

Sprache für I4.0-Komponenten - Semantik der Interaktionen von I4.0-Komponenten

Christian Diedrich, ifak - Institut für Automation und Kommunikation e. V., Magdeburg;
Alexander Belyaev, **Tizian Schröder**, Otto-von-Guericke-Universität Magdeburg;
Torben Deppe, RWTH-Aachen, Aachen
Martin Hankel, Bosch Rexroth GMBH Stuttgart,
Daniel Nehls, TU Berlin, Berlin
Florian Pethig, Fraunhofer IOSB-INA, Institutsteil für industrielle Automation, Lemgo;
Johannes Reich, SAP AG Walldorf,
Jens Vialkowitsch, Bosch GmbH, Stuttgart
Friedrich Vollmar, Konsultant, Frankfurt
Jürgen Bock, KUKA Deutschland GmbH, Augsburg
Alexander Willner, Fraunhofer Fokuz, Berlin

Kurzfassung

Verwaltungsschalen bilden zusammen mit den Assets der digitalen Fabrik I4.0-Komponenten. Die Interaktionen zwischen den Verwaltungsschalen der Industrie 4.0-Komponenten bilden das I4.0-System zur Umsetzung der Wertschöpfungsketten. In mehreren I4.0-Szenarien sind Verhandlungen oder Einigungsprozesse über technische Machbarkeit, organisatorische oder preisliche Vereinbarungen von den I4.0-Komponenten abzuwickeln. Diese müssen in einer interoperablen Art und Weise ausgeführt werden. In diesem Beitrag wird das Konzept der Interaktionssemantik für I4.0-Komponenten vorgestellt. Es entsteht im Fachausschuss 7.20 „Semantik und Interaktion von I4.0-Komponenten“ der VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik (GMA) in Zusammenarbeit mit Juristen der AG4 der Plattform I4.0, die sich mit den entsprechenden Fragestellungen beschäftigen.

Stichwörter: Industrie 4.0, I4.0-Komponente, Verwaltungsschale, Interaktionsmodell

1 Einleitung

Basis für dynamische, selbstorganisierende, selbstoptimierende und unternehmensübergreifende Wertschöpfungsnetzwerke ist die Verfügbarkeit aller relevanten Informationen zu einem geforderten Zeitpunkt. Einmal entstandene Informationen sollen da verfügbar sein, wo sie gebraucht werden. Die Voraussetzung dafür ist eine hinreichende digitale Abbildung der physischen Welt in die Informationswelt und die Möglichkeit, die Informationen interoperabel auszutauschen. Ein I4.0-System besteht aus I4.0-Komponenten. I4.0-Komponenten wiederum bestehen jeweils aus einem Asset und seiner digitalen Repräsentation in Form einer

sogenannten Verwaltungsschale. Die relevanten Daten von Assets, etwa die formale Beschreibung von Eigenschaften, Konfigurationsparametern, Zuständen, Prozessfähigkeiten usw. sollen hinreichend in der Informationswelt repräsentiert werden. Dazu werden diese in ihren Verwaltungsschalen in der Form von Merkmalen abgelegt und sollen für die anderen I4.0-Komponenten nutzbar sein.

Zur Durchführung von I4.0-Szenarien interagieren die I4.0-Komponenten miteinander. Diese Interaktionen können horizontal sein, d.h. zwischen Komponenten derselben Betriebs- und Fabrikebene, oder vertikal, d.h. von der Produkt- und Sensor-Ebene bis hin zur Geschäftsebene innerhalb oder sogar über die Grenzen eines Unternehmens hinaus (Abbildung 1). Die vertikale Interaktion repräsentiert Hierarchien, in denen die Informationen bereitstellenden Interaktionspartner als Server agieren und die anfordernden Interaktionspartner die Clients sind.

Ein Beispiel von vertikalen Interaktionen ist der Informationsaustausch zwischen Produktionsressourcen und übergeordneten IT-Systemen. MES und ERP erfüllen die Aufgaben der Überwachung, Planung, Dokumentation und Kontrolle des Produktionsprozesses und die Interaktionen mit Ressourcen erfolgen in der Form von Befehlen und Informationsaufrufen. Eine Verwaltungsschale fungiert in diesem Szenario als eine standardisierte Schnittstelle von Assets, die die standardisierten Informationsmodelle enthält und einen Zugang auf die Parameter ermöglicht. Das Verhalten von zugegriffenen I4.0-Komponenten ist aus Sicht von übergeordneten Systemen eindeutig deterministisch.

Anders ist es bei den Szenarien, wo eine gewisse Autonomie von den Interaktionspartnern, d.h. Produktionssystemen oder ihren Komponenten gefordert wird, z.B. dynamische Optimierung von Produktionsauslastung, Vermeidung von Stillstandzeiten durch Maschinenausfälle, auftragsgesteuerte Produktion, sowie die dynamische Orchestrierung von Produktionsressourcen für die kostenoptimale Fertigung in der Losgröße Eins. In diesen Szenarien treffen Interaktionspartner aufeinander, die beide gleichwertig interaktions-initiierend agieren. Deshalb wird diese auch als horizontale Interaktion bezeichnet. In dieser Interaktion wirken beide Partner als „Peer“ weshalb sie als Peer-to-Peer Interaktion bezeichnet wird. In Abbildung 1 sind beide Interaktionsformen dargestellt.

