



DIPLOMARBEIT

Methode zur anforderungsbasierten Modellprüfung

A methodology for requirement-based model checking

eingereicht von

cand. ing. Dominik Nils Sonnek

geb. am 13.05.1993 in Kirchheim unter Teck

Matrikel-Nummer: 4603612

Prüfer:

- Prof. Dr.-Ing. habil. Karsten Menzel
- Dr.-Ing. Matthias Weise

Betreuer/in:

- Dipl.-Ing. Frank Opitz
- Dipl.-Ing. Nicolas Mitsch

Dresden, den 01.10.2021



Aufgabenstellung für die Diplomarbeit

Name: cand. ing. Dominik Nils Sonnek
Vertiefung: Konstruktiver Ingenieurbau

Matrikel Nr.: 4603612

Thema: Methode zur anforderungsbasierten Modellprüfung

A methodology for requirements-based model checking

Kontext:

Ab 2020 müssen Infrastrukturprojekte unter Verwendung der BIM-Methodik geplant und durchgeführt werden. Da das digitale Bauwerksmodell im Zentrum der BIM-Methodik steht, gelten auch besondere Anforderungen an den Inhalt dieser Modelle.

Während die automatisierte Prüfung auf Objektkollisionen derzeit weit verbreitet ist, wird die semantische Modellprüfung aktuell vorwiegend händisch bzw. stichpunktartig durch eine Sichtprüfung durchgeführt. Diese manuelle Prüfung ist nicht nur sehr zeitaufwendig, sondern auch anfällig für Fehler und lässt so nur eine unzureichende BIM-Qualitätsprüfung zu.

Eine umfängliche und automatisierte Modellprüfung hilft zum einen, die Qualitätssicherung der Planung bzgl. der BIM-Vorgaben sicherzustellen und zum anderen die Digitalisierung von Planungsprozessen in Bauprojekten zu unterstützen und den Einsatz der BIM-Methode weiter voranzutreiben.

Zielsetzung der Diplomarbeit:

Im Rahmen dieser Arbeit soll recherchiert werden, wie die Modellprüfung derzeit durchgeführt wird und wie eine umfängliche Prüfung der BIM-Vorgaben erfolgen sollte.

Auf Basis der Recherche sollen die inhaltlichen und semantischen Anforderungen an digitale Bauwerksmodelle herausgearbeitet werden, die abhängig von der Leistungsphase von Auftraggebern gestellt werden.

Auf Basis der herausgearbeiteten Anforderungen soll eine Methode entwickelt werden, wie Modelle sowohl auf ihre inhaltliche Vollständigkeit als auch auf ihre semantische Richtigkeit überprüft werden können. Die Validierung des Lösungsansatzes soll an einem oder mehreren BIM-Modellen - vorzugsweise von Infrastrukturprojekten der Deutschen Bahn - erfolgen.

Arbeitsumfang der Diplomarbeit:

Im Rahmen der Ausarbeitung sollen die folgenden Punkte bearbeitet werden:

1. Erarbeitung der Grundlagen zur Modellprüfung:
 - a. Recherchen zum aktuellen Stand der Modellprüfung in der Praxis.
 - b. Aufstellen inhaltlicher und semantischer Qualitätsanforderungen aus Sicht eines Auftraggebers an digitale Bauwerksmodelle.
 - c. Ableitung einer Methodik zur umfänglichen semantischen Modellprüfung.
 - d. Erläuterung technischer Randbedingungen zur Modellprüfung.
2. Entwicklung einer automatisierten Vorgehensweise zur inhaltlichen Vollständigkeitsprüfung von digitalen Bauwerksmodellen.
3. Mögliche Erweiterung der Methode um eine semantische Normenprüfung.
4. Validierung anhand eines oder mehrerer BIM-Modelle der Deutschen Bahn in unterschiedlichen Softwarelösungen.

Verantwortliche und Termine:

Verantwortlicher Hochschullehrer
und Erstprüfer

Prof. Dr.-Ing. habil. Menzel

Zweitprüfer

Dr.-Ing. Matthias Weise (AEC3)

Wiss. Betreuer TU Dresden

Dipl.-Ing. Nicolas Mitsch

Mentoren DB-Netz

Dipl.-Ing. Frank Opitz (DB-Netz)

ausgehändigt am

01.06.2021

einzureichen am

01.10.2021



**Karsten
Menzel**

Digital
unterschieben von
Karsten Menzel
Datum: 2021.05.26
15:37:34 +02'00'

Prof. Karsten Menzel

SELBSTSTÄNDIGKEITSERKLÄRUNG

Ich versichere, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Ich reiche sie erstmals als Prüfungsleistung ein. Mir ist bekannt, dass ein Betrugsversuch mit der Note „nicht ausreichend“ (5,0) geahndet wird und im Wiederholungsfall zum Ausschluss von der Erbringung weiterer Prüfungsleistungen führen kann.

Name: Sonnek
Vorname: Dominik Nils
Matrikelnummer: 4603612

Dresden, den 01.10.2021

DANKSAGUNG

Zunächst möchte ich mich an dieser Stelle bei den Mitarbeitern des Instituts für Bauinformatik bedanken, die mir während der Ausarbeitung dieser Projektarbeit unterstützend zur Seite standen. Ein besonderer Dank gilt dabei:

Prof. Dr.-Ing. habil. Karsten Menzel für die Schaffung der Rahmenbedingungen meiner Diplomarbeit. Er stand mir stets mit wertvollen Anregungen und Ratschlägen zur Seite.

Dipl.-Ing. Nicolas Mitsch danke ich für die Betreuung dieser Diplomarbeit. Selbst während eines neuen Beschäftigungsverhältnisses brachte er die Zeit auf, mir bei Fragen und Problemen zu helfen.

Ferner gilt mein Dank Dr.-Ing. Matthias Weise von der AEC3 für die fachlichen Rücksprachen und die Unterstützung mit den Softwarelösungen.

Ein weiterer Dank geht an die Mitarbeiter der DB Netz AG, die mich als Teil ihres Teams aufnahmen und mir stets zur Seite standen. Bedanken möchte ich mich bedanken bei:

Dipl. Ing. Frank Opitz, für die Betreuung dieser Diplomarbeit und dafür, dass er mir stets Hilfestellungen und Unterstützung bot.

M. Sc. Julian Trujillo-Lopez für die vielen Hilfestellungen und Erläuterungen insbesondere im Rahmen der Überarbeitung des SOM.

Ich möchte mich auch bei meiner Familie und meinen Freunden bedanken, die mich während meines Studiums und insbesondere während der Bearbeitungsphase dieser Arbeit unterstützt haben.

Ein besonderer Dank gilt zum einen Lotta Gohlke für das Korrekturlesen dieser Arbeit und die moralische Unterstützung.

Ein besonderes Dankeschön gebührt weiterhin Michael Maier, der mir durch Korrekturlesen und konstruktive Ratschläge zur Seite stand.

Zum andern gebührt ein besonderer Dank meinem Vater Dr. Mathias Sonnek für das Korrekturlesen, die vielen Anregungen und die emotionale Unterstützung.

I INHALTSVERZEICHNIS

I	Inhaltsverzeichnis.....	VI
II	Abbildungsverzeichnis.....	VIII
III	Tabellenverzeichnis.....	XII
IV	Abkürzungsverzeichnis.....	XIII
V	Abstract.....	XIV
1	Einleitung.....	15
1.1.	Zielsetzung.....	15
1.2.	Vorgehen und Methodik der Arbeit.....	15
1.3.	State of the Art	16
1.4.	Normen, Richtlinien und Regularien	17
1.5.	Standardisierte Formate.....	21
1.5.1.	Der IFC-Standard.....	21
1.5.2.	Der mvdXML-Standard	21
1.6.	Genutzte Softwarelösungen.....	23
1.6.1.	BIMQ	23
1.6.2.	Solibri	23
1.6.3.	Desite md pro.....	24
1.6.4.	BIMCollab Zoom.....	24
1.6.5.	IFC Documentation Generator.....	25
1.6.6.	FZKViewer	26
1.6.7.	Xbim Xplorer.....	26
2	Grundlagen der Modellprüfung.....	28
2.1.	Grundbegriffe	28
2.1.1.	Properties	28
2.1.2.	Property Sets.....	29
2.1.3.	Fertigstellungsgrade von BIM-Leistungen	30
2.1.4.	Modellprüfung im Allgemeinen	34
2.1.5.	Modellprüfung im Rahmen dieser Arbeit	37
3	Modellanforderungen.....	40
3.1.	AIA und BAP.....	40
3.1.1.	Auftraggeberinformationsanforderungen	41

3.1.2.	BIM-Abwicklungsplan	42
3.2.	Prüfregeln	43
3.3.	Das Semantische Objektmodell der DB Netz AG	49
3.3.1.	Erläuterung und Zielsetzung.....	49
3.3.2.	Auswahl der Gewerke.....	50
3.3.3.	Herausforderungen	51
3.3.4.	Neustrukturierung des SOM.....	52
3.3.5.	Die BIMQ-Implementierung.....	58
4	Model Content Checking	62
4.1.	Zu prüfende Modelle	62
4.1.1.	Modell 1: Brückenrampe.....	63
4.1.2.	Modell 2: Stahlhohlträgerbrücke	63
4.1.3.	Modell 3: Eisenbahnüberführung Ingweiler	64
4.2.	Die Prüfungsvorbereitung.....	66
4.3.	Resultate der Modellprüfung	70
4.3.1.	Modell 1	70
4.3.2.	Modell 2	72
4.3.3.	Modell 3	74
4.3.4.	Bewertung der Resultate.....	79
5	Validation Checking.....	81
5.1.	Auswahl der Validierungsaspekte.....	83
5.2.	Die ausgewählten Relationen.....	84
5.3.	Relationsprüfung.....	87
5.3.1.	IfcRelDefinesByProperties.....	87
5.3.2.	IfcRelAssociatesMaterial.....	94
5.3.3.	IfcRelAggregates.....	96
5.3.4.	Vollständige Prüfung eines Beispielmodells.....	101
6	Fazit und Ausblick.....	108
VI	Literaturverzeichnis	i
VII	Digitales Anlageverzeichnis	viii
VIII	Anlagenverzeichnis	ix

