

Vertikale Modellintegration in Rahmenwerken – Evaluation von Integrationsansätzen und Untersuchung der Implementierbarkeit anhand eines Fallbeispiels

Stephan Bögel, Werner Esswein

Technische Universität Dresden

Lehrstuhl für Wirtschaftsinformatik, insb. Systementwicklung

stephan.boegel@tu-dresden.de; werner.esswein@tu-dresden.de

Abstract: Die modellgestützte Simulation mobiler Baumaschinen wird in dem vom BMBF geförderten Projekt INPROVY untersucht. Zur Problemstrukturierung wurde das INPROVY-Rahmenwerk vorgestellt, das das Simulationsproblem in unterschiedliche Modellebenen aufteilt. Die vertikale Integration der verschiedenen Modellebenen ist Gegenstand dieses Artikels. Es wird eine Evaluation nach der Methode des merkmalsbasierten Vergleichs der Integrationsansätze durchgeführt, die den Stand der Forschung repräsentieren. Die daraus gewonnen Erkenntnisse werden im Rahmen eines Fallbeispiels auf ihre Implementierbarkeit hin geprüft.

1 Einleitung und Motivation

In vielen technischen Disziplinen ist die Simulation von Maschinen oder Abläufen ein wichtiges Mittel um Erkenntnisse zu gewinnen und Kosten zu sparen. Häufig erfolgt die Simulationserstellung am Einzelfall orientiert und wenig systematisch. Um eine systematischere Vorgehensweise zu etablieren, wird die modellgestützte Erstellung von Simulationsmodellen mobiler Arbeitsmaschinen im Rahmen des vom BMBF geförderten Forschungsprojektes INPROVY¹ untersucht.

Da mobile Baumaschinen, wie Radlader oder Betonspritzpumpen, aus Komponenten unterschiedlicher Zulieferer bestehen, die in die Geschäftsprozesse der beteiligten Unternehmen integriert sind, ist die systematische Erstellung wiederverwendbarer Simulationsmodelle eine komplexe Aufgabe. Zur Komplexitätsreduktion wurde daher ein Rahmenwerk erstellt, welches das Problem in handhabbare Betrachtungseinheiten unterteilt [EGL09].

Das Rahmenwerk besteht u.a. aus den Ebenen [EGL09]:

- Fachebene: Beschreibt die Struktur der mobilen Baumaschine aus der Perspektive der Geschäftsprozesse. Hier findet sich beispielsweise das Konzept „Produktkom-

¹„Integrative Produktentwicklung mit virtuellen Prototypen“ (INPROVY), Gefördert im Rahmenkonzept „Forschung für die Produktion von morgen“ <http://www.inprovoy.de>

ponente“. Diese ist durch eine Artikelnummer charakterisiert, die in Bestell- oder Lagerprozessen genutzt wird.

- Simulationsebene: Betrachtet die problemorientierte Aufarbeitung der Maschinensimulation. Hier findet sich beispielsweise das Konzept „Simulationskomponente“.

Um eine Änderung an einer Ebene an alle betroffenen Ebenen zu propagieren und Inkonsistenzen zwischen den Ebenen zu erkennen, müssen die Ebenen integriert werden.

Insbesondere die Abstimmung der Fach- und der Simulationsebene ist für die Simulation entscheidend. Nur wenn das Unternehmensmodell mit den Simulationsmodellen integriert wird können Informationen aus den Geschäftsprozessen zur Simulationserstellung genutzt und die Simulationsergebnisse schnell in die Geschäftsprozesse integriert werden. Hierbei handelt es sich um eine *vertikale* Integration von Rahmenwerksebenen. Eine durchgängige, modellgestützte Nutzung und Erstellung von Simulationsmodellen erfordert Lösungsansätze für die vertikale Integration der Ebenen.

Damit diese Ziele erreicht werden können, müssen die Abhängigkeiten zwischen den einzelnen Betrachtungsebenen beschrieben – modelliert – werden [ARW08]. Es stellt sich die Frage, *wie* diese Abhängigkeiten modelliert werden können.

Im Rahmen der vorliegenden Arbeit wird eine Evaluation von Modellintegrationsansätzen durchgeführt, die den Stand der Forschung repräsentieren. Neben der Bewertung dieser Ansätze werden wesentliche Gestaltungsdimensionen identifiziert, die notwendig sind, um einen Ansatz auf das Problem der vertikalen Integration in Rahmenwerken anzupassen.

Sowohl der Betrachtungsgegenstand der Evaluation als auch das Ziel der vorliegenden Arbeit legen eine konstruktivistische Perspektive nahe. Diese Forschung wird dem Design Science Ansatz zugeordnet [HMPR04]. Angelehnt an die Forschungsmethodik von [VH05] wird in Kapitel 2 das Problem theoriegeleitet untersucht. Anschließend erfolgt die Anforderungsanalyse, die Formulierung des Evaluationsrahmens sowie die Durchführung der Evaluation in Kapitel 3. Das Fallbeispiel wird in Kapitel 4 untersucht und in Kapitel 5 wird weiterer Forschungsbedarf aufgezeigt.

2 Theorie der vertikalen Modellintegration

Das Problem der vertikalen Modellintegration von Ebenen in Rahmenwerken wird im Folgenden anhand der Integrationsdimensionen Integrationsrichtung (vertikal vs. horizontal), Integrationsart (verbindend vs. vereinigend) und Integrationsansatz (Transformation vs. Zuordnung) eingegrenzt. Die Eingrenzung des Problems erfolgt durch Betrachtung verschiedener Merkmale des Integrationsbegriffs und den Ausschluss bestimmter Merkmalsausprägungen aus der Betrachtung. Eine Übersicht ist in Tabelle 1 gegeben.

Integrationsmerkmal	Merkmalsausprägungen		
	Verbindung		Vereinigung
Integrationsrichtung	vertikal	horizontal	hybrid
Integrationsrealisierung	Organisationsebene	Technikebene	Modellebene
Integrationsgegenstand	Daten, Prozesse, Funktionen, Objekte, Modelle, Methoden		
Integrationsreichweite	Aufgabe	Individuum	Organisationseinheit
	innerbetrieblich	zwischenbetrieblich	überbetrieblich
Automatisierbarkeit	automatisch	semi-automatisch	manuell
Sprachdimension	Syntax	Semantik	Pragmatik
Integrationsansatz	Transformation		Zuordnung

Tabelle 1: Übersicht Integrationsmerkmale und Ausprägungen

2.1 Vertikale vs. Horizontale Modellintegration

Als mögliche *Integrationsrichtungen* werden die vertikale und die horizontale Modellintegration unterschieden [Ros02]. In Anlehnung an das generische Rahmenwerk von SINZ [Sin99] sind in Abbildung 1 Ausschnitte aus zwei exemplarischen Rahmenwerken dargestellt. Diese werden von zwei unterschiedlichen Unternehmen eingesetzt.

In Rahmenwerk A benutzen Unternehmen 1 und Unternehmen 2 den gleichen Architekturrahmen mit identischen Modellierungssprachen. In Rahmenwerk B verwenden die beiden Unternehmen unterschiedliche Modellierungssprachen. Dies resultiert in unterschiedlichen Metamodellen $MM_{k,x}$ innerhalb einer Ebene.

