

Zweitveröffentlichung



Sinz, Elmar J.

Ein Architekturrahmen für die Modellierung betrieblicher Informationssysteme

Datum der Zweitveröffentlichung: 24.09.2024

Verlagsversion (Version of Record), Beitrag in Sammelwerk

Persistenter Identifikator: urn:nbn:de:bvb:473-irb-982231

Erstveröffentlichung

Sinz, Elmar J. (1995): „Ein Architekturrahmen für die Modellierung betrieblicher Informationssysteme“. In: Festschrift zum 60. Geburtstag von Walter Augsburg, Bamberg: Univ., S. 83–87.

Rechtehinweis

Dieses Werk ist durch das Urheberrecht und/oder die Angabe einer Lizenz geschützt. Es steht Ihnen frei, dieses Werk auf jede Art und Weise zu nutzen, die durch die für Sie geltende Gesetzgebung zum Urheberrecht und/oder durch die Lizenz erlaubt ist. Für andere Verwendungszwecke müssen Sie die Erlaubnis der Rechteinhaberinnen und Rechteinhaber einholen.

Für dieses Dokument gilt das deutsche Urheberrecht.

EIN ARCHITEKTURRAHMEN FÜR DIE MODELLIERUNG BETRIEBLICHER INFORMATIONSSYSTEME

Elmar J. Sinz¹

Einführung

Modelle stellen das wichtigste methodische Hilfsmittel zur Beherrschung komplexer Systeme dar. Dies gilt insbesondere auch für Analyse, Gestaltung und Durchführung betrieblicher Systeme sowie ihrer informationsverarbeitenden Teilsysteme, der betrieblichen Informationssysteme.

Unter einem Modell versteht man ein System, welches ein anderes System zielorientiert abbildet. Ein Modell besteht daher aus drei Komponenten, einem Objektsystem S_O (Ursystem), einem Modellsystem S_M (Bildsystem) sowie einer Modellabbildung $f: K_O \rightarrow K_M$, welche die Komponentenmenge K_O des Objektsystems S_O in die Komponentenmenge K_M des Modellsystems S_M abbildet [FeSi94, 18f] (Bild 1). Struktur- und Verhaltenstreue zwischen S_O und S_M wird durch die Verwendung isomorpher bzw. homomorpher Modellabbildungen f erreicht.

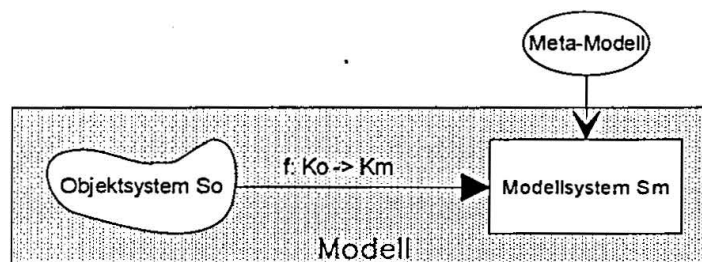


Bild 1: Modell und Meta-Modell

Bei der Modellierung betrieblicher Systeme stellt das Objektsystem einen Ausschnitt der betrieblichen Realität, also ein reales System dar. Das Modellsystem ist ein formales System. Zur Spezifikation von Modellsystemen sind insbesondere semi-formale Ansätze geeignet, die ihrerseits in Form von Meta-Modellen spezifiziert werden. Ein Meta-Modell gibt somit den Beschreibungsrahmen für die semi-formale Spezifikation von Modellsystemen vor.

Im allgemeinen ist die zur Beherrschung betrieblicher Systeme notwendige Komplexität der zugehörigen Modellsysteme so hoch, daß es dem Menschen nicht möglich ist, ein Modellsystem in seiner Gesamtheit zu erfassen. Aus diesem Grund werden Modellsysteme zielorientiert in Teilmodellsysteme gegliedert. Jedes Teilmodellsystem stellt ein vollständiges Bild des Objektsystems unter einem bestimmten Blickwinkel dar. Die Teilmodellsysteme werden entsprechend ihrer Beziehungen untereinander angeordnet und bilden die Modellebenen des Modellsystems.

¹ Univ.-Prof. Dr. Elmar J. Sinz, Otto-Friedrich-Universität Bamberg, Lehrstuhl für Wirtschaftsinformatik, insbesondere Systementwicklung und Datenbankanwendung, D-96045 Bamberg, Tel. (0951) 863-2512, Fax (0951) 863-2513, Internet elmar.sinz@sowi.uni-bamberg.de

Innerhalb der einzelnen Teilmodellsysteme wird die Bewältigung der Komplexität durch die Bildung von Sichten unterstützt. Jede Sicht beschreibt bestimmte Systemmerkmale eines Teilmodellsystems. Die Sichten einer Modellebene ergeben zusammen das jeweilige Teilmodellsystem.