II ABBILDUNGSVERZEICHNIS

Abbildung 1: Modellprüfung nach [56]	22
Abbildung 2: Ein Beispiel einer IFC Documentation Generator Datei	26
Abbildung 3: Beispielüberlegungen eines LOG [16]	31
Abbildung 4: Beispiel eines LOD nach [42]	33
Abbildung 5: Die Unterteilung der Arten der Modellprüfung nach [73]	35
Abbildung 6: Schematische Darstellung des Model Content Checking [73]	36
Abbildung 7: Leitende Entscheidungsprüfung nach [74]	37
Abbildung 8: Baum des Porphyrios [86]	44
Abbildung 9: Wahrheitstabelle der Aussagenlogik [87]	44
Abbildung 10: Pfad zur Prüfregelementwicklung [1]	45
Abbildung 11: Beispiel einer RASE-Analyse einer Anforderung nach [90]	46
Abbildung 12: RASE angewandt auf die verschiedenen Modellprüfungsansätze nach [74] 46	
Abbildung 13: Auszug eines TIO-bearbeiteten Regelsatzes [85]	47
Abbildung 14: Zusammenspiel der Vorgehen für die RASE, vergleiche [1]	47
Abbildung 15: Auszug der Zuordnung eines IfcConstraint entsprechend buildingSMART [50, 53, 54]	48
Abbildung 16: IfcConstraint Grade [50, 53, 54]	48
Abbildung 17: PropertySet-Zuweisung in BIMQ	53
Abbildung 18: Bohrpfahlelement in BIMQ	53
Abbildung 19: Auszug der Schnellübersicht des SOM, Kürzel: AE - Allgemeine_Eigenschaften, B - Bauteil und E - Einrichtung	53
Abbildung 20: UML-Diagramm des Brueckenteilbereichs Ueberbau, eine Raute beschreibt hierbei eine Aggregation, ein Dreieck eine Vererbung	55
Abbildung 21: UML-Diagramm des Brueckenteilbereich Unterbau	56

Abbildung 22: Auszug der Attribute von IfcRail [50]	58
Abbildung 23: Objektkarte eines Brueckentragwerk in BR Gewerk	60
Abbildung 24: Auszug des BIMQ-Anforderungskataloges an die Testmodelle	61
Abbildung 25: Brückenmodell 1	63
Abbildung 26: Brückenmodell 2	63
Abbildung 27: Brückenmodell 3	64
Abbildung 28: Auszug der IFC-Übersetzungs-txt.-Datei.....	66
Abbildung 29: Einstellungen einer Prüfredel in Solibri	67
Abbildung 30: Darstellung von Regelsätzen in Desite	68
Abbildung 31: Darstellung der Regeln/Smart-Views in BIMcollab ZOOM.....	69
Abbildung 32: Auswertung des Brückenmodells 1 in Solibri	70
Abbildung 33: Auswertung des Brückenmodells 1 in Desite	71
Abbildung 34: Auswertung des Brückenmodells 1 in Bimcollab Zoom (l.) und Attributprüfung(r.)	71
Abbildung 35: Auswertung des 2. Brückenmodells in Solibri	72
Abbildung 36: Auszug der Auswertung des 2. Brückenmodells in Desite	73
Abbildung 37: Auswertung der Widerlager in BIMcollab ZOOM Komponentenprüfung (l.) Objekteigenschaftsprüfung (r.).....	73
Abbildung 38: Smart-View-Einstellungen.....	74
Abbildung 39: Auswertung der Modellprüfung des 3. Brueckenmodells in Solibri	75
Abbildung 40: Auszug der Auswertung des 3. Brueckenmodells in Desite	75
Abbildung 41: Beispiel für die Fehlermeldungen der Gelaenderkomponenten	76
Abbildung 42: Attributtypen von achse und einbauWinkel.....	76
Abbildung 43: Beispiele für die Fehlermeldungen für Bohrpfaehle und Fluegelwaende	76
Abbildung 44: Integerprüfung im exportierten Java-Script-Code	77
Abbildung 45: Prüfergebnis des 3. Brückenmodells nach einer Codeanpassung	77
Abbildung 46: Die fehlerfreie Prüfung des 3. Brückenmodells.....	78

Abbildung 47: Beispiele der Smart-View-Ergebnisse für die Gelaender, Bohrpfaehle, Fluegelwaende und Seitenfuehrungsbloecke	78
Abbildung 48: IfcRelDefinesByProperties Struktur [50].....	84
Abbildung 49: IfcRelAssociatesMaterial für ein Material (oben) und Materialschichten (unten).....	85
Abbildung 50: IfcRelAggregates von unten (oberes Bild) und oben (unteres Bild) in der Hierarchie [50].....	86
Abbildung 51: IFC Documentation Generator: Fundamental concepts and assumptions (links) und MVDs (rechts).....	88
Abbildung 52: Schema der Propertyzuweisung	88
Abbildung 53: MVD Komponenten.....	89
Abbildung 54: Regelwertzuweisung im IFC Documentation Generator.....	89
Abbildung 55: mvdXML Allgemeine_Eigenschaften Templates	90
Abbildung 56: mvdXML Allgemeine_Eigenschaften View	90
Abbildung 57: Auszug der Prüfung des Brückenmodells 1	91
Abbildung 58: Schema der Bauteilklassifizierung	91
Abbildung 59: Template für die Zuweisung einer bauteilKlassifikation.....	92
Abbildung 60: View für die bauteilKlassifikation	92
Abbildung 61: Auswertung der bauteilKlassifikations-Probe.....	93
Abbildung 62: Fehlermeldungen bei der mvdXML-Prüfung im FZKViewer.....	93
Abbildung 63: Auswertung der Probe auf Materialvorkommen im Xbim Xplorer (l.) und im FZKViewer (r.).....	94
Abbildung 64: IfcMaterial nach [54] (l.) und ein Material im Brückenmodell 2(r.).....	95
Abbildung 65: „Material der Kategorie Stahlbeton“ im Xbim Xplorer und im FZKViewer.....	95
Abbildung 66: Attribuierung der Widerlagerkomponenten in Revit.....	96
Abbildung 67: Metaobjekt Widerlager in IFC-STEP.....	97
Abbildung 68: Template einfache Aggregation (Bestandteile ohne AE).....	97
Abbildung 69: Die Spezifizierung der zu prüfenden Modellkomponenten minimal (o.) und präzise (u.).....	98

Abbildung 70: Aggregation IsDecomposedBy eines Widerlagers im View.....	99
Abbildung 71: Ergebnis der einfachen Aggregationsprüfung.....	99
Abbildung 72: Anforderungen des Widerlagers.....	100
Abbildung 73: Widerlagerprüfung mit der Klassifikation im PropertySet.....	100
Abbildung 74: Prozesskette eines BMC.....	101
Abbildung 75: Concept der Grenzwertprüfung der Brückenattribute	106
Abbildung 76: Ergebnisse des Model Content Checking.....	107
Abbildung 77: Ergebnisse der Model Validation	107

III TABELLENVERZEICHNIS

Tabelle 1: Übersicht der Normen nach CEN TC 442	17
Tabelle 2: British Standard BSI-PAS 1192.....	18
Tabelle 3: Weitere BIM-spezifische Normen.....	19
Tabelle 4: Richtlinien des VDI 2552.....	19
Tabelle 5: Handreichungen der BIM4INFRA2020	20
Tabelle 6: Unterteilung des BMC in dieser Arbeit	37
Tabelle 7: Bewertung der Model Checker.....	80
Tabelle 8: Auswahl von Aspekten der Model Validation.....	81
Tabelle 9: Die geforderte Aggregationsstruktur der Brücke	102
Tabelle 10: Materialanforderungen der einzelnen Modellkomponenten	103

IV ABKÜRZUNGSVERZEICHNIS

Abkürzung	Ausgeschrieben
AEC	Architecture, Engineering, Construction
AG	Auftraggeber
AIA	Auftraggeberinformationsanforderungen
AN	Auftragnehmer
BSI-PAS	British Standard Institute – Publicly Available Specifications
BAP	BIM-Abwicklungsplan
BASt	Bundesanstalt für Straßenwesen
BIM	Building Information Modelling
BMVI	Bundesministerium für Verkehr und digitale Infrastruktur
BMC	BIM-based Model Checking
BR	Brücke
CDE	Common Data Environment
COBie	Construction Operations Building Information Exchange
DOC	Dokumentation
FB	Fahrbahn/Oberbau
HOAI	Honorarordnung für Architekten- und Ingenieurleistungen
IFC	Industry Foundation Class
IDM	Information Delivery Manual
LOD	Level of Development
LOG	Level of Geometry
LOI	Level of Information
LOIN	Level of Information Need
LPH	Leistungsphase
LST	Leit- und Sicherungstechnik / Stellwerkstechnik LST
MVD	Model View Definition
OIR	Organisation Information Requirements
OLA	Oberleitungsanlage
RASE	Requirements, Applicability, Selection and Exception
RRR	Requirements, Regulations, Recommendations
SOM	Semantisches Objektmodell
SWRL	Semantic Web Rule Language

V ABSTRACT

In einer digitalisierten Welt, in der auch die Planung und Ausfuhrungsdokumentation von Bauprojekten anhand eines digitalen Modells, auch BIM-Modell genannt, erfolgt, müssen diese geprüft werden. Die automatisierte Überprüfung derartiger Modelle wird BIM-based Model Checking genannt. Ziel dieser Arbeit ist es, diesen Begriff detailliert zu erläutern, die derzeitigen technischen Möglichkeiten aufzuzeigen sowie Grundlagen für weitere Arbeiten und Entwicklungen zu diesem Thema zu schaffen.

In diesem Rahmen wird die Struktur sowie der Ablauf eines BIM-based Model Checking anhand von zwei Teilbereichen der Modellprüfung beschrieben: Model Content Checking und Model Validation. Ferner wird erläutert, auf welche Weise Modellanforderungen in computerkompatible Prüfregeln transformiert werden können.

