Technische Universität München
Ingenieurfakultät Bau Geo Umwelt
Lehrstuhl für Computergestützte Modellierung und Simulation
Prof. Dr.-Ing. André Borrmann

Analyse von BIM-basierten Kollaborationswerkzeugen

Michael Dietl

Bachelor's Thesis

im Studiengang Bauingenieurwesen
zur Erlangung des akademischen Grads eines
Bachelor of Science (B. Sc.)

Autor:	Michael Dietl
Matrikelnummer:	[redacted]
1. Betreuer:	Prof. Dr.-Ing. André Borrmann
2. Betreuer:	Cornelius Preidel M. Sc.
3. Betreuer:	Dr.-Ing. Lorenz Lachauer
Ausgabedatum:	12. Juni 2015
Abgabedatum:	11. Dezember 2015

Abstract

Worldwide a steady movement towards digitalization in the building industry can be recognized as innovative governmental institutions increase using the cooporative planning method Building Information Modeling (BIM). The focus of this method is the digital image of a building, which resembles a 3D-database with geometric and semantic information. This is the reason why BIM serves as a model based communication platform for all project members. The open format Industry Foundation Classes (IFC) provides the opportunity to merge propietary formats of different software vendors – throughout all disciplines. Furthermore, it enables exchange of information avoiding communication misunderstandings regarding the building model. From the viewpoint of information technology it prevents the incorrect decoding and loss of data sets. As actors from different disciplines have their own priorities concerning the creation of components a data exchange as described above often proves difficult because of the complex data structure in BIM. In order to generate a central building information model all different models must be united in a general one. In this process complications are inevitable which automatically implicate certain changes in submodels. Aiming for an efficient communication about model changes the BIM Collaboration Format (BCF) established. Representing a plug-in, the BCF Manager developed by a company named KUBUS offers a collaborative connection between certain software products by using the BCF. In this thesis the workflow of cooperating actors who use the BCF Manager is explained by a practical scenario.

Zusammenfassung

International lässt sich der Trend zur Digitalisierung im Bauwesen verfolgen, indem innovative staatliche Einrichtungen auf die kooperative Planungsmethode Building Infomation Modeling (BIM) setzen. Dadurch, dass im Zentrum ein digitales Abbild eines Gebäudes steht, das einer dreidimensionalen Datenbank mit geometrischen und semantischen Informationen gleicht, dient es als modellbasierte Kommunikationsplattform für alle Projektbeteiligten. Das offene Datenformat Industry Foundation Classes (IFC) führt proprietäre Formate vieler Softwareanbieter – über alle Disziplinen hinweg – auf einen gemeinsamen Nenner zusammen und ebnet dahingehend den Weg, dass jeder – bezogen auf das Gebäudemodell – die gleiche Sprache spricht. Dies verhilft auf informationstechnischer Sicht dazu, dass ausgetauschte Datensätze weder fehlerhaft übersetzt werden noch verloren gehen. Ein solcher Datenaustausch gestaltet sich aber aufgrund einer derart komplexen Datenstruktur, wie sie bei einem BIM-Modell vorliegt, äußerst schwierig, zumal Fachplaner unterschiedlicher Disziplinen Prioritäten hinsichtlich der Modellierung von Bauteilen jeweils anders setzen. Im Sinne eines zentralen Building Information Models, das es zu generieren gilt, müssen all diese Fachmodelle in ein umfangreiches Gesamtmodell überführt werden. Dabei scheinen auftretende Komplikationen unumgänglich zu sein, was Änderungen an den jeweiligen Teilmodellen impliziert. Um solche Modellabänderungen zu kommunizieren, hat sich das BIM Collaboration Format (BCF) etabliert. Darauf aufbauend schafft es der BCF Manager, entwickelt von der Firma KUBUS, in intelligenter Weise über eine Plug-In-Schnittstelle eine kollaborative Verbindung zwischen ausgewählten Softwareprodukten herzustellen. Diese Arbeit beschreibt den Workflow bei einer Zusammenarbeit von Planern über den BCF Manager anhand eines praxisnahen Szenarios.

Inhaltsverzeichnis

Abstract 2

Zusammenfassung 2

1 Einführung und Motivation 5

1.1 Einleitung 5

1.2 Ziel der Arbeit 6

1.3 Aufbau der Arbeit 6

2 BIM − Building Information Modeling 7

2.1 Begriffsdefinition BIM 7

2.2 Open BIM 8

2.3 IFC Workflow – Zusammenabeit im BIM-Prozess 9

3 Daten- & Informationsaustausch – Stand in Deutschland 10

4 Hintergründe zur Software-Interoperabilität 12

4.1 Industry Foundation Classes – ein standardisiertes neutrales Datenformat 13

4.2 Information Delivery Manual – Standardisieren von Datenaustauschszenarie 18

5 Beschreibung des BIM Collaboration Formats 22

5.1 Optimierung des IFC-Workflows durch KUBUS 24

BCF Manager 25

BIMcollab 27

5.2 Der BCF Prozess 29

1. Untersuchen des Modells auf Unstimmigkeiten und Ableiten von BCF-Dateien 29

2. Versenden der BCFZIP-Datei an Projekbeteiligte und Betrachten der Mängel in der präerieten Software über den BCF Manager 30

3. Beheben der Mängel und Informieren über die Änderungen 30

6 Beispielszenario 31

6.1 Modellierung in Allplan 32

6.2 IFC-Import in ausgewählte Programme 33

Import in Graphisoft ArchiCAD 19 34

Import in Autodesk Revit 2016 34

Import in den Solibri Model Checker 36

6.3 Interaktion der verschiedenen Programme – BCF Workflow 36

1. Projekteinstellungen in BIMcollab 36

2. Geometrische Kollisionserkennung im Solibri Model Checker und Anweisungen über BIMcollab 36
3. Bearbeitung des Issues in Revit 37
4. Auswerten der Änderungen in ArchiCAD 39
5. Abschließende Kontrolle im Solibri Model Checker 41
7 Fazit 43
7.1 Ausblick 44
Anhang 46
Abkürzungsverzeichnis 47
Literaturverzeichnis 48
Abbildungs- und Tabellenverzeichnis 50

1 Einführung und Motivation

Building Information Modeling beschreibt im Zuge der fortschreitenden Digitalisierung im Bauwesen einen Prozess, wonach sämtliche Projektbeteiligten kooridiniert an einem digitalen Abbild eines Gebäudes zusammenarbeiten. Anders als in Deutschland, wo die Einführung von Building Information Modeling (BIM) noch am Anfang steht, stellen Finnland, Singapur, die USA, Großbritannien und Australien die großen Vorreiter diesbezüglich dar. Dabei beweisen sich die jeweiligen Regierungen als dahinterstehende Antriebskräfte, indem nationale BIM-Richtlinien, die quasi als Standardisierungswerkzeug dienen, ausgearbeitet worden sind. In Anlehnung an diese normierten Vorgaben wird eine BIM-basierte Projektabwicklung in den genannten Staaten nämlich zum Teil schon seit Jahren bei öffentlichen Bauvorhaben staatlich vorgeschrieben. (Borrmann, Building Information Modeling, 2015)

So ist BIM zwar längst kein bloßes Konzept mehr, aber die konservative Einstellung der Bauindustrie (v.a. in Deutschland) wird sich ohne konkrete Vorgaben und Anweisungen zur Umsetzung von BIM in Form solch einer Richtlinie nicht ändern (Egger, Hausknecht, Liebich, & Przybylo, 2013). Zudem Bedarf es einem grunsätzlichen Umdenken in der Arbeitsweise über alle Phasen einer Projektrealisierung hinweg zu mehr Zusammenarbeit und Kommunikation.

Wie solch eine Kollaboration im Zuge der BIM-Methode unter den verschiedensten Projektbeteiligten in der arbeitsteiligen Bauindustrie – vom Planen über das Bauen hin zum Betreiben des betrachteten Gebäudes – im Detail abläuft, stellt die große Motivation hinter dieser Arbeit.

1.1 Einleitung

Der Aufruf zu mehr Kommunikation und zu einem partnerschaftlicheren Zusammenwirken aller Projektbeteiligten in der Bauindustrie klingt auch in dem am 29.06.2015 erschienen Endbericht zur „Reformkommission Bau von Großprojekten“ durch, welche 2013 vom damaligen Verkehrsminister Dr. Peter Ramsauer anlässlich der Bauverzögerungen am Berliner Flughafen und an der Elbphilharmonie in Hamburg initiiert wurde, um Zeit- und Kostenrahmen bei Infrastrukturgroßprojekten künftig besser einzuhalten. (BMVI, 2015) Dahingehend empfiehlt die Kommission unter anderem, auf Digitalisierung zu setzen, und betont, dass sich die Realisierung von Großprojekten über Building Information Modeling, das sehr stark auf kollaboratives Arbeiten ausgerichtet ist und das auf ein offenes und konsistentes Daten- und Informationsmanagment basiert, wesentlich effizienter gestalten lässt. (Dobrindt, 2015)

Im Hinblick auf die BIM-Methode fokusiert sich die projektbezogene Kommunikation auf ein zentrales Modell. Um demzufolge eine modellbasierte Kommunikation unter allen Projektbeteiligten im BIM-Prozess zu gewährleisten, haben sich über die Jahre hinweg zwei Datenformate etabliert: zum Einen die *Industry Foundation Classes (IFC)*, zum Anderen das *BIM Collaboration Format* (*BCF)*. Vereinfachend lässt sich sagen, dass über das IFC-Format digitale Gebäudemodelle übergeben werden können, anhand derer modellierten Bauteilen über das BCF-Format Anmerkungen angehängt werden können.

Zudem hat die Firma KUBUS eine Plug-In-Schnittstelle namens *BCF Manager* jeweils für Graphisoft ArchiCAD und Autodesk Revit – zwei renommierte BIM-Modellierungssoftwareprodukte – entwickelt, um die beiden genannten Formate zusammenzuführen und somit die Kollaboration schon in der frühen Gebäudemodellierungsphase zu optimieren.

1.2 Ziel der Arbeit

Ziel ist es, den von KUBUS entwickelten BCF Manager anhand von einem praxisnahen Szenario hinsichtlich der Eignung für Anwendungen in der Baupraxis zu untersuchen und dessen Vor- und Nachteile im Vergleich zu alternativen Methoden auszuarbeiten. Daneben soll die Funktionsweise des BCF Managers bezogen auf das Zusammenspiel zwischen IFC- und BCF-Formaten analysiert und auftretende Komplikationen während des IFC bzw. BCF Workflows dokumentiert werden.

1.3 Aufbau der Arbeit

Grundsätzlich wird die behandelte Thematik vom Groben ins Detail – vom umfassenden Building Information Modeling hin zur genauen BCF-Datenstruktur – in der Arbeit beschrieben und anschließend in einem Beispielszenario dargestellt.

So wird in Kapitel 2 zuerst die weitreichende Methode des Building Information Modelings erklärt und aufgezeigt, wie das grundsätzliche Vorgehen beim Datenaustausch im BIM-Prozess abläuft.

Im Anschluss gibt Kapitel 3 einen Überblick, wie der Informationsaustausch derzeit in Detschland vonstattengeht.

Des Weiteren gewähren Kapitel 4 und 5 einen tieferen Einblick, wie die Datenformate im Detail strukturiert sind, worauf sich der BCF Manager stützt, dessen prinzipielles Arbeitsweise einbettet im „BCF Prozess“ abschließend beschrieben wird.

Zu guter Letzt veranschaulicht Kapitel 6 alle zuvor definierten Begriffe bzw. Prozesse in einem Beispielszenario und verdeutlicht die kommunikative Optimierung des IFC Workflows durch den BCF Manager.

2 BIM − Building Information Modeling

Das Schlagwort „Building Information Modeling“ (BIM) ist in der Baubranche schon seit einigen Jahren in aller Munde und wird als die zukunftsträchtigste Methode in der digitalen Entwicklung des Bauwesens angesehen. Trotzdem ist die dahinterstehende Philosophie des BIM in der Öffentlichkeit geprägt von sehr vielen Interpretationen, Vorbehalten und Missverständnissen. So umfasst der BIM-Gedanke zum Beispiel weit mehr als das bloße Modellieren einer 3D-Abbildung des betrachteten Gebäudes mittels CAD-Software und das Anheften von Informationen oder Attributen an die entsprechenden Objekte und Bauteile.

2.1 Begriffsdefinition BIM

Ein Building Information **Model** repräsentiert – wie eingangs erwähnt – das digitale Abbild eines Gebäudes sehr umfangreich anhand einer detaillierten Informationsdatenbank. Vordefinierte bauspezifische Objekte (z.B. Wände, Stützen, Türen usw.) stellen dabei die Informationsträger des Modells dar. Diese Bauteile können neben den geometrischen Abmessungen auch weitere nicht-geometrische Zusatzinformationen aufweisen. Diese werden in der Fachsprache als Semantik bezeichnet und können beispielsweise Materialeigenschaften, bauphysikalische Eigenschaften oder Kosten. Das Generieren und darüber hinaus das Modifizieren und Verwalten des Modells über den gesamten Lebenszyklus eines Bauwerks – von der ersten Vorplanung bis zum Rückbau – beschreibt den Prozess des Building Information **Modeling**. In der Nutzung des Modells und dessen Informationen über sämtliche Phasen des Projektrealisierungsprozesses hinweg spiegelt sich der Kernaspekt des BIM-Gedanken wider: Einmal eingegeben Informationen sollen fortan weiterverwendet werden, indem, sie stets aus dem Modell abgeleitet werden können. Daraus ergibt sich ein effizienter Datenaustausch, der fehlerbehaftetes und aufwendiges Wiedereingeben von Informationen überflüssig macht. Simulationen und Analysen können so direkt am BIM-Modell durchgeführt werden ohne erst davor ein vergleichbares Referenzmodell erstellen zu müssen. (Borrmann, Building Information Modeling, 2015)

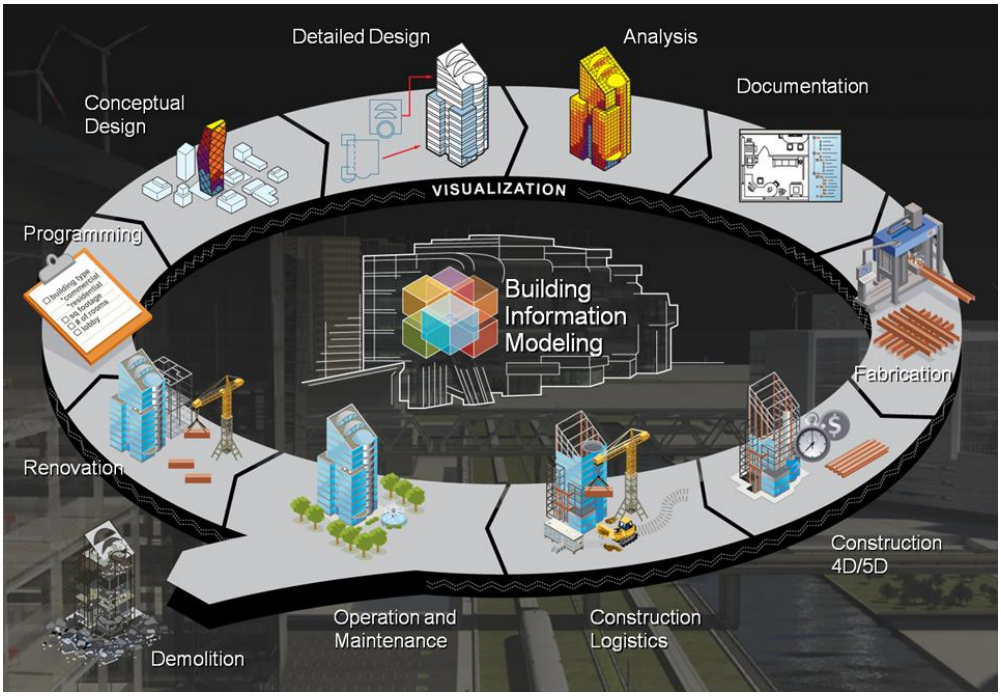


Abbildung 1 Der gesamte Lebenszyklus eines Building Information Model[1]

2.2 Open BIM

Die umfassende Arbeitsteilung bzw. Spezialisierung im Bauwesen, welche in unterschiedliche Fachdisziplinen aufgefächert ist, stellt eine große Einschränkung hinsichtlich des gewünschten Datenaustauschs dar. Die Aufspaltung in die verschiedenen Fachbereiche, mit deren zum Großteil eigen zugeschnittenen Programmen, impliziert die Heterogenität vieler Softwareprodukte in der Bauindustrie. Dabei setzten die meisten Softwarehersteller auf proprietäre Datenformate, wodurch die Interoperabilität stark gehemmt wird. (Borrmann, Building Information Modeling, 2015)

Eine Lösung bildet Open BIM, eine Methode auf Initiative der Non-Profit-Organisation *buildingSMART* für ein standardisiertes neutrales Datenformat, namens *Industry Foundation Classes* (IFC, siehe Abschnitt 4.1), zur umfangreich Beschreibung von Bauwerksmodellen. Dieses offene, herstellerunabhängige Datenformat hat sich seit der Einführung seiner ersten Version 1997 über die Jahre etabliert und wird – in seiner aktuellen Version *IFC 2x3* – von den führenden BIM-Tools in der Baubranche unterstützt. (Borrmann, Building Information Modeling, 2015)

Neben dem modellbasierten Datenaustausch über das IFC-Format unterstützt Open BIM ein weiteres offenes Datenformat namens *BIM Collaboration Format* (BCF, siehe Abschnitt 5). In Analogie zur konventionellen Arbeitsweise mit 2D-Plänen gleicht dieses Format einer

[1]http://buildipedia.com/images/masterformat/Channels/In_Studio/The_Daily_Life_of_Building_Information_ModelingBIM/BIM_Illustration.jpg

Revisionswolke, die Änderungen auf zu überprüfenden Zeichnungen markiert. (Borrmann, Building Information Modeling, 2015)

2.3 IFC Workflow – Zusammenabeit im BIM-Prozess

Modell, die mit jeweils anderen Softwarelösungen erzeugt worden sind, müssen unter Planern der verschiedenen Disziplinen austauschbar sein, um die Methode des Open BIM im Büroalltag integrieren zu können. Der modellbasierte Datenaustausch nach dem Open BIM-Konzept entspricht in diesem Zusammenhang einem IFC Workflow.