Weiterhin unterscheiden sich zwei Modelle $M_{i,1}$ und $M_{j,1}$, die auf zwei unterschiedlichen Ebenen liegen, im Betrachtungswinkel [Sin99]. Da die Modelle dasselbe Unternehmen beschreiben, beziehen sie sich auf denselben Betrachtungsgegenstand. Zwei Modelle $M_{i,1}$ und $M_{i,2}$ aus zwei unterschiedlichen Unternehmen oder Organisationseinheiten, die aber auf der gleichen Ebene liegen, unterscheiden sich dann im Betrachtungsgegenstand, nicht aber im Betrachtungswinkel.

Das Szenario einer horizontalen Integration tritt beispielsweise auf, wenn die beiden Unternehmen 1 und 2 fusionieren oder eine enge Kooperation anstreben (etwa mit dem Ziel

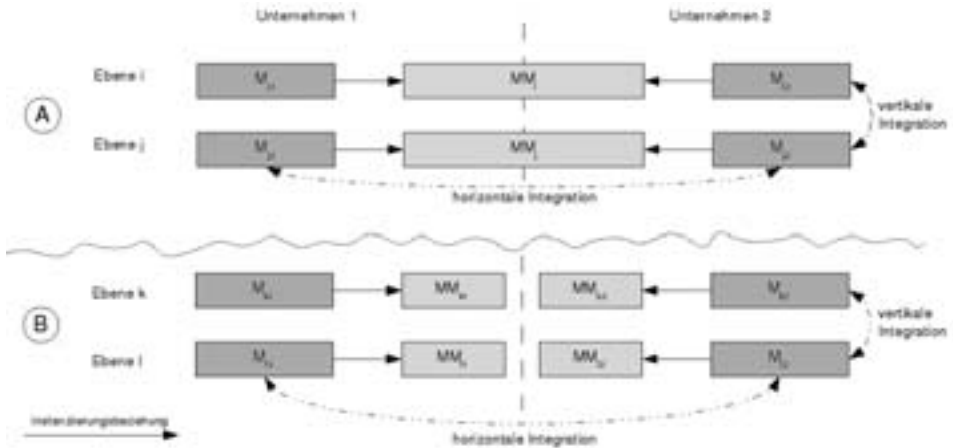


Abbildung 1: Vertikale vs. horizontale Modellintegration in Rahmenwerken

einer integrierten Versorgungskette). Eine solche Integration umfasst beispielsweise die Modelle $M_{i,1}$ und $M_{i,2}$ in Rahmenwerk A. Verwenden beide Unternehmen unterschiedliche Modellierungssprachen, wie in Rahmenwerk B angedeutet, so müssen diese bei der Integration berücksichtigt werden. Die Überwindung von Sprachgrenzen ist also bei der vertikalen Integration immer zu berücksichtigen, bei der horizontalen Integration nur im Fall B. Im Rahmen von INPROVY wird ein Rahmenwerk zur modellgestützten Simulation entwickelt, das die verwendeten Modellierungssprachen vorschreibt. Es liegt hier daher Fall A vor.

2.2 Verbindende vs. Vereinigende Modellintegration

Bezüglich der *Integrationsart* kann eine Integration durch verbinden oder vereinigen erfolgen. Bei einer Integration durch Verbindung werden vorher unverbundene Elemente, die aber logische Beziehungen zueinander aufweisen, zu einem System verbunden [Ros02]. Bei einer Integration durch Vereinigung geht ein neues, zusammengeführtes Modell hervor, welches das alte Modell ersetzt [Ros02].

In Bezug auf die vertikale Integration von Rahmenwerkebenen widerspricht eine Integration durch Vereinigen dem Zweck der Komplexitätsreduktion durch die Aufteilung in einzelne Ebenen. Die vertikale Integration setzt daher eine Integration durch Verbinden voraus.

2.3 Transformation vs. Zuordnung

Die Integration von Modellebenen kann durch Zuordnung oder Transformation erfolgen [Sin99]. Ein Beispiel für einen transformationsorientierten Ansatz ist die Model Driven Architecture (im Folgenden MDA) [MM03]. FRANKEL ET AL. zeigen, wie sich die MDA zur Integration von Modellebenen in dem Rahmenwerk von ZACHMAN [Zac87] einsetzen lässt [FHM⁺03]. Jede Rahmenwerksebene wird einem MDA-Modelltyp zugeordnet, zwischen denen eine top-down Transformationsbeziehung besteht.

Eine solche Transformationsbeziehung besteht in dem INPROVY-Rahmenwerk im Sinne eines Vorgehensmodells, bei dem die Ebenen von oben nach unten durchlaufen werden [EGL09]. Dieses Vorgehen ist sinnvoll bei der (initialen) Erstellung von Simulationsmodellen, stößt aber bei der Rückführung von Ergebnissen von unteren Ebenen in die höherliegenden Ebenen an seine Grenzen.

Bei den hier betrachteten semi-formalen Modellen kann nicht davon ausgegangen werden, dass diese automatisch neu generiert werden können [Geh07]. Eine manuelle Transformation ist aufgrund des Aufwandes nicht nach jeder Änderung möglich. Im Sinne eines *round-trip model engineering* müssen Wege gefunden werden die Modellebenen so zu integrieren, dass eine Änderung an einer Ebene nicht dazu führt, dass die darunter liegenden Ebenen komplett neu generiert werden müssen [SK04].

In diesem Fall ist eine Integration durch Zuordnung sinnvoll. Zuordnungen sind darüber hinaus Voraussetzung für Transformationen, bei denen Transformationsregeln auf Basis der Metamodellelemente beschrieben werden [MM03]. Es wird daher die Integration durch Zuordnungen untersucht.

2.4 Konflikte bei der Modellintegration

Die Schwierigkeit bei der Integration der Modellebenen besteht in der Überwindung von Integrationskonflikten. Ursprünglich aus der Schemaintegration von Datenbanken entstanden [BLN86] und dann auf die konzeptuelle Modellierung übertragen [Ros02], werden Namens-, Struktur- und Typkonflikte unterschieden.

Ein Typkonflikt liegt vor, wenn derselbe Sachverhalt in zwei Modellen durch unterschiedliche Sprachkonstrukte modelliert wird [Ros02]. Es liegt damit eine unterschiedliche Verwendung der Modellierungsgrammatiken vor. Beispielsweise kann in einem Entity-Relationship-Modell ein Sachverhalt als Entity-Typ oder als Attribut modelliert werden. Ein spezieller Typkonflikt liegt vor, wenn Wertebereiche unterschiedlich verwendet werden. In diesem Fall wird von einem Wertebereichskonflikt gesprochen [Geh07].

Wird der gleiche Sachverhalt in einem Modell (semantisch) unterschiedlich beschrieben, so wird dies als Strukturkonflikt bezeichnet. Es wird zwischen Abhängigkeits-, Abstraktions- und Detaillierungskonflikten differenziert [Geh07].

Ein Abhängigkeitskonflikt liegt dann vor, wenn gleiche Strukturen durch sich widersprechende Beziehungsarten modelliert werden. Beispiele hierfür sind unterschiedliche Mul-

tiplizitäten oder eine unterschiedliche Modellierung von Generalisierungs-/Spezialisierungsbeziehungen [Geh07]. Zu einem Abstraktionskonflikt kommt es, wenn der gleiche Sachverhalt unterschiedlich abstrakt bzw. generalisiert modelliert wird [Geh07]. Beispielsweise könnte in einem Modell der Sachverhalt als Buch und in einem anderen Modell als Publikation bezeichnet werden. Ein Detaillierungskonflikt dagegen tritt auf, wenn ein Realitätsausschnitt in unterschiedlicher Ausführlichkeit dargestellt wird [Geh07].

