegen sind Informationen über Produkte und die Fähigkeiten von Ressourcen, die Produkte herzustellen, von Interesse [7]. Das zu erstellende Informationsmodell muss die Möglichkeit aufweisen, alle Informationen über ein System in verschiedenen Anwendungskontexten abzubilden.

**A3: Durchgängige Nutzung der im Engineering entstandenen Informationen**

Bei der Analyse der Anwendungsszenarien ist erkennbar, dass alle Informationen, die während des Betriebs der Systeme notwendig sind, bereits während des Engineerings der Systeme modelliert werden müssen. Um eine redundante Informationsaufbereitung zu vermeiden, muss das Informationsmodell in die Phasen des Engineerings integrierbar sein, sodass die während des Engineerings erstellten Informationen im späteren Anwendungskontext möglichst aufwandsminimal genutzt werden können. Die Vorteile dieser Durchgängigkeit sind in [8] beschrieben.

Zur Erfüllung der dargestellten Anforderungen wurde ein Konzept entwickelt, welches im Wesentlichen zwei Modelle nutzt. Das erste Modell repräsentiert das System, welches die Abbildung aller für den Anwendungskontext notwendigen Informationen durch drei Teilmodelle realisiert (Kapitel 3), siehe auch [9] und [10] für Modellsichten auf Systeme. Durch Integration eines Eigenschaftsmodells in das Systemmodell (Kapitel 4) wird sichergestellt, dass die Eigenschaften der Systemobjekte (beschrieben durch Merkmale, Zustände und Parameter) auch beim Austausch zwischen mehreren Systemmodellen automatisiert interpretierbar sind.

Abbildung 1 stellt dies dar. Dabei wurde das Ziel verfolgt, sowohl für das Systemmodell als auch das Eigenschaftsmodell ein Metamodell entsprechend [11] zu entwerfen, sodass eine technologieunabhängige Darstellung erfolgen kann.

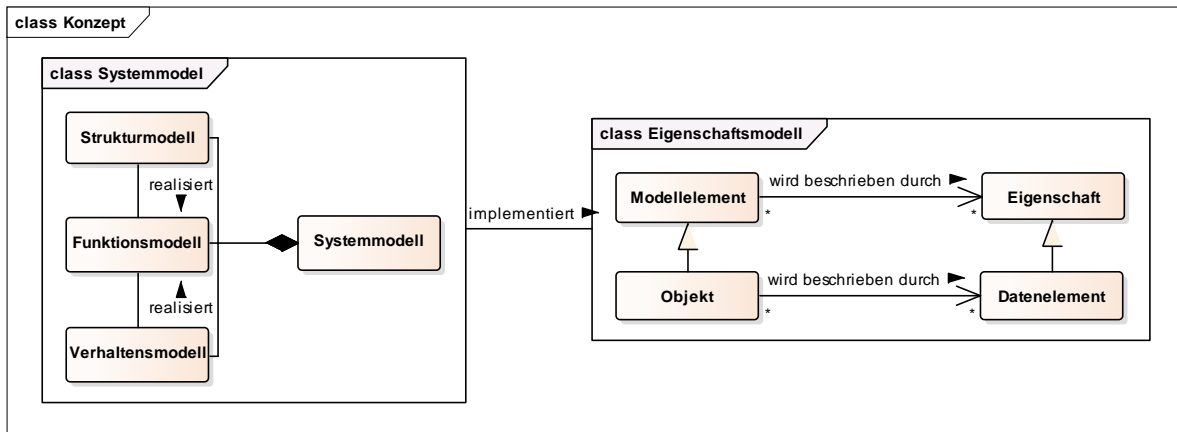


Abbildung 1: Konzept des SemAnz4.0 Informationsmodells

#### 4. Systemmodellierung

Basis des im Rahmen des Projektes SemAnz4.0 entwickelten Systemmetamodells ist die VDI2206 [12]. Demnach verfügt ein mechatronisches System über ein zu steuerndes bzw. zu regelndes Grundsystem, eine Informationsverarbeitung, die mit Hilfe von Sensorik bestimmte Informationen aus dem Grundsystem und der Umgebung aufnimmt und entsprechend einer Verhaltenslogik über Aktorik auf das Grundsystem einwirkt. Während das Grundsystem determiniert, welche Änderungen an Stoff- Energie- und Informationsflüssen vorgenommen werden sollen, stellt die Informationsverarbeitung das vordefinierte Verhalten des Systems dar, welches die Anforderungen erfüllen soll, und nutzt dazu die vorliegende Aktorik und Sensorik. Hieraus lassen sich drei wesentliche Informationsklassen mechatronischer Systeme ableiten, welche die drei Teilmodelle des Systemmetamodells bilden. Das Systemmetamodell ist in Abbildung 2 dargestellt und wird nachfolgend erläutert. Zunächst repräsentiert das Funktionsmodell (grün) das Grundsystem der VDI2206 und stellt entsprechend [13] eine lösungsneutral beschriebene Beziehung zwischen Ein- und Ausgangs- sowie Zustandsgrößen eines Systems dar. Nimmt man an, dass die Ausführung einer Funktion zu einem Prozess führt, so kann das Funktionsmodell mit Hilfe der Formalisierten Prozessbeschreibung (FPB) entsprechend VDI3682 [14] beschrieben werden. Die Verbindung vom Grundsystem zum mechatronischen System wird darin über eine technische Ressource gelöst. Das Strukturmodell (gelb) repräsentiert die im mechatronischen System enthaltenen physischen Objekte wie z.B. Aktorik und Sensorik und stellt die Material-, Informations-, Energie- und

Geometrieschnittstellen innerhalb des Systems entsprechend [15] dar, sowie eventuelle Schnittstellen zur Systemumgebung.

