

# Intelligentes Verknüpfungsmanagement am Beispiel einer 3D\_Visualisierung

Reinhard Hübler, Heiko Willenbacher<sup>1</sup>, Christian Koch  
Professur Informations- und Wissensverarbeitung  
Bauhaus-Universität Weimar

## 1. Motivation

Aufgrund der zunehmenden Komplexität der Bauplanungsprozesse unter starken terminlichen Restriktionen werden Bauprojekte in immer stärkerem Maße durch räumlich verteilt und zeitlich variant tätige hochspezialisierte Fachplaner auf Basis stark differenzierter Hard- und Softwaresysteme bearbeitet.

Zur qualifizierten Erledigung ihrer Aufgaben sind die einzelnen Fachplaner demnach auf Informationen räumlich entfernter und zeitlich vorgelagerter oder parallel ausgeführter Prozesse angewiesen. Daraus ergibt sich die Notwendigkeit einer durchgängig computergestützten Informationsbereitstellung zwischen den einzelnen Bearbeitern und Applikationen und den durch sie unterstützten Prozessen.

Das hier betrachtete Forschungsvorhaben gliedert sich in den Sonderforschungsbereich 524<sup>2</sup> ein. In diesem SFB spielen planerische Prozesse für die Revitalisierung von Bauwerken eine wesentliche Rolle. Aufgrund der möglichen Vielfalt der involvierten Prozesse kann die Revitalisierung als lebenszyklusweite Betrachtung der Bauwerksplanung angesehen werden.

Deshalb bildet die Integration der verschiedenen Prozesse und der korrespondierenden Bearbeiter und Applikationen in einer gemeinsamen Arbeits- und Planungsumgebung einen Schwerpunkt des SFB 524. Ziel dieser Umgebung ist die Gewährleistung einer koordinierten Kooperation und Kommunikation zur Unterstützung einer qualifizierten prozeßspezifischen Entscheidungsfindung.

Gerade für die Bauplanung und die hierbei sehr schwach strukturiert auftretenden Prozesse stellt eine Integration auf Basis des gemeinsamen Planungsgegenstandes Bauwerk den adäquaten Ansatzpunkt zur Schaffung einer derartigen integrierten Arbeits- und Planungsumgebung dar [Wi02].

## 2. Grundlagen

Dem aktuellen Stand der Forschung entsprechend, besteht die Grundlage der gegenstandsbezogenen Integration in einer deskriptiv orientierten Bereitstellung der prozeßübergreifenden Datengesamtheit in Form eines Bauwerksmodells (objektorientiert strukturierte Klassenbeschreibung) und den daraus zu instanzierenden konkreten Bauwerksmodelldaten (Abbildung 1). Die Bauwerksmodelldaten bilden in ihrer Gesamtheit das "Virtuelle Bauwerk" als gemeinsame Arbeitsgrundlage für die beteiligten Fachplaner und Applikationen.

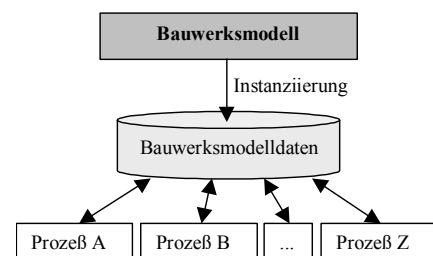


Abbildung 1: Integration auf Basis eines Bauwerksmodells

<sup>1</sup> email: [heiko.willenbacher@informatik.uni-weimar.de](mailto:heiko.willenbacher@informatik.uni-weimar.de)

<sup>2</sup> SFB 524 - Titel: "Werkstoffe und Konstruktionen für die Revitalisierung von Bauwerken". Dieses Projekt wird von der Deutschen Forschungsgemeinschaft DFG gefördert.

Aufgrund verschiedenster Probleme hinsichtlich der Komplexität und Dynamik von Bauwerksmodellen [WH00] wurde im SFB 524 ein spezieller "Verknüpfungsbasierter Bauwerksmodellansatz" entwickelt. Dieser Ansatz zeichnet sich dadurch aus, dass sich das Bauwerksmodell dynamisch aus verschiedenen prozeß- bzw. domänenspezifischen Partialmodellen (Klassenbeschreibungen) konstituiert (Abbildung 2).

