

2013

Nutzung des Viewpoint-Konzepts zur Unterstützung kollaborativer Modellierung – Konzeption und prototypische Implementierung

Julian Krumeich

Institut für Wirtschaftsinformatik (IWi), Deutsches Forschungszentrum für Künstliche Intelligenz GmbH (DFKI), Saarbrücken, Deutschland, julian.krumeich@iwi.dfki.de

Dirk Werth

Institut für Wirtschaftsinformatik (IWi), Deutsches Forschungszentrum für Künstliche Intelligenz GmbH (DFKI), Saarbrücken, Deutschland, Dirk.Werth@iwi.dfki.de

Peter Loos

Institut für Wirtschaftsinformatik (IWi), Deutsches Forschungszentrum für Künstliche Intelligenz GmbH (DFKI), Saarbrücken, Deutschland, loos@iwi.uni-sb.de

Follow this and additional works at: <http://aisel.aisnet.org/wi2013>

Recommended Citation

Krumeich, Julian; Werth, Dirk; and Loos, Peter, "Nutzung des Viewpoint-Konzepts zur Unterstützung kollaborativer Modellierung – Konzeption und prototypische Implementierung" (2013). *Wirtschaftsinformatik Proceedings 2013*. 79.
<http://aisel.aisnet.org/wi2013/79>

This material is brought to you by the Wirtschaftsinformatik at AIS Electronic Library (AISEL). It has been accepted for inclusion in Wirtschaftsinformatik Proceedings 2013 by an authorized administrator of AIS Electronic Library (AISEL). For more information, please contact elibrary@aisnet.org.

Nutzung des Viewpoint-Konzepts zur Unterstützung kollaborativer Modellierung – Konzeption und prototypische Implementierung

Julian Krumeich, Dirk Werth, und Peter Loos

Institut für Wirtschaftsinformatik (IWi),
Deutsches Forschungszentrum für Künstliche Intelligenz GmbH (DFKI),
Saarbrücken, Deutschland
{julian.krumeich, dirk.werth, peter.loos}@iwi.dfki.de

Abstract. In der Wirtschaftsinformatik werden Modelle zur Entwicklung von Informationssystemen eingesetzt. Trotz der inhärenten Komplexitätsreduzierung weisen diese häufig einen komplexen Charakter auf. Der wahrgenommene Nutzen durch den Modellnutzer ist daher vermindert. Dies wird intensiviert, wenn mehrere Subjekte aus unterschiedlichen Domänen bei der Modellerstellung und -nutzung im Sinne einer kollaborativen Modellierung involviert sind. Um die Modellierung hierbei zu unterstützen, gilt das Viewpoint-Konzept als erfolgsversprechend. Es ermöglicht eine Reduzierung der Komplexität von Modellen, indem diese in Stakeholder-spezifische Fragmente heruntergebrochen werden. Der vorliegende Beitrag konzipiert eine Viewpoint-basierte Modellierungsmethode, die das Viewpoint-Konzept mit der kollaborativen Modellierung verknüpft. Hierdurch wird den typischen Problemstellungen der kollaborativen Modellierung ebenfalls entgegengewirkt. Neben der Konzeption der Viewpoint-basierten Modellierungsmethode präsentiert der Beitrag überdies einen Proof-of-Concept anhand einer prototypischen Implementierung.

Keywords: Kollaborative Modellierung, Sichtenkonzept, Viewpoint-basierte Modellierung, konstruktionsorientierte Forschung

1 Einleitung

1.1 Motivation

Bei der Entwicklung von Systemen – im Kontext der Wirtschaftsinformatik typischerweise Informationssystemen – sind Modelle ein weitverbreitetes Mittel, um ein besseres Verständnis auf ein zugrundeliegendes Artefakt zu erhalten. Folgt man der allgemeinen Modelltheorie Stachowiaks [1], so sind Modelle „stets Modelle von etwas, nämlich Abbildungen, Repräsentationen natürlicher oder künstlicher Originale, die selbst wieder Modelle sein können.“

Auch wenn Modelle durch ihr inhärentes Verkürzungsmerkmal eine Komplexitätsreduzierung anstreben, so ist ihnen dennoch häufig – insbesondere bei der Entwick-

lung äußerst komplexer Systeme – ein weiterhin komplexer Charakter zuzusprechen. Die wahrgenommene Zweckdienlichkeit durch den Modellnutzer bleibt daher vermindert. Dies wird noch intensiviert, wenn mehrere Subjekte – die gegebenenfalls noch aus unterschiedlichen Domänen stammen – in die Modellerstellung und spätere -nutzung involviert sind. Bieten Modelle in einem solchen Umfeld lediglich eine einzelne, starre Perspektive auf das jeweilige Modell, so ist der Modellnutzen nur relativ gering für das einzelne beteiligte Subjekt erfahrbar [2].

Daher gilt es, die einzelnen Stakeholder, die bei der Modellerstellung als auch -nutzung beteiligt sind, stärker in den Fokus zu rücken, um damit den Nutzen der Modellierung zu erhöhen. Da die einzelnen Stakeholder in der Regel spezifische Fähigkeiten, Verantwortlichkeiten sowie Wissen und Expertise vorweisen [3], benötigen sie individuelle Perspektiven auf die Entwicklung des zugrundeliegenden Systems. Konsequenterweise sollte eine Betrachtung und Manipulation der Modelle und Modellfragmente nach den jeweils spezifischen Bedürfnissen ermöglicht werden [4].

1.2 Forschungsbeitrag

Um die Modellierung unter Berücksichtigung dieser Anforderungen zu unterstützen, gilt das Viewpoint-Konzept als erfolgversprechend. Dieses ermöglicht eine Reduzierung der Komplexität von Modellen, indem diese in Stakeholder-adäquate Fragmente heruntergebrochen werden [5]. Dadurch können die an der Modellierung beteiligten Akteure stärker in den Fokus des Modellierungsprozesses gerückt und so das Verständnis und die Produktivität gesteigert werden [3]. Gleichzeitig trägt das Viewpoint-Konzept zur Erstellung von besseren, das zugrundeliegende System adäquater abbildenden, Modellen bei. Dies konnte bereits durch Studien belegt werden [6].

Durch die Involvierung mehrerer Stakeholder in den Modellierungsprozess resultiert allerdings auch eine Komplexitätssteigerung durch die erschwerte Koordination des zugrundeliegenden Modellierungsprojektes. Bei einer solchen Konstellation wird von kollaborativer Modellierung gesprochen [7-8], die als ein „*process where a number of people actively contribute to the creation of a model*“ definiert wird [9]. Die Anwendung dieses Modellierungsprinzips ist jedoch nicht als trivial zu betrachten. [9] führte in diesem Kontext eine Expertenbefragung unter IT-Beratern durch, um herauszufinden, was die größten Probleme bei der kollaborativen Modellierung sind. Als Ergebnis ergab sich u. a. „low model acceptance“, „participants feel misunderstood“, „slow progress, facilitator overload“ und „limited model comprehension“.

