

Technische Universität München

Ingenieur fakultät Bau Geo Umwelt

Lehrstuhl für Computergestützte Modellierung und Simulation

Möglichkeiten der modellbasierten Konformitätsprüfung von Brückenentwürfen des Straßenbaus mit IFC4x3

Masterarbeit

für den Master of Science Studiengang Bauingenieurwesen

Autor:	Sandra Kraus
Matrikelnummer:	██████████
1. Betreuer:	M.Sc. Sebastian Esser
2. Betreuer:	Dipl.-Ing. Ina Heise
Ausgabedatum:	01. Dezember 2022
Abgabedatum:	31. Mai 2023

Vorwort

Zu Beginn möchte ich mich herzlich bei meinen Betreuern der TU München, Ina Heise und Sebastian Esser, für ihre Ratschläge und Verbesserungsvorschläge bedanken.

Ebenfalls möchte ich mich bei meiner Betreuerin Paulina Sala von der Zilch + Müller Ingenieure GmbH bedanken, die mir bei Fragen zu Theorie und Modellierung stets sehr viel weitergeholfen hat.

Ich möchte mich auch bei meiner Familie, meinen Freunden und meinen Kollegen für ihre Geduld und die vielen aufmunternden Worte bedanken.

Abstract

The advancing digitalisation of the construction industry results in an increased use of Building Information Modeling (BIM) in construction projects including bridge construction projects. In public-sector projects, which are the norm in bridge construction, data must be exchanged in a vendor-neutral format. For BIM models the IFC format from buildingSMART international (bSI), which is standardized in ISO 16739-1, is usually used. The IFC data model is being extended with the current version 4x3 for infrastructure construction. Among other things, it is hoped that this will now provide a balanced data basis for BIM projects in bridge construction. The use of semantically rich BIM models enables a change in the review of bridge designs. Where up to now 2D plan documents have been checked manually in a time-consuming manner, there is now great potential for automated code compliance checking against the directives using models. Automated code compliance checking is divided into four steps: the translation of the regulations, the preparation of the building model, the execution of the checking process and the report of the results. In the present work, mainly the first two steps are investigated in relation to the guidelines for bridge designs of road construction. Model requirements and rules are developed from the guidelines. Thus it can be estimated whether the model requirements can be fulfilled with the data model of IFC4x3 and with which effort the guidelines of bridge construction can be translated into computer-interpretable rules. The investigation of the data model and the comparison of IFC4x3 with the model requirements show a positive result. However, the compliance with the model requirements strongly depends on the implementation of the IFC4x3 data model in the respective BIM authoring or model checking software. When deriving rules from the guidelines, it can be seen that a very large part of the guidelines can be converted into simple rules. A small part of the guidelines has to be expressed in complex rules that contain a lot of expert knowledge. Overall, there is a great potential that bridge designs of road construction can be checked for compliance with the guidelines in a model-based way in the future.

Zusammenfassung

Die fortschreitende Digitalisierung der Baubranche resultiert in einem vermehrten Einsatz von Building Information Modeling (BIM) auch im Brückenbau. Bei Projekten öffentlicher Auftraggeber, welche im Brückenbau den Regelfall darstellen, muss der Datenaustausch in einem herstellernerneutralen Format erfolgen. Für BIM-Modelle wird dafür üblicherweise das in ISO 16739-1 standardisierte IFC-Format von buildingSMART international (bSI) verwendet. Das IFC-Datenmodell wird mit der aktuellen Version 4x3 für den Infrastrukturbau erweitert. Daraus erhofft man sich unter anderem, dass nun eine ausgewogene Datengrundlage für BIM-Projekte im Brückenbau zur Verfügung gestellt werden kann. Die Verwendung von informationsreichen BIM-Modellen ermöglicht dabei einen Wandel bei der Prüfung von Brückenentwürfen. Wo bisher 2D-Planunterlagen aufwändig manuell geprüft werden, ergibt sich heute ein großes Potential für automatisierte Konformitätsprüfungen gegenüber den Richtlinien anhand von Modellen. Eine Konformitätsprüfung gliedert sich in vier Schritte: die Übersetzung der Regelwerke, die Vorbereitung des Gebäudemodells, die Durchführung des Überprüfungsprozesses und der Report der Ergebnisse. In der vorliegenden Arbeit werden vorwiegend die beiden ersten Schritte in Bezug auf die Richtlinien für Brückenentwürfe des Straßenbaus untersucht. Dabei werden aus den Richtlinien Modellanforderungen und Regeln erarbeitet. Damit kann abgeschätzt werden, ob die Modellanforderungen mit dem Datenmodell von IFC4x3 erfüllt und mit welchem Aufwand die Richtlinien des Brückenbaus in computer-interpretierbare Regeln übersetzt werden können. Die Untersuchung des Datenmodells und der Vergleich von IFC4x3 mit den Modellanforderungen zeigen ein positives Ergebnis. Die Einhaltung der Modellanforderungen hängt jedoch stark von der Implementierung des IFC4x3-Datenmodells in der jeweiligen BIM-Authoring- oder Model-Checking-Software ab. Bei der Ableitung von Regeln aus den Richtlinien zeigt sich, dass ein sehr großer Teil der Richtlinien in einfache Regeln überführt werden kann. Ein kleiner Teil der Richtlinien muss in komplexen Regeln, die viel Fachwissen enthalten, ausgedrückt werden. Insgesamt zeigt sich ein großes Potential, dass Brückenentwürfe des Straßenbaus zukünftig modellbasiert auf die Einhaltung der Richtlinien geprüft werden können.

Inhaltsverzeichnis

Abbildungsverzeichnis	VIII	
Tabellenverzeichnis	X	
Abkürzungsverzeichnis	XI	
1	Einführung und Problemstellung	1
1.1	Einführung	1
1.2	Problemstellung	2
1.3	Aufbau der Arbeit	2
2	Technische Grundlagen	3
2.1	Grundbegriffe der BIM-Methodik	3
2.2	Modellierung von Straßenbrücken	6
2.2.1	Trassierung von Straßen	6
2.2.2	Modellierung von Brücken	7
2.2.3	Unterschiede zum Hochbau	9
2.3	Industry Foundation Classes (IFC)	10
2.3.1	buildingSMART International (bSI)	10
2.3.2	Einführung IFC	11
2.3.3	Aufbau in Schichten	13
2.3.4	Vererbungshierarchie	16
2.3.5	Objektbeziehungen	18
2.3.6	Geometrierepräsentationen	20
2.3.7	Platzierung von Objekten	22
2.3.8	Relevante Neuerungen der Version IFC4x3	23
2.4	Zusammenfassung	29
3	Literaturrecherche zu Automated Code Compliance Checking	30
3.1	Entwicklung von ACCC	30
3.2	Bestandteile von ACCC	31
3.3	Übersetzung von Regelwissen	32
3.3.1	Herausforderungen	32

3.3.2	Beispiele und Forschungsansätze	33
3.4	Aufbereitung von BIM-Modellen.....	38
3.4.1	Inhaltliche Herausforderungen und Lösungsvorschläge	38
3.4.2	Herausforderungen durch IFC	39
3.5	Durchführung der Überprüfung	40
3.5.1	Allgemeines	40
3.5.2	Beispiele für eine technische Umsetzung	41
3.6	Erkenntnisse für die Umsetzung von ACCC im Brückenbau	44
3.7	Zusammenfassung	45
4	Informationsanforderungen für die Konformitätsprüfung	47
4.1	Begriffserklärung Informationsanforderungen	47
4.2	Stand der Technik IRs	48
4.3	Richtlinien für Brückenentwürfe im Straßenbau	53
4.4	Entwicklung von Austauschforderungen für Modelle aus den Richtlinien	54
4.4.1	Festlegung der Detaillierung der Modelle	54
4.4.2	Allgemeines Vorgehen.....	55
4.4.3	Darstellung des Vorgehens an konkreten Beispielen.....	57
4.4.4	Ergebnisse	59
4.5	Vergleich der Modellaustauschanforderungen mit IFC4x3	61
4.6	Zusammenfassung	63
5	Aufbereitung der Richtlinien zu computer-interpretierbaren Regelsätzen	65
5.1	Klassifizierung der Regeln	65
5.2	Auswertung der Analyse der Richtlinien	66
5.3	Auswahl exemplarischer Regeln für die Implementierung	69
5.3.1	Class 1	69
5.3.2	Class 2.....	70
5.3.3	Class 3.....	71
5.3.4	Class 4.....	72
5.4	Zusammenfassung	74
6	Implementierung einzelner Regelsätze	75
6.1	Einblick in die verwendete Software	75
6.2	Erläuterung des Beispielmodells.....	78

6.3	Implementierung	82
6.3.1	Class 1	82
6.3.2	Class 2	83
6.3.3	Class 3	85
6.3.4	Class 4	87
6.4	Diskussion der Ergebnisse der Implementierung	90
6.5	Zusammenfassung	91
7	Zusammenfassung und Ausblick	93
7.1	Zusammenfassung	93
7.2	Ausblick	94
	Literaturverzeichnis	96
	Anhang A	107
	Anhang B	112
	Anhang C	113

Abbildungsverzeichnis

Abbildung 1: Zeitstrahl der IFC-Versionen (Quelle: Laakso und Kiviniemi 2012)	12
Abbildung 2: Übersicht über das Schema von IFC 4.3.x (buildingSMART International 2022a)	14
Abbildung 3: Beispiel Vererbungshierarchie anhand IfcSlab (Quelle: Eigene Darstellung anhand von buildingSMART International 2022a).....	17
Abbildung 4: Aussparungsbeziehung (Quelle: buildingSMART International 2022a - Figure 5.4.3.62.A).....	19
Abbildung 5: Aufteilung der Alignment-Entities in semantische Aspekte (links) und geometrische Repräsentation (rechts) (Quelle: Pinzenöhler 2022)	25
Abbildung 6: Hierarchie der räumlichen Unterteilung von Gebäuden (grün) und anderen Bauwerken (blau) in IFC (Quelle: Eigene Darstellung, angelehnt an Borrmann et al. 2021a)	26
Abbildung 7: Struktur und Bestandteile einer automatisierten Konformitätsprüfung (Quelle: Eigene Darstellung, angelehnt an Eastman et al. 2009)	31
Abbildung 8: Zusammenhang zwischen verschiedenen Informationsanforderungen und -modellen (Quelle: Eigene Darstellung, angelehnt an DIN EN ISO 19650-1 Bild 2).....	47
Abbildung 9: Zuordnung von Anforderungen aus der RAB-Ing zu eigenen Kategorien von Modellinformationen (Quelle: Eigene Darstellung)	57
Abbildung 10: Zuordnung von Anforderungen an Kappenanker aus der RE-Ing zu eigenen Kategorien von Modellinformationen (Quelle: Eigene Darstellung).....	58
Abbildung 11: Auszug aus der Mapping-Datei für den IFC4x3-Export aus Civil 3D (Quelle: Eigene Darstellung)	76
Abbildung 12: Beispielmodell "Hohlkastenbrücke" in Desite md pro (Quelle: Eigene Darstellung).....	78
Abbildung 13: 3D-Darstellung des Überbauquerschnitts in Autodesk Revit 2022 (Quelle: Eigene Darstellung)	79
Abbildung 14: Darstellung der in X/Y-Ebene projizierten Hohlkasten-Bodenplatte (unten) im Vergleich zum Beispielmodell in Desite md pro (Quelle: Eigene Darstellung).....	80

Abbildung 15: Hilfskörpermodell in Desite md pro (Quelle: Eigene Darstellung)	81
Abbildung 16: Desite-Skript für Beispielregel Class 1 (Quelle: Eigene Darstellung) .	83
Abbildung 17: Auszug aus dem Desite-Skript der Beispielregel für Class 2 (Quelle: Eigene Darstellung).....	84
Abbildung 18: Auszug aus dem Desite-Skript zur Beispielregel in Class 3 - Schritt 1 (Quelle: Eigenen Darstellung)	86
Abbildung 19: Ergebnis des Desite-Skripts der Class 3-Regel für das Beispielmodell (Quelle: Eigene Darstellung)	87
Abbildung 20: Auszug aus dem Desite-Skript der Class 4-Regel – Schritt 5 (Quelle: Eigene Darstellung).....	89
Abbildung 21: Ergebnis des Desite-Skripts der Class 4-Regel für das Beispielmodell (Quelle: Eigene Darstellung)	89

Tabellenverzeichnis

Tabelle 1: Zusammenstellung der einzelnen Modellierungsschritte einer Brücke nach Nöldgen 2021 mit eigenen Anmerkungen (durch Pfeile gekennzeichnet).....	7
Tabelle 2: Übersicht über die Unterschiede der Modellierung von Hoch- und Brückenbau	9
Tabelle 3: Zuordnung von erarbeiteten Anforderungen aus der jeweiligen Richtlinie zu einer Kategorie für benötigte Modellinformationen	59
Tabelle 4: Zuordnung von erarbeiteten Anforderungen aus der jeweiligen Richtlinie zu den Klassifizierungen von Regeln nach Solihin und Eastman 2015	67

Abkürzungsverzeichnis

ACCC	Automated Code Compliance Checking
AIA	Auftraggeberinformationsanforderungen
AIM	Asset-Informationsmodell
AIR	Asset-Informationsanforderung
API	Application Programming Interface
BAP	BIM Abwicklungsplan
BASt	Bundesanstalt für Straßenwesen
BCF	BIM Collaboration Format
BIM	Building Information Modeling
BMDV	Bundesministerium für Digitales und Verkehr
bSD	buildingSMART Deutschland
bSDD	buildingSMART Data Dictionary
bSI	buildingSMART International
DD	Data Dictionary
DOC	Textbasierte Methoden zur Erstellung von IRs
EIR	Informationsaustauschanforderungen
engl.	Englische Übersetzung des Begriffs
GUID	Global Unique Identifier
IDM	Information Delivery Manual

IDS	Information Delivery Specification
IFC	Industry Foundation Classes
IR	Modellanforderung (engl. <i>Information Requirement</i>)
LD	Linked Data
LoD	Level of Detail
LOD	Level of Development
LOIN	Level of Information Need
MVD	Model View Definition
NLP	Natural Language Processing
OIR	Organisatorische Informationsanforderungen
OWL	Ontologies Web Language
PDT	Product Data Template
PIM	Projekt-Informationsmodell
PIR	Projekt-Informationsanforderungen
RASE	Markup Sprache (Requirement, Applicability, Selection, Exception)
RDF	Resource Description Framework
SHACL	Shapes Constraint Language
SMC	Solibri Model Checker
XLS	Tabellenbasierte Methoden zur Erstellung von IRs

1 Einführung und Problemstellung

1.1 Einführung

Im Rahmen von Brückenbauprojekten des Bundes für den Straßenbau ist für den Entwurf der Brücken eine Reihe von verschiedenen Richtlinien zu beachten, z.B. die RAB-Ing, die RE-Ing und die ZTV-Ing. Die Einhaltung dieser Richtlinien wird bisher im Regelfall anhand von 2D-Zeichnungen geprüft. Eine derartige manuelle Prüfung ist fehleranfällig und zeitaufwändig (Preidel et al. 2021).

Dazu kommt, dass heutzutage in sehr vielen Ingenieurbüros auch im Ingenieurbau mit Hilfe von dreidimensionalen, semantisch angereicherten BIM-Modellen geplant wird, aus welchen dann eine Planableitung für die Entwurfsunterlagen erfolgt. Gerade bei gekrümmten oder schiefen Bauwerken, welche im Brückenbau den Regelfall darstellen, ist eine RAB-konforme Planableitung auch mittels moderner Software häufig schwierig oder nur mit großem Aufwand umzusetzen.

Eine (teil-)automatisierte Modellprüfung hinsichtlich der Konformität mit den Richtlinien, auch Automated Code Compliance Checking (ACCC) genannt, würde dabei helfen, viele Planungsfehler bereits in der Entwurfsphase zu entdecken und die Modell- und Planungsqualität zu erhöhen (Preidel et al. 2021). Mit einer digitalen Prüfung am Modell könnte die aufwändige Planableitung in der Entwurfsphase entfallen.

Zur Unterstützung der BIM-Einführung im öffentlichen Straßen- und Brückenbau wurde der BIM-Leitfaden 2.0 erstellt. In diesem werden eine Effizienzsteigerung, eine gemeinschaftliche Projektorientierung, die Digitalisierung von Planungsprozessen und eine verbesserte Planungssicherheit als BIM-Ziele festgelegt. Im Rahmen der Digitalisierung der Planungsprozesse sollen auch Genehmigungsprozesse digitalisiert und automatisiert werden, was eine zukünftige Anwendung von ACCC für die Qualitätssicherung in diesem Bereich nahelegt. (Bayerisches Staatsministerium für Wohnen, Bau und Verkehr 2022)

1.2 Problemstellung

In dieser Masterarbeit wird untersucht, inwiefern eine Konformitätsprüfung bezüglich der Richtlinien für den Entwurf von Straßenbrücken mit den Modelldaten von IFC4x3-Modellen ermöglicht werden kann und welche Austauschforderungen dafür von den Modellen erfüllt werden müssen.

Darüber hinaus werden aus den Richtlinien beispielhaft Regeln für eine solche Konformitätsprüfung erstellt und hinsichtlich der Komplexität ihrer Implementierung in Model-Checking-Software beurteilt. Dies wird abschließend durch eine exemplarische Implementierung validiert.

1.3 Aufbau der Arbeit

In Kapitel 2 werden zunächst die technischen Grundlagen erläutert, die für das Verständnis der Arbeit notwendig sind. Dabei wird ein Überblick über die BIM-Methodik und Modellierung von Straßenbrücken im Vergleich zu Hochbauten gegeben und das IFC-Datenschema inklusive der Neuerungen in IFC4x3 genau erläutert.

Anschließend wird in Kapitel 3 ein Überblick über den Stand der Forschung zum Thema automatisierte Konformitätsprüfung (engl. *Automated Code Compliance Checking*, kurz: ACCC) gegeben. Es werden die Bestandteile einer Konformitätsprüfung erläutert und jeweils Beispiele für Forschungsansätze beziehungsweise technische Umsetzungen der einzelnen Bestandteile analysiert. Aus diesen Erkenntnissen werden Rückschlüsse für ACCC im Brückenbau gezogen.

In Kapitel 4 werden die Informationsanforderungen für die Konformitätsprüfung erarbeitet. Zunächst wird ein Überblick über die Bedeutung von Informationsanforderungen und verschiedene Möglichkeiten der Formulierung dieser gegeben. Dann werden die für den Entwurf von Brückenbauwerken im Straßenbau relevanten Richtlinien vorgestellt, aus denen im Anschluss die Informationsanforderungen herausgearbeitet werden. Diese werden abschließend mit dem Datenmodell von IFC4x3 verglichen.

Aus den Richtlinien werden in Kapitel 5 computer-interpretierbare Regeln abgeleitet. Die einzelnen Anforderungen werden zunächst kategorisiert, um eine Aussage zur Implementierbarkeit des gesamten Regelwerks treffen zu können. Dann werden für jede Kategorie Beispielregeln erarbeitet, welche im anschließenden Kapitel 6 zur Validierung der Aussage zur Implementierbarkeit in Software implementiert und auf ein Beispielmodell angewendet werden.

2 Technische Grundlagen

Zu Beginn der Arbeit werden wesentliche Grundlagen, die für das Verständnis der Thematik nötig sind, erläutert. Zunächst werden Grundbegriffe des modellbasierten Arbeitens definiert. Anschließend wird erst auf die Modellierung von Straßenbrücken eingegangen und dann auf die *Industry Foundation Classes*.

2.1 Grundbegriffe der BIM-Methodik

In diesem Kapitel werden einige grundlegende Begriffe zum modellbasierten Arbeiten kurz erklärt.

BIM

In DIN EN ISO 19650-1 wird die *Bauwerksinformationsmodellierung* (engl. Building Information Modeling, kurz: BIM) als eine Methode im Bauwesen definiert, bei welcher digitale Modelle von Bauwerken als zuverlässige Entscheidungsgrundlage zur Unterstützung von Planungs-, Bau- und Betriebsprozessen verwendet werden (DIN 2019).

Die konsequente disziplinübergreifende Nutzung von BIM-Modellen über alle Lebenszyklusphasen des Bauwerks hinweg, bezeichnet man als *BIG BIM*. Wird ein BIM-Modell nur als Insellösung für einen kleinen Teil eines Projekts verwendet, spricht man von *little BIM*.

Bei der Anwendung von *BIG BIM* werden durch alle Fachplaner¹ digitale Modelle erzeugt, in welchen die für das Projekt relevanten Daten in einer computer-interpretierbaren Form vorliegen. Diese Fachmodelle können zur Koordination übereinandergelagt werden, wodurch Änderungen und Unstimmigkeiten besser erkannt werden als mit 2D-Planunterlagen.

Außerdem können die Informationen in den Modellen ohne Verluste durch Datenbrüche und menschliche Fehler bei der Interpretation von 2D-Planungsunterlagen zwischen den Fachdisziplinen und Planungsphasen ausgetauscht werden. So kann eine

¹ Im Rahmen dieser Arbeit werden für eine bessere Lesbarkeit nur männliche Personenbezeichnungen verwendet, welche gleichermaßen für männliche, weibliche und diverse Personen gelten. Auf eine Doppelnennung beziehungsweise gegenderte Bezeichnungen wird verzichtet.

aufwendige manuelle Wiedereingabe von Daten im Laufe des Projekts deutlich reduziert werden z.B. in Simulations- und Berechnungstools sowie Facility-Management-Systemen. Dadurch kann die Produktivität und Qualität der Planungen erhöht werden. (Borrmann et al. 2021b)

Der so entstehende fortlaufende Informationsfluss ohne nennenswerte Datenverluste wird im Rahmen dieser Arbeit als *Durchgängigkeit der Projektinformationen* verstanden.

Der Austausch von Modelldaten erfolgt bei BIM entweder softwareübergreifend mit Hilfe von speziellen Modellaustauschformaten wie IFC (sh. Kap 2.3) oder innerhalb derselben Software mit deren proprietären Dateiformat. Bei einer softwareübergreifenden modellbasierten Zusammenarbeit spricht man von *Open BIM*. Das Gegenteil *Closed BIM* beschreibt die modellbasierte Zusammenarbeit mit nur einer Software. (Borrmann et al. 2021b)

Bei Bauprojekten öffentlicher Auftraggeber, dem Regelfall im Brückenbau, muss grundsätzlich hersteller- und produktneutral geplant werden. Deswegen wird bei solchen Projekten mit *Open BIM* gearbeitet, was die Verwendung eines herstellernerutralen Formats wie IFC voraussetzt. (Bayerisches Staatsministerium für Wohnen, Bau und Verkehr 2022)

AIA & BAP

Für eine gute Zusammenarbeit der Akteure sind vertragliche Vereinbarungen nötig. Bei BIM-Projekten, vor allem wenn *BIG BIM* in Kombination mit *Open BIM* angewendet wird, muss der Inhalt sowie die Übergabe von Modellen vertraglich geregelt werden.

Vom Auftraggeber müssen vorab *Auftraggeberinformationsanforderungen* (AIA) aufgestellt werden, mit denen BIM-Ziele und BIM-Use Cases sowie Anforderungen an Modellinhalte, Level of Development und verwendete Datenformate festgelegt werden.

Der Auftragnehmer erstellt daraufhin einen *BIM-Abwicklungsplan* (BAP), in welchem er darlegt, wie er die AIAs im Projekt genau erfüllt. Da die Akteure zwischen den Planungsphasen häufig wechseln und die Planungsphasen in der Regel getrennt ausgeschrieben werden, werden AIA und BAP meist für die jeweilige Phase, z.B. Ausführungsplanung oder Bauausführung, formuliert. (Borrmann et al. 2021b)

BIM-Use Case

Ein BIM-Use Case wird nach VDI-Richtlinie 2552 Blatt 2 definiert als Durchführung eines spezifischen Prozesses bzw. eines Arbeitsschrittes unter Verwendung der BIM-Methode zur Erreichung der zuvor definierten BIM-Ziele (VDI 2022). Welche Use Cases in einem Projekt umgesetzt werden, wird durch die AIAs festgelegt. So kann die BIM-Methode für jedes Projekt passend und bedarfsorientiert eingesetzt werden. (Bayerisches Staatsministerium für Wohnen, Bau und Verkehr 2022)

Zu den bekanntesten und verbreitetsten Use Cases zählen die Modellkoordination verschiedener Fachgewerke und die Mengenermittlung (Borrmann et al. 2021b). Auch die Konformitätsprüfung (engl. Code Compliance Checking), auf die in Kapitel 3 näher eingegangen wird, ist ein BIM Use Case.

In Deutschland wurde vom Konsortium *BIM Deutschland*, welches vom *Bundesministerium für Digitales und Verkehr* (BMDV) und dem *Bundesministerium für Wohnen, Stadtentwicklung und Bauwesen* (BMWSB) als nationales Zentrum für die Digitalisierung des Bauwesens ins Leben gerufen wurde, eine Liste mit 21 standardisierten BIM-Anwendungsfällen veröffentlicht (BIM Deutschland 2022). Zu den harmonisierten Use Cases gehören unter anderem die Mengenermittlung und die Durchführung der Prüfläufe für den Genehmigungsprozess anhand von BIM-Modellen und daraus abgeleiteten erforderlichen Unterlagen (BIM Deutschland 2023).

International wurde von buildingSMART International (bSI) der *Use Case Management Service* (UCMS) eingerichtet, um die Erfassung und den Austausch von Use Cases zu ermöglichen (buildingSMART International 2023).

Level of Development (LOD)

Unter *Level of Development* (LOD), deutsch: Ausarbeitungsgrad, versteht man die Beschreibung der Informationstiefe bzw. -reife von BIM-Modellen. Welches LOD wann im Projekt von einem Modell eingehalten werden muss, wird durch die AIAs vorgegeben.

Ein LOD setzt sich zusammen aus dem *Level of Geometry* (LOG), welches die geometrische Detaillierung des jeweiligen Modellelements beschreibt, und dem *Level of Information* (LOI), welches die zu hinterlegenden alphanumerischen Informationen angibt. Der LOD nimmt mit fortschreitender Planung beständig zu, da immer mehr Informationen, z.B. Abmessungen oder Baustoffe, endgültig festgelegt werden.

Vor der Verwendung des Begriffs LOD wurde im BIM-Bereich häufig der Begriff *Level of Detail* (LoD) verwendet, welcher lediglich den geometrischen Detaillierungsgrad eines Modells berücksichtigt, ohne den Reifegrad der Planung und die Attribuierung der Modellelemente zu betrachten. Im Vergleich zum LoD kann mit dem LOD auch eine Aussage über die Zuverlässigkeit der Modellinformationen getroffen werden. (Abualdenien et al. 2021)

In Europa gilt in diesem Zusammenhang die EN 17412, welche den *Level of Information Need* (LOIN), deutsch: Informationsbedarfstiefe, beschreibt (DIN 2021a). Der LOIN legt notwendige Modellinformationen zur Erfüllung eines bestimmten Zwecks fest (Abualdenien et al. 2021).

2.2 Modellierung von Straßenbrücken

In diesem Kapitel werden die Besonderheiten bei der Modellierung von Straßenbrücken herausgestellt. Dabei erfolgt zunächst ein kurzer Abriss zur Trassierung von Straßen, dann eine Vorstellung der Modellierung von Brücken und abschließend ein Vergleich mit der Modellierung von Hochbauten.

2.2.1 Trassierung von Straßen

Da eine Brücke an jeder Stelle einer Streckenachse liegen kann, wird zuerst die Trassierung im Straßenbau erläutert.

Bei der Streckenplanung in Deutschland wird zunächst die Lage der Streckenachse im Grundriss festgelegt, bestehend aus Geraden, Kreisen und Übergangsbögen. Entlang dieser horizontalen Achse wird dann ein Höhenprofil entwickelt, welches i.d.R. aus Geraden unterschiedlicher Neigung und Kuppen- bzw. Wannenausrundungen besteht. Die Trassierung sowohl im Lage- als auch im Höhenplan ist dabei vom Straßentyp, z.B. Autobahn, und der Entwurfsklasse sowie der Entwurfsgeschwindigkeit abhängig. Im dritten Schritt wird der Querschnitt der Straße anhand des benötigten lichten Raumes ermittelt. Um die Entwässerung der Straße auch in einem horizontalen Streckenabschnitt zu gewährleisten, muss der Querschnitt eine Querneigung von mindestens 2,5 Prozent aufweisen, welche aus fahrdynamischen Gründen in Kurven zur Kurveninnenseite geneigt sein sollte. Der Übergang zwischen unterschiedlichen Querneigungen (Fahrbahnverwindung) erfolgt durch Drehung des Querschnittes um eine vorgegebene Achse. (Appelt und Bracher 2020)

Die so erzeugte dreidimensionale Streckenachse wird im Rahmen dieser Arbeit im Zusammenhang mit BIM und IFC auch mit ihrer englischen Übersetzung *Alignment* bezeichnet.

Mit dieser Art der Trassierung werden in der Modellierungssoftware 2,5D-Modelle der Straße erzeugt. Die erste Modellierung und sämtliche nachfolgenden Änderungen der Trasse erfolgen dabei getrennt nach Lageplan, Höhenplan und Querschnitt. Aus diesen Angaben kann im Anschluss eine dreidimensionale Darstellung der Straße generiert werden. (Häußler und Borrmann 2020)

2.2.2 Modellierung von Brücken

Die Brückengeometrie hängt im Regelfall von der Streckenachse ab, weshalb die Brückengeometrie häufig gekrümmt und daher räumlich komplex ist. Der Verlauf der Fahrbahnoberfläche der Brücke inklusive der zugehörigen Querneigung(en) wird aus der Streckenplanung übernommen. Die Fahrbahn auf der Brücke wird durch eine gleichmäßig dicke, leichte Schutz- und Verschleißschicht gebildet. Die Querneigung muss somit durch die darunterliegende Brückenkonstruktion ausgebildet werden. Die Alternative, dass die Querneigung durch den Asphaltaufbau ausgebildet wird, ist i.d.R. wegen dem resultierenden höheren Gewicht des Aufbaus nicht möglich. (Nöldgen 2021)

Es ist vorteilhaft, wenn ein Brückenmodell mit Abhängigkeiten zur Streckenachse erstellt wird. Bei kleinen Änderungen der Streckenachse während der Planungsphase können so die Brückenbauteile automatisch an den neuen Verlauf der Achse angepasst werden. (Lukas et al. 2008)

Das grundsätzliche Vorgehen bei der 3D-Modellierung einer Straßenbrücke gliedert Nöldgen 2021 folgendermaßen:

Tabelle 1: Zusammenstellung der einzelnen Modellierungsschritte einer Brücke nach Nöldgen 2021 mit eigenen Anmerkungen (durch Pfeile gekennzeichnet)

Schritt	
1	Import eines digitalen Geländemodells (DGM) z.B. im Format LandXML, bestehend aus Koordinatenpunkten der Vermessung und einer Triangulation dieser, dann Erzeugen eines Volumenkörpers des Geländes daraus → Falls kein DGM vorliegt, muss das Gelände ggf. aus 2D-Punkten konstruiert werden

2	Import oder vereinfachte Modellierung von vorhandener Infrastruktur, z.B. bestehende zu überbrückende Straßen inkl. Lichtraumprofil, Leitungen, Bauwerke und Fundamente
3	<p>Import oder Konstruktion der geplanten Streckenachse auf der Brücke und der zugehörigen Querprofile aus der Verkehrsanlagenplanung und Konstruktion der Fahrbahnoberfläche auf der Brücke daraus</p> <p>→ Somit sind die Lage des Bauwerks und die Randbedingungen aus der Umgebung festgelegt.</p>
4	<p>Erzeugen von vereinfachten Hüllgeometrien für die Baustelleneinrichtung (BE), damit räumliche Abhängigkeiten zwischen der BE und dem Bauwerk für die technische Planung abgebildet werden können</p> <p>→ Abhängig vom jeweiligen Projekt kann dieser Schritt auch entfallen oder erst nach Fertigstellung der Modellierung des Bauwerks erfolgen</p>
5	<p>Modellierung des Brückenüberbaus entlang der Straßenachse passend zur bereits modellierten Fahrbahnoberfläche mit Abhängigkeiten zur Straßenachse zur automatischen Anpassung der Brücke bei Änderungen</p> <p>→ Der Brückenoberbau wird in diesem Schritt mit allen nötigen Bestandteilen, z.B. Kappen, Geländer, Schutzeinrichtungen, Randsteinen und Lichtraumprofilen modelliert.</p> <p>→ Die Modellierung von Bauteilen umfasst in diesem Zusammenhang auch eine semantische Anreicherung des Modells mit Informationen durch Attribute der Objekte, z.B. Materialangaben oder Klassifizierungen</p>
6	Modellierung der Unterbauten (Widerlager, Pfeiler) abhängig von Überbau und Gelände bzw. dem zu überbrückenden Medium (Straße, Fluss, Tal, etc.) und Modellierung der Fundamente

Je nach Use Case und gewünschtem Level of Development (LOD) des Modells können im Anschluss weitere Modellbestandteile wie Böschungstrepfen oder Beleuchtung im Modell ergänzt werden. Des Weiteren können Aushubkörper, Bauabläufe oder

Varianten dargestellt werden. Auch eine Weiterverwendung des Modells für Simulationen oder die statische Berechnung ist denkbar.

Grundsätzlich hängt die Detaillierung des Modells vom gewünschten LOD ab. In der Vorplanung (etwa LOD200) ist beispielsweise eine Modellierung der Schutzeinrichtung als Hüllkörper ausreichend, während für die Ausführungsplanung für die Baustelle (LOD400) genaue Angaben, z.B. zur Befestigung und dem einzubauenden Produkt der Schutzeinrichtung, benötigt werden. (Nöldgen 2021)

2.2.3 Unterschiede zum Hochbau

Die Modellierung von Ingenieurbauten, z.B. Straßen, Brücken, Bahnlinien und Tunneln, unterscheidet sich grundlegend von der Modellierung von Hochbauten. In der nachfolgenden Tabelle werden die Unterschiede zwischen Brücken und in der Praxis häufig auftretenden Standard-Hochbauten zusammengestellt.

Tabelle 2: Übersicht über die Unterschiede der Modellierung von Hoch- und Brückenbau

Hochbau	Brückenbau
Häufig gerade Bauteiloberkanten	i.d.R. geneigte Bauteiloberkanten aufgrund von Entwässerung/Querneigung
Häufig einfache Bauteilgeometrien (Quader oder Zylinder)	Teils komplexe Bauteilgeometrie (z.B. Flügel)
Gerade 2D- Achsen, meist mit rechtwinkliger Anordnung zueinander	Komplexe 3D-Längsachse oft mit schiefwinklig dazu angeordneten 2D-Lagerachsen, da Widerlager und Pfeiler i.d.R. der überquerten Strecke folgen
Vertikale Aufteilung des Bauwerks in Geschosse	Aufteilung des Bauwerks in Tragwerksteile (Überbau, Unterbauten) oder Fahrbahnen
Konstante Trägerquerschnitte	Häufig variable Trägerquerschnitte (Vouten im Auflagerbereich, variable Querschnittshöhen)

Positionierung der Bauteile im Bezug zu Achsen und Geschoss	Positionierung der Bauteile relativ zur Straßenachse bzw. relativ zu anderen Bauteilen mit teils komplexer Geometrie
Modellierung des umgebenden Geländes untergeordnet (nur für Objektplanung Baugrube und Freianlagenplanung)	Modellierung des umgebenden Geländes nötig, um Bauwerk in Gelände einzupassen (Hinterfüllung der Widerlager, Böschungstrepfen, Anschluss an Straße, ...)
	Modellierte Bauteile müssen Richtzeichnungen (RIZ-Ing) einhalten -> zusätzliche geometrische Zwangspunkte

Grundsätzlich lässt sich feststellen, dass die korrekte Modellierung und Platzierung von Bauteilen im Brückenbau komplexe geometrische Operationen erfordert, welche durch viele Zwangspunkte erschwert werden. Standard-Hochbauten können dagegen oft mit einfachen quader- oder zylinderförmigen Bauteilen modelliert werden, welche relativ zu geraden Achsen positioniert werden. Im Unterschied dazu ist die Modellierung von außergewöhnlichen Hochbauten, z.B. großen Umbaumaßnahmen am Bestand oder ungewöhnlicher Architektur, deutlich aufwendiger und kann daher mindestens genauso komplex sein, wie eine Modellierung im Ingenieurbau.

Nachdem die Modellierung der Brücke abgeschlossen ist, kann im nächsten Schritt ein Datenaustausch zwischen unterschiedlichen Gewerken zu Koordinationszwecken oder zur Überprüfung des Modells mit Model Checking Software erfolgen. Dafür wird meist das IFC-Format verwendet, welches nachfolgend erläutert wird.

2.3 Industry Foundation Classes (IFC)

In diesem Abschnitt wird das IFC-Format von bSI genauer analysiert. Dabei werden zunächst die Grundlagen, dann die Grundprinzipien des Schemas und anschließend die Neuerungen durch die Einführung der Version IFC4x3 erläutert.

2.3.1 buildingSMART International (bSI)

Die länderübergreifende Organisation *buildingSMART International* (bSI) hat es sich zur Aufgabe gemacht, die Umsetzung von BIM im Bauwesen und Facility Management zu unterstützen. Dafür stellt sie verschiedene Lösungen und offene, herstellerneutrale Standards für digitale Kollaboration und Workflows zur Verfügung. Außerdem kümmert

sich bSI um die Zertifizierung sowohl von BIM-Software als auch von BIM-Anwendern. (buildingSMART International 2022c)

Die Organisation wurde 1995 als *Industry Alliance for Interoperability* (AIA) (kurz darauf *International Alliance for Interoperability*) von einem Zusammenschluss mehrerer amerikanischer Unternehmen, z.B. Autodesk, gegründet, um einen Standard für die Interoperabilität von BIM-Software, die Industry Foundation Classes (IFC), zu entwickeln (Wix 1999). Im Jahr 2006 erfolgte eine methodische Umorientierung der Organisation, die von nun an die wirtschaftlichen Vorteile eines interoperablen Planungsprozesses in den Mittelpunkt stellte. Im Rahmen der Umorientierung wurde der Name zum heutigen *buildingSMART* geändert. (Laakso und Kiviniemi 2012)

bSI ist in mehrere Chapters und Rooms untergliedert. Die Chapters sind nationale Organisationen, z.B. buildingSMART Deutschland e.V., die die Ziele von bSI im jeweiligen Land vertreten und umsetzen (buildingSMART International 2021b). Die Rooms sind chapterübergreifende Gruppen von Spezialisten, die sich jeweils mit einem bestimmten Thema, z.B. Infrastruktur oder Regulierungsprozesse, befassen (buildingSMART International 2022b).

2.3.2 Einführung IFC

Die *Industry Foundation Classes* (IFC) sind der wichtigste Standard von bSI und werden in der ISO 16739-1 geregelt (ISO 2018). Sie stellen ein Datenmodell dar, mit dem sowohl die geometrischen als auch die semantischen Daten eines Bauwerks detailliert beschrieben werden können. Die Verwendung von IFC als herstellerneutrales Datenaustauschformat ist im Bauwesen weit verbreitet. (Borrmann et al. 2019)

Mit IFC wird die Interoperabilität zwischen unterschiedlichen BIM-Softwares unterstützt. Zu diesem Standard gehören ein Datenschema mit zugehöriger Dokumentation in HTML und Definitionen von Eigenschaftensätzen und Mechanismen zum Austausch von Daten. (buildingSMART International 2022a)

Diese werden als computer-interpretierbare Schemata in der Datenmodellierungssprache EXPRESS, der XML-Schemadefinitionssprache XSD und der Ontologiebeschreibungssprache ifcOWL veröffentlicht (buildingSMART Technical 2022a). Alle diese Schemata und Dokumentationen sind öffentlich zugänglich.

