

Jürgen Demharter, Sebastian Fuchs, Sven-Eric Schapke und  
Raimar J. Scherer

---

## Zusammenfassung

Mit dem generischen Multimodell wird eine allgemeine Methode zur Abbildung verknüpfter, heterogener Fachmodelle vorgestellt und mit dem Multimodellcontainer wird ein hierzu korrespondierendes, allgemeines Datenformat zum Austausch von Multimodellen präsentiert. Multimodelle und Multimodellcontainer werden anschließend für die Domäne des Bauprojektmanagements weiter spezialisiert. Dabei kommen Vokabulare zur Beschreibung von Fach- und Linkmodellen zum Einsatz. Diese Informationen über Modelldomäne, Projektphase, Detaillierung und Status werden als Metadaten gespeichert. Mit dieser Methode lassen sich auch Multimodellvorlagen erstellen, die Spezifikationen für die zu erzeugenden Multimodelle darstellen.

---

## 2.1 Einführung

Mit dem Übergang von bauwerksorientierter zu prozessorientierter Arbeitsweise erlangt die domänenübergreifende Bereitstellung von Informationen generell wachsende Bedeutung, bspw. bei der Erstellung von Controllingkennwerten oder der Vorbereitung von Simulationen des Bauablaufs, der Baukosten oder des Bauprojektrisikos. Allerdings können die aktuellen Datenmodelle im Bauwesen diese Herausforderung nicht allein meistern.

Multimodellierung ist eine Methode, um beliebig austauschbare Bauinformationsmodelle, BIM, aus unterschiedlichen Fachmodellen zusammenzustellen und als Einheit zu verknüpfen. Hierbei sind alle Modelle gleichberechtigt. Ein führendes Modell ist nicht not-

---

J. Demharter (✉)

RIB Software AG, Stuttgart, Deutschland

S. Fuchs · S.-E. Schapke · R.J. Scherer

Institut für Bauinformatik, Technische Universität Dresden, Dresden, Deutschland

wendig. Eine bauwerksmodellzentrierte Modellbildung ist möglich (Teil I, Kap. 7), aber nicht notwendig. Multimodelle bestehen aus heterogenen Fachmodellen und expliziten, externen Links zwischen deren Elementen [1, 2]. Der Ansatz ermöglicht die Überbrückung der Anwendungsgrenzen einzelner Fachmodelle. Die aufgabenspezifische Kombination etablierter Datenmodelle erlaubt deren optimierten Einsatz in Hinblick auf semantische Praktikabilität, Konfliktfreiheit sowie Kommunikation, Koordination und Benutzerakzeptanz. BIM ist in diesem Kontext nicht nur als Bauwerksinformationsmodell zu verstehen, sondern als Bauprojektinformationsmodell, kurz Bauinformationsmodell, im Sinne der in Teil I, Kap. 1 dargelegten Bedeutung von BIM als Building Information Modelling. BIM Modelle sind im eigentlichen Sinn Multimodelle, denn sie formen die notwendigen, umfassenden Informationsräume für die einzelnen Bauaufgaben innerhalb der Bauprozesse und können Informationen aus beliebigen Fachgebieten kombinieren.

Multimodelle stellen somit einen vielversprechenden Ansatz dar, um die kollaborative Bearbeitung sowie die Analyse von modellbasierten Projektinformationen über die Grenzen von Organisationen und Fachdisziplinen hinaus zu unterstützen. Im Folgenden wird der allgemeine Multimodellansatz erläutert sowie das Datenschema und das Datenformat vorgestellt, die spezifiziert wurden, um den interoperablen Austausch der Multimodelle auf der Mefisto Plattform zu ermöglichen.

---

## 2.2 Allgemeiner Multimodellansatz

### 2.2.1 Das generische Multimodell

Für einen Austausch von domänenübergreifenden Informationen mit Multimodellen muss der Zugriff auf die zugrunde liegenden Daten geregelt werden. Ein wesentlicher Aspekt ist dabei die Datenstruktur zur Bündelung und Verlinkung von Fachmodellen. Mit dem generischen Multimodell werden ein Datenschema sowie zugehörige Bedingungen und Metadaten zum neutralen Austausch und Zugriff auf Multimodelle beschrieben [3]. Der Ansatz erlaubt, wie im Folgenden gezeigt, die Einbeziehung von Fachmodellen aus beliebigen Domänen. Hierfür ist das Konzept Fachmodell ebenfalls zu verallgemeinern und es wird das Konzept Elementarmodell eingeführt. Elementarmodelle werden in diesem Zusammenhang wie folgt definiert:

► **Definition** Ein Elementarmodell *EM* ist eine übertragbare Instanz eines Datenmodells, das einen abgegrenzten Informationsraum mittels einer vereinbarten Semantik beschreibt, der nicht weiter unterteilt wird.

Das bedeutet u. a., dass ein Elementarmodell nicht zwangsläufig ein korrespondierendes, explizites Schema benötigt. Bspw. sind XML Dokumente ohne XSD möglich. Lediglich die Bedeutung der Daten muss dem Empfänger in irgendeiner Form bekannt sein.

Des Weiteren wird das Konzept des Linkmodells eingeführt:

► **Definition** Ein Linkmodell *LM* ist eine serialisierbare, übertragbare Instanz eines Datenmodells mit zugehörigem Schema, das Referenzen zwischen Elementen verschiedener Elementarmodelle, sogenannte Links, speichert.

Linkmodelle ermöglichen somit die Speicherung und den Austausch von Verknüpfungen sog. Links zwischen Elementarmodellelementen in einer einheitlichen Art und Weise. Die Identifikation von Elementen erfolgt dabei über Elementidentifikatoren IDs, die innerhalb eines Elementarmodells eindeutig sind.

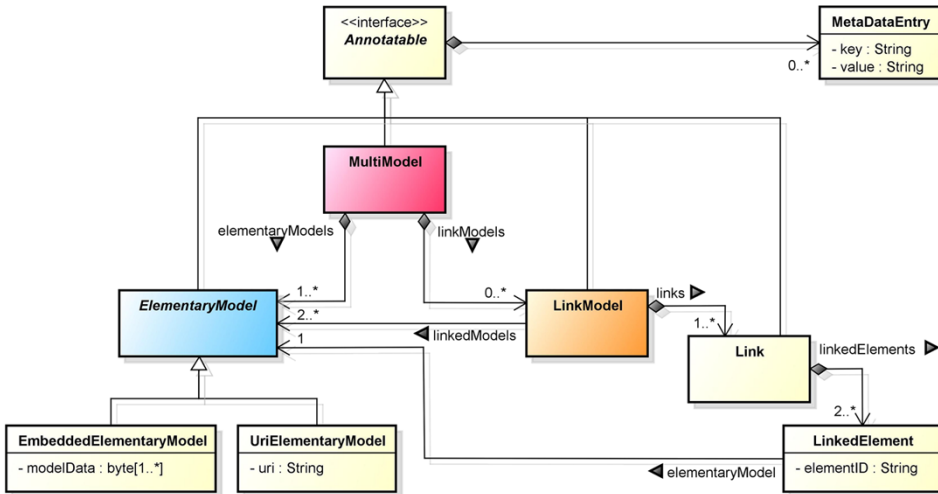
► **Definition** Ein Link *L* ist eine Menge von Identifikatoren von Elementen aus verschiedenen Elementarmodellen, die miteinander verknüpft sind.

Die Links werden außerhalb der Elementarmodelle gehalten. Somit können die Elementarmodelle von konventionellen Softwareprodukten weiterhin uneingeschränkt genutzt werden. Multimodellanwendungen ist es darüber hinaus möglich, die zusätzlichen Informationen der domänenübergreifenden Links auszuwerten.

► **Definition** Ein Multimodell *MM* ist eine serialisierbare, übertragbare Zusammensetzung aus einer nicht leeren Menge von Elementarmodellen *EM* und einer leeren oder nicht leeren Menge von Linkmodellen mit Elementen von *EM* als Subjekt.

Ausgehend von diesen Grundlagen wurde das generische Multimodell als objektorientiertes Datenmodell mithilfe des Ecore Meta Modells [4] entwickelt. Auf diese Weise konnten eine Java Implementierung sowie eine korrespondierende XML Serialisierung inklusive Parser automatisch aus dem Modell generiert werden. Abbildung 2.1 zeigt das resultierende Klassendiagramm nach [3].

