

Standards für Industrie 4.0

Com4.0-Basic Basic Models of Communication

White Paper Version 1.0

Prof. Dr.-Ing. Ulrich Epple
01.12.2016

Lehrstuhl für Prozessleittechnik
Prof. Dr.-Ing. Ulrich Epple
RWTH Aachen
D-52064 Aachen, Deutschland
Telefon +49 241 80 94339
Fax +49 241 80 92238
www.plt.rwth-aachen.de

1 Inhaltsverzeichnis

1	INHALTSVERZEICHNIS	1
2	ÜBERSICHT COM4.0-BASIC.....	2
2.1	ÜBERTRAGUNGSEBENE	3
2.1.1	<i>Aspekte des Übertragungssystems</i>	<i>3</i>
2.1.2	<i>Anforderungen an das Übertragungssystem</i>	<i>4</i>
2.1.3	<i>Security.....</i>	<i>5</i>
2.1.4	<i>Systemintegrität.....</i>	<i>5</i>
2.2	INTERAKTIONSEBENE	6
2.3	SEMANTIKEBENE.....	7
3	METRA4.0-BASIC MESSAGE TRANSMISSION.....	8
3.1	GRUNDMUSTER.....	8
3.2	ÜBERTRAGUNGSPROZEDUR.....	9
4	MODINT4.0-BASIC MODEL INTERACTION	12
4.1	GRUNDMUSTER.....	12
4.2	MODELL - META-META-MODELL.....	12
4.3	STANDARD MODELL-INTERAKTIONSDIENSTE	13
4.3.1	<i>Variablenzugriffsdienste.....</i>	<i>13</i>
4.3.2	<i>Erkundungsdienste</i>	<i>13</i>
4.3.3	<i>Modelltransformationssdienste.....</i>	<i>13</i>
4.3.4	<i>Modellverwaltungsdienste</i>	<i>14</i>
4.4	ORGANISATION DER INTERAKTION.....	14
5	DATPUB4.0-BASIC DATA PUBLISHING	15
5.1	GRUNDMUSTER.....	15
5.2	VERHÄLTNIS PUBLISHER-DATASOURCE	15
5.3	VERHÄLTNIS SUBSCRIBER-PUBLISHER.....	15
6	KOMPONENTENSTRUKTUREN.....	17
6.1	DIE TSR INFRASTRUKTURKOMPONENTE	17
6.2	DYNAMISCHE EINBINDUNG NEUER DEVICES UND KOMPONENTEN	17
7	SECURITY ASPEKTE.....	19
7.1	ALLGEMEIN	19
7.2	FÜR KOMPONENTENORIENTIERTE DEVICES.....	19

2 Übersicht Com4.0-Basic

Unter dem Begriff Kommunikation fasst man sämtliche Aspekte des Informationsaustauschs zwischen Anwendungen zusammen. Es ist offensichtlich, dass Kommunikation ein sehr vielgestaltiges Themenfeld umfasst.

Das vorliegende Papier behandelt einige grundlegenden Aspekte der Kommunikation. Getroffene Festlegungen sind, wie in Abbildung 2.1 dargestellt, der generischen und der konzeptionellen Modellebene zuzuordnen. Ziel ist die Spezifikation eines allgemeinen, technologieunabhängigen Kommunikationsmodells das minimalistisch gerade so ausgeprägt ist, dass es in sich konsistent und abgeschlossen eine Basis-Interoperabilität zwischen den Kommunikationspartnern ermöglicht. Die Abbildung auf bestimmte Technologien oder die Beschreibung von konkreten Kommunikationslösungen sind nicht Teil dieses Papiers.

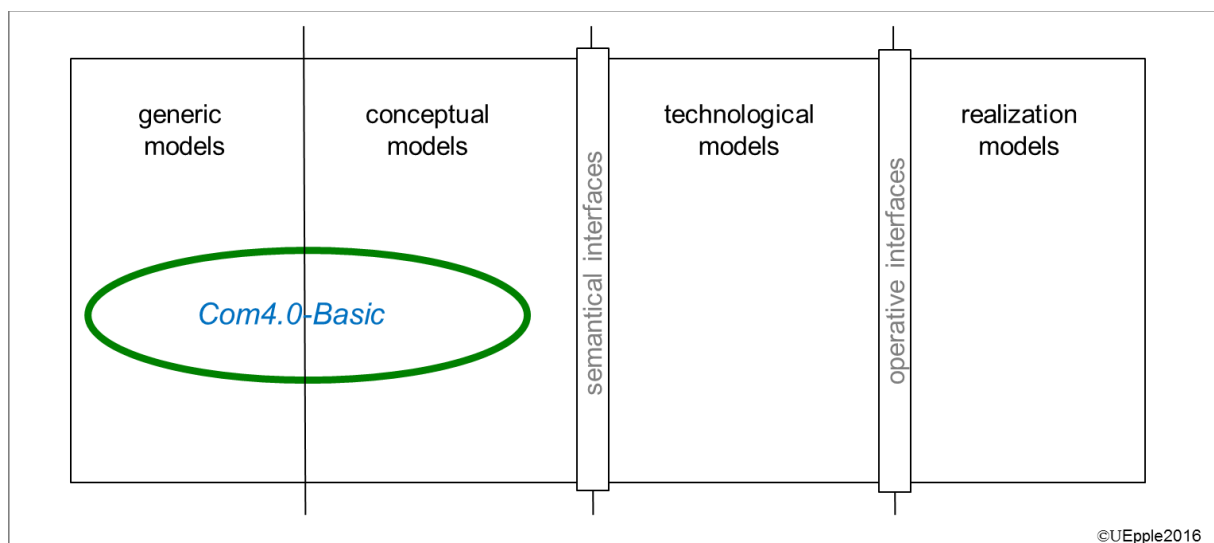


Abbildung 2.1: Einordnung von Com4.0-Basic in die Modellabstraktionsebene

Das Papier konzentriert sich auf die in industriellen IT-Systemen wesentlichen Interaktionsmuster. Grundlage ist eine Gliederung der Kommunikationsaufgabe in die drei Themenbereiche: *Übertragung*, *Interaktion* und *Semantik*. In **Abb. 2.2** ist diese Gliederung dargestellt.