Aufgrund der ständig laufenden Erweiterungen und Verbesserungen des IFC-Standards werden von bSI in unregelmäßigen Abständen aktualisierte Versionen von IFC

inklusive der zugehörigen Dokumentationen veröffentlicht. Abbildung 1 gibt einen Überblick über die Veröffentlichungen zwischen 1994 und 2012.

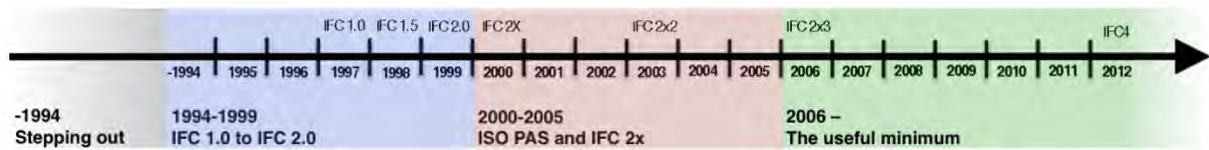


Abbildung 1: Zeitstrahl der IFC-Versionen (Quelle: Laakso und Kiviniemi 2012)

Viele dieser Versionen stellen lediglich Zwischenstände dar und werden nach Veröffentlichung der endgültigen Version zurückgezogen. Aktuell (Stand Anfang 2023) gibt es zwei offiziell gültige, in ISO 16739 standardisierte Versionen, IFC2x3 TC1 (2.3.0.1) von 2007 und IFC4 ADD2 TC1 (4.0.2.1) von 2017. (buildingSMART Technical 2022a)

Die Entwicklung der neuesten Version IFC4x3 TC1 (4.3.1.0) ist von Seiten von bSI so weit abgeschlossen, dass eine Anwendung der Version sowie eine Implementierung in Software möglich ist. Außerdem wurde die Version IFC4x3 von der ISO als vorläufiger Standard akzeptiert und sollte im Laufe des Jahres 2023 als neue ISO 16739 herausgegeben werden. (Kelly 2022)

Bei der Entwicklung von IFC4x3 wurde das bisher hochbaufokussierte Format für den Infrastrukturbereich erweitert, sodass nun auch Straßen- und Bahnbaumodelle in IFC detailliert abgebildet werden können.

Die freie Zugänglichkeit, die Herstellerneutralität und die weite Verbreitung des Standards, vor allem der Version IFC2x3, machen IFC unersetzbar bei der Realisierung von *openBIM*. Ein weiterer Vorteil ist, dass die Daten im IFC-Format auch lange nach der Planung eines Bauwerks und der damit verbundenen Modellerstellung lesbar bleiben und so über die gesamte Lebensdauer eines Bauwerks, zwischen 50 und 100 Jahre, genutzt werden können. (Borrmann et al. 2021a)

Damit die Daten aus älteren IFC-Versionen über den Lebenszyklus eines Bauwerks lesbar bleiben, legt bSI bei der Weiterentwicklung von IFC besonderen Wert auf Kompatibilität zwischen alten und neuen Versionen. Der Fokus liegt hierbei auf der *Backward-Kompatibilität*. Das bedeutet, dass eine Software, welche eine neuere IFC-Version unterstützt, auch ein Modell, welches in einer älteren IFC-Version vorliegt, lesen kann. Wenn möglich wird auch eine *Forward-Kompatibilität* eingehalten, bei der eine Datei, die mit einer neueren Version geschrieben wurde, von einer Software, die nur eine ältere Version unterstützt, gelesen werden kann. (buildingSMART International 2022a)

Die Kompatibilität zwischen verschiedenen IFC-Versionen lässt sich anhand ihrer vierstelligen Nummerierung ableiten, z.B. 4.3.1.0. Die Notation lautet „*Major.Minor.Addendum.Corrigendum*“. *Major-Versionen* enthalten Erweiterungen und/oder Streichungen des Anwendungsbereichs des Schemas und können daher Brüche in der Kompatibilität enthalten. Bei den *Minor-Versionen* mit Funktionserweiterungen ist eine Kompatibilität für das *Core Schema* garantiert, für andere Definitionen aber nicht. Verbesserungen bereits bestehender Funktionen werden als *Addendums* veröffentlicht und sind *forward-kompatibel*. Als *Corrigendums* werden nur Verbesserungen der Dokumentation veröffentlicht, bei denen volle Kompatibilität garantiert werden kann, da das Datenschema nicht verändert wird. (buildingSMART Technical 2022a)

Somit ist das *Core Schema* von IFC 4x3 (4.3.1.0) mit der standardisierten Version IFC4 ADD2 TC1 (4.0.2.1) kompatibel, mit IFC2x3 TC1 (2.3.0.1) bestehen aber u.U. Inkompatibilitäten.

Das IFC-Datenschema IFC4x3 beschreibt >870 *Entities* (Datenstrukturklassen), >430 *Typen*, >750 *Property und Quantity Sets* und etwa 2500 *Eigenschaften* sowie *Funktionen* und *Regeln*. Dazu kommen >340 *Property Enumerations*, bei denen ein Attributwert namentlich aus einer Liste möglicher Attributwerte ausgewählt werden kann. (buildingSMART International 2022a)

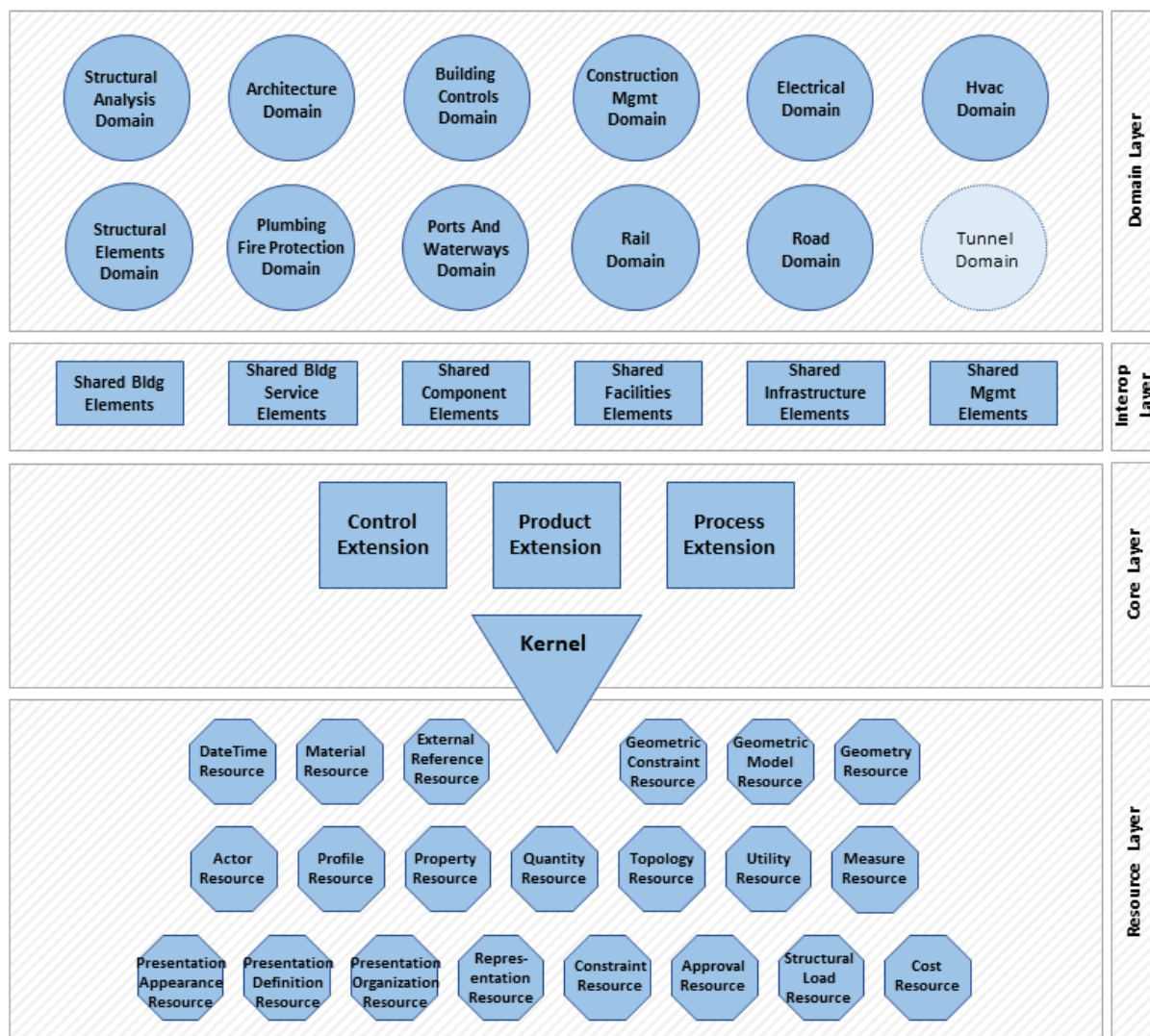
Im Folgenden wird *Entities* als Oberbegriff für die gerade aufgezählten Begriffe verwendet.

Mit Property Sets können Modellobjekten zusätzliche Eigenschaften zugewiesen werden. Die Eigenschaften werden als Name-Wert-Datentyp-Einheit-Tupel definiert und thematisch in den Property Sets gruppiert. Zusätzlich zu den von bSI standardisierten können auch benutzer- oder projektspezifische Property Sets erstellt werden. (Borrmann et al. 2021a)

In den folgenden Kapiteln wird IFC in der aktuellen Version 4x3 genauer erläutert.

2.3.3 Aufbau in Schichten

Um die Vielzahl an Entities zu strukturieren, ist das Datenmodell des IFC-Formats in 4 Schichten unterteilt, welche in Abbildung 2 dargestellt sind. Die Definition jeder einzelnen *Entity* wird genau einer Schicht zugeordnet. Das bedeutet, dass jede *Entity-Definition* in IFC genau einer der blauen Kacheln in Abbildung 2 zugeordnet werden kann. (buildingSMART International 2022a)



Industry Foundation Classes version 4.3.x Architecture overview
 © buildingSMART International Ltd.

Abbildung 2: Übersicht über das Schema von IFC 4.3.x (buildingSMART International 2022a)

Die unterste Schicht, das *Resource Layer*, enthält die Definitionen von 21 Ressourcen (buildingSMART International 2022a). Zu diesen Ressourcen gehören unter anderem Definitionen zur Beschreibung der Topologie (*Topology Resource*), der Geometrie (*Geometry* und *Geometric Model* und *Constraint Resource*) oder des Materials (*Material Resource*) eines Modellobjekts. Die hier beschriebenen grundlegenden Datenstrukturen werden lediglich von Objekten mit Klassen höherer Layer referenziert und können nicht als eigenständige Objekte im IFC-Modell existieren, da sie keine eigene GUID (*Global Unique Identifier*) besitzen. (buildingSMART International 2022a)

Im darüberliegenden *Core Layer* werden die grundlegendsten Elemente definiert. Es besteht aus dem Kernel und drei Extensions. Alle Elemente dieses Layers leiten sich aus der obersten generischen Entity *IfcRoot* ab, welcher im Kernel definiert wird. Dadurch erhalten sie eine GUID und optional einen Namen sowie Informationen zu

Eigentümerschaft und Versionsverlauf. Die vorgegebenen Strukturen, Beziehungen und Konzepte des *Core Layer* werden in den höheren Schichten referenziert und spezialisiert.

Den Ausgangspunkt des gesamten IFC-Schemas bildet der *IfcKernel*. In diesem werden die grundlegendsten Objekte, Beziehungen und Eigenschaften beschrieben. Die entsprechenden abstrakten Basisklassen *IfcObjectDefinition*, *IfcRelationship* und *IfcPropertyDefinition* leiten sich direkt von *IfcRoot* ab. Diese Grundelemente werden mit den drei *Core Extensions* weiter spezifiziert.

Die *Control Extension* enthält Klassen von Kontrollelementen und definiert die Zusammenhänge zwischen diesen Klassen, Objekten, die von *IfcObjectDefinition* abstammen, und zugehörigen Objekten aus dem *Resource Layer*. In der *Process Extension* werden Informationen darüber erfasst, wie Prozesse in einer logischen Reihenfolge abgebildet werden können. Das beinhaltet auch die Planung und Terminierung von Arbeiten inklusive der dafür benötigten Verfahren und Ressourcen. Mit Hilfe der *Product Extension* werden physische Objekte spezifiziert, welche i.d.R. eine physische Form und eine räumliche Positionierung im Modell haben. Beispiele für solche Objekte sind räumliche Projektstrukturen wie *IfcFacility*, Raster und dreidimensionale Infrastrukturachsen (*IfcAlignment*) sowie Basiselemente für Bauteile und Öffnungen. (buildingSMART International 2022a)

Die nächsthöhere Schicht ist das *Interoperability Layer*. Darin werden in sechs Datenschemas spezialisiertere Elemente, die von mehreren Domänen gemeinsam benutzt werden, definiert. Das Interoperability Layer ist wichtig für domänenübergreifende Zusammenarbeit, da mit den hier festgelegten Elementen für mehrere Domänen relevante Bauwerksinformationen ausgetauscht werden können (buildingSMART International 2022a). Hier beschriebene Entities werden von Entities aus dem Core Layer abgeleitet und beinhalten beispielsweise wichtige Bauteilklassen wie *IfcWall* oder *IfcSlab* (Borrmann et al. 2021a).

Die oberste Schicht bildet das *Domain Layer*, in welchem spezialisierte Elemente für die 12 unterschiedlichen, in Abbildung 2 dargestellten, Domänen beschrieben werden (buildingSMART International 2022a). Die in Abbildung 2 angedeutete Domäne Tunnelbau wird aber erst mit IFC4x4 umgesetzt und steht daher noch nicht zur Verfügung

(Kelly 2022). Die Elemente in dieser Schicht sind in sich abgeschlossen, d.h. sie können nicht weiter spezialisiert werden, weshalb sie nicht von Elementen anderer Layer referenziert werden können (buildingSMART International 2022a).

2.3.4 Vererbungshierarchie

Durch den eben beschriebenen Aufbau des Datenmodells in Schichten lässt sich eine immer weitere Spezifizierung der Entities ausgehend vom *Core Layer* zu den oberen Schichten erkennen. Dabei wird das aus der Datenmodellierung bekannte Konzept der *Vererbung* verwendet.

Vererbung beschreibt in diesem Zusammenhang, dass eine spezialisiertere Klasse (Kindklasse) von einer generelleren Klasse (Superklasse) Eigenschaften erbt. Die Kindklasse hat im Regelfall weitere zusätzliche Eigenschaften, die für die weitere Spezialisierung notwendig sind. Umgekehrt stellt eine Superklasse eine *Generalisierung* der Kindklasse dar. Durch dieses Konzept erhält ein Datenmodell eine Hierarchie. (Artus et al. 2021)

Zum besseren Verständnis der Vererbungshierarchie wird exemplarisch ein Gehweg (engl. Sidewalk) in die Hierarchie des IFC-Datenschemas eingeordnet (Abbildung 3).

Ein Gehweg wird in IFC der Entity *IfcSlab* zugeordnet und mit dem Attribut *PredefinedType* als Typ *Sidewalk* festgelegt (rote Ellipse in der Abbildung). Das Attribut ist vom Datentyp *IfcSlabTypeEnum* und kann als Enumeration aus einer Liste möglicher Werte (*Type Values*) ausgewählt werden (blau hinterlegtes Feld mit abgerundeten Ecken).

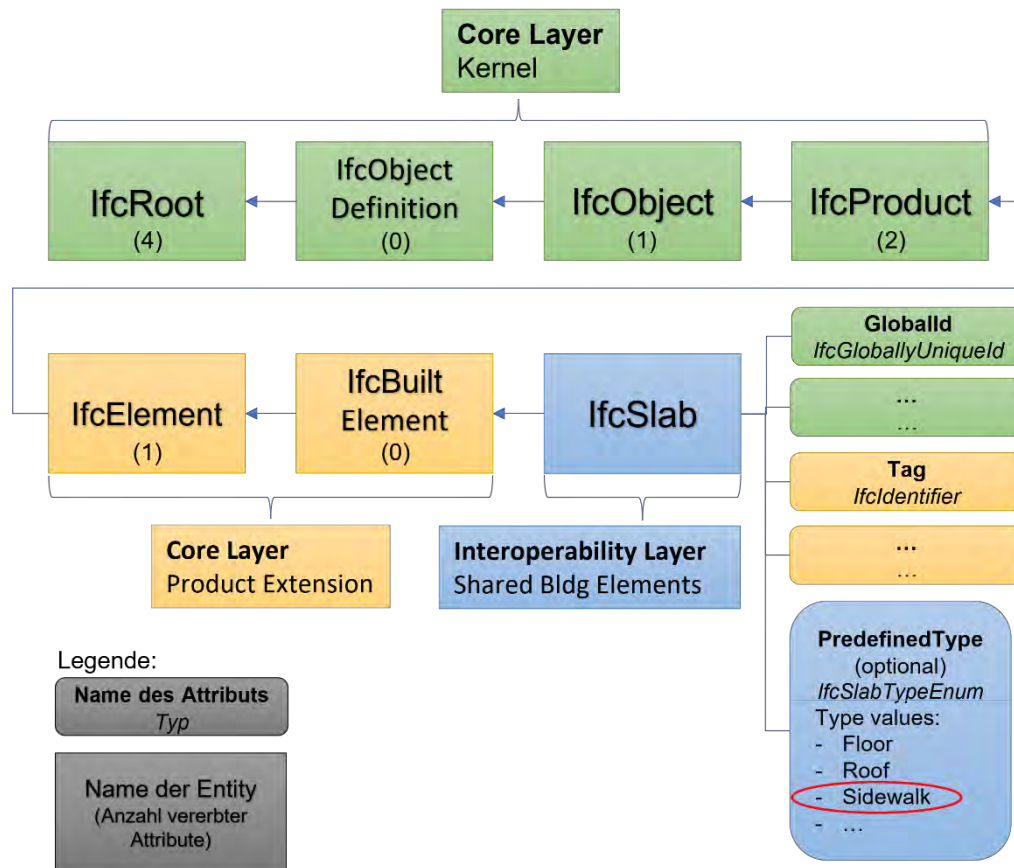


Abbildung 3: Beispiel Vererbungshierarchie anhand *IfcSlab* (Quelle: Eigene Darstellung anhand von *buildingSMART International 2022a*)

Wie alle anderen Entities außerhalb des *Resource Layer* lässt sich *IfcSlab* über eine *Generalisierung* (blaue Pfeile) durch die verschiedenen Schichten des IFC-Datenmodells bis zur generischen Superklasse *IfcRoot* zurückverfolgen. In der Abbildung ist das *Interoperability Layer* blau dargestellt, die *Product Extension* des *Core Layer* gelb und der *Kernel* des *Core Layer* grün. Attribute werden farblich der Schicht zugeordnet, in der sie definiert werden. Die Zahlen in den Klammern in Abbildung 3 geben die Anzahl an direkten Attributen an, die *IfcSlab* von der jeweiligen Oberklasse erbt. Folgt man dem Weg der Pfeile, lässt sich erkennen, dass *IfcSlab* insgesamt 8 Attribute erbt, zum Beispiel *GlobalId* von *IfcRoot* und *Tag* von *IfcElement*. Nur das optionale Attribut *PredefinedType* wird in der Kindklasse *IfcSlab* neu definiert.

Zusätzlich zu den acht direkten Attributen erbt *IfcSlab* 26 *inverse Attribute*. Mit diesen können Abfragen zu verwandten Daten definiert werden. Beispielsweise kann mit dem inversen Attribut *HasOpenings* die Verknüpfung zwischen einem *IfcSlab* und einer oder mehreren Bauteilöffnungen beschrieben werden. (*buildingSMART International 2022a*)

2.3.5 Objektbeziehungen

Die mittels Vererbung spezifizierten Kindklassen können, anderes als ihre abstrakten Superklassen, instanziiert und die erzeugten Objekte anschließend miteinander in Beziehung gesetzt werden. In diesem Zusammenhang wird *Objekt* als eine Instanz einer Klasse verstanden.

Die Beziehungen zwischen Objekten werden in IFC mittels dazwischen geschalteten Beziehungsobjekten beschrieben. Diese sind Instanzen von Unterklassen von *IfcRelationship*, das direkt von *IfcRoot* abstammt. (Borrmann et al. 2021a)

Es gibt sechs übergeordnete Beziehungstypen in IFC. *IfcRelAssigns* weist ein Objekt einem anderen zu, zum Beispiel eine Materialdefinition zu einem Modellobjekt. Mit *IfcRelAssociates* kann eine Verbindung zu einer Informationsquelle beschrieben werden, beispielweise zu einer externen Bibliothek oder einer Klassifizierung. *IfcRelConnects* beschreibt die Verbindung von Objekten unter bestimmten Voraussetzungen wie etwa die Zuordnung eines Objekts zu einem Raum. Objekte und Eigenschaften werden über *IfcRelDeclares* zu einem Projekt oder einer Projektbibliothek zugeordnet. Die Subklassen von *IfcRelDefines* werden verwendet, um Property Sets oder Objekttypen zu Objekten zuzuordnen oder Property Set Vorlagen (*Templates*) zu Property Sets. (buildingSMART International 2022a)

Mit *IfcRelDecomposes* wird eine Teil-Ganzes-Beziehung ermöglicht, zum Beispiel die Unterteilung eines Tragwerks in Balken und Stützen oder von Gesamtkosten in Teilkosten (buildingSMART International 2022a). Dabei wird unterschieden nach Nesting-Beziehungen (*IfcRelNests*), bei denen Teile eine vorgegebene Reihenfolge haben, und Aggregations-Beziehungen (*IfcRelAggregates*) mit Teilen ohne Reihenfolge. Außerdem werden hier auch Aussparungsbeziehungen (*IfcRelVoidsElement*) eingeordnet. (Borrmann et al. 2021a)

Alle übergeordneten Beziehungstypen außer *IfcRelDeclares* können nicht instanziiert werden und haben dafür stattdessen mehrere spezifizierte Unterklassen (buildingSMART International 2022a).

Als Beispiel werden in Abbildung 4 die Beziehungen zwischen einer Öffnung in einer Wand und der räumlichen Struktur eines Bauwerks erläutert. Objekte sind in der Abbildung blau hinterlegt und Beziehungen gelb.

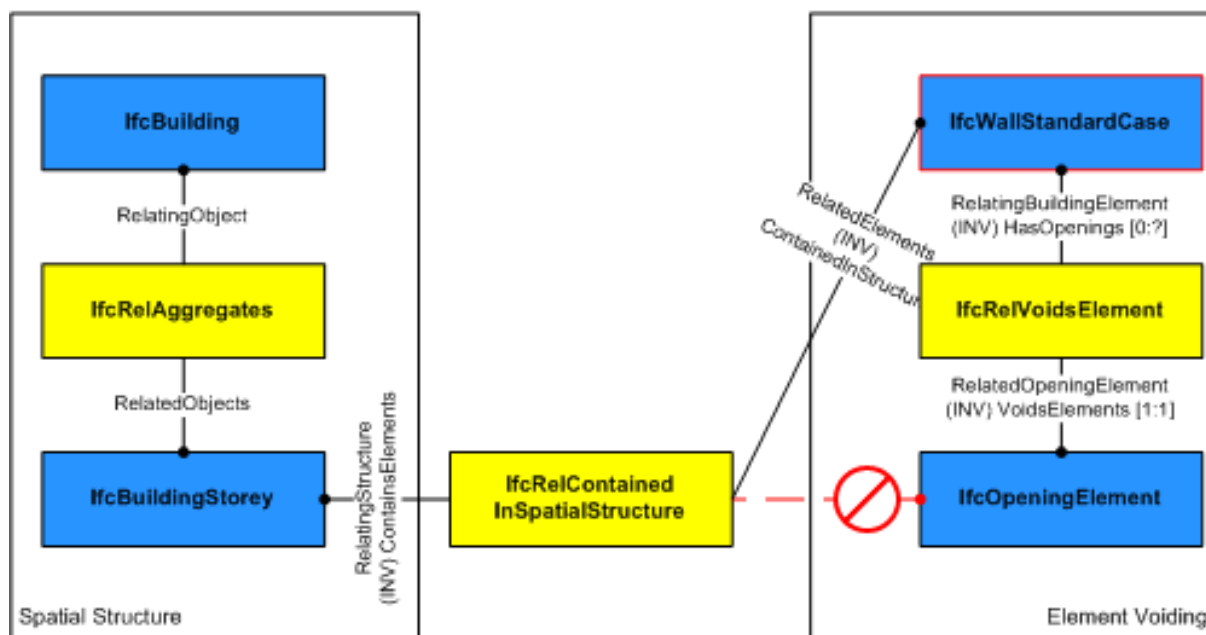


Abbildung 4: Aussparungsbeziehung (Quelle: buildingSMART International 2022a - Figure 5.4.3.62.A)

Zunächst wird die räumliche Hierarchie (engl. *Spatial Structure*) eines IFC-Modells betrachtet (linke Seite). Ein Gebäude (*IfcBuilding*) ist in mehrere Stockwerke (*IfcBuildingStorey*'s) unterteilt. Die Teil-Ganzes-Beziehung zwischen Bauwerk und Stockwerk wird durch eine *IfcRelAggregates*-Beziehung beschrieben. Sie verbindet das Bauwerk (*RelatingObject*) mit den ihm zugehörigen Stockwerken (*RelatedObjects*). Eine Wand wird über eine *IfcRelContainedInSpatialStructure*-Beziehung, einer Subklasse von *IfcRelConnects*, einem Stockwerk zugeordnet. Eine zugehörige Aussparung wird durch ein *IfcOpeningElement* repräsentiert und über eine *IfcRelVoidsElement*-Beziehung der Wand zugeordnet. Sie ist ausschließlich über die zugehörige Wand in die räumliche Struktur des Bauwerks eingeordnet und hat keine eigene Beziehung zum zugehörigen Stockwerk.

Eine Wand kann dabei nur einem Geschoss zugeordnet werden und eine Aussparung nur einer Wand. Im Gegenzug kann ein Geschoss mehrere Wände beinhalten und eine Wand mehrere Aussparungen. Hat die Wand mehrere Aussparungen, so hat sie mehrere *IfcRelVoidsElement*-Beziehungen, da diese als 1:1-Beziehungen nur mit jeweils einem *IfcOpeningElement* verknüpft sein können. Verläuft eine Wand über mehrere Geschosse, so kann sie nur einem zugeordnet sein, i.d.R. dem untersten Geschoss, in dem sie sich befindet, und wird von den darüberliegenden Geschossen lediglich referenziert über *IfcRelReferencedInSpatialStructure*. (buildingSMART International 2022a)

2.3.6 Geometrierepräsentationen

Die bisher erläuterten Grundprinzipien von IFC behandeln ausschließlich die Semantik der Objekte, welche strikt von der Geometrie getrennt ist. Die Semantik ist für alle Objekte ausschlaggebend, da sie einem Objekt eine Identität im Modell gibt. Für Objekte einer Kindklasse von *IfcProduct* wird i.d.R. eine geometrische Darstellung im Modell erforderlich. Da je nach Anwendungsfall u.U. eine andere Geometrierepräsentation benötigt wird, ist es in IFC möglich, einem semantischen Objekt keine, nur eine oder mehrere Repräsentationen zuzuordnen. (Borrmann et al. 2021a)

Grundsätzlich wird bei der Beschreibung von Geometrie zwischen zwei Verfahren unterschieden. Unter dem *expliziten Verfahren* versteht man eine Beschreibung eines Körpers über seine Oberfläche, während beim *impliziten Verfahren* die einzelnen Schritte des Konstruktionsprozesses des Körpers festgehalten werden. (Borrmann und Berkahn 2021)

Zu den *expliziten Verfahren* zählen die *Boundary Representation Methode* (BRep) und die *triangulierte Oberflächenbeschreibung*. Bei der gängigsten Methode BRep werden zur Beschreibung der Topologie eines Körpers dessen Berandungselemente hierarchisch miteinander verknüpft. Das bedeutet, ein Körper wird über die begrenzenden Flächen (engl. *faces*) beschrieben, diese über ihre Kanten (*edges*) und diese wiederum über ihre Anfangs- und Endpunkte (*vertices*). Die Zuordnung der Berandungselemente zueinander erfolgt entweder tabellarisch oder über einen Vertex-Edge-Face-(vef)-Graph. Die Geometrie des Körpers wird durch die xyz-Koordinaten der Punkte und bei gekrümmten Kanten bzw. Flächen zusätzlich durch die geometrische Beschreibung der Kanten festgelegt. Wird ein Körper als vereinfachte Variante von BRep mit einem Dreiecksnetz angenähert spricht man von *triangulierter Oberflächenbeschreibung*. (Borrmann und Berkahn 2021)

Explizite Methoden werden eher für die Visualisierung von Bauwerken oder dem sie umgebenden Gelände genutzt. Sie können von vielen Softwares einfach interpretiert werden, haben aber u.U. einen hohen Speicherverbrauch, wenn gekrümmte Flächen im Modell vorliegen. Außerdem kann explizite Geometrie nicht modifiziert werden. Für Use Cases, die eine Änderung des Modells erfordern, ist daher eine implizite Repräsentation nötig. (Borrmann et al. 2021a)

Die bekanntesten impliziten Methoden sind *Constructive Solid Geometry* (CSG) und *Rotations- und Extrusionsverfahren*. Bei CSG werden zur Konstruktion eines Körpers

die Grundkörper Würfel, Zylinder und Pyramide über Boolesche Operationen (Vereinigung, Schnitt & Differenz) kombiniert. Beim für BIM-Anwendungen implementierten erweiterten CSG können beliebige Grundkörper kombiniert werden, was besonders bei der Modellierung mit Abzugskörpern genutzt wird. Das Grundprinzip bei *Rotations- und Extrusionsverfahren* ist das Entlangziehen einer 2D-Geometrie (Profil) an einem dreidimensionalen Pfad. Je nachdem wie das Profil oder der Pfad festgelegt sind, spricht man von Extrusion (gerader Pfad), Rotation (Drehen des Profils um eine Achse), Sweep (beliebiger Pfad) oder Lofting (Übergang von einem Profil zu einem anderen entlang eines Pfads mittels Interpolation). (Borrmann und Berkahn 2021)

Die Vorteile einer impliziten Repräsentation sind eine gute Nachvollziehbarkeit und Modifizierbarkeit der Konstruktionsschritte und ein geringerer Datenumfang. Allerdings müssen bei einem Datenaustausch das Sender- und Empfängerprogramm die Geometriedaten auf dieselbe Weise interpretieren, was die Implementierung einer solchen Schnittstelle sehr komplex macht. (Borrmann und Berkahn 2021)

Bei der Planung von Infrastrukturanlagen werden von den meisten Softwares 2,5D-Modelle erzeugt. Dabei wird das Bauwerk über die Achse im Lageplan, die Gradienten im Höhenplan und den Querschnitt beschrieben und nur intern als 3D-Modell hinterlegt. Die so definierten Parameter werden gespeichert, wodurch sich eine implizite Geometrierepräsentation ergibt. So bleiben die Parameter zugänglich und können während der Planungsphase geändert werden. Aus den Parametern kann für die Visualisierung oder andere Use Cases ein explizites 3D-Volumenmodell erzeugt werden. Dieses kann jedoch nicht geändert werden und muss deswegen bei eventuellen Änderungen der Parameter neu generiert werden. Die Zugänglichkeit der Parameter kann auch für den Use Case ACCC (sh. Kapitel 3) genutzt werden. (Häußler und Borrmann 2020)

In IFC können sowohl explizite als auch implizite Verfahren zur Geometrierepräsentation verwendet werden, um unterschiedliche Use Cases abzudecken.

Die Zuordnung von Geometrierepräsentationen zu Objekten geschieht über das in *IfcProduct* definierte Attribut *Representation*. Dadurch entsteht der Verweis erst zu *IfcProductRepresentation* und damit zu *IfcRepresentation*. Mit dem dort definierten Attribut *Items* können einem Objekt ein oder mehrere *IfcRepresentationItem(s)* zugewiesen werden. Diese *Items* können geometrische oder topologische Beschreibungen oder Stilvorgaben (*IfcStyledItem*) zur Darstellung von Geometrierepräsentationen oder Materialien sein. (buildingSMART International 2022a)

Die geometrischen und topologischen *Items* werden im *Resource Layer* definiert und gehören entweder zur *Geometric Model Resource*, der *Geometry Resource* oder der *Topology Resource* (vgl. Abbildung 2 unten). (Borrmann et al. 2021a)

In der *Geometry Resource* werden Entities für die Darstellung der geometrischen Form eines geometrischen Elements definiert, z.B. Punkte, Vektoren, Kurven, Flächen oder Operatoren für die Koordinatentransformation. Unter der *Geometric Model Resource* finden sich Beschreibungen von geometrischen Objektmodellen wie CSG-Modelle, Halbräume, Sweeps, B-Rep, Oberflächen- und tessellierte Modelle und die für eine präzise Geometriebeschreibung nötigen Daten. Die *Topology Resource* enthält Entities für die Beschreibung von Topologie (z.B. *Vertex*, *Edge*, *Face*) und topologischen Strukturen (z.B. Pfade, Schleifen oder Schalen) sowie der Orientierung dieser. (buildingSMART International 2022a)

2.3.7 Platzierung von Objekten

Um die einzelnen geometrischen Objekte in einem BIM-Modell anzeigen zu können, müssen sie an eine konkrete Position im Modell platziert werden. Die zugehörigen Entities in IFC werden in der *IfcGeometricConstraintResource* beschrieben. Das IFC-Datenmodell von IFC4x3 unterstützt vier Arten von Objektplatzierung:

- Absolute Positionierung (bezogen auf ein Weltkoordinatensystem)
- Relative Positionierung (bezogen auf die Position eines anderen Objekts)
- Achsbezogene Positionierung
- Lineare Platzierung entlang einer Bezugskurve (ggf. mit Versatz)

Jedem Objekt, dem eine Geometrie zugeordnet ist, muss ein Platzierungsschema, welches von *IfcObjectPlacement* abhängt, zugeordnet werden. (buildingSMART International 2022a)

In IFC wird die absolute Platzierung i.d.R. nur für ein Objekt verwendet, welches anschließend von den anderen Objekten und den Achsen referenziert wird. Dieses Konzept wird als *Local Placement* bezeichnet und sorgt dafür, dass das Modell sehr flexibel für Änderungen bleibt. Ändert sich beispielsweise die globale Position, werden die Koordinaten nur an einer Stelle geändert und alle anderen relativen Koordinaten bleiben unverändert.

Im Hochbau wird meist das räumliche Objekt *IfcSite*, welches das Baufeld abbildet, absolut positioniert. Das Bauwerk (*IfcBuilding*) wird dann relativ zum Baufeld positioniert, das Geschoss (*IfcBuildingStorey*) relativ zum Bauwerk und beispielsweise eine Wand (*IfcWall*) relativ zu ihrem Bezugsgeschoss. (Borrmann et al. 2021a)

Für den Infrastrukturbau ist die lineare Platzierung entlang einer Kurve unabdingbar. In der Regel werden die Trassen der Bauwerke im globalen Koordinatensystem definiert und für die weiteren Bestandteile des Bauwerks als Platzierungskurven festgelegt. Die Positionen der Bauteile werden dann mit Hilfe einer Stationierung entlang der Trasse und einem Versatz zur Trasse angegeben. (Jaud et al. 2021)

2.3.8 Relevante Neuerungen der Version IFC4x3

Mit der Version IFC4x3 gehen umfangreiche Neuerungen im IFC-Datenmodell vor allem für den Infrastrukturbereich einher. Für einen detaillierten Vergleich zwischen IFC2x3 und IFC4x3 sei hier auf die Masterarbeit von Taray aus dem Jahr 2022 verwiesen. Sie kommt dabei zum Schluss, dass mit der Repräsentation von BIM-Modellen in IFC4x3 eine zuverlässige Datengrundlage für Infrastrukturprojekte geschaffen werden kann, welche anschließend für eine Vielzahl von nachfolgenden Anwendungen verwendet werden kann. (Taray 2022)

Einige der Neuerungen werden nachfolgend kurz erläutert.

Positionierung und geometrische Beschreibung von Objekten

Die wichtigsten Neuerungen mit IFC4x3 betreffen die Positionierung von Objekten und deren geometrische Beschreibung.

Im Infrastrukturbau ist es grundsätzlich nötig die Trassen der Bauwerke zu definieren, Objekte entlang einer Trasse zu platzieren (*linear placement*) sowie das umliegende Gelände annähernd wahrheitsgetreu im Modell abzubilden. Daher wurden diese Konzepte bereits 2017 in einem frühen Stadium der IFC-Erweiterung in das IFC-Datenmodell eingefügt. (Jaud et al. 2020)

In IFC4x3 wurde das in DIN EN ISO 19148 beschriebene Konzept des linearen Bezugssystems (engl. *linear referencing*) integriert (DIN 2021b). Damit wird die unter 2.3.7 genannte „Lineare Platzierung entlang einer Bezugskurve (ggf. mit Versatz)“ ermöglicht. Um die Semantik dieser linearen Positionierung zu beschreiben, wurden die Entities *IfcPositioningElement*, *IfcLinearPositioningElement*, *IfcReferent* und *IfcRelPo-*

sitions eingeführt (Jaud et al. 2020). *IfcPositioningElement* ist eine abstrakte Oberklasse und beschreibt Elemente, welche zur Positionierung anderer Elemente verwendet werden, also Hochbauachsen (*IfcGrid*), Trassen (*IfcAlignment* als Kindklasse von *IfcLinearPositioningElement*) oder Positionen mit einem festgelegten Versatz zu einer Trasse (*IfcReferent*). *IfcRelPositions* beschreibt die Beziehung, mit der ein Objekt relativ zu einem *IfcPositioningElement* platziert werden kann. (buildingSMART International 2022a)

Zusätzlich zur Positionierung wurde auch die Beschreibung von Geometrie mit IFC4x3 grundlegend erweitert, um Infrastrukturbauten detailliert abbilden zu können. In der *Geometry Resource* wurden viele neue Entities zur Geometriebeschreibung ergänzt, z.B. *IfcClothoid*, *IfcGradientCurve* und *IfcOffsetCurveByDistance*. Die möglichen Geometriepäsentationen in der *Geometric Model Resource* wurden ebenfalls ausgeweitet. Zu den neu hinzugefügten expliziten Repräsentationen gehören tessellierte Geometrie, Triangulation und erweiterte B-Rep. Diese können für Visualisierungen und Modellierung von Gelände verwendet werden. Bei den impliziten Repräsentationen wurde die Flexibilität verbessert, beispielsweise durch *IfcSectionedSolidHorizontal*. (Taray 2022)

Alignment

Unter dem englischen Begriff *Alignment* werden in IFC drei zusammenhängende Konzepte aus dem Infrastrukturbau verstanden: Die Vorgabe eines Referenzsystems für lineare Positionierung, die Gewährleistung und Optimierung sicherer Fahrzeugbewegungen und die geometrische Konstruktion linearer Infrastrukturbauten. (Pinzenöhler 2022)

Die Definition von räumlichen Trassen (=Alignment) ist sehr relevant für den Infrastrukturbau und wurde daher bereits in IFC4.1 zum ersten Mal in IFC integriert. Die Entities, die zur Umsetzung von Alignment in BIM-Modellen benötigt werden, sind, wie in IFC üblich, in semantische und geometrische Aspekte aufgeteilt (Abbildung 5). (buildingSMART International 2022a)

Semantisch ist ein *IfcAlignment* in IFC als Kindklasse von *IfcLinearPositioningElement* definiert und kann somit zur Positionierung von Bauteilen verwendet werden (buildingSMART International 2022a). Es wird über die Beziehung *IfcRelNests* aus einem *IfcAlignmentHorizontal*, einem *IfcAlignmentVertical* und im Bahnbau zur Beschreibung der Überhöhung einem *IfcAlignmentCant* konstruiert (vgl. Abbildung 5 unten).