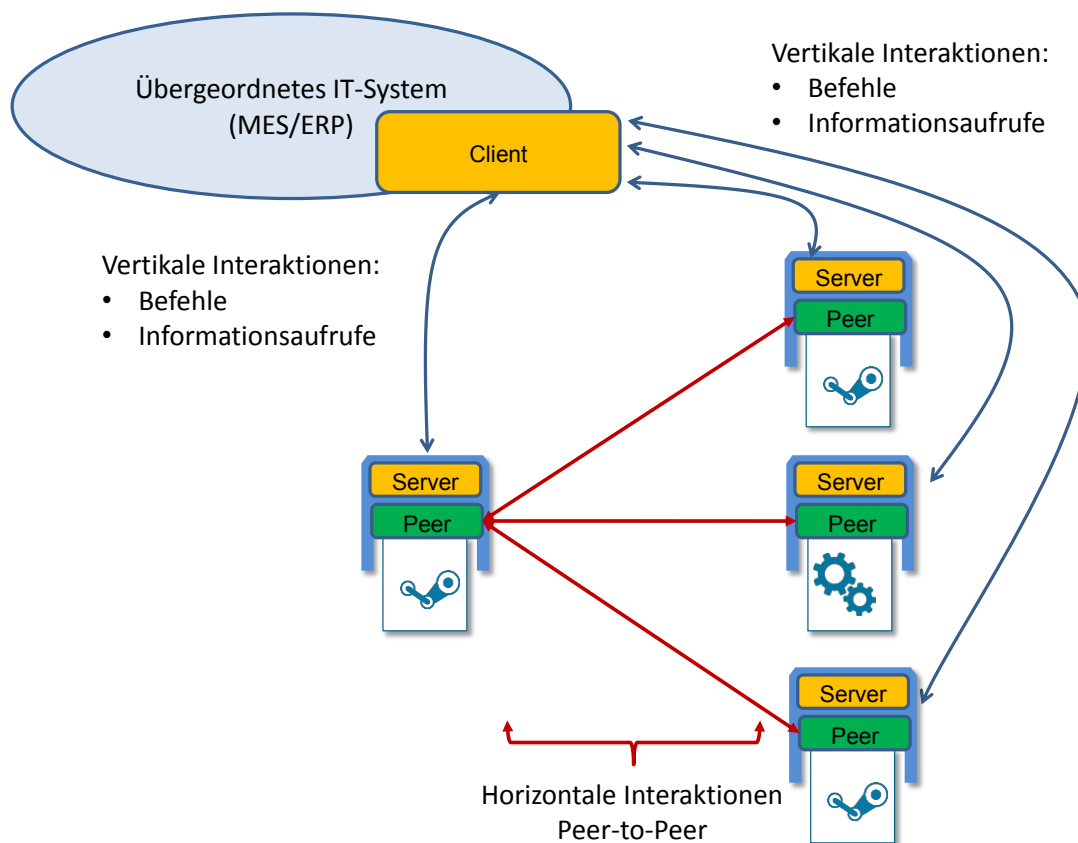


Abbildung 1: Horizontale und vertikale Interaktionen in einem I4.0-System

Das Interaktionsmodell, das diesem Dokument zugrunde liegt, legt das folgende einfache Schema zur Klassifikation von Komponenteninteraktionen nahe. Es basiert auf den beiden Dimensionen

1. Informationsfluss: unidirektional versus bidirektional
2. Informationsverarbeitung: zustandsbehaftet versus zustandslos, deterministisch versus nicht-deterministisch und synchron versus asynchron

Die drei auf die Informationsverarbeitung bezogenen Unterdimensionen wurden ausgewählt, weil sie unmittelbaren Einfluss auf die geeignete Syntax der Interfaces haben.

Interaktionen mit unidirektionalem Informationsfluss sind:

- Deterministisch: Pipes
- Nicht-deterministisch: Anonyme Observation, Publish-Subscribe

Interaktionen mit bidirektionalem Informationsfluss sind:

- Horizontale Interaktionen (Peer-to-Peer in Abbildung 1): Hier verhalten sich die Interaktionspartner bezogen auf die 3 semantischen Unterdimension symmetrisch, nämlich zustandsbehaftet, nichtdeterministisch und asynchron. Diese Interaktionen werden durch multi- oder bilaterale Interfaces von Protokollrollen beschrieben. Die

Kenntnis aller Rollen der Interaktionspartner ist notwendig, um wichtige Eigenschaften der Interaktion wie etwa Vollständigkeit, Konsistenz, die Freiheit von Deadlocks, Livelocks oder Starvation, zu garantieren.

- Vertikale Interaktionen (Client/Server (Abbildung 1): Hier verhalten sich die Interaktionspartner bezogen auf die 3 semantischen Unterdimensionen asymmetrisch. Diese Interaktionen werden durch asymmetrische Interfaces beschrieben, nämlich top-down Funktionsaufrufe (mit Ausnahmen) und bottom-up Events. Tatsächlich wird im Interface nur die Komponente beschrieben, die die Funktionalität, bzw. die Events zur Verfügung stellt.