Um die technischen Möglichkeiten zu beleuchten, werden mehrere Modelle mittels einiger gängiger Model Checker auf ihre Anforderungskonformität untersucht. Es wird deutlich, dass diese Softwarelösungen dem ersten Teilbereich des BIM-based Model Checking gerecht werden, wobei die häufige Nutzung von proprietären Datenformaten mit ihren Abweichungen vom IFC-Standard als Hauptproblem des BIM-based Model Checking identifiziert wird.

Abschließend wird mittels des mvdXML-Formats eine Grundlage erarbeitet, anhand welcher das BIM-based Model Checking vollständig durchgeführt werden kann. Dadurch wird der hohe Eignungsgrad des mvdXML-Formats für das BIM-based Model Checking verdeutlicht.

1 EINLEITUNG

1.1. ZIELSETZUNG

Ziel dieser Arbeit ist es, einen Überblick über die BIM-basierte Modellprüfung (im Englischen BIM-based Model Checking, kurz BMC) und ihre Automatisierung zu geben. Hierunter fällt sowohl die Aufgabe zu klären was diese Begrifflichkeit bedeutet, als auch verschiedene Möglichkeiten der Modellprüfung voneinander abzugrenzen und zu erläutern.

Darauf basierend soll in Zusammenarbeit mit der DB Netz AG eine theoretische und praktische Grundlage für das Vorgehen und den Ablauf des BMC erarbeitet werden.

Ferner soll diese Arbeit als Grundlage für weitere Forschungen und die Lehre im Bereich des BMC dienen.

1.2. VORGEHEN UND METHODIK DER ARBEIT

Zunächst werden grundlegende Begrifflichkeiten zur Beschreibung von BIM-Leistungen die in dieser Arbeit verwendet werden, eingeführt und erläutert. Anschließend wird diskutiert, was unter BMC zu verstehen ist und es werden verschiedene mögliche Teilbereiche beleuchtet. Basierend auf den erarbeiteten Erkenntnissen werden zwei aufeinander folgende Unterdisziplinen des BMC definiert:

- Model Content Checking
- Model Validation

Daran anschließend wird die Frage geklärt, welche Modellaspekte beim BMC geprüft wird. Es wird also untersucht, welche Anforderungen ein Modell erfüllen muss und in welchen Dokumenten diese festgehalten werden. In diesem Zusammenhang erfolgt eine Beschreibung, auf welche Weise diese Anforderungen interpretiert werden können, so dass eine Überführung in computergestützte automatisierte Prüfregeln möglich ist.

Ergänzend zu diesen theoretischen Grundlagen wird in Zusammenarbeit mit der DB Netz AG ein Anforderungskatalog für vier Gewerke erarbeitet, die häufig in Projekten der DB Netz AG auftreten. Dieser Anforderungskatalog wird zur Automatisierung in BIMQ implementiert.

Im darauffolgenden Schritt wird das Model Content Checking anhand mehrerer Beispiele veranschaulicht. In diesem Zusammenhang werden mehrere gängige BMC-

Softwarelösungen bezüglich ihrer Arbeitsweisen und Funktionalität untersucht und verglichen. Als Basis dienen dabei die in BIMQ implementierten Anforderungen an die Bahngewerke.

Weiterhin werden Aspekte der Model Validation beleuchtet. Auf Basis des mvdXML-Standards wird eine Grundlage zur Validierung ausgewählter Modellaspekte konzipiert.

Abschließend wird der gesamte Prozess des BMC, von der Anforderungserstellung bis zum Abschluss des Model Validation, anhand eines reduzierten Anforderungskatalogs und eines überschaubaren Brückenmodells vorgeführt.

1.3. STATE OF THE ART

In den letzten Jahrzehnten hat sich die Welt stark gewandelt. Insbesondere digitale Technologien haben die Industrie revolutioniert und so spricht man heute oft von der Industrie 4.0. Auch vor der AEC Industrie macht diese Entwicklung nicht Halt [1]. Mit Konrad Zuse legte sogar ein Bauingenieur wesentliche Grundlagen für die gesamte Digitalisierung. Dennoch liegt die Baubranche verglichen mit anderen Industriezweigen bei der Verwendung digitaler Methoden und Werkzeuge zurück.

Das Hauptaugenmerk der Digitalisierung im Bau liegt in den letzten Jahren auf der BIM-Methodik [2]. Im Kern beschreibt diese die digitale Zusammenarbeit der verschiedenen Gewerke und Beteiligten in der Planung, Ausführung und dem Betrieb baulicher Projekte. In einigen Ländern ist die Nutzung des BIM bereits weit vorangeschritten. Beispielsweise ist die BIM-Methodik in Großbritannien als Teil des Planungs- und Bauprozesses fest vorgeschrieben.

Die Bundesrepublik Deutschland liegt im Vergleich noch zurück. Daher beschloss das Bundesministerium für Verkehr und digitale Infrastruktur 2015 den „Dreistufenplan zur Einführung der BIM-Methodik im Infrastrukturwesen“ [2], welcher die BIM-Methodik seit 2021 für alle neu ausgeschrieben Infrastrukturprojekte vorgibt.

In diesem Prozess spielt die Prüfung erbrachter BIM-Leistungen eine maßgebende Rolle, sowohl bezogen auf die Qualität der erbrachten Leistungen als auch in der Schaffung von Vertrauen in die BIM-Methodik. Zusätzlich könnten durch automatisierte Prüfungen von BIM-Modellen Genehmigungsverfahren und Koordinationsprozesse beschleunigt werden.

Während die Prüfung von BIM-Modellen in den Prozessen der BIM-Methodik fest eingeplant ist, ist die Durchführung der Prüfung selbst dennoch vergleichsweise wenig definiert und automatisiert. Wie im Rahmen dieser Arbeit noch zu sehen sein wird, sind die Definitionen der Modellprüfung meist sehr allgemein gehalten [3] und oft erfolgt die Prüfung von BIM-Modellen noch manuell. Hierbei muss jedoch unterschieden werden zwischen

der geometrischen Prüfung, welche in vielen Softwarelösungen wie *Navisworks* oder *Desite md pro* bereits weit vorangetrieben wurde und der semantischen Prüfung von BIM-Modellen.

Für Letztere gibt es Softwarelösungen zur Regelerstellung wie BIMQ und einige Softwarelösungen zur automatischen Prüfung ([4–6]). Diese beschränken sich jedoch meist auf die Attribute der Modellkomponenten und hier teilweise auch nur auf die Prüfung ihrer Existenz. Eine darüber hinausgehende Prüfung semantischer Aspekte, wie beispielsweise der Verknüpfung der Modellkomponenten untereinander, ist in den meisten Fällen höchstens rudimentär möglich. Dies ist unter anderem der Tatsache geschuldet, dass die meisten Softwarelösungen proprietäre Datenformate nutzen, anstatt den durch die DIN EN 16739 [7] als Standard für die AEC-Industrie vorgesehene IFC-Standard (siehe 1.5.1).

1.4. NORMEN, RICHTLINIEN UND REGULARIEN

Die Ausgangsbasis für diese Arbeit bildet eine Übersicht der gültigen Normen und Richtlinien bezüglich BIM in Deutschland. Da der BSI-PAS 1192 als Orientierung für die Europäische Normreihe dient, sind auch die BIM-spezifischen Teile des BSI-PAS 1192 aufgelistet.

Tabelle 1: Übersicht der Normen nach CEN TC 442

Normbezeichnung	Inhaltliche Aspekte
DIN EN ISO 12006-2	Klassifizierung der gebauten Umwelt: Konzeptionen, Aspekte, Struktur, Beispiele
DIN EN ISO 12006-3	EXPRESS-Spezifikation eines sprachunabhängigen Informationsmodells
DIN EN ISO 16739 (IFC)	Strukturelle Beschreibung des IFC, Begriffe
DIN EN ISO 16757-1	Katalogstrukturen, -Einflüsse, -Beispiele, Semantik
DIN EN ISO 16757-2	Geometrien, Darstellung, Detailierungsgrade, EXPRESS-Spezifikationen diverser Geometrien
DIN EN ISO 19650-1	Definition und Erläuterungen BIM-relevanter Begriffe
DIN EN ISO 19650-2	BIM für die Planungs-, Bauausführungs- und Inbetriebnahmephase: Struktureller Ablauf, BIM-Rollen, BIM-Aufgaben, Informationsanforderungen
DIN EN ISO 19650-3	BIM für das Gebäudemanagement: Begrifflichkeiten, OIR, BIM-Rollen, BIM-Aufgaben, Informationsanforderungen, Datenmanagement
DIN EN ISO 19650-5	BIM-Sicherheit: Sicherheitsaspekte und Risiken, BIM-Rollen, Erstellung von Sicherheitskonzepten und Strukturen

CEN/TR 17439	Leitfaden zur Implementierung von ISO 19650-1 und -2: u.a. EIR, AIA, BAP, CDE
DIN EN 17412	BIM-Definitionsgrade: LOG, LOI, Level of Documentation, definition des Level of Information Need (LOIN)
DIN EN ISO 21597-1	Struktur und Organisation digitaler Informationen als Container
DIN EN ISO 21597-2	Dynamische Semantik: Erweiterung der DIN EN ISO 21597-1 zur Erweiterung von Containern um Anwenderspezifische Schemata
DIN EN 23386	Merkmalsbeschreibung, -Erstellung, -Pflege, Datenkataloge und Datenkatalognetzwerke
DIN EN 23387	Datenvorlagen für Bauobjekte: Datenvorlagenstruktur (UML), Verknüpfung mit IFC, EXPRESS-Spezifikation von Datenvorlagen

[8-25, 7]