Grundlegend wird dabei in sog. Referenzmodellen gearbeitet: Im frühen Planungsstadium entwirft der Architekt das Architekturmodell, welches beispielsweise der Tragwerksplaner über das IFC-Datenformat wiederum einlesen kann. Im Anschluss referenziert der Tragwerksplaner das Architekturmodell in seiner eigenen Software. Das heißt, er filtert nur diejenigen Informationen bzw. Bauteile aus dem Modell des Architekten, die für ihn letztendlich von Bedeutung sind, um ein statisches Modell abbilden zu können (z.B. tragende Wände, Fundamente, Decken usw.). Das Ergebnis stellt das Strukturmodell dar, das die Grundlage für statische Berechnungen und weiterführend für Analysen bildet. Abschließend exportiert der Tragwerksplaner sein Struktur- bzw. Analysemodell per IFC-Datei. (BIMwelt, 2015)

Ähnlich verhält es sich bei der Technischen Gebäude Ausrüstung und weiteren Disziplinen: Am Ende verfügt jeder einzelne Fachplaner über ein separates Teilmodell und trägt dafür die Verantwortung.

Führt schließlich ein Projektmanager die einzelnen Modelle in einer Software für Modellprüfung zusammen, können Kollisionen detektiert werden, um dann anschließend Änderungen an den betreffenden Teilmodellen vorzunehmen.

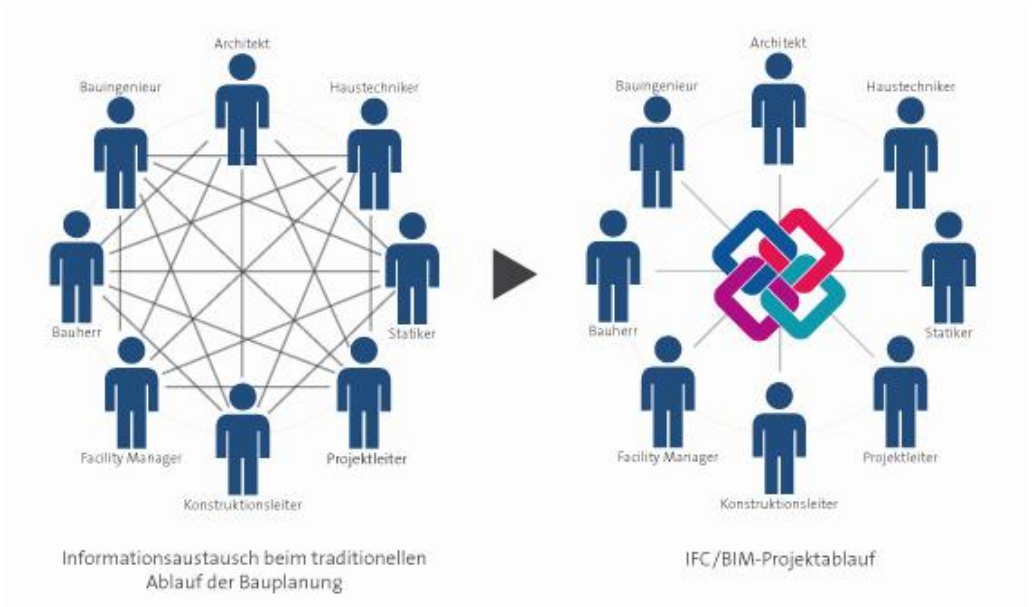


Abbildung 2 IFC Workflow nach dem Open BIM-Schema[2]

[2] https://www.wincasa.ch/getattachment/7c7c0953-67d3-48a3-82a1-aed521cb303f/BIM_Grafik.jpg.aspx

3 Daten- & Informationsaustausch – Stand in Deutschland

Wie in den einleitenden Kapitel bereits erwähnt, ist die Einführung von BIM in den USA, Großbritannien, Finnland, Australien und in Singapur anders als in Deutschland schon weit fortgeschritten. Vor allem was die praktische Integration in den verschiedenen Berufsbereichen im Bauwesen betrifft (z.B. bei der Ausbildung zum Bauzeichner), liegt Deutschland eindeutig im Rückstand. Aber auch der Technisierungsgrad in der Bauausführung ist nicht so ausgeprägt wie im Ausland. Grundsätzlich fehlt es größtenteils noch am korrekten Verständnis und an der Akzeptanz für BIM. (Egger, Hausknecht, Liebich, & Przybylo, 2013)

Die starre Einteilung in Leistungsphasen nach der Honorarordnung für Architekten und Ingenieure (HOAI) stellt nämlich aus Sicht der Skeptiker das vermeintlich größte Hemmnis beim Einsatz von BIM in der Planung dar. So „ […] *legt der Planer bei der Anwendung der BIM-Methodik bereits in der Vorplanung (Leistungsphase 2 nach §34 HOAI) Bauteilspezifikationen fest, die nach dem klassischem Planungsprozess der HOAI eher der Entwurfsplanung bzw. Genehmigungsplanung zuzurechnen sind.* […] *Durch den Einsatz von BIM kommt es folglich zu einer Verschiebung des Aufwands in frühere Planungsphasen* […]. *“* (Liebchen, 2014)

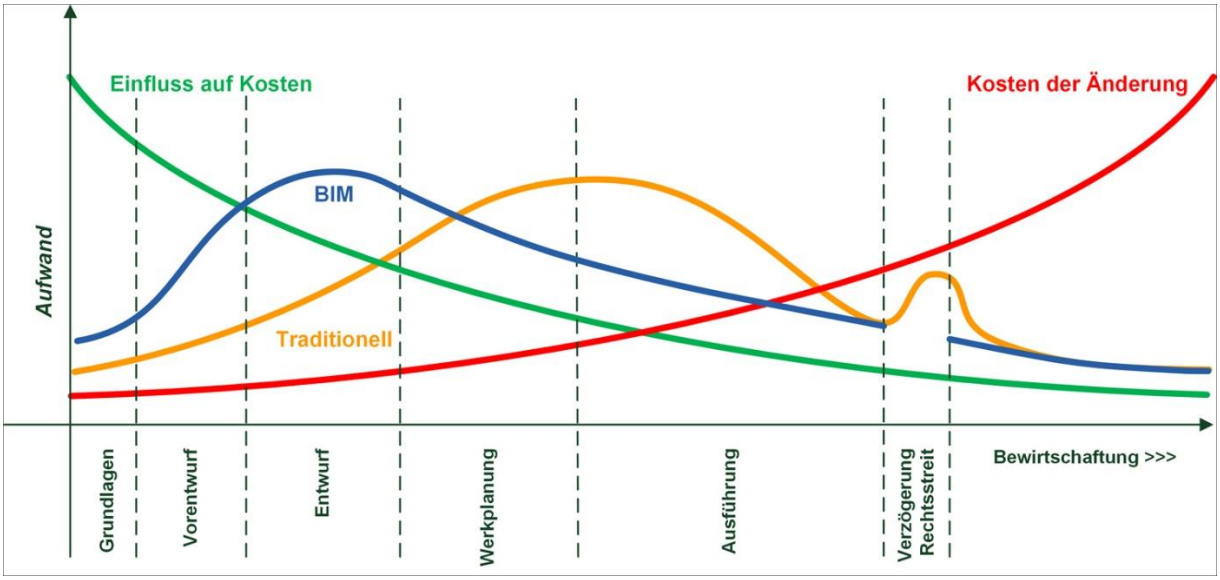


Abbildung 3 Vorverlagerung von Planungs- und Entscheidungsprozessen gegenüber der konventionellen Arbeitsweise[3]

Dahingehend müssen Anpassungen bei der Vergütung von Planungsleistungen getroffen werden und zugehörige Ausschreibungs- und Vertragsvorlagen bereitgestellt werden. Um künftige BIM-Projekte reibungslos ablaufen zu lassen, sollten Arbeitsabläufe und Verantwortlichkeiten – als rechtlicher Hintergrund – konkret festgelegt werden. Aus diesem Grund werden schon seit längerem Zeitraum BIM-Richtlinien und Vorgaben gefordert, was u.

[3] http://www.architekturbuch.de/out/pictures/master/product/2/440337-bim-kompendium-mc-leamy-kurve.jpg

a. auch aus dem BIM-Leitfaden für Deutschland (Egger, Hausknecht, Liebich, & Przybylo, 2013) hervorgegangen ist.

Bis zum jetzigen Zeitpunkt (Stand 2015) ist noch keine allgemeingültige BIM-Richtlinie eingeführt worden. Nach Borrmanns Einschätzung würde die Einführung allerdings nicht mehr lange auf sich warten lassen:

"Die Entwicklungen der jüngeren Zeit, wie die Gründung der planen-bauen 4.0 GmbH als deutsche BIM Task Group, lassen erwarten, dass diese Aufgaben in naher Zukunft angegangen werden." (Borrmann, Building Information Modeling, 2015)

Dennoch arbeiten in Deutschland vereinzelt innovative Planungsbüros und Baufirmen bereits verlässlich mit BIM und orientieren sich dabei meist an Richtlinien von privaten – oft ausländischen – Auftraggebern für eine effiziente und nachhaltige Arbeitsweise. Daraus lässt sich ableiten, dass die Etablierung des BIM-Gedankens nicht seitens der Planer vorangetrieben wird, sondern dadurch, dass Bauherrn künftig die Nutzung von BIM zur Verbesserung der Planungssicherheit explizit fordern. (Egger, Hausknecht, Liebich, & Przybylo, 2013)

Überwiegend geschieht der Informationsaustausch aber auf herkömmlicher Weise ohne den Einzug von BIM. Informationen werden also nicht vom Gebäudedatenmodell abgeleitet, sondern Pläne wie Grundrisse, Schnitte und Detailzeichnungen werden in zweidimensionaler computergestützter Ausarbeitung – gemäß der Jahrhunderte alten Arbeitsweise mit dem Zeichenbrett – per E-Mail zwischen den Planungspartner ausgetauscht. Dies bietet die Möglichkeit, digitale Informationen schnell an die vielfältigen Schnittstellen zwischen den verschiedenen beteiligten Unternehmen gleichzeitig zu versenden. Andererseits birgt diese Möglichkeit auch Risiken, wenn sämtliche Projektteilnehmer geforderte Informationen nicht diszipliniert aufarbeiten. Die Informationsweitergabe erfolgt dann unstrukturiert und Daten können willkürlich weiter verteilt werden. Beispielsweise führt es zu gravierdenden Fehlern, die nur unter großem Aufwand behoben werden können, falls in einer unternehmensinternen Arbeitsgruppe mit redundaten Datensätzen und unterschiedlichen Bearbeitungsständen – gespeichert auf verschiedenen Rechnern – ohne einem vorherigen Datenabgleich an einem veralteten Stand weitergearbeitet wird. (Günthner & Borrmann, 2011)

Eine Lösung für eine zentrale Steuerung des Informationsflusses zur sinnvollen Nutzung der großen Mengen stellt daher ein geeignetes Datenmanagement dar, wie es BIM vorsieht.

4 Hintergründe zur Software-Interoperabilität

Nach der Begriffsdefiniton (Abschnitt 2.1) fordert BIM die Weiterverwendung einmal eingepflegter digitaler Informationen über den kompletten Lebenszyklus (vgl. Abbildung 1) eines Gebäudes hinweg. Um also keine Daten zu verlieren oder neu generieren zu müssen, finden schon in der frühen Planungsphase einer Projektrealisierung verschiedenste Datenaustauschszenarien statt. Dies gilt aber auch in einem fortgeschrittenerem Stadium bei der Übergabe von Datensätz an ausführende Baufirmen und schlussendlich auch an Facility-Management-Unternehmen – ehe der Rückbau einsetzt.

Dadurch dass die von den einzelnen Disziplinen eingesetzte Software zum Teil derart auf das jeweilige Fachgebiet spezialisiert ist – sei es im Entwurf, bei Berechnungen oder Analysen usw. –, hat sich eine große Bandbreite an Softwareprodukten mit proprietären Datenformaten herauskristallisiert. Daher gestaltet sich der Austausch von Informationen herstellerbedingt äußerst schwierig.

Hier setzt der Begriff der Interoperabilität an, der einen verlustfreien Datenfluss zwischen zwei unterschiedlichen Softwareherstellern fordert. Doch in der Bauindustrie wird die Heterogenität unter den Softwareprodukten zusätzlich durch grundlegende Randbedingungen aus der Berufspraxis erschwert. Bormann schildert die Erschwernisse wie folgt:

- *„Bauplanung und –ausführung werden in der Regel von unterschiedlichen Unternehmen durchgeführt,*
- *die Planung selbst durchläuft unterschiedliche Phasen, die häufig von verschiedenen Planungsbüros übernommen werden,*
- *es gibt eine Vielzahl von einzubindenden Fachplanern, die jeweils wiederum in verschiedenen Unternehmen arbeiten,*
- *die Bauindustrie ist stark fragmentiert und setzt sich aus einer Vielzahl kleiner und mittlerer Unternehmen zusammen (Statistiken im Euroraum belgen, dass 93 % der Firmen weniger als 10 Beschäftigte haben),*
- *die Zusammenarbeit der verschiedenen Unternehmen geschieht nicht in einem festen Rahmen über einen längeren Zeitraum, stattdessen werden Ad-hoc Partnerschaften auf Projektbasis gegründet.“* (Borrmann, Building Information Modeling, 2015)

Daher ist die Notwendigkeit eines offenen, herstellerunabhängigen Datenformats für das fachübergreifende Zusammenarbeiten hinsichtlich des Open BIM-Gedankens essentiell.

Im Folgenden wird das IFC-Format als ein solches, welches Interoperabilität ermöglicht und dabei geometrische und semantische Informationen nach dem BIM-Konzept enthält, genauer beschrieben.

4.1 Industry Foundation Classes – ein standardisiertes neutrales Datenformat

Mit den Industry Foundation Classes steht – wie schon in Abschnitt 2.2 erwähnt – ein konkretes und offenes Datenformat mit einer objektorientierten Datenstruktur, die sowohl die Geometrie als auch die Semantik von Bauteilen umfasst, für den herstellerunabhängigen Modellaustausch zur Umsetzung von Open BIM über den gesamten Lebenszyklus eines Gebäudes bereit.

Entwickelt wurde dieses Format von der Non-Profit-Organisation buildingSmart, die bis zum Jahre 2005 noch den Namen *International Alliance for Interoperability (IAI)* trug und seit 1995 besteht. Hierbei handelt es sich um eine Initiative, gestartet durch führende Sofwarehersteller und innovative Ingenieurbüros aber auch durch Baufirmen, mit dem Ziel Standards und Vorgaben für ein plattformunabhängiges Datenmodell auszuarbeiten. Diese Initiative entstand in den 1980er Jahren u. a. auch unter den Umständen, die Standardisierungsmaßnahem jenes Datenmodells aus dem ISO-Standard STEP – *„Standard for the Exchange of Product model data“*, also Standard für den Austausch von Produktmodelldaten – herauszulösen. Ziel war es, die langwierigen internationalen Normierungsprozesse der Internationalen Organisation für Normierung (ISO) vorerst zu umgehen und um damit die Entwicklung schneller voranzutreiben (Borrmann, Building Information Modeling, 2015). Das IFC-Format basiert daher auf dem STEP-Standard (ISO 10303), der den Rahmen für die Datenstruktur stellt und dabei neben den geometrischen Informationen einzelner Bauteile sowohl deren angehängten Eigenschaften als auch Beziehungen zu anderen Objekten beschreibt. Die dahinterstehende objektorientierte Datenmodellierunssprache von IFC bildet EXPRESS, die genauso aus dem zugrunde liegenden STEP-Standard übernommen wurde wie das Datenaustauschformat *STEP Physical File (P21)*. (Borrmann, Building Information Modeling, 2015)

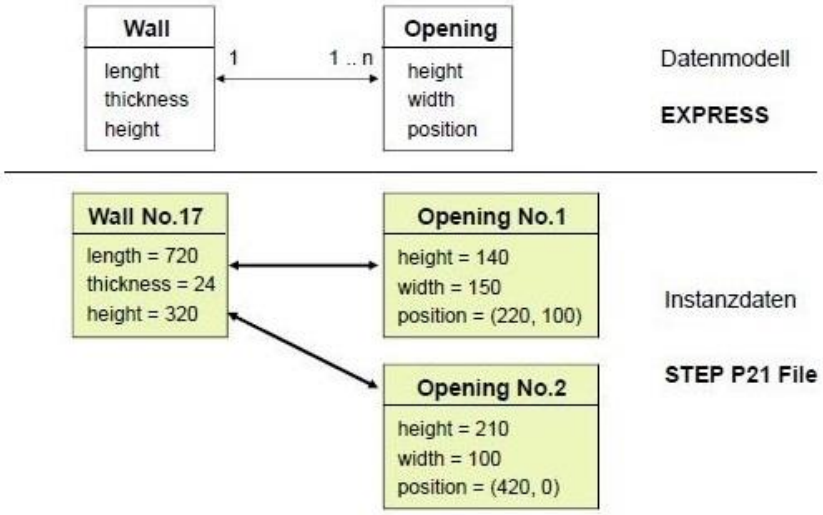


Abbildung 4 Darstellung des Datenmodells und der Instanzdaten nach dem IFC[4]

Anhand der Abbildung 4 liegt der Unterschied zwischen beiden Begriffen erklärt: EXPRESS definiert ein übergeordnetes Schema (bzw. Datenmodell), welches ausgeschmückt mit

[4] Borrmann, A. (3. November 2015). *Objektorientierte Modellierung Datenaustausch mittels IFC*. TU München: Vorlesung, Bau- und Umweltinformatik Ergänzungsmodul.

konkreten Instanzdaten, den „STEP P21 Files“, seine Äusprägung in einem visuellen Building Information **Model** findet.

Wie jede objektorientierte Programmiersprache unterscheidet EXPRESS zwischen vielerlei Klassen, die Attribute und Beziehungen zu anderen Klassen aufweisen können. Nur spricht man bei EXPRESS nicht von „Klassen“ sondern von „Entitytypen“. Ein Entitytyp ist abstrahiert von einem Objekt der realen Welt, beispielsweise einer Wand.