Bei einer unterschiedlichen Verwendung der Fachsprache wird ein Konflikt als Namenskonflikt bezeichnet. Es lassen sich Synonymkonflikte (unterschiedliche Wörter werden gleich verwendet) und Homonymkonflikte (gleiche Wörter werden unterschiedlich verwendet) unterscheiden [Geh07].

Da die beschriebenen Konflikte aus der Schemaentwicklung von Datenbanken bzw. der Prozessmodellintegration stammen, wurden diese bisher nur bei der horizontalen Integration identifiziert. Es stellt sich damit die Frage, ob diese Konflikte auch bei der vertikalen Integration auftreten.

In dem Fallbeispiel in Kapitel 4 wird gezeigt, dass die beschriebenen Konflikte auch bei der vertikalen Integration auftreten.

3 Evaluation

Für die Evaluation wird die Methode des merkmalsbasierten Vergleichs verwendet. Hierzu wird eine Liste von idealen Eigenschaften eines Integrationsansatzes aufgestellt, anhand derer die Ansätze evaluiert werden [SR98].

Problematisch an dieser Methode ist ein hohes Maß an Subjektivität. Zur intersubjektiven Nachvollziehbarkeit der Ergebnisse muss sowohl die Auswahl als auch die Interpretation ausreichend dokumentiert werden [SR98].

3.1 Evaluationsrahmen

Das Vorgehen bei der Erstellung des Evaluationsrahmens orientiert sich an den von HEINRICH vorgeschlagenen Schritten [Hei00]. Die Evaluationsobjekte wurden durch eine systematische Literaturrecherche gefunden [Die08]. Als Filter wird Tabelle 1 herangezogen. Um die Anforderung der Evaluationsmethode nach einer guten Dokumentation der Messung erfüllen zu können, müssen die betrachteten Ansätze selbst ausreichend dokumentiert sein [SR98]. Es werden daher nur Ansätze untersucht, die zwei oder mehr Publikationen aufweisen.

In Tabelle 2 sind die Evaluationsmerkmale zusammen mit den verwendeten Metriken aufgeführt.

Im Folgenden werden die Merkmale kurz erläutert. Zentrale Anforderung an einen Ansatz zur Modellintegration ist die *Überwindung der Integrationskonflikte*. Nur wenn diese

Kriterium		Metrik
A1	Überwindung der Integrationskonflikte	
	1	Überwindung Typkonflikt
	2	Überwindung Strukturkonflikte
	3	Überwindung Namenskonflikte
A2	Werkzeugunterstützung (Automatisierbarkeit)	Automatisch, halb automatisch, manuell
A3	Implementierungsaufwand	Gering, mittel, hoch
A4	Unterstützung der	
	1	Wiederverwendbarkeit
	2	Anpassbarkeit
	der Metamodelle	

Tabelle 2: Evaluationsmerkmale und Metriken

Konflikte überwunden wurden kann von einem integrierten System gesprochen werden. Wird ein Konflikt sowohl auf Ebene der Syntax als auch der Semantik überwunden, so gilt die Anforderung als vollständig erfüllt. Wird ein Konflikt nur auf Ebene der Syntax überwunden, so wird die Anforderung als teilweise erfüllt bewertet. Typkonflikte können schon auf Ebene der abstrakten Syntax aufgelöst werden [Geh07]. Die Anforderung ist damit erfüllt, wenn der Typkonflikt auf Ebene der Syntax oder der Semantik gelöst wird.

Aufgrund der enormen Anzahl können bei komplexen Modellen nicht alle Beziehungen auf *Modellebene* manuell erstellt werden. Es müssen also Ansätze gefunden werden, die eine automatische oder halb-automatische Definition der Zuordnungen auf Modellebene erlauben [WBMW09]. Es wird daher bewertet, inwieweit ein Ansatz eine *Automatisierbarkeit* der Verbindungen auf Modellebene zulässt. Dieses Merkmal bezieht sich damit auf das Integrationsziel Aktualität [Ros02].

Bei der Integration müssen die Autonomiekosten dem Integrationsaufwand gegenübergestellt werden [Ros02]. Der *Aufwand*, der für die Implementierung eines Ansatzes erforderlich ist, findet daher Eingang in die Bewertung, auch wenn Kosten und Nutzen hier nicht quantitativ bewertet werden können. Einfluss haben alle Voraussetzungen, die für die Implementierung eines Ansatzes erfüllt werden müssen. Es wird hierzu geprüft, ob zusätzliche Artefakte, wie Beziehungs-Metamodelle oder Ontologien, erstellt oder bestimmte Maßnahmen durchgeführt werden müssen.

Zentrale Qualitätsmerkmale von Rahmenwerken sind *Wiederverwendbarkeit* und *Anpassbarkeit* des Rahmenwerks [Sin99]. Die Integration soll den Vorteil einer Aufteilung allerdings nicht dadurch zunichte machen, das die Teile eng gekoppelt werden. Die Anforderung Wiederverwendbarkeit und Anpassbarkeit bezieht sich daher auf die *Metamodelle*. Diese sollten, zumindest in ähnlichen Projekten, wiederverwendet werden können und an-

passbar sein. Müssen die Metamodelle verändert werden, damit eine Integration stattfinden kann, so wird die Anforderung mit „schränkt ein“ bewertet.

3.2 Ansätze der Modellintegration – Stand der Forschung

Im Folgenden wird der Stand der Forschung zu Modellintegrationsansätzen dargestellt und diese evaluiert. Die Ansätze lassen sich zunächst in Metamodell-basierte (syntaktische) und Ontologie-basierte (semantische) Integrationsansätze aufteilen [ABH⁺06].

Folgende Ansätze wurden durch die Literaturrecherche gefunden, aber aus oben erläuterten Gründen nicht in die Evaluation aufgenommen:

- FELLMANN ET AL. untersuchen Modellbeziehungen mit semantischen Wikis. Die Beziehungen werden allerdings nicht auf Ebene der Modellelemente betrachtet, sondern die Modelle als geschlossene Artefakte. Der Ansatz wird nur durch eine Publikation beschrieben [FT09].
- SIMON ET AL. beschreiben einen horizontalen Ansatz zur Integration von Prozessmodellen. Im Fokus steht allerdings die vereinigende Integration der Modelle [SM07].
- FENGEL ET AL. stellen einen Ansatz zur semantischen Verlinkung heterogener Modelle vor. Hierzu werden die Modellelemente der zu integrierenden Modelle mit Schlagworten versehen. Der Ansatz wird im Rahmen eines Forschungsprojektes verfolgt und ist noch nicht durch eine ausreichende Anzahl an Publikationen beschrieben [FRN08].
- HAHN untersucht die Integration verteilter Produktmodelle mit Hilfe von Semantic-Web-Technologien. Die Dokumentation des Ansatzes ist für die Evaluation ebenfalls nicht ausreichend [Hah05].

3.2.1 Metamodell-basierte Integrationsansätze

Metamodell-basierte Integrationsansätze ermöglichen eine Integration auf Ebene der abstrakten Syntax.

Multiperspektivische Unternehmensmodellierung (MEMO) Der Ansatz behandelt die vertikale Integration der Ebenen „Strategie“, „Organisation“ und „Informationssystem“ [Fra02]. Die Integration der Sprachen erfolgt über gemeinsame Konzepte der Metamodelle, d.h. ein Konzept wird in mehreren Metamodellen verwendet [Fra94]. Bei der MEMO SML wird beispielsweise das Konzept Organisationseinheit aus der MEMO OrgML in Zusammenhang mit den Konzepten „StrategicBusUnit“ und „HumanRessource“ verwendet [Fra02].