Tabelle 2: British Standard BSI-PAS 1192

Norm	Inhaltliche Aspekte
BSI-PAS 1192-2	BIM für die Planung und Bauausführung: EIR, AIA, BAP, CDE
BSI-PAS 1192-3	BIM für das Gebäudemanagement: Begrifflichkeiten, Prozessstrukturen, BIM-Rollen, CDE, Datenmanagement
BSI-PAS 1192-4	Leitfaden zur Nutzung von COBie
BSI-PAS 1192-5	BIM-Sicherheitsmanagement: Sicherheitsaspekte und Risiken, BIM-Rollen, Erstellung von Sicherheitskonzepten und Strukturen
BSI-PAS 1192-6	Nutzung und Schutz von Sicherheits- und Gesundheitsinformationen in BIM: Risikobewertung, Datenhaltung und -Teilung, Strategiekonzeption, Implementierung

[26-30]

Tabelle 3: Weitere BIM-spezifische Normen

Norm	Inhaltliche Aspekte
DIN EN ISO 29481-1 (IDM)	Bauwerksinformationsmodelle – Methodik und Format: IDM-Konzept, -Erstellung, -Prozessstrukturen, -Muster und -Vorlagen
ISO 29481-2 (IDM)	Bauwerksinformationsmodelle – Interaktionsframework: IDM-Komponenten, Interaktionsplan, Interaktionsframeworkstrukturen, EXPRESS Darstellung der Frameworkkomponenten

[10, 9]

Tabelle 4: Richtlinien des VDI 2552

Richtlinienblatt der VDI 2552	Inhaltliche Aspekte	Status
1	Grundlagen: Erläuterung der Methodik und Teilbereiche	Fertig
2	Begrifflichkeiten	Entwurf
3	Modellbasierte Mengenermittlung zur Kostenplanung, Terminplanung, Vergabe und Abrechnung: u.a. Zielsetzungen, Bauteiltypen, Fertigstellungsgrade, Regelungen	Fertig
4	Anforderungen an den Datenaustausch: Übergabe, IDM, BIM-Anwendungsfälle, Modellarten, -Inhalte, -Richtlinien	Fertig
5	Datenmanagement: CDE, BIM-Daten, Kooperative Bearbeitung, Technische Umsetzung	Fertig
6	Betrieb	Projekte
7	Prozesse: Anforderungen, Informationsaustausch, Rollen, Informationslieferungen	Fertig
8.1	Qualifikationen-Basiskenntnisse: Leitbild, Übersicht Basiskenntnisse	Fertig
8.2	Qualifikationen-Erweiterte Kenntnisse	Projekt
9	Klassifikationssysteme: Klassifikationssysteme inkl. Bestandteile, Verwendung, Austausch	Entwurf
10	AIA und BAP: Strukturen, Abläufe, Zielsetzungen	Fertig
11.1-11.7	Informationsaustauschanforderungen: Verschiedene Teilgebiete	Projekt

[31-39]

Im Rahmen des Dreistufenplans zur Einführung der BIM-Methodik im Infrastrukturwesen wurden von Seiten der BIM4INFRA2020 die nachfolgenden Handreichungen ausgearbeitet.

Tabelle 5: Handreichungen der BIM4INFRA2020

Handreichungen und Leitfäden Nummer	Inhaltliche Aspekte	Quelle
1	Grundlagen und BIM-Gesamtprozess	(12)
2	Leitfaden und Muster für AIA: Zu berücksichtigende Aspekte, Leistungsbeschreibungen, Mustervorlagen	(13)
3	Leitfaden und Muster für BAP: Zu berücksichtigende Aspekte, Leistungsbeschreibungen, Ablaufbeschreibungen, Mustervorlagen	(14)
4	Leitfaden zur Leistungsbeschreibung: BIM-spezifische Vertragsanlagen, Musterstruktur der Leistungsbeschreibung	(15)
5	Muster besonderer Vertragsbedingungen: BIM-spezifische Aspekte der Vertragsunterlagen	(16)
6	Steckbriefe der wichtigsten BIM-Anwendungsfälle: Übersicht, Erläuterung, Zuordnung zu den Leistungsphasen	(17)
7	Handreichungen BIM-Fachmodelle und Ausarbeitungsgrad: Übersicht, Erläuterung, Musterbeschreibungen Ausarbeitungsgrade	(18)
8	Neutraler Datenaustausch im Überblick: Intention, Open BIM	(19)
9	Datenaustausch mit IFC: Erläuterung IFC, Spezifikationen für Revit und Nemetschek (stand 2018)	(20)
10	Technologien im BIM-Umfeld: Beispiele und zugehörige Datenformate (stand 2019), Ausschreibungsempfehlungen	(21)

[40–49]

1.5. STANDARDISIERTE FORMATE

Die nachfolgend beschriebenen Standards sind zum einen mit **IFC** das standardisierte Datenaustauschformat der AEC-Industrie, zum anderen das standardisierte Format zur Beschreibung von Prüfregeln für Modelle im IFC-Format, **mvdXML**.

1.5.1. DER IFC-STANDARD

Die Norm DIN EN 19739 legt den IFC-Standard als das Standard-Datenaustauschformat für die AEC-Industrie fest. Hierbei werden analog zu den Regelungen der IFC-Spezifikationen der buildingSMART ev. die einzelnen Begrifflichkeiten definiert und beschrieben. Der IFC-Standard stellt gleichzeitig ein mögliches Metamodell für weite Teile der AEC-Leistungen dar.

Der IFC Standard ist der Struktur nach objektorientiert und gliedert die Funktionen und Informationen der AEC-Industrie entsprechend des Vererbungskonzepts. Dabei werden die Entitäten in möglichst allgemeingültiger Form beschrieben. Die individuelle, anwendungsspezifische Anpassung der Entitäten erfolgt mittels Zuweisung entsprechender Property Sets.

Eine große Bedeutung kommt im IFC Standard den Relationen zu, also den Verknüpfungen der verschiedenen Entitäten untereinander. Für diese werden teilweise sogar eigene Entitäten definiert, sogenannte „objectified Relationships“.

Der IFC Standard wird von der international agierenden buildingSMART kontinuierlich weiterentwickelt. Die neuste Version ist der IFC 4.3 RC3 [50], in dem als eine der Erweiterungen Teile des in den letzten Jahren entwickelten IFC INFRA integriert wurden. Der Teilbereich IFC Bridge wurde unter anderem als Teil des Dreistufenplans des BMVI zur Einführung der BIM-Methodik im Infrastrukturwesen erarbeitet [2, 51, 52].

Aufgrund der nachlaufenden Einarbeitung des IFC Standards in die entsprechenden Modellierungssoftwarelösungen, wird häufig auf die Versionen IFC2x3-Standard [53] und IFC4Add2 [54] zur Beschreibung von BIM-Leistungen und zum Datenaustausch zurückgegriffen.

Auf die für die Zielsetzung der vorliegenden Arbeit relevanten Aspekte des IFC wird an den entsprechenden Stellen detailliert eingegangen (siehe beispielsweise 5.3).

1.5.2. DER MVDXML-STANDARD

Bei mvdXML handelt es sich um einen offenen Standard zur Codierung von Model View Definitions (MVD) [55], der von buildingSMART zur Verfügung gestellt wird [56]. MVDs beschreiben Teilmengen des IFC-Standards, welche den Exchange Requirements der AEC-Industrie genügen [57]. Die mvdXML-Dateien können entweder manuell erstellt werden

oder unter Nutzung entsprechender Softwarelösungen wie dem IFC Documentation Generator (1.6.5) oder BIMQ (1.6.1).

Eine detaillierte Beschreibung der verschiedenen Aspekte einer mvdXML-Datei findet sich in [58] und Ergänzungen in [59]. In mvdXML-Dateien können auf Basis von Templates, welche die IFC-Strukturen widerspiegeln, Regeln und Einschränkungen für verschiedenste Entitäten und ihre Attribute definiert werden. Mittels dieser Regeln können die BIM-Leistungen geprüft werden. Das heißt: während der IFC-Standard als Format zur Beschreibung und zum Austausch von BIM-Modellen dient, beschreibt mvdXML die Regeln welche die IFC-Modelle erfüllen müssen und erlaubt eine Prüfung eben jener anhand der mvdXML-Regeln.

Ziel ist es, mittels mvdXML eine Ausgangsbasis zur Validierung von Modellen zu schaffen [59]. Abbildung 1 zeigt schematisch, wie mvdXML in diesem Zusammenhang verwendet werden kann:

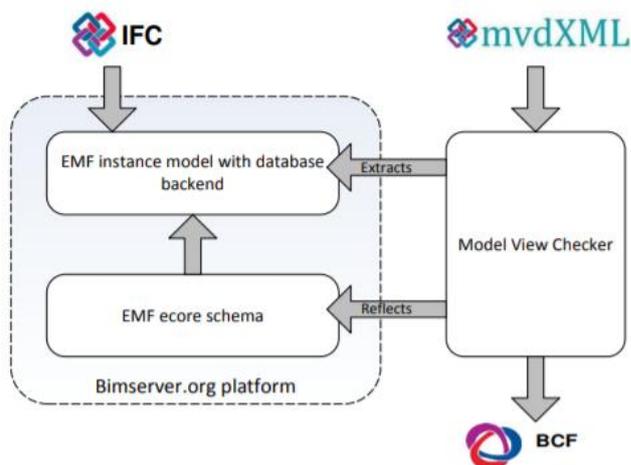


Abbildung 1: Modellprüfung nach [56]

Die Modelle sollen mit der mvdXML Datei abgeglichen und entsprechende Auswertungsberichte in Form einer BCF-Datei (BIM Collaboration Format) ausgegeben werden. BCF ist ein offener Standard zur Darstellung prüfregelabhängiger Modellfehler [56].

Eine Herausforderung bei der Nutzung von mvdXML ist es, Tools zu finden, welche dieses Format unterstützen. Hier bieten sich unter anderem der FZKViewer (1.6.6) in eingeschränkter Form an oder alternativ der Xbim Explorer (1.6.7). Im Gegensatz zu diesen Lösungen nutzen die meisten Model Checker eigene Formate (1.6).

1.6. GENUTZTE SOFTWARELÖSUNGEN

Im Nachfolgenden wird ein kurzer Überblick über die genutzten Softwarelösungen mit ihren grundlegenden Funktionsweisen, Lösungsansätzen und Einschränkungen gegeben. Detaillierte Beschreibungen der jeweiligen Modellprüfungen erfolgen in späteren Kapiteln.