Die Klasse *MultiModel* bildet den Einstiegspunkt in die Datenstruktur. Sie besteht aus Elementarmodellen, Klasse *ElementaryModel* und Linkmodellen, Klasse *LinkModel*. Elementarmodelle können entweder als Array von Bytes eingebettet, *EmbeddedElementaryModel* oder per URI referenziert *UriElementaryModel* werden, um die zu übertragende Datenmenge zu minimieren. In einem Linkmodell können mehrere Links, Klasse *Link*, zusammengefasst werden. Außerdem werden die in diesen Links verwendeten Elementarmodelle noch einmal explizit aufgeführt, Referenz *LinkModel::linkedModels*. Links sind generell mehrwertig, d. h. ein Link kann *n* Elemente aus *m* Elementarmodellen beinhalten ( $n \geq 2, m \geq 2, n \geq m$ ). Die Klasse *LinkedElement* repräsentiert ein solches Element. Es besteht aus einer ID und dem zugehörigen Elementarmodell. Da IDs nur innerhalb eines Elementarmodells eindeutig sein müssen, was bei vielen Modellen des Bauwesens der Fall ist, ergibt sich erst durch das Hinzuziehen des Elementarmodells ein eindeutiger Schlüssel bzw. eine globale ID. Allen Komponenten außer *LinkedElement* können Metadaten in Form von Schlüsselwertpaaren zugeordnet werden. Da dort beliebige Daten,



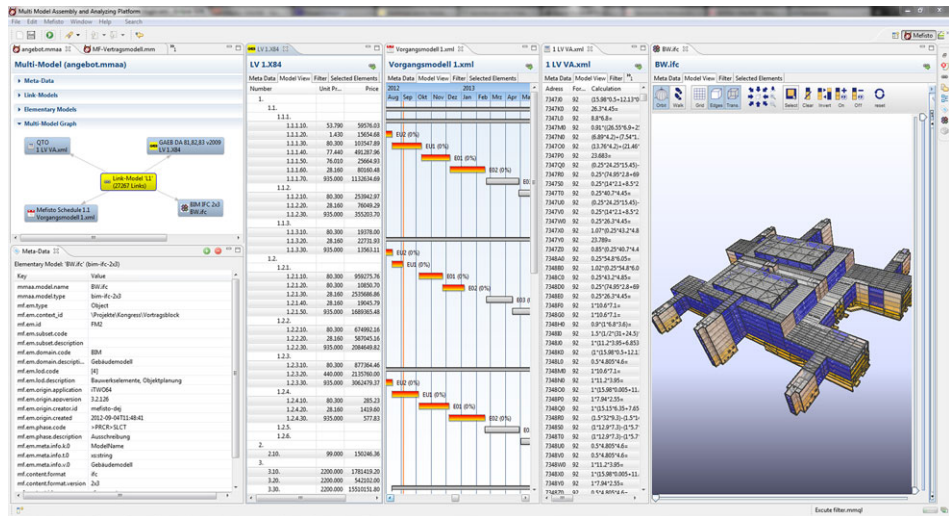
**Abb. 2.1** Schema des generischen Multimodells als UML Klassendiagramm

deren Semantik nicht im Multimodellschema definiert ist, platziert werden können, sollte deren Bedeutung in externen Katalogen erfasst werden, um diese Informationen eindeutig kommunizieren zu können.

Das generische Multimodell, d. h. das in Abb. 2.1 dargestellte Datenschema des Multimodells, wurde mittels der generierten Java Klassen in die M2A2 Multimodell Filter- und Analyse Plattform der TU Dresden (Teil II, Kap. 15) und der Mefisto Version von RIB iTWO der RIB Software AG (Teil II, Kap. 16 und 20) implementiert. In der M2A2 Plattform können auf dieser Grundlage Elementarmodelle, zugehörige Parser sowie Elementarmodellfilter, -viewer und -converter implementiert und getestet werden. Abbildung 2.2 zeigt ein Bildschirmfoto der Anwendung. Die Software basiert auf dem in [1] vorgeschlagenen Multimodellframework. Dieses ist um beliebige Elementarmodelltypen erweiterbar und erlaubt die Erstellung, Analyse, Bearbeitung, Filterung und Visualisierung von Multimodellen, die aus den entsprechenden Elementarmodellen bestehen können.

### 2.2.2 ID basierte Links

Die lose Kopplung von Elementarmodellen im Multimodell beruht auf der Verwendung von ID basierten Links. Dabei dienen Zeichenketten als Identifikator für Elemente. Zeichenketten sind leicht serialisierbar und behalten auch außerhalb des Elementarmodells ihre Bedeutung. Dies ermöglicht das externe Aufstellen, Speichern und Nachverfolgen von Elementreferenzen auf Datenebene. Der Multimodellansatz fordert demnach die Existenz von ID Attributen in jedem Elementarmodell. Für den Fall, dass keine ID Attribute vorhanden sind, bspw. bei unzulänglicher Modellqualität oder weil das Datenschema



**Abb. 2.2** Bildschirmfoto der M2A2 Multimodell Filter- und Analyse Plattform der TU Dresden. Geöffnetes Multimodell (o. l.), Metadaten des Bauwerksmodells (u. l.) sowie die Fachmodellviewer für Leistungsverzeichnis-, Terminplan-, Mengen- und 3D Bauwerksmodell (v. l. n. r.)

überhaupt keine solchen Attribute vorsieht, können nach [3, 5] unter anderem folgende Auswehlösungen angewendet werden:

### 1. Nichtmanipulative Auswehlösungen

Nichtmanipulative Auswehlösungen lassen das Elementarmodell unberührt und schließen aus den existierenden Daten auf die Identität eines Elements.

(a) In Relationalen Datenbanksystemen gespeicherte Elementarmodelle: Der Primärschlüssel des Elements kann direkt als ID verwendet werden. Zusammengesetzte Schlüssel können durch einen geeigneten Separator, bspw. Komma oder Semikolon, konkateniert werden. Im Multimodell sollen Metadateneinträge für das Elementarmodell dem Empfänger eines Multimodells den Zugriff auf diese Elemente vereinfachen. Die Methode ist auf andere Persistenzsysteme übertragbar, wenn diese eine inhärente, dauerhafte und als Zeichenkette repräsentierbare Identität ihrer Objekte bereitstellen.

(b) Feature Path mit Collectionindex: Die Identität eines Elements wird über den Navigationsweg in der Datenstruktur des Elementarmodells ermittelt. Dabei werden die Features, bspw. Referenzen oder Attribute, ausgehend vom Wurzelknoten in der angegebenen Reihenfolge ausgewertet. Mehrwertige Features können über die Angabe eines nullbasierten Collection Index, d. h. die Zählweise beginnt bei 0, berücksichtigt werden. Der Ausdruck `Root→Building(1)→Wall(17)` würde demnach die 18. Wand des 2. Gebäudes eines fiktiven Bauwerksmodells liefern. Ein nicht gesetzter Collectionindex liefert für mehrwertige Features das erste Element.

## 2. Manipulative Ausweichlösungen

Bei manipulativen Ausweichlösungen werden IDs neu generiert und im Elementarmodell gespeichert. Damit verstoßen die generierten IDs zwar gegen das Multimodellprinzip, Fachmodelle nicht zu verändern, sie sind jedoch eine Alternative für Fälle, in denen die Anwendung nicht manipulativer Methoden unmöglich ist. Hierbei kann man unterscheiden in manipulative Methoden, die mit dem Datenschema kompatibel sind (a, b) und solchen, die das Datenschema erweitern und damit zu missverständlichen Informationen führen können.

- (a) Das Elementarmodellschema sieht ID Attribute vor, sie sind in der Modellinstanz jedoch nicht belegt: Fehlende IDs können nach der Direktive des Schemas erzeugt und an den dafür vorgesehenen Stellen gesetzt werden.
- (b) Das Elementarmodellschema sieht keine ID Attribute vor. Falls das Datenformat Kommentare unterstützt, bspw. wie XML oder INI, können IDs generiert und dort gesetzt werden. Dafür sollte eine geeignete Zeichenkette der ID vorangestellt werden, um den speziellen Kommentar von regulären Anmerkungen zu unterscheiden.
- (c) Das Fachmodellschema sieht keine ID Attribute vor. ID Attribute können den Elementen hinzugefügt werden, falls das Datenformat dies generell unterstützt, z. B. XML oder JSON. Dazu muss ein einzigartiger und repräsentativer Attributname gewählt werden. Das Hinzufügen neuer Attribute verletzt allerdings das Elementarmodellschema und kann zu missverständlichen Informationen führen.

---

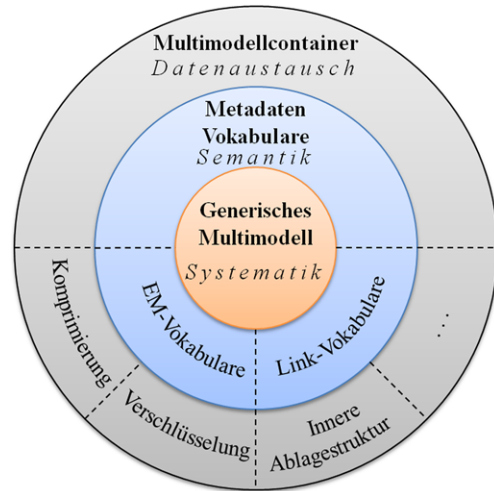
## 2.3 Multimodellspezialisierung

Das generische Multimodell bildet die Systematik für allgemeine Multimodelle ohne Beschränkung auf ein Anwendungsgebiet. Damit ist es sogar vollständig außerhalb des Bauwesens oder gemischt zur Verbindung von Bauwerksinformationsmodellen, BIM, mit beliebigen anderen Modellen einsetzbar. Dies erlaubt vielfältige Anwendungen, bspw. die Verbindung von Baustellenmodellen mit Stadtmodelldaten wie z. B. Google Earth's Keyhole Markup Language (Teil I, Kap. 7) [6] oder City Geography Markup Language [6] sowie meteorologischen Daten, bspw. im Format von EnergyPlus [7].