Diese setzen sich ebenfalls über *IfcRelNests* jeweils aus Segmenten (*IfcAlignmentSegment*) zusammen, also Teilstücken der Gesamtkurve mit derselben mathematischen Beschreibung. (Pinzenöhler 2022)

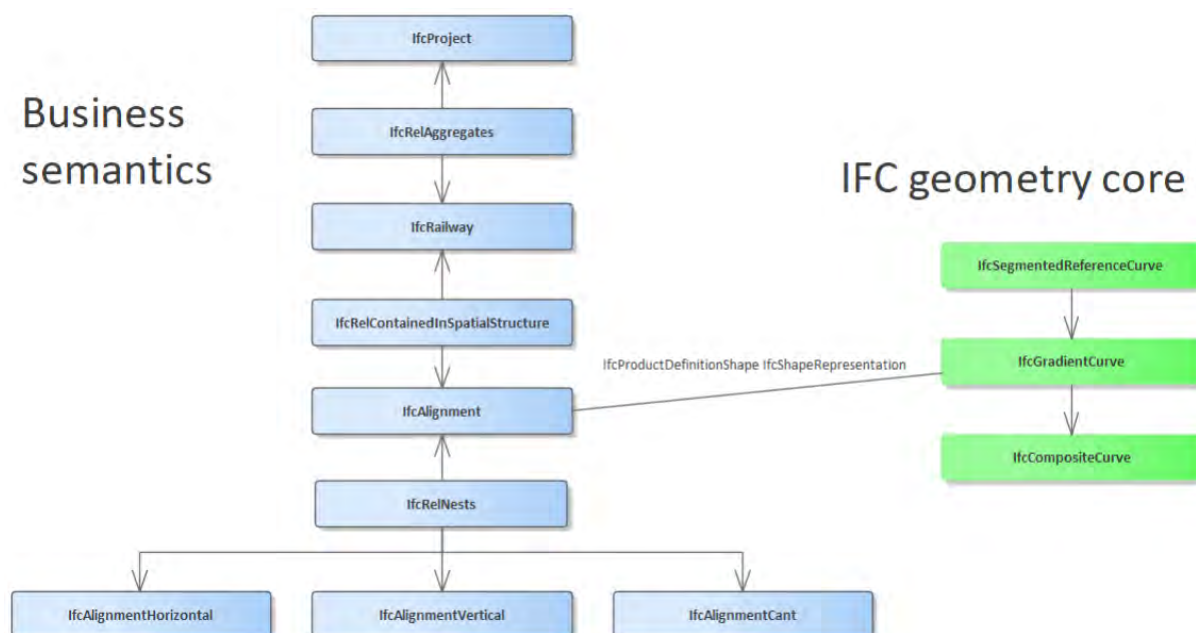


Abbildung 5: Aufteilung der Alignment-Entities in semantische Aspekte (links) und geometrische Repräsentation (rechts) (Quelle: Pinzenöhler 2022)

Die Lage einer Infrastrukturtrasse, *IfcAlignmentHorizontal*, wird im x/y-Koordinatenraum definiert. Das Höhenprofil der Trasse, *IfcAlignmentVertical*, wird mit Koordinaten entlang des *IfcAlignmentHorizontal* und z-Koordinaten festgelegt. Die Überhöhung für den Bahnbau, *IfcAlignmentCant*, wird mit Lagekoordinaten entlang des *IfcAlignmentHorizontal* und Höhenkoordinaten relativ zur entsprechenden z-Koordinate entlang des *IfcAlignmentVertical* beschrieben. (buildingSMART International 2022a)

Die Entwurfsparameter des jeweiligen Segments werden mit dem Attribut *DesignParameters* des Typs *IfcAlignmentParameterSegment* angegeben. *IfcAlignmentParameterSegment* wird in der *IfcGeometricConstraintResource* definiert und ist eine generische Oberklasse. Deren Kindklassen sind *IfcAlignmentHorizontalSegment*, *IfcAlignmentVerticalSegment* und *IfcAlignmentCantSegment*. Die Unterteilung der Segmente nach horizontal, vertikal und Überhöhung ist notwendig, da jeweils unterschiedliche Entwurfsparameter verwendet werden. (buildingSMART International 2022a)

Die Zuordnung der geometrischen Aspekte, z.B. einer *IfcGradientCurve*, zu einem *IfcAlignment* erfolgt wie in Kapitel 2.3.6 erläutert über *IfcProductRepresentation* bzw. *IfcProductDefinitionShape* und *IfcShapeRepresentation* (vgl. Abbildung 5, Verbindung blau/grün).

Für den Brückenbau ist es häufig nötig ein Alignment für die Brücke zu definieren, das vom Alignment einer Straße oder Bahnlinie abhängt. Bei dem abgeleiteten Alignment der Brücke sollte dabei *IfcOffsetCurveByDistances* als *Shape Representation* hinterlegt sein. (buildingSMART International 2022a)

Bei der Analyse unterschiedlicher Use Cases für das *IFC- Bridge Project* stellen Borrmann et al. 2019 fest, dass eine Alignment-basierte Geometrirepräsentation für den in dieser Arbeit untersuchten BIM Use Case *Konformitätsprüfung (Code Compliance Checking)* notwendig ist. Durch die Einführung von Alignment in IFC4x3 ist diese Grundvoraussetzung somit erfüllt.

Räumliche Hierarchie

Mit IFC4x3 wurde die räumliche Hierarchie, welche die Bauwerksstruktur von einzelnen Räumen (*IfcSpace*) bis hin zum Gesamtbauprojekt (*IfcProject*) abbildet, für den Infrastrukturbau angepasst (Borrmann et al. 2021a). Während in IFC2x3 jedes Bauwerk unabhängig seiner Funktion als Gebäude (*IfcBuilding*) deklariert wurde, ist in IFC4x3 an dessen Stelle die neue Entity *IfcFacility* getreten (Taray 2022). Die räumliche Hierarchie für Gebäude des Hochbaus ist in Abbildung 6 grün dargestellt. Sie ist in IFC4x3 identisch zur Hierarchie in IFC2x3 mit dem Unterschied, dass die Entity *IfcBuilding* nun von *IfcFacility* abstammt, statt von *IfcSpatialStructureElement* (buildingSMART International 2022a). Die neue räumliche Hierarchie für Infrastrukturbauwerke in IFC4x3 ist in Abbildung 6 blau dargestellt.

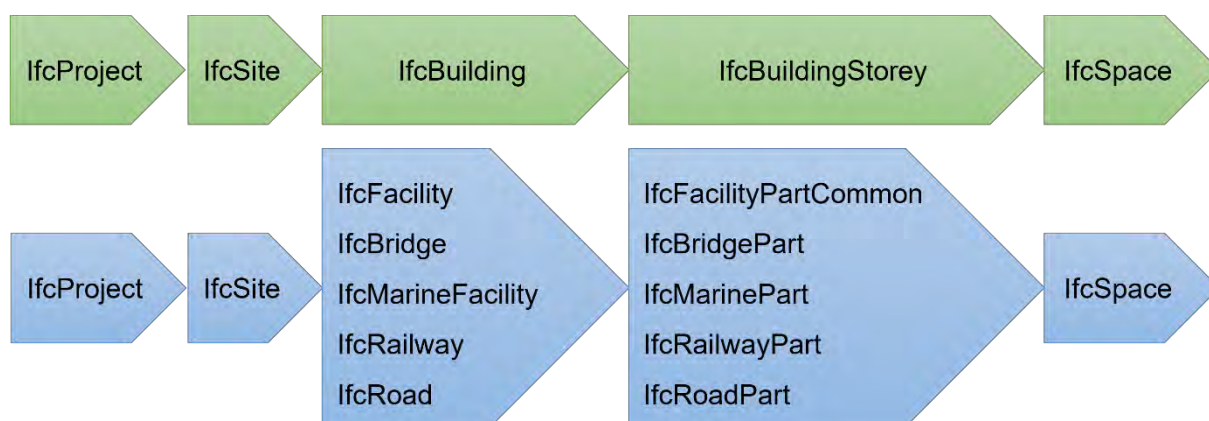


Abbildung 6: Hierarchie der räumlichen Unterteilung von Gebäuden (grün) und anderen Bauwerken (blau) in IFC (Quelle: Eigene Darstellung, angelehnt an Borrmann et al. 2021a)

Die beiden größten Unterteilungen *IfcProject* und *IfcSite*, welche das Bauprojekt und das Baufeld abbilden, sind in IFC4x3 und IFC2x3 gleich. Für eine weitere Aufgliederung kann in IFC4x3 ein Bauwerk entweder allgemein als *IfcFacility* kategorisiert werden oder spezifischer mit einem Untertyp von *IfcFacility*, mit dem die Funktion des

betreffenden Bauwerks festgelegt werden kann. In IFC4x3 werden als Untertypen zunächst *IfcBuilding*, *IfcBridge*, *IfcMarineFacility*, *IfcRailway* und *IfcRoad* beschrieben, welche zukünftig z.B. durch *IfcTunnel* ergänzt werden können. (buildingSMART International 2022a)

Im Hochbau erfolgt eine weitere Unterteilung des Gebäudes (*IfcBuilding*) in der Regel nur vertikal in Stockwerke, welche in IFC2x3 und IFC4x3 als *IfcBuildingStoreys* beschrieben werden (vgl. Abbildung 6, in grün). Dagegen wird in der Praxis eine Straße häufig horizontal in Segmente oder Spuren oder eine Brücke in Oberbau und Unterbauten aufgegliedert. Diese Art der bauwerksspezifischen Unterteilung ist mit den in IFC4x3 neu definierten Entities vom abstrakten Supertyp *IfcFacilityPart* möglich. Für alle Arten von Bauwerken außer *IfcBuilding* erfolgt eine weitere Unterteilung mit den Kindklassen von *IfcFacilityPart*: *IfcFacilityPartCommon*, *IfcBridgePart*, *IfcMarinePart*, *IfcRailwayPart* oder *IfcRoadPart*. Diese beinhalten alle ein Attribut *PredefinedType*, welches je nach Bauwerksart aus einer Liste möglicher Untergliederungen festgelegt werden kann. Für Brücken gilt beispielsweise der *Enumeration Type IfcBridgePart-TypeEnum*, der unter anderem die Brückenteile Gründung (*Foundation*), Pylon, Unterbau (*Substructure*) oder Überbau (*Superstructure*) beinhaltet. (buildingSMART International 2022a)

IfcSharedInfrastructureElements

Im Interoperability Layer wurde mit IFC4x3 eine neue Spezialisierung *IfcSharedInfrastructureElements* eingeführt (Taray 2022). Damit können Objekte der Geotechnik wie Bohrlöcher oder geotechnische Volumenmodelle sowie Strukturen von Erdarbeiten, z.B. Einschnitte, Auffüllungen oder Bodenverbesserungen, beschrieben werden. Außerdem werden dort Definitionen für Schilder und Signale festgelegt. (buildingSMART International 2022a)

Erweiterungen im Domain Layer

Im Vergleich zur Version IFC2x3 wurden für die Version IFC4x3 die vier Domänen *Rail*, *Road*, *Tunnel* und *Ports & Waterways* im *Domain Layer* ergänzt, wobei die *Tunnel Domain* noch im Entwicklungsprozess ist und erst mit IFC4x4 erwartet wird. Die Domäne *Facilities Management* aus IFC2x3 entfällt in 4x3, da deren Inhalt ins domänenübergreifende *Interoperability Layer* verschoben wurde. (Taray 2022)

In der *Road Domain* werden straßenbau-spezifische Konzepte festgehalten. Dazu gehören zum einen Straßentypen und verwandte Räume wie Autobahnen, ein- und

mehrspurige Straßen, Rad- und Fußwege sowie zugehörige Kreuzungen, z.B. Über- und Unterführungen, Rampen und Kreisverkehre. Andererseits werden der Aufbau des Belags von Straßen, Parkplätzen und anderen Einrichtungen sowie die Ausstattung von Straßen wie Beschilderung, Stromleitungen, Entwässerungen und Sicherheitseinrichtungen beschrieben. In IFC4x3 nicht berücksichtigte Elemente des Straßenbaus sind Bahnübergänge, Straßenbahnen und Stadtplanung.

Die *Ports & Waterways Domain* beschreibt entsprechende Elemente des Hafen- und Wasserwegebaus, auf die hier nicht näher eingegangen wird.

In der *Rail Domain* werden einige bahnbau-spezifische Elemente zusammengestellt, wobei große Teile des Bahnbaus bereits in höheren Schichten des Datenmodells untergebracht sind. Dazu gehören Schientypen, andere Spezialelemente des Gleisbereichs wie Schwellen und Entgleisungsvorrichtungen und räumliche Unterteilungen, die nur in Bahnbauprojekten auftreten, wie Weichenbereiche. (buildingSMART International 2022a)

Model View Definitions (MVDs)

Mit der Einführung von IFC4x3 wurden von bSI sogenannte *Basis-MVDs* vorgeschlagen, um eine Interoperabilität zwischen MVDs zu gewährleisten. Werden unterschiedliche MVDs verwendet, so besteht die Gefahr, dass Daten nach dem Export aus einer Software in einer anderen Software unlesbar sind.

Eine MVD besteht aus *Concept Templates*, die Vereinbarungen vorgeben, wie das IFC-Datenschema verwendet wird und *Exchange Requirements*, die eine Zusammenstellung von Entities und Eigenschaften für einige Use Cases darstellen. Dazu kommt eine Beschreibung, wie eine Software mit den auszutauschenden Daten umzugehen hat.

Von bSI werden offiziell drei MVDs für IFC4x3 vorgeschlagen. Die *Reference View* wurde bereits mit IFC4 eingeführt und ist mit dieser Version kompatibel. Deswegen darf sie nicht verändert werden. Die *Design Transfer View* wurde in IFC4x3 vorgeschlagen, aber nicht konkret definiert, obwohl sie von Interessentenvertretern schon lange gewünscht wird. Für den Infrastrukturbau neu hinzugekommen ist die *Alignment Based Reference View*, mit welcher alignment-basierte Daten ausgetauscht werden können. (buildingSMART International 2021a)

2.4 Zusammenfassung

Die BIM-Methode ist inzwischen im Bauwesen etabliert und umfasst viele voneinander abhängige Teilprozesse und vertragliche Vereinbarungen.

Auch im Straßen- und Brückenbau wird immer mehr die Anwendung der BIM-Methode von den Auftraggebern gefordert (Bayerisches Staatsministerium für Wohnen, Bau und Verkehr 2022). Daher werden immer häufiger Modelle von Infrastrukturbauten angefertigt, die über den gesamten Lebenszyklus des Bauwerks verwendet werden.

Der Vergleich der BIM-Modellierung von Hoch- und Brückenbauten zeigt gravierende Unterschiede hinsichtlich der geometrischen Komplexität, der Platzierung von Bauteilen und dem grundsätzlichen Aufbau der Modelle. Daraus lässt sich schließen, dass ein geeignetes Format zur Speicherung und zum Austausch von BIM-Daten sehr flexibel sein muss, um viele Arten von Bauprojekten in ausreichender Detaillierung abbilden zu können.

Die Analyse der Grundlagen von IFC zeigt ein großes Potential des Formats für die Speicherung von Informationen im Modell und deren Übergabe von einer Software zur anderen. Durch die Betrachtung der Neuerungen mit der Version IFC4x3 wird deutlich, dass das Format in dieser Version für viele Use Cases erheblich besser für Infrastrukturbauten geeignet ist als die bisher etablierte Version 2x3. Das zeigt sich vorrangig durch die Implementierung von Alignment und linearer Positionierung, welche für den Infrastrukturbau unabdingbar sind.

Bei der Weiterentwicklung von IFC für Brücken mit dem *IFC-Bridge Project* wurde der BIM-Use Case *Konformitätsprüfung* (engl. *Code Compliance Checking*) aus Zeitgründen nicht berücksichtigt (Borrmann et al. 2019). Inwieweit IFC4x3 eine modellbasierte Konformitätsprüfung von Brücken trotzdem unterstützt, wird im Rahmen dieser Arbeit evaluiert.

3 Literaturrecherche zu Automated Code Compliance Checking

In diesem Kapitel werden die Entwicklung und der Stand der Technik von *Automated Code Compliance Checking* (kurz: ACCC) zusammengefasst. Das Forschungsfeld ACCC beschreibt eine Automatisierung der Konformitätsprüfung, d.h. der Überprüfung von Planungsunterlagen auf die Einhaltung der geltenden Richtlinien.

Zunächst wird auf die Entwicklung von ACCC eingegangen, dann auf die Hauptbestandteile eines solchen Prüfprozesses. Die Hauptbestandteile werden anschließend genauer analysiert, auch im Hinblick auf technische Umsetzungen. Danach werden Rückschlüsse auf ACCC im Infrastrukturbau gezogen.

3.1 Entwicklung von ACCC

Für das Bauwesen gelten viele Gesetze, Normen, Richtlinien und Empfehlungen aus der Praxis und Bauausführung. Die Einhaltung der Gesetze und der anerkannten Regeln der Technik muss für alle Bauprojekte durch die Planer gewährleistet werden. Daher müssen die erstellten Projektunterlagen vor der Einreichung bei den zuständigen Behörden auf die Einhaltung der geltenden Richtlinien geprüft werden. Diese Überprüfung wird *Konformitätsprüfung* genannt. Die Behörden führen im Anschluss eine eigene Konformitätsprüfung durch, deren Bestehen die Grundvoraussetzung für eine Baugenehmigung ist. (Preidel et al. 2021)

Werden Planungsfehler erst auf der Baustelle erkannt und nicht während der Konformitätsprüfung, müssen kostenintensive Nachbesserungsarbeiten durchgeführt werden (Dimyadi et al. 2016). Des Weiteren könnten Kosten eingespart werden, wenn für die Konformitätsprüfung und die daraus resultierenden Anpassungen der Planung möglichst wenige Iterationsschritte benötigt werden.

Eine Konformitätsprüfung wird heute weitgehend manuell anhand von herkömmlichen 2D-Planungsunterlagen durchgeführt, was einen großen Aufwand darstellt. Zudem muss die Konformitätsprüfung für Änderungen und Anpassungen der Planung iterativ wiederholt werden, was eine monotone Tätigkeit für die Prüfer bedeutet. Daraus resultiert eine große Fehleranfälligkeit des Prozesses für menschliches Versagen. (Preidel et al. 2021)

Mit der heute immer mehr etablierten BIM-Methode werden die Bauwerksinformationen in computer-interpretierbaren Modellen vorgehalten, welche für eine (teil-)automatisierte Konformitätsprüfung viel besser geeignet sind als herkömmliche 2D-Unterlagen. (Eastman et al. 2009)

Mit der stufenweisen Einführung der BIM-Methode in allen Aspekten des Bauens in Deutschland rückt das Thema ACCC mehr in den Vordergrund. Beispielsweise hat die Bayerische Staatsbauverwaltung – Bereich Straßen- und Brückenbau schon 2020 mit der stufenweisen Einführung der BIM-Methode begonnen und will die Methode bis 2025 fest in ihren Prozessen verankert haben (Bayerisches Staatsministerium für Wohnen, Bau und Verkehr 2022)

Auf der internationalen Standardisierungsebene bei buildingSMART International gibt es unter anderem für ACCC-Themen den *Regulatory Room*. Dieser Room hat es sich zur Vision gemacht, dass die Regulierungsprozesse Schritt für Schritt mittels openBIM automatisiert werden. (buildingSMART International 2022d).

3.2 Bestandteile von ACCC

Ein grundlegender Beitrag zur Forschung in *Automated Code Compliance Checking* stammt von Eastman et al. 2009. Darin analysieren die Autoren einige frühe, industrielle Entwicklungen zu ACCC und stellen vier essenzielle Komponenten einer automatisierten Konformitätsprüfung heraus: (1) Übersetzung der Regelwerke, (2) Vorbereitung des Gebäudemodells, (3) Durchführung des Überprüfungsprozesses und (4) Report der Ergebnisse. Der Zusammenhang der Komponenten ist in Abbildung 7 dargestellt.

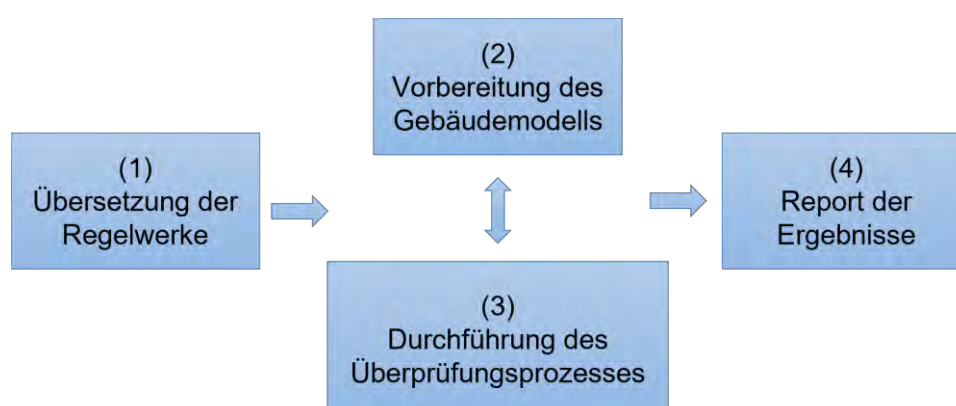


Abbildung 7: Struktur und Bestandteile einer automatisierten Konformitätsprüfung (Quelle: Eigene Darstellung, angelehnt an Eastman et al. 2009)

Grundsätzlich müssen gemeinsame Konventionen für die Komponenten (1) bis (3) geschaffen werden, denn die Regeln aus (1) und die Strukturen und Attribute des Modells aus (2) müssen zueinanderpassen, damit die Zugänglichkeit zu den benötigten Informationen für eine Überprüfung in (3) gewährleistet ist (Eastman et al. 2009).

Eastman et al. 2009 geben auch drei Strategien an, wie diese Zugänglichkeit ermöglicht werden kann: entweder liegt die Information explizit, z.B. als Parameter, im Modell vor oder sie kann vom Computer implizit während des Überprüfungsprozesses aus dem Modell selbst abgeleitet werden. Für besonders komplexe Fälle muss die Möglichkeit bestehen, dass die Information von einem externen Programm, zum Beispiel einem Energieanalyse-Tool, ermittelt und in das Model-Checking-Tool zurückübergeben wird. (Eastman et al. 2009)

Die einzelnen Komponenten (1) bis (3) werden in den folgenden Abschnitten auch bezüglich der erforschten technischen Möglichkeiten genauer analysiert. Der Report der Ergebnisse (4) hängt stark von der jeweiligen Software ab und wird deswegen nicht im Einzelnen analysiert.

3.3 Übersetzung von Regelwissen

3.3.1 Herausforderungen

Grundsätzlich gibt es zwei Möglichkeiten der Übersetzung: die *Black-Box*- und die *White-Box-Methode*. Bei der *Black-Box-Methode* wird das Regelwissen in fest-implementierte Programm-Routinen überführt und somit für den Anwender unsichtbar. Das Regelwissen kann dadurch nur noch vom Computer, aber nicht mehr vom Menschen gelesen und nachvollzogen werden. Im Gegensatz dazu steht die deutlich aufwendiger zu implementierende *White-Box-Methode*, bei der die einzelnen Verarbeitungsschritte des Prüfprozesses auch für den Anwender sichtbar und nachvollziehbar sind. Das ist wichtig, da die Planer und die prüfenden Behörden die Verantwortung für die Korrektheit der Planung und damit die Modellprüfung tragen. (Preidel et al. 2021)

Ein Problem bei der Übersetzung von Regelwissen ist die Vielzahl an zu beachtenden Richtlinien, welche das Wissen auf unterschiedliche Art, z.B. als Fließtext, Tabelle oder Piktogramm, abbilden. Viele Anforderungen aus den Richtlinien erfordern zudem ein komplexes Urteilsvermögen der Nutzer, wodurch eine Implementierung erschwert wird (Dimyadi et al. 2016). Die Übersetzung muss daher durch Fachleute aus dem Bauwesen erfolgen und kann nur teilweise automatisiert werden. Hinzu kommt, dass die

Richtlinien in natürlicher Sprache oft widersprüchlich oder unklar sind (Sydora und Stroulia 2019).

In letzter Zeit wurde auch der Wunsch nach leistungsorientierten Richtlinien lauter, welche mehr Innovationen bei Bauprojekten versprechen. Um solche Anforderungen zu erfüllen, werden in der Regel Simulationen für den Nachweis einer regelkonformen Planung benötigt. Die Formulierung leistungsorientierter Anforderungen als computer-interpretierbare Regeln oder einem regelkonformen Vergleichsmodell erfordert noch mehr Fachwissen und ist noch relativ unerforscht. (Dimyadi et al. 2016)

3.3.2 Beispiele und Forschungsansätze

In der Literatur können viele Forschungsansätze zum computer-interpretierbaren Formulieren von Regeln gefunden werden. Einige davon werden nachfolgend zusammengefasst.

SMARTcodes und RASE

Das Projekt *SMARTcodes* wurde ab 2006 für den International Code Council (ICC) entwickelt, um die Konformitätsprüfung zu vereinfachen. Der enthaltene *SMARTcodes builder* dient der Abbildung von Richtlinien im Fließtext als computer-interpretierbare, interoperable Regelsätze. (Eastman et al. 2009)

Bei der Verarbeitung der Richtlinien wird die *RASE-Technologie* verwendet, um wichtige Elemente des Fließtextes zu identifizieren und kategorisieren (Preidel 2020). *RASE* ist ein Akronym für die vier darin verwendeten Operatoren *Requirement*, *Applicability*, *Selection* und *Exceptions*. Mit dieser Technik können die für eine Konformitätsprüfung relevanten Informationen automatisiert aus Richtlinien-texten extrahiert und in konkreten Regeln ausgedrückt werden, welche anschließend in BIM-basierte Model-Checking-Software, z.B. SMC, implementiert werden können. Der Text wird dabei entsprechend der vier Operatoren mit *Tags* gekennzeichnet, welche die Struktur des Richtlinien-textes hervorheben und dadurch das Verstehen des Inhalts vereinfachen. (Hjelseth und Nisbet 2011)

Die mit RASE extrahierten Regeln müssen für die konkrete Konformitätsprüfung in ein *IFC-Constraint-Modell* überführt werden, welches die Modellanforderungen an die zu prüfenden IFC-Modelle darstellt. Mit *IfcConstraint* können Randbedingungen direkt im IFC-Modell definiert werden und über *IfcRelAssociatesConstraint* mit Objekten verknüpft werden. (Preidel 2020)

Mit der Anwendung von RASE bei *SMARTcodes* wird die Erstellung von Regelsätzen für die Konformitätsprüfung erleichtert und damit zumindest eine teilautomatisierte Erstellung von ausführbarem Code ermöglicht, auch wenn die endgültige Übersetzung der Regeln aufgrund deren Komplexität weiterhin meist manuell erfolgt (Eastman et al. 2009).

Es wird weiterhin an einer Erweiterung dieser Technologie geforscht, jedoch findet sie als Hilfsmittel bei der manuellen Regelübersetzung bereits Anwendung, beispielsweise in den Arbeiten von Häußler et al. 2021 zum ACCC im Bahnbau und Schliski 2021 zum ACCC mit der deutschen Musterbauordnung.

Semantic Web

Das Semantic Web beschreibt ein Konzept, bei dem maschinenlesbare Links zwischen Daten verfolgt werden, so dass ein semantisches Datennetz entsteht. Die Daten können über Datei- und Netzwerkgrenzen hinaus verknüpft werden, wofür sie eine globale ID (Uniform Resource Identifiers - URI) benötigen. Zu den Daten gehören sowohl Datenschemas und Vokabulare als auch Instanzdatensätze. Das Ziel dabei ist eine Interoperabilität zwischen heterogenen Daten und Informationsquellen. (Beetz et al. 2021b)

Bezogen auf ACCC kann die Methodik des Semantic Web dazu verwendet werden, Bauwerksmodelldaten mit Regeln und Randbedingungen aus den zugehörigen Richtlinien zu verknüpfen. BIM-Modelle müssen im extra für diesen Zweck standardisierten Format ifcOWL vorliegen, um vom Semantic Web ausgewertet werden zu können. (Beetz et al. 2021b)

Die Abfrage der Daten geschieht über die Grafenabfragesprache *SPARQL* (W3C 2013). Die Regeln können als WENN-DANN-Regel mit Hilfe der *Semantic Web Rule Language* (SWRL) ausgedrückt werden (W3C 2004). Mit der *Shapes Constraint Language* (SHACL) können Randbedingungen formuliert und geprüft werden (Beetz et al. 2021b).

Ein Beispiel für eine Anwendung des Semantic Web im Zusammenhang mit ACCC ist der Ansatz von Bus et al. 2018. Dabei wird eine Ontologie für die ausgewählten Richtlinien auf Basis einer vereinfachten Version von ifcOWL erstellt. (Bus et al. 2018)

Für die Konformitätsprüfung wird bei Bus et. al zunächst ein Dokument erstellt, in dem der Datenaustausch, die Best-Practices für die Modellierung, Export-Parameter, Nomenklatur und Klassifizierungen für alle Projektbeteiligten festgelegt werden. Zusätzlich werden darin Property Sets angegeben, mit denen die IFC-Eigenschaften für die Konformitätsprüfung erweitert werden müssen. Ein konform dazu erstelltes IFC-Modell wird in mehreren Schritten inklusive einer Transformation von IFC zu RDF (*Resource Description Framework*) und einem geometrischen und semantischen Pre-Processing an die Ontologie angeglichen. Die Richtlinien werden in zwei Schritten vom Fließtext zu SPARQL-Abfragen verarbeitet. Erst übersetzen Domäne-Experten die Richtlinien in semi-formale Regeln, dann übersetzen Computerwissenschaftler die semi-formalen Regeln in SPARQL-Abfragen, welche nur das Vokabular der Ontologie verwenden. So wird sichergestellt, dass Experten des Bauwesens die den Abfragen zugrunde liegenden Regeln und Randbedingungen verstehen können. Zum Schluss kann eine automatisierte Regelprüfung durchgeführt werden, die die Ergebnisse der Prüfung im BCF-Format ausgibt. (Bus et al. 2018)

Das Konzept wurde auf verschiedene französische Hochbau-Richtlinien erfolgreich angewendet. Zum Beispiel wurde die Zugänglichkeit zu Toiletten und die Brandschutzanforderungen bezogen auf die Gebäudehöhe erfolgreich mit der Methode geprüft. (Bus et al. 2018)

Natural Language Processing (NLP)

Unter *Natural Language Processing* versteht man Methoden und Algorithmen, die der Verarbeitung von Fließtexten in natürlicher Sprache dienen. Derartige Entwicklungen zählen zum Forschungsgebiet der künstlichen Intelligenz und werden in letzter Zeit vermehrt auch im Zusammenhang mit ACCC eingesetzt. (Preidel 2020)

Das Ziel bei der Anwendung von NLP-Methoden im ACCC ist es, die Bedeutung von Vorschriften in natürlicher Sprache zu erfassen und daraus automatisch computerinterpretierbare Regeln für die Weiterverarbeitung zu generieren. (Zhang und El-Gohary 2012)

Ein Beispiel ist der Ansatz von Zhang und El-Gohary 2012. Dabei werden pattern-matching-basierte Regeln in Kombination mit konfliktlösenden Regeln zur Extraktion der relevanten Informationen auf Fließtexte angewendet. Es werden syntaktische und semantische Muster in den Texten untersucht. Um weniger Muster untersuchen zu müssen, werden die Texte mit Phrasenstrukturgrammatik (PSG-)Tags versehen und

semantische Informationselemente getrennt. Zur Erkennung dieser Informationselemente wird Ontologie verwendet. Die Methodik wurde erfolgreich am International Building Code von 2009 erprobt, einer amerikanischen Hochbau-Richtlinie. (Zhang und El-Gohary 2012)

Ein weiteres Beispiel wurde von Xu und Cai 2021 für die unterirdische Versorgungsinfrastruktur entwickelt. Das Ziel des Ansatzes ist es, die Interpretation und Transformation der Richtlinien vom Text bis zu logischen Regelsätzen zu automatisieren. Dabei werden zunächst syntaktische und semantische Merkmale aus dem Text herausgearbeitet. Diese werden anschließend mit der Hilfe von zwei extra entwickelten Ontologien, einer Domänenspezifischen und einer Räumlichen, auf ihre semantischen Entsprechungen gemappt. Damit können semantische und logische Formalisierungen über deontische Logikklauseln erzeugt werden. Der Ansatz wurde erfolgreich an räumlichen Anforderungen aus der Richtlinie für die Unterbringung von Versorgungsleitungen erprobt. (Xu und Cai 2021)

Problematisch an NLP im Zusammenhang mit ACCC ist, dass die Regeln in natürlicher Sprache häufig unklar und widersprüchlich sind und daher nicht vollautomatisch ohne eine Prüfung durch einen Menschen weiterverarbeitet werden können (Sydora und Stroulia 2019).

Die Algorithmen der NLP-Ansätze können als Black-Box-Methoden betrachtet werden, die resultierenden Regeln sind hingegen für Nutzer verständlich und daher White-Box-Methoden, was den Black-Box-Charakter aufweicht.

Rule-Checking Languages

Einige Ansätze beschäftigen sich mit eigens kreierten, domänenspezifischen Programmier- bzw. Abfragesprachen. Dazu gehören unter anderen die *Building Environment Rule and Analysis Language* (BERA) von Lee 2011 und die *BIM Rule Language* (BIMRL) von Solihin 2016.

BERA

Mit der Regelprüfungssprache BERA sollen Regeln für die Prüfung von BIM-Modellen einfach erstellt, analysiert und angewendet werden können (Lee 2011). Die Ziele bei der Entwicklung von BERA waren eine einfache Bedienbarkeit für Anwender aus dem

AEC-Bereich ohne spezifische Programmierkenntnisse, eine hohe Problemlösungskompetenz, die Erweiterbarkeit der Sprache und die Übertragbarkeit auf unterschiedliche BIM-Anwendungen (Lee 2011).

Dafür wird das abstrahierte BERA Object Model (BOM) verwendet, mit welchem die für die Modellprüfung benötigten Informationen, die aus IFC-Modellen stammen, so vorgehalten werden, dass sie mithilfe der BERA-Syntax einfach abgerufen werden können (Lee 2011).

Im Rahmen der Entwicklung wurde BERA auf den SMC aufgesetzt und mit Hochbau-Richtlinien zu Gebäudezirkulation und Raumprogrammen getestet (Lee 2011). Dabei zeigt BERA ein Potential für ACCC, auch wenn der Fokus bisher nur auf räumlich orientierten Richtlinien liegt und für komplexere Vorgaben noch weitere Entwicklungsarbeit geleistet werden muss (Preidel 2020).

BIMRL

Als domänenspezifische Abfragesprache auf Basis eines relationalen Datenbankschemas ermöglicht BIMRL schnelle, flexible Abfragen sowohl für geometrische also auch für räumliche und graphen-basierte Informationen (Solihin 2016).

Die sehr komplexen IFC-Modelle werden in BIMRL auf ein vereinfachtes Schema übertragen. Die Geometrie des ursprünglichen Modells wird dabei mit mehreren geometrischen Repräsentationen (Brep, Polyhedron, Octree, Oriented Bounding Box, Axis-Aligned Bounding Box) gespeichert, was einen schnellen Zugriff auf die für unterschiedliche Abfragen benötigten geometrischen Daten erlaubt. (Solihin et al. 2017)

Des Weiteren soll BIMRL als erweiterbare Regeldefinitionssprache dienen und die Speicherung sowie den Austausch von Regeln ermöglichen (Solihin 2016).

Solihin 2016 wendet die BIMRL in seiner Dissertation auf vier Beispiele aus dem Hochbau an, wobei sich eine breite Anwendbarkeit der Sprache zeigt (Solihin 2016). Für eine Anwendung in der Praxis ist jedoch noch weitere Entwicklungsarbeit nötig.

Visual Programming Languages

Preidel 2020 stellt einen Ansatz des ACCC mit einer visuellen Programmiersprache, der *Visual Code Checking Language* (VCCL), vor. Diese Sprache soll es Fachleuten des Bauwesens, die über keine fundierten Programmierkenntnisse verfügen, ermögli-

chen einfach und verständlich Regeln für die Konformitätsprüfung zu erstellen und anzuwenden. Die Methode stellt somit einen *White-Box-Ansatz* dar, da der Anwender die einzelnen Schritte der Prüfung sehen und dadurch genau nachvollziehen kann.

Den Kern der Sprache bilden die sogenannten atomaren Methodenknoten, welche Operationen beschreiben. Bei der Erstellung eines Programms mit VCCL werden vom Nutzer eingegebene oder aus einem BIM-Modell abgeleitete Daten (Input) mit den gewünschten Methodenknoten verbunden und innerhalb der Knoten verarbeitet. Das Endergebnis der Verarbeitung wird anschließend als Output vom Programm ausgegeben. Die Bibliothek der atomaren Methodenknoten kann erweitert werden und gespeicherte Programme können miteinander verschachtelt werden. Der Ansatz ist dadurch sehr vielversprechend. Er wurde an Beispielen aus dem Brandschutz im Hochbau erprobt. (Preidel 2020)

Die VCCL kann neben der Formulierung von computer-interpretierbaren Regeln auch für das Pre-Processing von BIM-Modellen für die Konformitätsprüfung verwendet werden. Das Ziel dabei ist die Ableitung der für eine bestimmte Prüfregel benötigten Informationen aus dem IFC-Modell. Neben dem IFC-Modell werden die nötigen Informationen anschließend als Input-Parameter in das VCCL-Programm übergeben. Die Ergebnisse des Pre-Processing können dann in das Modell als neue IFC-Objekte überführt werden, welches danach mittels MVD-Spezifikationen getestet werden kann. (Preidel et al. 2018)

3.4 Aufbereitung von BIM-Modellen

Die Aufbereitung von BIM-Modellen für die Konformitätsprüfung ist von zahlreichen Herausforderungen geprägt. Diese betreffen sowohl den Inhalt des Modells als auch das Datenaustauschformat.

3.4.1 Inhaltliche Herausforderungen und Lösungsvorschläge

Eine große Problematik ist die starke Abhängigkeit von der Qualität der zu prüfenden Modelle. Die BIM-Modelle müssen einen geeigneten Detaillierungsgrad sowie einen möglichst fehlerfreien und vollständigen Informationsgehalt enthalten, um zuverlässige Ergebnisse bei der Konformitätsprüfung zu erzeugen (Sacks et al. 2018). Daraus resultieren strikte Anforderungen an die Modelle, welche von den BIM-Autoren umge-

setzt werden müssen (Nawari 2012). Die Überprüfung dieser Anforderungen kann beispielsweise mit der IDM/MVD-Methode erfolgen, welche unter Kapitel 4.2 erläutert wird.

Auch die Abwägung zwischen einer aufwändigen Modellerstellung mit einer großen Sorgfaltspflicht des BIM-Autors und einer aufwändigen Implementierung mit starken Inferenz-Fähigkeiten des Prüfprogramms ist eine ständige Fragestellung bei ACCC (Eastman et al. 2009). Die Erstellung von Regeln sollte daher Hand in Hand mit der Erstellung von Modellanforderungen angegangen werden. Beispielsweise könnte ein zusätzlicher Parameter im Modell eine aufwändige Regel deutlich vereinfachen.