1.6.1. BIMQ

Bei BIMQ handelt es sich um ein Modellinformationsmanagementsystem zur Konzeption und Fortschreibung von Modellstrukturen und -anforderungen sowie zur Unterstützung einer automatisierten anforderungsbasierten Modellprüfung.

BIMQ ermöglicht es dem Anwender, komplexe und detailreiche Anforderungskataloge an zu erbringende BIM-Leistungen auf Basis von Modellen, Elementen und Eigenschaften zu erstellen. Hierbei können die Anforderungen spezifizierten Leistungsbildern und Anwendungsfällen zugewiesen werden.

Die Anforderungen an die BIM-Leistungen können sowohl auf BIM-Datenaustauschformate wie IFC2x3 und Ifc4Add2 als auch auf softwarespezifische Formate für beispielsweise Revit und ProVI abgebildet werden.

Ergänzend ermöglicht es BIMQ, zur Unterstützung der AIA (3.1.1) Objektkarten, also übersichtliche Darstellungen aller Modellkomponenten zu erstellen. Diese verdeutlichen dem AN die Struktur und die Anforderungen, welche an die Modellkomponenten gestellt werden.

Ferner bietet BIMQ die Möglichkeit, die Anforderungen in Prüfredeln zu übersetzen. Die Prüfredeln können dabei in Datenformate der gängigen Model Checker *Solibri*, *Desite* und *BIMcollab ZOOM* [60–62] sowie in den offenen Standard mvdXML exportiert werden.

1.6.2. SOLIBRI

Die Softwarelösung „*Solibri*“ wird von *SOLIBRI INC.* als Tochter der *Nemetschek Group* [4] entwickelt und vertrieben. Sie bieten wie auch *Desite md pro* neben den Funktionen eines Model Viewers (Visualisierung und Wiedergabe der Properties) die Möglichkeit zur Prüfung der Modelle.

Solibri bietet hierfür eine Auswahl an Vorlagen für verschiedene Arten der Modellprüfung, darunter auch die später durchgeführte Vollständigkeitsprüfung (Kapitel 4).

Wie auch bei den nachfolgend aufgeführten Produkten *Desite md pro* und *BIMcollab ZOOM* besteht die Möglichkeit, die in BIMQ generierten Prüfredeln in eine für *Solibri* nutzbare Excel-Datei zu exportieren und diese in Prüfredeln zu integrieren [61]. Im Ruleset

Manager können Prüfregelein anhand von Vorlagen generiert werden. Hierbei können die Regeln der BIMQ-Exporte in die einzelnen PrüfregeleinKomponenten importiert werden.

Die Resultate der Prüfungen werden entsprechend ihrer „Gewichtung“ sortiert und als Bericht ausgegeben. Die Fehlermeldungen können auf verschiedene Weise strukturiert und ausgegeben werden.

Solibri arbeitet mit dem IFC-Format und stellt auch die Relationen in eingeschränkter Weise dar.

1.6.3. DESITE MD PRO

Beim „*Desite md pro*“ handelt es sich um ein BIM-Management-Tool der Firma *thinkproject* [5], das sowohl einen Model Viewer als auch einen Model Checker enthält.

Die Modellprüfung erfolgt in *Desite md pro* anhand von Prüfregelein, die in der Skriptsprache „*JavaScript*“ implementiert sind. Dadurch ergibt sich eine hohe Flexibilität und eine schnelle Erweiterbarkeit der Möglichkeiten zur Prüfung, wie z.B. Vollständigkeit, die Einhaltung von Wertebereichen, etc.

Es besteht die Möglichkeit die in BIMQ generierten Prüfregelein in einen für *Desite md pro* nutzbaren, „*JavaScript*“-Code aus BIMQ zu exportieren und in *Desite md pro* zu importieren [60].

Die Nutzung dieser Prüfregelein ist unkompliziert. Sie können anders als in Solibri direkt importiert und ausgeführt werden und bedürfen keiner weiteren Bearbeitung. Die importierten Regeln werden automatisch so verwaltet, dass bei Bedarf auch nur Teile der Regelsätze ausgeführt werden können.

Die Ergebnisse einer durchgeführten Prüfung werden für alle geprüften Komponenten in einer Übersichtsdarstellung, angezeigt und können so systematisch aufgearbeitet werden.

Eine Auffälligkeit dieser Softwarelösung ist, dass sie Relationen auflöst. Beispielsweise werden bei *IfcRelContainedInSpatialStructure* die Properties der verknüpften *IfcSpatialStructure* einfach als PropertySet an die in ihr enthaltenen, Objekte angehängt.

Dies wiederum schränkt die Möglichkeiten in der Prüfung einer Modellstruktur ein. Im weiteren Verlauf dieser Arbeit wird „*Desite md pro*“ kurz „*Desite*“ genannt.

1.6.4. BIMCOLLAB ZOOM

Mit „*BIMcollab ZOOM*“ bietet das Unternehmen *BIMcollab* eine BIM-Softwarelösung an, die den reinen Model Viewer um eine Clash Detection und um Smart-Views erweitert [6].

Smart-Views zeigen das Ergebnis der Prüfung von Modellkomponenten in einer farbco-dierten graphischen Darstellung an. Anhand der Farben lässt sich sofort erkennen, ob

eine Komponente die Prüfung erfolgreich bestanden hat oder ob Fehler vorliegen. Im Gegensatz zu den zuvor beschriebenen Softwarelösungen wird kein Fehlerbericht ausgegeben.

Auf die gleiche Weise können beispielsweise alle Objekte markiert werden, die ein bestimmtes Attribut aufweisen.

Wie auch die zuvor aufgeführten Softwarelösungen besteht die Möglichkeit, die in BIMQ generierten Prüfregelein in eine für *BIMcollab ZOOM* nutzbare „*bcsv*“-Datei zu exportieren. Beim einfach durchzuführenden Import in BIMcollab ZOOM [62] werden die Prüfregelein automatisch in Smart-Views übersetzt, welche ohne weitere Bearbeitung direkt ausgeführt werden können.

Relationen können in diesem Tool nicht direkt eingesehen oder abgerufen werden.

Die Grundversion, bei der im Wesentlichen nur die Clash Detection Funktionalität eingeschränkt ist, kann kostenlos heruntergeladen werden.

1.6.5. IFC DOCUMENTATION GENERATOR

Dieses Freeware-Tool von buildingSMART wurde ursprünglich zur strukturierten Darstellung und Weiterverwendung von IFC-Daten zur Verfügung gestellt. Mittlerweile wird es von GitHub gepflegt und veröffentlicht [63].

Dieses Tool unterstützt insbesondere die Erstellung von MVD-Dateien. Dazu können verschiedenste Templates erzeugt werden, welche bedarfsgerecht in entsprechende Anforderungen an MVDs eingefügt werden können.

Die Erstellung der Templates und Modellanforderungen erfolgt anhand einer Art Entscheidungsbaum. Dieser leitet den Anwender für jede Entität und jedes Attribut durch sämtliche Möglichkeiten der Präzisierung und Verknüpfung (für ein Beispiel siehe Abbildung 2). Diese Art der Modellierung wird in 2.1.4 nochmals aufgegriffen.

Wie in Kapitel 5.3 noch zu sehen sein wird, gibt es im IFC Documentation Generator einige Probleme bei der Erstellung der mvdXML-Dateien, weshalb es insbesondere bei komplexeren Anforderungen sinnvoll ist, die generierte mvdXML-Datei im Code zu erweitern.

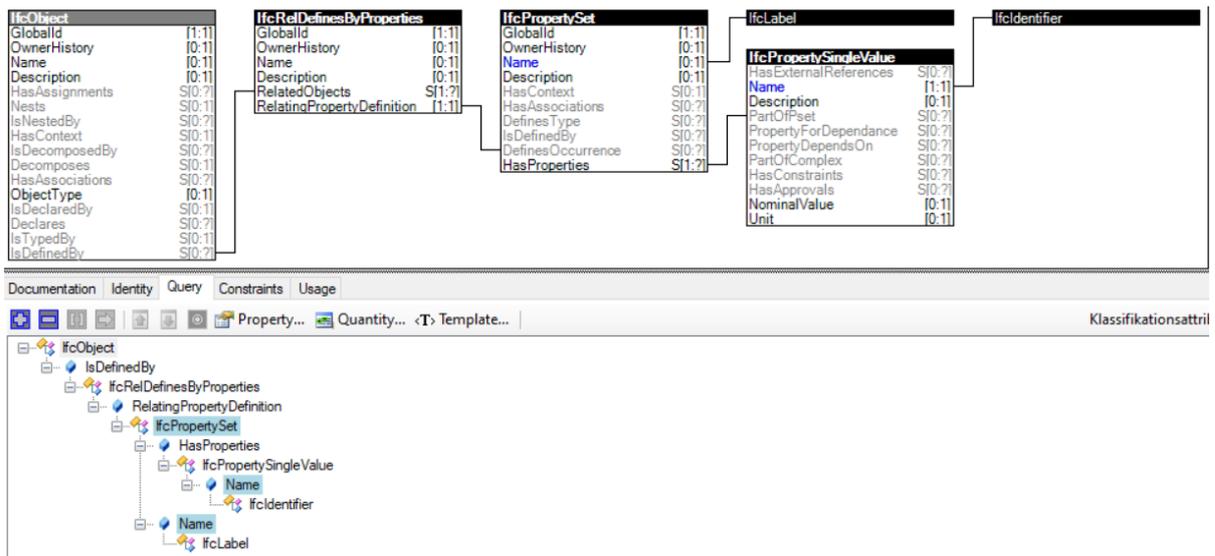


Abbildung 2: Ein Beispiel einer IFC Documentation Generator Datei

1.6.6. FZKVIEWER

Beim *FZKViewer* handelt es sich um eine Freeware die vom „Institut for Automation and Applied Informatics“ des KIT veröffentlicht und weiterentwickelt wird [64]. Der *FZKViewer* ist ein Model Viewer, der im Gegensatz zu den meisten kommerziellen Pendanten und Model Checkern kein eigenes Datenformat nutzt oder die IFC-Dateien abwandelt. Er arbeitet auf Basis des EXPRESS-Codes der Modelle und erlaubt dem Anwender auch diese Modellebene zu erfassen.