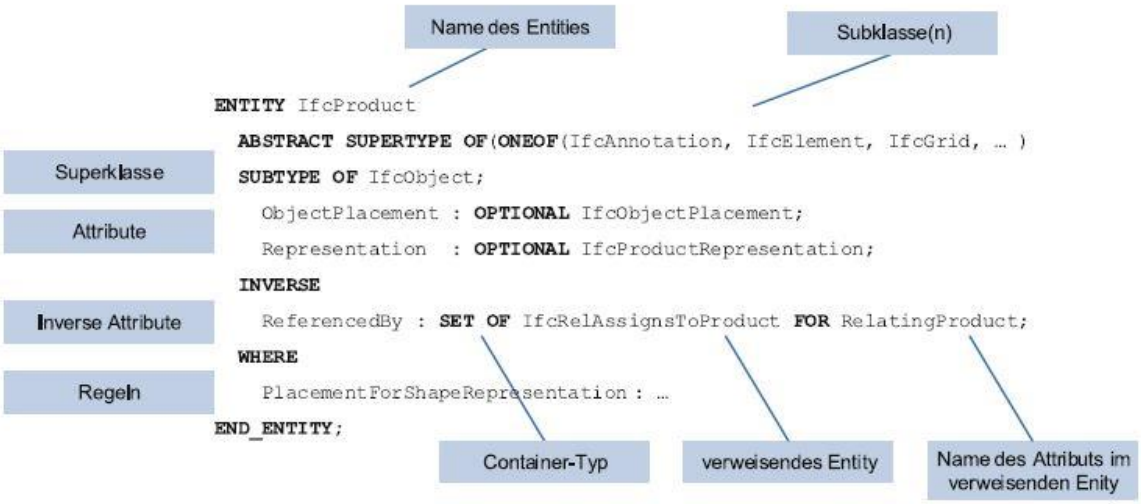


Abbildung 5 Beispielhafte Definition eines Entitytyps[5]

[5] Borrmann, A. (2015). *Building Information Modeling.* (A. Borrmann, M. König, C. Koch, & J. Beetz, Hrsg.) Wiesbaden: Springer.

Das IFC-Datenmodell ist sehr umfassend und deshalb in verschiedene Schichten aufgebaut, um Erweiterungen daran zuzulassen und Wartungsarbeiten zu vereinfachen.

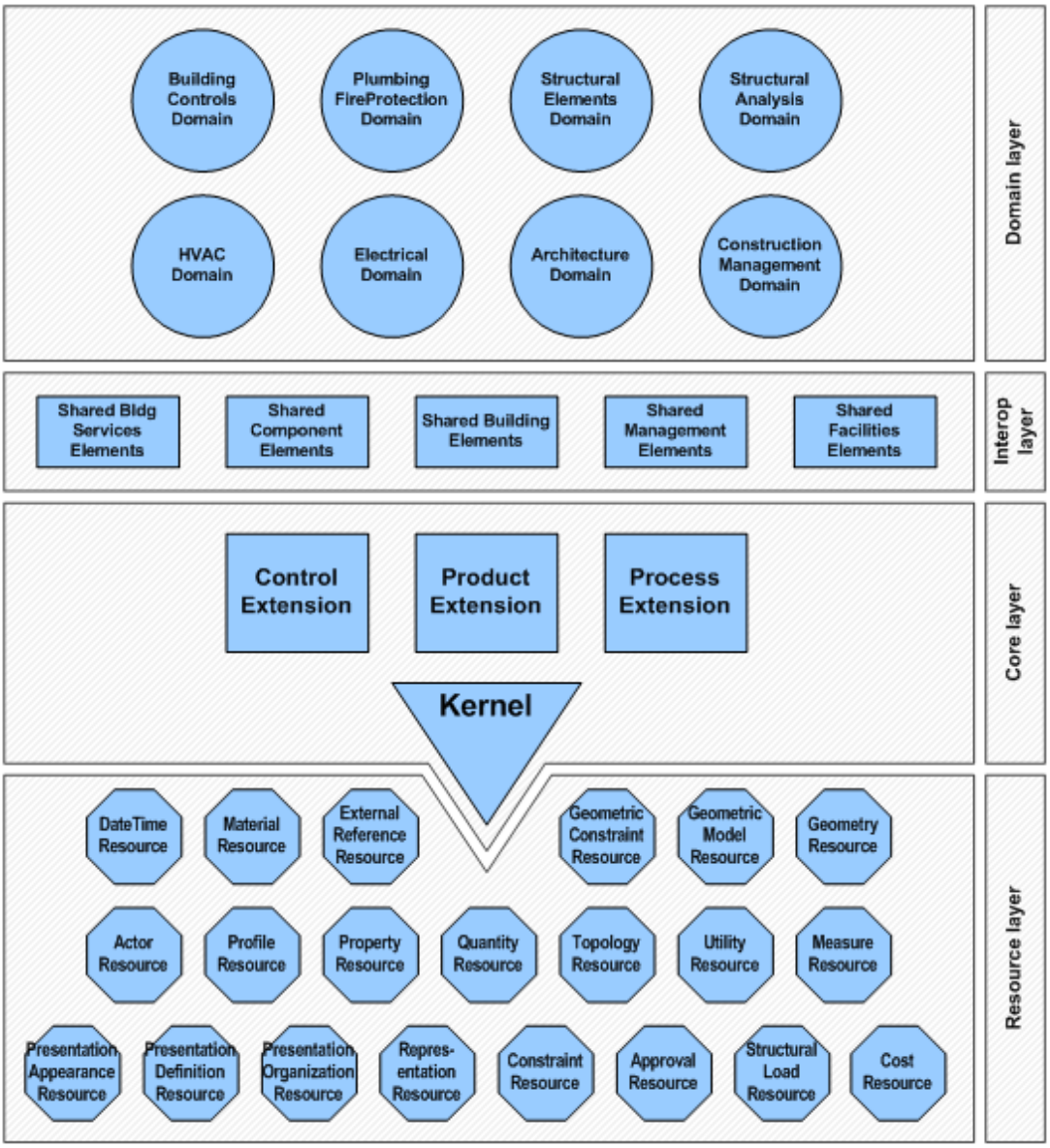


Abbildung 6 Schichtenaufbau des IFC-Datenmodells[6]

Im Mittelpunkt des IFC-Datenmodells steht das Kernel-Schema. Diesem lässt sich eine besondere Bedeutung zuschreiben, denn es beinhaltet abstrakte Basis-Klassen wie *IfcRoot*, *Ifc Object*, *IfcActor*, *IfcProcess*, *IfcProdukt*, *IfcProject* und *IfcRelationship* (Borrmann, Building Information Modeling, 2015).

Davon ausgehend kommt der *IfcRoot* eine besondere Bedeutung zu, denn es stellt den Ursprung aller Entitys im Datenmodell und bildet daher die Überleitung zur Vererbungshierarchie. Da die Vererbungshierarchie Verknüpfungen unter den Klassen beschreibt, indem festgelegt wird welche Attribute einer Klasse einer anderen weitervererbt werden, repräsentiert sie ein

[6] http://www.buildingsmart-tech.org/ifc/IFC4/final/html/img/IFC4_layered_architecture.png

grundlegendes Konzept der Objektorientierung. (Borrmann, Building Information Modeling, 2015)

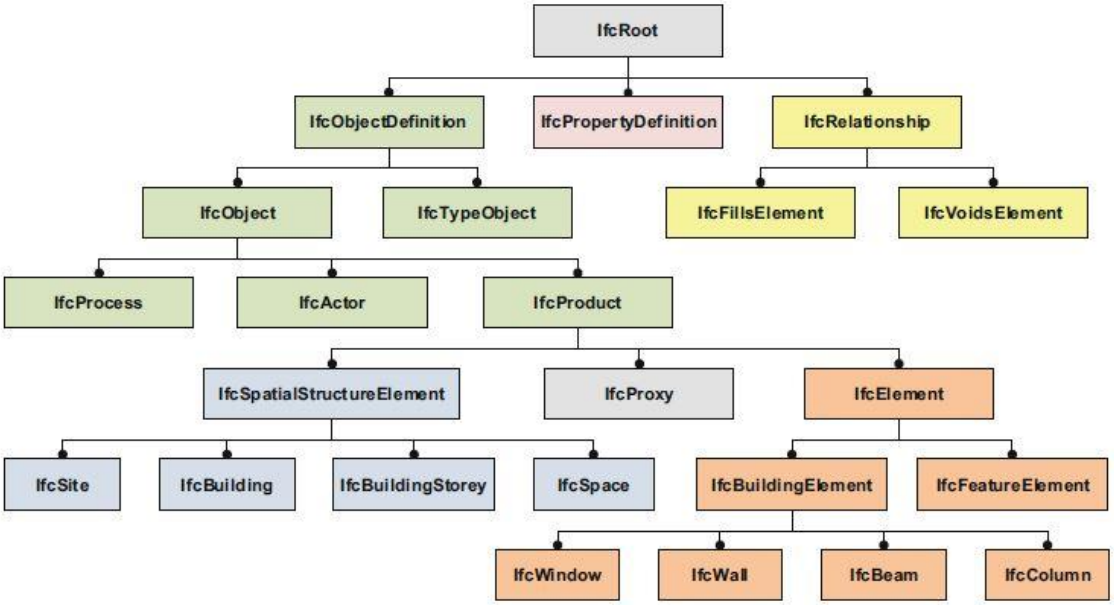


Abbildung 7 IfcRoot als Ursprung der Entitys und Ausschnitt der bedeutendsten Entitytypen (von insgesamt 653 Entitys) in der Vererbungshierarchie[7]

Ein wesentliches Attribut, das in dem Entitytyp IfcRoot verankert ist, stellt – wie sich aus Kapitel 5 ableiten lässt – die *Globally Unique Identifier (GUID)* dar. Dabei handelt es sich um eine im Modell einmalig vergebene Identifikationsnummer zur eindeutigen Beschreibung eines Objekts. Darüberhinaus trägt diese ID Informationen - hinsichtlich des Modellierungsprozesses - darüber, wer (vgl. Modellierer) welches Objekt (vgl. Bauteil) in seiner zeitlichen Abfolge erstellt oder modifiziert hat. (Borrmann, Building Information Modeling, 2015)

Diese spezielle ID lässt sich auch in sog. Model Viewern oder auch in Model Checkern verfolgen. Beispielhaft sei hier der Solibri Model Checker (SMC) der Firma Solibri aufgeführt, der über das IFC-Format importierte Gebäudemodelle in ihrer geometrischer Darstellung zuverlässig abbildet und Informationen über vorhandene Bauteile übersichtlich auflistet.

Um die Mächtigkeit der GUID zu demonstrieren, sei exemplarisch die Verbindung zwischen der Darstellung des Beispielmodells aus Abschnitt 6 im SMC und der Darstellung dieses Modells nach ISO 10303-21 (ISO 10303-21:2002-01, 2002) als konkrete Ausprägung mit IFC-Instanzdaten (STEP P21 Files) aufgezeigt:

[7] Borrmann, A. (2015). *Building Information Modeling.* (A. Borrmann, M. König, C. Koch, & J. Beetz, Hrsg.) Wiesbaden: Springer.

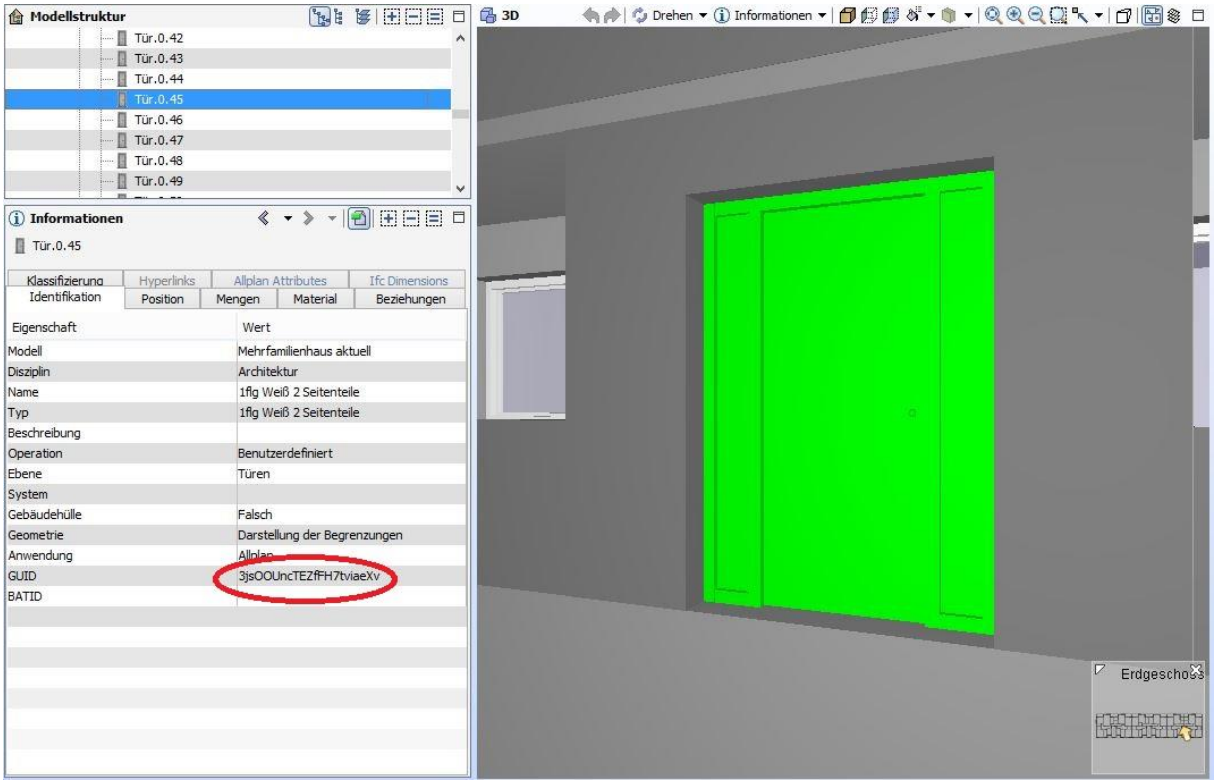


Abbildung 8 Ansicht einer Tür im Solibri Model Checker

Zuerst wurde im Model Viewer eine beliebige Tür des Gebäudemodells ausgewählt, um dann die GUID in den Informationen zu dieser Tür auszulesen. Im Anschluss wurde diese GUID im Editor der zugehörigen IFC-Datei gesucht.

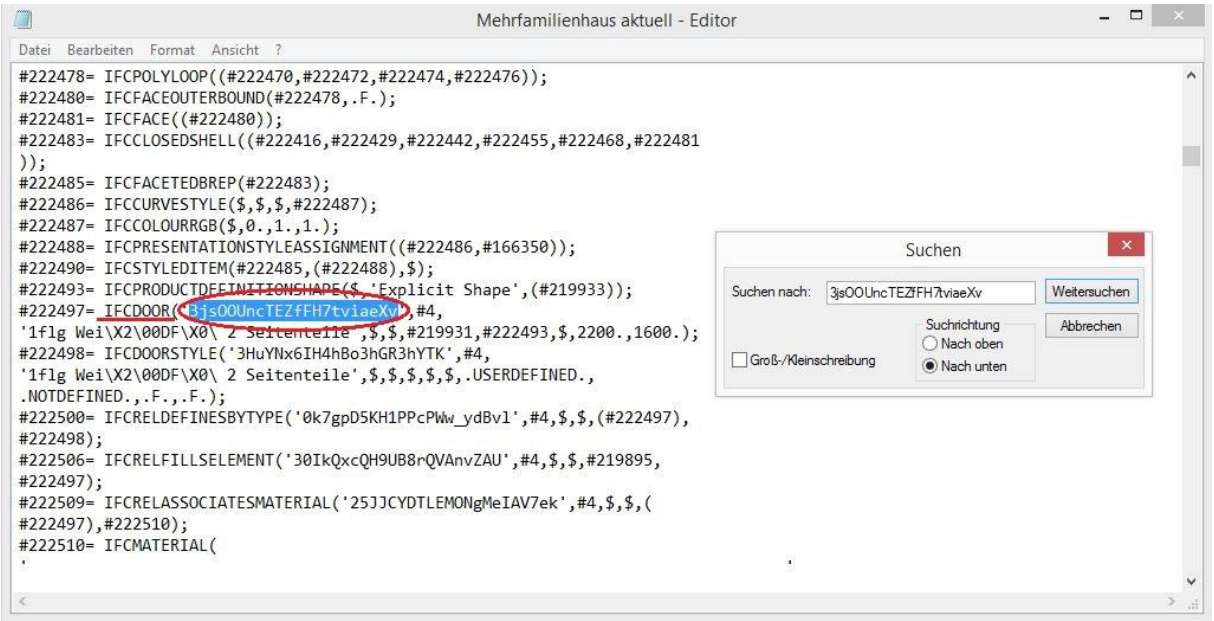


Abbildung 9 Darstellung der ausgewälten Tür (aus Abbildung 8) im Editor

Wie man der Abbildung 9 entnehmen kann, stimmt die ID aus dem Model Viewer mit derjeniger aus dem Editor überein und auch im Editor wird ersichtlich, dass es sich um dieselbe weiße einflügelige Tür („IFCDOOR“) mit zwei Seitenteilen handelt. Damit lässt sich dieses

Bauteil immer wieder finden, auch wenn sich das Bauteil hinsichtlich seiner geometrischen bzw. semantischen Beschreibung im IFC-Datenmodell verändern sollte.

4.2 Information Delivery Manual – Standardisieren von Datenaustauschszenarie

Die IFC, die in objektorientierter Weise strukturiert sind, entsprichen nach obiger Beschreibung einem neutralen Datenformat zur Beschreibung eines digitalen Gebäudemodells in einem umfassenden Datenmodell. Demnach repräsentiert das IFC-Format den Gegenstand des modellbasierten Datenaustausches nach dem Open BIM-Konzept innerhalb eines Datenaustauschszenarios. Diese ist in einem Planungs-, Ausführungs- und Ojektüberwachungsprozess zum Aufbau und zur Pflege eines digitalen Abbilds des geplanten Gebäudes eingebettet. Betrachtet man hinsichtlich des Planungsprozesses beispielsweise einen IFC Workflow, wie in Abschnitt 2.3 dargestellt, so wird ersichtlich, dass die Übermittlung des kompletten Datenmodells von einem Architekten an einen Tragwerksplaner nicht von Belangen ist: Dementsprechend sind für einen Statiker weder Möblierungsobjekte noch Wandtapezierungen (usw.) eines Modells von Interesse, sondern lediglich zur Lastabtragung maßgeblich beteiligte tragende Bauteile (wie Decken, Wände o.ä.). Das heißt, man würde nur bestimmte Teilmengen aus dem IFC-Schema herausfiltern. Dies geschieht über die Reihung folgender Teilschritte: *Information Delivery Manual (IDM), Business Process Modeling and Notation (BPMN)* und *Model View Definition (MVD)*. Dabei wird geregelt, „[…] *welche* Informationen von *wem wann welchem* Projektbeteiligten *wie* zur Verfügung gestellt werden sollen.“ (Borrmann, Building Information Modeling, 2015)

Grundsätzlich lässt sich bei dieser Interaktion (von IDM, BPMN und MVD) zwischen fachlichen Anforderungen und deren technischen Umsetzung unterscheiden (Borrmann, Building Information Modeling, 2015). Hält man sich anhand des vorherigen Beispiels vor Augen, dass die technische Umsetzung in eine Software von einem der Baubranche fachfremden Informatiker implementiert würde, so ist es einleuchtend, dass ein solcher nicht in der Lage ist, bauspezifische Zusammenhänge zu erkennen, um entscheiden zu können, welche Objekte des Modells statischen Anforderungen genügen.