Es handelt sich folglich um eine verbindende Integration durch Zuordnung. Im Kern entspricht dieser Ansatz der Integration von Sichten innerhalb eines Modells. In der Konsequenz entsteht eine „Super“-Sprache, die sich über alle Modellebenen erstreckt.

Enterprise Model Integration (EMI) Zentrale Konzepte des EMI Ansatzes sind Mappings und Integrationsregeln [KBJK03]. Mappings beschreiben, welche Teile der Metamodelle integriert werden. Integrationsregeln definieren, wie diese Mappings umgesetzt werden [ZS07].

Mappings stellen eine Integration über Zuordnungen her. Integrationsregeln können entweder eine verbindende Integration (durch Angleichungsregeln) oder eine vereinigende Integration (durch Verbindungsregeln) realisieren. Integrationsmodelle enthalten die Mappings zwischen den Sprachkonzepten der zu integrierenden Modelle [ZS07]. Dies entspricht den von SINZ beschriebenen Beziehungs-Metamodellen [Sin99].

3.2.2 Ontologie-basierte Integrationsansätze

Ontologie-basiert Ansätze streben eine Integration auf der Ebene der Semantik an.

Integration über natürlichsprachliche Ausdrücke nach HÖFFERER Der Ansatz von HÖFFERER ordnet natürlichsprachlichen Ausdrücken, die ein Modellelement enthält, Konstrukten einer Ontologie zu [HÖ7]. HÖFFERER betrachtet die horizontale Integration von Modellen, wie sie bei der Integration von Prozessen unterschiedlicher Unternehmen auftritt [KH08].

Integration über semantische Annotationen nach LIN Ebenso wie bei HÖFFERER wird bei LIN die horizontale Integration von Prozessmodellen betrachtet [LSH⁺06]. Im Unterschied zu HÖFFERER werden allerdings dem *Modellelement* Konzepte aus der Ontologie zugeordnet und nicht den natürlichsprachlichen Ausdrücken, die das Modellelement beschreiben [LK09].

3.3 Evaluationsergebnis

Tabelle 3 fasst das Evaluationsergebnis zusammen.

		Metamodell-basiert		Ontologie-basiert	
		MEMO	EMI	HÖFFERER	LIN
Int. Richtung		vertikal	vertikal/horizontal	horizontal	horizontal
A1	1	+	+	+	+
	2	-	o	+	+
	3	o	-	+	+
A2		-	o	o	-
A3		+	o	-	-
A4	1	-	+	+	+
	2	-	+	+	+

Legende
 + (A1: erfüllt / A2: automatisch / A3: gering / A4: unterstützt)
 o (A1: teilweise erfüllt / A2: halb automatisch / A3: mittel)
 - (A1: nicht erfüllt / A2: manuell / A3: hoch / A4: schränkein)

Tabelle 3: Evaluationsergebnis

3.3.1 Vergleichende Bewertung der Ansätze (Evaluationsziel 1)

Ein Vergleich der Integrationsrichtung zeigt, dass unter den untersuchten Ansätzen nur die Metamodell-basierten zur vertikalen Integration eingesetzt werden. Die Ontologie-basierten werden bisher nicht zur Integration in vertikaler Richtung eingesetzt. Hier besteht eine Forschungslücke, die in dem Fallbeispiel in Kapitel 4 näher untersucht wird.

Mit allen Ansätzen ist es möglich, Typkonflikte zu überwinden. Strukturkonflikte lassen sich mit den ontologiebasierten Ansätzen vollständig überwinden. Mit dem EMI-Ansatz ist durch die verfeinerten Beziehungstypen eine eingeschränkte Explikation von Strukturkonflikten auf der Ebene der abstrakten Syntax möglich [ZS07]. Auf Ebene der Semantik ist dies nicht möglich, da bei materialen Domänen die Semantik nicht auf die Syntax zurückgeführt werden kann [Geh07].

Namenskonflikte können vollständig nur mit Hilfe der Ontologie-basierten Ansätze überwunden werden. Mit dem MEMO-Ansatz können diese teilweise vermieden werden, indem fachsprachliche Ausdrücke in das Metamodell übernommen werden [Fra02]. Ein einheitliches Verständnis der übernommenen fachsprachlichen Ausdrücke ist allerdings nicht sichergestellt.

Die Anforderung Automatisierbarkeit wird von keinem Ansatz vollständig erfüllt. EMI und der Ansatz nach HÖFFERER erlauben, durch ihre Teil-Automatisierbarkeit, eine Unterstützung des Modellerstellers durch ein Werkzeug. Die Ansätze nach LIN und MEMO erlauben nur eine manuelle Integration und sind für die Integration komplexer Rahmenwerke nicht geeignet.

Den geringsten Implementierungsaufwand weist der Integrationsansatz MEMO auf. Die Ontologie-basierten Ansätze erfordern den höchsten Aufwand, allerdings erfüllen sie auch die meisten Anforderungen.

Der MEMO-Ansatz stellt bezüglich der Wiederverwendbarkeit den schlechtesten Ansatz dar, da die Modellierungssprachen der unterschiedlichen Ebenen nicht getrennt voneinander genutzt werden können. Durch die Nutzung von getrennten Integrationsmodellen im EMI-Ansatz kann dieser als wesentlich besser in Bezug auf die Wiederverwendbarkeit angesehen werden. Die Ontologie-basierten Ansätze erfordern keine Veränderung der Metamodellebene und sind daher in Bezug auf die Wiederverwendbarkeit ebenfalls als positiv zu beurteilen. Das gleiche gilt für die Anpassbarkeit der Metamodelle.

In Bezug auf die Integration der INPROVY Rahmenwerksebenen konnten die Anforderungen A1, A2 und A4 als Sachziele bei der Implementierung identifiziert werden. Demnach ist nur der Ansatz nach HÖFFERER für die Lösung der Integrationsproblematik geeignet. Tabelle 3 zeigt allerdings auch, dass eine geschickte Kombination der Eigenschaften der Metamodell-basierten Ansätze die Sachziele erfüllen könnte. Hierzu sind die relevanten Gestaltungsdimensionen zu identifizieren.

3.3.2 Identifizierte Gestaltungsdimensionen (Evaluationsziel 2)

Die vergleichende Betrachtung der Ansätze zeigt die folgenden Gestaltungsdimensionen auf. Bei den Metamodell-basierten Ansätzen besteht Gestaltungsspielraum hinsichtlich der *Detailliertheit des Metamodells*. Ein höheres Maß an Detailliertheit bedeutet, dass die Zeichenklassen, deren Beziehungen im Metamodell beschrieben werden, feiner aufgelöst werden. Je detaillierter das Metamodell ist, desto besser lassen sich auch die Beziehungen zwischen zwei Metamodellen beschreiben.

Ein grobgranulares Metamodell enthält beispielsweise die Konstrukte „Stelle“ und „Organisationseinheit“, wohingegen ein feingranulares Metamodell beispielsweise die Stellentypen „Controller“ und „Finanzbuchhalter“ unterscheidet [Win09].

Ein hohes Maß an Detailliertheit steht allerdings in Konflikt mit der Wiederverwendbarkeit [Fra02]. Werden Begriffe aus der Fachsprache in das Metamodell aufgenommen, so kann nicht davon ausgegangen werden, dass diese in einem anderen Unternehmen gleichermaßen verwendet werden.