Betrachtet man diese Probleme, so lässt sich feststellen, dass das Viewpoint-Konzept diese aber gerade zu lösen vermag und damit einen erfolgsversprechenden Beitrag zur kollaborativen Modellierung leisten könnte. Daher entwickelt der vorliegende Beitrag eine Viewpoint-basierte Modellierungsmethode, bei der das Viewpoint-Konzept genutzt wird, um eine effektive Einbeziehung mehrerer Stakeholder in den kollaborativen Modellierungsprozess zu erreichen. Überdies zeigt der Beitrag einen Proof-of-Concept des Konzeptes anhand einer prototypischen Implementierung.

1.3 Forschungsansatz und Aufbau des Beitrages

Dieser Beitrag folgt dem konstruktionsorientierten Forschungsansatz, bei dem ein betriebliches, d. h. prinzipiell praktisches Problem motivierend für die Entwicklung eines innovativen IT-Artefakts steht, mit dessen Hilfe eine Lösung der zugrundeliegenden Problemstellung erreicht werden soll. Diesem Forschungsansatz folgend entwickelt dieser Beitrag zunächst eine konzeptionelle Methode zur in Abschnitt 1.1 vorgestellten Ausgangsproblematik der betrieblichen Praxis. Darauf aufbauend wird überdies eine konkrete Instanz des Artefaktes in Form eines Prototyps vorgestellt.

Nach [10] sind bei der konstruktionsorientierten Forschung sieben Guidelines zu beachten, um eine stringente und problemrelevante Forschung durchzuführen. In diesem Sinne wird durch die Einleitung und Motivation die Relevanz des Forschungsproblems dargestellt (*Guideline 2*). Das in Abschnitt 3 entwickelte Konzept stellt ein IT-Artefakt (*Guideline 1*) dar, welches durch die prototypische Implementierung – wie sie in Abschnitt 4 vorgestellt wird – ein nachvollziehbares und prüffähiges Artefakt darstellt (*Guideline 4*). Dieses gilt es noch nach *Guideline 3* zu evaluieren (siehe hierzu Abschnitt 5). Um auf vorhandenem Wissen aufzubauen (*Guideline 6*), greift der in Abschnitt 3 entwickelte Ansatz auf eine zuvor durchgeführte Arbeit zurück, die wiederum auf weiteren existierenden Arbeiten aufbaut. Durch die Vorstellung der verfolgten Forschungsmethodik ist der *Guideline 5* ebenfalls Rechnung getragen. Letztlich wird durch die Einreichung dieses Beitrags die Erfüllung der letzten *Guideline 7* angestrebt, die eine Verbreitung des Konzepts vorsieht.

Der Aufbau des vorliegenden Beitrags ist wie folgt gegliedert: Nachdem in Abschnitt 1 die Motivation sowie der angewendete Forschungsansatz aufgezeigt wurden, werden in Abschnitt 2 verwandte Arbeiten im Bereich des Viewpoint-Konzepts und der kollaborativen Modellierung vorgestellt. Im Anschluss daran wird in Abschnitt 3 das Viewpoint-Konzept eingeführt und mit der kollaborativen Modellierung zur Viewpoint-basierten Modellierungsmethode verbunden. Diese wird im nachfolgenden Abschnitt 4 anhand einer prototypischen Implementierung instanziiert. Der Beitrag schließt in Abschnitt 5 mit einem Ausblick auf zukünftige Arbeiten.

2 Verwandte Arbeiten

Der Ursprung des Viewpoint-Konzepts geht auf die 1985 veröffentlichte Arbeit von [11] zurück [12]. Der dort entwickelte „MultiView“ Ansatz zielt auf die Unterstützung des Entwicklungsprozesses von computergestützten Informationssystemen ab, indem dieser komplexe Prozess in fünf unterschiedliche Perspektiven bzw. Viewpoints unterteilt wird. Ein weiterer Grundstein auf dem Gebiet der Viewpoint-Forschung stellt die in der Literatur häufig zitierte Arbeit von [3] dar. Während [11] mit ihrem Ansatz primär auf den Einsatz von Viewpoints in der Softwareentwicklung abzielen, ist der Ansatz von [3] potenziell bei der Entwicklung jedes Artefakts, welches nicht in einem trivialen Entwicklungsprozess konstruierbar ist, anwendbar.

Mit „IEEE 1471:2000“ existiert zudem eine Standarddefinition des Viewpoint-Konzepts, die auch von der ISO/IEC als „ISO/IEC 42010:2007“ Standard im Jahr 2007 angenommen wurde [13]. Demzufolge lässt sich das Viewpoint-Konzept bzw.

ein Viewpoint als „a specification of the conventions for constructing and using a view“ definieren. Abgesehen von der eigentlichen Definition des Viewpoint-Konzepts als auch der verschiedener Viewpoints in unterschiedlichen Anwendungsdomänen – hier ist bspw. im Bereich des Enterprise Architecture Managements das TOGAF-Rahmenwerk zu nennen [14] – sind auch Arbeiten zu erwähnen, die stärker auf die tatsächliche Nutzung des Viewpoint-Konzepts fokussieren.

Hier sei die Arbeit von [15] genannt. Diese schlagen ein methodisches Rahmenwerk vor, welches das Zusammenwirken zwischen einzelnen Viewpoints zu verbessern versucht. In diesem Kontext ist auch der „Hybrid Multi-View Modeling“ Ansatz von [16] aufzuführen. Dieser verfolgt das Ziel, eine (aus technischer Sicht) effiziente Synchronisation unterschiedlicher Views im Modellierungsprozess zu ermöglichen. Dies bedeutet, dass Änderungen, die in einer View eines Viewpoints durchgeführt wurden, mittels bestimmter Techniken erkannt und auf inkrementelle Weise an die weiteren existierenden Views propagiert werden. Hiermit wird das Ziel verfolgt, insbesondere bei größeren Modellen, keinen übermäßigen Overhead durch die Neuberechnung der durch die Modelländerung beeinflussten Views zu generieren.

Mit einem stärkeren Fokus auf die konkrete Umsetzung des Viewpoint-Konzepts in Modellierungswerkzeugen ist auf den erst kürzlich vorgestellten Beitrag von [5] zu verweisen. In diesem werden Anforderungen an die tatsächliche Umsetzung des Viewpoint-Konzepts bei der Entwicklung von Modellierungswerkzeugen untersucht. Auch eine durchgeführte Vorarbeit zu diesem Beitrag betrachtet solche werkzeugspezifischen Anforderungen an das Viewpoint-Konzept [2].

Zwar wird von manchen, in der Literatur zu findenden Ansätzen, die Zusammenführung verschiedener nach dem Viewpoint-Konzept erstellten Modelle erforscht, kollaborative Aspekte hinsichtlich der tatsächlichen Modellierung lassen sich dagegen in der Viewpoint-Forschung nicht erkennen.