Deswegen werden im ersten Schritt prozessbeteiligte Fachleute voran gestellt, die den Informationsfluss eines Teilprozesses analysieren, Datenübergabepunkte definieren und dabei genau untersuchen, welche Informationen in welchem Datenformat zwischen welchen Schnittstellen ausgetauscht werden (vgl. Nr. 1-3 in Abbildung 10). Die beiläufige Darstellung dieser Untersuchung (vgl. Nr. 2 in Abbildung 10) stellt eine grafische Notation aller Geschäftsprozesse in sog. Prozessplänen, den *Process Maps*, dar, anhand derer ein Anforderungskatalog (*Exchange Requirements)* in tabellarischer Form abgeleitet wird – sollte es sich an einem Übergabepunkt um einen modellbasierten Informationsaustausch handeln (vgl. Nr. 4 in Abbildung 10). (Borrmann, Building Information Modeling, 2015)

Anschaulich käme das der Definition aller an einem statischen Strukturmodell erforderlichen Objekte und Informationen gleich (z.B. Betonfestigkeitsklassen verschiedener Bauteile usw.).

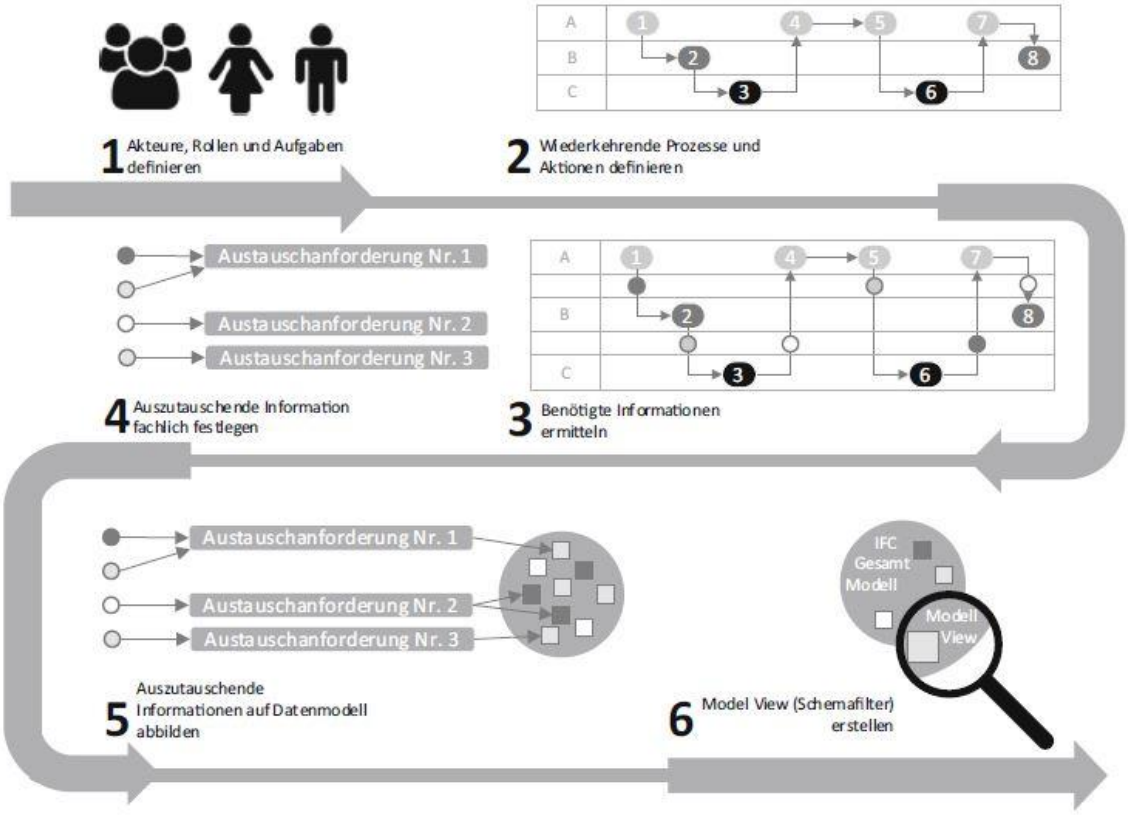


Abbildung 10 Übersicht aller Teilvorgänge vom IDM zum MVD[8]

Diese aufgegliederte Übersicht über den Teilprozess in Verbindung mit den aufgestellten fachlichen Anforderungen vereinbart die von buildingSmart entwickelte Methode Information Delivery Manual, deren Gebrauch und methodischen Mechanismen in der Norm ISO 29481 verankert sind (ISO 29481-1:2010-05, 2010). Innerhalb eines IDM ist das BPMN-Diagramm als eine konkrete Anwendung eines Prozessplanes zur anschaulichen und grafischen Darstellung des Datenflusses bzw. des Datenaustausches weitverbreitet (vgl. Abbildung 11).

[8] Borrmann, A. (2015). *Building Information Modeling.* (A. Borrmann, M. König, C. Koch, & J. Beetz, Hrsg.) Wiesbaden: Springer.

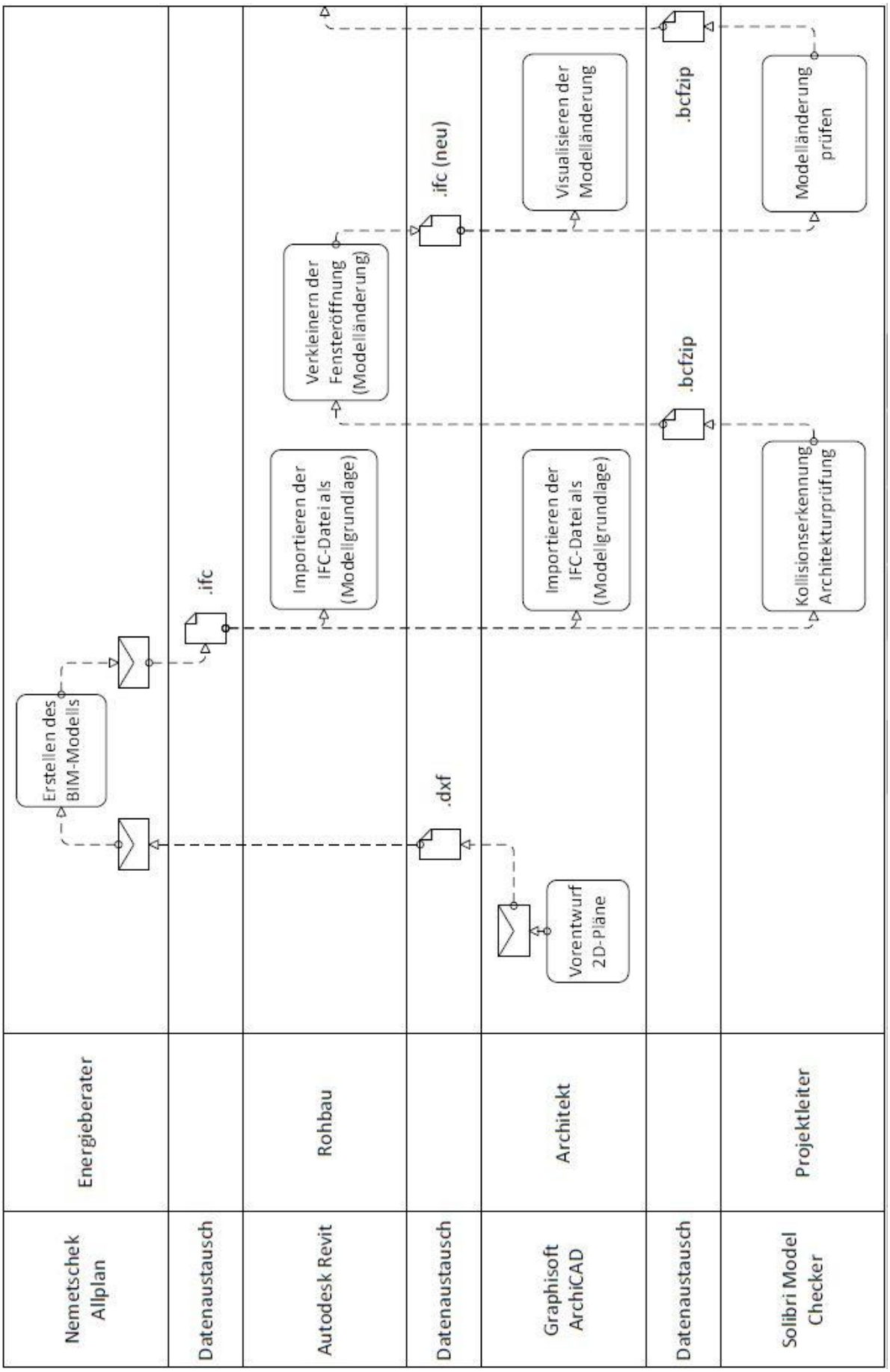


Abbildung 11 BPMN-Diagramm bezüglich des Beispielszenarios in Abschnitt 6 - erstellt mit Microsoft Visio

Im zweiten Schritt folgt die technische Umsetzung, also die Implementierung der zuvor definierten und auf das (komplette) Datenmodell abgebildeten Exchange Requirements in Software (vgl. Nr. 5 & 6 in Abbildung 10). Demnach werden bei diesem Schritt oftmals mehrere dieser Anforderungen zu einer konkreten Model View Definition zusammengefasst. Das Resultat entspricht letztendlich einer Teilmenge des IFC Datenschemas, die „maßgeschneidert“ auf das zugrundeliegenden Datenaustauschszenario ist, da nur diejenigen Entitäten und Attribute aus dem Datenmodell übernommen werden, die bei einem spezifischen Austausch von Interesse sind. (Borrmann, Building Information Modeling, 2015)

Abschließend sei darauf hingewiesen, dass Model View Definitions die Grundlage bilden, Softwareprodukte zu zertifizieren, denn die gezielte Steuerung von Exchange Requirements in Abhängigkeit von Information Delivery Manuals bietet ein ideales Instrument bei der Festlegung von Zertifizierungssystemen. (Borrmann, Building Information Modeling, 2015)

5 Beschreibung des BIM Collaboration Formats

Das *BIM Collaboration Format (BCF)* wurde in Kooperation der beiden Firmen Solibri und Tekla in der buildingSmart-Organisation im Jahr 2010 entwickelt. Das Ziel ist es, Änderungen an einem Gebäudemodell im Laufe eines Workflows über ein offenes Format auf Basis des IFC-Datenmodells in intelligenter Weise zwischen verschiedenen BIM-Softwareprodukten zu kommunizieren und Informationen anfügen zu können ohne dabei das gesamte IFC-Modell übertragen zu müssen. (Stangeland, 2012) Anders als beim IFC-Datenaustausch – eingeschränkt durch MVDs oder auch nicht – werden nämlich keine Modelldaten übergeben, sondern ähnlich einem „virtuellen Notizzettel“ werden lediglich Informationen über die betroffenen Bauteile beigefügt. Sowohl dazugehörige Kommentare als auch ein Screenshot zur Visualisierung können angeheftet werden. (Borrmann, Building Information Modeling, 2015)

Die dahinterstehende Idee ist es, die Kommunikation im Arbeitsprozess aus dem Modell auszulagern, um so die Verwaltung und Dokumentation von auftretenden Mängel und Problemen (engl. issues) beim Zusammenführen von Teilmodellen verschiedener Disziplinen (vgl. IFC Workflow in Abschnitt 2.3) zu vereinfachen (Stangeland, 2012). So entziehen sich komplexere Überlagerungen von solchen Modellen schnell der menschlichen Vorstellungskraft.

Besonderer Bedeutung kommen dabei den sog. Model Checkern zu, die einzelne Gebäudemodelle bzw. überlagerte Teilmodelle anhand festgelegter Regeln (z.B. die Erkennung von geometrischen Kollisionen) überprüfen und dabei Konflikte erkennen, die dann anschließend als BCF-Datei abgelegt oder versendet werden können.

Aus technischer Sicht basiert das BCF-Format, welches sich in eine ZIP-Datei komprimieren lässt, auf der plattformunabhängigen Programmiersprache *Extensible Markup Language (XML)*. Die komprimierten Dateien sind verschiedenen Verzeichnissen untergeordnet, wobei ein einzelnes Verzeichnis genau einen Mangel repräsentiert und über eine eindeutige *Universally Unique Identifier (UUID)* – ähnlich der GUID im IFC-Datenschema – genau bestimmt ist. Ein der UUID gleichnamiges Verzeichnis komprimiert dabei drei Dateien: eine *Markup*-XML-Datei, eine *Viewpoint*-XML-Datei und eine Abbildung des Mangels im PNG-Format als sog. „Snapshot“. Im Folgenden sollen die Bestandteile anhand einer BCFZIP-Datei, die im Solibri Model Checker bezüglich des Bespielszenarios (vgl. Abschnitt 6) generiert worden ist, näher beschrieben werden. (Borrmann, Building Information Modeling, 2015)

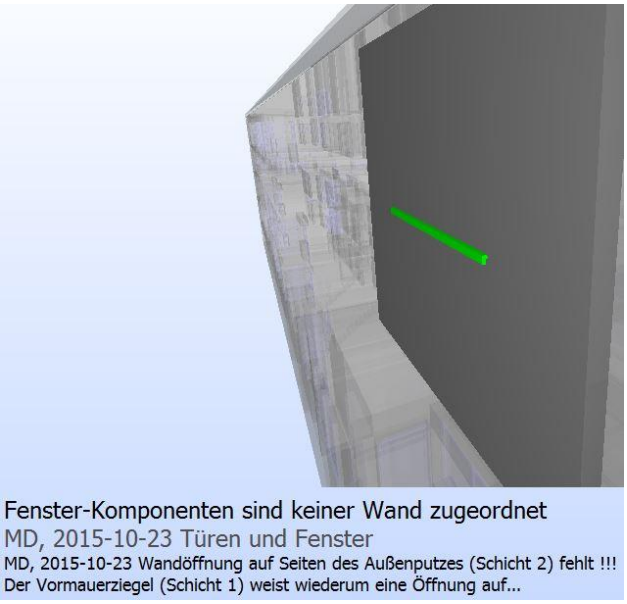


Abbildung 12 Ein in Solibri erkannter Issue: Die Außenputzschicht überdeckt das dahinter liegende Fenster

Den Ausgangspunkt stellt, wie bereits vorher beschrieben, ein im Solibri Model Checker erkannter Mangel im Beispielmodell aus Kapitel 6 (vgl. Abbildung 12). Daraus lässt sich die dazugehörige *Markup*-XML-Datei ableiten (vgl. Abbildung 13), die strukturierende Informationen zum Mangel beinhaltet wie etwa den Namen und die Beschreibung des Mangels oder das Datum, den Namen des Autors bzw. des zugewiesenen Verantwortlichen und die Priorität des Mangels.


```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Markup xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="markup.xsd">
    <Header>
        <File IfcProject="0ck1dnk1r3qfGYdEqoptRH">
            <Filename>C:\Users\Michael\Desktop\Bachelorarbeit\IFC
\Mehrfamilienhaus.ifc</Filename>
            <Date>2015-10-21T17:22:43+02:00</Date>
        </File>
    </Header>
    <Topic Guid="63165b4b-fbc6-4f42-93d8-6cbfa2347737"
TopicType="Error" TopicStatus="Zugewiesen">
        <Title>Fenster-Komponenten sind keiner Wand
zugeordnet</Title>
        <Index>0</Index>
        <CreationDate>2015-10-23T17:38:55+02:00</CreationDate>
        <CreationAuthor>MD</CreationAuthor>
        <ModifiedDate>2015-10-23T17:38:55+02:00</ModifiedDate>
        <ModifiedAuthor>MD</ModifiedAuthor>
        <AssignedTo></AssignedTo>
        <Description>TÃ¼ren und Fenster
</Description>
    </Topic>
    <Comment Guid="9690dae5-252b-42e7-9bd5-b234f4afad63">
        <VerbalStatus>Zugewiesen</VerbalStatus>
        <Status>Error</Status>
        <Date>2015-10-23T18:03:59+02:00</Date>
        <Author>MD</Author>
        <Comment>WandÃ¶ffnung auf Seiten des AuÃŸenputzes
(Schicht 2) fehlt !!!
Der Vormauerziegel (Schicht 1) weist wiederum eine Ã-ffnung
auf...</Comment>
        <Topic Guid="63165b4b-fbc6-4f42-93d8-6cbfa2347737"/>
        <Viewpoint Guid="c841da49-da81-4ec8-8463-81ba8740b0bf"/>
    </Comment>
    <Viewpoints Guid="c841da49-da81-4ec8-8463-81ba8740b0bf">
        <Viewpoint>viewpoint.bcfv</Viewpoint>
        <Snapshot>snapshot.png</Snapshot>
    </Viewpoints>
</Markup>
```


Abbildung 13 Die *Markup*-XML-Datei des aus Abbildung 11 hervorgegangenem Mangel

Weiter beschreibt die *Viewpoint*-XML-Datei die dem angefügten „Snapshot“ (im PNG-Format) zugrundeliegende Kameraposition bzw. –richtung und darüberhinaus alle auf dem „Snapshot“ abgebildeten Bauteile, die über die GUID des zugehörigen IFC-Datenschemas verlinkt sind und daher jederzeit im IFC-Modell lokalisiert werden können. Dadurch, dass sich beim „Snapshot“ des angeführten Beispiels sehr viele Bauteile im Hintergrund befinden, ist in Abbildung 14 nur ein Auszug der kompletten *Viewpoint*-XML-Datei dargestellt.


```
Selected="false" Visible="true">
            <OriginatingSystem>Allplan</OriginatingSystem>
        </Component>
        <Component IfcGuid="0AuXTmbKn1uPQbwLxMsJ8d"
Selected="false" Visible="true">
            <OriginatingSystem>Allplan</OriginatingSystem>
        </Component>
        <Component IfcGuid="1NNG_Sejj5VeMEShUcqbs5"
Selected="false" Visible="true">
            <OriginatingSystem>Allplan</OriginatingSystem>
        </Component>
    </Components>
    <PerspectiveCamera>
        <CameraViewPoint>
            <X>235.07060992100836</X>
            <Y>-84.52143530877503</Y>
            <Z>10.386856030243477</Z>
        </CameraViewPoint>
        <CameraDirection>
            <X>-0.8318141999632207</X>
            <Y>0.4958972260864183</Y>
            <Z>-0.24934128799567648</Z>
        </CameraDirection>
        <CameraUpVector>
            <X>-0.21417003985796199</X>
            <Y>0.12768035058920335</Y>
            <Z>0.9684156762984878</Z>
        </CameraUpVector>
        <FieldOfView>60.0</FieldOfView>
    </PerspectiveCamera>
</VisualizationInfo>
```


Abbildung 14 Auszug aus der *Viewpoint*-XML-Datei des aus Abbildung 11 hervorgegangem Mangel

5.1 Optimierung des IFC-Workflows durch KUBUS

Seit der Einführung von BCF im Jahre 2010 mit seiner ersten Version 1.0 wurde das Format 2014 zu Version 2.0 aktualisiert und es wird derzeit neben den Programmen Solibri Model Checker, Magicad, Tekla Structures, Tekla BIMsight, DDS von vielen weiteren BIM-Anwendungen unterstützt (Solibri, 2013).