Eine weitere Gestaltungsdimension wird von den *Integrationsbeziehungen* aufgespannt. Die Ansätze unterscheiden sich darin, welche Typen von Beziehungen zugelassen werden. Diese Gestaltungsdimension wird auch von WEIDLICH ET AL. als wichtig erachtet [WBMW09]. Integrationsbeziehungen können einfacher oder komplexer Art, wie Generalisierungen oder Aggregationen, sein.

Interessant ist ein Unterschied zwischen den beiden Ontologie-basierten Ansätzen. Komplexe Beziehungen werden entweder nur innerhalb der Ontologie angewendet (HÖFFERER) oder zwischen Ontologie und Modell (LIN). Der Konstruktionsaufwand verschiebt sich damit von der Ontologiekonstruktion hin zur Verbindung von Modell und Ontologie. Auch wenn die Wiederverwendbarkeit der Ontologien nicht untersucht wurde, da diese nicht Bestandteil von Rahmenwerken sind, so ist die erste Möglichkeit wesentlich besser. Die

Beziehungen innerhalb der Ontologie können wiederverwendet werden, die Beziehungen zwischen Modell und Ontologie nicht.

Der Einfluss der beiden Integrationsdimensionen wird in dem nachfolgenden Fallbeispiel untersucht.

4 Fallbeispiel

Das Beispiel stammt aus dem Forschungsprojekt INPROVY und betrachtet eine Betonspritzpumpe (vgl. Abbildung 2). In Abschnitt 4.1 des Fallbeispiels wird der Einfluss der identifizierten Gestaltungsdimensionen untersucht. In Abschnitt 4.2 wird die Anwendung von Ontologien auf die vertikale Integration anhand des Fallbeispiels geprüft.

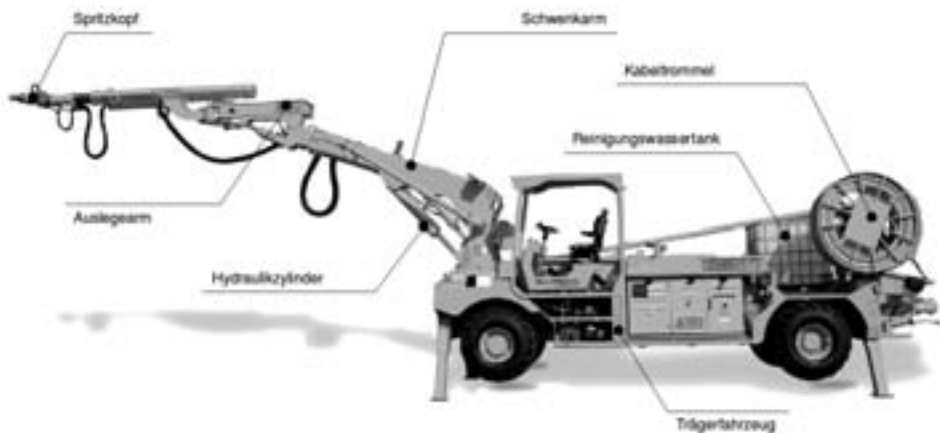


Abbildung 2: Betonspritzpumpe

Auf der Fachebene des INPROVY-Rahmenwerks befindet sich ein konzeptuelles Produktmodell, das die Produktkomponenten dieser Betonspritzpumpe darstellt. Eine Produktkomponente ist etwa dadurch charakterisiert, dass sie eine Artikelnummer besitzt und mit Hilfe dieser Nummer von einem Zulieferer bestellt oder im Lager aufgefunden werden kann. Die Beschreibung der Produktkomponenten ist eine statische Sicht auf die Baumaschine, auf die in einer Prozesssicht referenziert werden kann, um etwa einen Bestellprozess zu beschreiben.

Auf der konzeptuellen Simulationsebene werden die Simulationskomponenten der Betonspritzpumpe beschrieben, die den Ausgangspunkt für verschiedene Simulationsanwendungen darstellen. Beispiele für solche Anwendungen sind die Echtzeitsimulation oder die Detailsimulation einzelner Komponenten zur Vorausentwicklung [EGL09]. Dieses konzeptuelle Simulationsmodell ist ebenfalls eine statische Sicht auf die Betonspritzpumpe, allerdings aus einem anderen Blickwinkel.

In dem Produktmodell werden Komponenten wie die Kabeltrommel oder der Reinigungswassertank, aufgrund verschiedener Artikelnummern, einzeln abgebildet. Im konzeptuellen Simulationsmodell sind, wenn der Schwerpunkt der Simulation etwa auf dem Spritzarm liegt, die Kabeltrommel und der Reinigungswassertank zu dem Modellelement „Trägerfahrzeug“ zusammengefasst.

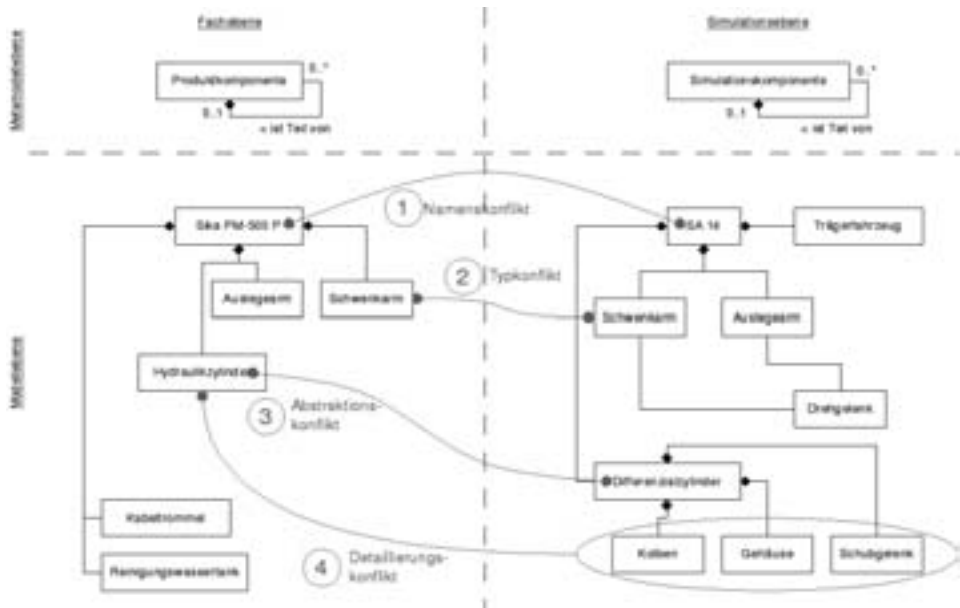


Abbildung 3: Modelle des Fallbeispiels

In Abbildung 3 ist ein Beispielmodell der Fachebene sowie ein Beispielmodell der Simulationsebene dargestellt. Eine Betonspritzpumpe besteht aus mehreren hundert Teilen. Beide Modelle stellen daher nur einen möglichen Ausschnitt dar. Über den Beispielmodellen sind (ebenfalls stark verkürzt) die Metamodelle der Beispielmodelle dargestellt. Zwischen den beiden Modellen bestehen mehrere Integrationskonflikte, von denen exemplarisch vier dargestellt sind.