Hinzu kommt, dass unter Umständen nicht alle für die Konformitätsprüfung benötigten Informationen explizit ins Modell eingepflegt werden (können). Sie müssen erst in einem vorgeschalteten Pre-Processing des Modells abgeleitet und/oder berechnet und ins Modell zurückgegeben werden. In der Regel ist für ACCC daher eine Anreicherung des Modells mit semantischen Informationen nötig. (Sacks et al. 2018)

Bloch 2022 definiert die *semantische Anreicherung von BIM-Modellen* als Prozess, in dem automatisiert zusätzliche Informationen aus einem bestehenden BIM-Modell abgeleitet und explizit dargestellt werden, welche anschließend von einer Ziel-Anwendung verwendet werden können. Sie stellt bei ihrer Recherche fest, dass in existierenden ACCC-Anwendungen viele der benötigten semantischen Informationen aufwändig manuell durch den Nutzer in der ACCC-Anwendung eingegeben werden müssen, wodurch sich ein großes Potential für eine Automatisierung der semantischen Anreicherung ergibt. Eine automatisierte Anreicherung ist jedoch nur effektiv, wenn die ursprünglichen Modellobjekte ausreichend charakteristische Merkmale aufweisen, um automatisch erkannt und kategorisiert zu werden. (Bloch 2022)

3.4.2 Herausforderungen durch IFC

BIM-Modelle werden im Regelfall im IFC-Format ausgetauscht, da dieses die Interoperabilität zwischen verschiedenen Softwares ermöglicht, aber gleichzeitig die Bedeutung der modellierten Objekte überträgt. Wegen seiner Standardisierung, Eindeutigkeit und Konsistenz ist IFC für ACCC ein wichtiges Datenformat. (Nawari 2018)

Andere mögliche Datenformate, z.B. der deutsche *Objektkatalog für das Straßen- und Verkehrswesen* (OKSTRA) oder das reine Geometrie-Format STL (*Standard Tessellation Language*) werden im Rahmen der Arbeit nicht untersucht, da hier nur die Möglichkeiten von IFC4x3 evaluiert werden.

Das vollständige Datenmodell von IFC ist sehr groß und stark strukturiert, was die Performance einer Modellprüfung anhand eines IFC-Modells verschlechtert (Solihin et al. 2017). Für die Umsetzung von einzelnen Use Cases werden jedoch nur Ausschnitte des IFC-Datenmodells benötigt (Windisch et al. 2012). Die Effizienz der Modellprüfung kann beispielsweise mit einer *Model View Definition* verbessert werden, mit der nur der für die Konformitätsprüfung relevante Teil des Datenmodells abgebildet wird (Nawari 2018). Windisch et al. 2012 stellen eine alternative Methodik zum Erstellen von anwendungsspezifischen Model Views vor. Mit ihrem *Model Filtering Framework* soll die Erstellung von Model Views und den zugehörigen Modellabfragen harmonisiert werden. (Windisch et al. 2012). Die Methodik ist nicht im Zusammenhang mit ACCC entwickelt worden, könnte aber durchaus eine Hilfestellung für die Aufbereitung von BIM-Modellen geben.

Bei einigen Ansätzen wird das IFC-Datenmodell alternativ auf einfachere interne Strukturen abgebildet, um den Prüfprozess zu beschleunigen, z.B. bei BIMRL und BERA (Solihin et al. 2017).

Für die Anwendung des Semantic Web im ACCC ist es grundsätzlich notwendig die IFC-Modelle in ifcOWL zu überführen (Bus et al. 2018).

3.5 Durchführung der Überprüfung

3.5.1 Allgemeines

Nachdem die Regeln formuliert und die Modelle vorbereitet sind kann die Konformitätsprüfung durchgeführt werden.

Beim ACCC werden meist geometrische und räumliche Vorgaben überprüft, weshalb ACCC-Programme in der Lage sein müssen, geometrische Daten zu berechnen und auszuwerten. (Solihin et al. 2017)

Die Durchführung kann auf zwei Arten erfolgen, als Plug-In in der Authoring-Software des Modells oder am herstellernerneutralen IFC-Modell in externer Model-Checking-Soft-

ware. Das Plug-In bietet dabei den Vorteil, dass ein sofortiges Feedback der Modellprüfung an den BIM-Autor direkt in seinem Modell übergeben wird, was die Korrektur des Modells vereinfacht. Ein Report der externen Software, zum Beispiel in Form von BCFs, muss für die Modellkorrektur zunächst wieder in die Authoring-Software importiert und von dieser interpretiert werden. (Sacks et al. 2018)

3.5.2 Beispiele für eine technische Umsetzung

Neben den bereits genannten Forschungsansätzen, welche bisher nur zu Validierungszwecken angewendet wurden, gibt es auch Beispiele für kommerzielle Anwendungen von ACCC. Einige davon werden nachfolgend kurz vorgestellt.

CORENET und FORNAX

Eine der ersten und bekanntesten Entwicklungen im ACCC stellt die Plattform CORENET aus Singapur dar, welche zur digitalen Einreichung von Bauplanungsdokumenten (Modul *e-Submission*) und zur Sammlung aller Projektinformationen (Modul *e-Info*) dient. Außerdem beinhaltet die Plattform ein Werkzeug zur Konformitätsprüfung von IFC-Modellen (Modul *e-Plan Check*), welches in der Lage ist, die Einhaltung der Hochbau-Vorschriften aus den Bereichen der Barrierefreiheit, des Brandschutzes, der Umweltmedizin sowie der Gebäudesteuerung zu überprüfen. (Eastman et al. 2009)

Eastman et al. 2009 geben an, dass die Aufbereitung des IFC-Modells für eine Konformitätsprüfung eine große Problematik für ACCC darstellt. Für CORENET wurde dafür die objektbasierte FORNAX-Bibliothek entwickelt, welche sowohl IFC-Erweiterungen als auch Regeldefinitionen enthält. Mit den darin definierten FORNAX-Objekten ist es möglich, benötigte Informationen aus semantischen Objekten im IFC-Datenschema zu extrahieren und Konformitätsprüfungen durchzuführen (Eastman et al. 2009).

Die ProgrammROUTINEN für die Konformitätsprüfung sind in den FORNAX-Objekten fest implementiert, weshalb es sich hier um einen Black-Box-Ansatz handelt. Für eine Anpassung oder Erweiterung der vorhandenen Regeln über die API sind fundierte Programmierkenntnisse sowie Zugriffsrechte nötig. (Preidel 2020)

Insgesamt konnten die Prüfverfahren der singapurischen Behörden durch CORENET deutlich beschleunigt werden und die Plattform wird von der Behörde und vielen Unternehmen erfolgreich eingesetzt (Eastman et al. 2009).

Solibri Model Checker (seit 2018 Solibri Office)

Der *Solibri Model Checker* (kurz: SMC) ist ein Software-Tool, das für die Überprüfung von BIM-Modellen entwickelt wurde, in der Praxis aber häufig auch für die Visualisierung von IFC-Modellen verwendet wird (Sacks et al. 2018).

Bei der Anwendung des SMC werden IFC-Modelle gelesen und auf eine interne Struktur, die an den IFC-Standard angelehnt ist, gemappt (Eastman et al. 2009). Für Modellprüfungen werden häufig mehrere Teilmodelle aus verschiedenen Autoren-Softwares zusammengefügt, was zu Inkonsistenzen führen kann. Dem wirkt der SMC entgegen, indem die Prüfregeln auf ein IFC-Projekt-Modell angewendet werden. (Sacks et al. 2018)

Im Anschluss können Regeln aus unterschiedlich komplexen, vordefinierten Regelvorlagen (sog. Rule Sets) generiert und auf das Modell angewendet werden. Die Regelvorlagen sind fest implementiert, können aber bei der Erstellung von Regelsätzen frei kombiniert und mittels der angegebenen Parameter angepasst werden. Die Vorlagen reichen von einfachen Attributprüfungen über Clash Detection und Prüfungen zur Barrierefreiheit bis hin zur Berechnung von Entfluchtungsrouten. (Preidel 2020)

Über die API der Software wäre es möglich die Regelvorlagen als Nutzer mit Programmierkenntnissen auf eigene Bedürfnisse und weitere Richtlinien anzupassen, diese ist aber nicht öffentlich zugänglich (Sydora und Stroulia 2019).

Durch die feste Implementierung der Regelvorlagen liegt beim SMC ebenfalls ein *Black-Box-Ansatz* vor, welcher allerdings durch zwei Faktoren aufgeweicht wird: zum einen ergibt sich durch die freie Kombination von Regelvorlagen durch den Nutzer ein gewisser Grad an Transparenz hinsichtlich des Prüfprozesses, zum anderen werden die bei der Prüfung festgestellten Probleme genau erläutert, was das Ergebnis der Prüfung sehr verständlich an den Nutzer weitergibt (Preidel 2020).

Für Konformitätsprüfungen, die über die im SMC enthaltenen Richtlinien hinausgehen, muss das Regelwissen vom Anwender selbst aus der Richtlinie abgeleitet und in SMC-Rule Sets übertragen werden.

DESITE md pro

Ein weiteres Beispiel für Model-Checking-Software ist DESITE md pro. Die Software DESITE in der Variante md pro von Thinkproject kann für die Visualisierung, Erweiterung und Analyse von BIM-Modellen verwendet werden. Sie erlaubt 4D-Simulationen, Informationsmanagement, Mengenermittlung und Modellprüfungen inklusive Kollisionprüfungen sowie benutzerspezifische Filterung von BIM-Modellen sowohl im IFC-Format als auch in herstellerspezifischen Formaten. (Thinkproject 2022)

Die regelbasierte Modellprüfung kann entweder mit vorbereiteten Attributprüfungen über das User Interface erfolgen oder über eine Java-Script-basierte API-Programmierung (Mensch+Maschine 2023). Grundsätzlich fällt Desite als etablierte Model-Checking-Software in eine ähnliche Kategorie wie der SMC. Die offen zugängliche API ist jedoch ein großer Vorteil gegenüber dem SMC. DESITE md pro ist außerdem in der Lage BIM-Modelle in IFC4.1 zu importieren (Thinkproject 2022). Es wird daher erwartet, dass die Software mit Modellen umgehen kann, die entlang einer Streckenachse erstellt wurden, da das Konzept *Alignment* bereits in IFC4.1 implementiert ist (vgl. Kapitel 2.3.8). Dafür spricht auch, dass die Software bereits in einer Forschungsarbeit zur Qualitätsprüfung von BIM-Modellen im Bahnbau von Häußler und Borrmann 2020 angewendet wurde.

Autodesk Revit Model Review

Mit dem Revit-Plug-In *Autodesk Revit Model Review* können Revit-Modelle darauf geprüft werden, ob sie hinsichtlich Beschaffenheit und Genauigkeit den Projekt-, Industrie- oder nutzerspezifischen Standards entsprechen. Die einzelnen Prüfungen können jederzeit passend mittels Vorlagen konfiguriert werden und die Ergebnisse können für einen Report im HTML-Format oder direkt zur manuellen Korrektur des Modells verwendet werden. In einigen Fällen kann die Korrektur des Modells auf Wunsch des Nutzers automatisch erfolgen. (Autodesk 2023d)

Mit dem Plug-In können unter anderem Parameter und ihre Werte geprüft werden, wobei bisher nur die englische Schreibweise von Parametern und Zahlen unterstützt werden. (Autodesk 2023d)

Auch hier müssen für eine Konformitätsprüfung selbst Regeln entsprechend der Richtlinien erstellt werden. Zudem sind nur sehr limitierte Prüfungen möglich, z.B. Attributprüfungen und vereinzelt Prüfungen zur Sichtbarkeit von Modellelementen und dem Zusammenhang zwischen Raumobjekten und deren Umschließung. (Autodesk 2023d)

3.6 Erkenntnisse für die Umsetzung von ACCC im Brückenbau

Aus den untersuchten Ansätzen, die vorwiegend an Hochbauregeln und -modellen erprobt sind, können einige Erkenntnisse für eine Umsetzung von ACCC im Brückenbau festgehalten werden.

Für eine fehlerfreie Modellprüfung müssen die Modelle explizite, korrekte und vollständige Informationen enthalten. Dazu gehören viele Informationen, die ein menschlicher Betrachter des Modells implizit erkennen kann, eine Model-Checking-Software jedoch nicht oder nur mit einem sehr aufwändigen Pre-Processing der Modelle. Das gilt beispielsweise für Räume, die im Modell implizit als Hohlraum vorliegen, für die Modellprüfung aber explizit definiert sein müssen. (Sacks et al. 2018)

Demnach muss im Brückenbau beispielsweise der Hohlraum innerhalb eines Hohlkasten-Querschnitts oder der Verkehrsraum auf einer Straße oder Kappe für die automatisierte Konformitätsprüfung explizit modelliert werden.

Um Modelle mit einer entsprechenden Qualität zu erstellen, ist es sinnvoll neben Regeln auch Modellanforderungen aus den Richtlinien abzuleiten. Dabei ist zu beachten, dass die Anforderungen an ein Gebäudemodell stark von den abgeleiteten Regeln aus den zu prüfenden Richtlinien abhängen und umgekehrt. Für jede Regel muss abgewogen werden, ob die Information, z.B. eine Bauteilabmessung, vom BIM-Autor explizit ins Modell geschrieben werden soll oder von der Prüfsoftware aus der Geometrie im Modell abgeleitet wird.

Wenn die so ermittelten Modellanforderungen eingehalten sind, kann sichergestellt werden, dass die benötigten expliziten Informationen tatsächlich explizit im Modell abgebildet sind. Um die Einhaltung der Modellanforderungen zu prüfen, sollte immer eine Vorprüfung des Modells gemacht werden bevor die eigentliche Konformitätsprüfung erfolgt. Bei der Verwendung von IFC als Datenaustauschformat ist es sinnvoll, das umfangreiche Datenmodell mit einer Model View auf das nötigste zu begrenzen. Diese Model View enthält dann alle Informationen, die zur Erfüllung der Modellanforderungen nötig sind.

3.7 Zusammenfassung

Für die Zukunft erhofft man sich für alle Bauprojekte, dass mittels automatisierten Modellprüfungen Planungsfehler und Abweichungen von Richtlinien frühzeitig erkannt werden können, sodass Fehler in der darauffolgenden Ausführungsphase vermieden werden können (Sydora und Stroulia 2019). Gleichzeitig sollen so der Arbeitsaufwand und die Kosten für die Konformitätsprüfung reduziert werden und die monotonen, iterativen Arbeitsschritte des Prozesses an den Computer übertragen werden (Preidel 2020).

Auch im Infrastrukturbau wird inzwischen vermehrt mit der BIM-Methode gearbeitet. Mit IFC4x3 steht nun ein herstellerneutrales Format zu Verfügung, welches unter anderem den Modellaustausch von Brückenmodellen deutlich verbessert. Dies ist die Grundvoraussetzung für eine interdisziplinäre Zusammenarbeit und eine digitale Modellprüfung von aus mehreren Fachdisziplinen zusammengesetzten Modellen.

Aus der Literaturrecherche geht hervor, dass der Fokus der Forschung und Entwicklung im Bereich ACCC bisher auf dem Hochbau liegt, insbesondere auf den Richtlinien zu Barrierefreiheit, Brandschutz und technischer Gebäudeausrüstung. Die Mehrheit der untersuchten Ansätze beschäftigt sich mit der Übersetzung von Regelwerken in computer-interpretierbare Form. Während bisher etablierte Ansätze wie der SMC und CORENET auf fest-implementierten Programmroutinen beruhen, sind neuere Entwicklungen wie SMARTcodes, BERA, BIMRL oder die VCCL mehr darum bemüht, den Anwender in die Regelerstellung miteinzubeziehen, um die Transparenz der Prüfprozesse zu erhöhen, was das Vertrauen in die Prüfergebnisse signifikant verbessert. Auch anderweitig bereits angewendete Methoden wie das Semantic Web oder NLP werden für die Aufbereitung von Regelwissen in Betracht gezogen, sind aber im Bereich ACCC noch wenig praktisch erforscht.

Für den Infrastrukturbau, der sich bezüglich der Modellierung deutlich vom Hochbau unterscheidet, wurden die untersuchten Ansätze bisher kaum erprobt. Aus den Untersuchungen zum Hochbau lassen sich jedoch Rückschlüsse bezüglich der Regelerstellung und Modellanforderungen für eine Anwendung von ACCC im Infrastrukturbau ziehen.

Über für die Konformitätsprüfung benötigte Modellanforderungen wurde bisher wenig veröffentlicht. Dabei ist die Zugänglichkeit zu den nötigen Informationen mit Hilfe von BIM-Modellen eine der größten Herausforderungen des ACCC. Wenn die Modelle die

Anforderungen nicht erfüllen und somit die nötigen Informationen nicht enthalten, ist eine Konformitätsprüfung nicht möglich. Die vorliegende Arbeit leistet hier einen Beitrag, indem IFC-basierte Modellanforderungen für die Konformitätsprüfung von Brückenentwürfen erstellt werden.

Die Literaturrecherche ergibt außerdem, dass die Formulierung der Regeln für die Konformitätsprüfung durch Fachleute des Bauwesens erfolgen muss. Auch eine Übersetzung in zwei Stufen ist denkbar, wobei die Baufachleute in der ersten Stufe semi-formale Regeln aus den Richtlinien ableiten, welche in der zweiten Stufe durch Computerwissenschaftler in Code übersetzt werden (Bus et al. 2018). Für Brückenentwürfe nach deutschen Richtlinien wurde diesbezüglich noch nichts veröffentlicht. In dieser Arbeit werden daher semi-formale Regeln aus den Richtlinien für Brückenentwürfe erstellt und für die Implementierung kategorisiert.

Aus den vorgenannten Entwicklungen zeigt sich darüber hinaus, dass eine vollständig automatisierte Erstellung von Regelsätzen für die Konformitätsprüfung nicht möglich ist, was auch an der Komplexität und der teils widersprüchlichen und unklaren Formulierung der Vorschriften liegt. Beispielsweise verlangt RE-Ing Teil 2 Abschnitt 4 Paragraph 1.1 (2) eine schadlose Wasserabführung. Ein Ingenieur weiß, wie das ermöglicht werden kann, ein Model-Checking-Programm jedoch nur bei sorgfältiger manueller Integration des nötigen Fachwissens durch den Ingenieur bei der Regelerstellung. In dieser Arbeit wird daher auf eine solche Automatisierung verzichtet und die exemplarischen Regelsätze werden stattdessen manuell erstellt.

4 Informationsanforderungen für die Konformitätsprüfung

In diesem Kapitel werden Informationsanforderungen für eine Konformitätsprüfung von BIM-Modellen von Straßenbrücken erarbeitet. Dabei wird zunächst ein Überblick über die Begriffe und den Stand der Technik bei der Formulierung von Modellanforderungen (engl. *Information Requirements*, kurz: IRs) gegeben und eine geeignete Methode für diese Masterarbeit festgelegt. Dann werden die geltenden Richtlinien für Straßenbrücken vorgestellt. Anschließend werden Austauschforderungen für Brückenmodelle zusammengestellt, welche zum Schluss mit dem Datenmodell von IFC4x3 verglichen werden.

4.1 Begriffserklärung Informationsanforderungen

Nach DIN EN ISO 19650-1 definiert sich *Informationsanforderung* als „Festlegung für was, wann, wie und für wen Informationen erstellt werden sollen“ (DIN 2019).

In dieser Norm werden verschiedene Arten von Informationsanforderungen und Informationsmodellen beschrieben, welche miteinander in einem engen Zusammenhang stehen (vgl. Abbildung 8).

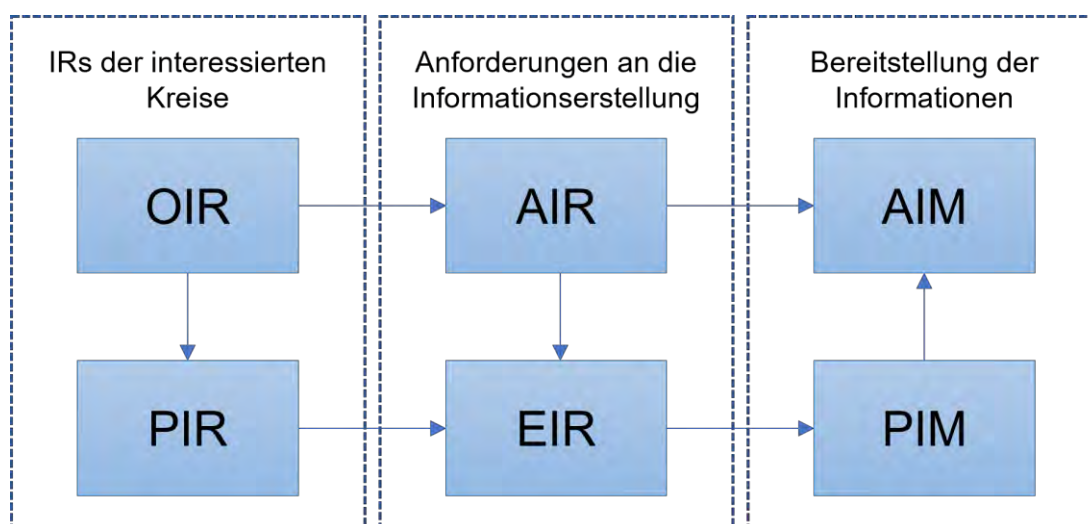


Abbildung 8: Zusammenhang zwischen verschiedenen Informationsanforderungen und -modellen (Quelle: Eigene Darstellung, angelehnt an DIN EN ISO 19650-1 Bild 2)

Zu Projektbeginn werden organisatorische Informationsanforderungen (OIR) festgelegt, welche zum Erreichen der strategischen Ziele des Auftraggebers benötigt werden. Diese werden im Anschluss durch AIRs und PIRs ergänzt. Die Asset-Informationsanforderungen (AIR) geben betriebswirtschaftliche und technische Aspekte für die

Erstellung von Informationen für das Bauwerk an. Mit den Projekt-Informationsanforderungen (PIR) werden die OIRs für ein bestimmtes Bauvorhaben konkretisiert. Aus den AIRs und den PIRs werden Austausch-Informationsanforderungen (engl. *Exchange Information Requirements*, kurz: EIRs) entwickelt. Mit Hilfe der EIRs wird sichergestellt, dass alle Projektanforderungen erfüllt werden. In den EIRs werden dabei detaillierte Informationsanforderungen für die einzelnen (Leistungs-)Phasen des Projekts festgelegt. Während des Projekts wird ein PIM (Projektinformationsmodell) erstellt, das im Laufe der Planung immer mehr Informationen enthält und jeweils am Ende einer Projektphase den zugehörigen EIRs entspricht. Es enthält zum Beispiel Bauteile, Kosten und Terminpläne. Nach Abschluss des Projekts wird auf der Grundlage des letzten PIM das AIM (Asset-Informationsmodell) gebildet. Das AIM beschreibt das Ergebnis des Projektabwicklungsprozesses und erfüllt die AIRs. Es enthält die für die Betriebsphase eines Bauwerks notwendigen Informationen und kann beispielsweise für die Verwaltung von Wartungsterminen der Haustechnik verwendet werden. Bei Instandsetzungsmaßnahmen können die Informationen des AIM als Planungsgrundlage verwendet werden. (DIN 2019)

In den IRs für die Konformitätsprüfung muss angegeben werden welche Informationen in welchem Detaillierungsgrad im Modell vorhanden sein müssen. Da die Konformitätsprüfung i.d.R. als Abschluss der Entwurfsphase durchgeführt wird, fallen diese IRs somit unter EIRs. Zur Beschreibung von IRs gibt es verschiedene Möglichkeiten, welche im folgenden Abschnitt erläutert werden.

4.2 Stand der Technik IRs

Damit ein Bauwerksmodell eine gute Informationsgrundlage für Projektentscheidungen abgibt, sind gut verwaltete und strukturierte Modellanforderungen hilfreich. In der Praxis stehen dafür zahlreiche Methoden zur Verfügung, welche teilweise standardisiert sind. (Tomczak et al. 2022)

Einige davon werden nachfolgend zusammengestellt. Für einen detaillierten Vergleich der Methoden sei hier auf Tomczak et al. 2022 verwiesen.

Textbasierte und Tabellenbasierte Methoden (DOC & XLS)

Aktuell wird in der Praxis am häufigsten eine *textbasierte Beschreibung* (kurz: DOC) von Modellanforderungen verwendet. Dabei werden die IRs mit üblichen Textverarbeitungsprogrammen formuliert und zum Austausch in PDF konvertiert. Teilweise wird

auch ein *tabellenbasiertes Vorgehen* (kurz: XLS) angewendet, welches sich etwas besser für eine Computerverarbeitung eignet. Der Vorteil einer solchen Vorgehensweise liegt darin, dass die Nutzer bereits mit der Software vertraut sind. Auch die große Flexibilität bei der Formulierung als Fließtext oder Tabelle ist ein Grund für die Verbreitung dieser Vorgehensweisen. (Tomczak et al. 2022)

Problematisch bei dieser Methodik ist, dass die Vorlagen der Text- bzw. Tabellendokumente nur bei den Autoren verwaltet werden und nicht standardisiert sind (Tomczak et al. 2022). Ein Vergleich von Anforderungen im Fließtext mit einem IFC-Modell ist nicht automatisiert möglich und erfordert eine große Sorgfalt und Sachkenntnis von der prüfenden Person. Jedoch ist jede einzelne Modellanforderung für die BIM-Autoren und die prüfenden Personen nachvollziehbar, auch wenn diese (noch) nicht über eine entsprechende Fachkenntnis verfügt.

Die Vor- und Nachteile der textbasierten Anforderungen gelten auch für die tabellarischen Anforderungen. Die tabellenbasierten Anforderungen können jedoch einfacher in eine computerbasierte Modellprüfung überführt werden.

Product Data Templates (PDT)

PDT wurde ursprünglich als Vergleichsmöglichkeit für Bauprodukte in der Ausschreibungsphase von Bauprojekten eingeführt (Tomczak et al. 2022). Das Ziel bei PDT sind standardisiert formatierte Bauproduktdateien, welche zwischen verschiedenen Software-Tools ausgetauscht werden können (Abdulkareem und Gherghiceanu 2022). Hierbei wird ein tabellarischer Satz Produkteigenschaften standardisiert und als Vorlage verwendet. Es besteht die Möglichkeit ein PDT aus einer öffentlichen Sammlung zu verwenden oder ein projektspezifisches PDT zu erstellen. Die vom Produkthersteller ausgefüllte Vorlage wird dann als *Product Data Sheet* bezeichnet und kann in BIM-Modelle integriert werden. (Tomczak et al. 2022)

In DIN EN ISO 23386 wird festgelegt, wie Eigenschaften in Datenkatalogen beschrieben, erstellt und gepflegt werden sollen (DIN 2020). Somit ist die Erstellung eines PDT zumindest zum Teil standardisiert (Tomczak et al. 2022). Durch die Vorgabe benötigter Produktdaten können Modellanforderungen über PDT ausgedrückt werden.

Information Delivery Manual (IDM)

In einem *Information Delivery Manual* (IDM) wird beschrieben, „*welche* Informationen von *wem wann welchem* Projektbeteiligten *wie* zur Verfügung gestellt werden sollen“

(Beetz et al. 2021a). Somit können konkrete Anforderungen an den Inhalt und die Qualität von Modellen festgehalten werden. Die Methodik wurde von bSI entwickelt und wird in der DIN EN ISO 29481-1 standardisiert (buildingSMART Technical 2023a). Ein IDM besteht demnach aus einem Interaktionsplan und/oder einem Prozess-Diagramm sowie einer oder mehrerer Informationsaustausch-Anforderungen. Die Informationsaustausch-Anforderungen werden in Informationseinheiten untergliedert, welche jeweils eine klare Beschreibung der Einheit und deren Erfordernisse, u.a. Datentypen und andere Informationsbeschränkungen, beinhalten. (DIN 2018)

Da der Ausgangspunkt bei der Entwicklung eines IDM der betroffene BIM-Use Case ist, stellt bSI den *Use Case Management Service* unter anderen als Grundlage für die Erstellung von Informationsanforderungen zur Verfügung. Darin werden aus realen Bauprojekten für unterschiedlichste Use Cases für die relevanten Projektphasen Informationsanforderungen an die verschiedenen Akteure zusammengetragen. (buildingSMART International 2023)

Für die Implementierung von IDMs in Software werden *Model View Definitions* (MVDs) verwendet. In der MVD wird eine Teilmenge eines bestehenden Datenmodells, i.d.R. IFC, vorgegeben, mit welcher die IRs aus dem jeweiligen IDM erfüllt werden können (DIN 2018). Das Problem dabei ist, dass auf diese Weise viele MVDs für unterschiedliche Anwendungsfälle erstellt und in BIM-Software implementiert wurden, die nicht miteinander kompatibel sind (buildingSMART Technical 2022b). Inzwischen gibt bSI deswegen für die jeweilige IFC-Version eine limitierte Anzahl sogenannter „Basis-MVDs“ vor, zum Beispiel die *Alignment Based Reference View* für IFC4x3 (buildingSMART International 2021a). Aufbauend auf den Basis-MVDs können dann IRs implementiert werden (buildingSMART Technical 2022b). In der Praxis werden MVDs nur zur Zertifizierung von Software-Implementierungen genutzt und nicht zur projektspezifischen Umsetzung von IDMs (Tomczak et al. 2022). Als Alternative zur IDM/MVD-Methode wird von bSI der weiter unten beschriebene IDS-Standard entwickelt (buildingSMART Technical 2022b).

Im Regelfall werden die IRs eines IDM als Text oder Tabelle zusammengestellt. Aber mit dem aktuell entwickelten Teil 3 des Standards soll eine standardisierte Formulierung von IRs als idmXML ermöglicht werden (Tomczak et al. 2022). Die Extensible Markup Language (XML) des World Wide Web Konsortiums (W3C) ist eine Auszeichnungssprache für Textdokumente. Eine XML-basierte Formulierung hat den Vorteil,

dass die so beschriebenen IRs von Menschen und Maschinen gelesen werden können. (Artus et al. 2021)

Information Delivery Specifications (IDS)

IDS ist ein XML-basierter Standard von bSI zur menschen- und maschineninterpretierbaren Formulierung von IRs. Mit IDS können Eigenschaften mit zulässigen Werten und Einheiten vorgegeben werden. IDS steht in enger Verbindung mit dem IFC-Datenschema und ist daher besonders für IFC-Modelle geeignet. Außerdem ermöglicht IDS die Einbindung des bSDD in die IRs durch Links. (van Berlo et al. 2023)

Eine IDS-Spezifikation gliedert sich in Anwendbarkeit (*Applicability*) und Anforderungen (*Requirements*). Durch die *Applicability* werden die Elemente herausgefiltert, die vom jeweiligen IR betroffen sind. Durch die *Requirements* können Materialien, Eigenschaften, Entities, Typen oder Klassifizierung der Elemente analog zum IFC-Datenschema gefordert werden. (Tomczak et al. 2022)

Zum aktuellen Zeitpunkt ist eine Definition von geometrischen Details mit IDS nicht möglich. (van Berlo et al. 2023)

Der *Use Case Management Service* von bSI kann auch bei der Erstellung einer IDS als Grundlage eingesetzt werden (buildingSMART Technical 2023b).

Data Dictionary (DD)

Ein Data Dictionary (dt. Datenkatalog) ist eine „Datenbank, die Metadaten enthält“ (DIN 2022). Im Zusammenhang mit IRs können in einem Data Dictionary Objekte mit den gewünschten Beziehungen, Eigenschaften und Klassifizierungen sowie Prozesse strukturiert abgelegt werden (Tomczak et al. 2022).

Ein Beispiel einer Implementierung ist das *buildingSMART Data Dictionary*. Es stellt verschiedene Klassifizierungen und Eigenschaften von Objekten gemäß unterschiedlichen Standards zur Verfügung. Diese sind zum Beispiel das IFC-Datenmodell, internationale Klassifizierungssysteme wie Uniclass, anwendungsspezifische Systeme wie ETIM oder sogar projektspezifische Standards. Die Daten aus dem bSDD können von BIM-Anwendern verwendet werden, um ihre Modelle mit zusätzlichen Informationen anzureichern. Das geschieht in der Regel über Verknüpfungen von BIM-Software zur bSDD-API. (buildingSMART International 2022e)

Mit einem spezifizierten Standard in einem Data Dictionary, der über eine API-Schnittstelle mit der entsprechenden Authoring- und Model-Checking-Software verknüpft ist,

können auf diese Weise computerinterpretierbare Anforderungen an die einzelnen Objekte gestellt werden.

Level of Information Need (LOIN)

Der LOIN (dt. Informationsbedarfstiefe) wird in der DIN EN 17412-1 beschrieben. Mit dem LOIN werden Modellanforderungen für einen bestimmten Rahmen angegeben, nicht für spezifische Stufen wie beim LOD (Tomczak et al. 2022). Für diesen Rahmen müssen Anwendungsziele, Meilensteine der Informationsbereitstellung, betroffene Akteure und Objekte innerhalb einer Gliederungsstruktur festgelegt werden (DIN 2021a).

Der LOIN umfasst geometrische und alphanumerische Informationen und die Dokumentation. Bei den geometrischen Informationen werden die Aspekte Detailierung, Dimensionalität, Lage, Darstellung und parametrisches Verhalten berücksichtigt. Für die alphanumerischen Informationen wird eine Identifikation des jeweiligen Objekts in der Gliederungsstruktur benötigt, zum Beispiel Name oder Klassifizierung, um dem Objekt einen Informationsgehalt in Form einer Liste erforderlicher Merkmale zuordnen zu können. Die Dokumentation zu den betroffenen Objekten wird als Satz erforderlicher Dokumente angegeben. (DIN 2021a)

Die so zusammengestellten Modellanforderungen sind strukturiert und menschenlesbar. Mit Teil 3 der DIN 17412, der noch in der Entwicklung ist, wird ein Datenschema für den LOIN erwartet (Tomczak et al. 2022). Mit diesem Datenschema sollen Modellanforderungen über den LOIN computerinterpretierbar formuliert werden können.

Auswahl einer Methode für die Zusammenstellung der Modellanforderungen

In dieser Arbeit werden die Modellanforderungen tabellarisch zusammengestellt, um sie menschenlesbar und intuitiv verständlich zu beschreiben. Für eine gute Vergleichbarkeit mit IFC4x3 werden die Anforderungen als konkrete benötigte Modellobjekte und Attribute angegeben. Diese werden nach Möglichkeit passenden IFC4x3-Entities und Properties zugeordnet. Damit ergibt sich ein auf IFC4x3 zugeschnittener Objekt- und Attributkatalog, mit dem eine Eignung von IFC4x3 für die Konformitätsprüfung übersichtlich evaluiert werden kann.

Nachdem die Methodik feststeht, können nun die betroffenen Richtlinien erst grob und dann in einzelnen Paragraphen hinsichtlich Modellanforderungen ausgewertet werden.

4.3 Richtlinien für Brückenentwürfe im Straßenbau

Für den Brückenbau der Bundesfernstraßen gelten in Deutschland verschiedene Richtlinien von der Bundesanstalt für Straßenwesen (BASt) und dem Bundesministerium für Digitales und Verkehr (BMDV). Diese werden in großen Teilen auf der Website der BASt veröffentlicht. Die für diese Arbeit relevanten Richtlinien werden im Folgenden kurz erläutert.

RAB-Ing

Die *Richtlinien für das Aufstellen von Bauwerksentwürfen für Ingenieurbauten* (RAB-Ing) enthalten formale Angaben zu den Entwurfsunterlagen. Sie geben zudem die für einen Entwurf und damit für das zugehörige BIM-Modell anzugebenden Informationen vor. Diese Informationen sind der RAB-Ing getrennt nach Grundriss und Schnitten zu entnehmen, ein Modell muss daher die geforderten Angaben für beides erfüllen. (BMDV 2022a)

RE-Ing

Für Bauwerksentwürfe von Brücken gilt Teil 2 der *Richtlinien für den Entwurf, die konstruktive Ausbildung und Ausstattung von Ingenieurbauten* (RE-Ing). Dort finden sich Planungsgrundsätze, konstruktive Anforderungen an die einzelnen Brückenteile wie Über- und Unterbauten sowie Anforderungen an die Brückenausstattung. (BMDV 2022b)

RiZ-Ing

Die RAB-Ing und die RE-Ing verweisen in mehreren Paragraphen auf die *Richtzeichnungen für Ingenieurbauten* (RiZ-Ing). Diese enthalten Baustoffangaben, geometrische Vorgaben, zum Beispiel Mindestabmessungen oder Mindestneigungen, sowie Ausführungsdetails (BASt 2022). Die grundlegenden geometrischen Vorgaben und die Baustoffe sind für Modelle relevant, die Details nur beschränkt, da sie im Regelfall nicht modelliert werden, sondern als 2D-Zeichnungen auf die Ausführungspläne ergänzt werden. Diese Details können in einem Modell über ein Bauteilattribut mit dem jeweiligen Bauteil verknüpft werden.

ZTV-Ing

Bei Ingenieurbauten gelten für die Ausführung grundsätzlich auch *Zusätzliche Technische Vertragsbedingungen und Richtlinien für Ingenieurbauten* (ZTV-Ing). Für den Ent-

wurf von Brücken relevant sind dabei vor allem die von der ZTV-Ing geforderten Mindestabmessungen für Betonbauteile (Teil 3 Abschnitt 2 – Bauliche Durchbildung) sowie die Angaben für die Bauwerksausstattung (Teil 6). Es werden auch anzugebende Informationen für die Verbauten genannt (Teil 2). Damit kann bei der Modellprüfung das Modell auf Vollständigkeit geprüft werden. (BMDV 2022c)

4.4 Entwicklung von Austauschforderungen für Modelle aus den Richtlinien

Nach der groben Betrachtung der einzelnen Richtlinien werden im nachfolgenden Kapitel die konkreten Modellanforderungen für eine Konformitätsprüfung von Brückentwürfen des Straßenbaus erarbeitet. Dabei wird zunächst der Detaillierungsgrad für die Modelle erarbeitet, dann der Rahmen der Masterarbeit eingegrenzt und dann das Vorgehen erst allgemein und dann an zwei konkreten Beispielen erläutert. Zum Schluss werden die Ergebnisse der Detailanalyse der Richtlinien vorgestellt.

4.4.1 Festlegung der Detaillierung der Modelle

Vor der Entwicklung von Austauschforderungen wird für den Use Case *Konformitätsprüfung* ein passender LoD bzw. LOD ermittelt. Da die Konformitätsprüfung beim Planer in Vorbereitung auf die Prüfung für die Baugenehmigung bei der Behörde geschieht, muss die Planung den Stand einer Genehmigungsplanung, also HOAI-Leistungsphase 4, aufweisen. Singer und Borrmann weisen in ihrer Machbarkeitsstudie *BIM für Bestandsbrücken* von 2016 der Genehmigungsplanung einen LoD 300 zu.

Die Anforderungen an diesen Detaillierungsgrad werden in der Machbarkeitsstudie folgendermaßen definiert:

Das Modellelement wird im Modell geometrisch als Objekt mit genauen Mengen, Größe, Form, als exakter Volumenkörper und mit der genauen Orientierung dargestellt. Die Anzahl, Größe, Form, Ort und Orientierung der Elemente kann direkt aus dem Modell gemessen werden, ohne auf andere Quellen zurückgreifen zu müssen. Semantische Informationen sind enthalten. (Singer und Borrmann 2016, S.48)

Für eine regelbasierte Konformitätsprüfung sind eine Geometrirepräsentation als exakter Volumenkörper mit korrekter Orientierung und das Abgreifen geometrischer Maße aus dem Modell ebenfalls nötig. Daher wird für die Modellanforderungen in dieser Masterarbeit ein LoD 300 festgelegt.

Die oben genannte Definition des LoD 300 deckt sich vollständig mit der weit verbreiteten Definition des LOD 300 von BIM Forum (BIM Forum 2020). Somit wird auch der LOD 300 für die Masterarbeit festgelegt.

4.4.2 Allgemeines Vorgehen

Festlegung des Rahmens für die Masterarbeit

Unter Berücksichtigung des LOD werden anschließend die Paragraphen der RAB-Ing, RiZ-Ing, RE-Ing und ZTV-Ing einzeln betrachtet. Im Rahmen der Masterarbeit wird sich dabei auf folgende Teile der genannten Regelwerke beschränkt:

- RAB-Ing Teil 4 Abschnitt 1 (Bauwerksplan Neubau)
- RE-Ing Teil 2 Abschnitt 1 bis 4
- RiZ-Ing: Abs, Bösch, Dicht, Flü, Fug, Gel, Jahr, Kap, Lag, Übe, Was, Zug
- ZTV-Ing Teil 2 Abschnitt 1 & 2, Teil 3 Abschnitt 2 & 3, Teil 6 Abschnitt 9 & 10

Da der Großteil der Anforderungen aus der ZTV-Ing die Ausführung betrifft, wurden viele Paragraphen im Rahmen der Analyse für die weniger detaillierte Entwurfsplanung nicht beachtet. Welche Paragraphen bzw. Richtzeichnungen jeweils zur Ableitung von Modellanforderungen betrachtet wurden, kann in den bearbeiteten Dokumenten im digitalen Anhang (Übersicht siehe Anhang B) eingesehen werden.