Verwaltungsschalen können beide Interaktionsformen anbieten, wobei die Peer-to-Peer-Interaktion dort Anwendung findet, wo eine gewisse Autonomie der I4.0-Komponente vorhanden ist und die Reaktionen auf Anfragen für andere I4.0-Komponenten nicht unbedingt bekannt sein müssen. Dies ist z.B. der Fall, wenn Umgebungsbedingungen (Menschen in einer Roboterzelle, bereits vereinbarte Taskliste ist voll) dazu führen kann, dass bestimmte Aktivitäten nicht ausgeführt werden können (Abbildung 2). Bei der Client/Server Interaktion herrscht die dienstorientierte Kommunikation vor und bei der Peer-to-Peer Interaktion die protokollorientierte Kommunikation. In beiden Interaktionsformen wird auf die Teilmodelle der Verwaltungsschale zugegriffen.

In beiden Fällen tauschen Client und Server wie auch die Peers Nachrichten aus, die von allen Beteiligten eindeutig interpretiert werden müssen. Während bei den Diensten typischerweise jeder einzelne Dienstaufwurf einen festen Ablauf hat (z.B. Lesen und Schreiben von Variablen und Methodenaufrufe), legt bei der Peer-to-Peer-Interaktion ein Protokoll mögliche Nachrichtenabläufe fest. Diese Abläufe sind von dem Kontext und dem Zustand der jeweiligen Partner abhängig.

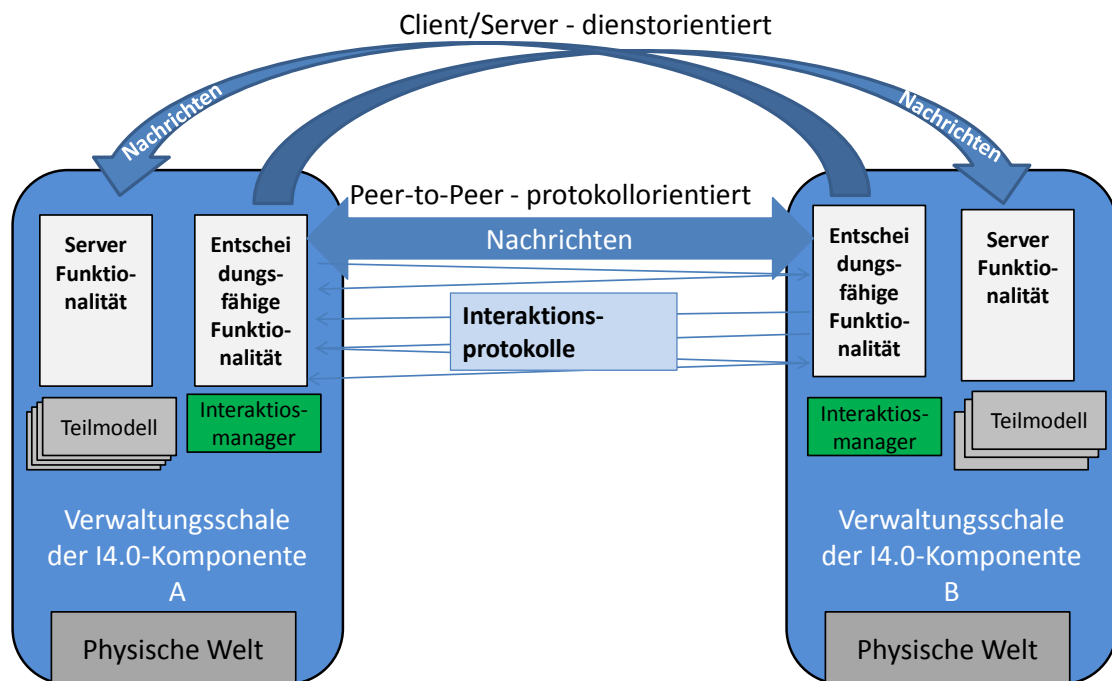


Abbildung 2: Funktionalitätskategorien von Verwaltungsschalen

2 I4.0-Sprache - Übersicht

Die Anforderungen an die I4.0-Sprache sind sehr heterogen, adressieren aber in jedem Fall eine hohe Nutzungsbreite in den verschiedensten Domänen und unterschiedliche Aussagestärken. Es wird offensichtlich, dass es nicht EINE einzige I4.0 Sprache gibt. Vielmehr wird durch Untergliederung der Sprache in einzelne Teilaspekte die notwendige Anpassung an die Anwendung gewährleistet.

Um den sich aus dem I4.0-Konzept ergebenden Anforderungen ([1], [2], [3], [4]) gerecht zu werden, wird die I4.0-Sprache als ein Regelsystem, bestehend aus drei Ebenen betrachtet. Der Begriff I4.0-Sprache wird als übergreifende Bezeichnung für die verschiedenen Aspekte des gegenseitigen Verstehens von I4.0-Komponenten verwendet. Er repräsentiert die Sprache mehr im linguistischen Sinne und nicht die formale Definition für die Anwendung in Softwarewerkzeugen. Es ist eine Konkretisierung des Konzepts, das in [5] veröffentlicht worden ist. Die I4.0-Sprache zwischen I4.0-Komponenten wird untergliedert in (Abbildung 3):

- **Vokabular der Sprache:** Die Merkmale, Merkmallisten, oder andere Annotationsformen von Datenelementen, die in den Nachrichten verwendet werden. → entspricht der Wortsyntax
- **Struktur der Nachrichten:** Struktur, die die Anordnungen der Inhalte und die für deren gegenseitige Einordnung notwendigen Elemente organisiert. → entspricht dem Satzbau
- **Interaktionsprotokolle:** Abläufe in den Dialogen zwischen I4.0-Komponenten, die die zu erfüllenden Aufgaben organisieren. → entspricht dem Dialogablauf