1. Namenskonflikt: In dem Fachmodell wird die Betonspritzpumpe mit dem Produktnamen bezeichnet, während in dem Simulationsmodell die firmeninterne Typenbezeichnung verwendet wird.
2. Typkonflikt: Der Schwenkarm ist in dem Fachmodell vom Typ „Produktkomponente“, in dem Simulationsmodell allerdings vom Typ „Simulationskomponente“.
3. Abstraktionskonflikt: Die unterschiedlichen Teile des Spritzarms sind durch Hydraulikelemente verbunden. Im Fachmodell wird ein solches Element als „Hydraulizylinder“ modelliert. Im Simulationsmodell wird dies als „Differenzialzylinder“ modelliert. Letzterer ist allerdings ein spezieller Hydraulizylinder.

4. Detaillierungskonflikt: Da die unterschiedlichen Elemente eines solchen Differentialzylinders für die Simulation wichtig sind, da sie das Verhalten beeinflussen, werden Kolben, Gehäuse und Schubgelenk im Simulationsmodell gesondert dargestellt. Im Fachmodell hingegen wird der Zylinder als ein Element modelliert, da nur das Teil in seiner Gesamtheit eine Artikelnummer besitzt.

4.1 Metamodell-basierte Lösung

Die erste Möglichkeit besteht in der Integration auf Basis der Metamodelle. Hierzu werden die Beziehungen zwischen den Elementen der beiden Metamodelle modelliert. In einem ersten Schritt kann eine Beziehung zwischen dem Element „Produktkomponente“ und dem Element „Simulationskomponente“ modelliert werden. Dies ist in Abbildung 4 dargestellt.

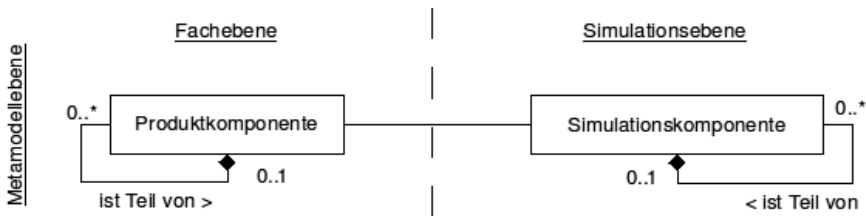


Abbildung 4: Beziehungen zwischen Metamodellen der Fach- und Simulationsebene

Durch die Modellierung einer solchen Beziehung auf Metamodellebene ist es möglich den oben beschriebenen Typkonflikt zu explizieren, indem eine Verbindung auf Modellebene zwischen den beiden Elementen „Produktkomponente:Schwenkarm“ und „Simulationskomponenten:Schwenkarm“ modelliert wird.

Die übrigen Integrationskonflikte gehen über die reine Zuordnung von Metamodellelementen hinaus. Namens- und Strukturkonflikte können hierdurch nicht dargestellt werden. Die vergleichende Evaluation der Integrationsansätze in Abschnitt 3.3.2 hat zwei Gestaltungsdimensionen identifiziert. Zum einen die semantische Verfeinerung der Beziehungen zwischen den Metamodellelementen und zum anderen die Verfeinerung der Zeichenklassen, die in den Metamodellen verwendet werden.

Eine Verfeinerung des Metamodells bedeutet, dass fachsprachliche Begriffe in das Metamodell aufgenommen werden. Die Zeichenklassen, deren Beziehungen im Metamodell beschrieben werden, zu spezialisieren, führt dazu, dass das Metamodell umfangreicher wird. Der Namenskonflikt kann durch das in Abbildung 5 dargestellte Metamodell gelöst werden. Das Beispiel zeigt zugleich den gravierenden Nachteil dieses Ansatzes auf. Eine solche Verfeinerung der Zeichenklassen führt zu einem sehr komplexen Metamodell, dessen Wiederverwendbarkeit stark eingeschränkt ist.

Eine weitere Gestaltungsmöglichkeit besteht in der Verfeinerung der Beziehungen zwischen den Metamodellelementen. In der vergleichenden Evaluation wurde beispielsweise



Abbildung 5: Verfeinerte Beziehungen zwischen Metamodellen der Fach- und Simulationsebene

eine Spezialisierungsbeziehung genannt. Diese könnte auf Modellebene verwendet werden, um den Abstraktionskonflikt zwischen dem Modellelement „Hydraulikzylinder“ und dem Element „Differenzialzylinder“ zu beschreiben. Dies muss für jedes Modell manuell erfolgen und ist bei großen Modellen daher nur unter unverhältnismäßig hohem Aufwand zu bewerkstelligen.

Die Verwendung dieser Spezialisierungsbeziehung auf Metamodellebene würde dagegen ebenfalls ein verfeinertes Metamodell voraussetzen, damit diese Beziehung zwischen der Zeichenklasse „Hydraulikzylinder“ und „Differenzialzylinder“ modelliert werden kann. Folglich entsteht ein ebenso schlecht wiederverwendbares Metamodell.

Mit Hilfe der Metamodell-basierten Ansätze können Typkonflikte expliziert werden. Abstraktions- und Strukturkonflikte können damit nur auf Ebene der abstrakten Syntax überwunden werden. Bei materialen Domänen kann die Semantik nicht auf die Syntax zurückgeführt werden [Geh07]. Damit ist die einheitliche Verwendung von Konstrukten verfeinerter Metamodelle nicht sichergestellt. Trotz dieser Einschränkungen konnte gezeigt werden, dass sich verfeinerte Beziehungen und ein detaillierteres Metamodell gegenseitig bedingen. Bei der Überwindung der Integrationskonflikte sind diese, aufgrund des überschaubaren Implementierungsaufwands, als positiv zu beurteilen. Der Widerspruch zwischen Wiederverwendbarkeit und Nutzbarkeit verfeinerter Metamodelle ist durch weitere Forschung zu adressieren. Denkbar ist die Trennung zwischen einem Kern-Metamodell und Fachsprachen-Metamodell-Erweiterungen [KR98], [KÖ6].

4.2 Ontologie-basierte Lösung

Die beschriebenen Integrationskonflikte könnten auch mit Hilfe von Ontologien expliziert und eine indirekte Integration der Modelle erreicht werden. In Abbildung 6 (oben) sind zwei Beispielontologien dargestellt.