In der Forschung zu kollaborativer Modellierung entwickelte [17] mit dem COMA-Werkzeug eine konkrete Werkzeugunterstützung zur Umsetzung der kollaborativen Modellierung. COMA steht für „Collaborative Modeling Architecture“, mit dessen Hilfe die Koordination von Gruppen bei der Modellierung von UML-Modellen erreicht werden soll. Ein weiterer Ansatz wird von [18] vorgeschlagen. Ihre „Fragment-Driven Process Modeling Methodology“ erlaubt die Entwicklung eines Prozessmodells durch mehrere Akteure auf kollaborative Weise. Hierzu modelliert jeder Akteur seine eigenen Aktivitäten, die sodann vom System zu einem Modell zusammengefügt werden. Allerdings wird hierbei lediglich ein Bottom-up Ansatz verfolgt, der synonym als Fragment-getriebener Ansatz bezeichnet wird. Folglich werden lediglich die einzelnen Prozessfragmente ablaforientiert zusammengesetzt. Eine kollaborative Modellierung am gleichen Prozessablauf bzw. Fragment und vor allem im Sinne der Viewpoint-basierten Modellierung wird nicht erzielt.

Ein Ansatz, der stärker auf das Zusammenfügen von verteilt erstellten Modellbereichen abzielt, ist der von [19] entwickelte Ansatz. Dieser zielt darauf ab, Modellierer beim Zusammenfügen und Integrieren verschiedener konzeptioneller Modelle zu unterstützen. Da hierbei ein Grad der Unsicherheit existiert, wird dieser mithilfe des Einsatzes von Unsicherheitsformalismen und der Nutzung von Erkenntnissen aus dem Bereich der „Belief Theory“ zu lösen versucht.

So zeigt sich bei der Betrachtung verwandter Arbeiten, dass bisher kein Ansatz existiert, der eine Verbindung des Viewpoint-Konzepts – der die Förderung der Involvierung verschiedener Akteure unterstützt – mit dem Forschungsfeld der kollaborativen Modellierung – das gerade die Koordinierung der Modellierung durch mehrere Akteure verfolgt – verbindet.

3 Methode zur Nutzung des Viewpoint-Konzepts zur Unterstützung kollaborativer Modellierung

3.1 Unterschiedliche Prinzipien der Modellbildung

Betrachtet man die beiden in der Betriebswirtschaftslehre überwiegend verwendeten Modelltheorien, nämlich den abbildungs- und konstruktionsorientierten Modellbegriff, so wird eine starke Bedeutung des subjektiven Einflusses auf das Modellverständnis deutlich (vgl. auch im Folgenden [20]). Während erstere das Subjekt in Bezug zum jeweiligen Modellzweck begreift und ihm einen beeinflussenden Einfluss auf die Modellgestaltung zuschreibt, hebt der konstruktionsorientierte Modellbegriff die Bedeutung des Modellierers als Subjekt hervor und unterstreicht dabei dessen Wechselwirkung mit dem Modellnutzer als Subjekt während des Konstruktionsprozesses von Modellen. Somit lassen sich zwei primäre Gestaltungsfaktoren erkennen, das „modellierende Subjekt“ und das „modellnutzende Subjekt“. In diesem Zusammenhang gilt es allerdings zu betonen, dass beide Ansätze jeweils von einem singulären Subjekt ausgehen bzw. eine singuläre Handlung eines subjektiven Kollektivs voraussetzen (vgl. Abbildung 1, links).

In der Realität sind bei Modellierungsvorhaben allerdings im Allgemeinen mehrere Einzelpersonen an der Modellbildung beteiligt. Auch ist zweifelsohne die Nutzung des Modells durch mehrere Personen zu attestieren. In der Praxis ist daher eher von einer kollaborativen Modellbildung auszugehen, d. h. dem Fall, in dem mehrere, nicht als Kollektiv handelnde Subjekte, gemeinsam ein Modell erzeugen (vgl. Abbildung 1, Mitte oben). Soweit sich eine solche multisubjektive Umgebung durch starke Autonomie der beteiligten Subjekte auszeichnet, kann nicht mehr von einer kooperativen Modellerstellung ausgegangen werden. Dies trifft insbesondere dann zu, wenn sich der Gegenstandsbereich des Originals zwischen den Modellierern unterscheidet. In diesem Fall muss eine kooperative Modellerstellung scheitern, da kein gemeinsamer Modellierungsgegenstand vorhanden ist. Vielmehr wird hier jeder Modellierer seine eigene Modellbildung durchführen, sodass für eine Menge von in Beziehung stehenden Originalen eine Menge von in Beziehung stehenden Modellsystemen entsteht. Eine solche Konstellation wird als multisubjektive Modellbildung bezeichnet und steht synonym für eine Modellbildung in multisubjektiver Umgebung autonomen Charakters (vgl. Abbildung 1, Mitte unten).

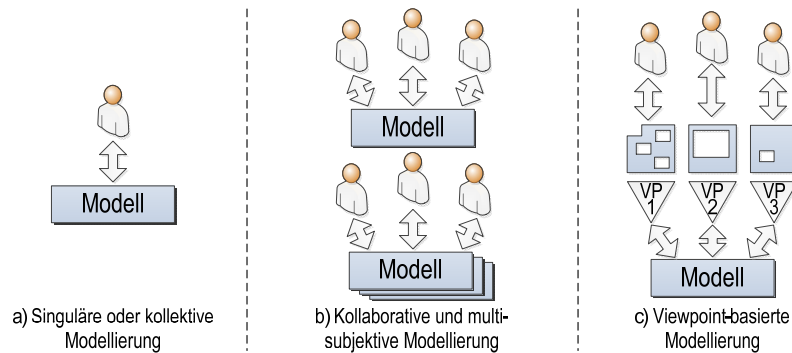


Abb. 1. Prinzipien der Modellbildung

Das in diesem Beitrag zugrundeliegende Modellierungsverständnis folgt dem sogenannten Viewpoint- bzw. Sichten-bezogenen Ansatz. Hier wird davon ausgegangen, dass im Modellierungsprozess – der Sichtweise der kollaborativen Modellierung folgend – mehrere Modellierer und Modellnutzer ein zu modellierendes System konstruieren und nutzen, die allesamt allerdings eine andere Zielsetzung mit dem Modell verfolgen können. Typischerweise lässt sich ein solches Szenario bei der Modellierung von Systemen unter Einbezug mehrerer Domänenexperten erkennen, die gemeinsam ein Gesamtsystem abzubilden versuchen. Folglich verlangt jedes beteiligte Subjekt – im Kontext des Viewpoint-Konzepts häufig als Stakeholder bezeichnet (vgl. bspw. [5]) – unterschiedliche Betrachtungsschwerpunkte und ggf. Repräsentationen zur Modellnutzung und -erstellung. Im Gegensatz zur multisubjektiven Modellierung, bei der mehrere Modelle durch die beteiligten Modellierer erstellt werden, wird beim Viewpoint-Ansatz hauptsächlich von der Erstellung eines Modells ausgegangen. Durch die Anwendung spezifischer Viewpoints auf dieses Modell wird sowohl dessen nutzerspezifische Darstellung als auch Anpassung erlaubt und dadurch der gemeinsame Modellierungsprozess adäquat auf die beteiligten Nutzer angepasst. Somit ermöglicht die Nutzung des Viewpoint-Konzepts eine Unterstützung der kollaborativen Modellierung. Die Verbindung beider Ansätze wird als Viewpoint-basierte Modellierung bezeichnet (vgl. Abbildung 1, rechts). Hierbei ist aus Mikrosicht die Viewpoint-spezifische Darstellung und Modellierung und aus Makrosicht die dynamische Koordination der Modellierung unter Beachtung der Anzahl (teilweise Vielzahl) an beteiligten Viewpoints zu berücksichtigen.