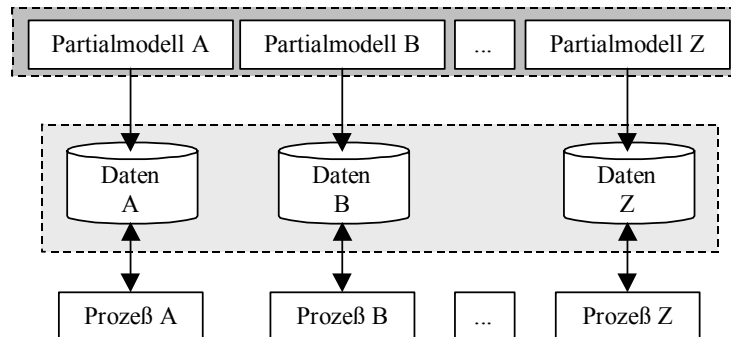


Abbildung 2: Partialmodellansatz

Eine weitere Besonderheit dieses Ansatzes liegt in der Dynamik der Partialmodelle. Das bedeutet, daß die Partialmodelle zur Laufzeit (Bauwerkslebenszyklus) modifizierbar sind und somit an spezielle a priori unbekannte Informationsanforderungen durch neu hinzugekommene Prozesse angepaßt werden können. Hierzu werden die Partialmodelle und die entsprechenden Instanzen in speziellen Modellverwaltungssystemen [Ha02] vorgehalten.

Die Gewährleistung des domänenübergreifenden Informationszugriffes und -austausches wird durch ein Verknüpfungsmanagement realisiert. Das Verknüpfungsmanagement verwaltet Verknüpfungstypen zwischen den Partialmodellen auf Modellebene und Instanzen der Verknüpfungstypen, die Verknüpfungen auf Bauwerksmodelldatenebene (Abbildung 3). Die Verknüpfungstypen definieren Beziehungen zwischen Elementen auf Modellebene. Dabei können dynamisch beliebige Quell- und Zielmodellelemente unterschiedlicher Partialmodelle in Be-

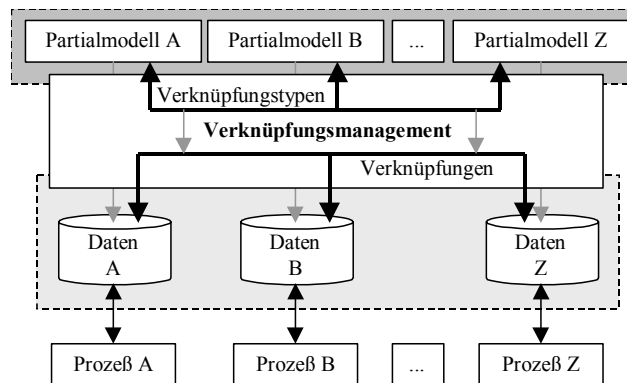


Abbildung 3: Verknüpfungsbasierter Bauwerksmodellansatz

ziehung gesetzt werden. Basierend auf den definierten Verknüpfungstypen können Verknüpfungen instanziiert werden, wobei während dieses Vorgangs spezielle korrespondierende Elemente (Quell- und Zielelemente) auf Datenebene spezifiziert und entsprechend des Verknüpfungstyps in Relation gesetzt werden [WHB01].

Der Verknüpfungsbasierter Bauwerksmodellansatz bietet somit ein adäquates Instrumentarium, um die Dynamik der Partialmodelle sowie des Bauwerksmodellverbundes handhaben zu können [Wi02].

### 3. Problemstellung

Die Möglichkeit, Partialmodelle und den Partialmodellverbund dynamisch modifizierbar zu halten, bietet ein Höchstmaß an Flexibilität hinsichtlich der involvierten Partialmodelle und der damit in die Arbeits- und Planungsumgebung integrierbaren Prozesse.

Bezogen auf den prozeßübergreifenden Informationsaustausch erfordert die Dynamik ein laufzeitdynamisches Verknüpfungsmanagement zur Gewährleistung dieser Flexibilität.