Darüber hinaus hat KUBUS, ein Partnerunternehmen von Solibri, eine Plug-In-Schnittstelle namens *BCF Manager* jeweils für Graphisoft ArchiCAD und Autodesk Revit - zwei renommierte BIM-Modellierungssoftwareprodukte – entwickelt. In Verbindung mit einem passenden *BCF Connector* für den SMC lassen sich damit Issues detektieren und im Anschluss problemlos über den BCF Manager in die zugrunde liegende BIM-Software kommunizieren. Es können aber auch ohne Umwege über einen Model Checker Issues in ArchiCAD bzw. Revit generiert und wiederum mit dem BCF Manger kommuniziert werden. (Solibri, 2013)

Der große Vorteil, der den IFC Workflow grundlegend erleichtert, stellt die ausgelagerte Kommunikation - über das Modell hinweg auf Basis des IFC-Datenschemas. Ist demzufolge eine IFC-Datei zwischen zwei verschiedenen Disziplinen, die getreu dem Open BIM-Prinzip

jeweils in Teilmodellen arbeiten, ausgetauscht worden, so können Software übergreifend durch Erzeugen von BCFZIP-Dateien Anmerkungen zu Bauteilen getroffen werden, indem deren einmalige GUIDs aus dem IFC-Datenmodell aufgegriffen werden. Ergeben sich im voranschreitenden Modellierungsprozess (in der Planungsphase) Änderungen innerhalb eines Teilmodells oder sollten Anpassungen anderer Teilmodelle gefordert werden (z.B. geometrische Abmessungen von Bauteilen), so unterstützt der BCF Manager als zentrales Verwaltungssystem einerseits das Erstellen einer solchen BCFZIP-Datei in einer Software, andererseits das Auslesen derjeniger in der „Zielsoftware“. Das beim Adressat ankommende Ergebnis in Form einer BCF-Datei veranschaulicht das übergebene Problem, indem durch einen einfachen Klick die korrespondierenden Bauteile in einer gewissen Kameraposition versehen mit angefügten Kommentaren bzw. Anweisungen angezeigt werden.

Alternativ würde man gemäß dem IFC Workflow für jede vorgenommene Änderung oder auch für jedes auftretende Problem, was innerhalb eines Teilmodells Auswirkungen auf ein anderes hat, eine neue IFC-Datei des modifizierten Modells erzeugen und versenden, obwohl sich aus Sicht des ursprünglichen Datenschemas nur eine Kleinigkeit geändert hat. Bezogen auf die Datenmenge ist es also erschwinglicher, Abwandlungen in einem Modell über eine BCF-Datei mitzuteilen und Anpassungen im ausgetauschten Teilmodell zu fordern, als eine überarbeitete umfassende IFC-Datei zu übermitteln.

Es zeigt sich, dass das IFC-Format aufgrund seiner komplexen Datenstruktur nur bedingt dafür geeignet ist, Modellierungsprozesse zu dokumentieren. Auch das Kommunizieren mit BCF-Dateien lässt sich über einen BCF Manager folglich wesentlich flexibler gestalten.

BCF Manager

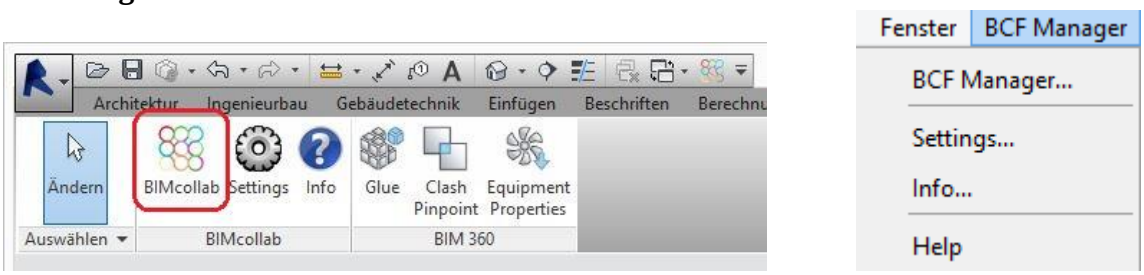


Abbildung 15 Die Plug-In-Schnittstelle des BCF Managers in Autodesk Revit (links) und ArchiCAD (rechts)

Im Grunde ist die Funktionsweise des BCF Managers simpel: Mit Hilfe der Plug-In-Schnittstelle werden bei einem Model-Check oder innerhalb eines Modellierungsprogramms Issues erzeugt und als BCFZIP-Datei lokal abgelegt. Im nächsten Schritt wird diese Datei dann in einer anderen Software anhand des BCF Managers eingelesen, wobei Informationen über die betroffenen Bauteile und eine genaue Kameraposition mit dem entsprechenden „Snapshot“ gemäß den im BCFZIP komprimierten Dateien (Markup, Viewpoint, Snapshot) aus der Ausgangssoftware ersichtlich sind. Demnach unterstützt der BCF Manager das Erzeugen und Auslesen von BCFZIP-Dateien als integrierte Schnittstelle der entsprechenden Software, um Konflikte im Modell zu lokalisieren bzw. zu beheben. Dadurch wird der BCF Workflow als ein Import- und Export-Plug-In gefördert.

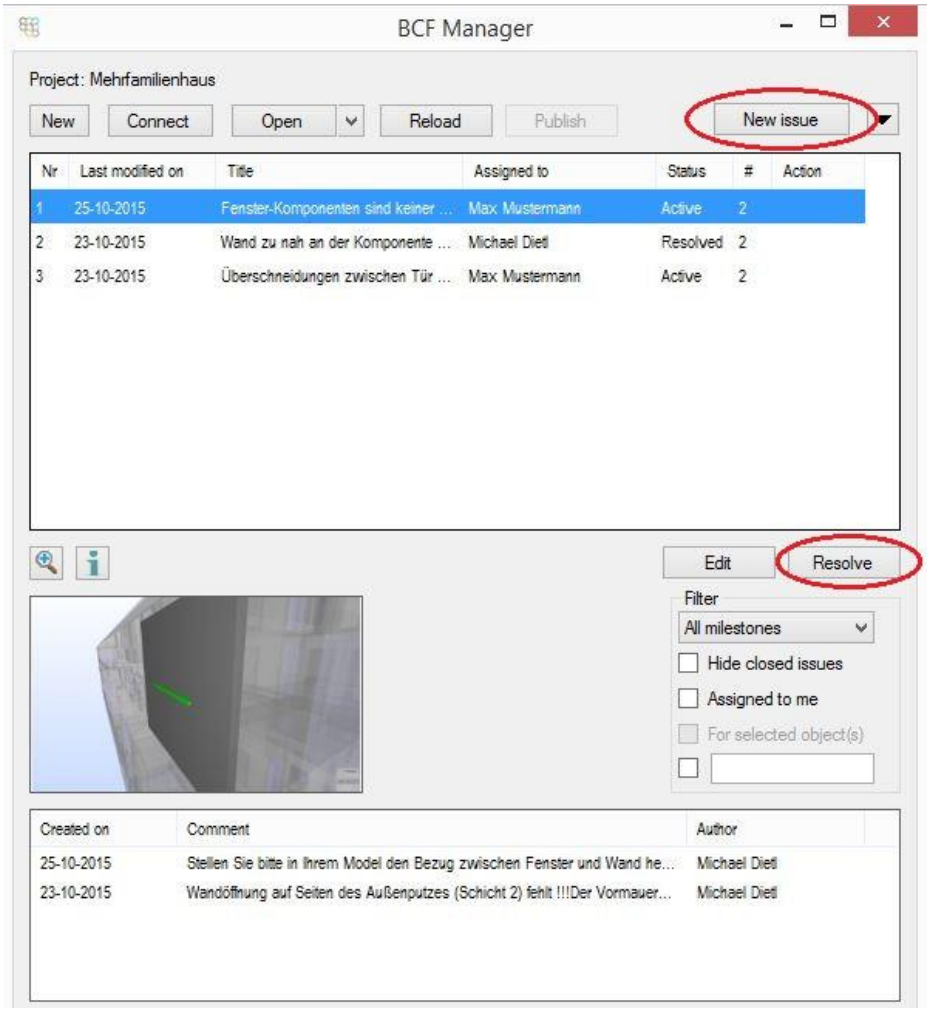


Abbildung 16 Die Benutzeroberfläche des BCF Managers in Revit oder auch in ArchiCAD mit drei erzeugten Issues aus dem Beispielszenario in Abschnitt 6

Die beiden wichtigsten Werkzeuge des BCF Managers stellen nach Abbildung 16 das Lösen bzw. das Erstellen eines neuen Issues.

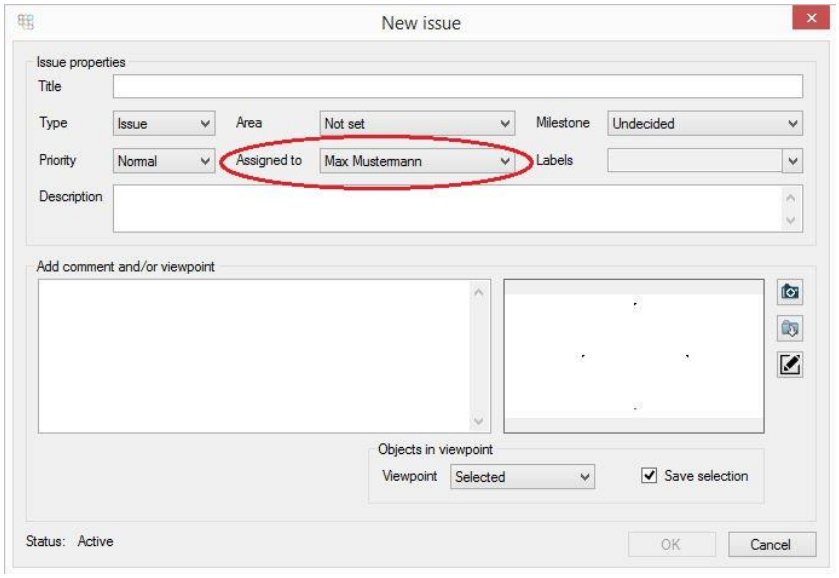


Abbildung 17 Benutzoberfläche innerhalb des BCF Manager zum Erzeugen eines neuen Issues mit den grundlegendsten Einstellungsmöglichkeiten & Informationen gemäß der BCF-Datenstruktur

BIMcollab

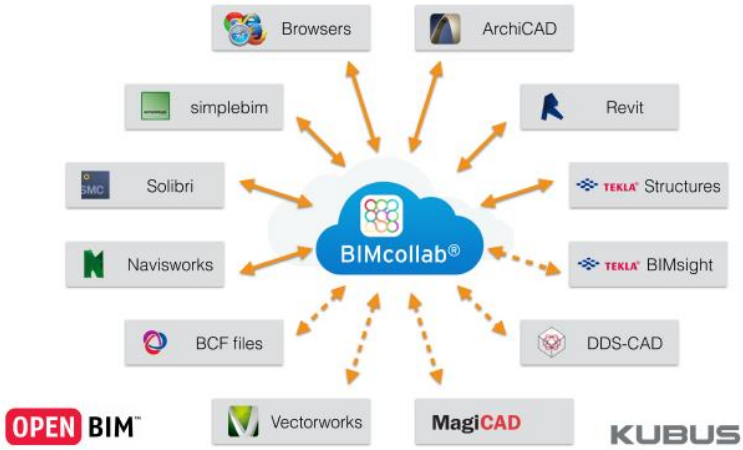


Abbildung 18 Überblick über die durch BIMcollab vernetzten Softwareprodukte[9]

BIMcollab entspricht im Wesen einer Ergänzung zum BCF Manager mit dem kleinen Unterschied, dass BCFZIP-Dateien nicht lokal abgespeichert werden, sondern auf eine cloudbasierte Internetplattform, von wo aus jeder Projektbeteiligte die hochgeladenen Issues einsehen kann. Der ergänzende Charakter zeigt sich dadurch, dass innerhalb des BCF Managers die Möglichkeit besteht, sich mit der BIMcollab-Plattform online zu verbinden, um alle im Projekt adressierten Issues zu laden, zu bearbeiten und schlussendlich wieder zu aktualisieren.

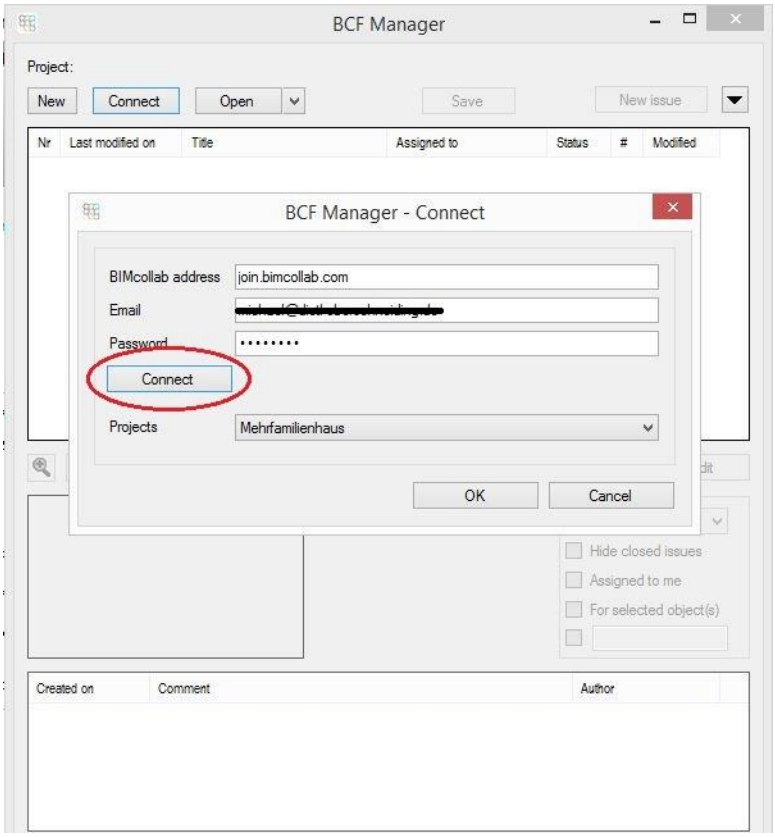


Abbildung 19 Herstellen einer Verbindung zu BIMcollab über den BCF Manager

[9] http://www.bimcollab.com/en/Files/Manuals/BCF-Manager-Manual.aspx

Hinter dieser „Connection“ zum BCF Manager steht, wie oben beschrieben, eine Webseite, die als zentrales internetgebundenes Problem- und Mängelmanagement fungiert. Gemäß Abbildung 20 erkennt man, dass BIMcollab primär zwischen einzelne Projekte unterscheidet, die des Weiteren vielerlei Issues enthalten. Wesentlich hierbei ist die auf einzelne *Team member* verteilte Arbeitsweise. Der Projektleiter legt als Administrator fest, ob eingeladene Projektteilnehmer als „Bearbeiter“ oder als „Betrachter“ involviert werden und regelt damit die am Projekt verteilten Rechte. Anhand von diesem Verwaltungssystems, auf das jeder Projektbeteiligte jederzeit zugreifen kann, wird gewährleistet, dass stets an den aktuellen Issues gearbeitet wird.

Abbildung 20 Informationen über das BIMcollab-Projekt „Mehrfamilienhaus“ aus dem Beispielszenario in Abschnitt 6

In gleicher Weise wie bem BCF Manager können nicht nur hochgeladene Issues angezeigt werden, sondern darüberhinaus besteht die Möglichkeit, auf der Website neue Issues zu generieren, um sie bei Bedarf über die mit BIMcollab verbundene BCF Manager-Schnittstelle einer Modellierungssoftware dort anzeigen zu lassen.

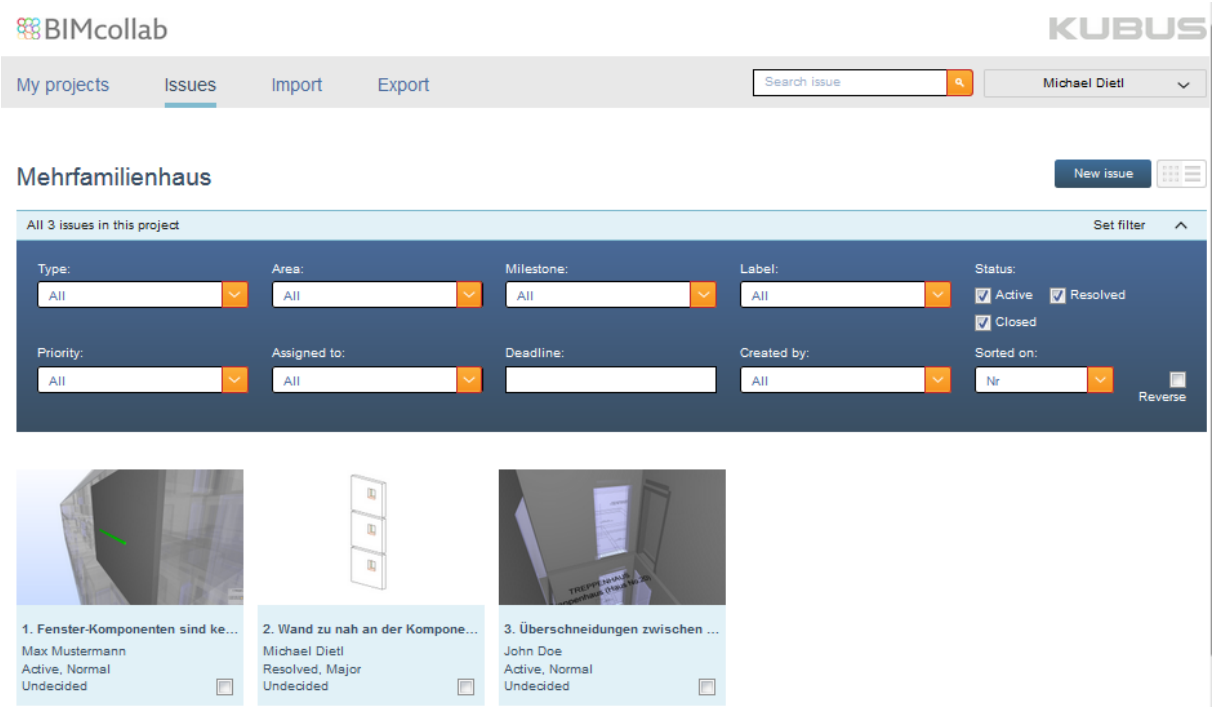


Abbildung 21 Übersicht über die Darstellung einzelner Issues in BIMcollab

Diese Plattform erleichtert letztendlich nicht nur das Versenden lokal gespeicherter BCFZIP-Dateien, sondern alle vorgenommenen Änderungen und Markierungen werden in übersichtlicher Weise und deren zeitlicher Abfolge webbasiert dokumentiert, sodass über die Planungsphase hinweg alle auftretenden Probleme auch bei der Ausführung in völliger Transparenz eingesehen werden können.