Im folgenden wird ein Architekturrahmen für die Modellierung betrieblicher Informationssysteme vorgestellt, welcher in allgemeiner Form eine Strukturierung von Modellsystemen in Modellebenen und Sichten unterstützt. Die Nutzung des Architekturrahmens wird anschließend anhand ausgewählter Modellebenen des SOM-Ansatzes [FeSi95] beispielhaft erläutert.

Ein Architekturrahmen für betriebliche Informationssysteme

Der Architekturrahmen (Bild 2) beschreibt die Strukturierung von Modellsystemen auf der Meta-Ebene, d.h. auf der Ebene der Meta-Modelle. Er unterscheidet hierzu die beiden Dimensionen Modellstruktur und Modellebenenhierarchie. Die Modellstruktur einer Modellebene umfaßt

- das zur Spezifikation der jeweiligen Modellebene verwendete Meta-Modell (MM),
- die auf der jeweiligen Modellebene verwendeten Sichten $S_1..S_n$, die in Form von Projektionen auf das Meta-Modell spezifiziert werden und
- eine Menge von Strukturmustern (Pattern) $P_1..P_m$, welche für das Meta-Modell der betrachteten Modellebene definiert sind.

Das Konzept der Patterns stammt ursprünglich aus der Architekturlehre und wurde in den letzten Jahren für die objektorientierte Softwareentwicklung adaptiert. Unter einem Pattern versteht man in diesem Zusammenhang ein generisches Strukturmuster, welches das Zusammenwirken von Objektklassen bzw. Objekten bei der Lösung einer spezifischen Problemstellung beschreibt (siehe z.B. [Gamm*95]). Ein typisches Beispiel ist das aus der objektorientierten Programmiersprache Smalltalk bekannte Model-View-Controller-Pattern. Patterns schränken die aufgrund des Meta-Modells einer Modellebene zulässigen Strukturen von Teilmodellsystemen ein und können somit als strukturelle Integritätsbedingungen oder auch als heuristisches Modellierungswissen verstanden werden. Im vorliegenden Architekturrahmen wird das Konzept der Patterns für beliebige Modellebenen verallgemeinert.

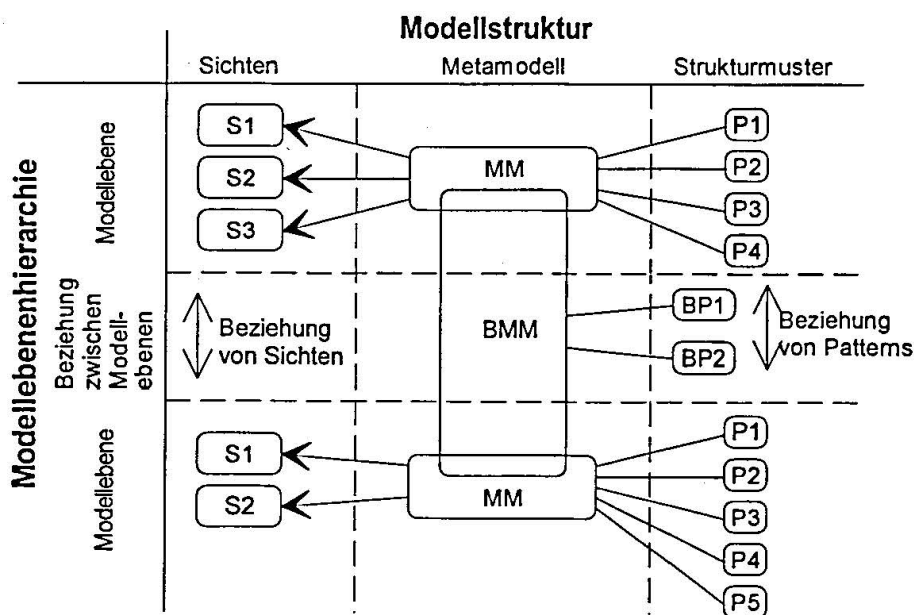


Bild 2: Architekturrahmen für die Modellierung betrieblicher Informationssysteme

Die Modellebenenhierarchie setzt Teilmodellssysteme unterschiedlicher - nicht notwendig benachbarter - Modellebenen zueinander in Beziehung. Jede Beziehung zwischen Modellebenen wird spezifiziert durch

- ein Beziehungs-Meta-Modell (BMM), welches zwei Meta-Modelle verknüpft, indem es Bausteine dieser Meta-Modelle zueinander in Beziehung setzt und
- Beziehungspatterns $BP_1..BP_k$, die Teilstrukturen aus Bausteinen verschiedener Modellebenen beinhalten und als heuristisches Modellierungswissen zur Verbindung von Modellebenen interpretiert werden können.