Die genannten Abschnitte stellen Anforderungen an die Entwurfsplanung von Neubauten von klassischen Massivbrücken, ausgenommen seilverspannte Brücken. Weiterhin werden nur Kreuzungen zwischen zwei Verkehrswegen und Kreuzungen zwischen einer Straße und einem Fließgewässer, ausgenommen Bundeswasserstraßen, betrachtet.

Analyse der einzelnen Paragraphen

Um die Modellanforderungen für die Konformitätsprüfung ermitteln zu können, muss zunächst erarbeitet werden, welche konkreten Informationen für die Regelprüfung benötigt werden. Dafür werden die Paragraphen der Richtlinien aufgeschlüsselt und in Regeln bzw. Modellanforderungen übersetzt.

Zunächst wird geprüft, ob der jeweilige Paragraph für die Entwurfsplanung von Bauwerken im gerade beschriebenen Rahmen der Masterarbeit relevant ist. Anschließend werden Anforderungen aus dem Paragraphen abgeleitet.

Im Rahmen der Arbeit wurden aus insgesamt 180 Paragrafen etwa 370 einzelne Anforderungen bzw. Regeln erarbeitet. Diese umfassen sowohl Modellinhalte wie Objekte und Attribute als auch explizite Vorgaben zur Geometrie der Bauteile, zu Mindestabständen, Anwendungsgrenzen von Richtzeichnungen, Zugänglichkeiten, Entwässerung und Verbauten.

Für jede dieser Anforderungen und Regeln wird überlegt, wie die benötigten Informationen im Modell enthalten sein sollen. Dafür werden drei Kategorien verwendet: *Objekt*, *Attribut* und *implizit*.

Zur Kategorie *Objekt* zählen die explizit modellierten Objekte im Modell. Ein Objekt kann in diesem Zusammenhang ein konkretes Bauteil der Brücke sein, beispielsweise ein Widerlager, oder ein Bestandteil des Bauwerksentwurfs, der im später ausgeführten Bauwerk nicht mehr erkennbar ist, wie eine Achse oder ein Lichtraum.

Unter *Attribut* werden explizite Attribute von Modellobjekten verstanden, die von einer Regelprüfung ausgewertet werden können, beispielsweise die Breite eines Verkehrsraumes, die zugehörige Richtzeichnung oder die Bewegungsrichtung eines Auflagers.

Alle anderen Regeln benötigen aus dem Modell abgeleitete Informationen, zum Beispiel die Stützweite der Brücke, und gehören somit zur Kategorie *implizit*.

Singer und Borrmann 2016 stellen fest, dass die geometrische Repräsentation eines Modells unter Umständen von einem BIM-Authoring- und einem Model-Checking-Tool unterschiedlich interpretiert wird, was zu Fehlern bei der Modellprüfung führen kann. Es wird daher versucht, möglichst viele geometrische Informationen in Attributen zu hinterlegen. Zudem ist es bei der Konformitätsprüfung einfacher explizite Attribute auszuwerten, als die geometrischen Informationen implizit ableiten zu müssen. Die Übergabe geometrischer Informationen als semantische Attribute stellt jedoch eine Datenredundanz dar, welche zu einer Inkonsistenz der Daten führen kann (Singer und Borrmann 2016). Deshalb muss bei der Modellierung genau darauf geachtet werden, dass die Attribute korrekte Informationen enthalten und an Änderungen angepasst werden.

Nachdem alle betroffenen Paragrafen der Richtlinien analysiert sind, werden alle Modellanforderungen für die Konformitätsprüfung zusammengestellt (sh. Kapitel Ergebnisse 4.4.4). Dabei wird jedes benötigte Bauteil mit allen benötigten Attributen benannt.

4.4.3 Darstellung des Vorgehens an konkreten Beispielen

Um die Arbeitsweise bei der Erstellung von Modellanforderungen aus den Richtlinien genauer zu beleuchten, werden zwei konkrete Beispiele nachfolgend erläutert.

Beispiel 1: RAB-Ing Teil 4 – 8 Regelquerschnitt

Als Beispiel für die Erarbeitung der Anforderungen aus der RAB-Ing wird ein Teil der Anforderungen an den Regelquerschnitt herangezogen.

Der genaue Wortlaut aus der Richtlinie lautet:

- (1) Der Regelquerschnitt des Überbaus ist im Maßstab M. 1:50 darzustellen, bei Stahlbrücken evtl. auch größer.
- (2) Der Schnitt ist rechtwinklig zur Brückenachse zu führen.
- (3) Darzustellen sind:
 - Breiten- und Querschnittsaufteilung,
 - Mindestkonstruktionshöhe,
 - Geländer,
 - Fahrzeugrückhaltesysteme mit Angabe der Aufhaltstufe und des Wirkungsbereichs,
 - ...

(BMDV 2022a)

Jeder Paragraf wird zuerst auf Relevanz hinsichtlich eines BIM-Modells und des Rahmens der Arbeit geprüft. In diesem Fall entfallen die Paragraphen (1) und (2), da sie für ein Modell nicht relevant sind. Die „Breiten- und Querschnittsaufteilung“, welche die Modellierung des Überbaus und der Verkehrsräume erfordert, ist bereits durch die Anforderungen von RAB-Ing Teil 4 Abschnitt 1 – 6 Längsschnitt und 7 – Grundriss abgedeckt und erfordert daher kein weiteres Objekt im Modell. Aus den Anforderungen „Mindestkonstruktionshöhe“, „Geländer“ und „Fahrzeugrückhaltesysteme ...“ werden Regeln bzw. Anforderungen abgeleitet (Abbildung 9).

Nummer	Werk	Regel	betrifft	Information im Modell
64	RAB	8 (3)	Mindestkonstruktionshöhe	implizit
65	RAB	8 (3)	Geländer	Objekt
66	RAB	8 (3)	Schutzeinrichtung	Objekt
67	RAB	8 (3)	Schutzeinrichtung	Attribut
68	RAB	8 (3)	Schutzeinrichtung	Attribut

Abbildung 9: Zuordnung von Anforderungen aus der RAB-Ing zu eigenen Kategorien von Modellinformationen (Quelle: Eigene Darstellung)

Die Mindestkonstruktionshöhe ist in einem Modell nicht explizit dargestellt, da keine Maßketten modelliert werden. Die Information selbst kann jedoch bei der Modellprüfung mit einer Regel aus den betreffenden Objekten im Modell abgeleitet werden. Sie ist somit implizit im Modell enthalten. Das Gelände und die Schutzeinrichtung müssen gemäß RAB explizit im Modell enthalten sein, d.h. für sie muss jeweils ein Objekt modelliert werden. Um die Anforderungen der RAB-Ing bezüglich der Aufhaltstufe und des Wirkungsbereichs zu erfüllen, werden dafür jeweils eigene Attribute benötigt, welche mit Nummer 67 und 68 in der Abbildung bezeichnet sind. Die Kennzeichnung welches Attribut zu welcher Anforderung zugeordnet ist befindet sich in einer nicht dargestellten Spalte.

Beispiel 2: RE-Ing Teil 2 Abschnitt 2 – 1 Überbauten – 1.1 Allgemeines (7)

In diesem Teil der RE-Ing werden folgende Anforderungen an die Kappenanker gestellt:

(7) Kappenverankerungen, die die Dichtungsschicht durchdringen, sind nicht zugelassen. Die aufzunehmenden Kräfte können z. B. bei hohen Lärmschutzwänden (LSW) über zusätzliche liegende Schlaufen als Anschlussbewehrung in den Kragarm und verlängerte Durchsteckanker durch die Fußplatte der LSW-Verankerung übertragen werden. Der Abtrag der Kräfte erfolgt hierbei über ein Fachwerkmodell. Für Bestandsbrücken gilt Nr. 3.3.6.

(BMDV 2022b)

Der erste Satz des Paragraphen beschreibt eine für die Entwurfsplanung relevante Forderung nach Kappenankern, die die Dichtungsschicht nicht durchdringen. Der zweite und dritte Satz enthält Anforderungen an die Bewehrung und die statische Berechnung und kann daher für die Entwurfsplanung entfallen. Auch der Verweis für die Bestandsbrücken entfällt, da sich der Rahmen der Arbeit auf Neubauten beschränkt. Für die Kappenanker werden anschließend die Anforderungen in die Kategorien der Modellinformationen eingeordnet (Abbildung 10).

Nummer	Werk	Regel	betrifft	Information im Modell
123	RE 2	1.1 (7)	Kappenanker	Objekt
124	RE 2	1.1 (7)	Kappenanker	implizit

Abbildung 10: Zuordnung von Anforderungen an Kappenanker aus der RE-Ing zu eigenen Kategorien von Modellinformationen (Quelle: Eigene Darstellung)

Um eine Durchdringung der Dichtungsschicht durch die Kappenanker auszuschließen, müssen sowohl die Dichtungsschicht als auch die Kappenanker explizit als Bauteile im Modell vorliegen. Die Dichtungsschicht bzw. Abdichtung wird bereits durch die RAB-Ing Teil 2 Abschnitt 4 – 8 (3) als Modellbestandteil gefordert, weshalb für die RE-Ing kein eigenes Objekt benötigt wird. Die Kappenanker müssen jedoch als *Objekt* in der Liste der Anforderungen ergänzt werden (Nummer 123). Die Durchdringung der Bauteile *Abdichtung* und *Kappenanker* liegt als implizite Information im Modell vor. Somit erfolgt eine Zuordnung dieser Regel zur Kategorie *implizit*.

4.4.4 Ergebnisse

Mit den eben genannten Prinzipien lassen sich die einzelnen Paragraphen der Richtlinien vollständig analysieren. Sie wurden in der Reihenfolge RAB-Ing, RE-Ing, RiZ-Ing und zum Schluss ZTV-Ing bearbeitet, weil sich der Detaillierungsgrad der Anforderungen so von einer Richtlinie zu nächsten steigert. In den meisten Paragraphen, ausgenommen die der RAB-Ing, werden nicht explizit Modellinhalte gefordert, sondern Anforderungen an den Entwurf gestellt. Um diese überprüfen zu können sind explizit modellierte Objekte nötig. Die vollständige Kategorisierung kann in der Excel-Tabelle im digitalen Anhang eingesehen werden (Übersicht siehe Anhang B).

Tabelle 3: Zuordnung von erarbeiteten Anforderungen aus der jeweiligen Richtlinie zu einer Kategorie für benötigte Modellinformationen

Richtlinie	Objekt	Attribut	implizit	gesamt
RAB-Ing	47	19	13	79
RE-Ing	21	48	86	155
RiZ-Ing	1	48	35	84
ZTV-Ing	3	16	31	50
gesamt	72	131	165	368

Es zeigt sich, dass die meisten Objekte im Modell bereits als Modellinhalt in der RAB-Ing gefordert werden. Somit ist es nicht nötig sie für eine Regel der nachfolgend bearbeiteten Richtlinien ein weiteres Mal aufzulisten. Für die RE-Ing, RiZ-Ing und ZTV-Ing

wurden nur dann Regeln in der Kategorie *Objekt* ergänzt, wenn das jeweilige Modellbestandteil in der RAB-Ing nicht bereits explizit gefordert wurde. Daraus folgt, dass die Regeln der später bearbeiteten Richtlinien vermehrt in die Kategorien *Attribut* und *implizit* fallen (vgl. Tabelle 3).

Nachdem alle Richtlinien bearbeitet wurden, werden aus den Elementen der Kategorien *Objekt* und *Attribut* die Modellanforderungen zusammengestellt. Die vollständige Liste aller zu modellierenden Bauteile inklusive zusätzlich benötigter Bauteilattribute für die Konformitätsprüfung im Rahmen der Masterarbeit befindet sich in Anhang A. Die Tabelle in Anhang A enthält bereits die Zuordnung der Modellanforderungen zum IFC4x3 -Datenschema, welche im nächsten Abschnitt genauer erläutert wird.

Einige Paragraphen stellen objektübergreifende Vorgaben, zum Beispiel eine Materialzuordnung oder die Zuordnung der Information, ob bei einem betreffenden Bauteil ein Außenbauteil vorliegt. Die Angabe der Baustoffe wird in RAB-Ing Teil 4 Abschnitt 1 – 4 gefordert, die Identifizierbarkeit von Außenbauteilen in RE-Ing Teil 2 Abschnitt 3 – 3.2 (1). Die übergreifenden Anforderungen sind nicht bei den einzelnen Objekten mit aufgeführt, um die Übersichtlichkeit der Tabelle zu erhöhen.

Insgesamt werden an 72 Objekte Modellanforderungen gestellt. Abhängig von der Art, der Länge oder dem Querschnitt der modellierten Brücke werden einige Objekte, beispielsweise ein Hohlkasten oder ein Pfeiler nicht benötigt. Die zwingend notwendigen Bestandteile eines Modells sind somit projektspezifisch festzulegen.

Für die Überprüfung von 131 der untersuchten Anforderungen müssen die Informationen in Attributen ausgewertet werden. Daraus ergeben sich für die Modellanforderungen 53 Attribute, die unterschiedlichen Objekten zusätzlich zu den grundlegenden Eigenschaften aus IFC4x3, wie die GUID, zugeordnet werden müssen.

Die große Diskrepanz der Zahlen ergibt sich unter anderem daraus, dass viele Anforderungen der Kategorie *Attribut* den Anwendungsbereich der zugehörigen Richtzeichnung vorgeben. Zum Beispiel wird die Entwässerung der einzelnen Brückenbestandteile mittels Entwässerungsrohren modelliert. Den Rohr-Objekten im Modell wird dann die zugehörige Richtzeichnung Was X zugeordnet. Bei der anschließenden Konformitätsprüfung wird geprüft, ob die Richtzeichnung entsprechend ihres Anwendungsbereichs korrekt zugeordnet wurde. Im Fall der Entwässerung beziehen sich die jeweiligen Anwendungsbereiche auf bestimmte zu entwässernde Bauteile, z.B. das Widerlager ohne Wartungsgang (Was 5) oder der Flügelbereich (Was 8). Somit gibt es allein

aus der RiZ-Ing für die Entwässerung sechs Anforderungen, die alle mit einem Attribut *zugehoerigeRiZ* abgedeckt werden können. (BASt 2022)

Ein weiterer Grund für die Diskrepanz ist, dass die Anforderungen an eine Abmessung eines Objekts häufig von anderen Faktoren, beispielsweise dem benachbarten Verkehrsraum, abhängen. Beispielsweise schreibt die RE-Ing Teil 2 Abschnitt 1 für einen ländlichen Weg auf der Brücke eine Schrammbordhöhe von mindestens 20cm vor (2.3 (2)), für eine ländlichen Weg unter der Brücke hingegen 3cm (2.4 (1)) (BMDV 2022b). Auch in diesem Fall wird dasselbe Attribut, *Höhe Schrammbord*, von mehreren Regeln benötigt und deshalb für die Modellanforderungen zusammengefasst.

4.5 Vergleich der Modellaustauschanforderungen mit IFC4x3

Um eine Aussage bezüglich der Möglichkeiten der modellbasierten Konformitätsprüfung von Brückenentwürfen des Straßenbaus mit IFC4x3 treffen zu können, werden die erarbeiteten Objekte und Attribute so weit wie möglich zu passenden IFC4x3-Entities und Properties zugeordnet. Das vollständige Ergebnis der Zuordnung wird in der Tabelle in Anhang A dargestellt.

Zuordnung von Objekten zu IFC-Entities

Für die Zuordnung der 72 erarbeiteten Objekte zu IFC4x3-Entities wird im ersten Schritt die bereits vorhandene Zuordnung von den BIM-Klassen der Verkehrswege 2.0 von buildingSMART Deutschland (bSD) zu IFC4x3 betrachtet (buildingSMART Deutschland 2023). Die dort zugeordneten IFC-Entities werden mit Hilfe der Dokumentation von IFC4x3 von bSI nachvollzogen (buildingSMART International 2022a). Für einige Objekte ist die Zuordnung von bSD mehrdeutig oder nicht vorhanden, weshalb in einem zweiten Schritt passende IFC-Entities anhand der IFC4x3-Dokumentation ausgewählt werden. Die betreffenden Entities mit der eigenen Zuordnung sind in der Tabelle im Anhang A blau gekennzeichnet.

Die Objekte werden so detailliert wie möglich zugeordnet, indem wo möglich die korrekten Enumeration-Typen beim IFC-Attribut *PredefinedType* angegeben werden. So wird zum Beispiel eine Presse als *IfcTransportElementTypeEnum.LIFTINGGEAR* in IFC4x3 übersetzt. Die Objekte, für die kein konkreter Enumeration-Typ verfügbar ist, können als Enumeration-Typ *USERDEFINED* festgelegt werden. Über eine projekt-spezifische Namenskonvention kann so eine Konformitätsprüfung einfach ermöglicht

werden. Insgesamt wird der Enumeration-Typ *USERDEFINED* bei sieben Objekten zugeordnet, beispielsweise bei den Tropfützen oder der Fahrbahnplatte.

Für alle 72 Objekte können passende IFC-Entities zugeordnet werden und es muss nicht auf Platzhalter-Klassen wie *IfcBuildingElementProxy* zurückgegriffen werden. Somit zeigt sich, dass IFC4x3 sehr gut dafür geeignet ist, alle benötigten Bestandteile eines Brückenmodells in ausreichender Genauigkeit abzubilden, sofern alle benötigten Entities in der Authoring-Software implementiert sind.

Zuordnung von Attributen zu IFC-Properties

Grundsätzlich werden für die Konformitätsprüfung für alle erarbeiteten Objekte im Modell eine GUID (Attribut *GlobalID*) und eine Geometrierepräsentation (Attribut *Representation*) benötigt. Bis auf Raum-Objekte und Achsen wird außerdem die Zuordnung eines Materials gefordert, was in IFC über *IfcRelAssociatesMaterial* geschieht. Für alle Bauteile, die als *IfcWall*, *IfcSlab*, *IfcBeam* oder *IfcRamp* modelliert sind, muss die Property *IsExternal* aus dem jeweiligen Property Set *Pset_...Common* ausgefüllt werden, um festzulegen, ob das Bauteil ein Außenbauteil ist. Die bauteilübergreifenden Attribute können alle mit Attributen der IFC-Entities und Standard-Property Sets abgedeckt werden.

Anschließend werden die zusätzlich benötigten Informationen einzelner Objekte betrachtet. Einige der zusätzlichen Informationen können in den Attributen der jeweiligen Entities hinterlegt werden. Das ist für 7 der 53 zusätzlich benötigten Bauteilattribute möglich. Beispielsweise kann das Ziel eines Verkehrswegs über das Attribut *Description* der zugehörigen Entity *IfcSpatialZone* beschrieben werden oder der Typ eines Spannglieds aus einer Liste möglicher Werte beim Attribut *PredefinedType* der Entity *IfcTendon* ausgewählt werden. Dabei muss genau festgelegt sein, welches Attribut für welche Information verwendet werden muss, da die Namen der Attribute der Entities sehr allgemein sind und es sonst zu Fehlern kommen kann.

Für viele Regeln werden zusätzlich zu den vorhandenen weitere Attribute benötigt. Um diese abzudecken, werden so weit wie möglich die im IFC4x3-Datenschema vorhandenen Property und Quantity Sets verwendet. Für 24 der geforderten 53 zusätzlichen Attribute können vorhandene Property und Quantity Sets verwendet werden. Für die Breite und die Höhe von Räumen, in diesem Fall Verkehrs- oder Hohlräumen, können beispielsweise die im Quantity Set *Qto_SpatialZoneBaseQuantities* festgelegten *IfcProperties Width* und *Height* genutzt werden (buildingSMART International 2022a).

Einen Spezialfall in Deutschland bildet die Verwendung der Richtzeichnungen aus der RiZ-Ing. Um den jeweiligen Objekten die passende Richtzeichnung zuzuordnen zu können ist ein eigenes Attribut nötig, welches über ein eigens generiertes Property Set zugeordnet werden kann. Insgesamt wird das Attribut *zugehoerigeRiZ* für 16 Objekte benötigt, welche durch ein Property Set abgedeckt werden können. Es ist zu beachten, dass manche Objekte mehrere zugeordnete Richtzeichnungen haben können. Ein Beispiel dafür ist das Geländer, da hierfür das Geländer, der Handlauf und die Verankerung jeweils in einer eigenen Richtzeichnung vorgegeben werden, im Modell eines Bauwerksentwurfs aber nur ein Objekt für das Geländer vorgesehen ist (BASt 2022).

Die noch verbleibenden sechs Attribute müssen als eigene Attribute in eigenen Property Sets erstellt werden.

Dass in IFC die Attribute einfach über Property Sets ergänzt werden können, ist ein großer Vorteil von IFC als Datenquelle für Konformitätsprüfungen. Mit den in IFC4x3 vordefinierten Attributen und Property Sets kann mehr als die Hälfte der nötigen Attribute abgedeckt werden. Dies vereinfacht eine Konformitätsprüfung insofern, dass die Regeln auf Attribute ausgelegt werden können, die immer denselben Namen haben. Für Attribute mit frei vergebenen Namen ist eine Fehleranfälligkeit der Regeln größer, vor allem wenn die aus den Regeln vorgegebene Namenskonvention im Modell nicht exakt eingehalten wird. Für die Konformitätsprüfung ist das eher unproblematisch, da von den 22 frei vorzugebenden Attributen 16 denselben Namen (*zugehoerigeRiZ*) haben und über ein einziges Property Set abgedeckt sind.

4.6 Zusammenfassung

Um eine Konformitätsprüfung für Brückenentwürfe an einem BIM-Modell durchführen zu können, müssen von dem Modell bestimmte Informationsanforderungen eingehalten werden.

Für die Formulierung der Informationsanforderungen werden in der Praxis unterschiedliche Methoden angewendet. Dazu gehören einfache nicht standardisierte *Fließtext-* oder *Tabellendarstellungen*, teilstandardisierte *Product Data Templates* oder *Data Dictionaries* und standardisierte Methoden wie *IDM/MVD*, *IDS* oder *LOIN*.

Aus den Richtlinien RAB-Ing, RE-Ing, RiZ-Ing und ZTV-Ing entstehen Modellanforderungen für Modelle von Brückenentwürfen. Der zugrunde liegenden HOAI-Leistungsphase *Genehmigungsplanung* kommt dabei ein LOD 300 zu.

Die Modellanforderungen für die Konformitätsprüfung von Brückenentwürfen von Neubauten klassischer Massivbrücken können aus den einzelnen Paragraphen der Richtlinien abgeleitet werden. Der daraus resultierende Objekt- und Attributkatalog mit 72 Objekten und 53 Attributen wird tabellarisch in Anhang A zusammengestellt.

Um eine gute Vergleichbarkeit der Anforderungen mit dem IFC4x3-Datenmodell zu schaffen, werden, soweit vorhanden, jedem Objekt und Attribut im Katalog die passenden IFC-Entities und IFC-Properties sowie die zugehörigen Property Sets zugeordnet. Dabei ergibt sich, dass alle 72 Objekte mit passenden IFC-Entities abgebildet werden können. Mehr als die Hälfte der geforderten Attribute können mit IFC-eigenen Attributen und Properties abgedeckt werden. Bei 16 Objekten muss zusätzlich eine Property für die zugehörige Richtzeichnung ergänzt werden, was mit nur einem eigenen Property Set ermöglicht werden kann.

Somit zeigt sich, dass die IFC-Version 4x3 sehr gut für die Abbildung der Modelldaten für die Konformitätsprüfung geeignet ist. Jedoch muss immer die Implementierung des IFC-Datenmodells in der jeweiligen BIM-Authoring und Model-Checking-Software berücksichtigt werden. Nur wenn alle benötigten IFC-Entities und Properties in der Software implementiert sind, können die Modellanforderungen eingehalten werden.

5 Aufbereitung der Richtlinien zu computer-interpretierbaren Regelsätzen

In diesem Abschnitt wird untersucht, inwiefern die Regelungen aus den Richtlinien des Brückenbaus in konkrete Regelsätze für eine Konformitätsprüfung eines BIM-Modells übertragen werden können. Die Regeln werden kategorisiert, um eine generelle Einschätzung zur Implementierbarkeit geben zu können. Die gewählte Klassifizierung wird zunächst erläutert und dann auf die Richtlinien angewendet. Anschließend werden beispielhaft einzelne Regeln für jede Kategorie detailliert beschrieben und für eine Implementierung aufbereitet.

5.1 Klassifizierung der Regeln

Solihin und Eastman 2015 stellen in ihrer Veröffentlichung zur Klassifizierung von Regeln vier Kategorien (*Classes*) mit steigender Komplexität vor.

In *Class 1* werden Regeln eingeordnet, welche explizit im Modell vorliegende Attribute und Objektreferenzen überprüfen. Dazu gehören unter anderem Tests, die prüfen, ob für nachgeschaltete komplexere Regeln alle nötigen Objekte und Attribute vorhanden sind, und Tests mit einfachen Vorgaben aus Richtlinien, z.B. bestimmte Attributwerte. (Solihin und Eastman 2015)

Ein Beispiel für eine *Class 1*-Regel ist die Überprüfung, dass alle Kappen und Gesimse in Ort beton geplant sind, was durch RE-Ing Teil 2 Abschnitt 2 Paragraf 1.1 (6) gefordert wird (BMDV 2022b). Die Information, wo das Bauteil gefertigt wird, kann mit einem expliziten Parameter im Modellobjekt hinterlegt werden, welcher bei der Regelprüfung ausgewertet wird.

Die *Class 2* beinhaltet Regeln, für die einfache abgeleitete Attributwerte benötigt werden. Dabei werden implizite Modelldaten ausgewertet und gegebenenfalls wenig komplexe geometrische Berechnungen durchgeführt. (Solihin und Eastman 2015)

Aus den Richtlinien des Brückenbaus fällt beispielsweise eine Überprüfung der lichten Durchfahrthöhe unter der Brücke gemäß RE-Ing Teil 2 Abschnitt 1 Paragraf 2.2 in die *Class 2* (BMDV 2022b).

Mit der *Class 3* werden Regeln abgedeckt, die eine erweiterte Datenstruktur benötigen. Das ist häufig der Fall, wenn die Regel komplexe Bedingungen aus einer Richtlinie abbilden soll. (Solihin und Eastman 2015)

Wenn etwa einem Hohlkasten im Modell die Richtzeichnung RiZ-Ing Zug 1 Blatt 2 zugeordnet ist, müssen bestimmte Randbedingungen vom restlichen Modell erfüllt werden. Es müssen Öffnungen im Boden des Hohlkastens, denen die Richtzeichnung Zug 7 Blatt 2 zugeordnet ist, mit festgelegten Abmessungen vorhanden sein. Deren Soll-Lage muss mit weiteren Randbedingungen zunächst aus dem Modell ermittelt werden. (BASt 2022) Alle Randbedingungen müssen in der *Class 3*-Regel hinterlegt werden.

Die komplexesten Regeln fallen in *Class 4*. Sie bilden häufig Anforderungen ab, die auf mehr als eine Art eingehalten werden können. Das Wissen, wie eine solche Anforderung eingehalten werden kann, muss in der *Class 4*-Regel enthalten sein und kann erweitert werden, wenn eine neue Lösung bekannt wird. (Solihin und Eastman 2015)

Für die Regeln in *Class 4* ist eine reine Analyse von vorgegeben Randbedingungen nicht ausreichend, da unterschiedliche Lösungen mit jeweils eigenen Randbedingungen angewendet werden können, um die Regel zu erfüllen. Ein Beispiel aus dem Brückenbau ist die Vorgabe der RE-Ing Teil 2 Abschnitt 3 Paragraf 3.1.2 (1), dass alle begehbaren Bauteile zugänglich sein müssen (BMDV 2022b). In der Prüfregel muss hinterlegt werden, was ein *begehbare*s Bauteil ausmacht und wie *Zugänglichkeit* erreicht werden kann. Während ein begehbare Bauteil über wenige Randbedingungen, i.d.R. lichte Höhe und lichte Breite des Innenraums, definiert wird, kann eine Zugänglichkeit auf unterschiedliche Weise sichergestellt werden, z.B. durch eine Tür und einen entsprechenden Gang im Widerlager oder eine Öffnung von oben in den Hohlkasten mit einer Steigleiter.

5.2 Auswertung der Analyse der Richtlinien

Die unter Kapitel 4 erarbeiteten Regeln werden nun entsprechend den gerade erläuterten Kategorien klassifiziert, um später mit der Implementierung weniger Regeln eine zuverlässige Aussage zur Implementierbarkeit für das gesamte Regelwerk treffen zu können. Die einzelnen Ergebnisse können der Tabelle im digitalen Anhang entnommen werden (siehe Anhang B). Eine Zusammenfassung der Ergebnisse ist in Tabelle 4 dargestellt.

Tabelle 4: Zuordnung von erarbeiteten Anforderungen aus der jeweiligen Richtlinie zu den Klassifizierungen von Regeln nach Solihin und Eastman 2015

Richtlinie	Class 1	Class 2	Class 3	Class 4	gesamt
RAB-Ing	75	4	-	-	79
RE-Ing	67	57	15	16	155
RiZ-Ing	50	33	1	-	84
ZTV-Ing	19	29	2	-	50
gesamt	211	123	18	16	368

Die erarbeiteten Anforderungen aus der RAB-Ing lassen sich beinahe vollständig in Class 1 einordnen, da in der RAB-Ing nur Modellinhalte gefordert werden und keine Anforderungen an Abmessungen, Abstände oder ähnliches gestellt werden. Die Überprüfung dieser Regeln in der Konformitätsprüfung dient vor allem dazu, die Vollständigkeit des Modells sicherzustellen. Die vier Anforderungen in Class 2 betreffen einerseits die Entwässerung kritischer Stellen und andererseits die Darstellung der kritischen Punkte, hier die Mindestkonstruktionshöhe und die kritische lichte Höhe. Mit den Regelsätzen für die RAB-Ing muss geprüft werden, ob eine Entwässerung an Überbau und Widerlager anschließt und ob die kritischen Abmessungen implizit aus dem Modell abgeleitet werden können. Konkrete Anforderungen an Lage und Abmessung der Entwässerung und an Mindestmaße werden dann in der RE-Ing gestellt.

In der RE-Ing werden viele Anforderungen an die Konstruktion der Brücke und der Brückenausstattung gestellt. Der Großteil der Anforderungen fällt in Class 1 und 2, da für die Einhaltung des jeweiligen Paragraphen konkrete geometrische Abmessungen mit Soll-Werten verglichen werden müssen. Dabei wird nach Möglichkeit ein passender Parameter, der die betroffene Abmessung enthält, zu den Bauteilen zugeordnet, um eine möglichst einfache Implementierung zu gewährleisten. Mit der Abmessung als Parameter erfolgt eine Einordnung in Class 1, mit einer impliziten Ableitung der Abmessung in Class 2.

Einige der Anforderungen der RE-Ing sind so formuliert, dass für die Beurteilung der zugehörigen Richtlinie Fachwissen nötig ist, welches in den Prüfreden hinterlegt werden muss. Ein Ingenieur kann beispielsweise einschätzen, ab welchen Abmessungen

eine Großbrücke vorliegt oder ob sich vor einem Widerlager Stauwasser bilden kann, während eine Prüfregel dafür konkrete Randbedingungen für eine Untersuchung benötigt. Diese Anforderungen sind in Class 3 und 4 eingeordnet. Wenn nur wenige Randbedingungen formuliert werden müssen, erfolgt eine Einordnung in Class 3, bei mehr Randbedingungen in Class 4. In Class 3 fallen zusätzlich Anforderungen, die die Ableitung von vielen impliziten Informationen erfordern oder die Berechnung aufwändiger Formeln beinhalten.

Für die Erfüllung der Anforderungen aus der RiZ-Ing wird vor allem geprüft, ob die jeweilige Richtzeichnung dem korrekten korrespondierenden Bauteil zugeordnet ist und ob der Anwendungsbereich der jeweiligen Richtzeichnung eingehalten ist.

Zudem werden ausgewählte geometrische Angaben geprüft, sofern sie den Detaillierungsgrad einer Entwurfsplanung nicht übersteigen. Das sind insbesondere Abmessungen der Kappen, Geländer, Flügel und Überbauten sowie die Anordnung der Pressen, Entwässerungsrohre und Bodenöffnungen im Hohlkasten. Da die untersuchten Anforderungen meist sehr konkrete geometrische Maße sind, fallen sie beinahe ausschließlich in Class 1 und 2. Die Unterscheidung in Class 1 und 2 erfolgt analog zur RE-Ing darüber, ob die Abmessung im Modell als Attribut oder implizit hinterlegt ist.

Alternativ zur Aufteilung der Richtzeichnungen in einzelne Anforderungen mit daraus resultierenden einfachen Regeln kann man auch für jede Richtzeichnung nur eine Regel schreiben, die dann in Class 3 eingeordnet ist. Mit dieser Regel werden dann jeweils alle einzelnen geometrischen und semantischen Anforderungen gesammelt geprüft. Das wird exemplarisch für RiZ-Ing Zug 1 Blatt 2 getestet, weshalb nur aus dieser Richtzeichnung eine Class 3-Regel abgeleitet wird. Für die restlichen Richtzeichnungen wird die Methode mit der Aufteilung in einzelne Anforderungen gewählt, da die einzelnen Anforderungen für die Bestimmung der nötigen Attribute im Rahmen der Modellanforderungen in Kapitel 4 bereits erarbeitet wurden.

Die ZTV-Ing wurde nur sehr beschränkt bei den Anforderungen an den Entwurf berücksichtigt, da sie zumeist Anforderungen an die Ausführung enthält. Die berücksichtigten Anforderungen betreffen vor allem Mindestabmessungen von Verbauten, Über- und Unterbauten und Geländer sowie Entwässerungsbestandteile. Es wird auch hier nach Möglichkeit ein Bauteilattribut für die betreffende Abmessung verwendet. Somit fällt der Großteil der Anforderungen der ZTV-Ing in Class 1 und 2. Die Ausnahme bilden hier die Anforderungen an die Anzahl und Verteilung der Brückenabläufe gemäß

ZTV-Ing Teil 6 Abschnitt 10 Paragraf 2.2 (1) und (2). Um deren Einhaltung zu prüfen, müssen komplexe geometrische Berechnungen angestellt werden, welche eine Einordnung in Class 3 erforderlich machen.

Insgesamt werden fast 60 Prozent der untersuchten Anforderungen in Class 1 eingestuft, mehr als 30 Prozent in Class 2 und jeweils unter 5 Prozent in Class 3 und 4. Daraus lässt sich erkennen, dass die Anforderungen an Brückenentwürfe von Massivbrücken des Straßenbaus sehr konkret in den Richtlinien definiert sind und wenig Spielraum für Interpretationen lassen. Für eine Implementierung in eine automatisierte Konformitätsprüfung ist das vorteilhaft, da vermehrt einfache Regeln erzeugt werden müssen. Auch die Verwendung von Attributen für möglichst viele Abmessungen ist dabei von Vorteil. Hier sollten möglichst viele standardisierte Attribute verwendet werden, z.B. Properties aus bSI-Property Sets. Damit werden Fehlinterpretationen des Model-Checking-Tools vermieden, die durch eine abweichende Benennung der betrachteten Eigenschaften entstehen können.

5.3 Auswahl exemplarischer Regeln für die Implementierung

Für jeder der vier Regel-Kategorien wird mindestens eine Regel für die Implementierung ausgearbeitet.

5.3.1 Class 1

Als Beispiel für eine Prüfung in Class 1 wird RE-Ing Teil 2 Abschnitt 2 – 1.1 (6) gewählt. Die Richtlinie besagt, dass Kappen fugenlos in Ortbeton geplant werden müssen. (BMDV 2022b)

Aus der Richtlinie lassen sich zwei Regeln für die Modellprüfung ableiten. Mit der einen Regel wird überprüft, dass die Kappenobjekte im Modell keine Fugen enthalten, mit der anderen, dass die Betonkappen planmäßig vor Ort ausgeführt werden. Die erste Regel fällt in Class 2 und wird für die exemplarische Implementierung nicht weiter beachtet. Die zweite Regel entspricht einer reinen Attributprüfung in Class 1 und wird nachfolgend für die Implementierung aufbereitet.

Zunächst müssen die Modellobjekte identifiziert werden, die von der Regel betroffen sind. Die Kappen müssen gemäß den erarbeiteten Modellanforderungen (Anhang A) modelliert, attribuiert und in IFC4x3 exportiert werden. Sie liegen somit im IFC-Modell als *IfcBeamTypeEnum.EDGEBEAM* vor und enthalten zusätzliche Informationen zur

zugehörigen Richtzeichnung und dem Herstellungsort (engl. *assembly place*). Sie können über den Objekttyp *IfcBeamTypeEnum.EDGEBEAM* identifiziert werden, da bei korrekter Modellierung und korrektem Export kein anderes Objekt im Modell diesem Objekttyp zugeordnet ist.

Mit der Regel wird anschließend für jede Kappe im Modell geprüft, ob dem Attribut *AssemblyPlace* der Wert *SITE* zugeordnet ist, was der geforderten Ausführung in Ortbeton entspricht.

Grundsätzlich besteht die Herausforderung bei Class 1-Prüfungen mehr darin, die Identifizierung der Objekte zu ermöglichen, als die Regel zu definieren. Die Verwendung von IFC4x3 für den Datenaustausch im Vergleich zu vorherigen IFC-Versionen bietet dabei den Vorteil, dass die Brückenbestandteile zu entsprechenden, sehr konkreten Objekttypen zugeordnet werden können. In den früheren IFC-Versionen, welche auf den Hochbau ausgelegt sind, werden die Brückenbestandteile i.d.R. mit dem identischen Objekttyp *IfcBuildingElementProxy* übergeben, der den zugehörigen Objekten keine konkrete Bedeutung zuordnet, und können daher nicht über ihren Objekttyp differenziert werden (buildingSMART International 2019).

Wenn eine Identifizierung über den IFC-Typ nicht möglich ist, kann der Name der Objekte verwendet werden. Dafür muss bei der Benennung von Objekten eine strikte Namenskonvention eingehalten werden, welche durch die Regeln vorgegeben wird. Über die Namenskonvention kann dem Objekt die jeweilige Bedeutung des Bauteils zugeordnet werden. Wird die Namenskonvention nicht eingehalten oder werden die Attribute vom BIM-Autor fehlerhaft befüllt, können die Objekte nicht identifiziert werden und eine Attributprüfung ist nicht möglich. Nach Möglichkeit sollte daher der standardisierte IFC4x3-Typ zur Identifizierung verwendet werden, welcher bei korrekter Modellierung, mit korrektem Mapping, automatisiert zugeordnet wird. Dadurch werden menschliche Fehler minimiert.

5.3.2 Class 2

In die Class 2 fällt die Überprüfung vieler geometrischer Abmessungen, von denen exemplarisch die Überprüfung der lichten Höhe zwischen der Oberkante der unterführten Straße und der Unterkante des Überbaus implementiert wird.

Die lichte Höhe hängt vom Straßentyp der unterführten Straße ab. Für Bundesfernstraßen gelten 4,70m, für Unterführungen ländlicher Wege 4,50m und für Viehbetrieb 2,50m (RE-Ing Teil 2 Abschnitt 1 – 2.2/2.4/2.5) (BMDV 2022b).

Zuerst wird die Distanz zwischen der Unterkante des Überbaus und der Straßenoberkante der unterführten Straße berechnet und dann mit dem Soll-Wert verglichen. Die in Frage kommenden Modellobjekte der unterführten Straße und des Überbaus werden analog zur Class 1 möglichst über den IFC-Typ oder den Namen des Objekts entsprechend einer Namenskonvention identifiziert.