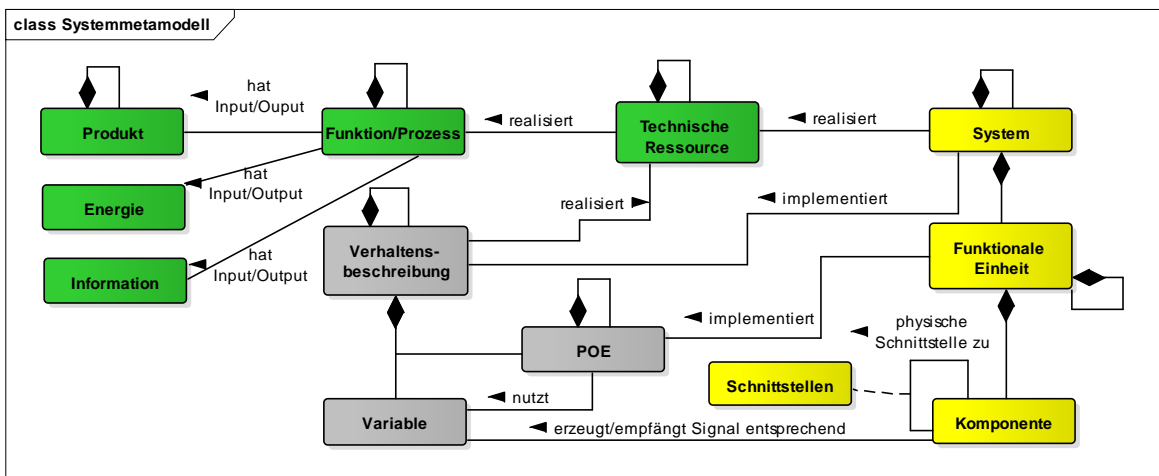


Abbildung 2: Metamodell der Systemmodellierung

In Anlehnung an [16] stellt eine Komponente ein Systemobjekt dar, welches entweder Aktorik/Sensorik/Mechanik oder Steuerungskomponente ist. Die funktionale Einheit ist in Anlehnung an [17] eine Komposition von Komponenten, welche mindestens einen Ablauf selbstständig ausführen kann. Das Verhaltensmodell (grau) bildet Prozessbeschreibungen sowie Zustandsbeschreibungen ab [9]. Unter Berücksichtigung von Anforderung A2 sind darin insbesondere Programmorganisationseinheiten und Variablen entsprechend [18] zu identifizieren. Die Anforderungen aus dem Funktionsmodell werden hardwareseitig durch das Strukturmodell und softwareseitig durch das Verhaltensmodell realisiert. Struktur- und Verhaltensmodell besitzen Verbindungen zwischen einander, um Variablen bestimmten Komponenten, sowie Abläufe bestimmten funktionalen Einheiten zuzuweisen. Das dargestellte Systemmetamodell ist technologieunabhängig und konnte für Anwendungen in der Praxis sowie Forschung in AutomationML<sup>1</sup> entsprechend [19] und Web Ontology Language (speziell: OWL-DL) entsprechend [20] erfolgreich implementiert werden, wobei AutomationML aufgrund seiner definierten Schnittstellen zu geometrischen (Struktur) und ablaufbeschreibenden (Verhalten) Modellen empfohlen wird. In Abschnitt 5 wird ein Ausschnitt des Systemmetamodells für ein Beispiel in AutomationML dargestellt. Eine vollständige Anwendung des Systemmetamodells unter Berücksichtigung aller drei Teilmodelle ist in [21] auf Basis von AutomationML zu finden.

<sup>1</sup> Eine AutomationML-Vorlage des Systemmetamodells sowie eine Regeltabelle für die metamodellkonforme Nutzung von Interfaces kann Online bezogen werden: <http://aut.hsu-hh.de/semanz40>

## 5. Modellierung von Systemeigenschaften

Neben der Strukturierung der Systeminformationen durch das in Abschnitt 3 dargestellte Systemmetamodell ist für die Repräsentation der Eigenschaften eines Systems auch eine semantische Modellierung notwendig. Durch Einführung eines entsprechenden Eigenschaftsmetamodells ist es damit unter verschiedenen Interaktionspartnern nicht nur nachvollziehbar, welchen Teil des Systems ein Objekt beschreibt, sondern auch, welche Eigenschaften das Objekt selbst beschreiben. Diese semantische Modellierung beschreibt die für die in Kapitel 2 genannten Anwendungsfälle relevanten Eigenschaften eines technischen Systems. Das hier genutzte Konzept der Eigenschaftsmodellierung, welches im Wesentlichen auf IEC 61360 [22] und [23] beruht, in [24] für das Engineering angewendet und in [25] beispielhaft umgesetzt wurde, sieht Datenelemente vor, die durch Typ- und Instanzbeschreibung gekennzeichnet sind (siehe Abbildung 3).