Je eindeutiger die Beziehung zwischen zwei Modellebenen durch das Beziehungs-Meta-Modell spezifiziert ist, desto geringer ist der durch Beziehungs-Patterns auszufüllende Freiraum und umgekehrt. Aus dem Beziehungs-Meta-Modell und den Sichten, die auf den in Beziehung gesetzten Meta-Modellen definiert sind, folgt transitiv eine Beziehung von Sichten.

Nutzung des Architekturrahmens am Beispiel des SOM-Ansatzes

Der Modellierungsansatz des Semantischen Objektmodells (siehe [FeSi95] sowie die dort angegebene Literatur) unterscheidet drei Ebenen für die Modellierung eines betrieblichen Systems bzw. eines betrieblichen Informationssystems:

- Unternehmensplan als Modell der Außensicht eines betrieblichen Systems.
- Geschäftsprozeßmodell als Modell der Innensicht eines betrieblichen Systems. Das Geschäftsprozeßmodell wird dabei als Spezifikation der Lösungsverfahren für die Umsetzung des Unternehmensplans verstanden.
- Spezifikation von Anwendungssystemen als Ressourcen zur Durchführung des automatisierten Anteils von Geschäftsprozessen.

Die Spezifikation von Anwendungssystemen kann aus Architektursicht weiter untergliedert werden in

- eine fachliche Spezifikation (Teil des Fachkonzepts) in Form eines Konzeptuellen Objektschemas (KOS) und eines Vorgangsobjektschemas (VOS),
- eine softwaretechnische Spezifikation (Teil des Softwarekonzepts), das neben einem um softwaretechnische Attribute erweiterten KOS und VOS ein Interface-Objektschema (IOS) und ein Konzeptuelles Objektschema persistenter Klassen (KOS-P) enthält, und
- eine Spezifikation der verwendeten Entwicklungsplattform in Form eines Technischen Objektschemas (TOS) [Amb93, FeSi93].

Im folgenden wird die Nutzung des Architekturrahmens am Beispiel des SOM-Ansatzes aufgezeigt. Hierzu werden die beiden Modellebenen (a) Geschäftsprozeßmodell und (b) fachliche Spezifikation von Anwendungssystemen herausgegriffen.

Bild 3 zeigt das Meta-Modell für Geschäftsprozeßmodelle im SOM-Ansatz [FeSi95]. In Form von Projektionen auf das Meta-Modell sind zwei Sichten spezifiziert, das Interaktionsmodell und das Aufgabensystem. Zusätzlich sind für das Meta-Modell eine Reihe von Strukturmustern definiert. Hierzu gehören das Client/Server-Muster für Leistungsbeziehungen zwischen betrieblichen Objekten, das Verhandlungsmuster für die nicht-hierarchische Koordination sowie das Regelungsmuster für die hierarchische Koordination betrieblicher Objekte.

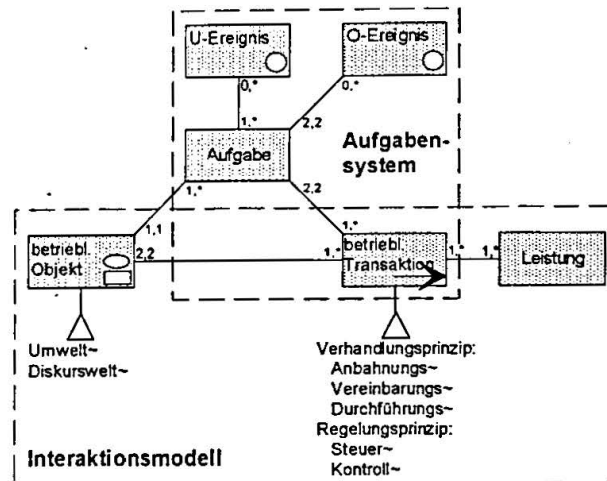


Bild 3: Meta-Modell für Geschäftsprozeßmodelle im SOM-Ansatz [FeSi95]

Bild 4 zeigt das Meta-Modell für die fachliche Spezifikation von Anwendungssystemen im SOM-Ansatz. Der SOM-Ansatz geht dabei von einer objektorientierten Spezifikation aus, bestehend aus Objekttypen (bzw. Klassen), die durch unterschiedliche Arten von Beziehungen verknüpft sind.