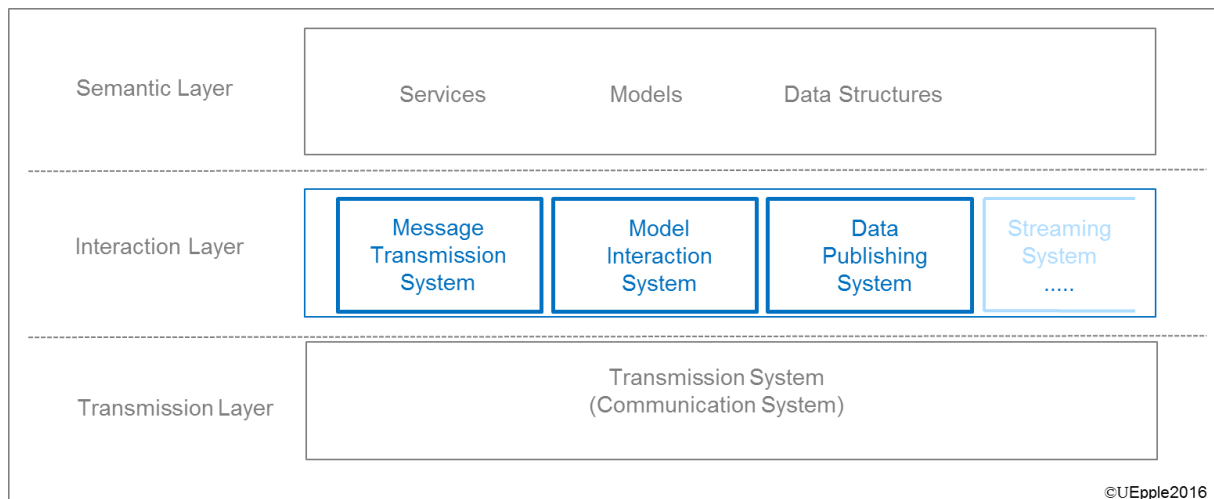


Abbildung 2.2: Gliederung der Kommunikationsaspekte in drei Ebenen

Die *Übertragung* und die *Interaktion* bilden nicht nur einen eigenen Themenbereich sondern können als operative, in sich geschlossene funktionale Ebenen aufgefasst werden. Sie bieten aufrufbare Funktionen („Dienste“) an, die sie den übergeordneten Ebenen zur Nutzung zur Verfügung stellen. Die Trennung in in sich geschlossene Ebenen bedeutet nicht, dass die Ebenen unabhängig voneinander gestaltet sind. So sind z.B. die Dienste der Übertragungsebene speziell auf die Anforderung der Interaktionsmuster abgestimmt, sodass durch die Schnittstelle zwischen der Übertragungs- und der Interaktionsebene keine Effizienzverluste entstehen.

Der Fokus dieses Papiers liegt auf der Interaktionsebene. Diese enthält die Muster nach denen die Anwendungen Informationen austauschen und kapselt die gesamte Übertragungsproblematik gegenüber den Anwendungen. In diesem Überblick sollen jedoch alle drei Ebenen kurz charakterisiert werden.

2.1 Übertragungsebene

In den Fällen in denen sich die Kommunikationspartner nicht in derselben lokalen Umgebung befinden, also z.B. durch Prozessgrenzen getrennt sind oder sich in unterschiedlichen Knoten eines Netzwerks befinden, muss jede einzelne auszutauschende Information über das Netzwerk übertragen werden. Die Realisierung der Übertragungsfunktion ist eine eigenständige Aufgabenstellung. Sie ist unabhängig von dem Interaktionsmuster zwischen den Komponenten und auch unabhängig vom Informationsinhalt zu lösen.

In dem hier verfolgten Modell gehen wir davon aus, dass es ein Übertragungssystem gibt, das die Übertragungsaufgabe in ihrem gesamten Umfang löst. In vielen Fällen wird ein solches operatives System, das die Übertragungsaufgabe insgesamt löst als auch als das „Kommunikationssystem“ bezeichnet. Um Verwechslungen zu vermeiden bleiben wir jedoch hier bei der Bezeichnung *Übertragungssystem*.

2.1.1 Aspekte des Übertragungssystems

In **Abb. 2.4** sind die verschiedenen Aspekte dargestellt, die bei der Betrachtung des Übertragungssystems zu beachten sind.

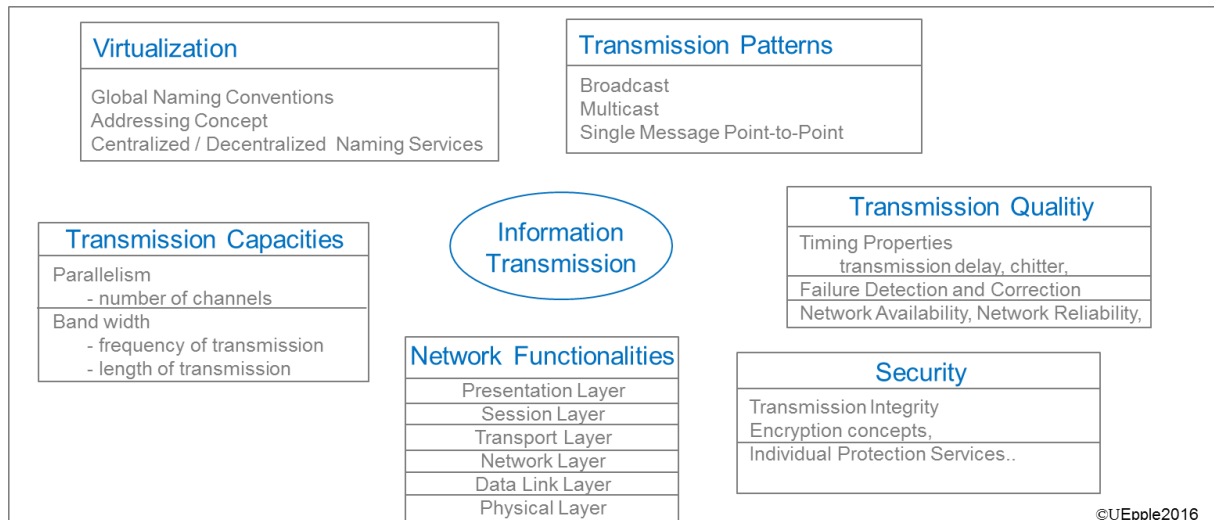


Abbildung 2.4: Aspekte des Übertragungssystems

In dieser Gliederung umfasst die Nachrichtenübertragung alle Aspekte die sich aus der Übertragungsaufgabe ergeben. Dazu gehören: die Funktionalität sämtlicher OSI Netzwerkebenen 1-6 (Network Functionalities), Leistungsparameter der Übertragung (Transmission Capacities), Qualitätsparameter der Übertragung (Transmission Quality), die verschiedenen Arten der Übertragung (Transmission Patterns), die Sicherstellung der Integrität der übertragenen Daten, der Schutz der Kommunikationspartner vor unzulässigen Requests und die Virtualisierung der Netzwerkstruktur gegenüber den Anwendungen (Virtualization). Insgesamt kann man festhalten, dass das Übertragungssystem alle Aspekte umfasst die sich aus der Notwendigkeit ergeben eine Information einem Empfänger über das Netzwerk zuzustellen. Ein Test, ob ein Aspekt zur Informationsübertragung gehört oder nicht, ist die einfache Prüfung ob dieser Aspekt auch relevant wäre, wenn sich alle an der Kommunikation beteiligten Anwendungen in einer lokalen Umgebung befinden würden. Ist dies der Fall, dann gehört dieser Aspekt nicht zur Informationsübertragung. Befinden sich alle an einer Kommunikation beteiligten Anwendungen in einer lokalen Umgebung, dann entfallen alle Aspekte der Informationsübertragung.

2.1.2 Anforderungen an das Übertragungssystem

Das konzeptionelle Design des Übertragungssystems als gerade auch mit dem Hinblick auf eine effiziente Realisierbarkeit mit verfügbaren Technologien ist eine schwierige Aufgabe und vermutlich an vielen Stellen heute noch nicht technisch zufriedenstellend zu lösen. Die Bereitstellung eines Vorgehensmodells wie ein solches Übertragungssystem systematisch angelegt, mit den vorhandenen Technologien umgesetzt und Schritt für Schritt ausgebaut werden kann ist eine der zentralen Aufgaben bei der Entwicklung der Industrie 4.0 Infrastrukturplattform. In diesem Papier wird diese Thematik nicht betrachtet. Es wird davon ausgegangen, dass ein solches Übertragungssystem zumindest in rudimentärer Form existiert und dass dieses jeder Anwendung in jeder lokalen Umgebung zur Verfügung steht.

Aus Sicht von Com4.0-Basic werden vom Übertragungssystem folgende äußere Eigenschaften erwartet:

P-22-01: Jeder Anwendung steht das Nachrichtenübertragungssystem lokal zur Verfügung.

P-22-02: Will eine Komponente an der Nachrichtenübertragung teilnehmen (als Empfänger oder als Sender), dann muss sie sich beim Übertragungssystem anmelden.