5.2 Der BCF Prozess

Um zu verstehen wie ein BCF Workflow abläuft und wie verschiedene Softwareprodukte (hier Revit, ArchiCAD und SMC) zusammenarbeiten, wird das grundsätzliche Vorgehen im Folgenden beschrieben. Beispielhaft soll Revit als Modellierungssoftware dienen, dem eine Kollisionserkennung im SMC vorausgeht.

1. Untersuchen des Modells auf Unstimmigkeiten und Ableiten von BCF-Dateien

Aus dem zugrunde liegenden Revit-Modell wird vorerst eine IFC-Datei exportiert, die dann im Anschluss über den SMC eingelesen werden kann. Der SMC prüft des Weiteren das importierte Modell anhand vorgebenen Regeln – dem Beispiel soll eine geometrische Kollisionserkennung gemäß einer einfachen Architekturprüfung genügen – und findet womöglich Modellierungsfehler, die als einzelne Issues in eine sog. Präsentation zusammengefasst werden. Beim Festlegen eines dieser Issues werden den kritischen Bauteilen, die über deren GUID aus dem IFC-Datenschema eindeutig identifizierbar sind, gemäß der BCF-Datenstruktur Kommentare oder auch zusätzliche Dateien (z.B. im PDF-Format o.ä.) bzw. ein URL-Link angeheftet, um die Mängel so gut wie möglich zu beschreiben und evtl. Lösungsvorschläge zu schildern. Außerdem können Issues schon beim Erzeugen konkret einem bestimmten Projektbeiteiligten zugewiesen werden.

Solch eine Präsentation kann im SMC dann im nächsten Schritt als eine BCFZIP-Datei, in der alle erzeugten Issues aufgelistet sind, lokal abgespeichert oder direkt auf den BCF Server von BIMcollab hochgeladen werden.

2. Versenden der BCFZIP-Datei an Projekbeteiligte und Betrachten der Mängel in der präerieten Software über den BCF Manager

Diese BCFZIP-Datei kann folglich an einen Revit-Modellierer zurückgeführt werden, um Verbesserungen am Modell vornehmen zu lassen, oder aber man adressiert sie an einen weiteren Projektbeteiligten, der auf Basis der derselben zuvor ausgetauschten IFC-Datei beispielsweise mit ArchiCAD arbeitet. Unabhängig davon ob die ZIP-Datei erst lokal gespeichert und anschließend per E-Mail verschickt worden ist oder hingegen zuvor auf BIMcollab hochgeladen worden ist, können die Issues in der jeweiligen Software über den BCF Manager geladen werden. Mit einem einfachen Klick wird die gespeicherte Kameraposition aus dem SMC übernommen und die betroffenen Bauteile werden hervorgehoben, während die beigefügten Kommentare und Anweisungen zur Verfügung stehen.

3. Beheben der Mängel und Informieren über die Änderungen

Ausgehend von dem übernommenen Blickwinkel auf das Problem können die betroffenen Bauteile aus Sicht des Modellierers ohne weiteres lokalisiert werden, um in der Folge die beschriebenen Änderungen vorzunehmen. Ebenso wie beim Erstellen von Issues können auch beim Beheben von Mängeln Kommentare beigefügt werden, die die vorgenommenen Modifikationen genauer dokumentieren und somit transparenter gestalten. Beim *Resolve* eines Issues wird der zuvor als „aktiv“ bezeichnete Issue im BCF Manager zwar überschrieben, doch die durchgeführten Änderungen werden in einer Historie weiterhin aufgelistet. Abschließend wird die neue BCFZIP-Datei mit dem gelösten Problem wiederum entweder lokal abgespeichert und an den Urheber des Mangels versendet, um ihn über die Änderungen zu benachrichtigen, oder eben auf den BCF Server von BIMcollab hochgeladen.

Dabei sei aber zu beachten, dass bei dieser versendeten BCF-Datei keine Modelländerungen übergeben werden. Es kann zwar ein Screenshot angefügt werden, der die Änderung kenntlich macht, aber dadurch, dass im BCF-Format keine IFC-Datenstruktur mit inbegriffen ist, bezieht sich der Issue weiterhin auf die ursprünglich ausgetauschte IFC-Datei.

6 Beispielszenario

Nachstehend sei in diesem Kapitel ein Szenario beschrieben, das alle aufgeführten Begrifflichkeiten mit deren beschriebenen Arbeitsabläufen in Zusammenhang mit der Kollaboration verschiedener Softwareprodukte veranschaulichen soll. Dahinter steht ein fiktiver, aber an die reale Arbeitswelt angepasster, Workflow:

Der Bauherr beauftragt einen Architekten damit, einen Vorentwurf (mit Grundrissen und Schnitten) eines Mehrfamilienhauses zu gestalten. Daraufhin erzeugt der Architekt mit Graphisoft ArchiCAD die kennzeichnenden 2D-Pläne. Diese Pläne dienen als Modellierungsgrundlage für das BIM-Modell, das - vom Bauherrn gefordert - durch ein Ingenieurbüro zur Mengenermittlung für eine überschlägige Ökobilanzierung in Nemetschek Allplan erstellt wird. Im Anschluss wird das Modell per IFC-Datei allen Projektbeteiligten zur Verfügung gestellt. Dazu zählen neben dem Architekten das ausfürende Bauunternehmen, das mit Autodesk Revit arbeitet, und ein vom Bauherrn engagierter Projektleiter, der den Überblick mit dem Solibri Model Checker behält. Auf Basis dieses IFC-Datenmodells werden letztendlich auftretende Mängel über den BCF Manager kommuniziert.

Programme:	**Anwendungen:**
Graphisoft ArchiCAD 19	2D-Pläne, Import der IFC- und BCF-Dateien
Nemetschek Allplan 2015	3D-Modellierung, Export der IFC-Datei
Autodesk Revit 2016	Import der IFC- und BCF-Dateien
Solibri Model Checker v9.6	Kollisionserkennung, Export von BCF-Dateien

Tabelle 1 Überblick über die verwendeten Softwareprodukte im Beispielszenario

Übersichtlicher ist der komplette Ablauf im BPMN-Diagramm aus Abschnitt 4.2 (siehe Abbilddung 11) aufgezeigt.

6.1 Modellierung in Allplan

Abbildung 22 Gerenderte Ansicht des kompletten Gebäudemodells

Das Building Information Model des Mehrfamilienhaus (vgl. Abbildung 22) ist in Allplan mit dem Ziel erzeugt worden, umgehend Mengen einzelner Bauteile zu ermitteln. Dazu stellt Allplan mit den sog. *Reports* ein sehr einfaches Instrument, um schon während der Erstellung des Gebäudemodells Eigenschaften (wie z.B. zugrunde liegende Geometriedaten, Mengen, Massen usw.) der bereits modellierten Bauteile in einem ausgelagerten Bericht auszuwerten. Abhängig vom jeweiligen Detaillierungsgrad des betrachteten Modells können beliebig genaue Berichte generiert werden. Wie man dem gerendertem Treppenhaus in Abbildung 23 entnehmen kann, sind hinsichtlich des Innenausbaus auch Fußboden- und Wandaufbau modelliert worden, deren Schichtenaufbau mit den jeweiligen Mengen in einem entsprechendem Report ausgelesen werden können (vgl. Abbildung 24).

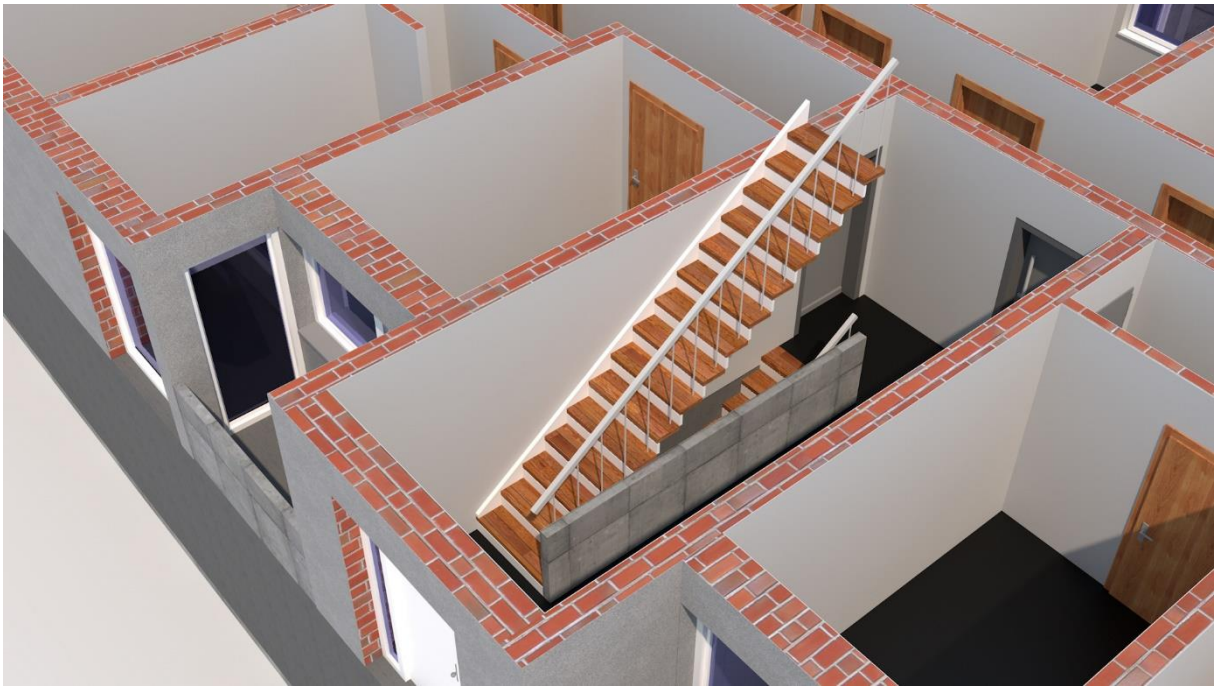
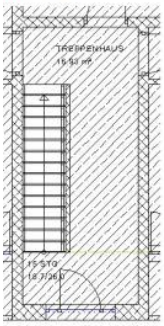


Abbildung 23 Rendering eines Treppenhauses im hinblick auf verschiedenen Wandaufbauten und Ausbauobjekte

Mit Abschluss der Modellierungsarbeiten ist aus dem kompletten Gebäudemodell eine IFC-Datei exportiert worden, die im weiteren Verlauf des Szenarios als Kollaborationsgrundlage zwischen den verschieden Plattformen dient.

Bezeichnung	Funktion	Grundfläche [m²]	Volumen [m³]	Umfang [m]
Erdgeschoss				
Treppenhaus (Haus No.20)	TREPPENHAUS	16,931	44,866	26,230



	Objektname	Material	Menge	
Ausbau	Bodenfläche	PVC Belag	14,192	m²
	Bodenfläche	Schwimmender Estrich	14,192	m²
	Bodenfläche	Trittschalldämmung	14,192	m²
	Bodenfläche	Wärmedämmung	14,192	m²
	Seitenfläche	Dispersionsfarbe scheuerfest	39,147	m²
	Seitenfläche	Kalk-Gips-Innenputz	39,147	m²
Inventar	Fenstermakro, -SmartPart	1flg Weiß 2 Seitenteile	1,000	St
Öffnungen	Fenstertüröffnung	Treppenhaus (Haus No.20)	-3,520	m²

Abbildung 24 Auszug aus einem in Allplan erzeugtem Raumbuch-Report

6.2 IFC-Import in ausgewählte Programme

Wie es der IFC Workflow gemäß der Open BIM-Philosophie vorsieht, wird die aus Allplan exportierte IFC-Datei im Folgenden in die am Szenario beteiligten Programme importiert. Die Beschreibung dieses eher trivialen Vorgangs sei der Tatsache geschuldet, dass der Import- bzw. Export von IFC-Dateien in manchen Softwareprodukten noch nicht ganz fehlerfrei abläuft (Stand 2015).

Import in Graphisoft ArchiCAD 19

Behält man im Auge, dass es sich um ein relativ großes Gebäudemodell handelt (Grundrissabmessungen: 15m x 65m) und die zu importierende IFC-Datei dementsprechend umfangreich ist, vollzieht sich das Einlesen innerhalb weniger Minuten verhältnismäßig schnell. Auch die geometrische Darstellung aller Bauteilkomponenten überzeugt mit großer Sorgfalt. Nur drei übereinander liegende Fenster werden von der jeweilig zugehörigen Wand überdeckt (vgl. Abbildung 25), wobei es sich hierbei aber um einen Fehler im Modellierungsprozess handeln muss, denn auch im SMC, wie sich später zeigt, werden die Fenster exakt genauso dargestellt.

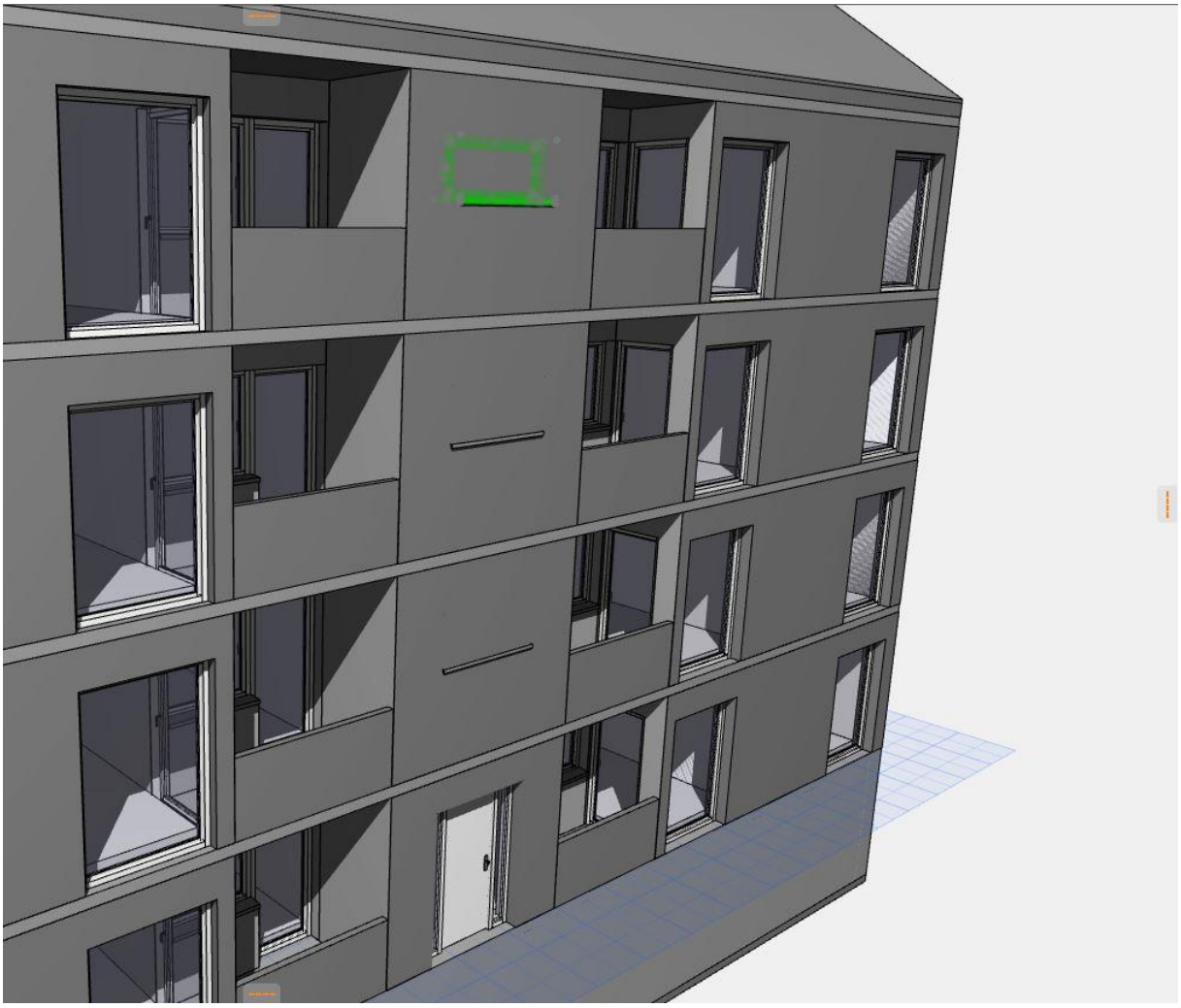


Abbildung 25 Von der jeweiligen Wand überdeckte Fenster in ArchiCAD

Import in Autodesk Revit 2016

Das erste Problem beim Importieren der IFC-Datei stellt der zeitliche Rahmen dieses Vorgangs dar: Um das komplette Modell einzulesen werden Stunden in Anspruch angenommen, was einen Import nur über Nacht ermöglicht. Ein möglicher Grund dafür könnte darin liegen, dass Revit zwar beispielsweise bei einer Wandkomponente aus dem IFC den Bauteiltyp „Wand“ erkannt hat, aber dabei für jede einzeln vorkommende Wand eine eigene Bauteilfamilie dazu erstellt hat, die aufgrund derart vieler Wände durchnummeriert worden sind: < Basiswand 219 >, < Basiswand 383 >, usw. (vgl. Abbildung 26). Zusätzlich zu den Wänden werden Fenster

und Türen in selber Weise nummeriert, wobei allein die Wände schon knapp 1000 neu definierte Bauteilfamilien zählen.