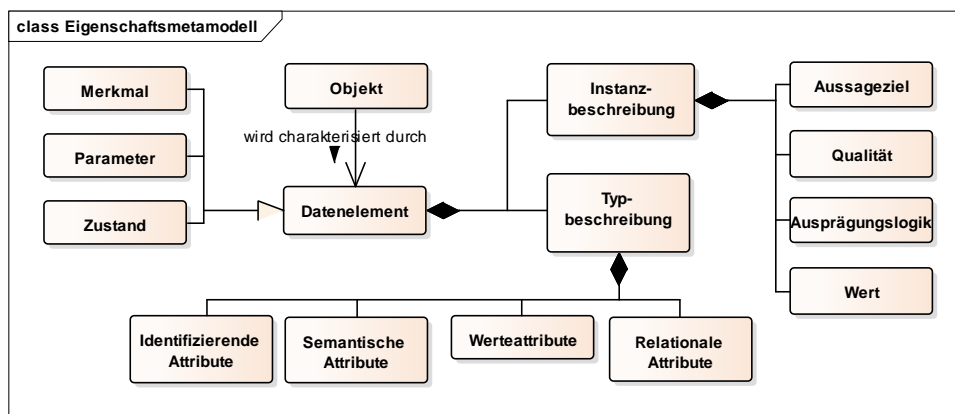


Abbildung 3: Modellierung von Datenelementen

Die Typbeschreibung - auch Datenelementtyp genannt (DET) - stellt eine Eigenschaft eines physischen oder logischen Objektes dar, die den Träger dieser Datenelemente charakterisiert. Bei der Instanziierung eines Datenelementtyps erhält dieses zusätzlich eine Instanzbeschreibung, um die Ausprägung der Eigenschaft - u. a. durch Angabe eines konkreten Wertes – abzubilden (vgl. Abbildung 3). Die folgenden Arten von Datenelementen werden unterschieden:

- Ein **Merkmal** ist eine allgemein erkennbare Eigenschaft eines Betrachtungsgegenstands, die zur Klassifizierung des Betrachtungsgegenstands genutzt werden kann. Merkmale sind Typbeschreibungen.
- Ein **Parameter** ist ein Datenelement (Variable), dessen Wert die Ausführung oder Wirkungsweise einer zugeordneten Funktion eines Betrachtungsgegenstands beeinflusst.

- Ein **Zustand** ist ein Datenelement (Variable), welches Informationen über das aktuelle Verhalten eines Betrachtungsgegenstands bereitstellt.

Die drei Arten von Datenelementen unterscheiden sich durch ihre zeitliche Veränderlichkeit. Der Wert eines Merkmals ändert sich im zeitlichen Verlauf nicht und bezieht sich auf Typen von Gegenständen. Ein Parameter bzw. Zustand hingegen kann seine Ausprägung im zeitlichen Verlauf ändern und bezieht sich auf Instanzen von Gegenständen. Den höchsten Standardisierungsgrad haben Merkmale erreicht. Für die Beschreibung der Parameter und Zustände eignen sich Industriestandards wie z.B. Profile von Feldbusprotokollen (z.B. [26,27]). Alle drei Arten von Datenelementen werden selbst durch Attribute eindeutig charakterisiert. Natürlichsprachliche Begriffsdefinitionen werden somit durch Extraktion und Ergänzung wesentlicher Informationen aus der Definition als Attribute der Datenelemente formalisiert und somit maschinenlesbar. Attribute der Typbeschreibung können gemäß IEC 61360 [22] folgendermaßen klassifiziert werden:

- Identifizierende Attribute, z.B.: ID, bevorzugter Name, Versionsnummer
- Semantische Attribute, z.B.: Definition, Quellendokument der Definition des Datenelementtyps, Formel
- Wertattribute, z.B.: Datentyp, Maßeinheit, Werteliste
- Relationale Attribute, z.B.: Übergeordnete Klasse eines Datenelementtyps

Zusätzlich zur den Attributen der IEC 61360 [22] sind im Rahmen des Konzepts ergänzende Attribute in Anlehnung an [23] aufgenommen worden, die die Instanzbeschreibung des Datenelementes näher charakterisieren. Ein wesentliches Element der Instanzbeschreibung ist das Aussageziel. Das Aussageziel gibt an, welche Rolle der Wert in einer Interaktion spielt, d.h. welche Aussage von dem Bereitsteller des Datenelements beabsichtigt wird. Es ist zwischen den folgenden Aussagezielen zu unterscheiden:

- Anforderungsausprägung (as required)
- Zusicherungsausprägung (as designed)
- Istausprägung (as built bzw. after use)

Durch die Festlegung des Aussageziels ist ein automatisierter Abgleich von Anforderungen und Zusicherungen möglich. Jede Instanz eines Datenelements referenziert über eine eindeutige ID die zugehörige Typbeschreibung. Der Aufbau der ID erfolgt für standardisierte Parameter gemäß ISO 29002-5 [28]. Zusätzlich können herstellereigenspezifische, nicht-standardisierte Parameter definiert und durch die Verwendung eines URIs identifiziert werden. Dies führt dazu, dass nicht nur innerhalb des Systemmodells eine einwandfrei interpretierbare Semantik entsteht, sondern auch modellübergreifend eine einheitliche Bedeutung der Modellelemente hergestellt wird.