Zusätzlich ist der *FZKViewer* seit der Freigabe der Version 6.0 am 16.07.2021 [64] auch in der Lage, mvdXML-Dateien einzulesen und die Modelle bezüglich dieser zu prüfen. Die Resultate werden in einem detaillierten Ergebnisbericht ausgegeben. Die noch sehr fehleranfällige Ausführung der mvdXML-Dateien bietet allerdings noch Raum für Verbesserungen. Gleiches gilt für die fehlende graphische Darstellung der fehlerhaften Modellkomponenten.

1.6.7. XBIM XPLORER

Beim *Xbim Xplorer* handelt es sich um ein open-source BIM-Toolkit, welches um diverse Plug-Ins erweiterbar ist [65]. Der *Xbim Xplorer* bietet sämtliche Möglichkeiten eines Model-Viewers. Darüber hinaus stehen in der Grundversion die Möglichkeit zur Prüfung der IFC-Konformität der Modellkomponenten und eine Kommandokonsole zur Codierung zur Verfügung.

Im Rahmen dieser Arbeit wurde insbesondere das mvdXML-Plug-In genutzt, welches den Import von mvdXML-Dateien erlaubt. BIM-Modelle können anhand der mvdXML-Dateien geprüft werden. Dies erfolgt weniger fehleranfällig als im *FZK-Viewer* und die Modelle

können ähnlich wie bei BIMcollab ZOOM entsprechend der Prüfung farblich markiert werden. Die Fehlermeldungen sind jedoch weniger detailliert als beim FZKViewer und Fehler der mvdXML-Dateien werden selten angezeigt.

Kritisch ist für diese Softwarelösung die Instabilität der Visualisierung der Geometrie.

Anmerkung: Zur Installation der Plug-Ins wählt man unter View Settings aus und anschließend den Reiter ‚Plugins‘. Dort kann man das Plug-In installieren. Man muss aber nach dem Laden aber noch „enable“ auswählen.

2 GRUNDLAGEN DER MODELLPRÜFUNG

2.1. GRUNDBEGRIFFE

In diesem Kapitel soll eine theoretische Wissensgrundlage geschaffen werden und die grundsätzliche Einordnung der praktischen Arbeitsleistung dieser Diplomarbeit erfolgen.

2.1.1. PROPERTIES

Der grundlegendste Begriff, welcher zu definieren ist, ist der des *Properties*, also einer Eigenschaft.

In den meisten BIM-Modellierungstools und BIM-Standards [7, 50, 53, 66, 67] werden Objekte über Klassen beschrieben. Diese werden bezüglich ihrer Funktionen, Verknüpfungen, Kennwerte und andere Informationsaspekte über ihre, meist als Properties bezeichneten, Eigenschaften beschrieben. Oftmals werden Properties auch mit Attributen gleichgesetzt, wobei letztere nur eine Teilmenge der Properties darstellen.

In der DIN EN 16739 [7], aber auch im COBie [66] und beispielsweise in den Terminologie-Definitionen der Vereinigten Staaten [67], werden Properties wie folgt definiert:

„Informationseinheit innerhalb einer Entität, die durch einen bestimmten Typ oder einen Bezug zu einer bestimmten Entität definiert wird“ [7]

Entsprechend können Properties eine Vielzahl von Informationen beschreiben.

Am naheliegendsten sind gängige Datentypen wie

- **Zahlen:** Integer, Double, Long, etc.
- **Wahrheitswerte:** Bool
- **Zeichenketten:** String
 - Hierbei kann/muss weiter spezifiziert werden. Als Datentyp mag es sich bei einem Property zwar technisch um eine Zeichenkette handeln, inhaltlich kann es sich aber um einen beschreibenden Text oder lediglich um Kennzeichen wie beispielsweise IDs oder gewerkspezifische Abkürzungen handeln.
- **Relationen/Verknüpfungen:** In diesem Fall beschreibt das Property die Relation zwischen entsprechenden Entitäten. Beispielsweise kann ein Widerlager auf die Brücke verweisen, deren Bestandteil es ist oder einer Tür wird der Typ ‚Brand-schutztür‘ zugewiesen.

Wie wir später sehen werden (Kapitel 3, 4 und 5), ist es in diesem Fall schwierig den Datentyp des Properties festzulegen. Dieser hängt von den Beschränkungen der genutzten Software, der vorgesehenen Art der Prüfung, Nutzung, etc. ab. Beispielweise bietet IFC eine Vielzahl möglicher Relationen, teilweise mit eigenen Entitäten und Referenzen. In Desite hingegen werden Relationen in PropertySets (2.1.2) übersetzt, die den Entitäten zugewiesen werden. Revit hingegen erlaubt nur eine begrenzte Auswahl an Relationen.

Neben den Datentypen wird in IFC noch die Art/Funktionsweise der Properties bzw. Attribute unterschieden [7]:

- **Direkte Properties:** Der Entität eindeutig zugeordnete Werte beliebiger Datentypen.
- **Inverse Properties:** Diese erlauben das Abfragen von in Relationen stehenden Daten.
- **Abgeleitete Properties:** Diese werden aus anderen Properties abgeleitet.

Um Redundanz zu vermeiden, muss hier stets genau überlegt werden welche Attribute in einer Entität gespeichert werden müssen.

Neben den in den IFC-Spezifikationen aufgeführten Properties [50, 53] können IFC-konform alle Arten von benötigten Properties erzeugt und genutzt werden.

2.1.2. PROPERTY SETS

Property Sets beschreiben eine Ansammlung/Bündelung von Eigenschaften welche eine oder mehrere Mindestanforderung(en) erfüllen [68]. Die Bündelung von Properties in PropertySets erlaubt eine strukturierte Übersicht und Gliederung sowie eine mögliche thematische Einordnung.

Da der IFC-Standard versucht, Entitäten so allgemein wie möglich zu halten erfolgt die bedarfsgerechte Individualisierung von Objekten über die Zuweisung von PropertySets. Diese erfolgt entweder über die Relation ***ifcRelDefinesByProperties*** oder indirekt über die Relation ***ifcRelDefinesByType***. Letztere weist einem Objekt typbezogene Gruppen von PropertySets zu.

Für die Zusammensetzung und Auswahl von PropertySets gibt es im Bereich der AEC-Industrie keine allgemeingültigen Regeln. Vielmehr erfolgt die Zusammenstellung projekt- und anwendungsbezogen.

Teilweise ergeben sich dadurch unübersichtliche wenig nutzbare PropertySets (3.3.1). Mangels Erfahrung mit der BIM-Methodik im Infrastrukturbereich werden Properties häufig der Einfachheit halber allesamt in einem oder zwei PropertySets ohne sinnvolle Strukturierung zusammengefasst. Es handelt sich bei den realisierten PropertySets häufig um Kompromisse.

Grundsätzlich liegt eine thematische Struktur nahe wie beispielsweise „kaufmännische Eigenschaften“, „physikalische Eigenschaften“, „Typeigenschaften“. Aber auch andere Arten der Gliederung sind möglich. Wie wir später sehen werden (3.3) können die PropertySets auch zur „Simulation“ einer Vererbungsstruktur genutzt werden, indem die Properties der beteiligten Entitäten der Vererbungskette in jeweiligen PropertySets zusammengefasst und den Klassen zugeordnet werden. Ein Beispiel für diese Vorgehensweise ist die Umsetzung der SOM-Strukturen in BIMQ, siehe 3.3.5.

2.1.3. FERTIGSTELLUNGSGRAD VON BIM-LEISTUNGEN

Im nachfolgenden Absatz wird teilweise auf die Ergebnisse und Recherchen einer vorangegangenen Projektarbeit zurückgegriffen [69].

Gängige Begrifflichkeiten der Beschreibung von Fertigstellungsgraden von BIM-Leistungen sind der LOG, LOI und LOD. Diese sind in vielen Normen definiert und beschrieben wie beispielsweise dem VDI 2552 Blatt 2 [33] und der DIN EN 16757-2 [21].

Level of Geometry (LOG)

Der LOG gibt den geometrischen Detailgrad eines Objektes wieder. Ein Objekt kann unterschiedlich detailliert dargestellt werden und je nach Zielsetzung sind unterschiedliche geometrische Detailgrade notwendig oder angemessen.

Will man die möglichen Anordnungen von Möbeln in einem Raum vergleichen reicht es beispielsweise aus, die Möbel durch Quader oder Rechtecke darzustellen, welche die größten Abmessungen angeben. Will man hingegen ein Möbelstück bewerben, benötigt man eine detaillierte und wirklichkeitsnahe Darstellung.

Der LOG ist der Versuch, den Grad der geometrischen Annäherung eines Modells oder eines einzelnen Objektes an die Realität zu bemessen. Sein Spektrum liegt zwischen einer Bounding Box und einem geometrischen Zwilling.

Die Steigerung der Genauigkeit wird in Abbildung 3 schematisch für ein simples Haus dargestellt. Ausgehend von einer Box, welche die größten Abmessungen des Bauwerkes sichert, wird das Objekt immer detaillierter dargestellt.

Schnell wird an diesem Beispiel klar, wie schwierig es ist, den LOG allgemeingültig zu definieren. Die Bounding Box als einfachster LOG ist noch eindeutig, aber alles Weitere gestaltet sich diffiziler.

Für das Haus stellt sich beispielsweise die Frage, für welchen LOG welche Details dargestellt werden sollen. Auch die Aufgabe, den LOG so festzulegen, dass er für mehrere Bauteile gültig ist, erweist sich als herausfordernd. Daher wird LOG verbal beschrieben und an die Anforderungen an die Modelle angepasst. Wie noch diskutiert werden wird, kann

die Beschreibung auf verschiedene Weisen erfolgen (siehe 3.1.1). Entsprechend ist eine genormte Bewertung des LOG in eindeutige Stufen/Kategorien bisher nicht erfolgt. In der DIN EN 17412 wird lediglich eine Steigerung der Detaillierung dargestellt ohne sie aber zu gewichten [16].