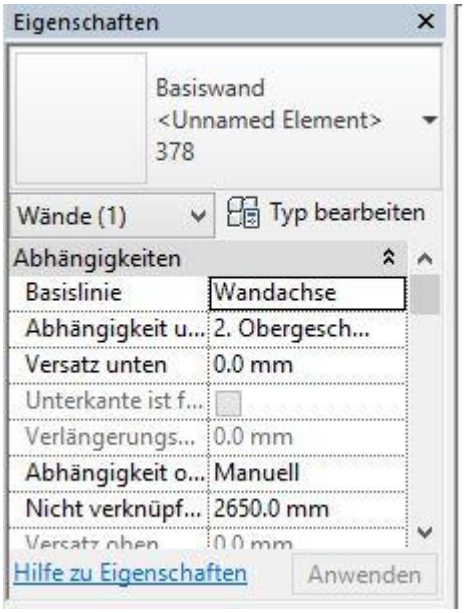


Abbildung 26 Auszug der Eigenschaften einer in Revit dargestellten Wand

Darüberhinaus bereitet die Visualisierung des Modells in Revit insofern Schwierigkeiten, dass in Allplan erstellte Betonbauteil schwarz dargestellt werden (vgl. Abbildung 27).

Zudem werden sämtliche Fenstertüren auf den Loggien in ihrer geomtrischen Anordnung falsch abgebildet und überlappen damit mit den angrenzenden Fenstern (vgl. Abbildung 27).

Was ebenfalls noch anzumerken ist: Diejenigen Wände, die die Fenster in ArchiCAD und in Solibri überdecken, weisen an passender Stelle eine Öffnung auf, weil die im ursprünglichen Wandaufbau gespeicherte Außenputzschicht nicht in Revit mit übernommen worden ist.

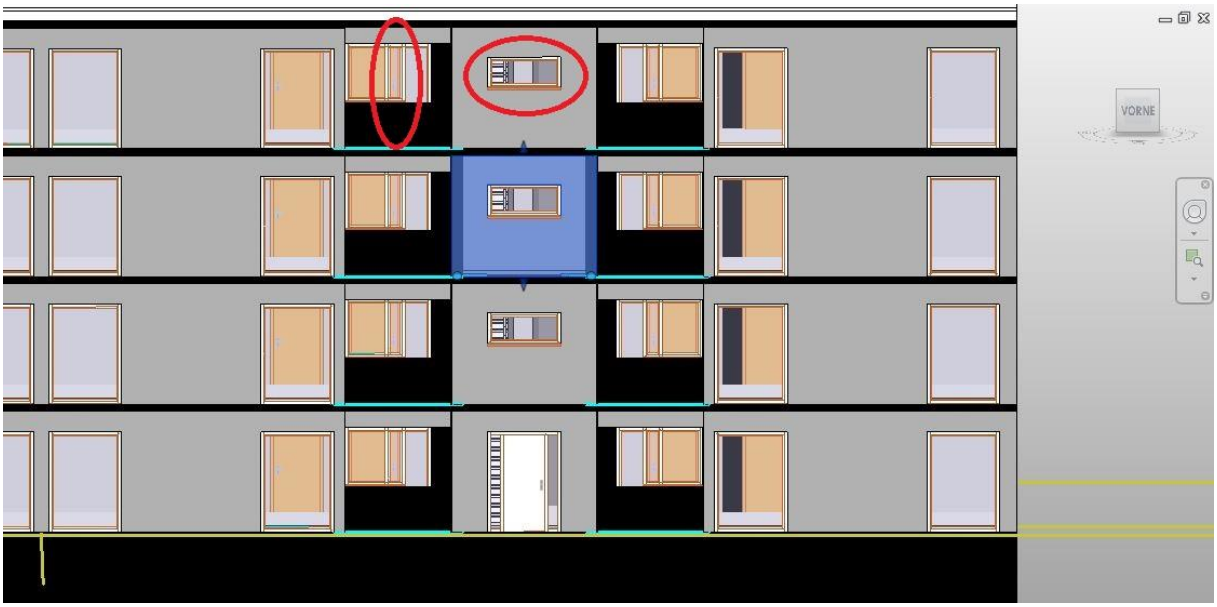


Abbildung 27 Fehlerhafte Darstellung in der Gebäudegeometrie und Probleme bei der Visualisierung von Betonbauteilen

Import in den Solibri Model Checker

Das Einlesen der IFC-Datei geschieht reibungslos innerhalb einer Minute und, wie auch schon beim ArchiCAD-Import beschrieben, tritt als einziger offensichtlicher Makel die drei von den den Wänden überdeckten Fenstern auf.

6.3 Interaktion der verschiedenen Programme – BCF Workflow

1. Projekteinstellungen in BIMcollab

Um die von KUBUS beschriebenen Funktionaliäten in vollem Umfang im Beispielszenario einzubeziehen und den Workflow sowohl detailliert zu dokumentieren als auch zentral zu verwalten, ist zu Beginn auf der BIMcollab-Plattform - parallel zum Gebäudemodell - ein Projekt mit dem Namen „Mehrfamilienhaus“ angelegt worden, zu dem zwei fiktive Projektteilnehmer (Max Mustermann und John Doe) eingeladen worden sind. Dabei arbeitet John Doe als Architekt mit der Software ArchiCAD, während Max Mustermann in seinem ausführenden Bauunternehmen hingegen bürointern Revit verwendet. Das Überprüfen des Modells auf Konsistenz übernimmt wiederum der beauftragte Projektleiter mit dem SMC.

Architekt	John Doe	Graphisoft ArchiCAD
Bauunternehmen	Max Mustermann	Autodesk Revit
Projektmanager	Michael Dietl	Solibri Model Checker

Tabelle 2 Überblick über die am Beispielprojekt beteiligten Parteien

2. Geometrische Kollisionserkennung im Solibri Model Checker und Anweisungen über BIMcollab

Im Zuge einer Architekturprüfung ist im SMC gemäß der Beschreibung aus Abschnitt 5.2 eine Präsentation erstellt worden, die folgende schwerwiegenden Mängel beinhaltet:

- Fenster-Komponenten sind keiner Wand zugeordnet

- **Wand zu nah an der Komponente Fenster**
  (entspricht dem zuvor oftmals erwähnten Problem mit den verdeckten Fenstern)
- Überschneidungen zwischen Tür und Wand

Hierbei sind bei der Bezeichnung die automatisch generierten Standardtitel beibehalten worden. Deswegen erscheinen sie ohne nähere Beschreibung nicht sehr aussagekrätig bzw. einleuchtend. Dennoch sind diese drei Issues zu einer BCFZIP-Datei zusammengefasst worden, welche anschließend mit Hilfe des BCF Connectors synchronisiert und auf BIMcollab hochgeladen worden sind.

Zurück auf der Webseite von BIMcollab: Der Issue „2. Wand zu nah an der Komponente Fenster“ ist dem Rohbau-Beauftragten Max Mustermann zugewiesen worden mit der Anweisung, er solle nicht nur die Verknüpfung zwischen Wand und Fenster im Modell wieder herstellen, sondern zusätzlich die Fensteröffnungen verkleinern, da neue Fenster eingebaut werden sollen (vgl. Abbildung 28).

Abbildung 28 Anweisung über Änderungen am Rohbau kommuniziert mittels BIMcollab

3. Bearbeitung des Issues in Revit

Nach dem Anmelden bzw. Verbinden mit BIMcollab im BCF Manger-Plug-In von Revit können die drei ursprünglich im SMC erzeugten Issues online bezogen werden und stehen Max Mustermann zur weiteren Bearbeitung bereit (vgl. hierzu auch Abbilung 16). Über einen einfachen Doppelklick auf den Issue werden innerhalb des Modells gemäß der hinterlegten Informationen aus der BCF-Datenstruktur automatisch die betroffenen Bauteile herausgefiltert und in der definierten Kameraposition dargestellt, woran direkt die geforderten Änderungen vorgenommen werden können (vgl. Abbildung 29).

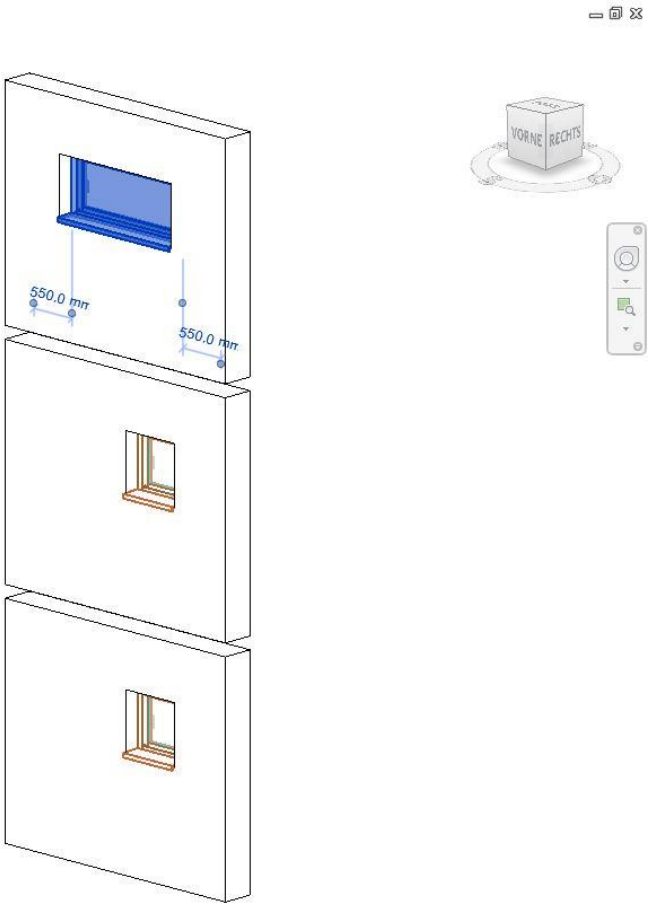


Abbildung 29 Bearbeitung des Modells anhand der in der BCF-Datei gespeicherten Ansicht

Schließlich kann man den Upload auf der BIMcollab-Plattform kontrollieren (vgl. Abbildung 30).

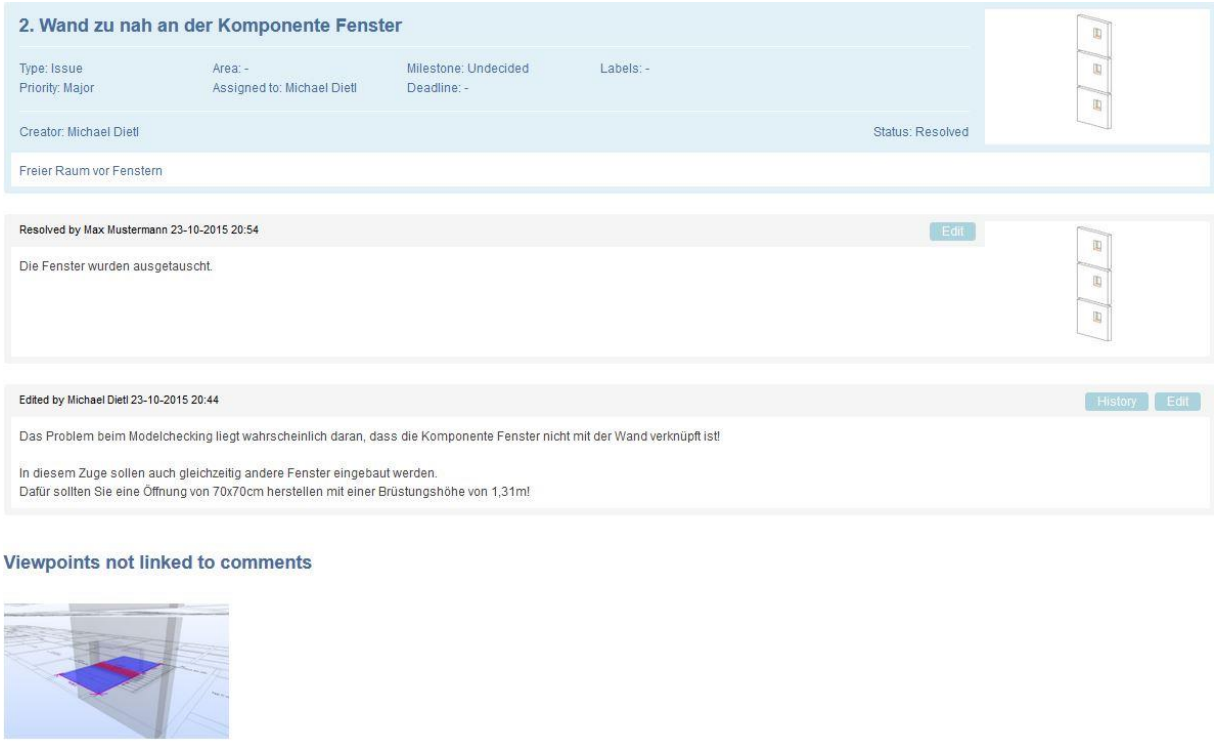


Abbildung 30 Darstellung des in Revit aktualisierten Issue auf der Webseite von BIMcollab

4. Auswerten der Änderungen in ArchiCAD

Will sich nun aber John Doe im Anschluss ein Bild von dem überarbeiteten Issue machen, indem er sich über die BCF Manager-Schnittstelle in ArchiCAD das gelöste Problem anzeigen lassen will, so taucht die Fehlermeldung nach Abbildung 31 auf.

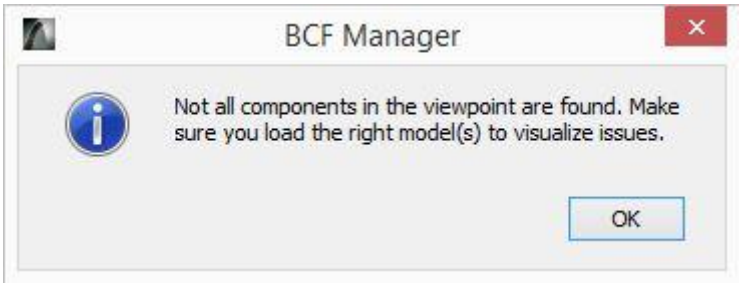


Abbildung 31 Fehlermeldung beim Öffnen der BCFZIP-Datei in ArchiCAD über den BCF Manager

Der Grund dafür liegt darin, dass die von Max Mustermann in Revit neu eingebauten Fenster nicht im ursprünglichen IFC-Datenmodell, also der aus Allplan exportierten IFC-Datei, worauf sich ArchiCAD bezieht, enthalten sind. Wie des Öfteren darauf hingewiesen, werden beim Austausch von BCFZIP-Dateien nämlich keine IFC-Informationen beigefügt.

Um dieser Begebenheit Abhilfe zu schaffen, muss Max Mustermann aus der überarbeiteten Version seines Revitmodells eine neue IFC-Datei exportieren und den Projektbeteiligten zur Verfügung stellen.

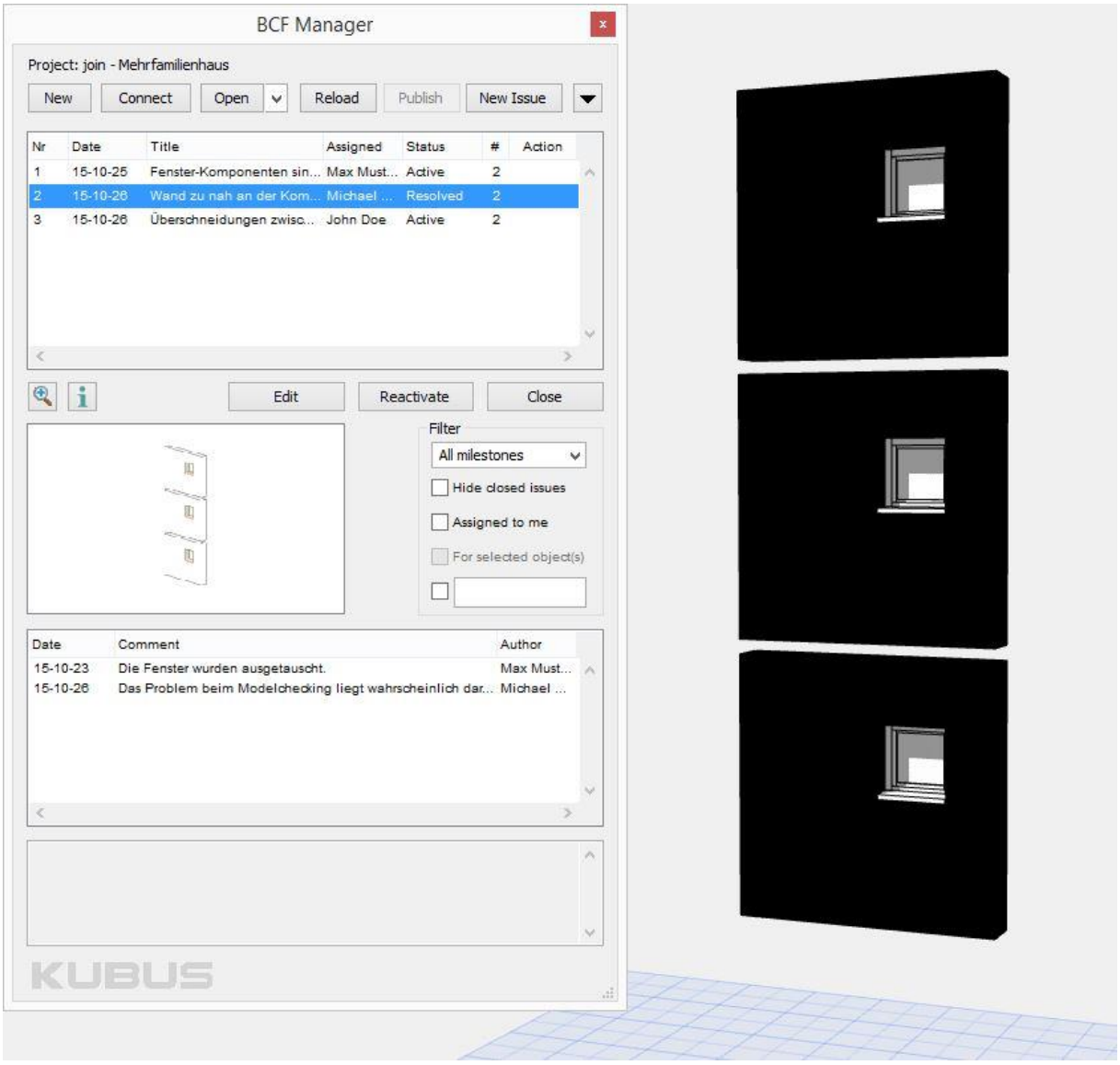


Abbildung 32 Ansicht des im BCF Manager hinterlegten und in Revit gelöstem Problem mit den überdeckten Fenstern

Zwar findet John Doe nach dem Import der neu generierten IFC-Datei die geänderten Fenster über den BCF Manager, worin der Status des betroffenen Issues mit „*Resolved*“ vermerkt ist, problemlos in seinem Modell (vgl. Abbildung 32), doch halten die anfänglichen Fehler bei der Darstellung des Modells, die beim allerersten Import in Revit aufgetaucht sind, nun auch in ArchiCAD Einzug. Das heißt, dass manche Außenwände fälschlicherweise schwarz dargestellt werden, wobei an dieser Stelle aber kein Schema abgeleitet werden kann, denn es betrifft sowohl Mauerwerkswände als auch Betonbauteile. Darüberhinaus ist die Durchnummerierung aller Bauteile (z.B. < Basiswand 383 > vgl. Abschnitt 6.3) übernommen worden und die Fenstertüren auf den Loggien überschneiden sich wie in Revit mit den anliegenden Fenstern. Was aber noch viel auffälliger ist, ist die Tatsache, dass sich an den Stirnseiten des Gebäudes aus unerklärlichen Gründen ein Teil einer Wand aus der Dachebene erhoben hat (vgl. Abbildung 33).