Aus dieser Dynamik und dem Versuch die auftretenden, a priori unbekanntenen Informationsanforderungen uneingeschränkt zu unterstützen, resultiert eine wesentliche Problematik des verknüpfungsbasierten Modellierungsansatzes.

Die Definition von Verknüpfungstypen und deren Ausprägung stellt einen hochkomplexen Prozeß dar, welcher erhebliche Fachkenntnisse speziell über die involvierten Partialmodelle voraussetzt.

Aufgrund der Partialmodelldynamik kann nicht von bekannten (standardisierten) Partialmodellen ausgegangen werden. Die Dynamik erlaubt es beispielsweise Klassen beliebig hierarchisch anzuordnen oder z.B. Klassen und Attribute beliebig zu benennen bzw. umzubenennen. Schlußfolgernd können bei den Verknüpfungserstellenden in der Regel nur beschränkte Kenntnisse über die Strukturen und Inhalte der Partialmodelle vorausgesetzt werden.

Daraus ergibt sich die Notwendigkeit unterstützender Methoden und Verfahren, um die Verknüpfungserstellung<sup>3</sup> so effektiv und effizient und für den Verknüpfungserstellenden so einfach und intuitiv wie möglich zu gestalten.

### 4. Lösungsansatz

Einen wesentlichen Ansatzpunkt stellt hierbei zunächst die adäquate Präsentation der in Relation zu setzenden Modell- und Datenelemente dar. Als vorteilhaft erweist sich diesbezüglich die zur Datenverwaltung eingesetzte objektorientierte generisch-deskriptive Modellverwaltungsphilosophie (s. Abschnitt 2) [St97], welche die Analyse und Repräsentation der Modellinhalte und Strukturen sowohl auf Modell- als auch auf Datenebene erlaubt (Abbildung 4).

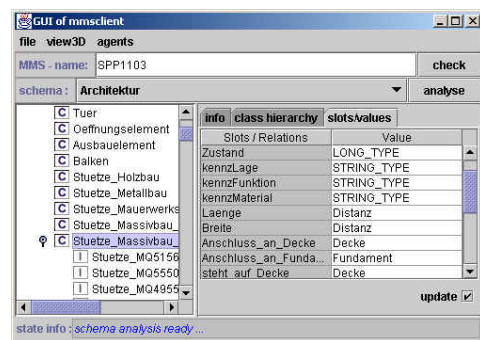


Abbildung 4: Modellanalyse, -visualisierung

Zur effektiven Unterstützung der Verknüpfungserstellung muß das Verknüpfungsmanagement über Hilfsmittel verfügen, welche in der Lage sind, selbständig bzw. semiautomatisch in Relation stehende Elemente zu erkennen und daraus Vorschläge für Verknüpfungstypen bzw. Verknüpfungen zu generieren.

Im hier beschriebenen Forschungsvorhaben wird ein regelbasierter Ansatz zur Entwicklung derartiger Hilfsmittel konzipiert und entwickelt. Dabei wird jedoch im ersten Schritt (Inhalt dieses Beitrages) auf Regelverarbeitungsmechanismen für Verknüpfungstypdefinitionen also für die Verknüpfung von Elementen auf Modellebene fokussiert.

Das Ziel der Regelverarbeitung besteht demnach darin, aufgrund speziellen Wissens über die in Beziehung zu setzenden Quell- und Zielmodellelemente, Vorschläge für Verknüpfungstypen zu generieren.

<sup>3</sup> Unter dem Begriff Verknüpfungserstellung wird die Verknüpfungstypdefinition sowie die Verknüpfungsspezifikation subsumiert.

Das für die Regelverarbeitung notwendige Wissen über die jeweiligen Modellelemente wird dabei durch die Analyse der deskriptiv vorgehaltenen Partialmodelle, welche über eine Schnittstelle mit selbsterklärendem Charakter verfügen (AKO-Interface [Ba96]), extrahiert.

Auf Modellebene handelt es sich hierbei zum einen um strukturelles Wissen. Dieses Wissen bezieht sich auf die partialmodellinternen Assoziationen (Generalisierung / Spezialisierung, Aggregationen) zwischen den Modellelementen.