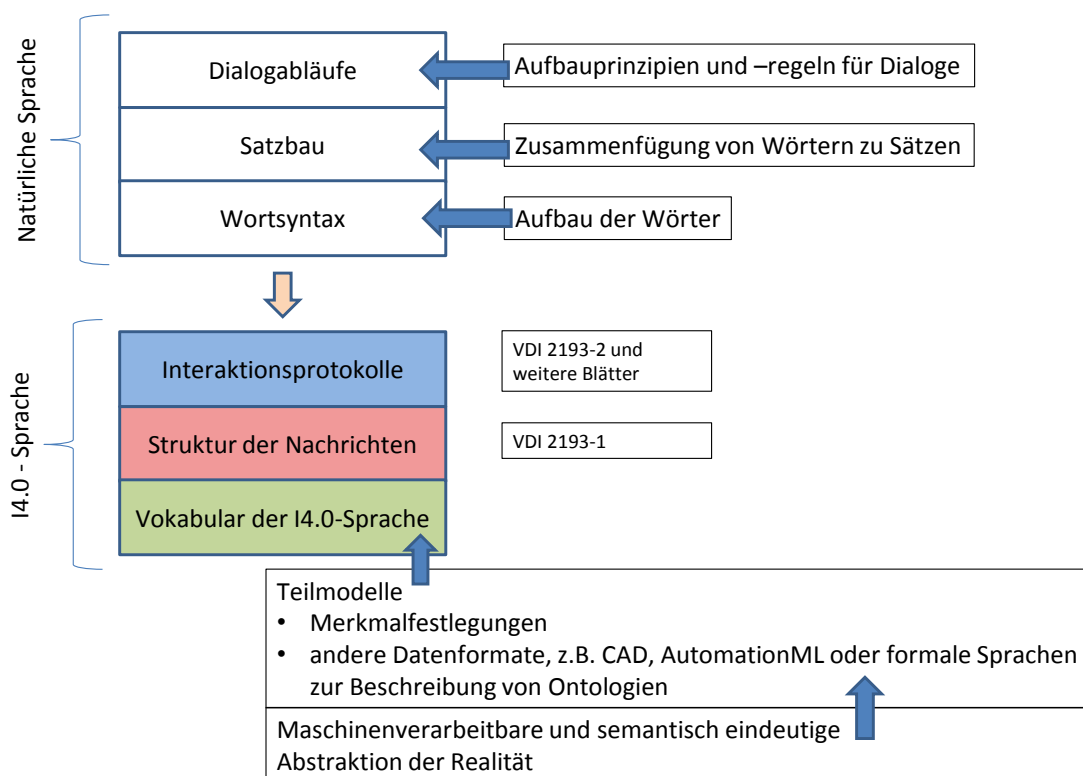


Abbildung 3: Aspekte der I4.0-Sprache

Das Vokabular besteht im Wesentlichen aus Merkmalen, die die Eigenschaften der durch die Teilmodelle beschriebenen Assets beinhalten. Die Merkmale können Parameter, Variablen, Zustands- oder Kataloggrößen sein (Beispiele siehe in Tabelle 1).

Die **Struktur der Nachrichten** besteht aus einem Identifikationsteil und dem inhaltlichen Teil. Die Identifikation adressiert auf der einen Seite die Verwaltungsschale und die Teilmodelle, enthält aber auch die Transferkennungen (z.B. Mein Zeichen, Dein Zeichen wie im professionellen Briefwechsel). Im inhaltlichen Teil (Payload) stehen der Zweck der Nachricht, die Merkmale sowie die Funktionalität (z.B. Bohren, Transportieren), die genutzt werden soll.

Der Zweck der Nachricht definiert beispielsweise, dass eine Anfrage gestellt wird oder eine Änderung vorgenommen werden soll.

Die **Interaktionsprotokolle** beschreiben die Abfolge von Nachrichten einschließlich der Abhängigkeit der Abfolge von den Zuständen der beiden Interaktionspartner.

Eine ausführlichere Beschreibung ist in [9] enthalten.

Tabelle 1: Beispielmerkmale für Datenelementbeschreibung nach IEC 61360

Attribut	Herstellername	Masse	Drehzahl Drehstrommotor
Attribute nach IEC 61360-1 /ISO 13584-42			
ID	0173-1#02-AAO677#002	0173-1#02-BAB576#005	0173-1#02-AAD222#006
Version	V10.0.1	V9.1	V9.1
Name	Herstellername	Gewicht	Drehzahl
Definition	Bezeichnung für eine natürliche oder juristische Person, die für die Auslegung, Herstellung und Verpackung sowie die Etikettierung eines Produkts im Hinblick auf das 'Inverkehrbringen' im eigenen Namen verantwortlich ist	Kraft ausgelöst durch die Masse eines Teiles	Quotient aus Anzahl der Umdrehungen und der zugehörigen Zeitspanne
Kurzname	MAN_NAME	Masse	Drehzahl
Symbol	-	m	ř
SI-Einheit	-	kg	Umdrehung/Minute (min ⁻¹)
Datentyp	String	Float	String
Werteformat	ASCII	Binary	ASCII
Gültiger Wertebereich	5-20 ASCII Zeichen	0 ... 100	0 ... 600 1/min bei 50 Hz 0 ... 720 1/min bei 60 Hz
Wert (Beispiel)	Beispielhersteller	15,4	300

Diese drei Aspekte werden dem RAMI 4.0 [1] in seiner vertikalen Dimension wie folgt zugeordnet (Abbildung 4). Diese Dimension benennt die Aufgabengebiete, die im Rahmen von Industrie 4.0 zu betrachten sind. Die beiden anderen Aspekte des RAMI4.0 (Lebenszyklus und Architektur) werden in (Abbildung 4) nicht gezeigt.