## 6. Nutzung der Modelle in Industrie 4.0 Anwendungsszenarien

Im vorliegenden Beispiel soll gezeigt werden, wie das vorgestellte Konzept (siehe Abbildung 1) im durchgängigen Engineering (Anwendungsszenario „SP2“ in [5]) schrittweise erstellt und später für das Szenario „Value-based Services“ (Anwendungsszenario „VBS“ in [5]) genutzt werden kann. Das Konzept stellt dabei eine Möglichkeit dar, um Objekte der Verwaltungsschale entsprechend [1] semantisch zu modellieren. Aus Platzgründen kann dies hier nur ausschnittsweise betrachtet werden. Zudem soll kurz darauf eingegangen werden, wie das Funktionsmodell in AutomationML dargestellt werden kann, für eine ausführliche Betrachtung sei auf [21] verwiesen.

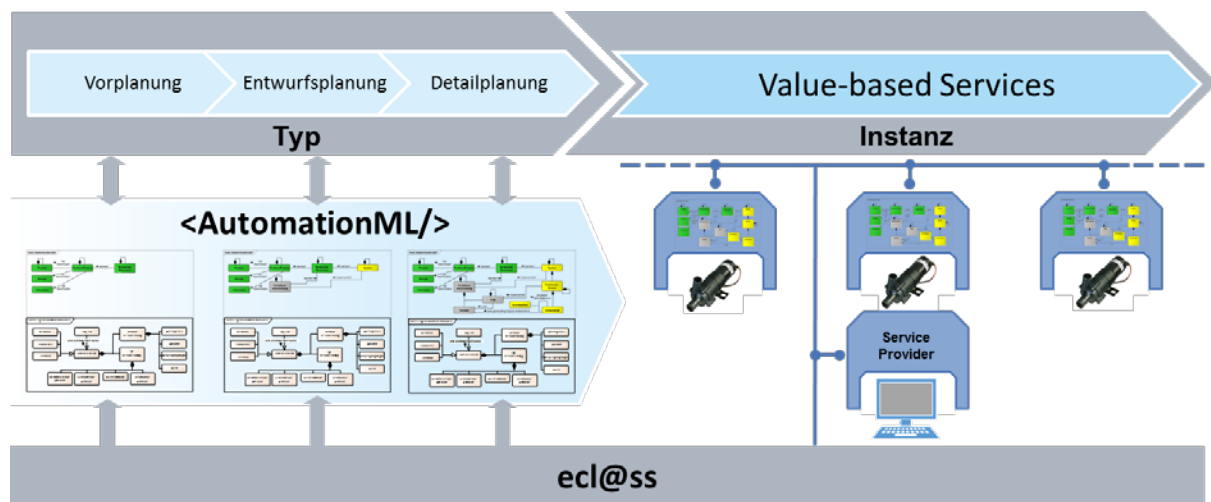


Abbildung 4: Nutzung des Informationsmodells aus SemAnz4.0 im Engineering Workflow

Als Beispielanlage dient hier ein Misch-Modul einer modularen verfahrenstechnischen Anlage<sup>2</sup> (siehe auch [29] Bild 2). In einer frühen Phase des Engineerings des Moduls wird das Funktionsmodell unter Nutzung der Formalisierten Prozessbeschreibung aufgestellt (Abbildung 2 und Abbildung 4, grüne Anteile des Modells). Es enthält in einem ersten Schritt die Daten des Verfahrensfließbildes, kann stufenweise erweitert werden und damit zur Erstellung des R&I-Fließbildes (Abbildung 5, Links) herangezogen werden [30]. Das Funktionsmodell (Abbildung 5, Mitte) dient, wie das R&I-Fließbild [31], mehreren Gewerken im Engineering-Prozess als gemeinsame Kommunikationsgrundlage in der Vor- sowie Entwurfsplanung, kann aber in seinem Informationsmodell darüber hinaus detaillierte Informationen über die Anforderungen an die Geräte und über die Produkte und Zwischenprodukte des Prozesses enthalten.