Zum anderen handelt es sich um inhaltliches Wissen. Unter diesem Begriff werden alle Informationen über ein spezielles Element subsummiert. Dabei sind insbesondere Namen (Klassennamen, Attributnamen) und Datentypen (Attributwertebereiche, Kardinalitäten) von Bedeutung.

Die strukturellen und inhaltlichen Informationen über die Elemente unterschiedlicher Partialmodelle können unter der Voraussetzung entsprechender Verknüpfungsregeln ausreichen, um in Beziehung stehende Elemente auf Modellebene herauszufinden und somit Verknüpfungstypen zu definieren.

Aufgrund der Komplexität und insbesondere der Dynamik der Partialmodelle werden diese Informationen jedoch in der Regel nicht ausreichen, um korrespondierende Modellelemente eindeutig und semantisch korrekt zu identifizieren. Aus dem selben Grund ist es unmöglich, die notwendigen Verknüpfungsregeln komplett vorzudefinieren. Dies würde letztlich bedeuten, alle Verknüpfungstypen vorzudefinieren.

Die im vorliegenden Ansatz verwendeten Regeln beziehen sich deshalb ausschließlich auf das dynamisch analysierte Wissen über die aktuellen Modellelemente und unterliegen demnach der Ungewißheit des Modellanalyseprozesses.

Schlußfolgernd kann nicht davon ausgegangen werden, das die Verarbeitung der Verknüpfungsregeln eindeutige, korrekte Verknüpfungstypen liefert. Das Ergebnis kann aber als Vorschlag für den Verknüpfungserstellenden aufbereitet werden. Dem Fachplaner (Verknüpfungserstellender) obliegt letztlich die Entscheidung, den Verknüpfungstypvorschlag zu akzeptieren oder ihn gegebenenfalls zu modifizieren.

## 5. Realisierung

Im derzeit untersuchten Beispielszenarium besteht die Aufgabe, Verknüpfungstypen zwischen den Elementen eines Partialmodells *Architektur* und eines Partialmodells *3D-Visualisierung* zu definieren. Dabei soll den Elementen des Architekturmodells (*Klassen: Stützen, Decken, Fundament,...*) eine graphische Repräsentation (*java3D Primitive: Box, Cone,...*) zugeordnet werden.

Das eigentliche Problem ergibt sich aus dem initial unbekanntem und nicht determiniertem Wissen über die Struktur und den Inhalt des Architekturmodells. So können Attribute, welche zur Darstellung einer Bauteilklassse relevant sind (*länge, breite, höhe, einfügepunkt,...*) beispielsweise erst durch Verfolgen mehrerer beliebig tiefer Assoziationen gefunden werden. Diese können sich prinzipiell jederzeit ändern. Außerdem können die Bauteile hierarchisch umgeordnet werden (worden sein) und es können sich Attributnamen ändern.

Im folgenden werden die untersuchten Sachverhalte am Beispiel der Zuordnung zwischen Stütze und Box dargestellt.

Die Aufgabe der Regelverarbeitung besteht deshalb konkret darin, die geometrischen Informationen zwischen den beiden Klassen aufeinander abzubilden, d.h. für die vorgegebenen Parameter (Attribute) der Box entsprechende Attribute der Klasse Stütze zu finden.

Für diese Zielstellung wurde die in Abbildung 5 dargestellte Handlungsfolge auf Basis der in Abbildung 6 illustrierten Konzeption entwickelt und entsprechend implementiert.