3.2 Statische Sicht der Viewpoint-basierten Modellierung

Wenngleich das Viewpoint-Konzept in diversen Domänen, wie dem Software Engineering oder dem Enterprise Architecture Management, Anwendung findet, so lässt sich kein domänenübergreifendes Verständnis des Konzepts attestieren [2]. Aus diesem Grund wurde in einer Vorarbeit zu diesem Beitrag eine domänenübergreifende Definition erarbeitet, die in diesem Abschnitt als Grundlage für die statische Konzeption der Viewpoint-basierten Modellierungsmethode genutzt wird. Anknüpfend daran

erfolgt im nachfolgenden Abschnitt 3.3 eine dynamische Auseinandersetzung mit der Viewpoint-basierten Modellierung.

Zur Konzeption der statischen Sicht werden zunächst grundlegende Begriffe eingeführt, basierend auf denen Viewpoint-spezifische Konzepte definiert werden.

Die Grundlage der Viewpoint-basierten Modellierung bilden *Metamodelle*. Diese definieren einen allgemeinen Rahmen zur Erzeugung von Modellen mittels einer Menge an Elementen sowie deren Beziehungen zueinander. Diese Elemente werden dabei als *Modellkonzepte* bezeichnet. Modellkonzepte sind daher Teil eines bestimmten Metamodells und dienen als Grundlage für die Instanziierung von Modellelementen in Modellinstanzen. Zur Erweiterung existierender Metamodelle können *Profile* eingesetzt werden. Diese nehmen ein Metamodell als Referenz und definieren die darin abstrakt spezifizierten Modellkonzepte für eine bestimmte Anwendungsdomäne um. Hierdurch wird die adäquate Adressierung einer Domäne durch das Metamodell ermöglicht. Ein Modell bzw. eine *Modellinstanz* stellt eine konkrete Ausprägung eines Metamodells dar und beinhaltet demnach eine bestimmte Menge an Modellelementen inkl. deren Beziehungen zueinander, die den im zugrundeliegenden Metamodell definierten Regeln folgen. Ein *Modellelement* ist somit als konkrete Ausprägung eines Modellkonzepts charakterisiert und stellt entweder ein Domänenobjekt oder eine Beziehung zwischen zwei oder mehreren Objekten dar.

Auf Basis dieser grundlegenden Begrifflichkeiten, welche die Basis für die Erzeugung von Modellen für eine bestimmte Anwendungsdomäne ermöglichen, werden nachfolgend Viewpoint-spezifische Begriffe eingeführt.

Entsprechend der Zielsetzung der Viewpoint-basierten Modellierung dienen *Viewpoints* der Unterstützung bestimmter Stakeholder, um ihren Beitrag zum zugrundeliegenden Systemdesign unter Berücksichtigung ihrer verschiedenen (fach-) spezifischen Perspektiven zu leisten. Unter Bezugnahme der zuvor eingeführten Modellbegriffe definiert ein Viewpoint demnach, welche Modellkonzepte von ausgewählten Metamodellen relevant für den jeweiligen Anwendungszweck, d. h. die Erzeugung, Nutzung und Manipulation von Modellinstanzen, eines Stakeholders sind. Somit bezieht sich ein Viewpoint auf eine Menge an Metamodellen und ggf. deren Erweiterungen durch Profile oder eine Teilmenge dieser. Folglich schränkt ein Viewpoint die ursprünglich zugrundeliegenden Metamodelle Stakeholder-spezifisch ein – die Definition lässt allerdings auch eine vollständige 1:1 Abdeckung zu. Um ein System durch seine zugrundeliegenden Modellinstanzen zielführend abzubilden, ist ein Viewpoint durch eine Menge an *ViewTypes* definiert, die er seinen Stakeholdern zur Abbildung von Modellinstanzen anbietet und die mittels Views instanziiert und visualisiert werden. Für jeden Viewpoint ist eine Menge an ViewTypes definiert, die jeweils eine bestimmte Perspektive eines Modells Stakeholder-spezifisch abbilden. Eine *View* – als konkrete Instanz eines ViewTypes – definiert ferner die Darstellung von Modellelementen für bestimmte Stakeholder und das Vorgehen, wie diese modifiziert werden können – dies wird für gewöhnlich mittels bestimmter Diagramme ermöglicht. Somit erlauben Views ihren Stakeholdern die Interaktion mit bestimmten Aspekten einer oder mehrerer Modellinstanzen, die dem Viewpoint zugrundeliegenden Metamodelle folgen. Während eine View die Darstellung von Modellelementen einer Modellinstanz realisiert, lässt die *Projektion* – als Spezialisierung einer View – über-

dies eine Stakeholder-spezifisch angepasste Visualisierung einzelner Modellelemente zu. Hierdurch ist nicht nur eine spezifische Selektion und Deselektion von Modellkonzepten möglich, sondern auch deren domänenadäquate Darstellung.

Die textuelle Darstellung der einzelnen Konzepte lässt bereits eine starke Verketten untereinander erkennen. Um den Zusammenhang des übergreifenden Viewpoint-Konzepts zu verdeutlichen, sind die einzelnen Konzepte daher in Abbildung 2 visualisiert.