<sup>2</sup> Siehe: <http://aut.hsu-hh.de/modva>



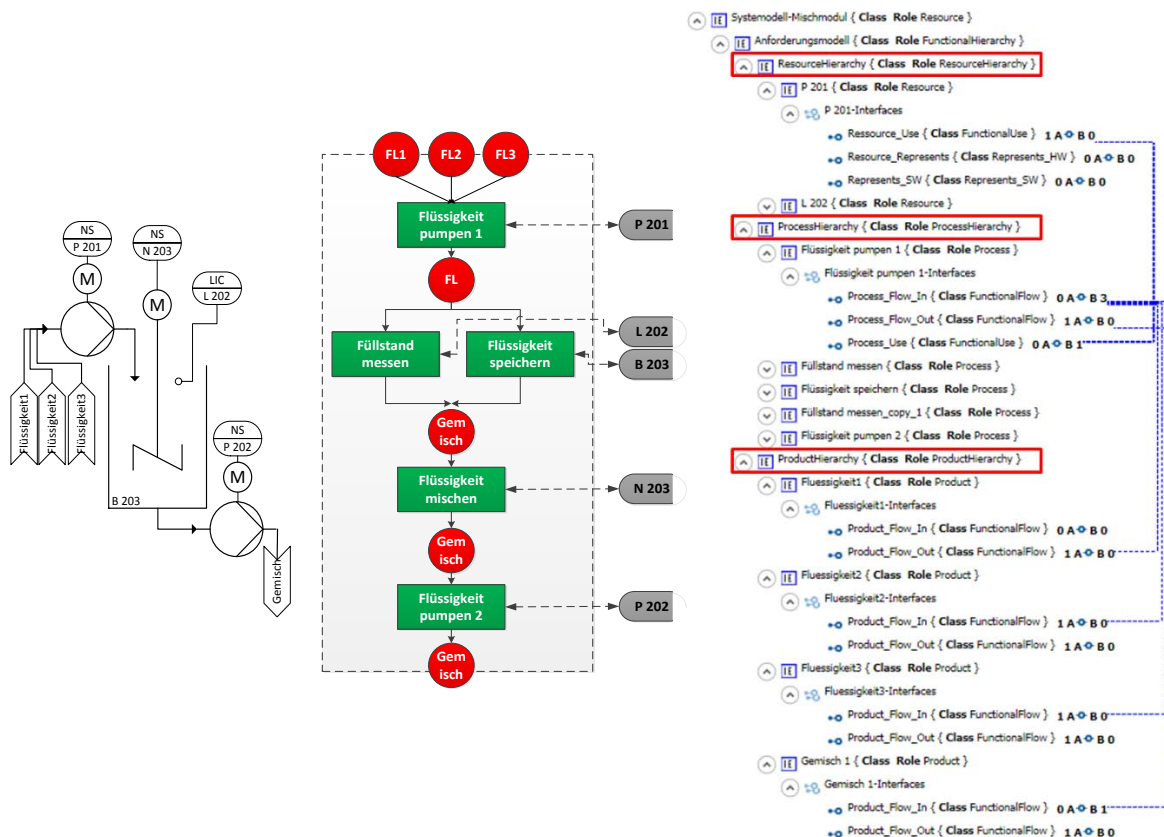


Abbildung 5: R&I Fließbild, Funktionsmodell (als FPB und in AML) des Mischmoduls

Zudem lässt es sich per EDV zwischen den Werkzeugen austauschen. Dazu wurde es in Anlehnung an [32] in AutomationML abgebildet (Abbildung 5, rechts). Die einzelnen Elemente der FPB (Produkte, Prozesse und Ressourcen) sind dabei jeweils in einer eigenen Hierarchie angeordnet (Abbildung 5, rechts, rote Markierungen). Aus den Technischen Ressourcen des Funktionsmodells lässt sich das Grundgerüst des Strukturmodells (Abbildung 4, gelbe Anteile des Modells) aufstellen. Im Strukturmodell lassen sich alle Strukturdaten des R&I-Fließbildes EDV-technisch auswertbar abbilden, dies kann ebenfalls mit Hilfe von AutomationML realisiert werden [33]. Das Verhaltensmodell wächst zusammen mit dem Strukturmodell auf (Abbildung 4, graue Anteile des Modells) und enthält Details zu den Abläufen in der Anlage oder kann Informationen über Verriegelungslogiken abbilden und wird somit insbesondere in der Detailplanung genutzt. Ist das Engineering für das Modul abgeschlossen, das Modul aufgebaut und verkauft, so kann das vollständige Informationsmodell dem Käufer zusammen mit dem Modul übergeben werden. Dieser kann dieses Informationsmodell dann an seinen speziellen Einsatzfall anpassen, zum Beispiel Produktinformationen eintragen. Dieses aktualisierte Informationsmodell kann dann zum Beispiel genutzt werden, um es zur Realisierung von „Value-based Services“ (siehe „VBS“ in [5]) einzusetzen.

Als einfaches Beispiel für einen „Value-based Service“ soll hier der Service der Überwachung einer Pumpe des Moduls gewählt werden. Dafür sind Zustandsinformationen und Kontextinformationen über die Pumpe des Moduls notwendig [6]. Dazu werden z.B. Informationen über das Produkt (Viskosität, Temperatur,...) und über ein- und ausgehende Energien (Stromverbrauch der Pumpe) aus dem Funktionsmodell des Moduls ausgewertet. Zusätzlich lassen sich Informationen über die Umgebung der Pumpe aus dem Strukturmodell ziehen. Zum Beispiel kann im vorliegenden Fall der Füllstandssensor (L 202) im nachfolgenden Tank (B 203) Aufschluss über Förderleistung der Pumpe (P 201) geben (siehe Abbildung 5). Dazu wird in einem definierten Service der Pumpe die Zeit gemessen, die die Pumpe benötigt, um den geforderten Füllstand zu erreichen. Diese Zeitmessungen werden über einen langen Zeitraum durchgeführt, und der Zeitunterschied wird ausgewertet. Zusätzlich wird der Stromverbrauch der Pumpe von der Steuerung zur Verfügung gestellt. Die Prognose wird mit Hilfe eines Verschleißmodells des Herstellers oder des Modulanbieters durchgeführt. Somit kann dem Betreiber eine Ausfallwahrscheinlichkeit für seine Pumpe mitgeteilt werden. Das Ergebnis des Services wird umso besser, je mehr Pumpen gleichen Typs vom Service-Anbieter überwacht werden, möglichst auch unter unterschiedlichen Umgebungsbedingungen. Dies allerdings bedeutet, dass servicerelevante Eigenschaften (Druck, Temperatur, Drehzahl,...) ebenfalls eine gemeinsame semantische Basis aufweisen müssen.