P-22-03: Das Übertragungssystem kennt systemweit die (logischen) Orte der angemeldeten Anwendung und/oder hat die Fähigkeit diese dynamisch zu ermitteln.

P-22-04: Das Übertragungssystem kennt die Übertragungswege zu einer Anwendung und/oder besitzt die Fähigkeit diese dynamisch zu ermitteln.

P-22-05: Die Qualität der Nachrichtenübertragung wird durch standardisierte Kennzahlen beschrieben [2]. Diese sind Teil der konzeptionellen Modelle und werden mit diesen zusammen festgelegt. Kennzahlen machen z.B. Zusicherungen zu maximalen Verzögerungszeiten, Ausfallwahrscheinlichkeiten, Sicherheitslevels etc.

P-22-06: Die QoS Parameter können sich aus der Art der Realisierung zwangsweise ergeben oder mit dem Übertragungssystem explizit dynamisch vereinbart werden.

2.1.3 Security

Dem Übertragungssystem können zwei Security-Aufgaben zugeordnet werden:

- Die Sicherstellung der Integrität der übertragenen Daten
- Der Schutz der Kommunikationspartner vor unerwünschten Requests

P-22-07: Die zentrale Aufgabe eines Übertragungssystems ist die unverfälschte, abhörsichere Übermittlung der Daten an den Empfänger unter Einhaltung der vereinbarten QoS (dazu gehört auch die Einhaltung der Zeitbedingungen). Es obliegt dem Übertragungssystem diese Anforderungen sicherzustellen, z.B. durch eine entsprechende Verschlüsselung der Übertragung und Authentifizierung der beteiligten Partner.

P-22-08: Das Übertragungssystem ist jedoch weder für den korrekten Inhalt der Daten noch für die innere Integrität der beteiligten Anwendungen verantwortlich.

P-22-09: Darüber hinaus kann das Übertragungssystem je nach Anforderung zusätzliche Dienste zum Schutz der Kommunikationspartner vor unerwünschten Requests anbieten. Dazu kann es Anwendungen z.B. anbieten, über einen Dienst festzulegen, von welchen Anwendungen sie überhaupt Requests akzeptieren. Requests anderer Anwendungen werden dann überhaupt nicht zugestellt. Eine zweite Möglichkeit des Schutzes ist die Begrenzung der Sendefrequenz. Zur Vermeidung von Pufferüberläufen, Kommunikationsengpässen und einer Flutung der Anwendungen mit Anfragen kann das Übertragungssystem von sich aus die Anzahl der Requests pro Zeiteinheit auf das mit einer Anwendung vereinbarte Maß begrenzen.

P-21-08: Das Übertragungssystem bietet ein bestimmtes Maß an Übertragungssicherheit an (z.B. eine gewisse Verschlüsselung). Genügt dieses Maß einer Anwendung nicht, dann kann diese Anwendung versuchen durch zusätzliche eigene Maßnahmen die Anforderungen zu erfüllen (z.B. zusätzlich selbst verschlüsseln). Diese Maßnahmen sind für das Übertragungssystem transparent

2.1.4 Systemintegrität

Für die Integrität des Übertragungssystems selbst gilt folgende Forderung:

P-22-10: Die Anwendungen betrachten das Übertragungssystem per se als vertrauenswürdig. Das heißt: Die Anwendungen vertrauen, dass das Übertragungssystem die von ihm zugesicherten QoS Eigenschaften auch tatsächlich besitzt. Dies besagt nichts über die Qualität der zugesicherten Eigenschaften, dies besagt nur, dass den vom Übertragungssystem gemachten Angaben absolut vertraut wird. Diese Annahme stellt weitreichende Anforderungen an die innere Integrität des Übertragungssystems und erfordert eine sorgfältige Analyse der Security des Übertragungssystems als System.

2.2 Interaktionsebene

Die Form der Interaktion zwischen zwei Kommunikationspartnern hängt wesentlich vom Zweck dieser Interaktion ab. Aus dieser Zwecksicht kann man mehrere grundlegende Interaktionsschemata unterscheiden. Wie in **Abb. 2.2** dargestellt sind für die industrielle Automation insbesondere die *Nachrichtenübertragung*, die *Modellinteraktion* und die *Datenpublizierung* von Interesse. Sie decken die Interaktionsanforderungen der I4.0-Domäne weitgehend ab. In **Abb.2.5** sind die charakteristischen Eigenschaften dieser drei Interaktionsschemata dargestellt.

Message Transmission	Model Interaction	Data Publishing
decided message between applications	read explore write	create delete link..
<ul style="list-style-type: none"> • messages directly between applications, • only applications have to understand the message, • no general semantic necessary, • network and environment is completely virtualized, • contract with I40 infrastructure components concerning the transmission quality, • defined unique access point for every I40 component. 	<ul style="list-style-type: none"> • standardized services of the I40 infrastructure components to interact with addressed information models • communication partner is I40 infrastructure component • special access points can be defined to reach special parts of the model • access points are associated with special access rights and QoS. • virtualization needs special functionality. • direct access to the internal model elements 	<ul style="list-style-type: none"> • efficient data publishing - actual states - mass data • standardized services of the I40 infrastructure components to establish an efficient publishing of data, • typically realized by a Pub/Sub-model, • clear separation of communication management services (subscribe, register, delete..) and productive exchange services, • close coupling with the functionality of the underlying network. • decoupling of data source and data user

©UEpple2016

Abbildung 2.5: Charakteristiken der drei grundlegenden Interaktionsschemata

Bei Bedarf können einzelne weitere Grund-Schemata hinzugefügt werden. Wichtig ist, dass diese Schemata passgenau auf die Struktur der Anforderungen zugeschnitten sind. Natürlich kann man mit jedem Interaktionsschema alle Anforderungen prinzipiell abdecken. Dies führt jedoch zu komplizierten und ineffizienten Abläufen und erschwert die Nutzung und das Management des darunter liegenden Übertragungssystems erheblich.

Die hier dargestellten Schemata stellen strukturelle Grund-Muster dar. Die eigentlichen Abläufe einer Interaktion, das heißt, das Wechselspiel zwischen Anfragen und Antworten werden in separaten Dienstprotokollen spezifiziert. Dienstprotokolle lassen sich ebenfalls standardisieren und als Ablaufmuster beschreiben. Im Fall der Modellinteraktion und der Datenpublizierung ergeben sich die Protokolle aus der Aufgabenstellung selbst. Das heißt, in diesen Fällen gibt es einen Satz von standardisierten Protokollen zur Realisierung der Interaktion. Diese werden in den folgenden Kapiteln skizziert. Für komplexere und spezifische Interaktionsmuster empfiehlt es sich grundsätzlich direkt auf die Nachrichtenübertragung als Schema aufzusetzen. Die Protokolle werden in den entsprechenden Dienstmusterspezifikationen festgelegt.

Die Schemata der Interaktionsebene werden in den folgenden Kapiteln ausführlich erläutert.

2.3 Semantikebene

Damit zwei Partner kommunizieren können müssen sie eine gemeinsame Semantik besitzen. Dabei ist es unerheblich ob diese Semantik explizit formuliert ist oder ob sie sich in den Funktionen der Partner nur durch eine bestimmte Syntax intrinsisch ausdrückt.