Das Vokabular ist der Informationsschicht zuzuordnen. In dieser Schicht sind die Daten in ein Informationsmodell einzugliedern. Die Struktur der Nachricht ist eine syntaktische Festlegung und ist dem Informationsmodell untergeordnet. Auf die Elemente des Informationsmodells (d.h. Parameter, Variablen, Zustandsgrößen, Methoden aber auch deren Typbeschreibungen) kann zugegriffen werden. Dazu sind die heute üblichen Zugriffe auf Ressourcenmodelle verwendbar (siehe Beispiel Kapitel 3).

Die durch die Nachricht adressierte Funktionalität ist der Funktionsschicht zuzuordnen. Die Interaktionsprotokolle verwenden das Ressourcenmodell der Informationsschicht, die Prozessfähigkeiten der Funktionsschicht und die Entscheidungsalgorithmen der Businessschicht. In der Businessschicht werden technische und ökonomische Entscheidungen mit Hilfe von Algorithmen getroffen. Sie repräsentieren somit geschäftliche Abläufe. Nicht alle I4.0-Komponenten müssen Business-Services bereitstellen. I4.0-Komponenten, die Business-Services bereitstellen, unterstützen beispielsweise Ausschreibungsverfahren oder vertragliche Verhandlungen zur Erbringung von Produktionsdienstleistungen.

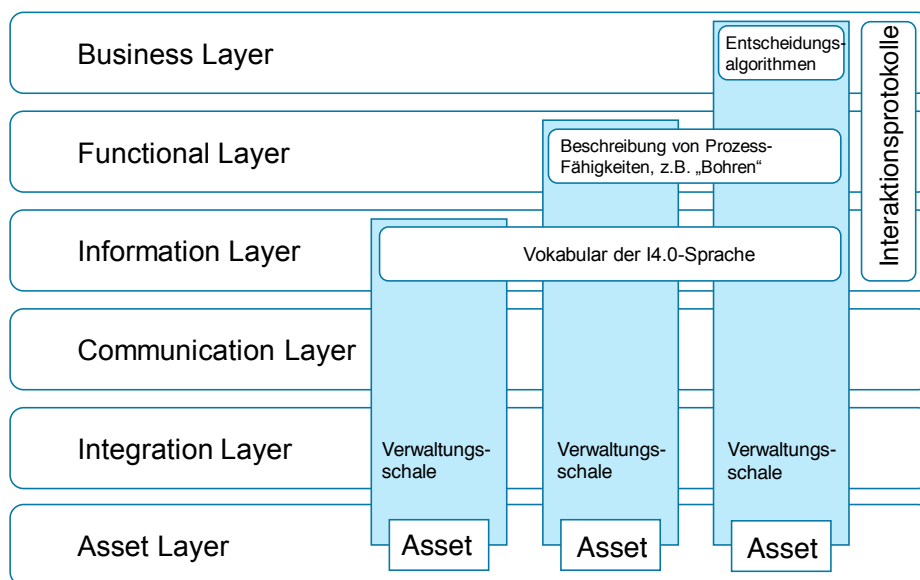


Abbildung 4: Einordnung der Aspekte der I4.0-Sprache in die vertikale Ebene des RAMI4.0 [9]

3 Beispiel Zugriff auf Parameter, Variablen, Zustandsgrößen, Katalogdaten unter Nutzung des Ressourcenmodells

Das Ressourcenmodell bezeichnet ein Modell in dem die Daten und Funktionen einer Komponente adressiert und auf sie mit wenigen Grunddiensten zugegriffen werden kann. Aus der IKT-Welt stammt die technologische Umsetzung des CRUD-Modells (CREATE, RETRIEVE, UPDATE, DELETE Dienste) oder GET und PUT bei http. In der Feldbustechnik stehen oftmals Read und Write Dienste zur Verfügung. Auch OPC UA unterstützt im Wesentlichen das Ressourcenmodell.

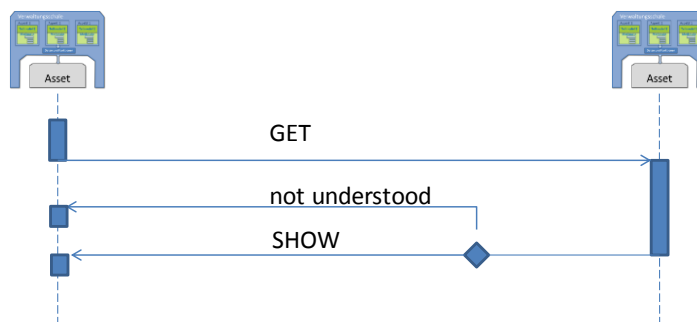


Abbildung 5: Interaktionsmuster "Informationsabfrage"