Abbildung 33 Fehlerhafte Darstellung des Modells in ArchiCAD nach Import der aus Revit neu exportierten IFC-Datei

5. Abschließende Kontrolle im Solibri Model Checker

Zum Schluss ist die neu generierte IFC-Datei ein weiteres Mal mit dem SMC überprüft worden mit der Erkenntnis, dass der bearbeitete Issue („Wand zu nah an der Komponente Fenster“) behoben worden ist, da ein ähnlicher Fehler bei der Kollisionserkennung nicht weiter auftritt. Doch abschließend sind die Überlappungen der Fensterüren mit den angrenzenden Fenstern auf den Loggien in einer lokal abgespeicherten BCFZIP-Datei festgehalten worden (vgl. Abbildung 34). Das mögliche Hochladen auf BIMcollab wäre wenig zielführend, denn die Basis dieses neu erzeugten Issues stellt im Vergleich zu den eingangs generierten eine neue IFC-Datei mit veränderter Datenstruktur, was die Mängel- und Problemverwaltung auf BIMcollab unübersichtlicher gestalten würde.

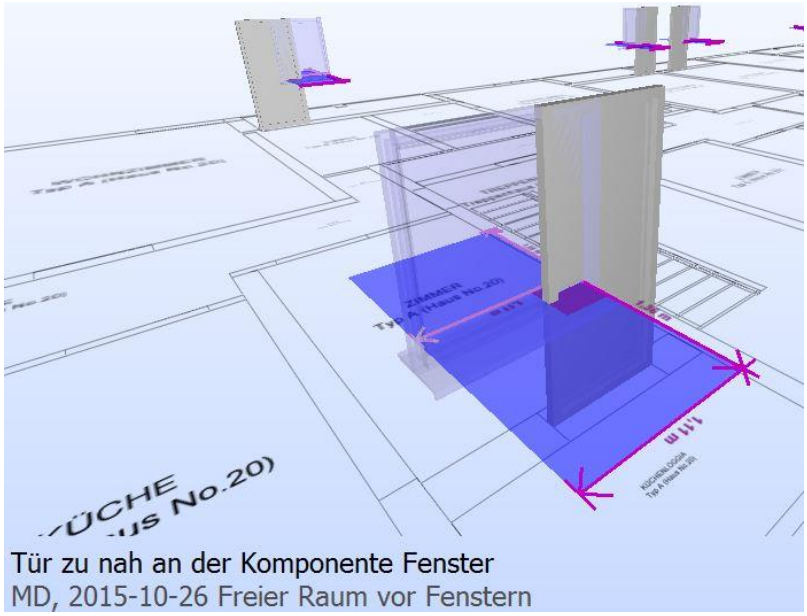


Abbildung 34 Überschneidung der Fenstertüren mit den benachbarten Fenstern – neuer Issue im Solibri Model Checker

7 Fazit

Als Erstes ist anzuführen, dass das IFC-Format zwar ein mächtiges Tool darstellt, um Gebäudemodelle in einem umfassend komplexen Datenmodell in geometrischer und semantischer Weise zu beschreiben und zu übergeben, aber der Import- bzw. Export-Vorgang läuft vereinzelt noch nicht reibungslos ab und führt z. T. zu fehlerhaftem Datenaustausch oder zum Datenverlust (vgl. Beispielszenario). Dies kann darauf zurückgeführt werden, dass solche Schnittstellen seitens der Softwarehersteller in ihren Produkten teilweise noch unsachgemäß implementiert sind, was wiederum der Komplexität der IFC-Datenstruktur geschuldet ist. (Borrmann, Building Information Modeling, 2015) Um diese Strukturen herunterzubrechen und gezieltere Datenaustauschszenarien durchführen zu können, werden demnach höhere Anforderungen an Model View Definitions gesetzt werden.

Wie sich gezeigt hat, entspricht das IFC-Datenmodell einem starren Gefüge, in dem einzelne Änderungen im Modellierungsprozess nur schwer erkenntlich hervorgehen. Dagegen gleicht das BCF-Format in Analogie zur konventionellen Arbeitsweise mit 2D-Plänen einer Revisionswolke, die Änderungen auf zu überprüfenden Zeichnungen markieren. Demzufolge hat sich das Arbeiten mit BCF-Dateien über den IFC Workflow hinaus sehr viel flexibler gestalten lassen, um Änderungen am Modell zu kommunizieren ohne dabei die zugehörige IFC-Datei beizulegen. Somit können Nachrichten an Bauteilen über eine spezielle ID angefügt werden.

Dabei stellt der BCF Manager von KUBUS eine Verknüpfung von BCF-Dateien und zugrunde liegendem IFC-Datenmodell her, indem mit Hilfe des Software-Plug-Ins Markierungen und Annotationen zu Objekten direkt in Modellierungsprogramme eingelesen bzw. auch dort neue Anmerkungen getroffen werden können. Unabhängig davon welche Software (Revit oder ArchiCAD) verwendet wird, ermöglicht der BCF Manager somit eine herstellerunabhängige Kommunikationsgrundlage mit BIMcollab im Hintergrund, das zur zentralen Verwaltung aller eingespeisten Mängel dient. Demnach ist BIMcollab mit einer dokumentierenden „To-do-Liste“ zu vergleichen, worin aber entsprechend der BCF-Datenstruktur wiederum keine IFC-Datenmodelle gespeichert sind.

Die strikte Trennung zwischen BCF-Dateien und den kompletten IFC-Datenmodellen anhand der vom Modell ausgelagerten Kommunikation bringt aber parallel dazu einen wesentlichen Nachteil mit sich: Werden im Zuge des ausgeübten Problem- und Mängelmanagement Änderungen am Modell vorgenommen, so struktiert sich auch gleichzeitig das IFC-Datenmodell im Hintergrund um. Folgerichtig finden die zuvor generierten BCF-Dateien, die über deren GUID an das ursprüngliche Modell gekoppelt sind, keinen Bezug mehr zum neuen Modell, wenn denn nicht eine IFC-Datei vom neu erzeugten Gebäudemodell in die jeweilige Software importiert werden würde.

Dieses gravierende Problem stellt letztendlich bezüglich des Änderungsmanagement einen Widerspruch in sich dar, denn diejenige BCF-Datei, die Änderungen kommuniziert und deren Bearbeitung induziert mit dem Vorteil, die enormen Datenmengen des IFC-Datenmodells

außen vor zu lassen, müsste in ihrer logischen Konsequenz bei jeder Modifikation des Modells mit einer neu generierten IFC-Datei verknüpft werden.

Aus dieser Sicht eignet sich der BCF Manager nicht für die frühe Planungsphase, in der zahlreiche Änderungen des Gebäudemodells auftreten, sondern viel mehr für spätere Phasen, wenn eine einheitliche IFC-Datei vom Modell abgeileitet werden kann, an der sich lediglich kleine Änderungen in der Bauausführung oder im Objektbetrieb ergeben.

Wenn man den im Beispielszenario beschriebenen IFC bzw. BCF Workflow nicht optimieren würde und dementsprechend zu jedem schwerwiegenden Modellierungsfehler eine neue IFC-Datei erzeugt, um sie anschließend allen Projektbeteiligten zukommen zu lassen, dann würde man im laufenden Projektrealisierungsprozess nicht sicherstellen können, dass alle Disziplinen gemeinsam an der aktuellsten IFC-Datei arbeiten.

Um Verbesserungen hinsichtlich des BCF Workflows zu erzielen, würde im ersten Schritt nahe liegen, einer BCF-Datei einen kleinen Ausschnitt des am urspünglichen Modell geänderten IFC-Datenstruktur beizufügen. Dahingehend müssten zwei IFC-Daten miteinander verglichen werden, um so eine Herangehensweise wiederum aus technischer Sicht zu implementieren, damit man den „Unterschied“ zwischen beiden Datenstrukturen erkennen könnte. Dies hat die logische Konsequenz zur Folge, dass sehr wohl eine komplette IFC-Datei an das BCF-Format angehängt werden müsste.

7.1 Ausblick

Einen projektbezogenen Lösungsansatz zu dem beschriebenen Problem stellt das *Common Data Environment (CDE)* dar (PAS 1192-2:2013, 2013). Hinter dieser Methode steht die Philosophie alle in einem Projekt anfallenden Datensätze und Informationen – seien es BIM-basierte Modelldateien oder konventionelle Textdateien o.ä. – zur besseren Kollaboration aus einer gemeinsamen internetbasierten Projektplattform zu beziehen bzw. hochzuladen.

Diesbezüglich bietet z.B. die Firma Nemetschek den offenen Datenserver „bim+“ an. Im Zentrum steht dabei ein internetgestützter 3D-Viewer zur Visualisierung der über das neutrale IFC-Format hochgeladenen BIM-Modelle. Darüber hinaus besteht die Möglichkeit, ähnlich wie beim BCF Manager, direkt am Modell Bauteile über einen *Spot* zu markieren und diesen innerhalb eines sog. *Topics* Informationen anzuhängen oder diese mit Kommentare zu versehen, um sie wiederum an Projektbeteiligten zu adressieren. Dadurch, dass die Plattform das BCF-Format unterstützt, können solche Topics als BCF-Datei exportiert oder aber Issues aus einem externen Model Checker importiert werden.

Der große Unterschied einer solchen Plattform gegenüber BIMcollab ist, dass das zugrunde liegende Gebäudemodell mit den zugehörigen Anmerkungen (in Form von BCF-Dateien) auf einer zentralen Webseite ineinandergreifen.

Des Weiteren ermöglicht bim+ bestehende Modelle anhand ihrer IFC-Datei zu reversionieren. Das heißt, zwei korrespondierende IFC-Dateien werden zuvor abgeglichen, um dann letztenlich nur den Unterschied in den beiden IFC-Datenstrukturen auf Basis der ursprünglichen Datei zu ändern. Dieses Vorgehen bschreibt eine mögliche Lösung des im Beispielszenario

beschriebenem Problem, weil dahingehend sämtlich generierte Topics (vgl. Issues) immer mit der aktuellsten IFC-Datei gekoppelt bleiben.

Anhang

Compact Disc

Auf der beigefügten Compact Disc sind folgende Datenenthalten:

- Der schriftliche Teil der Arbeit als Worddokument
- Das modellierte Gebäude in Allplan
- Die IFC-Daten des Gebäudemodells
- Die BCFZIP-Dateien der erzeugten Issues
- Ein Film mit einer Kamerfahrt um das gerenderte Modell herum

Abkürzungsverzeichnis

BIM	Building Information Modeling
BCF	BIM Collaboration Format
BPMN	Business Process Modeling and Notation
CDE	Common Data Environment
GUID	Globally Unique Identifier
HOAI	Honorarordnung für Ingenieure und Architekten
IAI	International Alliance for Interoperability
IFC	Industry Foundation Classes
ISO	Internationale Organisation für Normung
IDM	Information Delivery Manual
MVD	Model View Definition
PAS	Publicly Available Specification
PNG	Portable Network Graphics
SMC	Solibri Model Checker
STEP	Standard for the Exchange of Product model data
UUID	Universally Unique Identifier
XML	Extensible Markup Language

Literaturverzeichnis

BIMwelt. (2015). *BIM[welt]*. Retrieved Oktober 20, 2015, from http://www.bimwelt.de/bim/bimkonzepte/open-bim/

BMVI. (2015, Juni Bundesministerium für Verkehr und digitale Infrastruktur). *Reformkommission Bau von Großprojekten*. Retrieved November 5, 2015, from https://www.bmvi.de/SharedDocs/DE/Artikel/G/reformkommission-bau-von-grossprojekten.html

Borrmann, A. (2015, November 3). *Objektorientierte Modellierung Datenaustausch mittels IFC*. TU München: Vorlesung, Bau- und Umweltinformatik Ergänzungsmodul.

Borrmann, A. (2015). *Building Information Modeling.* (A. Borrmann, M. König, C. Koch, & J. Beetz, Eds.) Wiesbaden: Springer.

buildingSmart. (2015, August). *mvdXML*. Retrieved Oktober 26, 2015, from http://www.buildingsmart-tech.org/downloads/mvdxml/mvdxml-1.1/release-candidate/mvdxml-1-1rc-documentation

buildingSmart. (2015). *Technical Vision*. Retrieved Oktober 20, 2015, from http://www.buildingsmart.org/standards/technical-vision/

Dobrindt, A. (2015). *Reformkommission Bau von Großprojekten.* Berlin: Bundesministerium für Verkehr und digitale Infrastruktur.

Egger, M., Hausknecht, K., Liebich, T., & Przybylo, J. (2013, November 30). *BIM-Leitfaden für Deutschland.* Bundesinstitut für Bau-, Stadt- und Raumentwicklung.

Günthner, W. A., & Borrmann, A. (2011). *Digitale Baustelle - innovativer Planen, effizienter Ausführen.* Berlin: Springer.

ISO 10303-21:2002-01. (2002). *Industrial automation systems and integration -- Product data representation and exchange -- Part 21:Implementation methods: Clear text encoding of the exchange structure*.

ISO 29481-1:2010-05. (2010). *Building information modelling - Information delivery manual - Part 1: Methodology and format*.

KUBUS. (2015). *Open BIM*. Retrieved Oktober 20, 2015, from http://www.bimcollab.com/en/BIM/OpenBIM

Liebchen, J. H. (2014, Juni). *Building Information Modeling (BIM) - Eine Herausforderung an die zeitgemäße Ausbildung von im Bauwesen tätigen Ingenieurinnen und Ingenieuren.*

PAS 1192-2:2013. (2013). *Specification for information management for the capital/delivery phase of construction projects using building information modelling*.

Solibri. (2013, Oktober). *KUBUS launches BCF Manager for both Revit and ArchiCAD*. Retrieved Oktober 30, 2015, from http://www.solibri.com/journal/kubus-launches-bcf-manager-%E2%80%A8for-revit-archicad/

Stangeland, B. K. (2012, März). *Open BIM Collaboration Format*. Oslo: buildinSmart conference.

Abbildungs- und Tabellenverzeichnis

Abbildung 1 Der gesamte Lebenszyklus eines Building Information Model 8
Abbildung 2 IFC Workflow nach dem Open BIM-Schema 9
Abbildung 3 Vorverlagerung von Planungs- und Entscheidungsprozessen gegenüber der konventionellen Arbeitsweise 10
Abbildung 4 Darstellung des Datenmodells und der Instanzdaten nach dem IFC 13
Abbildung 5 Beispielhafte Definition eines Entitytyps 14
Abbildung 6 Schichtenaufbau des IFC-Datenmodells 15
Abbildung 7 IfcRoot als Ursprung der Entitys und Ausschnitt der bedeutendsten Entitytypen (von insgesamt 653 Entitys) in der Vererbungshierarchie 16
Abbildung 8 Ansicht einer Tür im Solibri Model Checker 17
Abbildung 9 Darstellung der ausgewälten Tür (aus Abbildung 8) im Editor 17
Abbildung 10 Übersicht aller Teilvorgänge vom IDM zum MVD 19
Abbildung 11 BPMN-Diagramm bezüglich des Beispielszenarios in Abschnitt 6 - erstellt mit Microsoft Visio 20
Abbildung 12 Ein in Solibri erkannter Issue: Die Außenputzschicht überdeckt das dahinter liegende Fenster 23
Abbildung 13 Die Markup-XML-Datei des aus Abbildung 11 hervorgegangenem Mangel 23
Abbildung 14 Auszug aus der Viewpoint-XML-Datei des aus Abbildung 11 hervorgegangem Mangel 24
Abbildung 15 Die Plug-In-Schnittstelle des BCF Managers in Autodesk Revit (links) und ArchiCAD (rechts) 25
Abbildung 16 Die Benutzeroberfläche des BCF Managers in Revit oder auch in ArchiCAD mit drei erzeugten Issues aus dem Beispielszenario in Abschnitt 6 26
Abbildung 17 Benutzoberfläche innerhalb des BCF Manager zum Erzeugen eines neuen Issues mit den grundlegendsten Einstellungsmöglichkeiten & Informationen gemäß der BCF-Datenstruktur 26
Abbildung 18 Überblick über die durch BIMcollab vernetzten Softwareprodukte 27
Abbildung 19 Herstellen einer Verbindung zu BIMcollab über den BCF Manager 27
Abbildung 20 Informationen über das BIMcollab-Projekt „Mehrfamilienhaus“ aus dem Beispielszenario in Abschnitt 6 28
Abbildung 21 Übersicht über die Darstellung einzelner Issues in BIMcollab 29
Abbildung 22 Gerenderte Ansicht des kompletten Gebäudemodells 32
Abbildung 23 Rendering eines Treppenhauses im hinblick auf verschiedenen Wandaufbauten und Ausbauobjekte 33
Abbildung 24 Auszug aus einem in Allplan erzeugtem Raumbuch-Report 33
Abbildung 25 Von der jeweiligen Wand überdeckte Fenster in ArchiCAD 34
Abbildung 26 Auszug der Eigenschaften einer in Revit dargestellten Wand 35
Abbildung 27 Fehlerhafte Darstellung in der Gebäudegeometrie und Probleme bei der Visualisierung von Betonbauteilen 36
Abbildung 28 Anweisung über Änderungen am Rohbau kommuniziert mittels BIMcollab 37
Abbildung 29 Bearbeitung des Modells anhand der in der BCF-Datei gespeicherten Ansicht 38
Abbildung 30 Darstellung des in Revit aktualisierten Issue auf der Webseite von BIMcollab 39
Abbildung 31 Fehlermeldung beim Öffnen der BCFZIP-Datei in ArchiCAD über den BCF Manager 39
Abbildung 32 Ansicht des im BCF Manager hinterlegten und in Revit gelöstem Problem mit den überdeckten Fenstern 40
Abbildung 33 Fehlerhafte Darstellung des Modells in ArchiCAD nach Import der aus Revit neu exportierten IFC-Datei 41
Abbildung 34 Überschneidung der Fenstertüren mit den benachbarten Fenstern – neuer Issue im Solibri Model Checker 42
Tabelle 1 Überblick über die verwendenten Softwareprodukte im Beispielszenario 31
Tabelle 2 Überblick über die am Beispielprojekt beteiligten Parteien 36