### 2.3.1 Multimodellraum

Werden dauerhaft Multimodellinformationen ausgetauscht, die zur gleichen Klasse von Bauinformationsprozessen gehören, ist es ratsam, diesen abgeschlossenen Informationsraum bzw. Multimodellinformationsraum, kurz Multimodellraum, zum Zweck der eindeutigen Datenübertragung abzugrenzen und näher zu beschreiben. Damit eröffnet sich auch die Möglichkeit, ein Multimodell zu spezifizieren, ohne dass es mit Daten gefüllt ist, was durch die in Abschn. 2.8 vorgestellten, Multimodellvorlagen realisiert wird. Es eröffnet weiterhin die Möglichkeit, Vorgaben vorzunehmen, nach denen eine Qualitätskontrolle von Multimodellen erfolgen kann (Teil I, Kap. 3, Teil II, Kap. 23). Je spezialisierter ein

**Abb. 2.3** Metaebenen der Multimodellspezialisierung



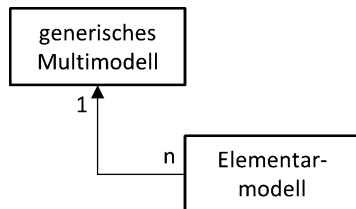
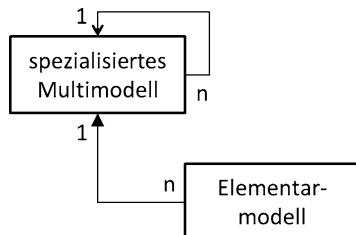
Informationsraum ist, umso höherwertigere Qualitätskontrollen können durchgeführt werden. Dieser Vorgang wird Spezialisierung genannt. Eine Spezialisierung wurde auch für das IFC Modell vorgenommen, auf das in Kapitel A3 eingegangen wird.

Abbildung 2.3 zeigt die übergeordneten Ebenen, an denen eine Konkretisierung des Multimodelldatenaustauschs vorgenommen werden kann. Der innere Ring steht für das generische Multimodell. Dessen Systematik der gleichgestellten, heterogenen und originalen Elementarmodelle sowie externer Linkmodelle ist die konstante Grundlage jeglicher Multimodelle. Der zweite Ring symbolisiert die semantische Beschreibung eines speziellen, abgegrenzten Informationsraums durch die Aufstellung eines kontrollierten Vokabulars für die Multimodellmetadaten. Der dritte Ring symbolisiert die Möglichkeit zur Definition eines individuellen Datenformats für den physischen Datenaustausch. Im Projekt Mefisto erfolgte eine Multimodellspezialisierung für den Informationsraum des Bauprojektmanagements. Die Eigenschaften des dabei entstandenen Multimodellcontainers werden im nachfolgenden Abschn. 2.4 beschrieben.

### 2.3.2 Multimodelltaxonomie

Die Grundtaxonomie der Multimodelle ist zweistufig, wie aus dem Multimodellschema in Abb. 2.1 ersichtlich ist. Die Taxonomie besteht aus den drei Konzepten, dem generischen Multimodell, dem Elementarmodell und dem Linkmodell (Abb. 2.4). Es bildet den Mechanismus, aus mehreren Elementarmodellen ein Multimodell zu formen. Elementarmodelle sind die Objektmenge der Multimodelle, Linkmodelle sind die Relationsmenge.

$$gMM = \{EM, LM\} \quad (2.1)$$

**Abb. 2.4** Grundtaxonomie**Abb. 2.5** Taxonomie des spezialisierten Multimodellraums

Die Elementarmodelle sind objektorientierte Modelle mit einer Objektmenge  $EO$  und einer Relationsmenge  $ER$ , die abgesehen von der Referenzierung externer Informationsquellen, wie z. B. Kataloge, abgeschlossen ist.

$$EM = \{EO, ER\} \quad (2.2)$$

Wird die Definition des generischen Multimodells nicht nur durch die Spezialisierung erweitert, sondern wird auch erlaubt, dass Multimodelle selbst Elemente von Multimodellen bilden können, ist es möglich, eine Taxonomie mit beliebiger Tiefe zu bilden,

$$MM_i = \{\{MM_j, EM\}, LM\}, \quad \forall MM_i \neq MM_j \quad (2.3)$$

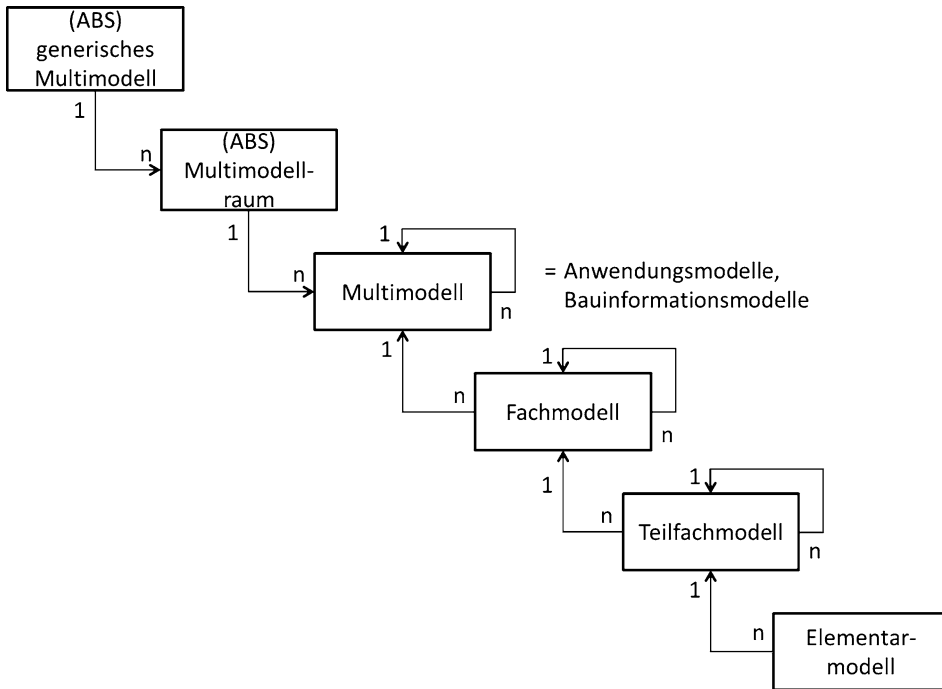
wobei gilt, dass das instantiierte Multimodell sich nicht selbst enthalten darf. Hierbei ist wieder zu fordern, dass die Multimodelle  $MM_j$ , die ein neues Multimodell  $MM_i$  bilden, einen abgeschlossenen Informationsraum abbilden (Abb. 2.5). Dies entspricht dem in der Praxis vorhandenen Konzept des Fachbereichs oder des Fachgebiets. Entsprechend werden diese Multimodelle in der Praxis auch als Fachmodelle bezeichnet.

► **Definition** Ein Fachmodell  $FM$  ist eine übertragbare Instanz eines Datenmodells, dessen Informationsraum ein Fachgebiet oder eine Fachdomäne darstellt. Es kann ein Elementarmodell als auch ein Multimodell sein.

$$FM \in \{MM, EM\} \quad (2.4)$$

Damit ist es möglich, das Konzept Teilfachmodell einzuführen, wie es häufig in der Praxis benutzt wird, das sich von Fachmodellen dadurch unterscheidet, dass der Informationsraum nicht durch das Fachgebiet begrenzt wird, sondern nur einen Teil des Fachgebiets repräsentiert. Es bleibt aber weiterhin die Forderung bestehen, dass es einen abgeschlossenen Informationsraum repräsentiert. Dies bedeutet, alle Links des Informationsraums



**Abb. 2.6** Modelltaxonomie

sind im Teilfachmodell enthalten (Abb. 2.6). In der Praxis gibt es eine Vielzahl weiterer Begriffe, wie Unterfachgebiet, sog. Subdomain, etc., die aber keine neue informatorische Bedeutung haben, sondern nur Bezeichnungen sind, um die geschachtelten Fachmodelle in den damit gebildeten Fachmodellebenen mit eindeutigen Namen zu bezeichnen.

Zu unterscheiden sind hiervon Sichten, sog. Views, von Fachmodellen, also durch Filter erzeugte Teilmengen der Fachmodelle. Diese unterliegen nicht der Forderung, dass sie durch einen Informationsraum vollständig abgeschlossen sind. Sie können beliebige Teile eines Fachmodells abbilden und werden deshalb als Teilmodelle bzw. Fachteilmodelle bezeichnet. Entsprechend beinhalten sie Links, die über das Fachteilmodell hinausreichen und als Konsequenz beim Filtern abgeschnitten werden. Für das Abschneiden bestehen verschiedene Möglichkeiten, wie z. B. direktes Abschneiden aller Relationen oder weitere Betrachtung von bestimmten Relationen oder die Beschränkung der Relationskettenglieder, d. h. der Relationstiefe [8, 14], was auch in Teil I, Kap. 14 kurz angesprochen ist. Diese Links bedürfen bei Änderung im Teilmodell oder bei der Vereinigung von Teilmodellen einer besonderen Beachtung, die einer der derzeit größten und noch unvollständig gelösten Probleme der Synchronisation und damit des Versionsmanagements von Modellen darstellen [8].

Mit der Möglichkeit der Spezialisierung des generischen Multimodells wird die Definition des allgemeingültigen, generischen Multimodells eingeschränkt, das spezialisierte

Multimodell jedoch aussagekräftiger. Ein spezialisiertes Multimodell sollte ein übergreifendes Fachgebiet wie z. B. Baumanagement abbilden, in dem viele Multimodelle existieren wie z. B. Multimodell Ausschreibung, Angebot, Vertrag, Bauübergabe, etc. Das spezialisierte Multimodell wird daher als Multimodellraum bezeichnet, wie in Abschn. 2.3 beschrieben ist. Damit kann die in Abb. 2.6 dargestellte Taxonomie gebildet werden. Das generische Multimodell als auch der Multimodellraum sollten als abstrakte Klassen angesehen werden.