Bei der Ermittlung der geringsten lichten Höhe im Modell ist zu beachten, dass die Lage des kritischen Punktes im Vorhinein nicht bekannt ist. Der kritische Punkt kann der höchste Punkt der Straße, der tiefste Punkt des Überbaus oder ein Punkt an einer beliebigen Stelle dazwischen sein. Zudem kommen häufig mehrere Modellobjekte für die kritische Höhe in Frage, z.B. der Hohlkasten oder eine Entwässerungsleitung für die Unterkante des Überbaus und verschiedene Fahrstreifen oder ein Gehweg für die Oberkante unter der Brücke.

Somit muss die Prüfregel alle möglichen Objektkombinationen untersuchen und iterativ die kritische Stelle ermitteln. Erst im Anschluss ist ein Vergleich dieser kritischen Höhe mit dem Regelwert möglich.

5.3.3 Class 3

Für die Class 3 wird die RiZ-Ing Zug 1 Blatt 2 als computerinterpretierbare Regel formuliert. Darin wird angegeben, wie der Zugang zu Spannbeton-Hohlkästen über Bodenöffnungen ermöglicht wird. Es wird vor jedem Widerlager eine Bodenöffnung im Hohlkasten mit $b/h = 1,00\text{m}/1,50\text{m}$ gefordert, welche nach Zug 7 Blatt 2 ausgeführt werden muss. Zusätzlich wird eine nach Zug 7 Blatt 2 ausgeführte Bodenöffnung mit $b/h = 1,20\text{m}/2,50\text{m}$ oberhalb eines Verkehrsweges oder danebenliegenden Stellflächen benötigt. Für alle Öffnungen werden zusätzlich Lasthaken und eine Gitterrostabdeckung gefordert, welche jedoch im Rahmen des Entwurfs vernachlässigt werden. (BASt 2022)

Die Abmessungen der Öffnungen können entweder berechnet oder als Attribute des Öffnungsobjekts übergeben werden. Um die Regelimplementierung zu vereinfachen werden die Öffnungsobjekte mit Attributen für die Höhe und die Breite übergeben. Auch für die komplexeren Regeln in Class 3 ist es unabdingbar, dass die zu prüfenden

Bauteile einfach durch die Prüfregel identifiziert werden können, z.B. über den *IfcObjectType* oder den Bauteilnamen.

Die Überprüfung der Soll-Lage der Öffnungen kann mit einer aufwändigen geometrischen Berechnung oder vereinfacht über die Erstellung von Hilfsobjekten, welche die Soll-Lage abbilden, erfolgen. Diese Soll-Körper können entweder direkt in der Prüfsoftware berechnet und modelliert werden oder müssen in einem vorangehenden Pre-Processing des Modells in einer anderen Software erzeugt und in die Prüfsoftware exportiert werden.

Für die Class 3-Regeln der Brückenrichtlinien sind einige aufwändige geometrische Operationen nötig, wie die Projektion von Objektkanten auf eine darunterliegende Fläche. Diese können möglicherweise nicht von allen kommerziellen Modellprüfungstools umgesetzt werden. Daher ist stattdessen gegebenenfalls ein Ergänzen des Modells mit einfachen Hilfskörpern in der BIM-Authoring Software nötig, um die entsprechenden Regeln auch ohne komplexe geometrische Operationen überprüfen zu können.

5.3.4 Class 4

Als Beispiel für die Class 4 wird RE-Ing Teil 2 Abschnitt 3 – 3.1.2 (1) betrachtet. Dieser Paragraph besagt, dass alle begehbaren Hohlräume in Widerlagern, Pfeilern und Überbauten zugänglich sein müssen. (BMDV 2022b)

Es wird nicht konkret angegeben, wie diese Zugänglichkeit erreicht werden kann. Die Zugänglichkeit muss bei der Prüfung durch den Planer mit dessen Fachwissen beurteilt werden. Jedoch werden in der RE-Ing und den Richtzeichnungen weitere Anforderungen an Hohlräume und Türen gestellt, die Anhaltspunkte für Randbedingungen in den Prüfregeln geben. Zudem gibt es viele unterschiedliche Möglichkeiten, um die Zugänglichkeit zu gewährleisten, welche stark von der Umgebung der Brücke und der jeweiligen Konstruktion abhängig sind.

Beispielsweise ist es oft notwendig eine begehbare Standfläche auf einem Pfeiler auszubilden, welche zur Inspektion der Lager dient. Diese kann beim Vorliegen eines Hohl-Pfeilers vom Pfeiler aus betreten werden und beim Vorliegen eines Hohlkastenquerschnitts durch eine Öffnung im Hohlkasten. Bei massiven Querschnitten und massiven Pfeilern muss für das Betreten eine Berme oder eine Stellfläche für eine mobile Einrichtung oder ein Laufsteg geplant werden. (RE-Ing Teil 2 Abschnitt 3 – 3.1.2 (9d))

Eine Prüfregel für den Nachweis der Zugänglichkeit muss alle Möglichkeiten abdecken und erweitert werden, falls eine neue Lösung gefunden wird. Außerdem müssen in der Prüfregel alle Informationen hinterlegt werden, die nötig sind, um begehbare Hohlräume zu identifizieren, die dann auf Zugänglichkeit geprüft werden können.

Im Rahmen dieser Arbeit wird nur ein kleiner Ausschnitt der zugänglichen Hohlräume betrachtet. Für die Beispielregel wird nur die Zugänglichkeit der Standfläche auf dem Pfeiler über eine Bodenöffnung im Hohlkasten, welcher über ein Widerlager betreten werden kann, evaluiert.

Grundsätzlich müssen für die *Begehbarkeit* von Hohlräumen Mindestabmessungen eingehalten werden:

- Der Innenraum des Hohlkastens muss mindestens 1m breit und 2m hoch ausgebildet werden. (RE-Ing Teil 2 Abschnitt 3 – 3.4 (1))
- Wenn im Hohlkasten Querträger vorliegen, müssen diese Öffnungen mit $b/h \geq 0,94/1,97m$ enthalten. Die Mindestmaße gelten dabei für alle Türen und senkrechten Durchgänge, die für eine Begehbarkeit notwendig sind. (RE-Ing Teil 2 Abschnitt 3 – Tabelle 2.3.1)
- Wenn die Pfeilerköpfe vom Hohlkasten aus betreten werden müssen, muss eine Durchstiegsöffnung von mindestens 80/80cm vorliegen und im Pfeilerkopf muss unterhalb der Durchstiegsöffnung eine Aussparung vorhanden sein, so dass die Standfläche auf dem Pfeilerkopf mindestens 2 m von der Unterkante des Überbaus entfernt ist. (RiZ-Ing Zug 1 Blatt 1)

Um die *Zugänglichkeit* zu den Hohlräumen zu gewährleisten, müssen unter anderem einige Bauwerksbestandteile vorliegen:

- Die Flächen unter der Brücke müssen über Zufahrtswege erreichbar sein und mit Aufstellflächen für Geräte ausgestattet sein. Diese können auch an einem unter der Brücke verlaufenden Verkehrsweg angeschlossen sein, z.B. auf dem Seitenstreifen. (RE-Ing Teil 2 Abschnitt 3 – 3.1.1 (1))
- Der Zugang zur Brücke vom Zufahrtsweg erfolgt über Zugangswege beziehungsweise Böschungstreppe, wenn die Böschungsneigung diese erfordert. (RE-Ing Teil 2 Abschnitt 3 – 3.1.1 (6))

- Um die begehbaren Hohlräume in Überbauten vom Widerlager aus betreten zu können, ist beispielsweise ein ausreichend großer Wartungsgang im Widerlager notwendig, welcher über eine Tür zum Zugangsweg und einen Durchgang zum Hohlkasten verfügt. (RE-Ing Teil 2 Abschnitt 3 – 3.1.2 (3))

Mit der Prüfregel wird einerseits geprüft, dass die Standfläche tatsächlich als begehbar ausgebildet ist und andererseits, dass auf dem Weg von draußen bis zum Pfeilerkopf alle Mindestabmessungen eingehalten sind, die für eine Zugänglichkeit vorgegeben werden.

5.4 Zusammenfassung

Die Implementierbarkeit der in Kapitel 4.3 erläuterten Richtlinien für Brückenentwürfe im Straßenbau wird mit einer Klassifizierung der einzelnen Regeln und einer beispielhaften Übersetzung einzelner Regeln aller Kategorien evaluiert.

Die Klassifizierung erfolgt nach vier Regelkategorien mit steigender Komplexität, die Solihin und Eastman in ihrer Veröffentlichung von 2015 vorstellen. Diese reichen von einfachen Attributprüfungen bis zur Analyse mehrerer Lösungsvarianten für Richtlinien mit großem Interpretationsspielraum.

Aus der Kategorisierung der Richtlinien ergibt sich, dass die überwiegende Mehrheit der Vorgaben für Brückenentwürfe sehr viele konkrete Anforderungen enthalten, die in einfachen Regeln ausgedrückt werden können. Für einige Anforderungen müssen jedoch komplexe Regeln formuliert werden, die domänenspezifisches Fachwissen enthalten.

Sowohl für einfache Attributprüfungen als auch für komplexe Regeln ist es notwendig, die betreffenden Objekte im Modell zu identifizieren. Das kann entweder über den IFC4x3-Objektyp oder den Namen des Objekts erfolgen. Für die erste Variante ist ein korrektes Mapping der Modellobjekte zu IFC4x3 erforderlich, für die zweite Variante die Einhaltung einer strikten Namenskonvention. Anschließend kann die jeweilige Regel für die betreffenden Objekte formuliert werden.

Die gewählten Beispiele aus den Richtlinien können in computer-interpretierbare Regeln übersetzt werden. Diese werden im nächsten Kapitel in der Software DESITE md pro implementiert und an einem Beispielmodell getestet.

6 Implementierung einzelner Regelsätze

In diesem Kapitel werden die erarbeiteten Regeln in Software implementiert. Die Implementierung soll hauptsächlich der Überprüfung der Regelkategorisierung dienen.

6.1 Einblick in die verwendete Software


In diesem Abschnitt werden die verwendeten Software-Applikationen kurz vorgestellt und die Aufgaben erläutert, für die die einzelne Software verwendet wurde. Alle erwähnten Modelle und weiteren Dateien werden im digitalen Anhang beigefügt. Anhang B gibt einen Überblick über den digitalen Anhang.

Autodesk Civil 3D

Für die Modellierung des Geländes, der Straßen und des Brückenüberbaus, bestehend aus einem Hohlkastenquerschnitt und zwei Kappen, wird die Software *Autodesk Civil 3D* in der Version 2022.2 verwendet, ein BIM-Authoring-Tool mit Schwerpunkt Infrastruktur (Autodesk 2023b). Der Export des Modells in IFC4x3 erfolgt über die Erweiterung *IFC 4.3 Extension Update 2 for Autodesk Civil 3D 2022* (Autodesk 2023a).

Beim IFC4x3-Export wird festgestellt, dass die modellierten Straßen- und Brückenelemente alle als *IfcBuiltElement* ausgegeben werden. Durch ein Mapping können der Hohlkasten, die Kappen und die Deckschichten jedoch passenden IFC-Entities zugeordnet werden (vgl. Abbildung 11). Dabei entspricht der Hohlkasten dem Code *Vollquerschnitt*, die Kappen dem Code *Vollquerschnitt Kappe* und die Deckschichten dem Code *Pave1*. Das Mapping entspricht nicht den Modellanforderungen gemäß Anhang A, da nicht alle dort geforderten IFC4x3-Entities in der Civil-IFC4x3-Schnittstelle implementiert sind. Es ist zudem nicht möglich in der IFC anhand der Eigenschaften der Deckschichten festzustellen, ob die jeweilige Deckschicht zur Straße auf der Brücke oder unter der Brücke gehört. Die dafür zusätzlich benötigten Civil-internen Eigenschaften können mit der IFC4x3-Schnittstelle nicht ohne erheblichen Aufwand exportiert werden. Mit der internen IFC2x3-Schnittstelle werden die internen Eigenschaften automatisch mit exportiert. Für die Regelimplementierung der in Kapitel 5.3 ausgewählten Regeln ist IFC4x3 nicht zwingend notwendig. Es wird daher entschieden, für die Validierung der Regelkategorisierung die Regeln auf IFC2x3-Modell anzuwenden.

Die Straßenbestandteile können direkt aus Civil in IFC2x3 exportiert werden. Für die Brücke ist eine detailliertere Modellierung nötig, welche mit Revit 2022 umgesetzt wird. Das Modell in Civil wird dementsprechend angepasst, so dass es nur noch das Gelände, die Straßen und den Brückenbelag enthält. Das aktualisierte Modell wird abschließend in IFC2x3 exportiert und in Revit und die Prüfsoftware DESITE md pro importiert.



```
ifcInfraExportMapping.json - Editor
Datei Bearbeiten Format Ansicht Hilfe
"MapShapeCode": [
  {
    "Code": "Vollquerschnitt",
    "IfcExportAs": "IfcBeamTypeEnum.SUPERSTRUCTURE",
    "Export": true
  },
  {
    "Code": "Vollquerschnitt Kappe",
    "IfcExportAs": "IfcBeamTypeEnum.EDGEBEAM",
    "Export": true
  },
  {
    "Code": "Pave1",
    "IfcExportAs": "IfcElementAssembly.PAVEMENT",
    "Export": true
  },
],
```

Abbildung 11: Auszug aus der Mapping-Datei für den IFC4x3-Export aus Civil 3D (Quelle: Eigene Darstellung)

Autodesk Revit

Für die detaillierte Modellierung der Brücke wird das BIM-Authoring-Tool *Autodesk Revit* in der Version 2022.1 gewählt (Autodesk 2023c). In dieser Version ist ein Im- und Export von IFC4x3-Dateien nicht möglich, weshalb ursprünglich Civil 3D für die Modellierung gewählt wurde. Wegen der oben genannten Probleme beim Export werden die Prüfmodelle jedoch ohnehin in IFC2x3 in DESITE übergeben, weshalb die detaillierte Modellierung in Revit erfolgen kann. Revit bietet dabei den Vorteil einer einfachen Attribuierung, welche mit einem Property Set schnell und einfach in IFC2x3 überführt werden kann.

Zunächst wird das IFC2x3-Modell der Straßen importiert. Ausgehend vom importierten Brückenbelag erfolgt in Revit die detaillierte Modellierung der Brücke mit Fahrbahnplatte, Hohlkasten, Querträgern, Öffnungen und Unterbauten sowie der Brückenausstattung. Die einzelnen Objekte werden mit allen für die Regeln benötigten Attributen modelliert, z.B. der zugehörigen Richtzeichnung oder der Öffnungsbreite.

Da der Export von Modellen in IFC4x3 aus Revit nicht möglich ist, wird zusätzlich ein Attribut in IFC übergeben, in welchem die Entsprechung des Objekts in IFC4x3 angegeben wird. Dieses Attribut wird später verwendet, um eine Identifizierung von Objekten über den IFC4x3-Objekttyp zu demonstrieren. Das Modell wird anschließend für die Übergabe der Daten in die Prüfsoftware in IFC2x3 exportiert. Die benötigten Attribute werden mit einem entsprechend erstellten Property Set in die IFC2x3-Datei überführt.

Thinkproject DESITE md pro

Die Implementierung der erarbeiteten Regeln geschieht in der Software *DESITE md pro* von Thinkproject in der Version 3.0.2., die bereits unter Kapitel 3.5.2 vorgestellt wird (Thinkproject 2022).

Das Programm bietet die Möglichkeit mithilfe einer gut dokumentierten und zugänglichen API die erarbeiteten Regeln als Skripte in der Programmiersprache *Java Script* auszudrücken (Thinkproject 2023). Die Skripte können vom Anwender vollständig eingesehen und somit nachvollzogen werden.

Die Geometrieanalysen in Desite basieren auf der *Bounding Box* beziehungsweise der *Object Oriented Bounding Box* (OOBB) der Objekte, was zu Ungenauigkeiten bei der Analyse gekrümmter Bauteile führt. Auch geometrische Operationen wie die Projektion von Kanten eines Objekts auf eine Fläche sind nicht durch die API-Funktionen abgedeckt. Daher wird für die Überprüfung der Lage der Bodenöffnungen des Hohlkastens mit Hilfskörpern gearbeitet. Es ist dabei nicht möglich geometrische Objekte temporär oder dauerhaft in DESITE zu erstellen, beispielsweise um die Soll-Lage einer Öffnung abzubilden. Für Regeln, die Hilfsobjekte benötigen, muss daher zur Vorbereitung der Modellprüfung ein IFC-Modell mit den nötigen Hilfskörpern in einer BIM-Authoring-Software erstellt werden.

6.2 Erläuterung des Beispielmodells

Für die Modellprüfung mit den implementierten Regeln wird ein vereinfachtes Testmodell für ein fiktives Neubauprojekt erstellt (Abbildung 12).

Die fiktive geplante Brücke entspricht einer Hohlkastenbrücke mit 3 Feldern mit Spannweiten zwischen 70 und 75 m über ein Tal und eine Autobahn. Über die Brücke führt eine zweispurige Staatsstraße und auf der nördlichen Kappe verläuft ein Geh- und Radweg. Es wird angenommen, dass die Streckenachse der Autobahn im Brückenbereich als Gerade mit einer Längsneigung zwischen 3,12% und 1,87% verläuft. Der Straßenquerschnitt der Autobahn enthält zwei mit Betonleitwänden baulich getrennte Fahrstreifen mit je zwei Fahrspuren und einem Seitenstreifen. Die Querneigung beträgt 2,0% und ist jeweils nach außen geneigt, um anfallendes Wasser in die außen angeordneten Entwässerungsmulden abzuleiten. Für die überführte Staatsstraße wird im Brückenbereich im Lageplan ein Radius von 2700m und im Höhenplan eine konstante Längsneigung von 1,85% angenommen.

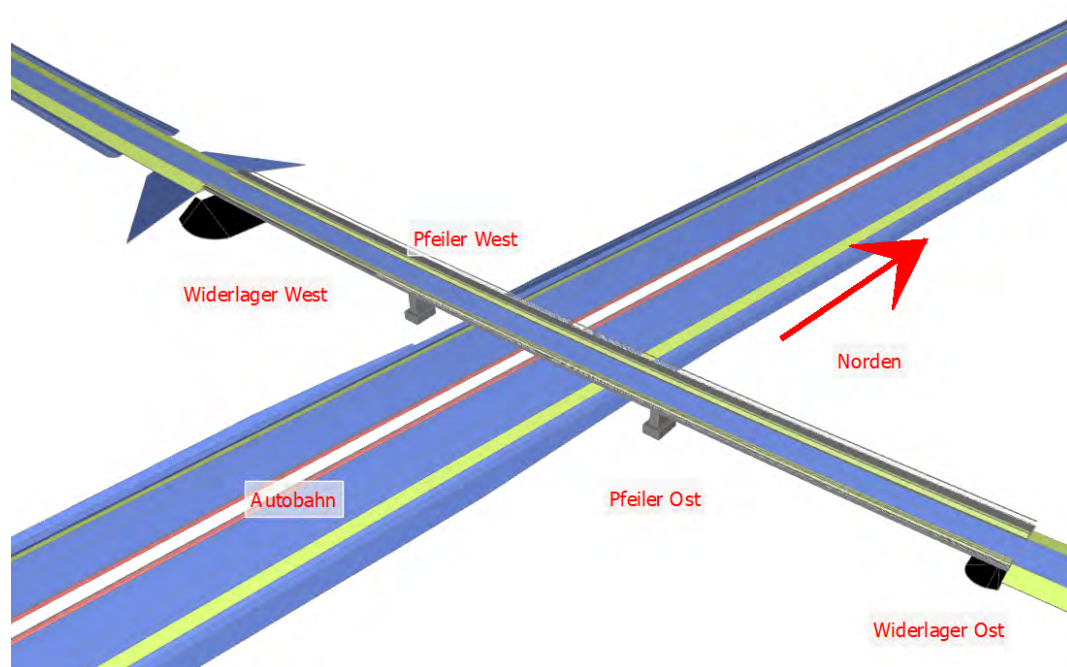


Abbildung 12: Beispielmodell "Hohlkastenbrücke" in Desite md pro (Quelle: Eigene Darstellung)

Das Modell der Brücke enthält einen vollständigen Überbau inklusive Geländer und Schutzeinrichtung und zwei vom Hohlkasten aus begehbare Massivpfeiler. Es wurde auf eine Modellierung der Widerlager inklusive Bermen und Böschungstrepfen verzichtet, da diese für die implementierten Regeln nicht relevant sind. Der Querschnitt des Brückenüberbaus ist in Abbildung 13 dargestellt.

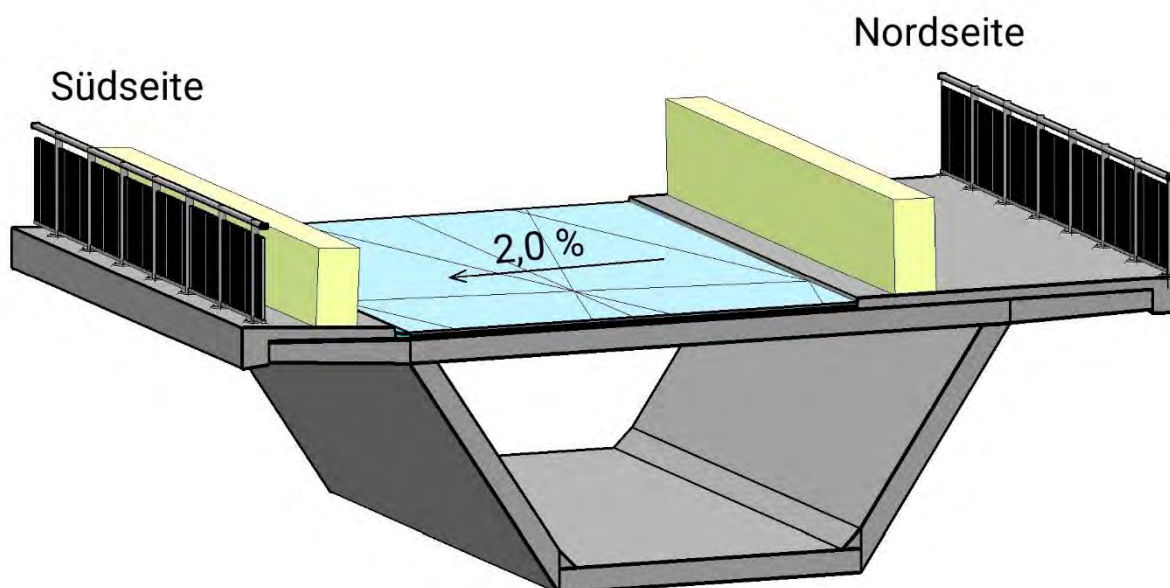


Abbildung 13: 3D-Darstellung des Überbauquerschnitts in Autodesk Revit 2022 (Quelle: Eigene Darstellung)

Der Brückenquerschnitt folgt den Vorgaben aus der Straßenplanung. Diese gibt im Beispielprojekt den Aufbau und die Lage des Straßenbelags vor und ist in Abbildung 13 hellblau dargestellt. Die aus der Straßenplanung resultierende Querneigung im Brückenbereich beträgt 2,0% und fällt in Richtung der Südseite. Der gesamte Querschnitt ist in Stahlbeton geplant. Die Fahrbahnplatte ist 8,05m breit und 30cm dick. Der Innenraum Hohlkastens ist unten ca. 3,90m breit und hat eine lichte Höhe von 2,70m. Die Umfassungsbauteile des Hohlkastens sind 30cm dick. Die Kappe, das Geländer, die Schutzeinrichtung (gelb) und der Kragarm der Fahrbahnplatte auf der Südseite entsprechen der Richtzeichnung Kap 1 Blatt 1. Auf der Nordseite gilt analog Kap 1 Blatt 3, da dort ein Geh- und Radweg verläuft. Die Geländer sind als Füllstabgeländer gemäß Gel 4 ausgebildet. (BASt 2022)

Die Massivpfeiler sind mit $b/h = 2,0/4,0$ m geplant und enthalten an der Oberkante eine Aussparung mit $b/h/t = 1,6/2,0/1,6$ m, die der Besichtigung der Lager dient. Über den beiden Pfeilern sind jeweils 2m breite Wände mit Durchgängen als Querträger angeordnet. An den beiden Enden des Hohlkastens sind 1m breite Wände mit Durchgängen als Querträger geplant. In der Bodenplatte des Hohlkastens sind insgesamt sechs Öffnungen enthalten, von denen mindestens fünf gemäß den Richtzeichnungen Zug 1 Blatt 1 für die Pfeiler und Zug 1 Blatt 2 für den Hohlkasten benötigt werden, weshalb sie den entsprechenden Anforderungen an Abmessungen und Lage entsprechen müssen.

Da das Modell zum Testen der implementierten Regeln verwendet wird, enthält es einige Fehler, welche in der Modellprüfung detektiert werden sollen. Einige der modellierten Öffnungen in der Hohlkasten-Bodenplatte und den Querträgern liegen entweder an der falschen Stelle oder mit falschen Abmessungen vor.

Im Rahmen des Exports des Modells nach IFC2x3 wird festgestellt, dass es nicht möglich ist, die modellierten Deckenöffnungen in einer quer- und längsgeneigten sowie gekrümmten Decke als eigene Öffnungsobjekte mit zusätzlichen Attributen zu exportieren. Um dennoch die Lage der Öffnungen in der X/Y-Ebene und die Abmessungen der Öffnungen mit den entsprechenden Soll-Werten vergleichen zu können, wird ein Hilfsmodell erstellt (Abbildung 14).

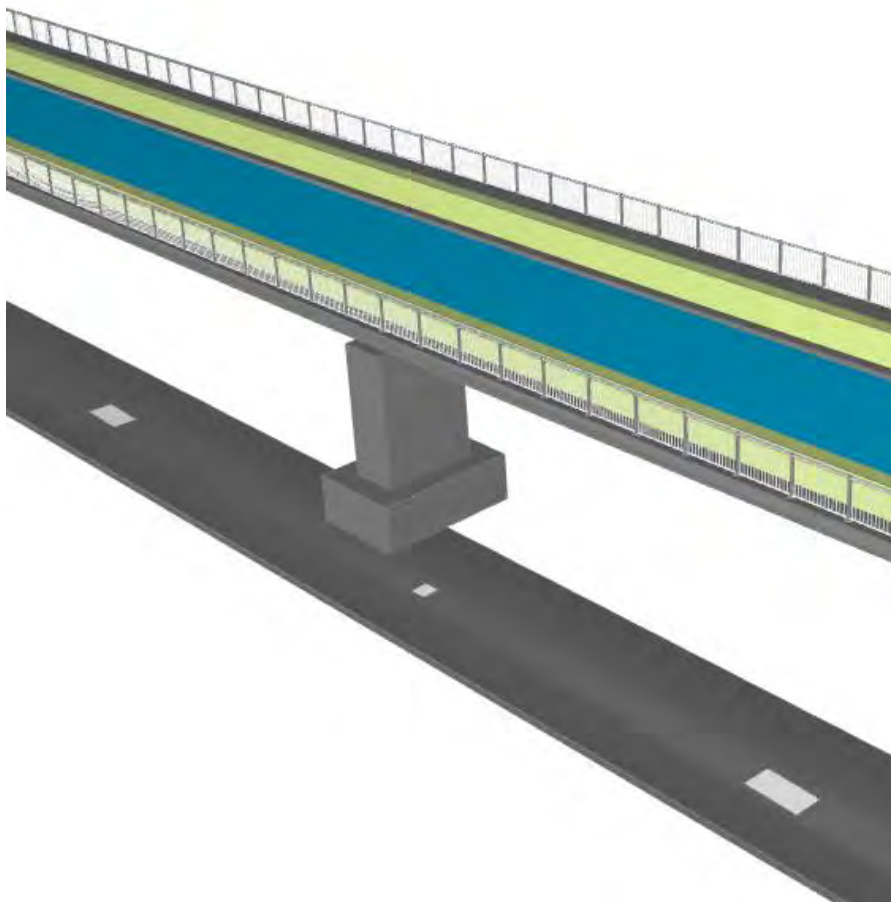


Abbildung 14: Darstellung der in X/Y-Ebene projizierten Hohlkasten-Bodenplatte (unten) im Vergleich zum Beispielmmodell in Desite md pro (Quelle: Eigene Darstellung)

In diesem wird die betroffene Decke, die Bodenplatte des Hohlkastens, zunächst in eine tiefer liegende X/Y-Ebene projiziert. Anschließend werden die beiden Längskanten, die korrekt als Kreissegmente vorliegen, durch Geraden ersetzt. Auf diese Weise können die Öffnungen als eigenständige Öffnungsobjekte exportiert werden, jedoch

nur ohne zusätzliche Attribute wie die zugehörige Richtzeichnung oder den Namen. Die Namen der Öffnungen sind jedoch zur Identifizierung der jeweils betroffenen Öffnungen für die Regeln der Class 3 und 4 notwendig. Um die Attribute trotzdem in die IFC übertragen zu können, werden die Öffnungen im Hilfsmodell als allgemeine Objekte nachmodelliert und attribuiert. Die allgemeinen Objekte werden dann über ihren Namen als Öffnungen ausgewiesen. Für die Öffnungen in den senkrechten Wänden ist hingegen kein Hilfsmodell nötig, da diese aus dem ursprünglichen Modell mit allen zusätzlichen Eigenschaften in IFC2x3 exportiert werden können.

Abschließend wird für die Regeln der Class 3 und 4 ein Hilfskörpermodell in Revit 2022 erstellt und in IFC2x3 exportiert (Abbildung 15). Die Hilfskörper *WL West*, *Seitenstreifen West*, *Seitenstreifen Ost* und *WL Ost* werden für die Class 3-Regel benötigt, um die Soll-Lage der Hohlkastenöffnungen vor den Widerlagern und oberhalb einer Zugangsstraße bzw. an die unterführte Straße angrenzenden Stellfläche, hier der Seitenstreifen, abzubilden. Sie sind bis zur tiefer liegenden Hilfsdecke modelliert. Für Überprüfung der Zugänglichkeit der Pfeiler bei der Class 4-Regel werden für jeden Pfeiler zwei Hilfskörper benötigt. Der kleinere Hilfskörper (bei Pfeiler Ost in Abbildung 15 gelb, bei Pfeiler West verdeckt) bildet die Standfläche auf dem Pfeiler ab, der größere (blau) die Soll-Lage der Durchstiegsöffnung vom Hohlkasten in den Pfeiler. Die Begehbarkeit des Hohlkastens wird mit dem Hilfskörper *Hohlkasten Mindestmaß* nachgewiesen. Für diesen Hilfskörper wird ein Sweep in Längsrichtung der Brücke erzeugt.

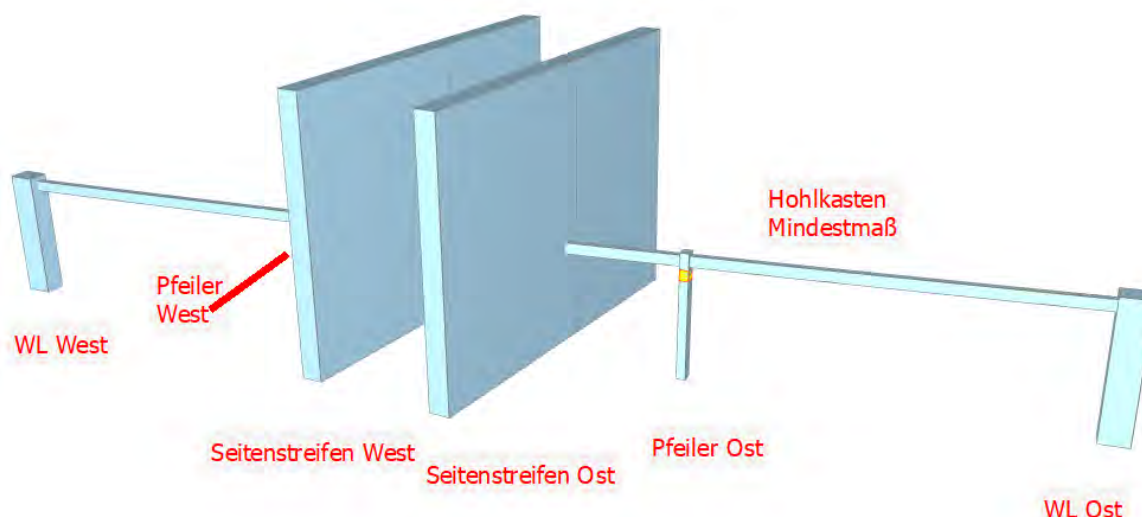


Abbildung 15: Hilfskörpermodell in Desite md pro (Quelle: Eigene Darstellung)

Nachdem alle benötigten Modellbestandteile vorliegen, kann mit der Implementierung der Regeln begonnen werden.

6.3 Implementierung

Nachfolgend wird für die einzelnen Regelkategorien die Implementierung der Beispielregeln über Java Script – Skripte in DESITE md pro erläutert. Die Funktionen der API von Desite werden dabei mit „desiteAPI.[Name der Funktion]“ aufgerufen und sind online dokumentiert (Thinkproject 2023).

6.3.1 Class 1

Die Beispielregel prüft, ob bei allen Kappen das Material „Ortbeton“ hinterlegt ist. Zunächst werden alle Modellobjekte abgerufen und die Kappenobjekte herausgefiltert. Die Kappen sind in IFC4x3 die einzigen Objekte, die mit dem Objekttyp *IfcBeamTypeEnum.EDGEBEAM* beschrieben werden, weshalb dieser zur Identifizierung der Kappen verwendet wird. Im vorliegenden IFC2x3-Modell wird dies über ein für diesen Zweck zugeordnetes Attribut *IFC4X3_ENTSPRECHUNG* im Property Set *MA_Pset_IFC4X3* demonstriert (Abbildung 16, Zeile 5). Auch der Baustoff wird im Beispielmodell mit einem eigens zugeordneten Attribut *BAUSTOFF* im Property Set *MA_Pset_Modellanforderungen* in IFC übergeben (Abbildung 16, Zeile 11). Der Wert des Attributes *BAUSTOFF* wird für alle gefundenen Kappen mit dem Sollwert „Ortbeton“ verglichen. Stimmen die Werte überein, wird ein Faktor hochgezählt, stimmen sie nicht überein, wird die ID der jeweiligen Kappe in eine Liste geschrieben (Abbildung 16, Zeilen 6-17). Zum Schluss wird das Ergebnis der Prüfung über die Konsole ausgegeben. Die Regel ist nicht erfüllt, wenn keine Kappen gefunden wurden (Zeile 19f) oder der hochgezählte Faktor nicht mit der Anzahl der gefundenen Kappen übereinstimmt. Im zweiten Fall werden in Desite nur die Kappen angezeigt, die die Regel nicht erfüllen (Zeile 23ff), so dass aus dieser Ansicht ein BCF mit der Bitte um Korrektur erstellt und an den BIM-Autor versendet werden könnte. Erfüllen alle Kappen die Regel, wird eine positive Rückmeldung angezeigt und die Ansicht des Modells in Desite bleibt unverändert (Zeile 21f).

```

1 // created 2023-05-17T14:43:01
2 // by kraus
3
4 var alle = desiteAPI.getAllElements("geometry");
5 var Kappen = desiteAPI.filterByProperty(alle, "MA_Pset_IFC4x3:IFC4X3_ENTSPRECHUNG",
6 "xs:string", "IfcBeamTypeEnum.EDGEBEAM", false);
7 var counter = 0;
8 var falseElements = [];
9
10 for (i=0; i < Kappen.length; i++)
11 {
12     var Material = desiteAPI.getPropertyValue(Kappen[i],
13 "MA_Pset_Modellanforderungen:BAUSTOFF", "xs:string", false);
14     if (Material == "ortbeton") {
15         counter = counter + 1;
16     } else {
17         falseElements.push(Kappen[i]);
18     }
19 }
20
21 if (Kappen.length == 0) {
22     console.log("Fehler! Keine Kappen gefunden!");
23 } else if (counter == Kappen.length) {
24     console.log("Regel erfüllt: Alle Kappen sind aus ortbeton geplant");
25 } else {
26     console.log("Die angezeigten Kappen erfüllen die Regel nicht.");
27     desiteAPI.showElementsOnly(falseElements);
28 }

```

Abbildung 16: Desite-Skript für Beispielregel Class 1 (Quelle: Eigene Darstellung)

6.3.2 Class 2

Für die Class 2 wird eine Regel implementiert, die die Einhaltung der lichten Höhe über der unterführten Straße überprüft (Abbildung 17). Für das Beispielmodell gilt dabei ein Soll-Wert von 4,7m, da die unterführte Straße eine Autobahn ist.

Zunächst werden die betroffenen Straßen- und Brückenbestandteile in Variablen hinterlegt. Für die Brückenbauteile wird dabei nach dem Attribut *NAME* im Property Set *MA_Pset_Modellanforderungen* gefiltert. Die Bauteile im IFC-Modell sind eindeutig benannt, so dass die Bodenplatte des Hohlkastens über ihren Namen *UEB_Hohlkasten_Bodenplatte* eindeutig identifiziert werden kann (Abbildung 17, Zeile 11). Für eine eindeutige Identifizierung der Asphalt-Deckschichten der Straße unter der Brücke sind zwei Filterschritte notwendig. Im ersten Schritt werden alle Bestandteile der unterführten Straße herausgefiltert. Alle Bestandteile gehören im Civil-Modell zum selben Profilkörper „Straße unten“. Diese Information ist im Attribut *3DProfilkörpername* hinterlegt. Es werden alle Objekte herausgefiltert, deren Wert mit „unten“ endet, was in JavaScript mit einem Stern (*) gekennzeichnet wird (Zeile 7). Für die Berechnung der lichten Höhe dürfen nur die relevanten Straßenbestandteile, hier die Fahrspuren, die Seitenstreifen und die Randstreifen, beachtet werden. In Civil werden die Deckschichten dieser Bauteile als „Pave1“ bezeichnet. Deshalb werden im zweiten Filterschritt nur Objekte, bei denen das Attribut *Codename* den Wert „Pave1“ enthält aus den Objekten der unterführten Straße herausgefiltert (Zeile 9).

```

1 // created 2023-05-17T12:36:45
2 // by kraus
3 // Regel ermittelt Lichte Höhe zwischen OK Straße und UK überbau und vergleicht wert mit SOLL =
  4,7m
4
5 var alle = desiteAPI.getAllElements();
6 //alle Bestandteil des Profilkörpers der unteren Straße
7 var allRoadElementsBelow = desiteAPI.filterByProperty(alle,"3D-Profilkörper-
  Modellinformationen:3DProfilkörpername", "xs:string", "*unten", false);
8 //Filter auf Straßenbestandteile mit asphalt (Fahrstreifen, Randstreifen, Seitenstreifen)
9 var roadElements = desiteAPI.filterByProperty(allRoadElementsBelow, "Informationen zur
  Profilart des 3D-Profilkörpers:Codename", "xs:string", "Pavel", false);
10
11 var bridgeElements = desiteAPI.filterByProperty(alle, "MA_Pset_Modellanforderungen:NAME",
  "xs:string", "UEB_Hohlkasten_Bodenplatte", false);
12
13 var minDist = 100000.0;
14 var roadIndex = -1;
15 var bridgeIndex = -1;
16
17 for (i = 0; i < roadElements.length; i++) {
18   for (j = 0; j < bridgeElements.length; j++) {
19     var dist = desiteAPI.calcDistance(roadElements[i], bridgeElements[j])
20
21     if (dist > -1 && dist != 0.0 && dist < minDist) {
22       minDist = dist;
23       roadIndex = i;
24       bridgeIndex = j;
25     }
26   }
27 }
28
29 }

```

Abbildung 17: Auszug aus dem Desite-Skript der Beispielregel für Class 2 (Quelle: Eigene Darstellung)

In einer doppelten Schleife werden mit der Funktion `desiteAPI.calcDistance` die Abstände zwischen allen Brückenelementen und allen Straßenelementen berechnet (Zeile 21). Um die insgesamt kleinste Distanz zu ermitteln, wird der berechnete Wert „dist“ mit dem Mindestwert „minDist“ verglichen (Zeile 23). Dabei wird sichergestellt, dass die berechnete Distanz weder negativ noch null ist. Wenn der berechnete Wert kleiner als der Mindestwert ist, wird der Mindestwert mit dem berechneten Wert überschrieben und der Index des zugehörigen Brückenelements sowie des zugehörigen Straßenelements gespeichert. Nachdem alle Elemente berücksichtigt sind, entspricht der Mindestwert der gesuchten minimalen lichten Höhe.