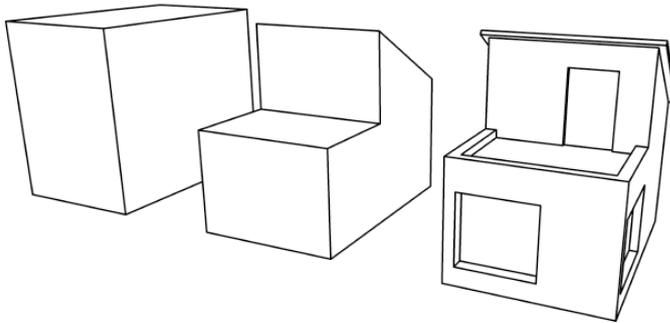


Abbildung 3: Beispielüberlegungen eines LOG [16]

Es gibt für das LOG auch die Bezeichnung Level of Detail (LOD). Da dies jedoch zu Verwechslungen mit dem Level of Development (LOD) führen kann wird in dieser Arbeit die Bezeichnung LOG verwendet.

Level of Information (LOI)

Ein Bauteil/Objekt besteht aus mehr als nur seiner Geometrie und BIM ist mehr als nur eine Sammlung von 3D-Modellen. Bei einer Wand beispielweise sind nicht nur ihre Abmessungen relevant, sondern es muss auch bekannt sein, aus welchem Material sie besteht. Eine Betonwand kann ein Fertigteil sein oder vor Ort geschalt und betoniert werden. Sie kann jedwede Art von Betonfestigkeit haben, der Zement stammt von einem gewissen Hersteller, der Beton kann wasserundurchlässig sein und hat einen bestimmten u-Wert. All diese und noch viele weitere Informationen können einem Objekt zugewiesen werden.

Der Detailgrad bzw. die Fülle an Informationen eines Modells wird als **LOI** bezeichnet. Im Rahmen dieser Arbeit wird der Schwerpunkt der Modellprüfung auf die Modellaspekte gelegt, die mittels des LOI beschrieben werden.

Wie beim LOG stellt sich auch beim LOI die Frage nach einem geeigneten Verfahren zur Bemessung. Dies ist insbesondere deshalb herausfordernd, weil unterschiedliche Projektbeteiligte unterschiedliche Informationen benötigen. So spielt es für die Statik keinerlei Rolle, wie viel eine zu bemessende Betonwand kostet. Für die Kostenkalkulation ist diese Information jedoch essenziell, wohingegen die für den Statiker entscheidenden Festigkeitswerte der Wand in den Hintergrund treten. Entsprechend können derselben Wand im Statik-Modell die Festigkeitsinformationen zugewiesen werden und im kaufmännischen Modell die Kostenaspekte. Nun stellt sich die Frage, ob die beiden Beschreibungen dieser Wand den gleichen Grad an Information haben. Ist eine der beiden detaillierter, da

sie vielleicht ein Attribut mehr hat als die andere oder werden unterschiedliche Attributarten unterschiedlich gewichtet?

Man befindet sich also in derselben Situation wie bei der Festlegung des LOG und es liegt nahe, einen analogen Lösungsansatz zu wählen: Der LOI muss verbal, abhängig von den zu lösenden Aufgaben beschrieben werden.

Ein Ansatz diese Problematik zu umgehen, besteht darin, auf Begriffe wie den LOI zu verzichten und die Informationsanforderungen der Modellkomponenten anhand der LPHs festzulegen. Dies ist nah an der Idee, die LODs den LPHs zuzuordnen, verzichtet jedoch auf generische Regelungen wie LPH1: LOD100. Dadurch erhöht sich die Flexibilität in der Anforderungsbeschreibung. Dieser Ansatz wurde auch im SOM (3.3.1) für alle LPHs angewandt, ebenso in der im Rahmen dieser Arbeit erarbeiteten Neustrukturierung (3.3.4) für die LPH 1 mit der Option auf ein Fortschreiben für die nachfolgenden LPHs (siehe auch 3.1.2).

Level of Development (LOD)

In der Regel werden für ein Projekt nicht nur Informationen entweder des LOG oder des LOI benötigt, sondern Angaben aus beiden Bereichen. Beispielsweise ist die Statik sowohl auf die Festigkeitswerte einer Betonwand als auch auf ihre Abmessungen und die Geometrie angewiesen. Entsprechendes gilt für die Kalkulation.

Die Kombination aus dem geometrischen Detailgrad LOG und dem Informationsgehalt LOI wird als **LOD** bezeichnet.

Wie für den LOG und LOI muss auch der LOD als Kombination der beiden verbal den Anforderungen entsprechend beschrieben werden. Es gibt zwar keine einheitlich genormte Einstufung des LOD, aber eine Einteilung in LOD100-LOD500 ist weitestgehend akzeptiert [70].

Diese wirft aber bei genauer Betrachtung wieder die Frage nach einer objektiven oder logischen Bewertung auf. Es kann, wie für den LOD oder LOI bereits diskutiert wurde, keine allgemeine Regelung für die Festlegung des LOD geben. Die beiden Kombinationen Wandabmessungen/Betonkosten und Wandabmessungen/Festigkeiten stellen gleichwertige LOD dar, trotz unterschiedlicher Aspekte. Auch erhebt sich die Frage, ob ein geometrischer Zwilling in Kombination mit gerade einmal einem Attribut (z.B.: *GlobalID*) einen höheren LOD hat als eine schematische Annäherung mit einer Fülle an Attributen.

Entsprechend schwierig ist die Einteilung. Als Hilfestellung führt die BIM4INFRA in den Handreichungen Beispiele für LODs auf, wie es exemplarisch in Abbildung 4 für ein LOD 200 für eine Brückenkappe zu sehen ist.

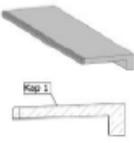
LOD	LOG	LOI
200	<p>Ein Modellelement vom Typ „Brückenkappe“ wird als Volumenkörper mit exakter Menge, Abmessung, Form, Lage und Orientierung modelliert. Der Bordstein wird als eigener Volumenkörper modelliert.</p>  <p>Abbildung: Beispielhafte Modellierung Brückenkappe LOD 300 nach Mini 2016⁴</p>	<p>Für ein Modellelement vom Typ „Brückenkappe“ werden folgende Attribute festgelegt:</p> <ul style="list-style-type: none"> ■ Name: Material <ul style="list-style-type: none"> ■ Datentyp: Text ■ Einheit: - ■ Name: Expositionsklasse <ul style="list-style-type: none"> ■ Datentyp: Text ■ Einheit: - ■ Name: Korrosionsschutz <ul style="list-style-type: none"> ■ Datentyp: Text ■ Einheit: - ■ Name: Bewehrungsgehalt <ul style="list-style-type: none"> ■ Datentyp: Dezimalzahl ■ Einheit: kg/m³ <p>Der Auftragnehmer kann zusätzliche Attribute definieren, die für die Erbringung seiner Leistungen erforderlich sind. Die Attribute werden durch den Auftragnehmer im Rahmen des BAP definiert und durch den Auftraggeber genehmigt. Es müssen konkrete Vorgaben zu Namen, Datentypen, möglichen Werten und Möglichkeiten zur automatischen Prüfung spezifiziert werden.</p>

Abbildung 4: Beispiel eines LOD nach [42]

Durch eine graphische Darstellung und prägnante Formulierungen wird verdeutlicht, welche geometrischen Informationen (LOG) und welche LOI-relevanten Angaben gefordert sind.

Die projektspezifischen Bestimmungen zu den LODs werden in den entsprechenden AIAs (3.1.1) und gegebenenfalls BAPs (3.1.2) getroffen. Trotz mehrerer Beispiele zur Festlegungen der LODs [42, 47, 70], kann es wie bereits diskutiert, keine einheitliche Regelung geben. Entsprechend werden, wie wir noch sehen werden, in den Hilfestellungen und Mustern zur Erstellung von AIAs und BAPs verschiedene Arten der Festlegung eines LODs beschrieben [32, 42, 43].

Level of Information Need (LOIN)

In der DIN EN 17412 [16] wird der LOD um den Aspekt der DOC erweitert und dadurch zum **Level of Information Need** (LOIN). DOC beschreibt hierbei die **Dokumentation** mit allen zugehörigen Aspekten. Zu diesen gehören unter anderem

- Anzahl der zu liefernden Dokumente
- die Art der Dokumente
 - Tabellen
 - Detaillisten
 - Datenblätter
 - Berichte
- Art der Verifizierung
- zu nutzende Datenaustauschformate

Dementsprechend beschreibt der DOC gewissermaßen die in den AIA und dem BAP getroffenen Regelungen bezüglich der BIM-Leistungsübergabe.

2.1.4. MODELLPRÜFUNG IM ALLGEMEINEN

Was ist unter BIM-basierter Modellprüfung beziehungsweise BIM-based Model Checking (BMC) zu verstehen?

Befragt man gängige BIM-Lehrbücher [3] erhält man eine Antwort wie

Modellprüfung oder auch Model Checking beschreibt die regelbasierte Prüfung von Modellen. Geometrie, Semantik und verlinkte Informationen werden bezüglich ihrer Übereinstimmung mit Planungsanforderungen, AIA, und Regularien ausgewertet.

Die Aufgabe der Modellprüfung besteht also darin herausfinden, ob die Ansprüche an die BIM-Leistung erfüllt sind.

Ansprüche an Modelle können verschiedener Art sein. Zu diesen gehören die Konformität mit den geforderten LOD (2.1.3). Hierunter fallen unter anderem eine vollständige und inhaltlich korrekte Attribuierung der Modellkomponenten gefordert. Darüber hinaus soll sichergestellt werden, dass alle benötigten Modellkomponenten vorhanden sind. Diese Anforderungen stellen eine Art Vollständigkeitsanforderung dar.