Ein Elementarmodell muss nicht absolut gesehen ein Modell sein, das nicht weiter in verlinkte Untermodelle aufgebrochen werden kann. Diese Eigenschaft als elementares Modell gilt nur im zugehörigen Multimodellraum. Gehört das Elementarmodell zu einem anderen Multimodellraum, so kann es zu einem Fachmodell werden, das aus anderen Modellen bestehen kann. Ebenso kann der Modellraum weiter spezialisiert werden. Als Konsequenz können dabei einzelne Elementarmodelle zu Fachmodellen bzw. auch zu Multimodellen werden.

Im Allgemeinen annotiert ein Multimodellsender im Zuge der Spezialisierung die Fach- und Linkmodelle mit Metadaten, welche die Semantik der Daten beschreiben, um eine Fehlinterpretation beim Multimodellempfänger zu vermeiden. Die Semantik der Daten ist ihre Klassifikation durch Klassennamen bzw. entspricht den Klassen des Datenmodellschemas. Sind die Bauwerkselemente nur durch geometrische Daten beschrieben, liegt noch kein semantisches Datenmodell vor. Die Bauwerkselemente haben nur eine Form, aber noch kein Wissen über ihre Funktionen. Dies wird ihnen erst über die Semantik mitgegeben. Eine noch mächtigere Form als die Semantik stellt die Ontologie dar, die in Teil II, Kap. 11 und 12 für die Prozesskonfiguration eingesetzt wird. Die Metadaten nutzen dabei das vereinbarte Vokabular des speziellen Informationsraumes. Dabei kann bspw. eingeschränkt werden, welche Fachmodelle in welchen Formaten genutzt werden und eine Klassifizierung der Fachmodelle hinsichtlich relevanter Parameter des Informationsraums, z. B. Domäne, Prozessfortschritt oder Detaillierungsgrad (Teil I, Kap. 1) erfolgen. Linkmodellen kann auf diese Weise ein spezialisierter Linktyp (Teil I, Kap. 4) zugeordnet werden, der den Zweck der Relation benennt.

---

## 2.4 Das spezialisierte Mefisto Multimodell

Im Gegensatz zum allgemeingültigen Datenschema des generischen Multimodells kann das spezialisierte Mefisto Multimodell des Multimodellraums Bauprojektmanagement viel speziellere Informationen über die Fach- und Linkmodelle des Bauprojektmanagements bereits auf Schemaebene abbilden. Da bei einem generischen Ansatz keinerlei fachliche Semantik im Schema modelliert werden darf, muss dieses im spezialisierten Multimodell mithilfe extern spezifizierter Metadaten, bspw. in Vokabularkatalogen, annotiert werden. Dies erschwert einerseits die Implementierung fachspezifischer Multimodellanwendungen, da semantische Spezifikationen nur informal oder semiformal und über unterschiedliche Dokumente verteilt vorliegen. Andererseits können so ad hoc neue Vokabulare auf-

gebaut und verändert werden, was in der Phase des Testens einer neuen Methode sehr wichtig ist, u. a. um ein optimiertes Vokabular zu finden, das dann zu einer Normierung führt. Langfristig gesehen können und sollen solche Vokabularkataloge standardisiert werden. Ein alternativer Ansatz hierzu ist, ein standardisiertes Ontologiemodell aufzustellen, wie es in Teil I, Kap. 4 entwickelt worden ist, das im Vergleich zu einem Vokabularkatalog wesentlich mächtiger ist, jedoch auch größeren Implementierungs- und Pflegeaufwand und zuvor entsprechende Absprachen bzw. Standardisierungen benötigt.

Im Mefisto Multimodellraum sind die fachlichen Belange des Bauprojektmanagements direkt modelliert. Das Mefisto Multimodell ist mehr als nur ein Modell. Der Begriff steht für den Multimodellraum Bauprojektmanagement, bestehend u. a. aus den Multimodellen Ausschreibung, Angebot, Vertrag, Baustelle, um nur die vier wichtigsten zu nennen. So existieren für Fachmodelle bspw. die Attribute für Projektphase, Bearbeitungsstatus und Level of Detail. Diese Informationen sind in anderen Informationsräumen, z. B. Gebäudeautomation in der Betriebsphase, möglicherweise gar nicht relevant. Auch die Domäne der Fachmodelle wird aus einer abgeschlossenen Liste erlaubter Werte gewählt. Dies ist möglich, da der spezialisierte, abgeschlossene Informationsraum des Bauprojektmanagements zuvor systematisch analysiert [9] und ein Ontologieframework (Teil I, Kap. 4) und ein Vokabularkatalog entworfen wurde. Zum Entwurfszeitpunkt der Mefisto Multimodellbeschreibung standen daher alle Fachmodelltypen und deren Bedeutung und Datenformate fest. Der erste detaillierte Entwurf des Vokabularkatalogs ist in [9] publiziert.

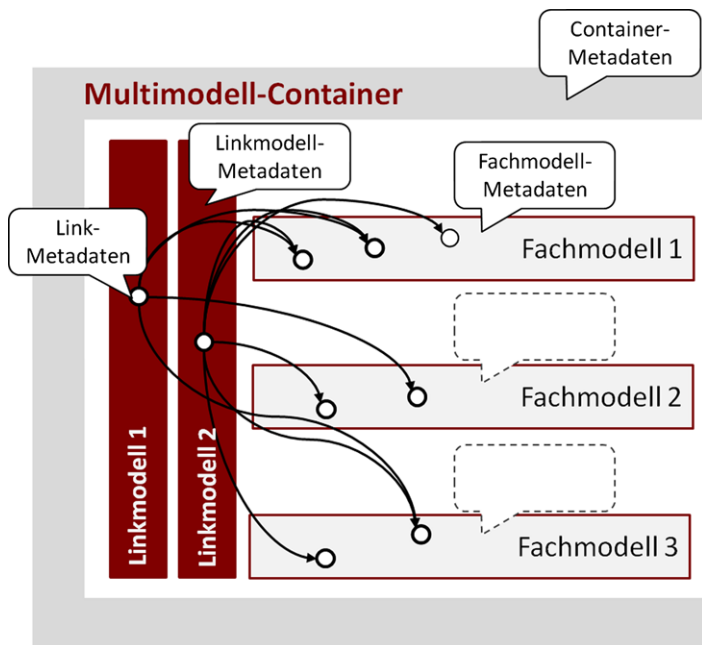
Die direkte Modellierung semantischer Daten im Mefisto Multimodellschema erleichtert die Implementierung fachspezifischer Anwendungen für das Bauprojektmanagement wie RIB iTWO, GRANID oder SiteSimEditor (Abb. 2.8 o. r.). Da relevante Attribute nun formal spezifiziert wurden, können dort mithilfe von Programmierwerkzeugen und Standardbibliotheken automatisiert Klassenstrukturen und Parser erzeugt und eingebunden werden. Des Weiteren sind Mefisto Multimodellcontainer wesentlich einfacher zu validieren, da semantische Daten nun explizit und automatisiert gegen ihr Schema geprüft werden können und nicht implizit und mit manuell erstellten Regeln gegen eine semiformale Spezifikation geprüft werden müssen. Im Folgenden wird die Multimodellspezialisierung des Projekts Mefisto detailliert beschrieben.

Seit einigen Jahren werden verschiedene 4D und 5D Softwareanwendungen angeboten, mit denen Bauwerksmodelle und andere Planungs- und Controllingmodelle verbunden und analysiert werden können. Während Softwareanwendungen wie Navisworks® von Autodesk Inc., RIB iTWO® von RIB Software AG, Synchro® von Synchro Ltd. oder Vico Office Suite® von Vico Software, Inc. den Import einer Vielzahl von Dateiformaten ermöglichen, gibt es bis heute jedoch keine einheitliche Form für den Austausch der dabei erstellten Multimodelle.

---

## 2.5 Multimodellcontainer

Für den Austausch von Multimodellen für Bauprojektmanagementinformationen wurde im Projekt Mefisto ein spezialisiertes Mefisto Multimodellschema mit einem expliziten



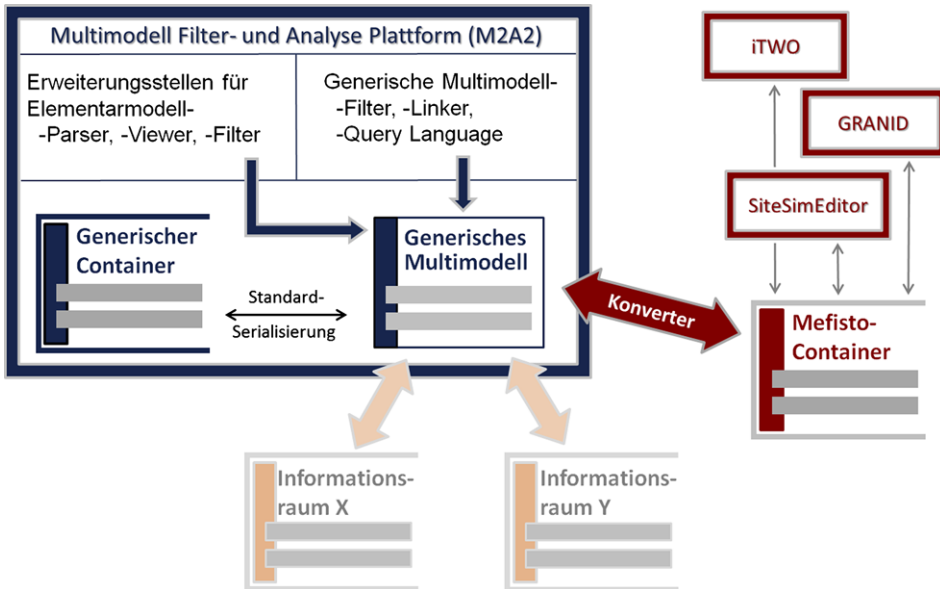
**Abb. 2.7** Prinzipieller Aufbau des Multimodellcontainers

Linkmodell (Abschn. 2.2) und ein spezialisierter Mefisto Multimodellcontainer als Austauschformat entwickelt. Im Gegensatz zu Datenstandards für einzelne Fachmodelle, wie z. B. die IFC für Bauwerksinformationsmodelle oder GAEB DA für Leistungsverzeichnisse, definiert der Multimodellcontainer lediglich eine Grundstruktur, in die verschiedene Fachmodelle und zugehörige Linkmodelle eingeordnet werden können. Die einzelnen Fachmodelle des Containers können somit nicht nur mit speziellen 4D und 5D Anwendungen, sondern immer auch mit den herkömmlichen Fachanwendungen bearbeitet werden.