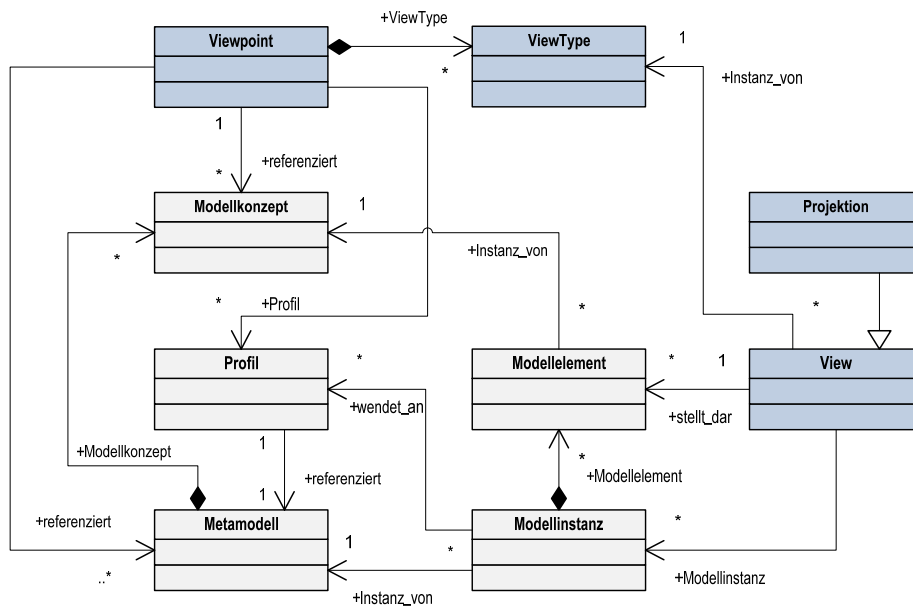


Abb. 2. Statische Sicht auf die Viewpoint-basierte Modellierung (in Anlehnung an [2])

3.3 Dynamische Sicht der Viewpoint-basierten Modellierung

Während die statische Konzeption der Viewpoint-basierten Modellierung primär die Aufspaltung von Modellen nach Stakeholder-spezifischen Anforderungen abdeckt, um diese zielgruppengerecht darzustellen, so muss überdies die dynamische Komponente im Sinne der Erzeugung und Manipulation von Modellinstanzen, d. h. die modellnutzende Komponente, ebenfalls konzipiert werden.

Zur Realisierung der Viewpoint-basierten Modellierung sollen die einzelnen Modellierungstätigkeiten so verteilt werden, dass das eigentliche Ziel der gemeinschaftlichen Schaffung einer konsistenten Modellinstanz erreicht wird. Die hierzu notwendigen Modellierungstätigkeiten werden demnach vom Verantwortlichen des jeweiligen Modellierungsprojektes in einzelne Modellierungsaufgaben heruntergebrochen. Ein reines Herunterbrechen ist allerdings nicht ausreichend, denn die einzelnen Aufgaben müssen den entsprechenden Modellierern zugeordnet werden.

Wird einem Stakeholder S_1 eine Modellierungsaufgabe zugeteilt, so wird ihm in seiner Modellierungsumgebung die zugrundeliegende Modellinstanz M basierend auf seinem Viewpoint VP_1 in der View $v_{VP_1,1}$ dargestellt (vgl. Abbildung 3, 1). Modifiziert er nun das zugrundeliegende Modell M mit seinem Domänenwissen, so kann hierdurch entweder eine konfliktfreie oder konfliktbehaftete Anpassung der Modellinstanz resultieren. Eine konfliktfreie Anpassung bedeutet (vgl. Abbildung 3, 2a), dass die Änderungen in seiner View $v_{VP_1,1}$ weder in Konflikt zu anderen Views $v_{VP_1,2-n}$ seines Viewpoints VP_1 noch zu anderen Viewpoints VP_{2-n} und deren Views $v_{VP_{2-n},1-m}$ als auch den kompletten zugrundeliegenden Metamodellen MM an sich stehen (vgl. hierzu [2]). In diesem Fall lässt sich die Modellanpassung ohne notwendige syntaktische Anpassung als konfliktfreie Modellinstanz M' speichern (vgl. Abbildung 3, 3a). Abgeleitet aus den zugrundeliegenden Metamodellen MM , die jeweils die zulässigen Beziehungen zwischen den Modellkonzepten definieren (vgl. Abschnitt 3.2), lassen sich syntaktische und zum Teil semantische Konflikte identifizieren. In der Literatur sind verschiedene Algorithmen hierzu publiziert worden (vgl. bspw. [21]), daher wird an dieser Stelle auf weitere technische Details verzichtet.

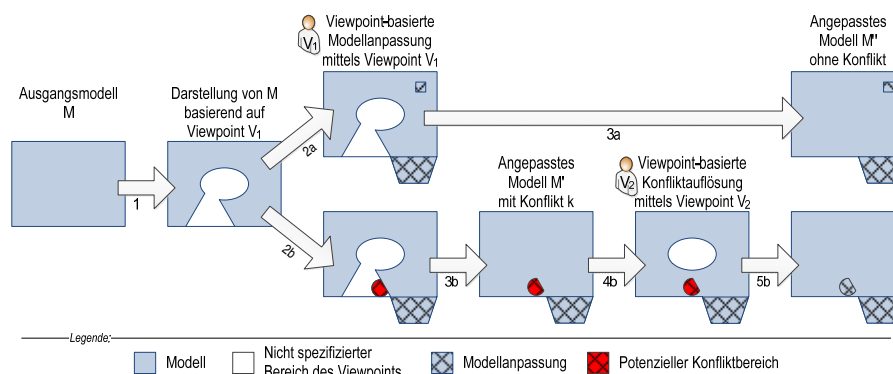


Abb. 3. Modellanpassungen bei der Viewpoint-basierten Modellierungsmethode

Während die konfliktfreie Modifikation als trivialer Fall anzusehen ist, gilt es für eine zielgerichtete kollaborative Modellierung auch den konfliktbehafteten Fall zu behandeln (vgl. Abbildung 3, 2b). An dieser Stelle sei explizit darauf hingewiesen, dass das Viewpoint-Konzept inkonsistente Zwischenzustände von Modellinstanzen erlaubt (vgl. bspw. [22]). Modifiziert Stakeholder S_1 daher die Modellinstanz M in seiner View $v_{VP_1,1}$ derart, dass seine Modellanpassungen einen Konflikt k in der angepassten Modellinstanz M' zu einem anderen Viewpoint VP_x oder einem zugrundeliegenden Metamodell MM_y darstellt (vgl. Abbildung 3, 3b) (globale Inkonsistenz), so muss eine syntaktische Anpassung erfolgen, die als zu erledigende Modellierungsaufgabe definiert werden kann. Sofern das konfliktbehaftete Modell M' konsistent zum gewählten Viewpoint VP_1 ist, kann der Stakeholder S_1 – sofern er nicht über weitere Viewpoints verfügt – eine globale Konsistenz nicht herstellen. Falls das Modell aber bereits inkonsistent zum gewählten Viewpoint ist (lokale Inkonsistenz), sollte Stakeholder S_1 in der Lage sein in einer anderen View v_x des ViewTypes VT_x den

Konflikt zu lösen. Bei lokaler Konsistenz, aber globaler Inkonsistenz kann somit Stakeholder S_1 die Inkonsistenz nicht lösen. Dies muss daher durch einen Stakeholder S_2 erfolgen, der mit seinem Viewpoint VP_2 die konfliktbehafteten Zustände k in M' auflösen kann (vgl. Abbildung 3, 4b). Hierzu muss ihm die entsprechende Modellierungsaufgabe zugeordnet werden. Nach der Anpassung der Modellinstanz M' durch Stakeholder S_2 ergibt sich nun erneut entweder ein konfliktbehafteter Zustand der Instanz oder eine konfliktfreie Modellinstanz M'' (vgl. Abbildung 3, 5b).