Im Bereich der Automaten muss man sich von der Vorstellung lösen, dass eine Anwendung einen Sachverhalt *versteht*, sich also über dessen Semantik bewusst ist. Vielmehr drückt sich die semantische Kenntnis einer Anwendung nur dadurch aus, dass sie einzelne Aufgabenstellungen kennt und die zu ihrer Lösung erforderlichen Funktionen besitzt. Sie besitzt also die Fähigkeit auf bestimmte Eingangszustände zu reagieren und die für diese Situation vorgesehenen Funktionen anzustoßen. Dieser Ablauf kann zwar sehr komplex sein, das ändert jedoch nichts an der prinzipiellen Tatsache. Um dies zu verdeutlichen: Ein System versteht nicht was das Merkmal 0173-1#02-BAG982#006 Nenndurchmesser in der Welt bedeutet, es besitzt jedoch Funktionen die Wertaussagen zu diesem Objekt automatisch vergleichen und verrechnen können.

Für den operativen Informationsaustausch muss ein Kommunikationspartner annehmen, dass der andere Partner die seiner Semantik entsprechenden Funktionen besitzt, d.h. dass er seine Informationen richtig interpretieren kann. So kann es genügen, dass ein Sensor den Wert „14.4“ meldet wenn sein Partner weiß, was er mit diesem Wert anfangen soll. Ist dem nicht so, dann muss man der Information weitere Metadaten hinzufügen (z.B. gemessen in °C) die als zusätzliche Informationen für die Auswertung benötigt werden. Es ist Angelegenheit des jeweiligen Anwendungsfalls, sich hier über das geeignete Maß an hinterlegtem gegenseitigem Vorabwissen zu verständigen.

In komplexeren Systemen wird man jedoch strukturell vorsehen, dass sich Kommunikationspartner im Betrieb dynamisch über ihre Fähigkeiten verständigen können. Dazu dient eine gemeinsame Semisemantik die jedoch selbst von allen Kommunikationspartnern unterstützt werden muss. Eine solche Semisemantik beschreibt z.B. wie mit Diensten umzugehen ist, wie Aussagen zu Merkmalen erkannt und behandelt werden, wie Abhängigkeiten zwischen Systemelementen beschrieben werden, wie Zusammenhänge als eine Liste von RDF Statements formuliert werde usw.

Auf der Basis einer solchen Semisemantik können Partner dann prüfen, ob sie überhaupt zusammenpassen, ob sie bereits ausreichende gegenseitige Kenntnisse besitzen, welche Metainformationen sie eventuell noch vorab austauschen müssen und inwieweit sie noch vorab ihre Konfiguration aufeinander abstimmen müssen (Laden entsprechender Standardbibliotheken?..).

Gerade in komplexen Kommunikationssystemen wird die Frage der Abstimmung der Semantik zwischen den beteiligten Anwendungen zu einem zentralen Problem. Dieser Aspekt soll hier jedoch nicht weiter vertieft werden. Es sei jedoch noch einmal herausgestellt, dass dieser Aspekt getrennt von der Frage der Interaktionsmodelle und der Frage der Übertragungskonzepte behandelt werden kann. Natürlich werden sich aus den Informationsaustauschanforderungen der Anwendung Anforderungen an das Interaktionsmodell und die Übertragungskonzepte ergeben, diese sind jedoch klar voneinander abgegrenzt.

3 MeTra4.0-Basic Message Transmission

3.1 Grundmuster

In **Abb. 3.1** ist das Grundprinzip eines nachrichtenbasierten Informationsaustauschs dargestellt.

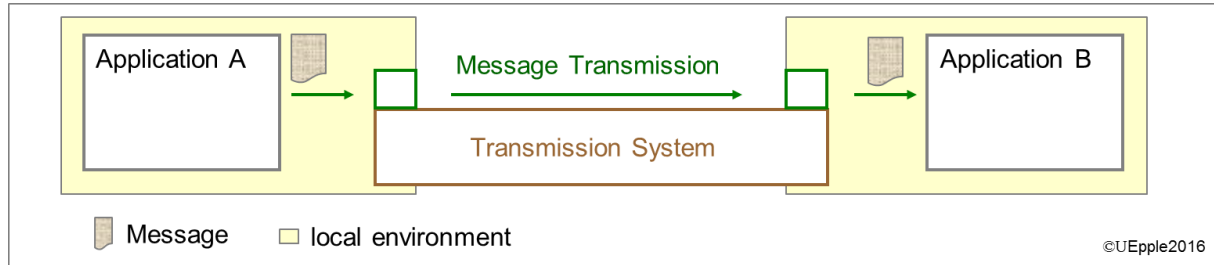


Abbildung 3.1: Grundprinzip des nachrichtenbasierten Informationsaustauschs

Eine Anwendung A will einer Anwendung B eine Nachricht zukommen lassen. Anwendung A befindet sich in einer lokalen Umgebung I und Anwendung B befindet sich in einer lokalen Umgebung II. Zur Übertragung der Nachricht an die Anwendung B nutzt Anwendung A ein Übertragungssystem. Das Übertragungssystem kennt den (logischen) Ort an dem sich die Anwendung B befindet und stellt Anwendung B die Nachricht unverändert zu.

Dieses Grundschema einer Nachrichtenübertragung wird im Metamodell „MeTra4.0“ (Message Transmission Metamodel) beschrieben. Es wird durch folgende Eigenschaften charakterisiert:

Interaktion

P-3101: MeTra4.0 betrachtet immer nur die Übertragung einer einzelnen Nachricht von einem Sender an einen Empfänger. Antworten sind eigene Nachrichten.

P-3102: Für MeTra4.0 ist es vollständig unerheblich, um welche Art von Nachricht es sich handelt. Es kann ein Auftrag, eine Anfrage, eine Registrierung, eine Antwort auf eine vorangegangene Nachricht, eine Änderungsmeldung oder irgendeine andere Nachricht sein.

Semantik

P-3103: MeTra4.0 enthält keinerlei Angaben zur Semantik der übermittelten Informationen. Es bietet auch keinerlei Funktionen, Dienste oder Modellvorstellungen zur Abstimmung der Semantik zwischen den Kommunikationspartnern.

Virtualisierung

P-3104: Im Kontext des MeTra4.0 Modells benötigt eine Anwendung keinerlei Kenntnisse über die Struktur des sie umgebenden Systems. Will sie einer anderen Anwendung eine Nachricht senden, dann benötigt sie nur deren (global) eindeutigen Namen. Sie muss weder wissen wo sich die Empfängeranwendung befindet, noch auf welchem Weg sie zu erreichen ist.

P-3105: Für MeTra4.0 ist das Übertragungssystem selbst und das gesamte Netzwerk vollständig transparent

3.2 Übertragungsprozedur

Im MeTra4.0-Modell sind die Sender und Empfänger von Nachrichten Anwendungen. Für die operative Übertragung einer Nachricht bietet MeTra4.0 senderseitig eine Annahmeschnittstelle an. Diese kann prinzipiell auf verschiedene Weise realisiert sein (Dienstschnittstelle, API, Briefkasten..). Je nach Anforderungen an die Interoperabilität und Modularität muss die Realisierung der Annahmeschnittstelle jedoch für bestimmte Umgebungstypen lokal oder sogar systemweit (freie dynamische Zuordnung von Komponenten?) standardisiert werden. Die eigentliche Annahme und Übertragungsfunktionalität ist in jedem Fall standardisiert und wird vom Übertragungssystem vollständig gekapselt.

In einer lokalen Umgebung können sich mehrere Anwendungen befinden. Jede Anwendung wickelt sowohl das Versenden als auch das Empfangen ihrer Nachrichten selbst ab.

In **Abb. 3.2** ist die Prozedur eines Nachrichtenaustauschs mit Anfrage und Antwort erläutert.