Jedes Objekt im Systemmodell kann, wie in [34] angeführt, mit Hilfe von Merkmalen beschrieben werden. Auf diese Weise kann sichergestellt werden, dass gleiche Objekteigenschaften auch in unterschiedlichen Modellen als gleich erkannt werden können. Dies ist mit einer Abwandlung zu [34] auch für Datenelemente im Allgemeinen möglich und wird nachfolgend erläutert. Ein Datenelement kann, je nach definierendem Standard, eine unterschiedliche Anzahl an beschreibenden Attributen besitzen, muss jedoch stets eine Struktur- sowie Inhaltsreferenz enthalten. Die Strukturreferenz stellt dar, nach welchem Standard die Bedeutung der Attribute des Datenelements zu interpretieren ist. Die Inhaltsreferenz stellt dar, nach welchem Standard die Werte der Attribute zu interpretieren sind. Abbildung 6 zeigt beispielhaft die Modellierung von Systemeigenschaften mit AutomationML anhand des eCI@ss-Merkmals „Pumpendrehzahl“, als Hierarchie links und als Tabelle rechts. Für das Datenelement (im Beispiel ein Merkmal) wird eine Hierarchie von AML-Attributen angelegt. Um das Merkmal nun hinreichend zu beschreiben, sind zwei Arten von Referenzen notwendig. Mit Hilfe der einmalig anzugebenden Strukturreferenz wird eindeutig definiert, welche Bedeutung die genutzten Attribute besitzen. Dies erfolgt über das Feld „RefSemantic“ des Attributs „Typbeschreibung“. Im Beispiel hat das Attribut „Typbeschreibung“ im Feld „RefSemantic“ den Wert „DIN EN 61360-1:2004-12“, welcher auf „DIN 61360“ [22] verweist.

The screenshot shows the AutomationML interface for modeling the data element "Pumpendrehzahl". On the left, a tree view under "Attribute" shows "Typbeschreibung" selected. The main area displays two "Attribute-Table" windows. The top window shows the "Typbeschreibung" table with columns "Name", "Value", and "RefSemantic". The bottom window shows the "Instanzbeschreibung" table with columns "Name", "Value", and "RefSemantic". The "Instanzbeschreibung" table contains 15 rows of data, including "P.T.ID", "P.T.Versionsnumme", "P.T.Änderungsnum", "P.T.Name", "P.T.Definition", "P.T.Quellendokume für Definition des DET", "P.T.Datentyp", "P.T.Maßeinheit", "P.T.Werteliste", "P.I.Qualität", "P.I.Zeitstempel", "P.I.Aussageziel", "P.I.Ausprägungslog", and "P.I.Wert". Each row has a "RefSemantic" field containing a code or URL.

Name	Value	RefSemantic
P.T.ID	02-BAJ155	RefSemantic [1]
P.T.Versionsnumme	9.1	RefSemantic [0]
P.T.Änderungsnum	1.0	RefSemantic [0]
P.T.Name	Pumpendrehzahl	RefSemantic [0]
P.T.Definition	Anzahl der Umdrehungen der Pumpe je Zeiteinheit	RefSemantic [0]
P.T.Quellendokume für Definition des DET		RefSemantic [0]
P.T.Datentyp	INTEGER_MEASURE	RefSemantic [0]
P.T.Maßeinheit	0112/2///62720#UAA843	RefSemantic [1]
P.T.Werteliste	-	RefSemantic [0]
P.I.Qualität	Good	RefSemantic [0]
P.I.Zeitstempel	08.02.2017 18:41:40,933	RefSemantic [0]
P.I.Aussageziel	Assurance	RefSemantic [0]
P.I.Ausprägungslog	Equality	RefSemantic [0]
P.I.Wert	10000	RefSemantic [0]

Abbildung 6: Beispiel für die Modellierung des Datenelements "Pumpendrehzahl" in AutomationML

Die Attributwerte der Typ- und Instanzbeschreibung werden wie folgt durch die Inhaltsreferenz beschrieben. Das Feld „RefSemantic“ enthält entweder eine URL oder eine ID eines definierenden Standards (siehe auch (ISO 29000)). Der Attributwert hingegen weist wenn möglich einen Code auf, der in einem Standard auf eine definierte Semantik verweist. Im Beispiel verweist das Attribut „ID“ über das Feld „RefSemantic“ mit Hilfe einer URL auf ecl@ss. Für das Attribut Maßeinheit hingegen zeigt der Attributwert „0112/2///62720#UAA843“ an und das Feld „RefSemantic“ muss nun auf den Standard „IEC61987“ verweisen, in welchem der Code auf die Maßeinheit „1/min“ führt. Wenn nun die Instanzbeschreibung „Wert“ der Pumpendrehzahl zweier Pumpen verglichen werden soll, muss zunächst über die Strukturreferenz geprüft werden, nach welchem Standard die Attribute strukturiert sind. Über

einen Abgleich der beiden „RefSemantic“ der Pumpen für das Attribut „Maßeinheit“ sowie dessen jeweiligen Attributwert kann dann die Interpretation der Maßeinheit in Zusammenhang mit der Instanzausprägung des Wertes „10000“ gesetzt werden. Durch Verwendung des Eigenschaftsmetamodells und der darin vorgesehenen Beschreibung ist dieses Merkmal auch außerhalb des Modells semantisch und syntaktisch eindeutig definiert und kann zur Überwachung der Pumpe und der Erstellung der Ausfallprognose von einem externen Service genutzt werden.