In der I4.0-Sprache wird für die Zugriffe im Sinne des Ressourcenmodells der Dienstyp auf den Zweck der Nachricht abgebildet. In Abbildung 5 ist „GET“ für eine Leseanfrage und „SHOW“ für die Bereitstellung der angefragten Daten dargestellt. Wenn z.B. nach einem Datum gefragt wird, welches nicht vorhanden ist, kann auch mit „not understood“ geantwortet werden. Im inhaltlichen Teil der Nachrichten sind dann die entsprechenden Daten enthalten. Die Nachrichten könnten dann wie folgt aussehen (beispielhaft in JSON, Tabelle 2):

Tabelle 2: Beispielnachrichten für Informationsabfrage

GET-Nachricht (Informationsaufruf)	SHOW-Nachricht (Bereitstellung von Informationen)	not understood-Nachricht (Die angefragten Datenelemente sind nicht bekannt)
<pre>{ "Message": { "IDENTIFICATION": "xxxzz", "DATENBEREICH": [{ "ZWECK": "GET", "DATENELEMENTE": "0173-1#02-AAO677#002" }] } }</pre>	<pre>{ "Message": { "IDENTIFICATION": "xxxzz", "DATENBEREICH": [{ "ZWECK": "SHOW", "DATENELEMENTE": "Beispielhersteller" }] } }</pre>	<pre>{ "Message": { "IDENTIFICATION": "xxxzz", "DATENBEREICH": [{ "ZWECK": "not understood", "DATENELEMENTE": "" }] } }</pre>

4 Beispiel Ausschreibungsverfahren

Als Voraussetzung für Vertragsverhandlungen zwischen I4.0-Komponenten etablieren die Vertragspartner einen Rahmenvertrag. Beispielhaft wird hier ein Ausschreibungsverfahren dargestellt. Die einzelnen Schritte des Vertragsrechts können in vereinfachter Form wie folgt dargestellt werden (Abbildung 6):

- Aufforderung zur Abgabe eines Angebotes / oder Ausschreibung
- Angebote
- Die Annahme eines Angebots führt zu einem Vertragsabschluss
- Leistungserbringung aus Vertrag

In Abbildung 6 initiiert die I4.0-Komponente B die Ausschreibung. Typischerweise nehmen viele potentielle Auftragnehmer daran teil, was aus Platzgründen in der Abbildung nicht dargestellt worden ist. Hier interagiert nur die I4.0-Komponente A.

Im Laufe des Verhandlungsprozesses gibt es auf beiden Seiten verschiedene maschinell zu treffende Entscheidungsvarianten. Ein wesentlicher Aspekt ist die Delegation des Verhandlungsmandats durch eine juristische Person auf die I4.0-Komponente, da die juristische Person alle Konsequenzen, die sich aus dem Verhandlungsergebnis ergeben, tragen muss. Die I4.0-Komponente hat ein begrenztes Mandat. Wenn es zu Situationen kommt, in dem der Rahmen des Mandats überschritten wurde, muss die I4.0-Komponente die Entscheidungsaufgabe an die delegierende Stelle zurückgeben. Dies muss in dem Regelwerk der I4.0-Komponente hinterlegt sein.

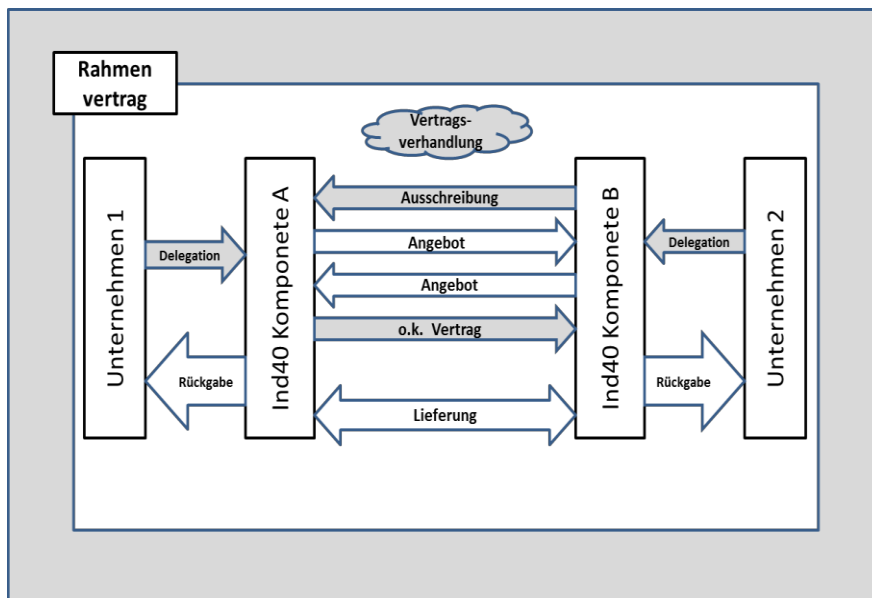


Abbildung 6: Konzept Ausschreibungsverfahren

Dieses Konzept wird wie in Abbildung 7 gezeigt mit den Mitteln der I4.0-Sprache umgesetzt. Zur Darstellung wird ein Sequenzdiagramm verwendet. Auch hier gilt, dass aus Platzgründen nur ein potentieller Auftragnehmer dargestellt worden ist. Die blauen Pfeile symbolisieren die Nachrichten. Die Pfeile sind mit dem Zweck der Nachricht bezeichnet, was durch die roten Pfeile aus der Struktur der Nachricht symbolisiert wird. In der Nachricht sind außerdem die Identifikation der Teilnehmer- und die Konversationsidentifikation enthalten. Der Datenbereich der Nachricht enthält die Datenelemente, die die Inhalte von Ausschreibungen und Angeboten referenzieren.