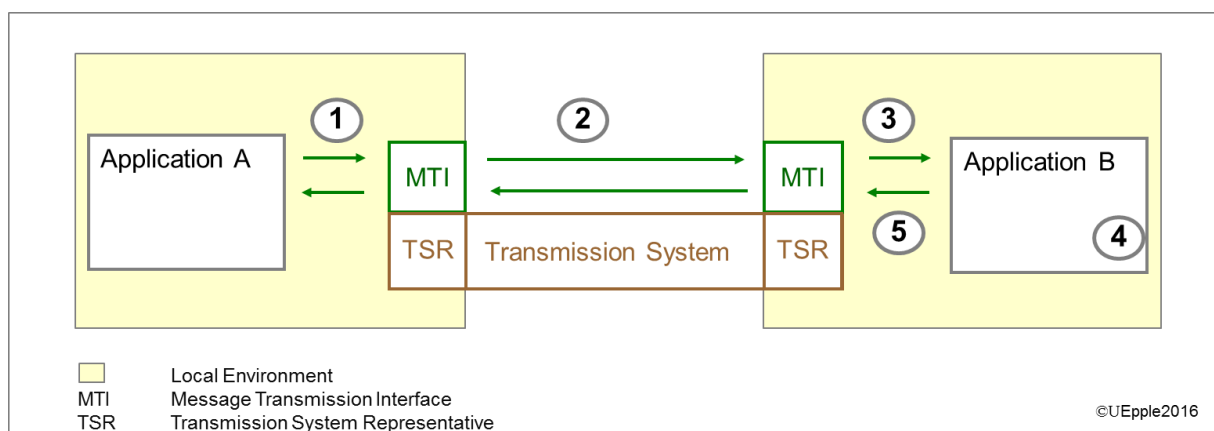


Abbildung 3.2: Beispielablauf eines Nachrichtenaustauschs

Szenario: Anwendung A sendet Anwendung B eine Anfrage und erhält darauf eine Antwort. Im ersten Schritt übergibt Anwendung A die Nachricht an den lokalen Repräsentanten des Übertragungssystems. Diesen bezeichnen wir als TSR (Transport System Representative). Der TSR stellt dafür spezielle Schnittstellen MTI (Message Transmission Interfaces) zur Verfügung. Die Möglichkeit der Ausgestaltung dieser Übergabeschnittstellen wird später in einem eigenen Kapitel beschrieben. Jede Nachricht hat einen Umschlag auf dem der Name des Empfängers und des Absenders steht. Der TSR übernimmt die Nachricht und leitet sie an den TSR weiter in dessen Umgebung sich die Empfängeranwendung befindet (2). Dieser TSR übergibt die Nachricht (3) an die Empfängerkomponente. Die Empfängerkomponente übernimmt die Nachricht, führt je nach Inhalt die entsprechenden Dienste (4) aus und sendet ihre Antwort (5) an den Absender zurück. Die Rücksendung der Antwort ist eine eigene unabhängige Nachrichtenübertragung. Das Übertragungssystem weiß nicht, ob es sich um eine Anfrage oder eine Antwort handelt.

Die hier skizzierte Austauschprozedur ist durch folgende Eigenschaften charakterisiert:

Nachrichtenformat

P-3201: Jede Nachricht wird für die Übertragung mit einem Header versehen. Der Header besteht, wie in **Abbildung 3.3** dargestellt, aus verschiedenen Teilen. Dem Namen (global eindeutigen ID) des Empfängers, dem Namen (global eindeutigen ID) des Absenders und zusätzlichen Verwaltungsinformationen für das Übertragungssystem und die empfangende Anwendung.

P-3202: Das Übertragungssystem interpretiert nur den Namen des Empfängers und den Namen des Absenders. Bestehen Übertragungsverträge zwischen der absendenden Anwendung und dem Übertragungssystem, dann kann die absendende Anwendung angeben nach welchem Übertragungsvertrag die Übertragung erfolgen soll.

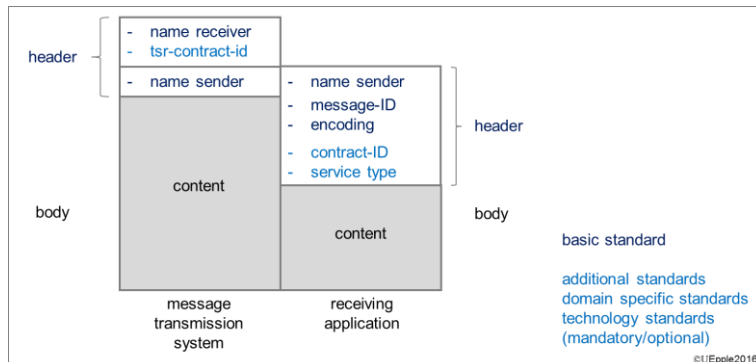


Abbildung 3.3: Format einer Nachricht bei der Übertragung

P-3203: Für die empfangende Anwendung besteht der Header zumindest aus dem Namen des Absenders, einer Nachrichten-ID und einer Information zur Dekodierung. Weitere Informationen wie z.B. zum Typ des Dienstes oder zu einem bestehenden Vertrag können folgen.

P-3204: Die Nachrichten-ID (message-ID) wird vom Absender vergeben und erlaubt eine eindeutige Identifizierung einer Nachricht. So kann z.B. bei einer Antwort auf diese ID verwiesen und so der Bezug zur ursprünglichen Nachricht hergestellt werden.

P-3205: Die Dekodier-ID (encoding-ID) zeigt dem Empfänger an, nach welchem Schema er die Nachricht dekodieren muss.

P-3206: Zusätzliche domänenspezifische oder technologische Standards können weitere zwingende oder optionale Elemente spezifizieren.

P-3207: Der Rumpf enthält die eigentliche Nachricht. Deren Aufbau und Semantik ist interne Angelegenheit der Anwendungen. Für die Nachrichtenübertragung ist der Rumpf nur eine nicht zu interpretierende Folge von Daten.

Namen

P-3208: In Com4.0 hat jede Anwendung einen global eindeutigen Namen (global eindeutige Identifizierung). Com4.0 macht keinerlei Vorgaben an die Gestaltung des globalen Namensraums. Falls dieser Namensraum hierarchisch gestaltet ist können in einem Cluster das vollständig in einem gemeinsamen Ober-Namensraum liegt die lokalen Namen ohne den vorangestellten Ober-Namensraum verwendet werden.

Adressen

P-3209: Das Modell sieht eine vollständige Virtualisierung der Verteilung der Anwendungen vor. Eine Anwendung weiß nicht wo sich die anderen Anwendungen befinden. Die Adressierung einer Anwen-

ung erfolgt prinzipiell über deren Namen. Das Wissen um die Zuordnung in welcher lokalen Umgebung sich eine Anwendung befindet liegt vollständig im Übertragungssystem.

P-3210: Wie das Wissen über die Zuordnung einer Anwendung zu einer lokalen Umgebung abgelegt ist (zentral/dezentral), wie es verwaltet wird (kanalbezogen/sessionbezogen/gecached) und wie es erfasst wird (dynamisch/statisch.. usw.) ist Teil des spezifischen technologischen Modells.

P-3211: Jede Anwendung die sich an der Nachrichtenübertragung beteiligen möchte muss sich beim Übertragungssystem explizit anmelden. Eine Anwendung die angemeldet ist nimmt direkt am Nachrichtenaustausch teil, eine Anwendung die nicht angemeldet ist, ist im System nicht sichtbar.

4 ModInt4.0-Basic Model Interaction

4.1 Grundmuster

In **Abb. 4.1** ist das Grundprinzip des Modell-Interaktions-Systems dargestellt.