Die minimale Höhe wird anschließend mit dem Soll-Wert verglichen und das Ergebnis der Prüfung auf der Konsole ausgegeben. Mit den gespeicherten Indices der Brücken- und Straßenelemente werden die beiden kritischen Elemente isoliert, damit der Berater schnell erkennen kann, welche Elemente für die kritische Höhe maßgebend sind.

Zur Überprüfung der Berechnung wird das Desite-Skript anschließend auf ein IFC2x3-Modell eines realen Bauprojekts angewendet (vgl. Anhang C, Screenshot 1). Das Bauprojekt umfasst eine Unterführung eines Radwegs, weshalb hier nur eine lichte Höhe von 2,50m eingehalten werden muss. Da die Objekte im zugehörigen Modell anders benannt sind als im Beispielmmodell, werden die Objektfilter entsprechend angepasst.

Im Verlauf der Planung wurde eine lichte Höhe von 2,611m ermittelt und als Parameter in einem Informationsbauteil hinterlegt (Screenshot 2). Die Berechnung über das Desite-Skript ergibt einen Wert von circa 2,6024m (Screenshot 3). Daraus ergibt sich eine Abweichung von etwa 8,6mm, welche vermutlich aus den unterschiedlichen Berechnungsmethoden entsteht. Für den Entwurf ist die Abweichung vertretbar, weshalb davon ausgegangen wird, dass die Berechnung der lichten Höhe mit der Desite-Funktion *desiteAPI.calcDistance* brauchbare Ergebnisse liefert.

6.3.3 Class 3

Mit der Beispielregel für Class 3 werden die Anordnung und die Abmessungen der Hohlkasten-Bodenöffnungen mit den Vorgaben aus der Richtzeichnung Zug 1 Blatt 2 verglichen (vgl. BASt 2022).

Die korrekte Lage der Öffnungen wird über einen geometrischen Abgleich zwischen einem Körper, der die Soll-Lage repräsentiert, und dem jeweiligen Öffnungsobjekt überprüft. Wenn das Öffnungsobjekt vollständig im Soll-Körper enthalten ist, ist die Lage des Öffnungsobjekts korrekt. Für Zug 1 Blatt 2 ist ein Soll-Körper über der unterführten Straße und ein Soll-Körper für jedes Widerlager nötig (vgl. auch Abbildung 15). Die Prüfredel besteht aus zwei Teilschritten und einer nachfolgenden Gesamtprüfung. Im ersten Schritt werden die bestehenden Hohlkastenöffnungen daraufhin analysiert, ob mindestens eine Öffnung mit $b/h \geq 1,2/2,5\text{m}$ oberhalb einer Zufahrtsstraße oder befahrbaren Stellfläche liegt. Es wird angenommen, dass die Seitenstreifen der Autobahn für das Entladen von Materialien verwendet werden dürfen, weshalb sie als Stellflächen gelten. Daher werden in Revit zwei Quader als Hilfskörper erzeugt, mit denen die Fläche des jeweiligen Seitenstreifens im Brückenbereich so nach oben und unten gezogen wurde, dass der Hilfskörper sowohl die vollständige Brücke als auch die ebene Hilfsdecke schneidet. Die Hilfskörper sind analog zu den anderen Objekten im Brückenmodell attribuiert und eindeutig benannt.

Die Filterung nach den betroffenen Objekten erfolgt analog zu den Brückenelementen in Class 2 nach dem Objektnamen. Die Information, ob die Öffnung in einem der beiden Hilfskörper liegt, wird mit der Desite-Funktion *desiteAPI.IsIncludedIn* ermittelt (Abbildung 18, Zeile 20f). Wenn mindestens eine Öffnung in einem Hilfskörper liegt, wird die Einhaltung der korrekten Abmessungen überprüft, indem die jeweiligen Attribute *HOEHE* und *BREITE* mit den Soll-Werten verglichen werden. Es ist zu beachten, dass die Zahlen, die Desite aus dem Modell übernimmt leicht abgerundet werden, z.B.

2,499999999998 statt 2,5, weshalb die Soll-Werte 1,2m und 2,5m auf 1,19m und 2,49m abgerundet werden (Zeile 27). Zudem ist nicht festgelegt welche Seiten der Öffnung mit der Breite und welche mit der Höhe beschrieben werden, weshalb die Soll-Werte für zwei Kombinationen geprüft werden. Das Teilergebnis von Schritt 1 wird für eine gute Nachvollziehbarkeit der Regel durch den Anwender auf der Konsole angezeigt. Für jeden Teilschritt wird zum Schluss eine Variable erzeugt, in der bei einem korrekten Teilergebnis der Boolean-Wert *true* hinterlegt wird und bei fehlerhaftem Teilergebnis der Wert *false* (Zeile 37ff).

```
11 //
12 //1. Schritt: min 1 Öffnung 2,5x1,2m über Straße
13
14 var counter1 = 0;
15 var counter2 = 0;
16
17 for (i = 0; i < OEF.length; i++)
18 {
19   var bool1 = desiteAPI.isIncludedIn(OEF[i], Hilfe1[0]);
20   var bool2 = desiteAPI.isIncludedIn(OEF[i], Hilfe1[1]);
21
22   if (bool1 == true || bool2 == true){
23     counter1 = counter1 + 1;
24     var Breite = desiteAPI.getPropertyValue(OEF[i], "MA_Pset_OEF:HOEHE", "xs:double", false);
25     var Hoehe = desiteAPI.getPropertyValue(OEF[i], "MA_Pset_OEF:BREITE", "xs:double", false);
26     if ((Breite > 1.19 && Hoehe > 2.49) || (Breite > 2.49 && Hoehe > 1.19)) {
27       counter2 = counter2 + 1;
28       console.log("Teilergebnis: Maße und Lage der Bodenöffnung korrekt");
29     } else {
30       console.log("Teilergebnis: Lage der Bodenöffnung korrekt, Maße falsch");
31     }
32   }
33 }
34 }
35 }
36
37 if(counter1 > 0 && counter2 > 0) {
38   console.log("Ergebnis Schritt 1: Min 1 korrekte Bodenöffnung über einer Straße vorhanden");
39   var schritt1passt = true;
40 } else {
41   console.log("Ergebnis Schritt 1: nötige Bodenöffnung über Straße fehlt oder nicht korrekt");
42   var schritt1passt = false;
43 }
```

Abbildung 18: Auszug aus dem Desite-Skript zur Beispielregel in Class 3 - Schritt 1 (Quelle: Eigenen Darstellung)

Für den zweiten Schritt wird angenommen, dass die Öffnung vor dem Widerlager maximal 5,0m von der Außenkante des Hohlkastens entfernt liegen darf, da in der Richtlinie hierfür keine exakten Angaben vorliegen. Somit wird für jedes Widerlager ein Quader mit einer Grundfläche von 5,0m mal der Breite der Hohlkastenbodenplatte als Hilfskörper erzeugt und analog zu den Hilfskörpern der Seitenstreifen ausreichend nach oben und unten verlängert, um die Hilfsdecke zu schneiden. Die Überprüfung der Lage und der Abmessungen erfolgt analog zum 1. Schritt für jedes Widerlager einzeln, mit dem Unterschied, dass die Soll-Abmessungen für die Widerlageröffnungen $b/h \geq 1,0/1,5\text{m}$ betragen.

Um ein Gesamtergebnis für die vollständige Regel zu ermitteln, werden die Variablen mit den Ergebnissen der Teilschritte ausgewertet. Sind alle Teilprüfungen erfüllt, wird ein positives Ergebnis auf der Konsole angezeigt, wenn nicht ein negatives. Auf eine

Aufschlüsselung welcher Teilschritt nicht erfüllt wurde, wird beim Gesamtergebnis verzichtet, da dies bereits durch die Teilergebnisse nachvollzogen werden kann (Abbildung 19).

```
Teilergebnis: Maße und Lage der Bodenöffnung korrekt  
Ergebnis Schritt 1: Min 1 korrekte Bodenöffnung über einer Straße vorhanden  
Teilergebnis: Maße und Lage der Bodenöffnung korrekt  
Ergebnis Schritt 2 WL 1: Korrekte Bodenöffnung vor Widerlager vorhanden  
Teilergebnis: Lage der Bodenöffnung korrekt, Maße falsch  
Ergebnis Schritt 2 WL 2: nötige Bodenöffnung vor Widerlager fehlt oder nicht korrekt  
RiZ Zug 1 Blatt 2 nicht eingehalten!  
undefined
```

Abbildung 19: Ergebnis des Desite-Skripts der Class 3-Regel für das Beispielmodell (Quelle: Eigene Darstellung)

Für das Beispielmodell ist die Regel insgesamt nicht erfüllt, da die Widerlageröffnung vor Widerlager West absichtlich mit falschen Abmessungen modelliert wurde, um ein negatives Testergebnis zu zeigen.

Es wäre sinnvoll auch bei dieser Regel die Objekte, die die Regel nicht erfüllen, im Anschluss an den Prüflauf zu isolieren, worauf im Rahmen der Arbeit jedoch verzichtet wurde.

6.3.4 Class 4

Für die Implementierung der Class 4-Beispielregel wurden einige Annahmen getroffen. Es wird davon ausgegangen, dass eine Zufahrtsstraße zu mindestens einem Widerlager vorhanden ist, von der aus der Hohlkasten über einen Zugangsweg, eine korrekt ausgeführte Böschungstreppe, eine ausreichend große Tür und einen ausreichend großen Wartungsgang betreten werden kann. Diese Bauteile sind nicht im Modell abgebildet und können daher nicht geprüft werden. Die Untersuchung der Zugänglichkeit der Pfeilerköpfe durch den Hohlkasten beginnt ab der Durchstiegsöffnung in der Abschlusswand des Hohlkastens am Widerlager.

Die Zugänglichkeitsprüfung ist in fünf Teilschritte untergliedert. Analog zur Class 3 werden für eine bessere Nachvollziehbarkeit die Teilergebnisse aller Teilschritte auf der Konsole ausgegeben und für jeden Teilschritt eine Boolean-Variable mit dem Teilergebnis erzeugt.

Im ersten Schritt wird die Begehrbarkeit des Hohlkastens ausgewertet, indem ein Hilfskörper mit einer Breite von 1,0m und einer Höhe von 2,0m entlang der Mittellinie die Bodenplatte des Hohlkastens erstellt wird (siehe *Hohlkasten Mindestmaß* in Abbildung 15). In der zugehörigen Teilregel wird überprüft, dass dieser Hilfskörper keines der

Umfassungsbauteile des Hohlkastens durchdringt. Das geschieht mit der Desite-Funktion *desiteAPI.checkIntersection*, welche „0“ ausgibt, wenn keine Durchdringung (engl. *clash*) vorliegt und „1“, wenn ein Clash vorliegt. Im Beispielmodell gibt es sechs Umfassungsbauteile, zwei Wände, eine Bodenplatte, die Fahrbahnplatte und zwei Vouten zwischen Wänden und Bodenplatte. Wie bei den vorherigen Regeln können die betreffenden Objekte aufgrund einer eindeutigen Benennung identifiziert werden.

Beim zweiten Schritt werden die Abmessungen der Durchstiegsöffnungen in den Querträgern geprüft. Die Voraussetzung dafür, dass die Öffnung geprüft werden muss, ist, dass der Hilfskörper *Hohlkasten Mindestmaß* den jeweiligen Querträger durchdringt. Das wird wieder mit *desiteAPI.checkIntersection* geprüft. Liegt eine Durchdringung vor, werden die Öffnungen des jeweiligen Querträgers mit der Desite-Funktion *desiteAPI.getOpenings* abgerufen und die Öffnungsabmessungen mit den Soll-Abmessungen $b/h \geq 0,94/1,97\text{m}$ abgeglichen.

Im dritten Schritt werden die Abmessungen der Öffnungen in den Abschlusswänden des Hohlkastens an den Widerlagern geprüft. Die Öffnungen können aufgrund der gewählten Benennung der Objekte direkt herausgefiltert werden. Die Soll-Abmessungen sind identisch zu Schritt 2.

Nachdem die Begehbarkeit des vollständigen Hohlkastens in Schritt 1 bis 3 geprüft wurden, kann in den nachfolgenden Schritten die Zugänglichkeit der Pfeilerköpfe vom Hohlkasten aus nachgewiesen werden. Für die Zugänglichkeit der Pfeilerköpfe müssen die Anforderungen nach der Richtzeichnung Zug 1 Blatt 1 eingehalten werden (BASt 2022).

Zuerst wird im vierten Schritt geprüft, dass Durchstiegsöffnungen mit $b/h \geq 0,80/0,80\text{m}$ vom Hohlkasten zu den Pfeilern vorhanden sind. Dafür werden analog zur Class 3-Regel Hilfskörper für die beiden Pfeiler erstellt, deren Grundfläche der Aussparung oben am Pfeilerkopf entspricht. Die Überprüfung von Lage und Abmessungen der Durchstiegsöffnungen geschieht analog der Class 3-Regel.

Im letzten Teilschritt werden die Pfeilerköpfe daraufhin untersucht, dass sie eine ausreichende Standfläche und Höhe für eine Begehung aufweisen. Die Standfläche auf dem Pfeiler sollte gemäß dem Objektkatalog in Anhang A als *IfcSpatialZoneTypeEnum.SECURITY* in IFC exportiert werden. Die Modellierung von Räumen auf einer Stütze wird von Revit 2022 nicht unterstützt, weshalb die Standfläche alternativ als

Hilfskörper in der Aussparung mit einer Höhe von 2,0m modelliert und exportiert wird (vgl. gelben Hilfskörper bei Pfeiler Ost in Abbildung 15).

Die Höhe über der Standfläche muss gemäß der Richtzeichnung mindestens 2,0m betragen. Um das zu prüfen, muss einerseits der Hilfskörper eine Höhe von mindestens 2,0m haben (Abbildung 20 Zeile 154: HR_PF_Tiefe) und andererseits sichergestellt sein, dass der Hilfskörper die Bodenplatte des Hohlkastens nicht durchdringt (Zeile 149). Für die Abmessungen im Grundriss der Standfläche gibt es keine Vorgaben, weshalb als Soll-Werte $b/h \geq 1,0/1,0\text{m}$ angenommen werden.

```

139 //
140 // 5. Schritt: Pfeilerköpfe gem. Zug 1 Blatt 1 ausgebildet für Besichtigung der Lager
141 // begehbarer Pfeilerkopf: lichte Höhe 2m (Riz), min 1mx1m Standfläche (eigene Annahme)
142 // begehbarer Raum als Hilfskörper modelliert
143
144 var HR_PF = desiteAPI.filterByProperty(alle, "MA_Pset_Modellanforderungen:NAME", "xs:string",
  "UNT_Hohlraum_Pfeiler*", false);
145 var counter5 = 0;
146
147 if(HR_PF.length == Pfeileranzahl) {
148   for (p=0; p< Pfeileranzahl; p++) {
149     { var clash8 = desiteAPI.checkIntersection(HR_PF[p], HK_BP, -0.00001);
150       if(clash8 == 0) {
151         var HR_PF_Breite = desiteAPI.getPropertyValue(HR_PF[p], "MA_Pset_OEF:BREITE",
  "xs:double", false);
152         var HR_PF_Hoeh e = desiteAPI.getPropertyValue(HR_PF[p], "MA_Pset_OEF:HOEHE", "xs:double",
  false);
153         var HR_PF_Tiefe = desiteAPI.getPropertyValue(HR_PF[p], "MA_Pset_OEF:TIEFE", "xs:double",
  false);
154         if(HR_PF_Breite > 0.999 && HR_PF_Hoeh e > 0.999 && HR_PF_Tiefe > 1.999) {
155           counter5 = counter5 + 1;
156         } else {
157           console.log("Abmessungen begehbarer Raum auf dem Pfeiler zu klein");
158           var bool5 = false;
159         }
160       } else {
161         console.log("Begehbarer Raum auf Pfeiler durchdringt Bodenplatte Hohlkasten!");
162         var bool5 = false;
163       }
164     }
165   if(counter5 == Pfeileranzahl) {
166     console.log("Abmessungen begehbarer Raum auf allen Pfeilern eingehalten");
167     var bool5 = true;
168   }
169 } else {
170   console.log("Anzahl Hilfskörper auf Pfeiler inkorrekt, Prüfung wurde nicht durchgeführt");
171   var bool5 = false;
172 }
173

```

Abbildung 20: Auszug aus dem Desite-Skript der Class 4-Regel – Schritt 5 (Quelle: Eigene Darstellung)

Nachdem alle Teilprüfungen durchgeführt wurden, kann aus den Teilergebnissen ein Gesamtergebnis generiert und auf der Konsole abgebildet werden (Abbildung 21).

```

Hohlkasten ist ausreichend groß
falsche Öffnung im Querträger
Min 1 Durchstiegsöffnung im Querträger fehlt oder hat falsche Abmessungen
min 1 Zugangsöffnung mit korrekten Abmessungen vorhanden
min 1 Durchstiegsöffnung Abmessungen zu klein
Abmessungen begehbarer Raum auf allen Pfeilern eingehalten
Zugänglichkeit nicht vollständig gewährleistet
undefined

```

Abbildung 21: Ergebnis des Desite-Skripts der Class 4-Regel für das Beispielmmodell (Quelle: Eigene Darstellung)

Auch die Class 4-Regel wird vom Beispielmmodell nicht eingehalten, da absichtlich bei einer senkrechten Durchstiegsöffnung im Querträger über Pfeiler West und einer

waagrechten Durchstiegsöffnung oberhalb von Pfeiler Ost falsche Abmessungen modelliert wurden.

6.4 Diskussion der Ergebnisse der Implementierung

Insgesamt zeigt sich, dass alle gewählten Regeln implementiert und erfolgreich auf das Beispielmodell angewendet werden können. Da für alle vier Regelkategorien funktionierende Beispiele implementiert wurden, wird davon ausgegangen, dass die in Kapitel 5.2 untersuchten Richtlinien dementsprechend als funktionierende Regeln in Software ausgedrückt werden können.

Die Voraussetzung für eine funktionierende Modellprüfung ist neben den Regeln vor allem die Modellierung und der Export von vollständigen Modellen. Sowohl für einfache als auch für komplexe Regeln ist die Identifizierung der betroffenen Objekte von höchster Bedeutung. Dafür ist einerseits eine eindeutige und korrekte Benennung der Objekte nötig und andererseits ein korrektes Mapping der Objekte in IFC. Wird mit den IFC4x3-Objekttypen zur Identifizierung gearbeitet, muss zudem sichergestellt sein, dass tatsächlich nur die gesuchten Objekte, z.B. die Kappen, in diesem Objekttyp vorliegen. Gerade für die Regeln, welche die Zugänglichkeit von Bauteilen überprüfen, ist es meistens notwendig, Räume und deren Abmessungen zu analysieren. Das setzt voraus, dass die zu untersuchenden Modelle in der Ursprungssoftware korrekte Raumobjekte enthalten und in IFC als *IfcSpatialZone* exportiert werden können. Mit dem für das Beispielmodell verwendeten Revit 2022 können Räume, die nicht innerhalb eines Hochbau-Gebäudes liegen, nur sehr beschränkt erzeugt werden, weshalb diese Modellanforderung nicht eingehalten werden kann. Die benötigten Räume, z.B. die freie Standfläche auf einem Pfeilerkopf, können nur durch Work-Arounds erzeugt und als *IfcBuildingElementProxy* exportiert werden.

Für die Beispielregeln ist durch die Work-Arounds und Hilfskörper ein Export des Modells in IFC2x3 ausreichend. Jedoch gibt es einige Anforderungen aus den Brückenrichtlinien, welche nur oder einfacher an einem Modell in IFC4x3 geprüft werden können. Das gilt insbesondere für Anforderungen, die die Anordnung von Objekten zueinander oder die Abmessungen der Querschnitte überprüfen, da für diese im Infrastrukturbau die Platzierung der Objekte entlang eines Alignments ausgewertet werden muss. Doch auch bei der Verwendung von IFC4x3 ist eine eindeutige Benennung der Objekte und ein korrektes Mapping unabdingbar.

Die ausgewählten Regeln können mit DESITE implementiert werden. Mit den gut dokumentierten API-Funktionen ist ein Formulieren von Regeln vergleichsweise einfach. Ein weiterer Vorteil ist, dass mit den API-Funktionen die Objekte, welche eine Regel nicht erfüllen, in der Ansicht isoliert werden können. Der Anwender kann auf diese Weise schnell fehlerhafte Objekte erkennen. Anschließend können entweder manuell durch den Nutzer oder automatisiert in den Regeln BCF-Dateien für die Probleme erstellt und exportiert werden. Nachteilig an DESITE ist, dass für die beiden komplexen Regeln der Class 3 und 4 wird ein zusätzliches Modell mit Hilfskörpern benötigt wird, dessen Erstellung einen aufwändigen Zwischenschritt zwischen Modellerstellung und Modellprüfung darstellt. Ein weiteres Problem besteht darin, dass für jedes Skript alle Objekte im Modell aufgerufen werden müssen, um die benötigten Objekte herausfiltern zu können. Bei großen Modellen kann das die Laufzeit der Skripte stark verlängern.

Grundsätzlich lässt sich festhalten, dass die Skripte deutlich aufwändiger zu erstellen sind, wenn die Teilergebnisse dokumentiert werden sollen. Da das aber die Nachvollziehbarkeit für den Anwender deutlich erhöht, ist eine Dokumentation der Teilergebnisse für alle Regeln wünschenswert. Im Gegenzug wird die Implementierung deutlich vereinfacht, wenn möglichst viele Informationen, insbesondere einfache Abmessungen wie die Öffnungsbreite, in Attributen im Modell vorliegen. Die Implementierung wird ebenfalls aufwändiger, wenn die Regel auf alle möglichen Brücken angewendet werden soll. Die Skripte für die Beispielregeln sind teilweise nur auf Modelle gleichwertig zum Beispielmodell anwendbar. Beispielsweise ist die Class 4-Regel auf Brücken mit nur zwei Pfeilern ausgelegt und die Class 2-Regel gilt nur wenn die unterführte Straße eine Fernstraße ist. Für eine breite Anwendung der Regeln müssen die aktuellen Skripte noch deutlich erweitert werden.

6.5 Zusammenfassung

In diesem Abschnitt werden die unter 5.3 erarbeiteten Regeln in DESITE md pro implementiert und auf ein Beispielmodell angewendet.

Das Beispielmodell besteht aus einer Modellierung eines Geländes und zwei sich kreuzenden Straßen in Civil 3D und einer darauf aufbauenden detaillierteren Modellierung einer 3-feldrigen Hohlkastenbrücke in Revit. Es wird ergänzt durch Hilfskörper für die Soll-Lagen der Aussparungen und ein Hilfsmodell der Bodenplatte des Hohlkastens. Alle für die Regelprüfung relevanten Objekte werden eindeutig benannt und über Attribute mit zusätzlichen Informationen wie der zugehörigen Richtzeichnung und der

Entsprechung des Objekts in IFC4x3 angereichert. Wenn möglich, werden in den Objekten ihre geometrischen Abmessungen als Attribute hinterlegt.

Es werden vier Regeln in DESITE md pro implementiert und erfolgreich auf das Beispielmmodell angewendet. Dafür ist eine korrekte Modellierung mit einer eindeutigen Bezeichnung der Objekte erforderlich. Die Regeln werden so formuliert, dass die Teilergebnisse der einzelnen Schritte ausgegeben werden, um den Prüfprozess für den Anwender nachvollziehbar zu machen.

Es zeigt sich, dass alle Regelkategorien implementiert werden können. Daher kann davon ausgegangen werden, dass die analysierten Richtlinien in computer-interpretierbare Regeln überführt werden können. Für die komplexen Kategorien Class 3 und 4 ist jedoch die Integration von Fachwissen in die Regeln erforderlich, so dass diese nur von oder in Kooperation mit Fachleuten des Bauwesens erstellt werden können.

7 Zusammenfassung und Ausblick

7.1 Zusammenfassung

Im Rahmen dieser Arbeit wurde evaluiert wie eine Konformitätsprüfung für Brückenentwürfe des Straßenbaus aufgebaut und durchgeführt werden kann und welche Anforderungen dafür von den zu untersuchenden BIM-Modellen erfüllt werden müssen. Zudem wurde das herstellernerneutrale IFC-Format in der aktuellen Version 4x3 untersucht, um dessen Potential als Format für die Datengrundlage für eine Konformitätsprüfung im Infrastrukturbau abzuschätzen. Zunächst wurde eine Literaturrecherche betrieben, durch welche die grundsätzlichen Bestandteile einer automatisierten Konformitätsprüfung sowie verschiedene Möglichkeiten einer technischen Umsetzung der einzelnen Teile identifiziert wurden. Anschließend wurden aus den Richtlinien Modellanforderungen erarbeitet, welche mit dem Datenschema von IFC4x3 verglichen werden. Danach wurden die Richtlinien hinsichtlich ihrer Formulierbarkeit als computerinterpretierbare Regeln untersucht, indem sie in vier Stufen nach ihrer Komplexität kategorisiert wurden. Abschließend wurden Beispielregeln für alle Kategorien formuliert, in Software implementiert und auf ein Beispielmodell angewendet. Da IFC4x3 bisher nur von wenigen Anwendungen unterstützt wird, wurden die Beispielregeln auf ein IFC2x3-Modell angewendet. Dafür wurden Regeln ausgewählt, für die nicht zwingend ein Modell in IFC4x3 benötigt wird. Die Implementierung dient somit nur als Validierung für die Formulierbarkeit von Regeln aus den Brückenbaurichtlinien und nicht für die erarbeiteten Modellanforderungen.

Insgesamt zeigte sich ein großes Potential von IFC4x3 als Datenformat für die Konformitätsprüfung von Brückenentwürfen. Dies ergab sich sowohl aus der Analyse des Datenschemas als auch aus dem Vergleich von IFC4x3 mit den erarbeiteten Modellanforderungen. Wegen der unzureichenden Implementierung von IFC4x3 in der gewählten Software konnte das Potential von IFC4x3 in dieser Arbeit jedoch nur theoretisch abgeschätzt werden. Aus den Modellanforderungen wurde ein auf IFC4x3 zugeschnittener Objekt- und Attributkatalog zusammengestellt. Mit dessen Einhaltung könnte die Vollständigkeit eines Modells in IFC4x3 für den Use Case Konformitätsprüfung sichergestellt werden. Bei der Analyse der Richtlinien wurde festgestellt, dass die

untersuchten Richtlinien des Brückenbaus weitgehend konkrete und verständliche Anforderungen an die Entwürfe enthalten, was für eine Formulierung der Richtlinien als computer-interpretierbare Regeln günstig ist. Für die Prüfung einiger Anforderungen ist es allerdings nötig Fachwissen des Bauwesens in die Regeln zu integrieren, beispielsweise indem konkrete, einfach prüfbare Randbedingungen durch Fachleute vorgegeben werden. Bei der abschließenden Implementierung von Beispielregeln wurde erkannt, dass ein korrektes Mapping der Modelle in IFC und eine eindeutige und nachvollziehbare Benennung der Objekte im Modell für eine Identifizierung der Objekte durch die Regeln unabdingbar sind. Außerdem sind für die Überprüfung von komplexeren Regeln häufig geometrische Hilfskörper nötig. Diese müssen entweder schon vom BIM-Autor in dessen Modell integriert oder vom Prüfer des Modells in einem extra Schritt erzeugt werden. Wenn diese Bedingungen erfüllt sind, können die Anforderungen an spezifische Bauteile beziehungsweise an die Beziehung zwischen den Bauteilen mit den Regeln überprüft werden. Es empfiehlt sich gerade bei komplexen Prüfungen, in denen viele Bedingungen verknüpft werden, Teilergebnisse der einzelnen Schritte für den Anwender zu dokumentieren, um den Prüfprozess möglichst nachvollziehbar zu gestalten.

Viele Anforderungen aus den Richtlinien können grundsätzlich an IFC2x3-Modellen geprüft werden. Für eine Konformitätsprüfung einiger Anforderungen wird jedoch IFC4x3 benötigt, da es eine lineare Platzierung erlaubt und erweiterte Entities für den Infrastrukturbau enthält. Eine weitreichende modellbasierte Konformitätsprüfung für Brückenentwürfe ist somit erst möglich, wenn IFC4x3 in naher Zukunft als Datenstandard standardisiert und etabliert ist. Insgesamt zeigt sich, dass eine Anwendung von Konformitätsprüfungen für Brückenentwürfe generell möglich ist. Es sind dafür aber noch weitere Schritte in Forschung und Entwicklung notwendig.

7.2 Ausblick

In dieser Arbeit wurde nur eine sehr begrenzte Auswahl von Regeln genauer betrachtet. Für eine praktische Anwendung der Regeln ist daher noch viel Implementierungsarbeit notwendig. Zudem ist eine tiefgehende praktische Erprobung der Regeln an IFC4x3-Modellen notwendig, sobald das Datenmodell ausreichend in BIM-Authorig-Softwares implementiert ist. Da die Erstellung der Regeln Hand in Hand mit der Erstellung von Modellanforderungen abläuft, ist nach der erweiterten Erprobung wahrscheinlich eine Änderung oder Erweiterung der Modellanforderungen nötig.

Es wurde auch nur ein Auszug aus den Richtlinien für Entwürfe des Brückenbaus berücksichtigt. Aus den Anforderungen der restlichen Richtlinien beispielsweise für Instandsetzungen oder integrale Bauwerke müssen in der Zukunft ebenfalls Modellanforderungen und Regeln abgeleitet werden, wenn die Entwurfsprüfung an BIM-Modellen erfolgen soll. Der erarbeitete Objekt- und Attributkatalog muss dementsprechend erweitert werden.

Der Katalog könnte in der Zukunft verwendet werden, um daraus standardisierte computer-interpretierbare Modellanforderungen zu formulieren, beispielsweise in einem IDS. In diesem Zusammenhang sollte zudem eine Peer-Review insbesondere für die Zuordnung der Anforderungen zu IFC4x3-Entities erfolgen. Bei der Erstellung IFC4x3-basierter Modellanforderungen muss dabei zudem beachtet werden, ob alle geforderten Entities in gängiger Software implementiert sind. Ist das nicht der Fall, können die formulierten Modellanforderungen in der Praxis nicht erfüllt werden.

Literaturverzeichnis

Beitrag

Abualdenien, J., Borrmann, A., König, M. (2021). *Ausarbeitungsgrade von BIM-Modellen*. In: A. Borrmann, M. König, C. Koch & J. Beetz (Hrsg.), *Building Information Modeling: Technologische Grundlagen und industrielle Praxis* (2. Auflage, S. 165–191). Springer Fachmedien Wiesbaden (VDI-Buch).

Appelt, A., Bracher, A. (2020). *12A Straßenwesen*. In: A. Albert (Hrsg.), *Schneider Bautabellen für Ingenieure mit Berechnungshinweisen und Beispielen* (24. Auflage, S. 12.2–12.70). Köln. Reguvis Fachmedien GmbH.

Artus, M., Koch, C., König, M. (2021). *Datenmodellierung*. In: A. Borrmann, M. König, C. Koch & J. Beetz (Hrsg.), *Building Information Modeling: Technologische Grundlagen und industrielle Praxis* (2. Auflage, S. 53–72). Springer Fachmedien Wiesbaden (VDI-Buch).

Beetz, J., Borrmann, A., Weise, M. (2021a). *Prozessbasierte Definition von Modellinhalten*. In: A. Borrmann, M. König, C. Koch & J. Beetz (Hrsg.), *Building Information Modeling: Technologische Grundlagen und industrielle Praxis* (2. Auflage, S. 147–163). Springer Fachmedien Wiesbaden (VDI-Buch).

Beetz, J., Pauwels, P., McGlenn, K., Tormä, S. (2021b). *Linked Data im Bauwesen*. In: A. Borrmann, M. König, C. Koch & J. Beetz (Hrsg.), *Building Information Modeling: Technologische Grundlagen und industrielle Praxis* (2. Auflage, S. 223–266). Springer Fachmedien Wiesbaden (VDI-Buch).

Borrmann, A., Beetz, J., Koch, C., Liebich, T., Muhič, S. (2021a). *Industry Foundation Classes - Ein herstellerunabhängiges Datenmodell für den gesamten Lebenszyklus eines Bauwerks*. In: A. Borrmann, M. König, C. Koch & J. Beetz (Hrsg.), *Building Information Modeling: Technologische Grundlagen und industrielle Praxis* (2. Auflage, S. 95–146). Springer Fachmedien Wiesbaden (VDI-Buch).

Borrmann, A., Berkahn, V. (2021). *Grundlagen der geometrischen Modellierung*. In: A. Borrmann, M. König, C. Koch & J. Beetz (Hrsg.), *Building Information Modeling:*

Technologische Grundlagen und industrielle Praxis (2. Auflage, S. 35–51). Springer Fachmedien Wiesbaden (VDI-Buch).

Borrmann, A., König, M., Koch, C., Beetz, J. (2021b). *Die BIM-Methode im Überblick*. In: A. Borrmann, M. König, C. Koch & J. Beetz (Hrsg.), *Building Information Modeling: Technologische Grundlagen und industrielle Praxis* (2. Auflage, S. 1–31). Springer Fachmedien Wiesbaden (VDI-Buch).

Borrmann, A., Muhic, S., Hyvärinen, J., Chipman, T., Jaud, S., Castaing, C. et al. (2019). *The IFC-Bridge project – Extending the IFC standard to enable high-quality exchange of bridge information models*. In: *Proceedings of the 2019 European Conference on Computing in Construction*, S. 377–386).

Jaud, S., Esser, S., Wikstroem, L., Muhic, S., Mirtschin, J., Borrmann, A. (2021). *A critical analysis of linear placement in IFC models*. In: *Proceedings of the 13th European Conference on Product and Process Modelling (ECPPM 2020 - 2021)*. Proceedings of the 13th European Conference on Product and Process Modelling (ECPPM 2020 - 2021) Online verfügbar unter <https://mediatum.ub.tum.de/node?id=1619740>.

Jaud, Š., Esser, S., Muhič, S., Borrmann, A. (2020). *DEVELOPMENT OF IFC SCHEMA FOR INFRASTRUCTURE*. In: *Proceedings of 6th international conference siBIM: Structured data is the new gold*. Proceedings of 6th international conference siBIM: Structured data is the new gold Online verfügbar unter <https://mediatum.ub.tum.de/node?id=1580384>.

Lukas, K., Ji, Y., Obergrießer, M., Borrmann, A. (2008). *Parametrische Modellierung von Brückenbauwerken*. In: *Tagungsband des 20. Forum Bauinformatik*. Online verfügbar unter https://publications.cms.bgu.tum.de/2008_Lukas_FBI.pdf, zuletzt geprüft am 27.02.2023.

Preidel, C., Borrmann, A., Beetz, J. (2021). *Prüfung der Einhaltung von Normen und Richtlinien mittels BIM*. In: A. Borrmann, M. König, C. Koch & J. Beetz (Hrsg.), *Building Information Modeling: Technologische Grundlagen und industrielle Praxis* (2. Auflage, S. 443–462). Springer Fachmedien Wiesbaden (VDI-Buch). DOI: 10.1007/978-3-658-33361-4_23.

Preidel, C., Bus, N., Borrmann, A., Fies, B. (2018). *Pre-Processing IFC Building Models for Code Compliance Checking based on Visual Programming*. In: *Proceedings of the international conference on Computing in Civil and Building Engineering 2018*.

Online verfügbar unter <https://mediatum.ub.tum.de/node?id=1616874>, zuletzt geprüft am 01.12.2022.

Solihin, W., Dimyadi, J., Lee, Y.-C., Eastman, C., Amor, R. (2017). *The Critical Role of Accessible Data for BIM-Based Automated Rule Checking Systems*. In: F. Bosché, I. Brilakis & R. Sacks (Hrsg.), *Proceedings of the Joint Conference on Computing in Construction (JC3)*. July 4-7, 2017, Heraklion, Crete, Greece. Lean and Computing in Construction Congress - Joint Conference on Computing in Construction. Heraklion, Crete, Greece, 7/4/2017 - 7/7/2017. International Council for Building Research, Studies and Documentation. Edinburgh, UK. Heriot-Watt University (2017 Lean and Computing in Construction Congress (LC3), volume 1), S. 53–60).

Tomczak, A., van Berlo, L., Bolpagni, M., Krijnen, T., Borrmann, A. (2022). *A review of methods to specify information requirements in digital construction projects*. In: *CIB World Building Congress 2022*. Online verfügbar unter https://www.researchgate.net/publication/362159919_A_review_of_methods_to_specify_information_requirements_in_digital_construction_projects.

Windisch, R., Katranuschkov, P., Scherer, R. (2012). *A generic filter framework for consistent generation of BIM-based model views*. In: *Proceedings of the 2012 International Workshop on Intelligent Computing in Engineering*. Online verfügbar unter https://www.researchgate.net/profile/ronny-windisch-2/publication/266700733_a_generic_filter_framework_for_consistent_generation_of_bim-based_model_views/links/5437d8c50cf2d5fa292b6f1d/a-generic-filter-framework-for-consistent-generation-of-bim-based-model-views.pdf.

Zhang, J., El-Gohary, N. (2012). *Extraction of Construction Regulatory Requirements from Textual Documents Using Natural Language Processing Techniques*. In: R.R. Issa, I. Flood & R. Issa (Hrsg.), *Computing in civil engineering*. Proceedings of the 2012 ASCE International Conference on Computing in Civil Engineering, June 17 - 20, 2012, Clearwater Beach, Florida. International Conference on Computing in Civil Engineering. Clearwater Beach, Florida, United States, June 17-20, 2012. American Society of Civil Engineers; International Conference on Computing in Civil Engineering. Reston, Va. American Society of Civil Engineers, S. 453–460).

Buch (Monographie)

buildingSMART Deutschland (2023). *BIM-Klassen der Verkehrswege 2.0. Vorstandardisierungsarbeit der buildingSMART-Fachgruppen Verkehrswege und Landschaftsarchitektur*. (2. überarbeitete und erweiterte Auflage) Berlin. bSD Verlag (bSD Schriftenreihe, Heft 1.01), Zuordnung der BIM-Klassen zu IFC4x3: <https://www.buildingsmart.de/Objektkatalog2.0>.

Nawari, Nawari O. (2018). *Building information modeling. Automated code checking and compliance processes*. Boca Raton. CRC Press.

Nöldgen, Markus (2021). *BIM im Brücken- und Ingenieurbau. Digitale Bauwerksmodelle mit NX, 3D-Konstruktion, Datenintegration, Datenaustausch und FE-Simulation*. (2. Auflage) Wiesbaden. Springer Fachmedien Wiesbaden. Online verfügbar unter <https://link.springer.com/book/10.1007/978-3-658-33181-8>.

Sacks, Rafael; Eastman, Charles M.; Lee, Ghang; Teicholz, Paul M. (2018). *BIM handbook. A guide to building information modeling for owners, managers, designers, engineers and contractors*. (Third edition) Hoboken, New Jersey. Wiley.

Graue Literatur / Bericht / Report

Bayerisches Staatsministerium für Wohnen, Bau und Verkehr (Hg.). (2022). *BIM-Leitfaden. Digitales Planen, Bauen und Betreiben im Bereich Straßen- und Brückenbau*. (Version 2.0). Unter Mitarbeit von K. Klemt-Albert, M. Bergmann, M. Köhncke, K. Neubaur und H. Rahebialbert.ing GmbH]. München. Online verfügbar unter [https://www.bestellen.bayern.de/application/applstarter?APPL=eshop&DIR=eshop&ACTIONxSETVAL\(artdtl.htm,APGxNO-DENR:352416,AARTxNR:03500293,AARTxNODENR:365681,USERxBO-DYURL:artdtl.htm,KATALOG:StMB,AKATxNAME:StMB,ALLE:x\)=X](https://www.bestellen.bayern.de/application/applstarter?APPL=eshop&DIR=eshop&ACTIONxSETVAL(artdtl.htm,APGxNO-DENR:352416,AARTxNR:03500293,AARTxNODENR:365681,USERxBO-DYURL:artdtl.htm,KATALOG:StMB,AKATxNAME:StMB,ALLE:x)=X).