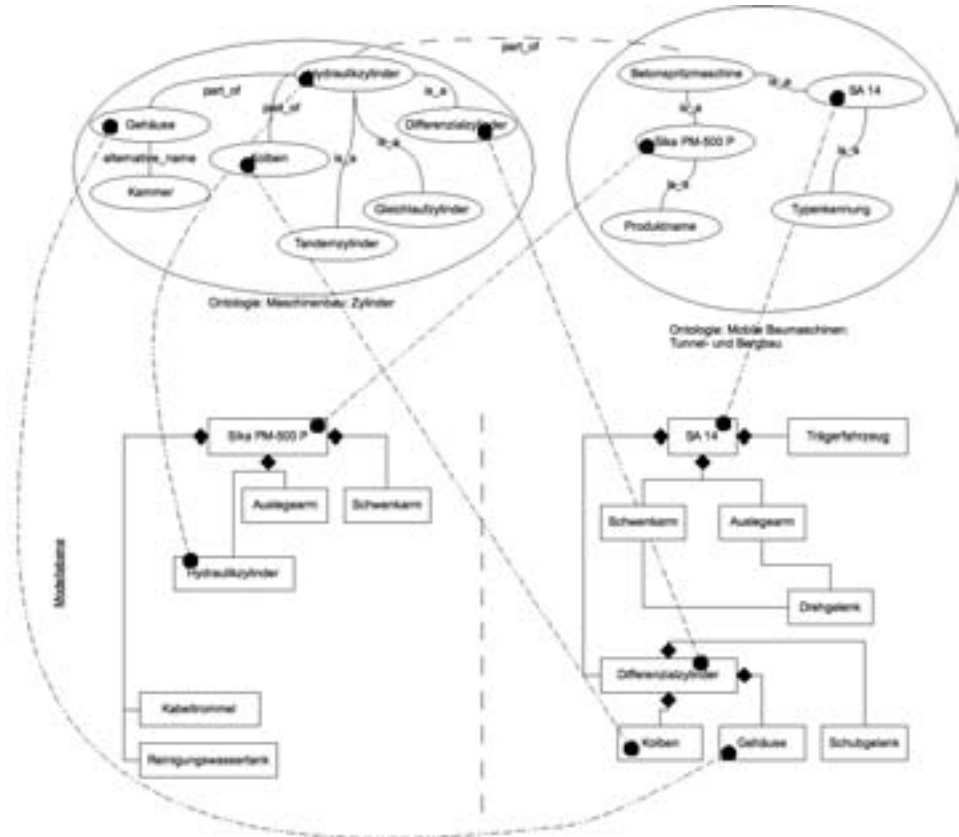


Abbildung 6: Ontologie-basierte Lösung

Die erste Ontologie beschreibt den Zusammenhang zwischen fachsprachlichen Ausdrücken aus dem Bereich Maschinenbau. Die Zweite beschreibt Begriffe aus der Domäne der mobilen Baumaschinen.

Auf Modellebene werden die einzelnen Elemente den Konzepten der Ontologien zugeordnet (in Abbildung 6 als gestrichelte Linie mit Punkten dargestellt). Dies könnte beispielsweise halb-automatisch mit Hilfe von Ähnlichkeitsmaßen durch das Modellierungswerkzeug erfolgen [HÖ7].

Der Namenskonflikt kann dadurch überwunden werden, dass die Ontologien ausgewertet

werden. Beide Begriffe stellen eine Betonspritzpumpe dar und unterscheiden sich dadurch, dass sie ein Produktname bzw. eine Typenkenntung sind. Alternativ wäre es auch möglich eine Synonymbeziehung zwischen „SA 14“ und „Sika PM-500 P“ zu modellieren. Der Abstraktionskonflikt zwischen „Hydraulikzylinder“ und „Differenzialzylinder“ wird dadurch überwunden, dass die beiden Begriffe mit einer Spezialisierungsbeziehung verbunden sind. Der Detaillierungskonflikt wird dadurch expliziert, dass Kolben und Gehäuse bzw. Kammer Teile eines jeden Hydraulikzylinders sind.

In dem Beispiel können alle Konflikte aufgelöst werden. Für die Integration werden jedoch zwei Ontologien benötigt. Diese müssen wiederum integriert werden. In dem Beispiel wird eine Teil-Ganzes-Beziehung zwischen zwei Konzepten manuell modelliert (in Abbildung 6 als grob gestrichelte Linie dargestellt). Bei umfangreichen Ontologien erreichen solche Beziehungen schnell eine große Zahl und verursachen einen hohen Integrationsaufwand. Durch das Beispiel wurde die Anwendbarkeit Ontologie-basierter Ansätze auf die vertikale Integration nachgewiesen.

5 Fazit und weiterer Forschungsbedarf

Die Evaluation hat gezeigt, dass der Ansatz von HÖFFERER als einziger alle Integrationskonflikte überwindet, Wiederverwendbarkeit und Anpassbarkeit unterstützt und eine zumindest halb automatische Integration zulässt. Negativ zu bewerten ist der hohe Implementierungsaufwand.

Darüber hinaus muss das insgesamt positive Abschneiden der Ontologie-basierten Ansätze relativiert werden. Wird davon ausgegangen, dass die Semantik von Zeichen durch ihren Gebrauch in der Sprache zustande kommt, so ist nicht sichergestellt, dass eine Ontologie diesen Gebrauch korrekt und vollständig abbildet [Geh07].

Ein weiteres Problem entsteht durch die Verwendung unterschiedlicher Ontologien. Da auf unterschiedlichen Ebenen des Rahmenwerkes verschiedene Fachsprachen verwendet werden, ist es wenig wahrscheinlich, dass alle Fachausdrücke für alle Modellebenen in einer Ontologie definiert werden können. Alle betrachteten Ansätze nutzen daher auch unterschiedliche Ontologien. Diese Ontologien müssen ihrerseits wieder integriert werden. Hierbei treten ähnliche Probleme auf wie bei der Integration der Modellebenen [VJBS98]. Das Integrationsproblem verschiebt sich daher nur auf eine andere Ebene.

Eine Lösung könnte darin bestehen, die in der Evaluation identifizierten Gestaltungsdimensionen auszuschöpfen. Unter Nutzung von detaillierten Metamodellen und verfeinerten Integrationsbeziehungen werden die Abhängigkeiten zwischen den Modellebenen durch ein Integrationsmodell beschrieben, das von einem Modellierungstool ausgewertet wird.

Die Anreicherung des Metamodells mit fachsprachlichen Ausdrücken erhöht die Komplexität und verringert die Wiederverwendbarkeit. Hierfür muss eine Lösung gefunden werden, welche die Erweiterbarkeit eines Kern-Metamodells sicherstellt.

Es ist eine Integrationsprache zu entwickeln, in der das Integrationsmodell erstellt wird.

Diese steht im Spannungsfeld zwischen den semi-formalen Sprachen, die integriert werden sollen und der Anforderung, durch eine Modellierungswerkzeug auswertbar zu sein.

Weiterhin hat die Evaluation gezeigt, dass die bisher nur bei der horizontalen Integration untersuchten Konflikte auch bei der vertikalen Integration zu überwinden sind.

Allen betrachteten Ansätzen liegt die Integration von Sprachen zugrunde. Sei es eine Integration der Modellierungssprachen durch die Metamodell-basierten Ansätze oder eine Integration der Fachsprachen durch Ontologie-basierte Ansätze. Hinter der Überwindung der Sprachbarrieren steht die Vorstellung, dass eine Spracheinigkeit zur Integration führt.

Eine Integration ohne Integrationsziel ist allerdings wenig realistisch. Eine Integration wird sich nur für eine Auswahl festgelegter Aspekte durch zweckgebundene Artefakte implementieren lassen. Hierzu wird ein entsprechender Prototyp entwickelt.

6 Danksagung

Das Forschungs- und Entwicklungsprojekt „Integrative Produktentwicklung mit virtuellen Prototypen (INPROVY)“ wird mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) innerhalb des Rahmenkonzepts „Forschung für die Produktion von morgen“ gefördert und vom Projektträger Karlsruhe (PTKA) betreut. Weitere Informationen unter <http://www.inprov.de>.

Literatur

- [ABH⁺06] K. Arnarsdóttir, A. J. Berre, A. Hahn, M. Missikoff und F. Taglino. Semantic mapping: ontology-based vs. model-based approach Alternative or complementary approaches? In *Proceedings of the Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability, Luxembourg*, 2006.
- [ARW08] Stephan Aier, Christian Riege und Robert Winter. Unternehmensarchitektur – Literaturüberblick und Stand der Praxis. *WIRTSCHAFTSINFORMATIK*, 50(4):292–304, 2008.
- [BLN86] C. Batini, M. Lenzerini und S. B. Navathe. A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys (CSUR)*, 18(4):364, 1986.
- [Die08] Die Sprecher der Wissenschaftlichen Kommission Wirtschaftsinformatik im Verband der Hochschullehrer für Betriebswirtschaft (WKWI) und des Fachbereichs Wirtschaftsinformatik der Gesellschaft für Informatik (GI-FB WI). WI-Orientierungslisten. *WIRTSCHAFTSINFORMATIK*, 50(2):155–163, April 2008.
- [EGL09] Werner Esswein, Steffen Greiffenberg und Sina Lehrmann. Framework zur modellgestützten Simulation. *Wissensportal baumaschine.de*, (2), 2009.
- [FHM⁺03] D. S Frankel, P. Harmon, J. Mukerji, J. Odell, M. Owen, P. Rivitt, M. Rosen und R. M Soley. The Zachman Framework and the OMG’s Model Driven Architecture. *Business Process Trends*, (9):1–14, 2003.