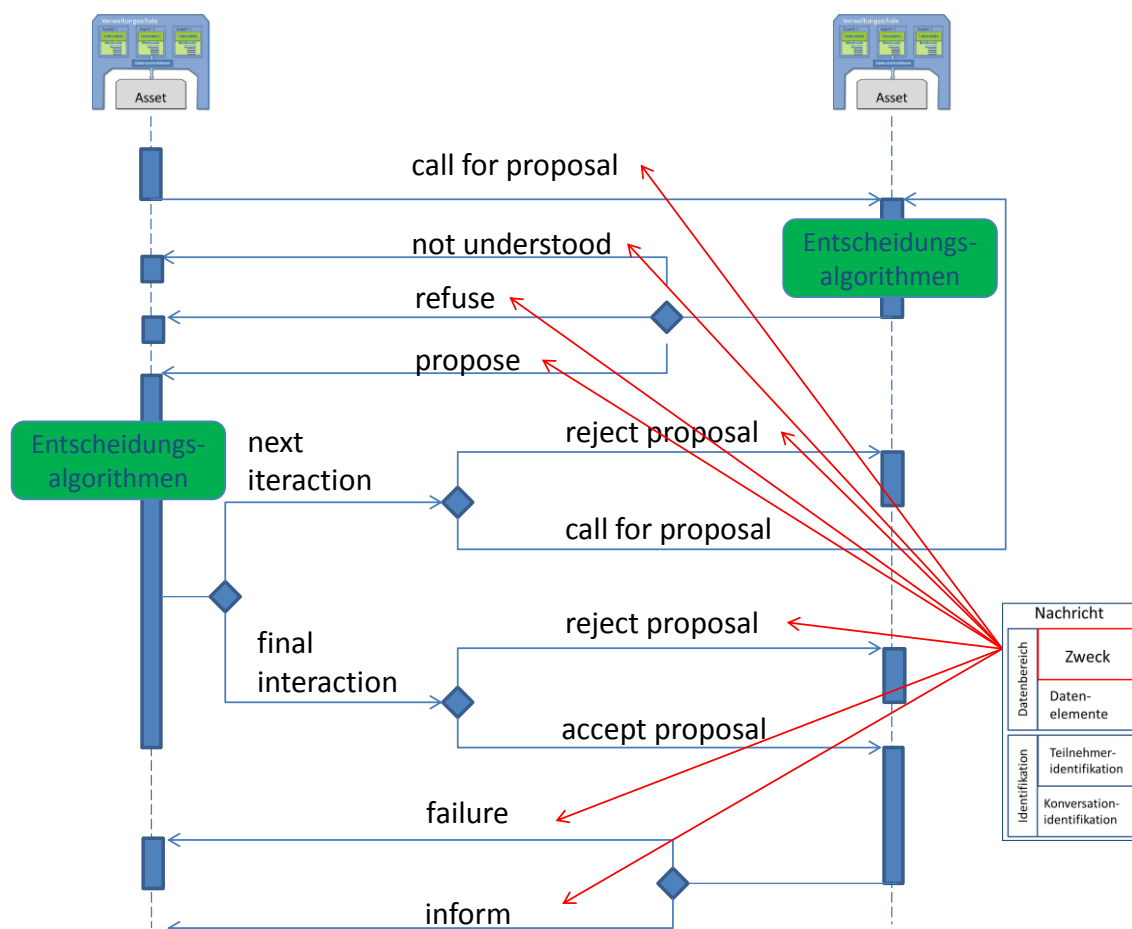


Abbildung 7: Interaktionsprotokoll "Ausschreibungsverfahren"

Das Ausschreibungsverfahren beginnt mit einer Angebotseinholung (Call for proposal). Der potentielle Auftragnehmer muss entscheiden, ob er daran teilnehmen will (freie Kapazitäten, technische Fähigkeiten) und fertigt entweder ein Angebot an (propose) oder lehnt dieses ab

(refuse¹). Dazu hat er interne, für Dritte nicht erkennbare Entscheidungsalgorithmen. Wenn zusätzliche Informationen erforderlich sind, können diese mit „not understood“ angefragt werden. Nachdem bei dem Anfragenden Antworten eingegangen sind (z.B. könnte eine Ausschreibungsdeadline benannt worden sein) hat dieser mit seinen internen, für Andere nicht erkennbare Entscheidungsalgorithmen, entweder neue Anfragen zu starten, z.B. an ausgewählte mit weiteren Details (Call for proposal) oder eine Angebotsbestätigung (accept proposal) zu senden. Ablehnungen (reject proposal) können auch versendet werden. Nach Auftragserteilung kann der Beauftragte über den aktuellen Status informieren (failure, inform). Tabelle 3 zeigt einige Beispiele aus dem Ablauf von Abbildung 7. Im Call for proposal werden Merkmale mit ihren angeforderten Werten vorgegeben (linke Spalte), die von den potentiellen Auftragnehmern mit ihren Zusicherungen (mittlere Spalte) beantwortet werden. Der Ausschreibende kann dann die Vergleiche durchführen und über eine Annahme oder eine Ablehnung von Angeboten entscheiden. Eine angefragte Komponente kann die Anfrage auch ablehnen (rechte Spalte).