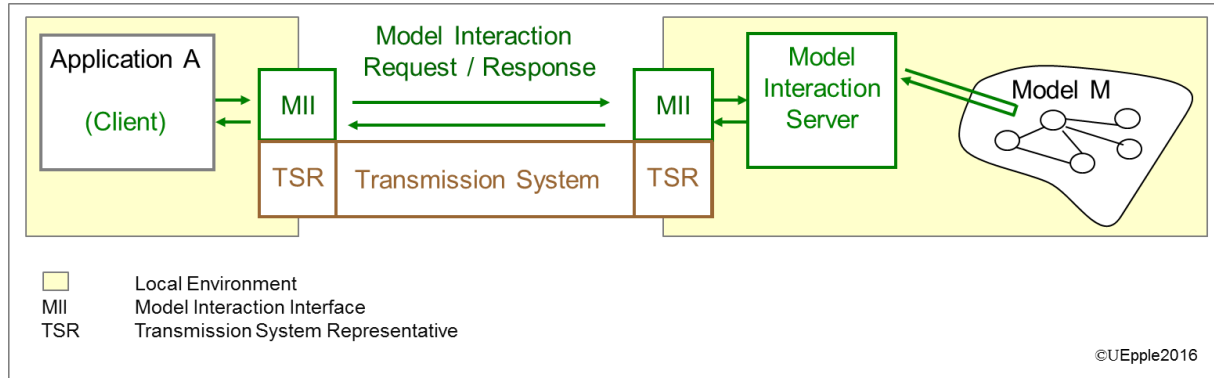


Abbildung 4.1: Grundprinzip des Modell-Interaktions-Systems

Eine Anwendung A möchte auf den Inhalt eines Modells M zugreifen. Dazu bietet das System standardisierte Modell-Interaktions-Dienste an, die durch einen Modellinteraktionsserver zur Verfügung gestellt werden. Der Modellinteraktionsserver befindet sich in der lokalen Umgebung des Modells. Die Realisierung des Zugriffs des Modellinteraktionsservers auf das Modell ist interne Angelegenheit der lokalen Systemumgebung.

4.2 Modell - Meta-Meta-Modell

Grundlage der Modellinteraktionsdienste ist ein gemeinsames Verständnis des Grundaufbaus eines Modells. Diese Grundstruktur leitet sich aus dem E/R-Meta-Meta-Modell eines Systems ab, sie ist in Abb. 4.2 dargestellt.

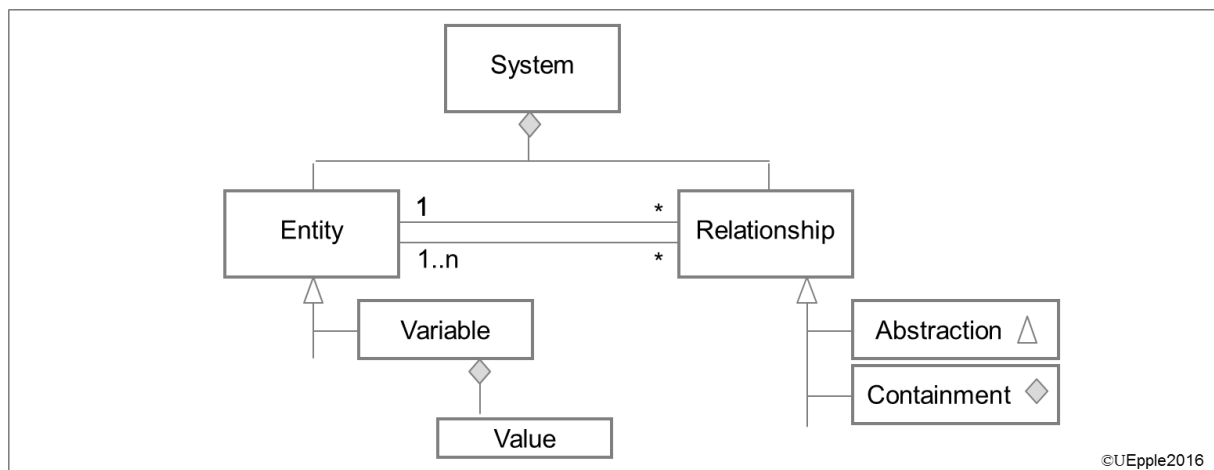


Abbildung 4.2: Den Modell-Interaktionsdiensten zugrundeliegender Teil eines allgemeinen E/R Meta-Meta-Systemmodells

4.3 Standard Modell-Interaktionsdienste

Das Modell-Interaktionssystem stellt Modell-Interaktions-Dienste zur Verfügung. Diese müssen funktional unabhängig von den verwendeten Technologien einheitlich standardisiert sein. Modellinteraktionsdienste dienen z.B. dem Lesen und Schreiben von Variablenwerten, dem Erkunden einer Modellstruktur, dem Navigieren innerhalb eines Modells, dem Anlegen, Löschen oder Umbenennen einer Entität, dem Anlegen und Löschen von Verlinkungen und der Modellverwaltung. Glücklicherweise unterstützen alle gängigen technologisch verfügbaren Modell-Interaktions-Systeme die grundlegenden Dienste mit sehr ähnlicher Funktionalität. Offensichtlich ist das E/R Meta-Meta-Systemmodell tatsächlich auch für die technologischen Systeme eine gemeinsame Grundlage.

4.3.1 Variablenzugriffsdienste

Lesen von Variablenwerten

Dies ist der einfachste und unproblematischste Dienst. Er erlaubt das rückwirkungsfreie Auslesen eines Variablenwerts. (typische technologische Bezeichnungen: read, get,...). Dieser Dienst kann bei Vorliegen der Leseberechtigung jederzeit ausgeführt werden. Der Lesedienst gibt mindestens den Variablenwert (und/oder einen Fehlercode) zurück.

Übertragen von Variablenwerten

Dieser Dienst ist ein komplexer, zustandsbehafteter Dienst. Er folgt einem einfachen Publisher/Subscriber-Muster. Der Klient (Subscriber) meldet sich beim Model Interaction Server (Publisher) für eine ständige Übertragung eintretender Wertänderungen oder Ereignisse an. Der Model Interaction Server steht mit dem Modell in ständigem Austausch und sendet die ermittelten Änderungen/Ereignisse dem Klienten. Wie dies im Einzelnen organisiert ist, hängt stark von der technologischen Realisierung ab. Insgesamt erhält jedoch der Klient bis auf Widerruf alle Änderungen/Ereignisse, für die er sich angemeldet hat, automatisch zugestellt. (typische technologische Bezeichnungen: pub/sub,...).

Schreiben von Variablenwerten

Mit diesem Dienst können neue Variablenwerte gesetzt werden. (typische technologische Bezeichnungen: write, set,...). Bei Vorliegen der Schreiberlaubnis (für Klienten) und prinzipieller Freigabe des Schreibvorgangs (von Seiten der Modellverwaltung) wird der bisherige Wert einfach einmalig mit dem neuen Wert überschrieben. Der Schreibdienst gibt mindestens eine Information über den erfolgten Schreibvorgang (und/oder einen Fehlercode) zurück.

4.3.2 Erkundungsdienste

Modellerkundung

Mit diesem Dienst kann der Aufbau eines Modells erkundet werden.

Der Dienst gibt die Namen aller Entitäten zurück, die ausgehend von einer vorgegebenen Entität mit dieser durch eine Assoziation (oder durch alle Assoziationen eines bestimmten Typs) verbunden sind. (typische technologische Bezeichnungen: explore, getstruc, getep...).

4.3.3 Modelltransformationssdienste

Create

Dieser Dienst legt eine neue Entität oder eine neue Assoziation (als benannte Entität) an (typische technologische Bezeichnungen: create...).

Delete

Dieser Dienst löscht eine Entität oder eine Assoziation (typische technologische Bezeichnungen: delete...).

Link/Unlink

Dieser Dienst legt eine Assoziationsinstanz vom Typ „Link“ an oder löscht diese (typische technologische Bezeichnungen: link/unlink...).