Der physische Datenaustausch erfolgt über serialisierte Multimodelle, sog. Multimodellcontainer, d. h. ein Multimodell wird über einen Multimodellcontainer ausgetauscht. Dabei liegt der Multimodellcontainer als eine komprimierte Datei vor. Hier werden die Fachmodelle selbst bzw. die Beschreibung ihres Speicherorts, die Linkmodelle sowie die Metadaten abgelegt. Die dazu notwendige Struktur, eine denkbare Verschlüsselung oder Datenkomprimierung sind mögliche Parameter zur Konfiguration eines spezialisierten Multimodellcontainers. Der Multimodellcontainer ist in XML formalisiert. Abbildung 2.7 zeigt den prinzipiellen Aufbau des Mefisto Multimodellcontainers.

Die Multimodellcontainerdatei mit der Dateinamenerweiterung mmc besteht dabei aus den Komponenten:

- Containerbeschreibung in Form der obligatorischen Datei MultiModel.xml in der MMC Root



**Abb. 2.8** M2A2 – Übergang zu abgeschlossenen Informationsräumen mittels Konverter [1]

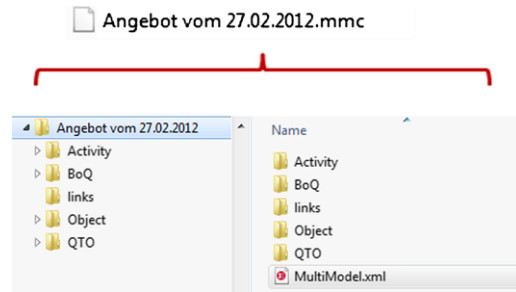
- einzelne Dateien für den Austausch der Fachmodelle
- einzelne Dateien für den Austausch der Linkmodelle

Die Multimodellcontainerdatei kann beliebig viele Linkmodelle mit beliebig vielen Links enthalten sowie Metadaten für jedes Fachmodell, jedes Linkmodell, jeden Link und den Container selbst. Das XML Datenmodell ist ausschnittsweise in Abschn. 2.7 wiedergegeben.

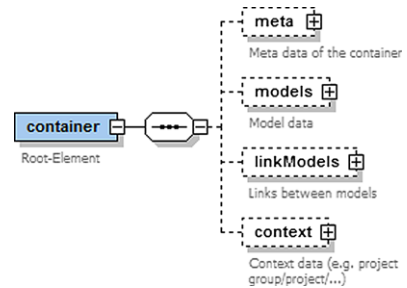
## 2.6 Filter- und Analyseplattform M2A2

Die allgemeingültige und universelle M2A2 Plattform (<http://www.youtube.com/watch?v=fkzrmRy3W58>) stellt die Verbindung zu spezialisierten, abgeschlossenen Informationsräumen über Konverter her (Abb. 2.8). Dadurch entsteht zwar ein einmaliger Programmieraufwand, dieser ist jedoch wesentlich geringer als die wiederholte Neuimplementierung der generischen Multimodelloperationen wie Filter, Linker oder der Multimodellabfragesprache, Multimodell Query Language, MMQL (Teil II, Kap. 15). Somit bleibt der hohe Aufwand für die allgemeingültigen, multimodell- und linkspezifischen Algorithmen in M2A2 geschützt, und die Plattform kann ohne Änderung auch in weiteren spezialisierten Multimodellinformationsräumen, bspw. für Energieeffizienz- oder Infrastrukturinformationsprozesse, eingesetzt werden. Der primäre Anwendungszweck der M2A2 Plattform als universelles Test- und Analysewerkzeug für beliebige Multimodelle bleibt somit gewahrt.

**Abb. 2.9** Mefisto  
Multimodellcontainer  
Ausschreibung in der  
Browseransicht



**Abb. 2.10** Datenstruktur der  
Multimodellcontainerbeschrei-  
bung

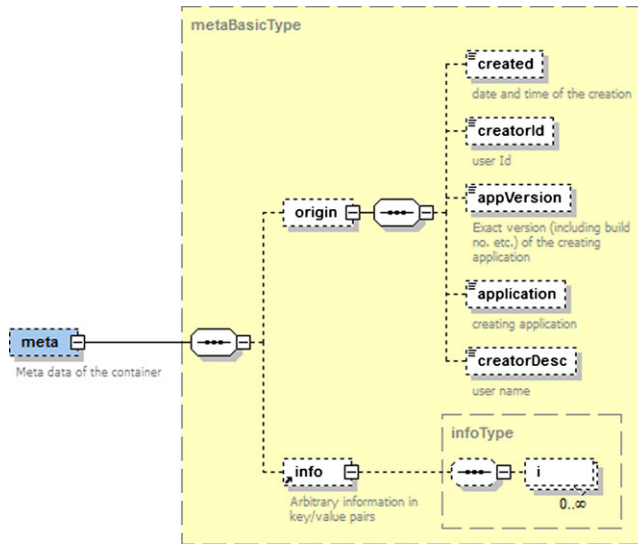


## 2.7 Datenmodell des Multimodellcontainers

Abbildung 2.9 zeigt beispielhaft einen Multimodellcontainer für eine Ausschreibung. Auf der Mefisto Plattform bestehen Multimodellcontainer aus mehreren Dateien, die in einer komprimierten Archivdatei ausgetauscht werden. Die Annotationsklassen bzw. -kriterien wurden in Teil I, Kap. 1, Gl. 1.1 bis 1.3 eingeführt. Die prozessorientierte Verwendung der Annotationskriterien wird in Teil II, Kap. 13 analysiert und praxisrelevante Wertebereiche werden vorgeschlagen. Im Folgenden wird das Schema der Annotationsklassen in XML, die die Kernmetadatenstruktur des Multimodellcontainers bildet, entwickelt. Ein Schema in XML wird auch als XSD, XML Schema Definition bezeichnet.

Wurzelement des Datenmodells der Metadaten ist die Klasse Container, welche die Referenzen zu den Fach- und Linkmodellen sowie die zugehörigen Metadaten enthält. Die Schemas der Metadaten sind in den nachfolgenden Abschnitten detailliert dargestellt. Im Weiteren wird in allen Kapiteln des Buchs vereinfacht vom Multimodellcontainer gesprochen, auch wenn es sich meistens um den hier spezialisierten Mefisto Multimodellcontainer handelt.

Das Datenmodell bzw. das Schema des Multimodellcontainers bildet die Grundlage für die Multimodellcontainerbeschreibung. Sie gliedert sich in vier Hauptbereiche zur Angabe von Container Metadaten <meta>, Fachmodellen <models>, Linkmodellen <linkModels> und zusätzlichen Kontextinformationen <context> (Abb. 2.10).



**Abb. 2.11** Datenstruktur der allgemeinen Multimodellcontainermetadaten

### 2.7.1 Formalisierung der Metadaten

In allen vier Bereichen können die Inhalte durch Multimodellcontainermetadaten zur deren Erstellung `<origin>` sowie durch frei zu vergebende Schlüsselwertepaaren ergänzt werden `<info>`, wie es in Abb. 2.11 für die Metadaten des Multicontainers (Abb. 2.7) dargestellt ist.

Die Fachmodelle `<model>` und die Linkmodelle `<linkModel>` können darüber hinaus durch weitere Metadaten zur fachlichen Beschreibung der Modellinhalte ausgezeichnet werden (Tab. 2.1). Abbildung 2.12 stellt die entsprechenden Metadatenelemente bzw. fachlichen und organisatorischen Annotationen für Fachmodelle und Linkmodelle im XML Schema, d. h. einer XSD der Multimodellcontainerbeschreibung dar.

### 2.7.2 Formalisierung der Fachmodelle

Neben den Metadaten stellen die Fachmodelle einen wesentlichen Inhalt der Multimodellcontainerbeschreibung dar. Die Definition der einzelnen Fachmodelle `<model>` enthält die Verweise auf die Inhalte in Form der Bereiche `<content>`, die mehrfach je Fachmodell zulässig sind, d. h. es darf dasselbe Fachmodell in verschiedenen Datenformaten und damit redundant ausgetauscht werden. Inwieweit diese redundanten Daten widerspruchsfrei sind, liegt in der Verantwortung des Senders.