Tabelle 3: Beispielnachrichten für Informationsabfrage

Call for proposal-Nachricht (Beschreibung von Anforderungen)	Proposal-Nachricht (Beschreibung von Zusicherungen)	Refuse-Nachricht (Ablehnung eines Angebotes)
<pre>{ "Message": { "IDENTIFICATION": "xxxzz", "DATENBEREICH": [{ "ZWECK": "CfP" }, { "DATENELEMENTE" : [{ "0173-1#02-BAB576#005" : "30" }, { "0173-1#02-AAD222#006" : "500" }] }] } } }</pre>	<pre>{ "Message": { "IDENTIFICATION": "xxxzz", "DATENBEREICH": [{ "ZWECK": "PROPOSE" }, { "DATENELEMENTE" : [{ "0173-1#02-BAB576#005" : "40" }, { "0173-1#02-AAD222#006" : "600" }] }] } } }</pre>	<pre>{ "Message": { "IDENTIFICATION": "xxxzz", "DATENBEREICH": { "ZWECK": "REFUSE" } } }</pre>

¹ Keine Antwort ist auch ein zulässiges Verhalten

5 Zusammenfassung

Durch den Austausch von Daten, die Nutzung von Diensten sowie einer umfassenden Kooperation von Komponenten eines Industrie 4.0-Systems entstehen dynamische, selbstorganisierende, selbstoptimierende und unternehmensübergreifende Wertschöpfungsnetzwerke. Die gewünschten Anwendungsszenarien der Industrie 4.0 können dann verwirklicht werden, wenn alle Daten und Informationen maschinell lesbar sind und Maschinen sich untereinander mittels einer genauen und eindeutigen Interaktionssemantik verstehen. Nach der Auffassung der Autoren erfordert die Verwirklichung dieser Visionen eine Interoperabilität auf einer Abstraktionsebene höher als Schicht 7 des ISO/OSI-Modells. In diesem Artikel wird ein dreischichtiges Regelwerk vorgestellt, das die verschiedenen Aspekte des gegenseitigen Verstehens von I4.0-Komponenten abbildet und die notwendigen Voraussetzungen für eine nahtlose Abwicklung von horizontalen und vertikalen Interaktionen zwischen Automatisierungskomponenten zusammenfasst.

6 Literaturverzeichnis

- [1] DIN SPEC 91345 „Referenzarchitekturmodell Industrie 4.0 (RAMI4.0) April 2016. DIN ICS 03.100.01; 25.040.01; 35.240.50
IEC PAS 63088
- [2] Industrie 4.0 Working Paper: Structure of the Administration Shell - Continuation of the Development of the Reference Model for the Industrie 4.0 Component. Published by Federal Ministry for Economic Affairs and Energy (BMWi) Public Relations 10119 Berlin. April 2016.
- [3] 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. Ergebnispapier Juni 2017.
http://www.plattform-i40.de/I40/Redaktion/DE/Downloads/Publikation/beziehungen-%20i40-komponenten.pdf?__blob=publicationFile&v=7
- [4] Plattform Industrie 4.0 (Hrsg.): Anwendungsszenario trifft Praxis: Auftragsgesteuerte Produktion eines individuellen Fahrradlenkers. April 2017
- [5] Christian Diedrich, Alexander Bieliaiev, Jürgen Bock, Andreas Gössling, Rolf Hänisch, Andreas Kraft, Florian Pethig, Jörg Neidig, Oliver Niggemann, Johannes Reich, Fried-

rich Vollmar, Jens Vialkowitsch, Jörg Wende: Fortschreibung des Interaktionsmodells für Industrie 4.0 Komponenten. Diskussionspapier. Plattform Industrie 4.0. November 2016.

- [6] Bangemann, F., Reich, J., Diedrich, Ch.: A Grammar for the Semantics of Component Interaction of Cyber-Physical Systems. IEEE International Symposium on Industrial Electronics, June 8-10. 2016 SF-006319 proceedings.
- [7] Christian Diedrich, Thomas Hadlich, Mario Thron: Semantik durch Merkmale für I4.0. Beitrag in B. Vogel-Heuser et al. (Hrsg.), Handbuch Industrie 4.0, Springer Nachschlage Wissen, DOI 10.1007/978-3-662-45537-1_63-1. Online ISBN 978-3-662-45537-1.
- [8] André Scholz, Constantin Hildebrandt, Alexander Fay, Tizian Schröder, Thomas Hadlich, Christian Diedrich, Martin Dubovy, Christian Eck, Ralf Wiegand: Semantische Inhalte für Industrie 4.0 – Semantisch interpretierbare Modellierung von technischen Systemen in kollaborativen Umgebungen. atp edition, [S.l.], v. 59, n. 07-08, p. 34-43, sep. 2017. ISSN 2364-3137.
- [9] Christian Diedrich, Jens Vialkowitsch, Jürgen Bock, Torben Deppe, Otto Schell, Alexander Willner, Friedrich Vollmar, Thomas Schulz, Florian Pethig, Jörg Neidig, Thomas Usländer, Johannes Reich, Daniel Nehls, Matthias Lieske, Alexander Belyaev: I4.0-Sprache – Vokabular, Nachrichtenstruktur und semantische Interaktionsprotokolle der I4.0-Sprache. Plattform I4.0, Herausgeber: Bundesministerium für Wirtschaft und Energie (BMWi) Öffentlichkeitsarbeit 11019 Berlin April 2018.