4.3.4 Modellverwaltungsdienste

Mit zusätzlichen Verwaltungsdiensten kann die gesamte Modellpflege und Verwaltung dienstbasiert abgewickelt werden. Dies ist z.B. besonders interessant um Modelle zur Laufzeit zu verwalten. Solche Verwaltungsdienste sind z.B. Dienste zum Laden und Archivieren von Modellen und Modellteilen, zum Sperren und Freigeben der Nutzung. Es ist zu prüfen ob diese Dienste mit in den Basisstandard Com4.0 als allgemeiner Standard aufgenommen werden sollten. Zunächst wird auf eine Aufnahme verzichtet und auf die Lösungen verwiesen, die die verfügbaren Technologien bieten.

4.4 Organisation der Interaktion

Der Modellinteraktionsserver ist eine standardisierte Systemanwendung. Die Modell-Interaktionsdienste werden nach einem standardisierten Protokoll abgewickelt. Außer bei der „Übertragung von Variablenwerten“ liegt allen anderen Diensten eine einfache einmalige Request/Response – Folge zugrunde.

Für die konzeptionelle Umsetzung der Modellinteraktionsdienste gibt es zwei Varianten:

Variante 1:

Fasst man den Modellinteraktionsserver als eine allgemeine Anwendung auf, dann können die Modellinteraktionsdienste auf eine Folge von einfachen Nachrichtenübertragungen zwischen zwei Anwendungen nach dem MeTra4.0 Schema abgebildet werden. In diesem Fall würde das Übertragungssystem keinerlei zusätzliche Funktionalität zur Abwicklung der Modellinteraktionsdienste benötigen. Dies ist eine Option die jeder Realisierung immer offensteht.

Variante 2:

In der zweiten Variante werden die Modellinteraktionsdienste als ein eigenes grundlegendes Schema angesehen, das im Übertragungssystem speziell abgebildet wird. Sie stehen der Anwendung über das Model Interaction Interface (MII) direkt zur Verfügung. Dafür gibt es technische und praktische Gründe. Der wesentlichste Grund für diese Wahl ist ein praktischer: Relevante bestehende Technologien (wie z.B. OPC-UA) bauen direkt auf den speziellen Diensten der Modellinteraktion auf. Wollte man diese nach Variante 1 realisieren, müssten die technologischen Standards noch einmal separat in eine Standardnachricht verpackt werden. Dies ist in vielen Fällen nicht erwünscht und soll durch Com4.0 auch nicht erzwungen werden. Mit Variante 2 ist eine direkte Abbildung der bestehenden Technologien auf das Com4.0-Konzept möglich..

5 DatPub4.0-Basic Data Publishing

5.1 Grundmuster

In **Abb. 5.1** ist das Grundprinzip des Data Publishing Systems dargestellt.

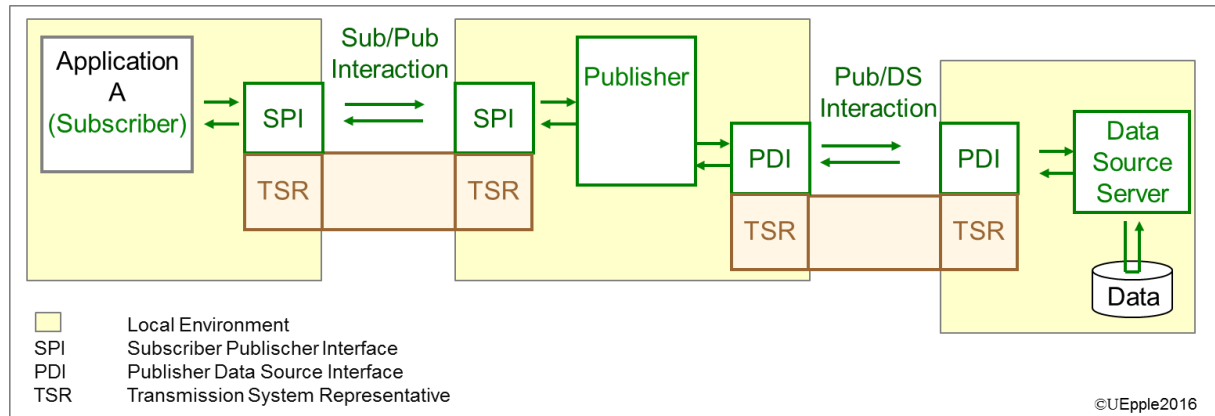


Abbildung 5.1 Grundmuster des Data Publishing

Wie dargestellt schiebt sich in dieser Struktur der Publisher als eine eigene aktive System-Anwendung zwischen die Datenquelle und den Datennutzer. Diese Struktur ist durch folgende Eigenschaften gekennzeichnet:

P-5101: Der Publisher ist eine eigene aktive Systemanwendung

P-5102: Der Informationsaustausch Publisher/Subscriber und DataSource/Publisher ist logisch vollständig getrennt und kann durch unterschiedliche Technologien realisiert werden.

5.2 Verhältnis Publisher-DataSource

Gegenüber der Data Source tritt der Publisher als ein einfacher standardisierter Klient auf. Er übernimmt die Daten und ist für deren Weiterleitung verantwortlich. Der Austausch kann je nach technologischer Lösung nach unterschiedlichen Mustern erfolgen. Ziel ist typischerweise eine einfache Realisierung auf Seiten der DataSource und eine effiziente Übertragung eines festgelegten Satzes von Daten. Die Initiierung erfolgt im Allgemeinen von Seiten des Publishers. Es sind jedoch auch Lösungen denkbar bei denen sich eine DataSource nach Inbetriebnahme automatisch bei einem standardisierten Publisher anmeldet und mit der Übertragung beginnt. Für das Zusammenwirken zwischen Publisher und DataSource gelten folgende Eigenschaften:

P-5201: Die Data Source vertraut dem Publisher und übergibt ihm ihre Daten nach Vereinbarung.

P-5202: Die Data Source weiß nicht an wen die Daten weitergeleitet werden und hat auf diese Weiterleitung keinen Einfluss.

5.3 Verhältnis Subscriber-Publisher

In diesem Verhältnis ist in jedem Fall der Subscriber der Initiator der Beziehung. Der Aufbau der Beziehung geht also immer vom Subscriber aus. Eine Beziehung muss explizit aufgebaut werden. Der Publisher stellt dafür geeignete Administrationsservices zur Verfügung. Der Publisher stellt die Daten in bestimmten Paketen zur Übertragung bereit. Wie diese Pakete gestaltet sind und wie die zeitliche

Bereitstellung erfolgt ist Teil der speziellen Publisher Technologie. Der Subscriber kann sich gezielt für den Bezug bestimmter Pakete anmelden. Der Subscriber kennt die Bedeutung der übertragenen Daten, im Allgemeinen kennt er die DataSource jedoch nicht.

6 Komponentenstrukturen

Für das Übertragungssystem in Com4.0-Basic besteht das Netzwerk aus Knoten und Kanten. Jeder Knoten ist ein Device (real oder virtuell) das eine lokale Umgebungen zur Verfügung stellt. In diesen Devices sind sämtliche Funktionen zur Realisierung und Verwaltung der Interaktionsmuster und des Übertragungssystems dezentral realisiert. Com4.0-Basic macht keinerlei Annahmen darüber wie diese Realisierung und das Zusammenspiel mit den lokalen Anwendungen organisiert ist.

6.1 Die TSR Infrastrukturkomponente

In diesem Kapitel wird ein Referenzmodell zur Einbettung des Com4.0-Basic Systems in eine lokale Komponentenstruktur beschrieben. Das Grundprinzip ist in Bild 6.1 dargestellt.