Über den Bereich `<contentOptions>` sind die einzelnen Formatinhalte eines Fachmodells näher beschrieben. Im nachfolgenden Beispiel wird hier die Datenaustauschphase des Leistungsverzeichnisses, `type="BoQ"`, konkretisiert. Dazu zählen das Format GAEB

**Tab. 2.1** Strukturierung der Metadaten von Fachmodellen und Linkmodellen

Metadatenklasse		Vokabularbeschreibung
Projektphase	<phase>	Hierarchische Klassifikation von Projektphasen und ihren Leistungsaktivitäten auf Basis der HOAI und detaillierter Prozessanalysen [9] entspricht Index p, nach Teil I, Kap. 1, Gl. 1.2
Modelldomäne	<domain>	Klassifikation von Fachmodellen nach Softwareanwendungen (tool), Domäne Datenmodell, Datenformaten und Praxiseinsatz matter, Indizes <i>d, k, f, m</i>
Status	<status>	Bearbeitungsstatus eines Modells [Request, Draft, Approved, Deprecated][Angefordert, Entwurf, Angenommen, Abgelehnt], Index <i>s</i>
Detaillierung	<levelOfDetail>	Vokabular von Detailstufen in Bezug auf das Bauwerksmodell in Analogie zu CityGML [6] Index <i>l</i>
Organisation	<origin>	Softwareanwendung (tool), Ersteller (actor), erstellt (created), Indizes <i>t, a, c</i>

DA84 [10] sowie die Definition bzgl. des Datenaustauschs der Einheitspreisanteile (unitrate):

```
...
<model contextId="\Projekte\Mefisto\M02 V1
(Angebot)\Projektvarianten\01" type="BoQ"
id="FM1">
  <meta/>
  <content format="gaebxml" id="1" formatVersion="3.1">
    <contentOptions>
      <i k="description" t="xs:boolean" v="0"/>
      <i k="extension" t="xs:string" v="DA84"/>
      <i k="qtosplit" t="xs:boolean" v="0"/>
      <i k="unitrate" t="xs:boolean" v="1"/>
      <i k="unitrate_breakdown" t="xs:boolean" v="1"/>
    </contentOptions>
    <file namespace="BoQ1">file:///BoQ/1/LV 1.X84</file>
  </content>
</model>
...
```

Die mehrfache Bereitstellung eines Fachmodells über verschiedene Datenaustauschformate erfolgt über die Definition entsprechend vieler <content> Bereiche. Im nachfolgenden Beispiel enthält der Multimodellcontainer das Fachmodell Bauwerksmodell, type=“Object“ als Datenaustauschformat ifc2×3 und cpixml. Die Differenzierung erfolgt über das Attribut id:

```
...
<model contextId="\Projekte\Mefisto\M02 V1
(Angebot)\Projektvarianten\01" type="Object" id="FM2">
  <meta/>
  <content format="ifc" id="c1" formatVersion="2x3">
```



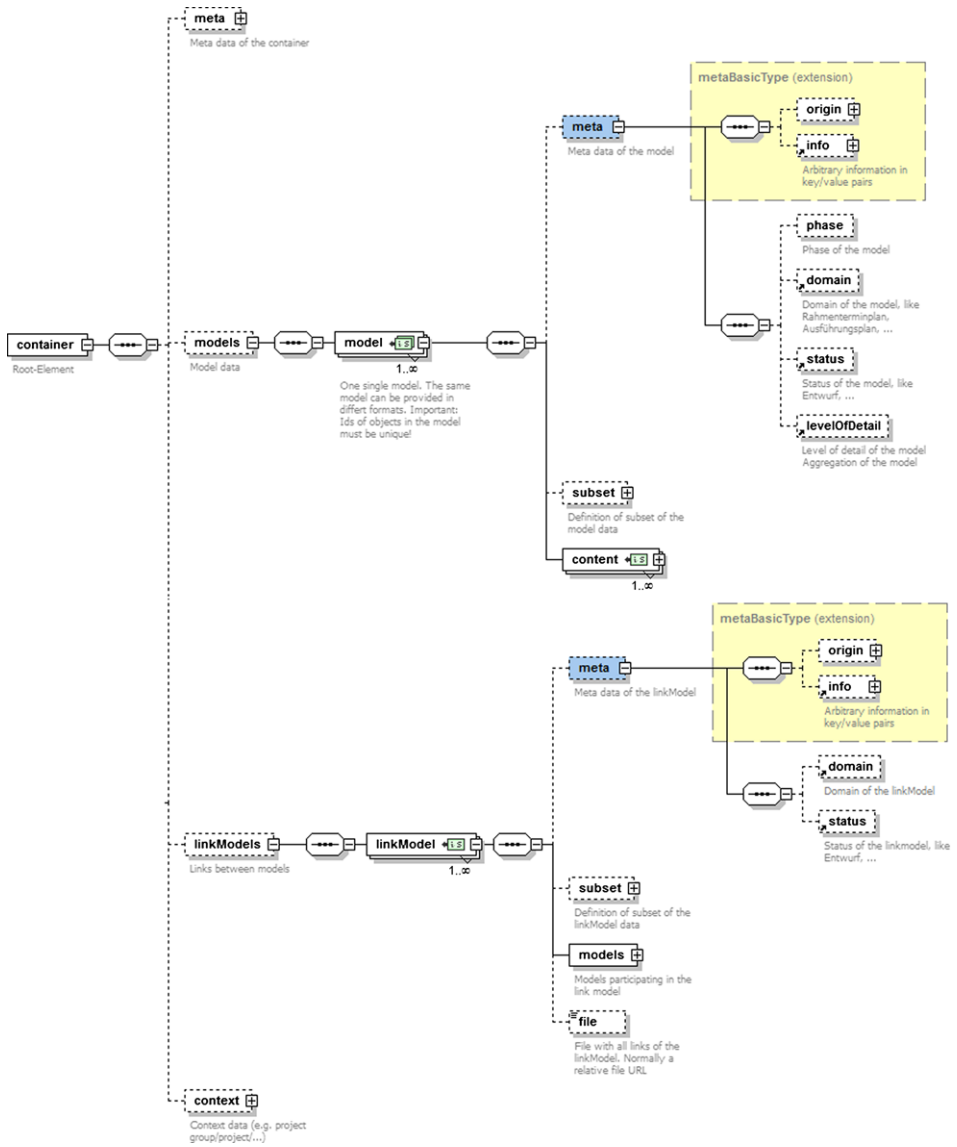


Abb. 2.12 Datenstruktur der Metadaten von Fach- und Linkmodellen im Multimodellcontainer

```
<file>file:///Object/c1/BW.ifc</file>
</content>
<content format="cpixml" id="c2" formatVersion="1.4">
  <file>file:///Object/c2/BW.cpixml</file>
  <file>file:///Object/c2/buildingStructure.cpixml</file>
</content>
</model>
...
```

Die Einschränkung der Dateninhalte eines Fachmodells erfolgt über die Definition eines <subset> Bereichs. Neben der programmübergreifenden Verständigung auf Filter für Erzeugung von bspw. Leistungsverzeichnis- (LV), voraussichtlichen Ausführungs- (VA), Abrechnungs- (RE) oder Leistungsmengen (LE) können auch programmindividuelle, adaptierte Filterausdrücke hinterlegt werden. Damit können diese Regeln von korrespondierenden Softwareprodukten automatisch interpretiert werden. Das nachfolgende Beispiel zeigt die Definition eines mit WQ benannten LV Mengenfilters für die Multimodellanwendungen M2A2 und RIB iTWO:

```
...
<model contextId="\Projekte\Mefisto\M02 V1
(Angelbot)\Projektvarianten\01" type="QTO" id="FM3">
  <meta/>
  <subset subsetCode="LV" subsetDesc="LV-Mengen">
    <subsetFilter application="M2A2">WQ</subsetFilter>
    <subsetFilter application="iTWO">WQ</subsetFilter>
  </subset>
  <content format="xml" id="1" formatVersion="1.0">
    <file namespace="QTO1">file:///QTO/1/1 LV VA.xml</file>
  </content>
</model>
...
```

### 2.7.3 Formalisierung der Linkmodelle

Als zweite Hauptkomponente stehen neben den Fachmodellen die Linkmodelle in der Containerbeschreibung zur Verfügung.

Im nachfolgenden Beispiel ist das Linkmodell als Mengensplitmodell type="QuantitySplit" zwischen den Fachmodellen Leistungsmodell, FM1, Bauwerksmodell, FM2, Mengenmodell, FM3 und Terminmodell, FM4 spezifiziert. Die Daten der einzelnen Links sind durch den Verweis auf die Linkdatei, <file>, definiert:

```
...
<linkModel type="QuantitySplit" id="L1">
  <meta/>
  <models>
    <model id="FM1"/>
    <model id="FM2"/>
    <model id="FM3"/>
    <model id="FM4"/>
  </models>
  <file>file:///links/links.xml</file>
</linkModel>
...
```

### 2.7.4 Spezifikationen für Fachmodelle

Die Fachmodelle im Container werden als unabhängige Informationsressourcen mit eigenem Datenschema und eigener Datenformalisierung behandelt, um die flexible Nutzung

**Tab. 2.2** Fachmodelle und Datenformate, die im Rahmen vom Projekt Mefisto unterstützt werden

Modelldomäne	Inhalte	Formate
Bauwerksmodelle	Bauwerke mit funktionalen, geometrischen oder topologischen Eigenschaften	IFC 2×3 [11], cpixml [12]
Baustellen-einrichtungsmodelle	Infrastrukturelle Einrichtungen, Geräte und Baustoffe mit geometrischen, funktionalen und leistungsbezogenen Parametern	Erweiterung IFC 2×3 XML Format für Gerätekataloge auf Basis BGL [5]
Organisationsmodelle	Akteure sowie deren Zuständigkeiten und Informationssysteme im Bauprojekt	XML Format auf Basis ifcXML 2×3 [13]
Leistungsmodelle	Qualitative und quantitative Beschreibungen von Bauleistungen	GAEB DA XML XML Format auf Basis REB VB 23.003 [10]
Kostenmodelle	Kostenansätze in Kostenplanung und Kostensteuerung, Kalkulation, Vergabe und Kostencontrolling	GAEB DA XML Neues XML Format für Kalkulationen
Terminmodelle	Vorgänge, Ereignisse und ihre Beziehungen sowie zugehörige Ressourcen und Kalender	XML Format auf Basis ifcXML 2×3 Neues XML Format für Terminpläne
Risikomodelle	Ereignisse und Maßnahmen mit kritischem oder positivem Einfluss auf die Projektergebnisse	Neues XML Format für Risikokataloge und -listen