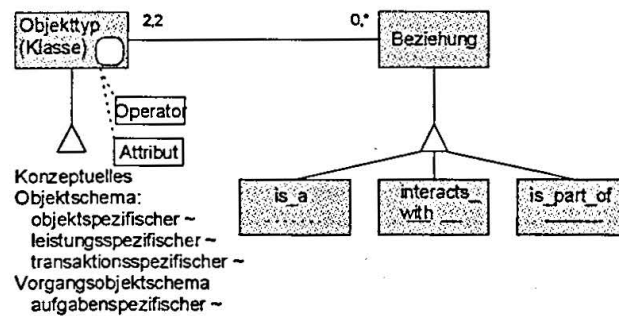


Bild 4: Meta-Modell für die fachliche Spezifikation von Anwendungssystemen im SOM-Ansatz [FeSi93]

Das Beziehungs-Meta-Modell wird aus Gründen der Übersichtlichkeit tabellarisch dargestellt. Jede Zeile der Tabelle entspricht dabei einer Beziehung im Beziehungs-Meta-Modell.

<p>Interaktionsmodell (Bild 3) Betriebliches Objekt Betriebliche Transaktion Leistung Aufgabensystem (Bild 3) Aufgabe Ereignis Transaktion</p>	<p>Konzeptuelles Objektschema (Bild 4) Objektspezifischer Objekttyp Transaktionsspezifischer Objekttyp + interacts_with-Beziehung(en) Leistungsspezifischer Objekttyp + interacts_with-Beziehung(en) Vorgangsobjektschema (Bild 4) Aufgabenspezifischer Objekttyp interacts_with-Beziehung interacts_with-Beziehung</p>
---	--

Bild 5: Beziehungs-Meta-Modell

Bild 5 zeigt, daß anhand des Beziehungs-Meta-Modells aus einem gegebenen Geschäftsprozeßmodell wesentliche Teile der fachlichen Spezifikation von Anwendungssystemen ableitbar sind. Beziehungspatterns besitzen in diesem Fall eine untergeordnete Bedeutung.

Die vollständige Spezifikation des Architekturrahmens für den SOM-Ansatz umfaßt Meta-Modelle, Sichten und Strukturmuster für die fünf Modellebenen (Unternehmensplan, Ge-

schäftsprozeßmodelle, Anwendungssystemspezifikation (fachlich), Anwendungssystemspezifikation (softwaretechnisch) und Entwicklungsplattform) sowie Beziehungs-Meta-Modelle und Beziehungspatterns zur Verknüpfung der Modellebenen.

Analog hierzu kann der Architekturrahmen für andere semi-formale Modellierungsansätze [Sinz95] sowie für Kombinationen aus unterschiedlichen Modellierungsansätzen genutzt werden.

Literatur

[Amb93] Amberg M.: Konzeption eines Software-Architekturmodells für die objektorientierte Entwicklung betrieblicher Anwendungssysteme. Dissertation, Bamberg 1993

[FeSi93] Ferstl O.K., Sinz E.J.: Der Modellierungsansatz des Semantischen Objektmodells (SOM). Bamberger Beiträge zur Wirtschaftsinformatik Nr. 18, 1993

[FeSi94] Ferstl O.K., Sinz E.J.: Grundlagen der Wirtschaftsinformatik, Band 1. 2. Auflage, Oldenbourg, München 1994

[FeSi95] Ferstl O.K., Sinz E.J.: Der Ansatz des Semantischen Objektmodells (SOM) zur Modellierung von Geschäftsprozessen. WIRTSCHAFTSINFORMATIK 37 (1995) 3, S. 209 - 220

[Sinz95] Sinz E.J.: Ansätze zur fachlichen Modellierung betrieblicher Informationssysteme. Entwicklung, aktueller Stand und Trends. Bamberger Beiträge zur Wirtschaftsinformatik Nr. 34, 1995. Erscheint in: Heilmann H., Heinrich L.J., Roithmayr F. (Hrsg.): Information Engineering. Oldenbourg, München 1996

[Gamm⁺95] Gamma E., Helm R., Johnson R., Vlissides J.: Design Patterns. Elements of Reusable Object-Oriented Software. Addison-Wesley, Reading, Massachusetts 1995