Singer, D., Borrmann, A. (2016). *Machbarkeitsstudie BIM für Bestandsbrücken*. Hg. v. Bundesanstalt für Straßenwesen (BASt). Technische Universität München, Leonhard Obermeyer Center, Lehrstuhl für computergestützte Modellierung und Simulation]. Online verfügbar unter <https://bast.opus.hbz-nrw.de/frontdoor/index/index/docId/1746>.

Hochschulschrift

Abdulkareem, R., Gherghiceanu, M. (2022). *Evaluations of Product Data Template initiatives* [Master Thesis, Jönköping University]. Jönköping. Online verfügbar unter <https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1729058&dswid=-6560>.

Lee, J.K. (2011). *Building environment rule and analysis (BERA) language and its application for evaluating building circulation and spatial program* [Dissertation, Georgia Institute of Technology]. Online verfügbar unter <https://smartech.gatech.edu/handle/1853/39482>.

Preidel, C. (2020). *Automatisierte Konformitätsprüfung digitaler Bauwerksmodelle hinsichtlich geltender Normen und Richtlinien mit Hilfe einer visuellen Programmiersprache*. Technische Universität München]. Online verfügbar unter <https://mediatum.ub.tum.de/1534486>.

Schliski, S. (2021). *BIM-Based Code Compliance Checking of the Musterbauordnung* [Master`s Thesis, Technische Universität München]. Online verfügbar unter <https://mediatum.ub.tum.de/node?id=1621703>, zuletzt geprüft am 13.10.2022.

Solihin, W. (2016). *A simplified BIM data representation using a relational database schema for an efficient rule checking system and its associated rule checking language* [Dissertation, Georgia Institute of Technology]. Online verfügbar unter <https://smartech.gatech.edu/handle/1853/54831>.

Taray, A. (2022). *Systematic Evaluation of the IFC Data Model for Infrastructural Assets and BIM Use Cases* [Master`s Thesis, Technische Universität München]. München. Online verfügbar unter <https://mediatum.ub.tum.de/node?id=1694737>, zuletzt geprüft am 04.01.2022.

Internetdokument

Autodesk (2023a). *Autodesk Civil 3D Hilfe | IFC 4.3 Extension for Autodesk Civil 3D | Autodesk*. Online verfügbar unter <https://help.autodesk.com/view/CIV3D/2022/DEU/?guid=GUID-7FAFA066-E981-4EC3-B074-58A57F7A1D46>, zuletzt aktualisiert am 10.05.2023, zuletzt geprüft am 22.05.2023.

Autodesk (2023b). *Civil 3D Funktionen | Funktionen 2024 | Autodesk*. Online verfügbar unter <https://www.autodesk.de/products/civil-3d/features>, zuletzt aktualisiert am 22.05.2023, zuletzt geprüft am 22.05.2023.

Autodesk (2023c). *Revit-Software | BIM-Software | Autodesk*. Online verfügbar unter <https://www.autodesk.de/products/revit/overview?term=1-YEAR&tab=subscription>, zuletzt aktualisiert am 22.05.2023, zuletzt geprüft am 22.05.2023.

Autodesk (2023d). *About Autodesk Revit Model Review*. Online verfügbar unter <https://help.autodesk.com/view/RVT/2020/ENU/?guid=GUID-22F67ECE-2791-4C4D-B86F-E889077D9072>, zuletzt aktualisiert am 21.04.2023, zuletzt geprüft am 28.04.2023.

BIM Deutschland (2023). *Liste der standardisierten Anwendungsfallbezeichnungen*. Online verfügbar unter <https://www.bimdeutschland.de/bim-deutschland/liste-der-standardisierten-anwendungsfallbezeichnungen>, zuletzt aktualisiert am 31.01.2023, zuletzt geprüft am 31.01.2023.

BIM Forum (2020). *Level of Development (LOD) Specification. Part I & COMMENTARY* [For Building Information Models and Data. Unter Mitarbeit von Jim Bedrick, Will Ikerd und Jan Reinhardt] Online verfügbar unter <https://bimforum.org/resource/level-of-development-specification/>, zuletzt aktualisiert am 13.04.2023, zuletzt geprüft am 13.04.2023.

buildingSMART International (2019). *IfcBuildingElementProxy*. Online verfügbar unter <https://standards.buildingsmart.org/IFC/RELEASE/IFC2x3/TC1/HTML/ifcproductextension/lexical/ifcbuildingelementproxy.htm>, zuletzt aktualisiert am 11.12.2019, zuletzt geprüft am 22.05.2023.

buildingSMART International (2021a). *MVD policy for IFC 4.x*. Online verfügbar unter <https://www.buildingsmart.org/standards/bsi-standards/standards-library/>, zuletzt aktualisiert am April 2021, zuletzt geprüft am 25.10.2022.

buildingSMART International (2021b). *Chapters - buildingSMART International*. Online verfügbar unter <https://www.buildingsmart.org/community/chapters/>, zuletzt aktualisiert am 24.06.2021, zuletzt geprüft am 30.01.2023.

buildingSMART International (2022a). *IFC4.3.1.0 Documentation*. Online verfügbar unter <http://ifc43-docs.standards.buildingsmart.org/IFC/RELEASE/IFC4x3/HTML/content/introduction.htm>, zuletzt aktualisiert am 27.01.2023, zuletzt geprüft am 27.01.2023, Lizenz: <https://creativecommons.org/licenses/by-nc-nd/4.0/>.

buildingSMART International (2022b). *Rooms - buildingSMART International*. Online verfügbar unter <https://www.buildingsmart.org/standards/rooms/>, zuletzt aktualisiert am 05.01.2022, zuletzt geprüft am 30.01.2023.

buildingSMART International (2022c). *What We Do - buildingSMART International*. Online verfügbar unter <https://www.buildingsmart.org/about/what-we-do/>, zuletzt aktualisiert am 12.01.2022, zuletzt geprüft am 30.01.2023.

buildingSMART International (2022d). *Regulatory Room - buildingSMART International*. Online verfügbar unter <https://www.buildingsmart.org/standards/rooms/regulatory/>, zuletzt aktualisiert am 27.05.2022, zuletzt geprüft am 20.12.2022.

buildingSMART International (2022e). *buildingSMART Data Dictionary - buildingSMART International*. Online verfügbar unter <https://www.buildingsmart.org/users/services/buildingsmart-data-dictionary/>, zuletzt aktualisiert am 15.08.2022, zuletzt geprüft am 30.01.2023.

buildingSMART International (2023). *Use Case Management*, zuletzt aktualisiert am 18.04.2023, zuletzt geprüft am 18.04.2023.

buildingSMART Technical (2022a). *IFC Schema Specifications - buildingSMART Technical*. Online verfügbar unter <https://technical.buildingsmart.org/standards/ifc/ifc-schema-specifications/>, zuletzt aktualisiert am 31.05.2022, zuletzt geprüft am 20.12.2022.

buildingSMART Technical (2022b). *Model View Definitions (MVD) - buildingSMART Technical*. Online verfügbar unter <https://technical.buildingsmart.org/standards/ifc/mvd/>, zuletzt aktualisiert am 17.10.2022, zuletzt geprüft am 18.04.2023.

buildingSMART Technical (2023a). *Information Delivery Manual (IDM) - buildingSMART Technical*. Online verfügbar unter <https://technical.buildingsmart.org/standards/information-delivery-manual/>, zuletzt aktualisiert am 14.03.2023, zuletzt geprüft am 14.03.2023.

buildingSMART Technical (2023b). *Information Delivery Specification IDS - buildingSMART Technical*. Online verfügbar unter <https://technical.buildingsmart.org/projects/information-delivery-specification-ids/>, zuletzt aktualisiert am 13.03.2023, zuletzt geprüft am 13.03.2023.

Kelly, R. (2022). *The status of IFC 4.3 and the benefit of further extensions as IFC 4.4 - buildingSMART International*. Hg. v. buildingSMART International. Online verfügbar unter <https://www.buildingsmart.org/the-status-of-ifc-4-3-and-the-benefit-of-further-extensions-as-ifc-4-4/>, zuletzt aktualisiert am 16.09.2022, zuletzt geprüft am 06.02.2023.

Mensch+Maschine (2023). *DESITE BIM-Produktfamilie*. Hg. v. SE, Mensch und Maschine Software. Online verfügbar unter <https://www.mum.de/produkte/desite-bim>, zuletzt aktualisiert am 25.04.2023, zuletzt geprüft am 25.04.2023.

Pinzenöhler, A. (2022). *IFC 4.3 Implementation & Validation Report. Appendix D - Alignment Rework Report* [Motivation for extending and changing IfcAlignment - overview of core Rail Domain requirements and their implementation in IFC 4.3, buildingSMART International (IR-2021-1039-PS-J), zuletzt geprüft am 21.02.2023, Lizenz: <https://creativecommons.org/licenses/by-sa/4.0/>.

Thinkproject (2022). *DESITE BIM - Thinkproject*. Online verfügbar unter <https://thinkproject.com/de/produkte/desite-bim/>, zuletzt aktualisiert am 06.04.2022, zuletzt geprüft am 25.04.2023.

Thinkproject (2023). *DESITE : API Documentation 3.0*. Online verfügbar unter <https://bimdocs.thinkproject.com/desiteAPI/3.0/index.html>, zuletzt aktualisiert am 20.04.2023, zuletzt geprüft am 03.05.2023.

van Berlo, L., Drogemuller, R., Omrani, S. (2023). *What is Information Delivery Specification (IDS) - buildingSMART International*. Hg. v. buildingSMART International. Online verfügbar unter <https://www.buildingsmart.org/what-is-information-delivery-specification-ids/>, zuletzt aktualisiert am 13.02.2023, zuletzt geprüft am 13.03.2023.

W3C (2004). *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*. Unter Mitarbeit von I. Horrocks, P. Patel-Schneider, H. Boley, S. Tabet, B. Grosf und M. Dean. Hg. v. W3C Member Submission. Online verfügbar unter <https://www.w3.org/Submission/SWRL/>, zuletzt aktualisiert am 24.06.2010, zuletzt geprüft am 10.01.2023.

W3C (2013). *SPARQL 1.1 Query Language*. Unter Mitarbeit von S. Harris, A. Seaborne und Prud'hommeaux. Hg. v. W3C Recommendation. Online verfügbar unter <https://www.w3.org/TR/sparql11-query/>, zuletzt aktualisiert am 09.10.2018, zuletzt geprüft am 10.01.2023.

Wix, J. (1999). *An Introduction to the International Alliance for Interoperability and the Industry Foundation Classes* [Industry Foundation Classes - Release 2.0. Unter Mitarbeit von Richard See. Hg. v. International Alliance for Interoperability (IAI). Online verfügbar unter

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahU-KEwiljY7Y6YD9AhV6i_0HHZX8DR4QFnoECAsQAQ&url=https%3A%2F%2Fiaa.web.lbl.gov%2FResources%2FIFC_Releases%2FIFC_Release_2.0%2FBETA_Docs_for_Review%2FIFC_R2_Intro-ToIAI_Beta_d2.PDF&usq=AOvVaw2B_zw95MD2_mUaAldXgXN-, zuletzt aktualisiert am 10.01.1999.

Norm

BAST 2022 *RiZ-Ing. Richtzeichnungen für Ingenieurbauten (RiZ-Ing)*. Online verfügbar unter <https://www.bast.de/DE/Publikationen/Regelwerke/Ingenieurbau/Entwurf/RIZ-ING.html?nn=1818004>, zuletzt geprüft am 14.10.2022.

BMDV 2022a *RAB-Ing. Richtlinien für das Aufstellen von Bauwerksentwürfen für Ingenieurbauten (RAB-Ing)*. Online verfügbar unter <https://www.bast.de/DE/Publikationen/Regelwerke/Ingenieurbau/Entwurf/RAB-ING.html?nn=1818004>, zuletzt geprüft am 14.10.2022.

BMDV 2022b *RE-Ing. Richtlinien für den Entwurf, die konstruktive Ausbildung und Ausstattung von Ingenieurbauten (RE-Ing)*. Online verfügbar unter <https://www.bast.de/DE/Publikationen/Regelwerke/Ingenieurbau/Entwurf/RE-ING.html?nn=1818004>, zuletzt geprüft am 14.10.2022.

BMDV 2022c *ZTV-Ing. Zusätzliche Technische Vertragsbedingungen und Richtlinien für Ingenieurbauten (ZTV-Ing)*. Online verfügbar unter <https://www.bast.de/DE/Publikationen/Regelwerke/Ingenieurbau/Baudurchfuehrung/ZTV-ING.html?nn=1818004>, zuletzt geprüft am 14.10.2022.

DIN 2018 *DIN EN ISO 29481-1. Bauwerksinformationsmodelle –Handbuch der Informationslieferungen – Teil 1: Methodik und Format (ISO 29481-1:2016); Deutsche Fassung EN ISO 29481-1:2017*.

DIN 2019 *DIN EN ISO 19650-1. Organisation und Digitalisierung von Informationen zu Bauwerken und Ingenieurleistungen, einschließlich Bauwerksinformationsmodellierung (BIM) - Informationsmanagement mit BIM*.

DIN 2020 *DIN EN ISO 23386. Bauwerksinformationsmodellierung und andere digitale Prozesse im Bauwesen –Methodik zur Beschreibung, Erstellung und Pflege von Merkmalen in miteinander verbundenen Datenkatalogen (ISO 23386:2020); Deutsche Fassung EN ISO 23386:2020*.

DIN 2021a *DIN EN 17412-1. Bauwerksinformationsmodellierung - Informationsbedarfstiefe - Teil 1: Konzepte und Grundsätze; Deutsche Fassung EN 17412-1:2020.*

DIN 2021b *DIN EN ISO 19148. Geoinformation - Lineares Bezugssystem (ISO 19148:2021); Englische Fassung EN ISO 19148:2021.*

DIN 2022 *DIN EN ISO 12006-3. Bauwesen - Organisation von Daten zu Bauwerken - Teil 3: Struktur für den objektorientierten Informationsaustausch.*

ISO 2018 *ISO 16739-1. Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries - Part 1: Data schema.* Online verfügbar unter <https://www.iso.org/standard/70303.html>.

VDI 2022 *VDI-Richtlinie 2552 Blatt 2. Building Information Modeling - Begriffe.*

Zeitschriftenaufsatz

Bloch, T. (2022). *Connecting research on semantic enrichment of BIM - review of approaches, methods and possible applications.* In: *Journal of Information Technology in Construction* 27, S. 416–440. DOI: 10.36680/j.itcon.2022.020.

Bus, N., Roxin, A., Picinbono, G., Fahad, M. (2018). *Towards French Smart Building Code: Compliance Checking Based on Semantic Rules.* In: *Linked Data for Architecture and Construction (LDAC'2018)* Online verfügbar unter <https://arxiv.org/pdf/1910.00334>.

Dimyadi, J., Clifton, C., Spearpoint, M., Amor, R. (2016). *Computerizing Regulatory Knowledge for Building Engineering Design.* In: *Journal of Computing in Civil Engineering* 30 (5), Artikel C4016001, C4016001. DOI: 10.1061/(ASCE)CP.1943-5487.0000572.

Eastman, C., Lee, J., Jeong, Y., Lee, J. (2009). *Automatic rule-based checking of building designs.* In: *Automation in Construction* 18 (8), S. 1011–1033. DOI: 10.1016/j.autcon.2009.07.002.

Häußler, M., Borrmann, A. (2020). *Model-based quality assurance in railway infrastructure planning.* In: *Automation in Construction* 109 (102971). DOI: 10.1016/j.autcon.2019.102971.

Häußler, M., Esser, S., Borrmann, A. (2021). *Code compliance checking of railway designs by integrating BIM, BPMN and DMN.* In: *Automation in Construction* 121 (103427), S. 103427. DOI: 10.1016/j.autcon.2020.103427.

Hjelseth, E., Nisbet, N. (2011). *Capturing Normative Constraints by Use of the Semantic Mark-Up Rase Methodology*. Online verfügbar unter https://www.academia.edu/32697265/Capturing_Normative_Constraints_by_Use_of_the_Semantic_Mark_Up_Rase_Methodology, zuletzt geprüft am 01.12.2022.

Laakso, M., Kiviniemi, A. (2012). *THE IFC STANDARD - A REVIEW OF HISTORY, DEVELOPMENT, AND STANDARDIZATION*. In: *Journal of Information Technology in Construction (ITcon)* 17 (9), S. 134–161). Online verfügbar unter <https://www.itcon.org/paper/2012/9>, Lizenz: <https://creativecommons.org/licenses/by/3.0/deed.de>.

Nawari, N.O. (2012). *Automating Codes Conformance*. In: *Journal of Architectural Engineering* 18 (4), S. 315–323. DOI: 10.1061/(ASCE)AE.1943-5568.0000049.

Solihin, W., Eastman, C. (2015). *Classification of rules for automated BIM rule checking development*. In: *Automation in Construction* 53, S. 69–82. DOI: 10.1016/j.autcon.2015.03.003.

Sydora, C., Stroulia, E. (2019). *Towards Rule-Based Model Checking of Building Information Models*. In: *ISARC Proceedings*, S. 1327–1333). Online verfügbar unter http://www.iaarc.org/publications/2019_proceedings_of_the_36th_isarc/towards_rule_based_model_checking_of_building_information_models.html.

Xu, X., Cai, H. (2021). *Ontology and rule-based natural language processing approach for interpreting textual regulations on underground utility infrastructure*. In: *Advanced Engineering Informatics* 48, S. 101288. DOI: 10.1016/j.aei.2021.101288.

Zeitungsartikel

BIM Deutschland (2022). *Digital bauen auf Grundlage der BIM-Anwendungsfälle*. In: *BundesBauBlatt* 15.05.2022 Online verfügbar unter https://www.bundesbaublatt.de/artikel/bbb_Digital_bauen_auf_Grundlage_der_BIM-Anwendungsfaelle-3767407.html, zuletzt geprüft am 31.01.2023.

Anhang A

In der nachfolgenden Tabelle sind die erarbeiteten Modellanforderungen zusammengestellt.

Zusätzlich zu den in der Tabelle genannten Attributen werden bauteilübergreifend folgende Attribute benötigt:

- die Zuordnung einer GUID (Attribut *GlobalID*)
- die Zuordnung eines Materials (über *IfcRelAssociatesMaterial*)
- die Zuordnung einer Geometrierepräsentation (Attribut *Representation*)
- die Zuordnung, ob ein Bauteil ein Außenbauteil ist (Property *IsExternal* für alle Bauteile, die als *IfcWall*, *IfcSlab*, *IfcBeam* oder *IfcRamp* modelliert sind, aus dem jeweiligen Property Set *Pset_...Common*)

Die schwarzen Werte in der Spalte *IFC4x3 – Entity* entsprechen der Zuordnung gemäß buildingSMART Deutschland 2023², die **blauen** Werte entsprechen der selbst erarbeiteten Zuordnung, wo die Zuordnung von bSD nicht vorhanden oder mehrdeutig war.

Objekt	IFC4x3 – Entity (gem. bSD/eigene Zuordnung)	Attribut/Property (ggf. mit Vorgabewert)	IFC4x3 – Pset (nur eigene Zuordnung)
Abdichtung	IfcCoveringType-Enum.MEMBRANE	zugehoerigeRiZ	eigenes
Baugrube	IfcEarthworksCut-TypeEnum. BASE_EXCAVATION		
Baugrubensicherung	IfcWallType-Enum.RETAINING-WALL	Status = <i>Temporary/New</i>	Pset_WallCommon
Befestigte Flächen außerhalb Böschung	IfcPavement		
Belag	IfcPavement		
Berme	IfcEarthworksFill		
Betonschutzmaßnahmen	IfcCoveringType-Enum.MEMBRANE		
Bohrproben Boden	IfcBorehole		
Böschung	IfcEarthworksCutType-Enum.CUT		
Böschungsbefestigung	IfcPavement		

² <https://www.buildingsmart.de/Objektkatalog2.0> (letzter Zugriff am 18.04.2023)

Böschungstreppe	IfcStair	zugehoerigeRiZ Auftrittsbreite Auftrittshöhe Breite Treppenlauf	eigenes Pset_StairCommon Pset_StairCommon Pset_StairCommon
Brückenabläufe	IfcWasteTerminalType- Enum.GULLYTRAP		
Brückenausrüstung	variabel		
Entwässerungsleitung	IfcDistributionFlow- Element	zugehoerigeRiZ	eigenes
Entwässerungsmulde Straße	IfcEarthworksCutType- Enum.TRENCH		
Erdung	IfcDistributionSystem- Enum.EARTHING		
Fahrbahnplatte	IfcSlabTypeEnum. USERDEFINED		
Flügel	IfcElementAssembly- TypeEnum.ABUT- MENT	zugehoerigeRiZ	eigenes
Fluss	IfcGeotechnicalStratum- TypeEnum.WATER	Flusswasserspiegel Hochwasserspiegel (als Differenz zum regulären Spiegel mit SeicheR- ange)	Pset_WaterStratum- Common
Freizuhaltende Flä- chen	IfcSpatialZoneType- Enum.RESERVATION		
Fugen	IfcRelConnectsWith- RealizingElements	zugehoerigeRiZ ConnectionType (z.B. <i>Bewegungsfuge, Ar- beits- fuge, Sollrissfuge, ...</i>)	eigenes optionales Attribut En- tity
Fugenband	IfcCoveringType- Enum.MEMBRANE		
Gelände	IfcGeographicElement- TypeEnum.TERRAIN		
Geländer	IfcRailingType- Enum.GUARDRAIL	zugehoerigeRiZ Höhe	eigenes Pset_RailingCommon
Gleise	IfcRail		
Globales Koordinaten- system	IfcGeographicCRS	ESPG-Code	Entity-Attribut Name
Gradiente	IfcAlignmentVertical		
Gründungskörper	IfcBridgePartType- Enum.FOUNDATION	Dicke	Qto_FootingBase- Quantities
Grundwasserstand	IfcGeotechnicalStratum- TypeEnum.WATER		
Handlauf Böschungstreppe	IfcRailingType- Enum.HANDRAIL		
Hinterfüllung	IfcEarthworksFillType- Enum.BACKFILL	HydraulicConductivity	Pset_SolidStratum- Capacity

Hohlkasten	Betonteile als IfcWall/IfcSlab, Hohlraum als IfcSpatialZoneTypeEnum.SECURITY		
Hohlräume	IfcVoid		
Homogenbereiche Boden	IfcGeotechnicalStratumTypeEnum.SOLID	Bodenkennwerte Setzungen	Pset_SolidStratumCapacity
Kappen	IfcBeamTypeEnum.EDGEBEAM	zugehoerigeRiZ AssemblyPlace (Site)	eigenes Pset_ConcreteElementGeneral
Kappenanker	IfcMechanicalFastener		
Lager	IfcBearing	zugehoerigeRiZ Bewegungsrichtung x/y/z Lagertyp (z.B. <i>Elastomeric</i>)	eigenes Pset_BearingCommon Entity-Attribut PredefinedType
Längsträger	IfcBeam	AssemblyPlace (Site)	Pset_ConcreteElementGeneral
Laufsteg	Steg als IfcBridgeTypeEnum.USERDEFINED, Handlauf als IfcRailingTypeEnum.HANDRAIL		
Leiter	IfcStairTypeEnum.LADDER		
Leitungen	IfcDistributionFlowElement	IfcRelAssignsToProcess (falls z.B. Verlegung nötig)	Entity-Attribut
Lichtraumprofil	IfcSpatialZone		
Lärmschutzwand	IfcWallTypeEnum.USERDEFINED		
Maßnahmen gegen Befahren/Zugang Unbefugter	IfcDiscreteAccessoryTypeEnum.USERDEFINED		
Mast	IfcElementAssemblyTypeEnum.MAST		
Öffnungen	IfcOpeningElement	zugehoerigeRiZ verschließbar	eigenes eigenes
Pfahl	IfcPile	Pfahllänge	Qto_PileBaseQuantities
Pfeiler	IfcBridgePartTypeEnum.USERDEFINED oder IfcColumn	zugehoerigeRiZ	eigenes

Pfeiler Standfläche	Standfläche oben auf Pfeiler als IfcSpatialZoneTypeEnum.SECURITY		
Pfeiler-/Widerlagerachse	IfcGrid		
Pressen	Presse als IfcTransportElementTypeEnum.LIFTINGGEAR	zugehoerigeRiZ zulässiges Anhebemaß	eigenes eigenes
Pressen Standfläche	hor. Stellfläche als IfcSpatialZoneTypeEnum.CONSTRUCTION		
Querträger	IfcBeam		
Sauberkeitsschicht	IfcSlabTypeEnum.USERDEFINED		
Schrammbord	IfcKerbTypeEnum.USERDEFINED	zugehoerigeRiZ Breite Höhe	eigenes Pset_KerbStone Pset_KerStone
Schutzeinrichtung	IfcRailingTypeEnum.GUARDRAIL	Aufhaltstufe Wirkungsbereich	eigenes eigenes
Schwingungsdämpfer	IfcVibrationDamper		
Seil im Geländer	IfcCableSegment		
Straße	IfcRoad	Straßenklasse Ziel Verkehrsweg zulässige Geschwindigkeit	Pset_RoadDesignCriteriaCommon Entity-Attribut: Description Pset_RoadDesignCriteriaCommon
Tropftüllen	IfcWasteTerminalTypeEnum.USERDEFINED	zugehoerigeRiZ	eigenes
Türen	IfcDoor	zugehoerigeRiZ	eigenes
Überbau	IfcFacilityPartCommonTypeEnum.SUPERSTRUCTURE	zugehoerigeRiZ	eigenes
Übergangskonstruktion	IfcElementAssemblyTypeEnum.DILATATIONPANEL	zugehoerigeRiZ mögliche Bewegung lärmmindernd zugelassen durch bast	eigenes Pset_ElementAssemblyTypeDilatationPanel eigenes eigenes
Verbau	variabel, z.B. IfcPile, IfcWallTypeEnum.RETAININGWALL, ...	variabel	variabel

Verkehrsräume	IfcSpatialZone	Breite Zweck (über Attribut Name)	Qto_SpatialZone-BaseQuantities Entity-Attribut
Vorspannung	Spannglieder als IfcTendon	PredefinedType (z.B. <i>WIRE</i> oder <i>BAR</i> , nicht <i>COATED</i>)	Entity-Attribut
Vorspannung	Spannköpfe Koppelstellen als IfcTendonAnchor		
Wartungsgang	Betonteile als IfcWall/IfcSlab, Hohlraum als IfcSpatialZoneType-Enum.SECURITY	Breite Hohlraum Höhe Hohlraum	Qto_SpatialZone-BaseQuantities
Widerlager	IfcElementAssembly-TypeEnum.ABUTMENT		
Widerlager Jahreszahl	IfcDate		
Zaun	IfcRailingType-Enum.FENCE		
Zufahrtsweg (befestigt)	IfcPavement	Breite	Pset_Pavement-Common

Anhang B

Der digitale Anhang beinhaltet:

- Zu Kapitel 4 und 5 (Ordner *Regelkategorisierung*)
 - o Die im Rahmen von Kapitel 4 bearbeiteten Richtlinien (4 PDF-Dokumente)
 - o Die Excel-Tabelle für die Richtlinienanalyse aus Kapitel 4 und 5 (Regelkategorisierung.xlsx)

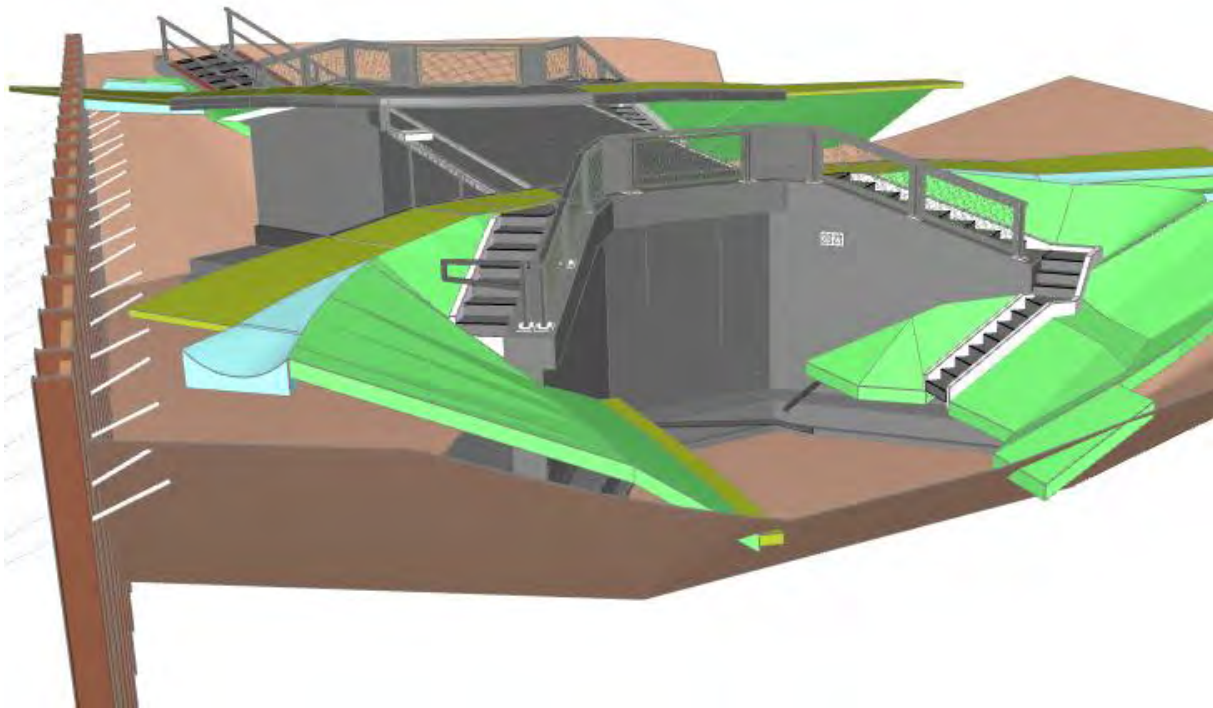
- Zu Kapitel 6.1 (Ordner *IFC4X3*)
 - o Civil 3D-Zeichnung *Hohlkastenbruecke.dwg* (enthält Gelände, Straßen und Brückenüberbau)
 - o Zwei abgeleitete IFC4x3-Dateien
 - o Konfigurationsdatei für IFC4x3-Export (*IfcInfraConfiguration.json*)
 - o Mappingdatei für IFC4x3-Export (*IfcInfraExportMapping.json*)

- Zu Kapitel 6.2 bis 6.3 (Ordner *Implementierung*)
 - o Desite-Projekt *Hohlkastenbrücke.pfs*
 - o Vier zugehörige Skripte in Java Script (Unterordner *Hohlkastenbrücke.scripts*)
 - o Zugehörige IFC-Dateien in IFC2x3 (4 Stück)
 - o Civil 3D-Zeichnung *Strassen.dwg* (enthält Gelände, Straßen und Brückenbelag)
 - o Revit-Modell *Hohlkastenbrücke1.rvt* (enthält Brückenüberbau inkl. Ausstattung und Pfeiler sowie die Hilfskörper)
 - o Revit Modell *Gerade_Hohlkastenbodenplatte_XY.rvt* (enthält die in XY-Ebene projizierte und begradigte Bodenplatte des Hohlkastens inklusive Öffnungen)
 - o Revit Property Set *Pset_Projekt.txt*

Anhang C

In diesem Anhang sind die Screenshots der Anwendung der implementierten Class 2-Regel auf ein reales Projekt abgebildet. Das Modell in IFC2x3 beinhaltet eine Radwegunterführung mit Gelände und Verbau. Das Modell kann aus Gründen des Datenschutzes dem digitalen Anhang nicht beigelegt werden. Die Funktionsweise des Desite-Skripts kann am Desite-Projekt *Hohlkastenbrücke* im digitalen Anhang nachvollzogen werden (vgl. Anhang B).

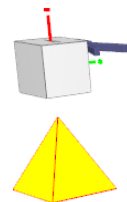
Screenshot 1: Übersicht über das Modell in DESITE md pro (Quelle: Eigene Darstellung)



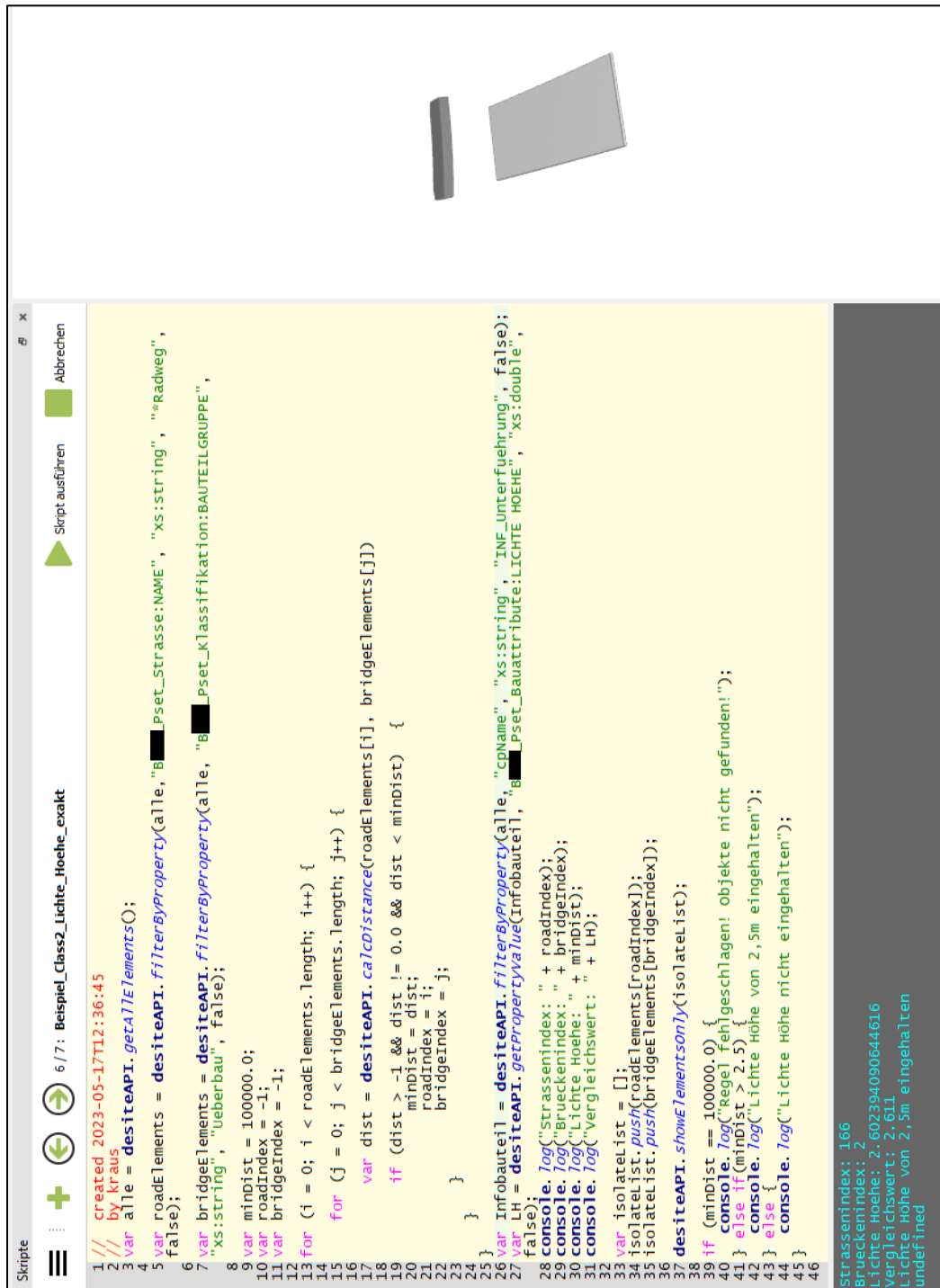
Screenshot 2: Informationsbauteil im Modell (rechts, gelb) mit darin hinterlegten Eigenschaften (links) (Quelle: Eigene Darstellung)

Domäne: **geometry**

	Eigenschaftname	Wert	Datentyp	
	B...Pset_Bauattribute			
1	: BAU_KM		xs:double	⊖
2	: BAUVERFAHREN	Ortbeton mit Schalung	xs:string	⊖
3	: BAUVOLUMEN	241,5400 [m ³]	xs:double	⊖
4	: BAUWEISE	Ortbeton	xs:string	⊖
5	: KREUZUNGSWINKEL	100 gon	xs:string	⊖
6	: LICHTE HOEHE	2,6110 [m]	xs:double	⊖
7	: LICHTE WEITE	3,5000 [m]	xs:double	⊖
8	: VOLUMEN	1,1250 [m ³]	xs:double	⊖



Screenshot 3: Ergebnis des Desite-Skripts der Class 2- Regel (Quelle: Eigene Darstellung)



```

Skripte
6/7: Beispiel_Class_Lichte_Hoeh_e_akt
Abbrechen
Skript ausführen
Abbrechen

1 // created 2023-05-17T12:36:45
2 // by kraus
3 var alle = desiteAPI.getAllElements();
4
5 var roadElements = desiteAPI.filterByProperty(alle, "B_pset_strasse:NAME", "xs:string", "%Radweg",
6 false);
7 var bridgeElements = desiteAPI.filterByProperty(alle, "B_pset_Klassifikation:BAUTEILGRUPPE",
8 "xs:string", "ueberbau", false);
9 var mindist = 100000.0;
10 var roadIndex = -1;
11 var bridgeIndex = -1;
12 for (i = 0; i < roadElements.length; i++) {
13   for (j = 0; j < bridgeElements.length; j++) {
14     var dist = desiteAPI.calculateDistance(roadElements[i], bridgeElements[j])
15     if (dist > -1 && dist != 0.0 && dist < mindist) {
16       mindist = dist;
17       roadIndex = i;
18       bridgeIndex = j;
19     }
20   }
21 }
22
23
24
25
26 var infoBauteil = desiteAPI.filterByProperty(alle, "CDName", "xs:string", "INE_Unterfuehrung", false);
27 var LH = desiteAPI.getPropertyValue(infoBauteil, "B_pset_bauattribut:LICHTE_HOEHE", "xs:double",
28 false);
29 console.log("Strassenindex: " + roadIndex);
30 console.log("Brueckenindex: " + bridgeIndex);
31 console.log("Lichte Hoehe: " + mindist);
32 console.log("Vergleichswert: " + LH);
33 var isolateList = [];
34 isolateList.push(roadElements[roadIndex]);
35 isolateList.push(bridgeElements[bridgeIndex]);
36
37 desiteAPI.showElementsOnly(isolateList);
38
39 if (mindist == 100000.0) {
40   console.log("Regel fehlgeschlagen! Objekte nicht gefunden!");
41 } else if (mindist > 2.5) {
42   console.log("Lichte Höhe von 2,5m eingehalten");
43 } else {
44   console.log("Lichte Höhe nicht eingehalten");
45 }
46
Strassenindex: 166
Brueckenindex: 2
Lichte Hoehe: 2.602394090644616
Vergleichswert: 2.611
Lichte Höhe von 2,5m eingehalten
undefined

```

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Master-Thesis selbstständig angefertigt habe. Es wurden nur die in der Arbeit ausdrücklich benannten Quellen und Hilfsmittel benutzt. Wörtlich oder sinngemäß übernommenes Gedankengut habe ich als solches kenntlich gemacht.

Ich versichere außerdem, dass die vorliegende Arbeit noch nicht einem anderen Prüfungsverfahren zugrunde gelegen hat.

München, 30. Mai 2023

Sandra Kraus