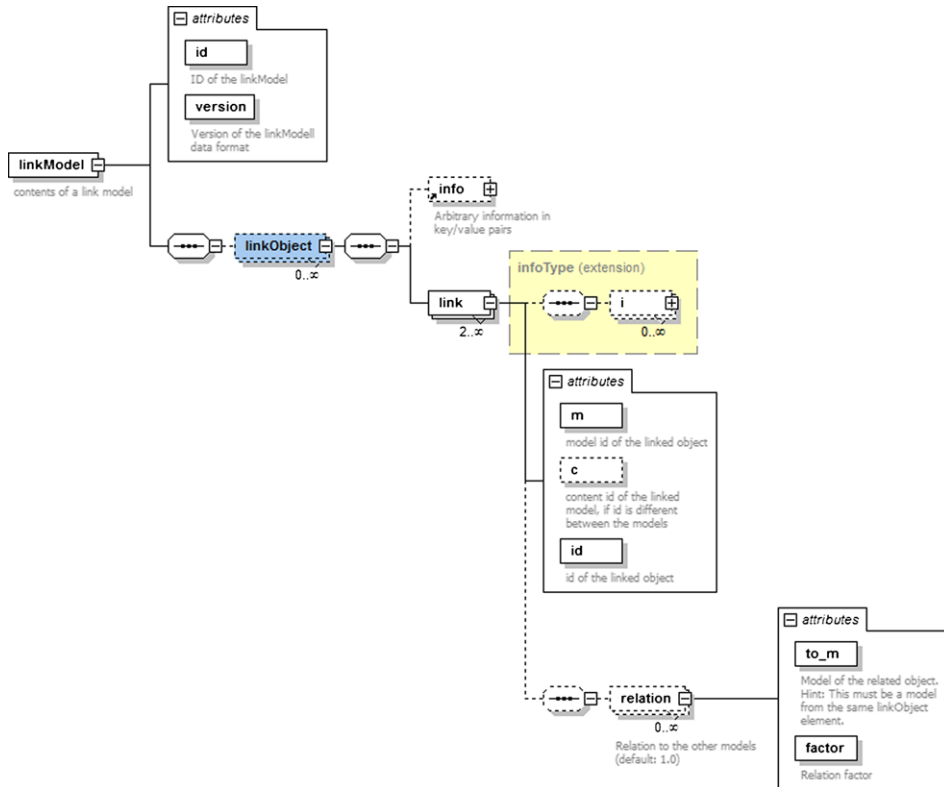
etablierter Datenformate zu ermöglichen. Für die weitere Unterstützung des interoperablen Datenaustausches können zusätzliche Leitlinien für deren Strukturierung und Formalisierung der Fachmodelle festgelegt werden.

Es wurden auf der Grundlage einer Szenarioanalyse [9] die in Tab. 2.2 aufgeführten Fachmodelldomänen und Datenformate ausgewählt. In einem Bauprojekt können solche Vorgaben im Rahmen allgemeiner Projekthandbücher definiert oder auch formal durch die in Abschn. 2.8 beschriebenen Multimodellvorlagen spezifiziert werden.

**2.7.5 Spezifikationen für Linkmodelle**

Die Linkmodelle stellen eine Liste von Linkelementen dar, die Linkinterdependenzen eines bestimmten Typs unter einer Reihe von Fachmodellen aufzeigen. Grundsätzlich wird dabei unterschieden zwischen (Teil I, Kap. 1, Abb. 1.2):

- horizontalen Interdependenzen unter Fachmodellen unterschiedlicher Domänen und Formate



**Abb. 2.13** Struktureller Aufbau des Linkmodells

- vertikalen Interdependenzen unter Fachmodellen unterschiedlicher Detaillierungen
- longitudinalen Interdependenzen unter Fachmodellen unterschiedlicher Versionen

Auf der Mefisto Plattform sind die Linkmodelle in separaten XML Dateien gespeichert, die auf dem neu entwickelten Linkmodellschema basieren, das in Abb. 2.13 dargestellt ist.

Die Linkelemente `<linkObject>` im Linkmodell sind als mehrwertige Relationen konzipiert und enthalten jeweils einen Satz zusammengehörender, mehrwertiger Links `<link>`, mit denen  $n$  ausgewählte Elemente aus  $m$  Fachmodellen referenziert werden. Da IDs nur innerhalb eines Fachmodells eindeutig sein müssen, wird ein verlinktes Element durch seine ID im Attribut `id` und sein Fachmodell im Attribut `m`, bei Partialmodellen oder Alternativformaten zusätzlich im Attribut `c` beschrieben. Darüber hinaus können die einzelnen Links mit Zusatzinformationen versehen werden, um bspw. ID Konflikte zu behandeln oder die Mengensplanteile eines Bauteils an einer verlinkten Mengenmodellposition anzugeben. Wie in Abb. 2.13 dargestellt, können auch wieder beliebige Schlüsselwertpaare hinzugefügt werden. Die Kombination aller Links eines Linkelements muss eindeutig sein,

d. h. es darf keine zwei Linkelemente mit derselben Menge an Verknüpfungen geben. Der Vorgang der Verlinkung ist in Teil II, Kap. 15 und 16 anhand von Beispielen dargestellt.

### 2.7.5.1 Horizontale Linkmodelle

Die Beispiele 1 bis 3 zeigen zwei Linkelemente zur Verlinkung der vier Fachmodelle Leistungsverzeichnis M1, Mengenmodell M2, Bauwerksmodell M3 und Terminplan, M4, wie sie in Abb. 2.2 dargestellt sind und in den erstellten Multimodellen der beiden Mefisto Projekte (Teil II, Kap. 23) zum Einsatz kommen.

#### Beispiel 1

```
...
<linkObject>
  <link m="M1" id="BoQ1::2.10." />
  <link m="M4" id="Activity1::001.002.002.001" />
  <link m="M4" id="Activity1::001.002.002.003" />
  <link m="M3" id="1csK8Ps2D7QvH56tc1tp74" />
  <link m="M2" id="QTO1::0001F0" />
</linkObject>
<linkObject>
  <link m="M1" id="BoQ1::2.20." />
  <link m="M4" id="Activity1::001.001.002.002" />
  <link m="M3" id="1csK8Ps2D7QvH56tc1tp4H" />
  <link m="M2" id="QTO1::0001G0" />
</linkObject>
...
```

Für Fälle, bei denen die ID zwischen den alternativen Datenformaten eines Modells abweicht, z. B. cpixml und ifc2×3, kann je content Format ein Link für das gleiche Fachmodell durch eine erweiterte optionale Angabe der content ID erfolgen (vgl. Attribut „c“ in Abb. 2.13 und Beispiel 2), wie es z. B. für das BIM Fachmodell des Mefisto Flughafen-terminals (Teil I, Kap. 7) zum Einsatz kommt.

#### Beispiel 2

```
...
<linkObject>
  <link m="M1" id="BoQ1::2.10." />
  <link m="M4" id="Activity1::001.002.002.001" />
  <link m="M4" id="Activity1::001.002.002.003" />
  <link m="M3" c="C1"
    id="Projektname::b4e5c2e2-afdb-4c97-8e3b-65ceb0d40241" />
  <link m="M3" c="C2" id="1csK8Ps2D7QvH56tc1tp74" />
  <link m="M2" id="QTO1::0001F0" />
</linkObject>
...
```

Für die Fälle, bei denen sich die Links lediglich auf Teile eines Objekts beziehen, wie bspw. bei der Mengenermittlung, kann für die Mengenrelation ein Faktor ausgegeben werden. Beispiel 3 zeigt Mengenzuordnungen der Menge „QTO1:0001E0“ aus dem Mengenmodell M2 mit dem Faktor factor=„7.001“ zu einer Unterposition, UPos des Kalkulationsmodells M6 und zu einem Vorgang im Terminplan M4.