Um die Zuordnung solcher Modellierungsaufgaben zu unterschiedlichen Stakeholdern und deren Viewpoints zu ermöglichen – die dazu noch weitestgehend automatisch erfolgen soll bzw. dem Projektverantwortlichen in Form einer Empfehlung ausgesprochen wird – muss eine geeignete Datenbasis existieren. Diese muss Informationen hinsichtlich der Erfahrung von Stakeholdern im Kontext der Viewpoints, der Erfahrung mit ähnlichen Aufgaben und die derzeitige Modellierungsauslastung beinhalten. Auf dieser Grundlage kann eine Empfehlungsgenerierung erfolgen. Da es nicht Ziel dieses Beitrages ist, eine feingranulare Darstellung jedes Teilaspektes der Viewpoint-basierten Modellierung zu liefern, sondern primär die Schaffung eines Verständnisses über das Zusammenspiel der drei unterschiedlichen Teilfunktionen im Sinne einer integrierten Gesamtfunktionalität, wird an dieser Stelle auf eine detaillierte Darstellung verzichtet. Die Autoren verfolgen derzeit die Veröffentlichung eines weiteren Beitrags, der die Empfehlungsgenerierung expliziert.

3.4 Veranschaulichung der Viewpoint-basierten Modellierung anhand eines Szenarios

Zur Veranschaulichung des Zusammenspiels der einzelnen Aspekte des Konzepts ist in Abbildung 4 ein beispielhaftes kollaboratives Modellierungsszenario aufgezeigt.

Der Projektmanager des Modellierungsprojektes erstellt eine Aufgabenstruktur, welche die Modellierung des Projektes abdeckt. Bei der Zuweisung der Modellierer zu den einzelnen Aufgaben unterstützt ihn das Empfehlungssystem und schlägt ihm für jede umzusetzende Modellierungsaufgabe den geeignetsten Viewpoint vor. Die einzelnen Stakeholder erhalten daraufhin in ihrer Modellierungsumgebung eine Mitteilung über eine neue Modellierungsaufgabe. Auch hier kommt das Empfehlungssystem zum Tragen, indem es die günstigste Aufgabensequenz für den Modellierer empfiehlt, anhand derer er seine Aufgaben erledigen soll. Durch die Wahl des Viewpoints erhält der Modellierer überdies bereits die geeignete Benutzerumgebung, um die Modellierungsaufgabe zu lösen. Nach Abschluss der Aufgabe speichert der Modellierer seine Modellinstanz ins zentrale Modellrepository. Die im Hintergrund verlaufende Konsistenzprüfung anhand der Metamodelle generiert daraufhin bei Bedarf neue Modellierungsaufgaben für ihn oder andere Stakeholder.

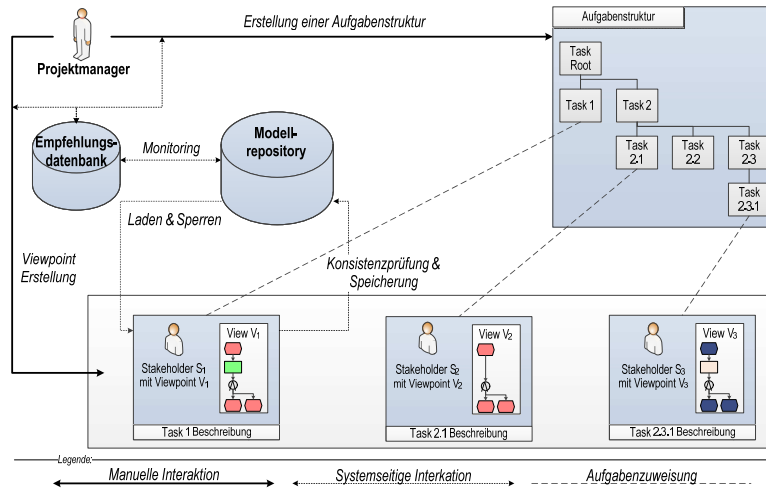


Abb. 4. Darstellung eines Viewpoint-basierten Modellierungsszenarios

4 Prototypische Konzeptimplementierung

Dem konstruktionsorientierten Forschungsansatz folgend, der diesem Beitrag zugrunde liegt (vgl. Abschnitt 1.3), wird in diesem Abschnitt das zuvor konzeptionell entwickelte Artefakt in seiner prototypischen Umsetzung *CIMFlex4CM* vorgestellt. *CIMFlex4CM* ist ein auf Eclipse aufbauendes Modellierungstool, das Benutzern eine eigene View auf gemeinsame Modelle erlaubt. Zudem realisiert das Tool die beschriebene Zuteilung von Modellierungsaufgaben. Abbildung 5 zeigt eine Viewpoint-spezifische Darstellung zweier Prozessmodelle in *CIMFlex4CM*, die auf den Metamodellen der EPK und BPMN beruhen und in getrennten Views dargestellt werden. Zudem ist ein Wizard zur Erzeugung von Viewpoints im unteren Bereich sichtbar.

Grundlegend basiert der entwickelte Prototyp auf den für *Eclipse* bereitgestellten Java-Frameworks *Eclipse Modeling Framework* (EMF) und *Graphical Modeling Framework* (GMF). Diese ermöglichen es, Modellierungstools in der Eclipse IDE zu entwickeln, ohne alle modellierungsspezifischen Funktionen von Grunde auf neu implementieren zu müssen. Hierzu erlaubt EMF anhand von strukturierten Metamodellen automatisch Java-basierten Quellcode zu erzeugen. Als Metamodelle dienen Ecore-Modelle, die auf dem EMOF (Essential Meta-Object Facility)-Standard basieren. Auf dieser Grundlage ist es nun mit GMF möglich, grafische Editoren auf den EMF-aufbauenden Modellen zu generieren (für weitere Details vgl. z. B. [23]).

Zur Realisierung der Viewpoint-basierten Modellierungsmethode wurde aufbauend auf EMF/GMF mittels zweier Generatoren die Erzeugung und Anpassung von Viewpoints und den dazugehörigen ViewTypes realisiert (vgl. Abbildung 6). Diese wurden basierend auf den in Eclipse bekannten Wizards implementiert.