## **7. Zusammenfassung**

Mit Hilfe des vorgestellten SemAnz4.0-Konzeptes ist es möglich, für Systeme, die auf eine Kollaboration bzw. Kooperation mit anderen Systemen angewiesen sind, die Semantik der ausgetauschten Informationen zu modellieren. Hierzu wurde einerseits ein Systemmetamodell vorgestellt, welches Objekte eines mechatronischen Systems klassifiziert, sowie Regeln zum Aufbau von Systemmodellen definiert. Andererseits wurde zur Beschreibung der Eigenschaften der Systemobjekte ein Eigenschaftsmetamodell zur Beschreibung von Datenelementen vorgestellt. Durch die Nutzung einer modellextern standardisierten Bibliothek von Datenelementen, wie z.B. eCI@ss, können dann die Eigenschaften der Systemobjekte als entsprechende Datenelemente beschrieben werden. Systemmodell und Modellierung von Systemobjekteigenschaften wurden exemplarisch am Beispiel eines Mischmoduls und des darin enthaltenen mechatronischen Objekts „Pumpe“ erläutert.

## **8. Danksagung**

Der Inhalt dieses Beitrags wurde im Rahmen des Projekts “Vorhandene Standards als semantische Basis für die Anwendung von Industrie 4.0 (SemAnz40)” erarbeitet. Dieses Projekt wird vom Bundesministerium für Wirtschaft und Energie gefördert, unter dem Förderkennzeichen 01FS15012, aufgrund eines Beschlusses des Deutschen Bundestags. Die Verantwortung für die Inhalte tragen allein die Autoren.

## **9. Quellen**

- [1] DIN SPEC 91345, 04.2016: Referenzarchitekturmodell Industrie 4.0 (RAMI4.0).
- [2] Kagermann, H.; Anderl, R.; Gausemeier, J.; Schuh, G.; Wahlster, W. (Hg.) (2016): Industrie 4.0 im globalen Kontext. Deutsche Akademie der Technikwissenschaften. München: Herbert Utz Verlag GmbH (acatech STUDIE).
- [3] Plattform Industrie 4.0 (2016): Forschungsagenda Industrie 4.0 – Aktualisierung des Forschungsbedarfs. Hg. v. Bundesministerium für Wirtschaft und Energie. Online verfügbar unter <http://www.plattform->

i40.de/I40/Redaktion/DE/Downloads/Publikation/forschungsagenda-i40.pdf?\_\_blob=publicationFile&v=4, zuletzt geprüft am 05.04.2017.

- [4] Fay, A.; Diedrich, C.; Thron, M.; Scholz, A.; Puntel Schmidt, P. (2015): Wie bekommt Industrie 4.0 Bedeutung? Normen und Standards als semantische Basis. In: atp - edition 57 (7-8), S. 30–41.
- [5] Plattform Industrie 4.0 (2016): Fortschreibung der Anwendungsszenarien der Plattform Industrie 4.0. Hg. v. Bundesministerium für Wirtschaft und Energie. Online verfügbar unter [http://www.plattform-i40.de/I40/Redaktion/DE/Downloads/Publikation/fortschreibung-anwendungsszenarien.pdf?\\_\\_blob=publicationFile&v=4](http://www.plattform-i40.de/I40/Redaktion/DE/Downloads/Publikation/fortschreibung-anwendungsszenarien.pdf?__blob=publicationFile&v=4), zuletzt geprüft am 05.04.2017.
- [6] VDI/VDE 2651-1, 09.2009: Plant Asset Management (PAM) in der Prozessindustrie.
- [7] Hildebrandt, C.; Hoang, X.; Scholz, A.; Fay, A.; Schreiber, A.; Graeser, O. (2016): Modellierung von Aufträgen und Produktionsressourcen in flexibilisierten Produktionsumgebungen. In: VDI - Verein Deutscher Ingenieure e.V. (Hg.): AUTOMATION. Automationskongress. Baden-Baden. VDI - Verein Deutscher Ingenieure e.V.
- [8] Fay, A.; Schröck, S.; Drumm, O.; Löwen, U.; Schertl, A.; Eckhardt, R. et al. (2016): Durchgängigkeit in Wertschöpfungsketten von Industrie 4.0. In: VDI - Verein Deutscher Ingenieure e.V. (Hg.): AUTOMATION. Automationskongress. Baden-Baden. VDI - Verein Deutscher Ingenieure e.V.
- [9] Hadlich, T. (2015): Verwendung von Merkmalen im Engineering von Systemen. Dissertation. Otto-von-Guericke-Universität Magdeburg, Magdeburg. Fakultät für Elektrotechnik und Informationstechnik.
- [10] Pohl, K.; Hönniger, H.; Achatz, R.; Broy, M. (Hg.) (2012): Model-Based Engineering of Embedded Systems. Berlin, Heidelberg: Springer Berlin Heidelberg.
- [11] DIN SPEC 40912, 11.2014: Kernmodelle – Beschreibung und Beispiele.
- [12] VDI 2206, 06.2004: Entwicklungsmethodik für mechatronische Systeme.
- [13] VDI 2221, 05.1993: Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte.
- [14] VDI/VDE 3682-1, 05.2015: Formalisierte Prozessbeschreibungen.
- [15] Ponn, J.; Lindemann, U. (2008): Konzeptentwicklung und Gestaltung technischer Produkte. Optimierte Produkte - systematisch von Anforderungen zu Konzepten. Berlin, Heidelberg: Springer (VDI-Buch).
- [16] Zawisza, J.; Hell, K.; Röpke, H.; Lüder, A.; Schmidt, N. (2016): Generische Strukturierung von Produktionssystemen der Fertigungsindustrie. In: VDI - Verein Deutscher Ingenieure e.V. (Hg.): AUTOMATION. Automationskongress. Baden-Baden. VDI - Verein Deutscher Ingenieure e.V.
- [17] DIN EN 61499-1, 09.2014: Funktionsbausteine für industrielle Leitsysteme.
- [18] DIN EN 61131-3, 06.2014: Speicherprogrammierbare Steuerungen - Teil 3: Programmiersprachen.
- [19] DIN EN 62714-1, 06.2015: Datenaustauschformat für Planungsdaten industrieller Automatisierungssysteme.
- [20] McGuinness, D.; van Harmelen, F. (2004): OWL Web Ontology Language. Overview. W3C - World Wide Web Consortium. Online verfügbar unter <https://www.w3.org/TR/owl-features/>, zuletzt geprüft am 05.04.2017.
- [21] Scholz, A.; Hildebrandt, C.; Wentzien, C.; Mathes, T.; Fay, A. (2017): Modellierung von Fertigungsfunktionen bringt Industrie 4.0 in Bestandsanlagen. In: VDI - Verein Deutscher Ingenieure e.V. (Hg.): AUTOMATION. Automationskongress. Baden-Baden. VDI - Verein Deutscher Ingenieure e.V.
- [22] DIN EN 61360-1, 12.2004: Genormte Datenelementtypen mit Klassifikationsschema für elektrische Bauteile.
- [23] Epple, U. (2011): Merkmale als Grundlage der Interoperabilität technischer Systeme. In: at - Automatisierungstechnik 59 (7), S. 440–450. DOI: 10.1524/auto.2011.0939.
- [24] Diedrich, C.; Hadlich, T.; Thron, M.: Semantik durch Merkmale für I40. Beitrag in B. Vogel-Heuser et al. (Hrsg.), Handbuch Industrie 4.0, Springer NachschlageWissen, DOI 10.1007/978-3-662-45537-1\_63-1. Online ISBN 978-3-662-45537-1.