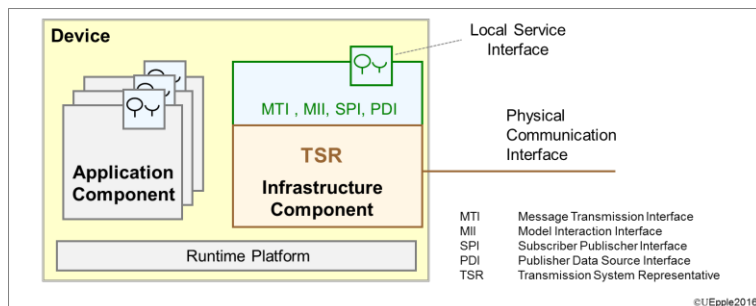


Abbildung 6.1 Einbettung der Kommunikationsfunktionalität in eine lokale Komponentenstruktur

In jedem Device gibt es neben den Anwendungskomponenten nun auch eine Infrastrukturkomponente die das Übertragungssystem lokal repräsentiert. Diese bezeichnen wir als TSR (Transmission System Representatives). Die Infrastrukturkomponente hat einen standardisierten lokalen Namen und stellt eine standardisierte Dienstschnittstelle zur Verfügung über die diese ihren Informationsaustausch abwickeln können. Die bereitgestellten Dienste dienen z.B.

- der Auslösung einzelner Übertragungsvorgänge
- der Vereinbarung von Nutzungsverträgen (Übertragungsleistung, QoS, Security),
- der Erkundung von möglichen Übertragungsangeboten, Partnern usw.,
- der Abfrage aktueller Zustände und Leistungsparameter,
- der Anmeldung von Komponenten zur Teilnahme am Übertragungssystem ,

Im Netzwerk bilden die TSR untereinander ein Cluster das die Funktionalität des Übertragungssystems realisiert.

6.2 Dynamische Einbindung neuer Devices und Komponenten

Durch die offene interne Komponentenschnittstelle ergibt sich, wie in Bild 6.2 dargestellt, die Möglichkeit Komponenten dynamisch im Betrieb an das Übertragungssystem anzubinden.

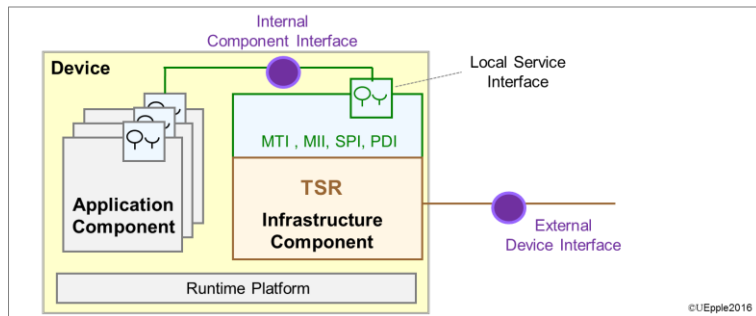


Abbildung 6.2 Die beiden Standard-Interfaces zur Anbindung eines Devices oder einer Anwendungskomponente an das Übertragungssystem.

Damit kann die dynamische Einbindung neuer Funktionalitäten in das Netzwerk auf zwei unterschiedliche Weisen erfolgen:

1. Anbindung des gesamten Devices direkt an das Bussystem (plug and play).
In diesem Fall muss das Device intern nicht komponentenorientiert strukturiert sein. Es kann als Black-Box seine Funktionalität verbergen. Es muss aber mindestens die Funktionalität mitbringen, die von jedem Mitglied des Übertragungssystems gefordert werden. Dazu gehören die operative Unterstützung der Übertragungsfunktionen und die Konformität mit den Vertrags-, QoS- und Sicherheitsmodellen. Darüber hinaus muss seine Realisierung zu den im System festgelegte Technologien und Realisierungsmodellen passen.
2. Anbindung einer Komponente innerhalb eines Devices (App).
In diesem Fall ist das Device bereits in das Übertragungssystem eingebunden. Aufgrund der komponentenorientierten internen Struktur kann eine Komponente nun intern an das Übertragungssystem angeschlossen werden. Dazu muss die Komponente die festgelegten Dienste kennen und sich beim Übertragungssystem entsprechend anmelden. Dies ermöglicht eine APP-artige Handhabung der Komponenten in einem Device. Neben der TSR Infrastrukturkomponente muss das Device eine für die Einbettung geeignete Laufzeitumgebung bereitstellen.

7 Security Aspekte

Die dargestellten Muster unterstützen die Gliederung der mit dem Informationsaustausch verbundene Security Aufgaben. In direktem Zusammenhang mit dem Austausch ergeben sich folgende Security-relevante Aspekte:

7.1 Allgemein

1. Integrität des Übertragungssystems

Security-relevante Aspekte:

- a. Die Verhinderung der Übertragung von unbekannten Servicetypen
- b. Die Bewahrung der Integrität der übertragenen Daten .
Die Daten müssen unverfälscht am Empfänger ankommen.
- c. Die Vertraulichkeit der übertragenen Daten.
Prinzipieller Schutz vor einem Mitlesen der Daten durch Dritte.
Schutz davor, dass eventuell mitgelesene Daten interpretiert werden können.
- d. Die Vertraulichkeit der Verbindung.
Prinzipieller Schutz der Information wann wer mit wem kommuniziert.
(insbesondere beim Datenaustausch zwischen Geschäftspartnern interessant).

2. Integrität der Interaktionsservices

Security-relevante Aspekte:

- a. Die Sicherstellung der Integrität der eigenen Funktionalität.
- b. Die Prüfung der korrekten Identität des Absenders.
- c. Die Prüfung der Existenz des Empfängers
- d. Die korrekte Vertragsgestaltung mit dem Übertragungssystem.
- e. Der vertragsgemäßen Nutzung des Übertragungssystems durch den Absender.
z.B. die Einhaltung vereinbarter Lastbegrenzungen (Länge der Nachrichten, maximale Senderate..) .
- f. Die korrekte und unverfälschte Zustellung an die Empfängeranwendung.

7.2 Für komponentenorientierte Devices

3. Integrität des Laufzeitsystems

Security-relevante Aspekte:

- a. Die Sicherstellung der Ausführung der Anwendungen und Systemdienste in den vereinbarten Echtzeitfenstern.
- b. Die Unabhängigkeit und Unverletzlichkeit der Komponenten.
- c. Die Verhinderung der Aktivierung nicht vertragsgemäßer Schnittstellen.
- d. Die Verhinderung der Aktivierung nicht vertragsgemäßer Komponenten.

4. Integrität der Anwendung

Security-relevante Aspekte:

- a. Die Ausführung innerhalb der vereinbarten maximalen Ausführungszeit.
- b. und Systemdienste in den vereinbarten Echtzeitfenstern.
- c. Die Unabhängigkeit und Unverletzlichkeit der Komponenten.
- d. Die Verhinderung der Aktivierung nicht vertragsgemäßer Schnittstellen.
- e. Die Verhinderung der Aktivierung nicht vertragsgemäßer Komponenten.

8 Literaturverzeichnis

- [1] N.N.: Interaktionsmodell für Industrie 4.0 Komponenten. Plattform Industrie 4.0, UAG Sematik,

http://www.plattformi40.de/I40/Redaktion/DE/Downloads/Publikation/interaktionsmodell-i40-komponenten-it-gipfel.pdf?__blob=publicationFile&v=6_

Abgegriffen: 01.12.2016.

- [2] N.N.: Informationsschnittstellen in der Prozessautomatisierung – betriebliche Eigenschaften. NAMUR Empfehlung NE139, NAMUR, 2012.