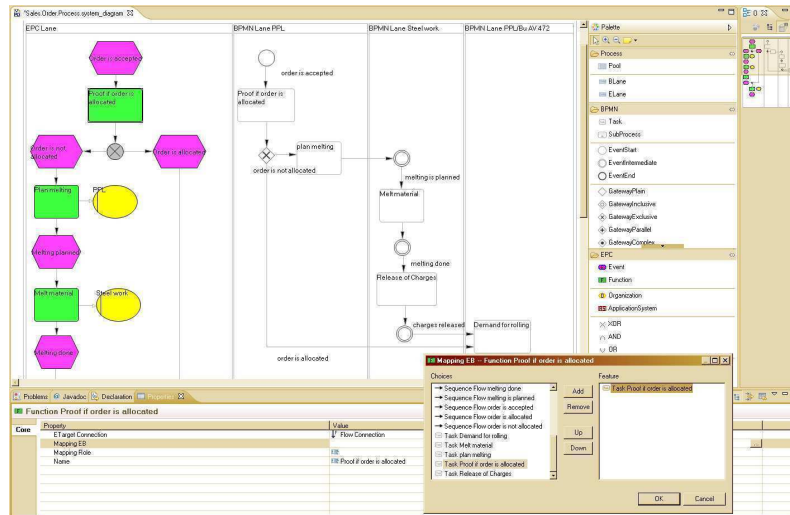


Abb. 5. Screenshot des CIMFlex4CM-Prototypes

Der entwickelte Viewpoint-Generator lädt hierzu das jeweilige Metamodell als Ecore-Datei und speichert die vom Viewpoint-Ersteller getroffene Selektion an Modellkonzepten in einer XML-basierten Datei ab. Neben der reinen Selektion von Konzepten können zudem sogenannte Flags gesetzt werden, die definieren, ob die ausgewählten Konzepte nur schreibgeschützt verwendet werden sollen und ob eine Hinterlegung mit weiteren Modellinstanzen möglich sein soll. Im letzten Fall muss zusätzlich das gewünschte Metamodell ausgewählt werden, um den hinterlegbaren Modelltyp zu definieren. Der Generator ermöglicht ebenfalls das Laden und Abändern bereits erstellter Viewpoints.

Zu den erstellten Viewpoints lassen sich mit Hilfe des zweiten Generators dazugehörige ViewTypes erzeugen. Hierzu wird die erzeugte Viewpoint-Datei als auch eine dazugehörige Default-View-Datei, welche die Standarddarstellung der Metamodellkonzepte definiert, geladen. Der ViewType-Generator ermöglicht es nun, die Viewpoint-spezifische Konzeptauswahl weiter einzuschränken als auch die Selektion von bestimmten Modellelementen zu ermöglichen. Falls gewünscht kann überdies eine View-spezifische Darstellungsform gemäß der Projektion definiert und ebenfalls als XML-basierte Datei gespeichert werden. Sind sowohl Viewpoint als auch eine Menge an ViewTypes erstellt, kann ein entsprechendes Eclipse Plugin generiert und an den jeweiligen Endnutzer ausgerollt werden. Dieser kann das Plugin laden und von seiner Eclipse-Instanz die gewünschte Modellinstanz aus einem zentral angelegten Modellrepository öffnen und sodann Viewpoint-spezifisch darstellen und bearbeiten.

Um die Viewpoint-basierte Modellierung (vgl. Abschnitt 3.3) zu realisieren und den Aspekt der Modellierungsaufgaben, die es innerhalb eines (kontinuierlichen) Modellierungsprojektes zu bewältigen gibt, in den Prototyp zu integrieren ist dieser mit dem Eclipse Mylyn Plug-In verknüpft. Mylyn stellt ein *Task and Application Lifecycle Management* Rahmenwerk dar, das ein aufgabenfokussiertes Interface sowie verschiedene *Issue Tracking* Funktionalitäten in Eclipse implementiert.

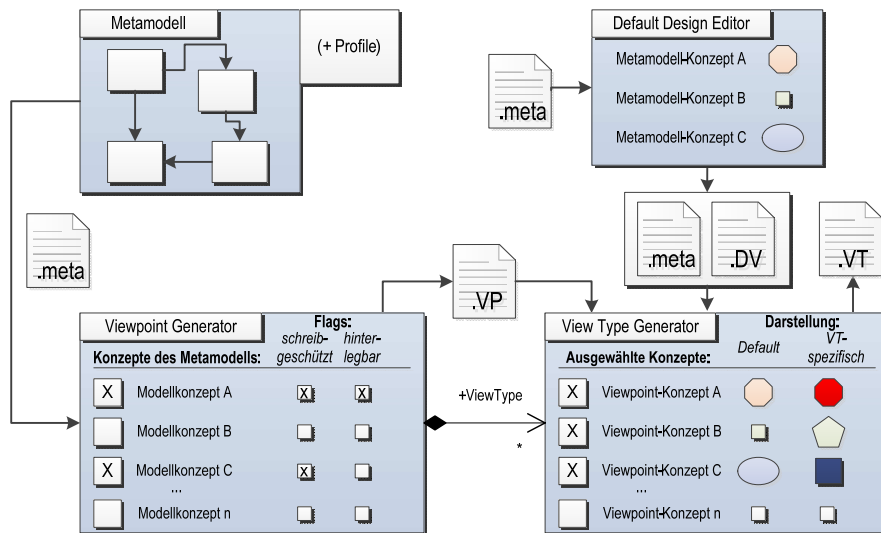


Abb. 6. Erzeugung von Viewpoints und ViewTypes

Zur optimalen Verteilung der einzelnen Modellierungsaufgaben an die geeigneten Stakeholder, ist das Mylyn Plugin überdies an das Empfehlungssystem Apache Mahout geknüpft, das die optimale Zuteilung der einzelnen Aufgaben basierend auf den vorhandenen Viewpoints und dazugehörigen ViewTypes realisiert und so bestimmten Stakeholdern zuordnet. Um eine Datengrundlage für das eingesetzte Empfehlungssystem zu schaffen, beobachtet das Issue Tracking System von Mylyn u. a. die Zuteilung und das Lösen von Aufgaben als auch entstandene Probleme hierbei. Hierdurch wird ebenfalls implizit ein Reporting innerhalb eines Modellierungsprojektes erzielt. Zur Empfehlungsgenerierung werden dann die einzelnen Stakeholder nach ihrer Erfahrung im Kontext des Viewpoints, ihrer Erfahrung mit ähnlichen Aufgaben und wie ihre derzeitige Auslastung mit Modellierungsaufgaben ist, eingestuft. Es gilt allerdings anzumerken, dass das eingesetzte System wie alle Recommender eine gewisse Datenbasis benötigt, um zufriedenstellende Empfehlungen geben zu können.

Somit ermöglicht der Prototyp eine gezielte Zuordnung einzelner Modellierungsaufgaben unter den kollaborierenden Teilnehmern. Überdies empfiehlt das integrierte Empfehlungssystem zusätzlich dem Nutzer indirekt die geeignete Modellierungsumgebung, die im Viewpoint bzw. den ViewType definiert ist. Als dritte Funktionalität ermöglicht das Empfehlungssystem zudem die optimale Aufgabenreihenfolge sowohl aus Projekt- als auch Stakeholder-bezogener Perspektive.