- [25] Diedrich, C.; Bock, J.; Gössling, A.; Hänisch, R.; Kraft, A.; Pethig, F.; Niggemann, O.; Reich, J.; Vollmar, F.; Wende, J.: Interaktionsmodell für Industrie 4.0 Komponenten. – EKA - Entwurf komplexer Automatisierungssysteme", 24.-25. Mai 2016, Magdeburg, Tagungs-CD.
- [26] VDI/VDE 3689-1, 06.2004: Profibus-Profil – Drehzahlveränderbare Antriebe
- [27] VDI/VDE 3689-2, 12.1995: Profibus-Profil – Profil für Niederspannung-Schaltgeräte
- [28] ISO/TS 29002-5, 02.2009: Industrial automation systems and integration
- [29] Holm, T.; Hempen, U.; Ladiges, J.; Fay, A.; Wassilew, S.; Altmann, P.; Urbas, L. (2016): DIMA im realen Einsatz. In: VDI - Verein Deutscher Ingenieure e.V. (Hg.): AUTOMATION. Automationskongress. Baden-Baden. VDI - Verein Deutscher Ingenieure e.V.
- [30] Ulrich, A.; Güttel, K.; Fay, A. (2009): Durchgängige Prozesssicht in unterschiedlichen Domänen - Methoden und Werkzeug zum Einsatz der formalisierten Prozessbeschreibung - Universal View on Processes in Different Domains - Methods and Tool for Formalised Process Descriptions - In: at - Automatisierungstechnik, Oldenbourg-Verlag, München, Heft 2/2009, pp. 80-92.
- [31] Behnen, D.; Mersch, H.; Quix, C.; Schmitz, D.; Zhang, M.; Fayzullin, K.; Brecher, C.; Epple, U.; Jarke M. (2010): Gemeinsamkeiten und Unterschiede in der Modellierung von prozesstechnischen und diskreten Produktionsanlagen. In: Entwurf komplexer Automatisierungssysteme (EKA). Magdeburg, 2010.
- [32] Jäger, T.; Christiansen, L.; Strube, M.; Fay A.: Durchgängige Werkzeugunterstützung von der Anforderungserhebung mittels formalisierter Prozessbeschreibung und AutomationML. In: Entwurf komplexer Automatisierungssysteme (EKA). Magdeburg, 2012.
- [33] DIN EN 62424, 2008: Festlegung für die Darstellung von Aufgaben der Prozessleittechnik in Fließbildern und für den Datenaustausch zwischen EDV-Werkzeugen zur Fließbilderstellung und CAE-Systemen
- [34] Graeser, O.; Hundt, L.; John, M.; Lobermeier, G.; Lüder, A.; Mühlens, S. et al. (2015): AutomationML and eCI@ss integration. Hg. v. AutomationML e.V. c/o IAF. Online verfügbar unter: [https://www.automationml.org/o.red/uploads/dateien/1448438009-20151030\\_WP\\_AutomationML\\_and\\_eClass\\_integration\\_v1.0\\_neu.pdf](https://www.automationml.org/o.red/uploads/dateien/1448438009-20151030_WP_AutomationML_and_eClass_integration_v1.0_neu.pdf), zuletzt geprüft am 05.04.2017.