- [Fra94] Ulrich Frank. *Multiperspektivische Unternehmensmodellierung. Theoretischer Hintergrund und Entwurf einer objektorientierten Entwicklungsumgebung*. Number 225 in GMD-Bericht. Oldenbourg, München u.a., 1994. Zugl.: Marburg, Univ., Habil.-Schr., 1993.
- [Fra02] Ulrich Frank. Multi-Perspective Enterprise Modeling (MEMO) - Conceptual Framework and Modeling Languages. *PROCEEDINGS OF THE 35TH HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES, HAWAII*, Seiten 1258—1267, 2002.
- [FRN08] J. Fengel, M. Rebstock und M. Nüttgens. Modell-Tagging zur semantischen Verlinkung heterogener Modelle. In *Proceedings der EMISA 2008, Bonn/St. Augustin 18.-19.09.2008*, Bonn u.a., 2008.
- [FT09] Michael Fellmann und Oliver Thomas. Management von Modellbeziehungen mit Semantischen Wikis. In *Proceedings der 9. Internationalen Tagung Wirtschaftsinformatik (WI 2009)*, 25.-27. Februar in Wien, Seiten 673–683, Wien, 2009. Österreichische Computer Gesellschaft.
- [Geh07] Andreas Gehlert. *Migration fachkonzeptueller Modelle*. Logos, Berlin, 2007.
- [Hö7] P. Höfferer. Achieving business process model interoperability using metamodels and ontologies. In *Proceedings of 15th European Conference on Information Systems*, Seite 1620–1631, 2007.
- [Hah05] A. Hahn. Integration verteilter Produktmodelle durch Semantic-Web-Technologien. *Wirtschaftsinformatik*, 47(4):278–284, 2005.
- [Hei00] Lutz J. Heinrich. Bedeutung von Evaluation und Evaluationsforschung in der Wirtschaftsinformatik. In Lutz J. Heinrich und Irene Häntschel, Hrsg., *Evaluation und Evaluationsforschung in der Wirtschaftsinformatik*, Seiten 7–22. Oldenbourg, München u.a., 2000.
- [HMPR04] A. R. Hevner, S. T. March, J. Park und S. Ram. Design science in information systems research. *Management information systems quarterly*, 28(1):75–106, 2004.
- [Kö6] Thomas Kühne. Matters of (Meta-) Modeling. *Software and Systems Modeling*, 5(4):369–385, 2006.
- [KBJK03] Harald Kühn, Franz Bayer, Stefan Junginger und Dimitris Karagiannis. Enterprise Model Integration. In *4th International Conference, EC-Web Prague, Czech Republic, September 2-5, 2003 Proceedings*, Jgg. 2738/2003 of *Lecture Notes in Computer Science*, Seiten 379–392, Berlin, 2003. Springer.
- [KH08] Dimitris Karagiannis und Peter Höfferer. Metamodeling as an Integration Concept. In Joaquim Filipe, Boris Shishkov und Markus Helfert, Hrsg., *Software and Data Technologies*, Jgg. 1 of *Communications in Computer and Information Science*, Seiten 37–50. Springer, Berlin u.a., 2008.
- [KR98] M. Kugeler und M. Rosemann. Fachbegriffsmodellierung für betriebliche Informationssysteme und zur Unterstützung der Unternehmenskommunikation. *Fachausschuss 5.2 der Gesellschaft für Informatik e. V. (GI): Informationssystem-Architekturen*, 5:8–15, 1998.
- [LK09] Yun Lin und John Krogstie. Quality Evaluation of a Business Process Semantic Annotations Approach. *International Journal of Interoperability in Business Information Systems (IBIS)*, 1(3), 2009.

- [LSH⁺06] Yun Lin, Darijus Straszunas, Sari Hakkarainen, John Krogstie und Arne Solvberg. Semantic Annotation Framework to Manage Semantic Heterogeneity of Process Models. In *Advanced Information Systems Engineering*, number 4001/2006 in Lecture Note in Computer Science, Seiten 433–446. Springer, Berlin, 2006.
- [MM03] Jishnu Mukerji und Joaquin Miller. MDA Guide Version 1.0.1, 2003.
- [Ros02] Michael Rosemann. *Komplexitätsmanagement in Prozeßmodellen*. Gabler, Wiesbaden, Marz 2002.
- [Sin99] E. J. Sinz. Architektur von Informationssystemen. In Peter Rechenberg und Gustav Pomberger, Hrsg., *Informatik-Handbuch*, Seiten 1035–1046. Hanser, München, 2. Auflage, 1999.
- [SK04] S Sendall und JM Köster. Taming Model Round-Trip Engineering. In *Proceedings of Workshop on Best Practices for Model-Driven Software Development*, 2004.
- [SM07] Carlo Simon und Jan Mendling. Integration of Conceptual Process Models by the Example of Event-driven Process Chains. In *8. Internationale Tagung Wirtschaftsinformatik WI, Karlsruhe, Germany, February 28 - March 2, 2007*, Seiten 677–694, Karlsruhe, 2007. Universitätsverlag Karlsruhe.
- [SR98] K. Siau und M. Rossi. Evaluation of information modeling methods-a review. In *Proceedings of the Thirty-First Annual Hawaii International Conference on System Sciences*, Jgg. 5, Seite 314–322, 1998.
- [VH05] Piet Verschuren und Rob Hartog. Evaluation in Design-Oriented Research. *Quality and Quantity*, 39(6):733–762, Dezember 2005.
- [VJBS98] P. R.S Visser, D. M Jones, T. J. M. Bench-Capon und M. J. R. Shave. Assessing heterogeneity by classifying ontology mismatches. In *Formal ontology in information systems: proceedings of the First International Conference (FOIS'98), June 6-8, Trento, Italy*, Seiten 148–162, 1998.
- [WBMW09] M. Weidlich, A. Barros, J. Mendling und M. Weske. Vertical Alignment of Process Models-How can we get there? In *CAiSE 2009 Workshop Proceedings-10th Workshop on Business Process Modeling, Development, and Support (BPMDS)*, ser. LNBIP, S. Nurcan, R. Schmidt, P. Soffer, and R. Ukor, Eds, Jgg. 29, Seite 71–84, 2009.
- [Win09] Robert Winter. *Management von Integrationsprojekten: Konzeptionelle Grundlagen und Fallstudien aus fachlicher und IT-Sicht*. Springer, Berlin, 1. Auflage, 2009.
- [Zac87] J. A. Zachman. A framework for information systems architecture. *IBM systems journal*, 26(3):276–292, 1987.
- [ZS07] Karagiannis D Zivkovic S, Kühn H. Facilitate Modelling Using Method Integration: An Approach Using Mappings and Integration Rules. In *Proceedings of the 15th European Conference on Information Systems (ECIS2007)*, Seiten 2038–2050, St. Gallen, 2007.