5 Zusammenfassung und Ausblick

Der vorliegende Beitrag konzipierte eine Viewpoint-basierte Modellierungsmethode, die das sogenannte Viewpoint-Konzept mit der kollaborativen Modellierung verknüpft. Durch die Nutzung von Stakeholder-spezifischen Viewpoints werden typische

Problemstellungen der kollaborativen Modellierung – z. B. die häufig geringe Akzeptanz von Modellen unter beteiligten Akteuren aus unterschiedlichen Domänen aufgrund deren eingeschränkten Verständnisses der übergreifenden Modelle – angegangen. Resultierend erleichtert und verbessert das Viewpoint-Konzept die Einbeziehung unterschiedlicher Stakeholder der verschiedensten Domänen in den kollaborativen Modellierungsprozess. Neben der Konzeption der Viewpoint-basierten Modellierungsmethode zeigte der Beitrag überdies dessen Proof-of-Concept anhand einer prototypischen Implementierung auf.

Das in diesem Beitrag publizierte Forschungsergebnis ist innerhalb des europäischen Forschungsprojektes *ViBaM* (Viewpoint-Based Modeling) entstanden. In Folge des Projektes planen die beiden beteiligten Industriepartner *SOFTEAM* und *IKV++ TECHNOLOGIES* die Integration des entwickelten Konzepts in deren kommerziellen Modellierungswerkzeuge *MODELIO* und *MEDINI ANALYZE*. Hierdurch wird die kollaborative Modellierung in den beiden Anwendungsdomänen der Projektpartner – die Unternehmensmodellierung sowie die Analyse funktionaler Sicherheit bei der Entwicklung von Automobilelektronik – unterstützt.

Danksagung. Die hier vorgestellten Forschungsergebnisse wurden im Rahmen des Forschungsprojektes *ViBaM* (Viewpoint-Based Modeling) erarbeitet, das vom Bundesministerium für Bildung und Forschung (BMBF) (FKZ 01QE1106B/C) innerhalb des Europäischen Förderprogramms Eurostars (E! 5529) gefördert wurde. Die Autoren möchten sich weiterhin bei den anonymen Gutachtern für die konstruktiven Anmerkungen bedanken, die zur Verbesserung dieser Artikel beigetragen haben.

Literatur

1. Stachowiak, H.: Allgemeine Modelltheorie. Springer, Berlin Heidelberg (1973)
2. Fischer, K., Panfilenko, D., Krumeich, J., Born, M., Desfray, P.: Viewpoint-Based Modeling - Towards Defining the Viewpoint Concept and Implications for Supporting Modeling Tools. In: EMISA 2012 - International Workshop on Enterprise Modelling and Information Systems Architectures, pp. 123-136. Köllen GmbH, Bonn (2012)
3. Finkelstein, A., Kramer, J., Nuseibeh, B., Finkelstein, L., Goedicke, M.: Viewpoints: A Framework for Integrating Multiple Perspectives. *Syst. Development* 2 (1), 31-57 (1992)
4. Dijkman, R.M., Quartel, D.A.C., van Sinderen, M.J.: Consistency in multi-viewpoint design of enterprise information systems. *Information and Software Technology* 50 (7-8), 737-752 (2008)
5. Goldschmidt, T., Becker, S., Burger, E.: Towards a Tool-Oriented Taxonomy of View-Based Modelling. In: Sinz, E.J., Schürr, A. (eds.): *Modellierung 2012*. LNI, Vol. 201, pp. 59-74. Gesellschaft für Informatik, Bonn (2012)
6. Easterbrook, S., Yu, E., Ar, J., Fan, Y., Horkoff, J., Leica, M., Qadir, R.A.: Do viewpoints lead to better conceptual models? An exploratory case study. In: *RE '05 Proceedings of the 13th IEEE International Conference on Requirements Engineering*, pp. 199-208. IEEE Computer Society, Los Alamitos (2005)
7. Frost & Sullivan: *Meetings Around the World: The Impact of Collaboration on Business Performance*. Frost & Sullivan White Papers (2007)

8. Renger, M., Kolfshoten, G.L., de Vreede, G.-J.: Challenges in Collaborative Modeling: A Literature Review. In: Dietz, J.L.G., Albani, A., Barjis, J. (eds.): *Advances in Enterprise Engineering I. LNBIP*, Vol. 10, pp. 61-77. Springer, Berlin Heidelberg (2008)
9. Rittgen, P.: Collaborative Modeling - A Design Science Approach. In: *HICSS '09 42nd Hawaii International Conference on System Sciences*. IEEE Computer Society, Los Alamitos (2010)
10. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. *MIS Quarterly* 28 (1), 75-105 (2004)
11. Wood-Harper, A.T., Antill, L., Avison, D.E.: *Information systems definition: the Multiview approach*. Blackwell Scientific Publications, Oxford (1985)
12. Lankhorst, M.: *Enterprise Architecture at Work: Modelling, Communication and Analysis*. Springer, Berlin Heidelberg (2009)
13. Software Engineering Standards Committee of the IEEE Computer Society: *IEEE Recommended Practice for Architectural Description of Software-Intensive Systems*. IEEE Computer Society, Los Alamitos (2007)
14. The Open Group. Welcome to TOGAF Version 9 – An Open Group Standard, <http://pubs.opengroup.org/architecture/togaf9-doc/arch/index.html>
15. Andrade, J., Ares, J., García, R., Pazos, J., Rodríguez, S., Silva, A.: A Methodological Framework for Viewpoint-Oriented Conceptual Modeling. *IEEE Transactions on Software Engineering* 30 (5), 282-294 (2004)
16. Cicchetti, A., Ciccozzi, F., Leveque, T.: Supporting incremental synchronization in hybrid multi-view modeling. In: *MODELS'11 Proceedings of the 2011th international conference on Models in Software Engineering*. LNCS, Vol. 7167, pp. 89-103. Springer, Berlin Heidelberg (2012)
17. Rittgen, P.: Collaborative Business Process Modeling - Tool Support for Solving Typical Problems. In: *CONF-IRM 2010 Proceedings*, paper 45. AIS Electronic Library (2010)
18. Kim, K.-H., Won, J.-K., Kim, C.-M.: A Fragment-Driven Process Modeling Methodology. In: *Computational Science and Its Applications – ICCSA 2005 Proceedings*. LNCS, Vol. 3482, pp. 817-826. Springer, Berlin Heidelberg (2005)
19. Bagheri, E., Ghorbani, A.A.: A belief-theoretic framework for the collaborative development and integration of para-consistent conceptual models. *Journal of Systems and Software* 82 (1), 707-729 (2008)
20. Werth, D.: *Modellierung unternehmensübergreifender Geschäftsprozesse: Modelle, Notationen und Vorgehen für prozessorientierte Unternehmensverbände*. EHV, Bremen (2005)
21. Paige, R.F., Brooke, P.J., Ostroff, J.S.: Metamodel-based model conformance and multiview consistency checking. *ACM Transactions on Software Engineering and Methodology (TOSEM)* 16 (3), article 11 (2007)
22. Finkelstein, A., Gabbay, D., Hunter, A., Kramer, J., Nuseibeh, B.: Inconsistency Handling in Multiperspective Specifications. *IEEE Transactions on Software Engineering* 20 (8), 569-578 (1994)
23. Rempp, G., Akermann, M., Löffler, M., Lehmann, J.: *Model Driven SOA*. Springer, Berlin Heidelberg (2011)