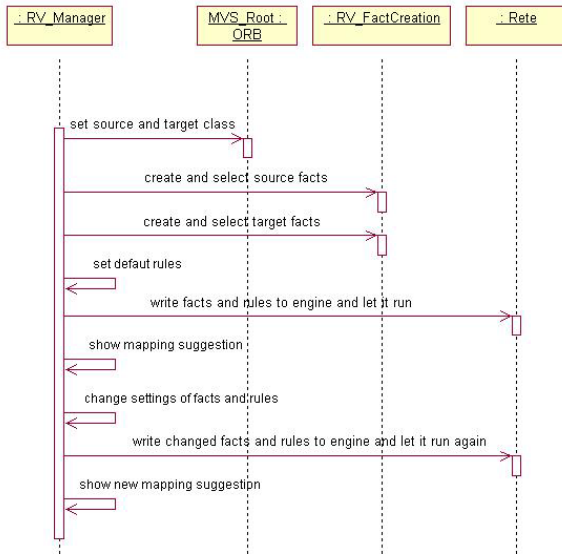


Abbildung 5: Sequenzdiagramm der Regelverarbeitung

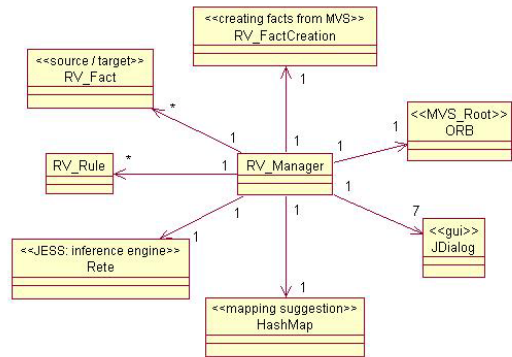


Abbildung 6: Klassendiagramm

Der Verknüpfungstyperstellende spezifiziert dialogbasiert jeweils eine Klasse des Architekturmodells (Klasse: *Stütze*) und eine Klasse des Repräsentationsmodells (Klasse: *Box*). Die konzipierte Regelverarbeitungs-komponente (RV\_Manager) analysiert zunächst das betreffende Partialmodellelement, um Fakten (RV\_Fact) für die Überprüfung durch Regeln (RV\_Rule) zu generieren (RV\_FactCreation). Die hier auf Verknüpfungstypenebene abgeleiteten Fakten beinhalten im wesentlichen Aussagen über die Attributnamen, deren Zugriffspfade und Datentypen. Die erzeugten Fakten können in den Regelverarbeitungsprozeß aufgenommen oder davon ausgeschlossen werden (Abbildung 7). Anschließend können generelle und individuelle Regeln definiert bzw. festgelegt werden. Die Regeln beziehen sich auf die Inhalte der Fakten und überprüfen in dem Fall Attributnamen bzw. Datentypen auf Gleichheit. Dabei können individuelle Regeln für spezielle Attribute oder generelle Regeln mit bestimmten Prioritäten verwendet werden (Abbildung 8). Die Fakten und Regeln werden der Regelverarbeitung übergeben (diese wird durch Jess<sup>4</sup> realisiert). Der dieser Bibliothek zugrunde liegende Rete-Algorithmus wertet die Fakten und Regeln

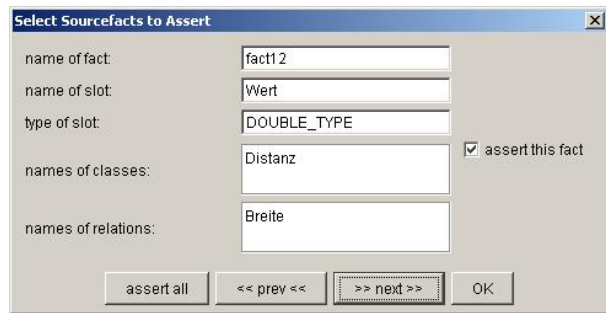


Abbildung 7: Fakt für Regelverarbeitung

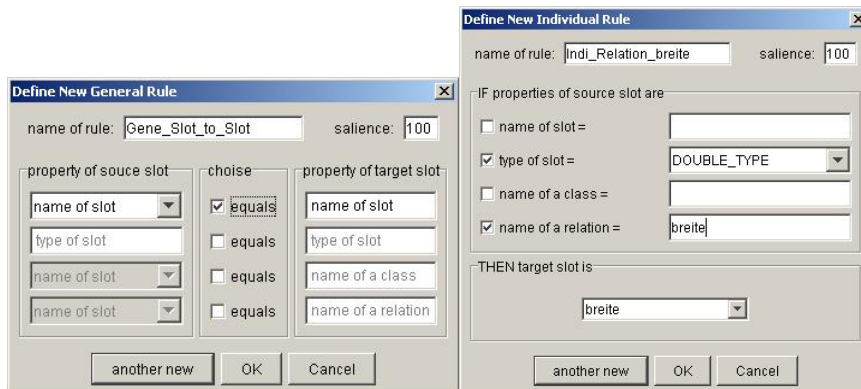


Abbildung 8: Regeldefinition

<sup>4</sup> Jess, the Java Expert System Shell

aus. Dabei wird jedem Parameter der Box ein den gegebenen Regeln und Fakten entsprechendes Attribut der Klasse Stütze zugeordnet.