### Beispiel 3

```
...
<linkObject>
  <link m="M1" id="BoQ1::2.10." />
  <link m="M4" id="Activity1::001.001.002.001">
    <relation to_m="M2" factor="7.001"/>
  </link>
  <link m="M3" id="1csK8Ps2D7QvH56tc1tp4H" c="1" />
  <link m="M3" id="131b1ARz1E2dYUR8GyPGbQ" c="2" />
  <link m="M2" id="QTO1::0001E0" />
  <link m="M6" id="[1]2.10.[1]">
    <relation to_m="M2" factor="7.001"/>
  </link>
</linkObject>
...
```

#### 2.7.5.2 Vertikale Linkmodelle

Das nachfolgende Beispiel zeigt Linkelemente zur Verlinkung der unterschiedlichen Detaillierungen eines Terminplans, hier die Verlinkung des Fachmodells M1 eines Rahmen-terminplans mit dem Fachmodell M2 eines Ausführungsterminplans.

```
...
<linkObject>
  <link m="M1" id="Activity1::001.002" />
  <link m="M2" id="Activity1::001.002.002.003" />
</linkObject>
...
```

#### 2.7.5.3 Longitudinale Linkmodelle

Ebenso kann die zeitliche Fortschreibung, d. h. die Versionierung eines Fachmodells über ein Linkmodell abgebildet werden. Das Beispiel zeigt die Linkelemente zur Verlinkung der identischen Vorgänge zwischen der Version 1 und 2, die hier über die Fachmodelle M1 und M2 abgebildet sind.

```
...
<linkObject>
  <link m="M1" id="Activity1::001.002.002.001" />
  <link m="M2" id="Activity1::001.002.002.002" />
</linkObject>
...
```

---

## 2.8 Spezifikationen für Multimodellvorlagen

Zur Datenspezifikation eines Multimodellcontainers wurden entsprechende Multimodellvorlagen entwickelt. Diese spezifizieren den zukünftigen Dateninhalt, indem per Metadaten entsprechend dem Vokabularkatalog des Ontologiemodells [9] (Abschn. 2.4) die Inhalte und Eigenschaften der zu erstellenden Fach- und Linkmodelle qualitativ vorgegeben werden. Hierzu wird der Multimodellcontainer auf die Metadaten reduziert, d. h. Modelldaten sind nicht enthalten.

Weiterhin wurde die Syntax der Multimodellmetadaten um freie Parameter, im Sinne von Variablen, erweitert, wodurch eine wesentlich flexiblere Erstellung von Multimodellen aus den Vorlagenvorgaben ermöglicht wird. So können auf diese Weise bereits durch den Vorlagenersteller sinnvolle Filterkonfigurationen in prägnanter Ausdrucksweise vorgegeben werden. Für BIMfit steht hierzu über die GMSD Methode eine mächtige, formalisierte Filterspezifikationssprache zur Verfügung (Teil II, Kap. 14). Diese Parametrisierung wurde durch eine entsprechende programmseitige Erweiterung in RIB iTWO erfolgreich auf ihre Funktionalität und Mächtigkeit getestet.

Das nachfolgende Beispiel zeigt eine parametrisierte Multimodellvorlage für die Abgabe eines Nachtragsangebots. Als freie Parameter in den Metadaten kamen hier die Multimodellcontainerbeschreibung „MMC\_Beschreibung“ und die Nachtragsnummer „Nachtrags\_Nr“ zum Einsatz, wie sie in Band 2, Teil II, Kap. 11 eingesetzt wird.

```
...
<?xml version="1.0"?>
<container guid="" formatVersion="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://mefisto-bau.de/
xsd/container.xsd">
  <meta>
    <info>
      <i k="ContainerName" v="{MMC_Beschreibung}" />
      <i k="ContainerDescription"
        v="MMC-Datenaustausch zur Angebotsabgabe
        eines Nachtrages." />
      <i k="ContainerType" v="MMC NTAngebot" />
      <i k="ContainerFromTemplate"
        v="Angebot_Abgabe_Nachtrag.mmt" />
      <i k="TemplateName" v="Angebotsabgabe (Nachtrag)" />
    </info>
  </meta>
  <models>
    <model type="BoQ" id="FM1">
      <meta/>
      <subset subsetCode="NT[{Nachtrags_Nr}]"
        subsetDesc="Nachtrag[{Nachtrags_Nr}]">
        <subsetFilter application="MMCFilterService">
          NT[{Nachtrags_Nr}]</subsetFilter>
        </subset>
        <content format="gaebxml" id="1" formatVersion="3.1">
          <contentOptions>
            <i k="extension" v="DA81" />
            <i k="description" t="xs:boolean" v="1" />
            <i k="unitrate" t="xs:boolean" v="1" />
            <i k="unitrate_breakdown"
              t="xs:boolean" v="1" />
          </contentOptions>
        </content>
      </model>
    </models>
  </container>
...
```

## 2.9 Zusammenfassung und Ausblick

Eine prozessorientierte Arbeitsweise erfordert eine hohe und flexible Vernetzung der heterogenen und interdisziplinären Baudaten. Das Multimodellkonzept ermöglicht mit seinen expliziten Linkmodellen den Austausch solcher domänenübergreifenden Informationsräume auf Basis der originalen Fachmodelle. Die Möglichkeit zur Spezialisierung erlaubt die Definition eines auf den Anwendungsbereich zugeschnittenen Multimodells, z. B. für Bauprojektmanagement sowie die Erstellung von Multimodellvorlagen, die sich als Handlungsvorschriften für Dritte eignen, was für ein Multimodell sie erstellen sollen, und zudem können Multimodellvorlagen die Basis für die Datenqualitätskontrolle bilden.

Die Multimodellmethode ist vor allem dazu geeignet, Dritten einen ganzheitlichen Informationsausschnitt aus einer Menge von mehreren Fachmodellen, wie sie derzeit schon erzeugt und ausgetauscht werden, zu übermitteln. Das Linkmodell bildet die bisher fehlende, verknüpfende Information ab. Eine Pflege der Links, die notwendig wird, wenn sich Fachmodelle verändern, ist derzeit noch nicht möglich. Dies ist kein Manko der Multimodellmethode, sondern der Tatsache geschuldet, dass es noch keine Werkzeuge und auch nicht ausreichende Methoden gibt, die diese Pflege übernehmen. Dies liegt derzeit noch in der Verantwortung des Erstellers bzw. Senders eines Multimodells. Mit dem derzeitigen Linkmodell der Multimodellmethode ist es erstmals möglich, die Informationsinseln der Fachmodelle zu überwinden. Die Ausführenden erhalten den benötigten ganzheitlichen und verlinkten Informationsausschnitt aus mehreren Fachmodellen.

Die Anwendbarkeit der Multimodellmethode wurde unter anderem durch die Erweiterungen der Industrieprodukte RIB iTWO von RIB und GRANID von gibGreiner nachgewiesen. Dabei zeigte sich, dass die Anforderungen und Erwartungen erfüllt und das Potential für weitere Anwendungsszenarien eröffnet wurden. Es ist daher vorhersehbar, dass der multimodellbasierte Datenaustausch Einzug in Bausoftwareprodukte mittelfristig finden wird. Durch die beiden o. g. Produkte ist er bereits schon jetzt nutzbar.

Der Multimodellcontainer besteht aus einem offenen Datenmodell. Dieses wird durch die buildingSMART weiterentwickelt. Eine entsprechende Projektgruppe wurde im Oktober 2013 im German chapter der buildingSMART eingesetzt.

---

## Literatur

1. Fuchs S, Katranuschkov P, Scherer RJ (2010) A framework for multi-model collaboration and visualisation. In: Menzel K, Scherer R (Hrsg) eWork and eBusiness in architecture, engineering and construction, European conference on product and process modelling 2010. CRC Press/Balkema, Rotterdam
2. Scherer RJ, Schapke SE (2011) A distributed multi-model-based management information system for simulation and decision making on construction projects. Adv Eng Inform 25:582–599
3. Fuchs S, Kadolsky M, Scherer RJ (2011) Formal description of a generic multi-model. In: Reddy S, Tata S (Hrsg) Proceedings of the 2011 IEEE 20th international workshops on enabling technologies: infrastructure for collaborative enterprises, WETICE'11. IEEE Computer Society, Washington, DC, USA, S 205–210



4. Steinberg D, Budinsky F, Paternostro M, Merks E (2008) EMF: eclipse modeling framework, 2. Aufl. Addison-Wesley Professional, Reading
5. BGL (2001) Baugeräteliste, <http://www.bgl-online.info/>
6. KLM (2013) Keyhole markup language, <https://developers.google.com/kml/documentation/>
7. US Department of Energy (2013) Weather data for EnergyPlus energy simulation software, [http://apps1.eere.energy.gov/buildings/energyplus/weatherdata\\_about.cfm](http://apps1.eere.energy.gov/buildings/energyplus/weatherdata_about.cfm)
8. Weise M (2006) (2006) An approach for description of changes in model-based object planning, in German: Ein Ansatz zur Abbildung von Änderungen in der modellbasierten Objektplanung. PhD thesis, Berichte des Instituts für Bauinformatik, Heft 4, ISBN 978-3-86005-557-1, Dresden
9. Schapke SE, Baling M, Hienz R, Muntzinger HD, Renz J, Zeller G (2010) Mefisto-Bericht B-1.1: Anforderungen an das Gesamtsystem. Institut für Bauinformatik, Technische Universität Dresden
10. GAEB (2013) Gemeinsamer Ausschuss Elektronik im Bauwesen, <http://www.gaeb.de/>
11. ISO 16739 (2005) International Organization for Standardization: ISO/PAS 16739:2005 Industry Foundation Classes, Release 2×, Platform Specification (IFC2× Platform). Standard
12. RIB iTWO (2013) <http://www.rib-software.com/de/landingpage/rib-itwo.html>
13. ifcXML for IFC2× Edition 3 (2013) <http://www.buildingsmart-tech.org/specifications/ifcxml-releases/ifcxml2x3-release/summary>
14. Weise M, Katranuschkov P, Scherer RJ (2003) Generalised model subset definition schema. In: Proceedings of the CIB-W78 conference 2003 – information technology for construction, Auckland

Informationssysteme im Bauwesen 1

Modelle, Methoden und Prozesse

Scherer, R.J.; Schapke, S.-E. (Hrsg.)

2014, XV, 609 S. 330 Abb., 109 Abb. in Farbe.,

Hardcover

ISBN: 978-3-642-40882-3