Das Ergebnis ist letztlich ein Abbildungsvorschlag (Abbildung 9) der vom Verknüpfungserstellendem direkt übernommen oder modifiziert werden kann. Dabei können weitere Fakten und Regeln hinzugefügt und die Regelverarbeitung erneut gestartet werden.

Der akzeptierte Vorschlag wird in einen Verknüpfungstyp codiert und an das Verknüpfungsmanagement weitergeleitet. Auf Basis dieses Verknüpfungstyps können nun zu allen Instanzen der Klasse Stütze geometrische Repräsentation als korrelierende Instanzen im 3D-Visualisierungsmodell erzeugt werden.

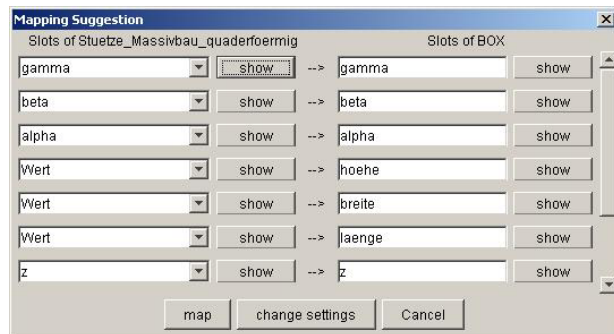


Abbildung 9: generierter Abbildungsvorschlag

## 6. Fazit

Mit dem implementierten Prototyp konnte gezeigt werden, daß die regelbasierte Analyse und Verarbeitung der deskriptiv vorliegenden Partialmodellinformationen einen geeigneten Ansatz zur Unterstützung des Verknüpfungserstellungsprozesses darstellt. Die Suche nach in Beziehung stehenden Attributen erledigt die Regelverarbeitung, wodurch sich der Prozeß der Verknüpfungstyperstellung wesentlich effizienter und einfacher gestaltet. Die dynamische Umsetzung der Zuordnungsvorschläge in Verknüpfungstypdefinitionen gewährleistet, daß die deskriptiv vorliegenden Partialmodellelemente beliebig entsprechend spezieller Intentionen repräsentiert werden können.

Den Gegenstand weiterer Untersuchungen bildet zum einen die Erweiterung der Regelverarbeitung auf das Suchen und in Beziehung setzen von Klassen (und nicht nur deren Attributen) auf Modellebene.

Zum anderen soll auf Modelldatenebene untersucht werden, welche Möglichkeiten die Regelverarbeitung bietet, um die Verknüpfungsspezifikation durch Analyse vorhandener Instanzen zu unterstützen.

## 7. Literatur

- [Ba96] Bachmann, U.: Untersuchung zur Strukturierung einer Schnittstelle von CAD-Tools und dynamischen Modellverwaltungssystemen, Diplomarbeit, Bauhaus-Universität Weimar, 1996
- [St97] Steinamnn, F.: Modellbildung und computergestütztes Modellieren in frühen Phasen des architektonischen Entwurfes, Dissertation, Bauhaus-Universität Weimar, 1997
- [WH00] Willenbacher, H.; Hübler, R.: Relationen zwischen Domänenmodellen Ansatz zur Schaffung einer integrierten computergestützten Bauplanungsumgebung; IKM 2000
- [WHB01] Willenbacher, H.; Hübler, R.; Bubner, A.: Verknüpfungsbasierter Bauwerksmodellansatz, Forum Bauinformatik München, VDI Verlag GmbH 2001
- [Ha02] Hauschild, T.: Collaborative, Dynamic Building Model Management Systems for Revivification Projects; CAD2002, Dresden March 2002
- [Wi02] Willenbacher, H.: Interaktive verknüpfungsbasierte Bauwerksmodellierung als Integrationsplattform für den Bauwerkslebenszyklus, Dissertation, Bauhaus-Universität Weimar, Juni 2002