Aber die Anforderungen an Modelle gehen darüber hinaus. Nur weil eine Modellkomponente über alle geforderten Attribute verfügt, heißt das nicht, dass die Attribuierung korrekt erfolgt ist. BIM-Modelle sollten auch **validiert** werden. Darunter können diverse Aspekte verstanden werden, welche die Einhaltung von Normen und Vorgaben durch das Modell beschreiben oder dass die Modellkomponenten geometrisch zusammenpassen (Kollisionsprüfung).

Wie in 1.3 bereits erwähnt, erfolgen derartige Arten der Prüfung in unterschiedlichen Graden der Automatisierung. Während die Clash Detection in vielen Softwarelösungen (BIM-collab ZOOM, Solibri, Desite, Revit, Bixel) bereits weitestgehend automatisiert ist, erfolgt die attributbezogene oder strukturelle Prüfung in vielen Fällen noch manuell, durch das schrittweise händische Prüfen der einzelnen Modellkomponenten. Die automatisierten Prüfungen dieser Aspekte ist, wie wir noch sehen werden (4), je nach Softwarelösung unterschiedlich ausgereift, wenn überhaupt vorhanden.

Die händische Prüfung stellt in vielen Fällen eine Herausforderung dar, denn sie fordert von den Prüfenden eine hohe Qualifikation und großes Wissen, mutet ihnen gleichzeitig aber eine oftmals monotone Arbeit zu. Während dies schon allein aus Mangel an Fachkräften im Bereich BIM problematisch ist, kommt noch erschwerend hinzu, dass die händische Prüfung ein fehleranfälliges Verfahren ist [71]. Entsprechend wird hier an automatisierten Ansätzen gearbeitet. Ziel ist es, das Modell schon im frühen Stadium des

Erstellungsprozesses inhaltlich korrekt zu gestalten, um spät auftretende Korrekturen zu vermeiden, die Aufwand und Kosten in die Höhe treiben [71].

Darüber hinaus können noch weitere Anforderungen gestellt werden, wie z.B. die Nutzung bestimmter Datenaustauschformate, Datenstrukturen, Dokumentationen, Aspekte, welche im DOC nach DIN EN 17412 geregelt werden (siehe 2.1.3) oder rechtliche Anforderungen an die Dokumente. Entsprechende Austauschformate zur Darstellung derartiger Regeln werden beispielsweise mit LegalDocML and LegalRuleML bereitgestellt [72].

Eine systematische Unterteilung der Arten der Modellprüfung bietet Prof. Hjelseth von der Norwegian University of Life Science (Abbildung 5). Er erweitert den Begriff der Modellprüfung gegenüber der obigen Definition und beschreibt vier Arten der Modellprüfung, die er in zwei Gruppen unterteilt [73, 74].

Der Vorschlag unterscheidet zwischen „Erfüllungsprüfungen“ und „Designlösungsprüfungen“. Als Erfüllungsprüfungen (compliance checking) führt er die bereits aufgeführten Arten der Anforderungen an Modelle auf, die unstrittig als Arten der BIM-Modellprüfung gelten können. Als Designlösungsprüfungen (design solution checking) sind solche Prüfungen zu verstehen, bei welchen die Auswahl und Flexibilität der modellierten/konzipierten Entwurfslösungen geprüft wird.

Concept group	Concept type	Purpose of checking	Outcome	Examples
Compliance checking	Validation checking	Validation	pass/fail	clash detections code compliance
	Model content checking	Content of information	a filtered list	relevant information for exchange
Design solution checking	Smart object checking	Integration (adaptation)	a modified model	size of building parts (objects) related to the building (model)
	Design option checking	Guidance	options and advice	knowledge system for selecting relevant solutions

Abbildung 5: Die Unterteilung der Arten der Modellprüfung nach [73]

Die Erfüllungsprüfungen beschreiben die regelbasierte Prüfung eines Modells und gliedern sich in „Model Content Checking“ und „Validation Checking“.

Das Model Content Checking dient hier als Ausgangslage, um relevante Informationen zu extrahieren und irrelevante Informationen auszusortieren wie es in Abbildung 6 dargestellt wird.

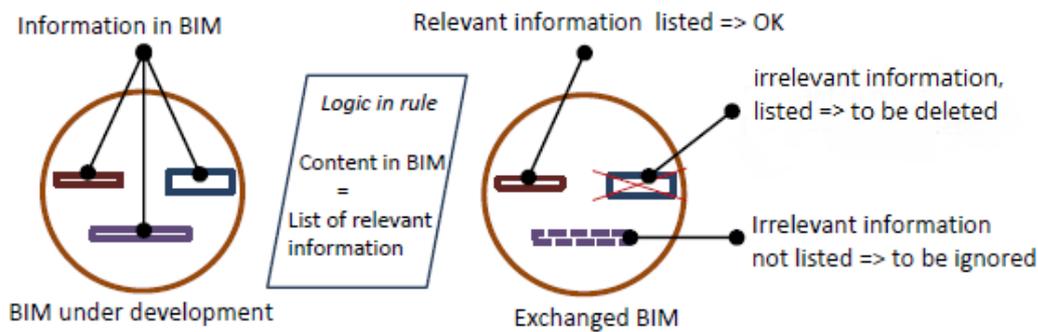


Abbildung 6: Schematische Darstellung des Model Content Checking [73]

Beim Validation Checking wird das Modell auf „Code Compliance“ [73] und „Clash Detection“ geprüft. Anders formuliert: Das Modell und die in ihm enthaltenen Informationen werden auf Regelkonformität untersucht.

Um die Ergebnisse der Prüfung richtig verstehen zu können ist es wichtig zu verstehen, dass ein validiertes Modell nicht zwangsläufig eine gute oder gar die beste Lösung ist, sondern lediglich sichergestellt ist, dass es sich um eine korrekte Lösung handelt [74].

Die zweite von Prof. Hjelseth vorgeschlagene Kategorie der Modellprüfung umfasst das „Smart Object Checking“ und „Design Object Checking“. Diese beiden Bestandteile sind nicht zwangsläufig als Modellprüfung zu bezeichnen, sondern können auch als Bestandteil des Modellierungsprozesses verstanden werden.

Als Smart Object Checking ist die „Selbstprüfung“ der einzelnen Modellkomponenten anhand ihrer Randbedingungen aufzufassen. Adaptive Bauteile [75] passen sich an die sie eingrenzenden geometrischen Strukturen an, behalten dabei aber ihre eigene geometrische Struktur bei. Alternativ kann man sich vorstellen, dass anhand von Grundrissen verschiedene Baumodelle generiert werden, die miteinander verglichen werden können, um die kostengünstigsten und optisch ansprechendsten Lösungen zu finden.

Als Beispiel für „Smart Object Checking“ führt Prof. Hjelseth eine Stahlbetonstütze auf, welche selbstständig ihren Bewehrungsgrad anpasst, abhängig von den auf sie einwirkenden Kräften: Ein Ansatz, der zwar wünschenswert ist, aber zum momentanen Zeitpunkt in Teilen noch als „Wunschdenken“ bewertet werden muss. Die momentan genutzten Modellierungstools sind noch sehr aufgabenspezifisch und auch in den Handreichungen der BIM4INFRA2020 wurde die Nachweisführung explizit nicht für die BIM-Nutzung empfohlen [46].

Mit dem Begriff Design Object Checking verbindet man die Überprüfung, ob Lösungsmöglichkeiten berücksichtigt wurden. Als Beispiel wird hier ein den Modellierungsprozess begleitender Entscheidungsbaum beschrieben. Er soll im Verlauf des Modellierungsprozesses durch die verschiedenen Lösungsmöglichkeiten führen. Eine Beispieldarstellung einer derartigen Entscheidungsführung wird in Abbildung 7 dargestellt.

In Norway ->
 Dwelling house ->
 The area is above Xm^2 ->
 A chimney and open fireplace is modeled.
 Next options to be considered:

- 1) Add a stove for heating.
- 2) Increase the volume of concrete so it acts as a heating reservoir.
- 3) Supplement with technical room for wood pellet heating in lowest floor.

Abbildung 7: Leitende Entscheidungsprüfung nach [74]

Ein weiteres Beispiel einer derart gestützten Modellierung ist die Arbeit mit dem IFC Documentation Generator (1.6.5). In diesem wird man ebenfalls durch die Modellierung geführt unter Berücksichtigung aller möglicher Pfade.

2.1.5. MODELLPRÜFUNG IM RAHMEN DIESER ARBEIT

Die Aufgabenstellung der vorliegenden Arbeit besteht in der Untersuchung der semantischen Aspekte der Modellprüfung. Diese sind den Erfüllungsprüfungen zuzuordnen und aus diesem Grund wird hier nicht weiter auf die Designlösungsprüfungen eingegangen. Der Teilbereich der Erfüllungsprüfungen ist nicht nur konform mit Model Checking-Definitionen wie sie in [3] zu finden sind. Vielmehr stellt diese Einteilung eine Grundlage für eine klare Struktur und Hierarchie der Modellprüfung dar: Eine Struktur, welche im Nachfolgenden konkretisiert werden soll.

Zunächst gilt es die beiden Teilbereiche der Erfüllungsprüfung klar voneinander abzugrenzen und die Teilbereiche zu umreißen. Hierfür muss definiert werden, was unter „Model Content“ und was unter „Modellvalidierung“ zu verstehen ist. Anhand dieser Unterteilung erfolgt das weitere Vorgehen bezüglich der Modellprüfung im Rahmen dieser Arbeit.

Tabelle 6 stellt die nachfolgend beschriebene Unterteilung des BMC in zwei Teilbereiche dar sowie im Falle der Model Validation auszugswise die zugehörigen Prüfungen.

Tabelle 6: Unterteilung des BMC in dieser Arbeit

Art der Modelprüfung	Prüfaspekte (Auszugsweise)
<i>Model Content Checking</i>	Geometrische und semantische Vollständigkeit
<i>Model Validation</i>	Clash Detection
	Strukturprüfung
	Überprüfung von Wertebereichen
	Normen und Richtlinien

Die Definition von „Model Content Checking“ erfolgt im Rahmen dieser Arbeit relativ nah an der Grundlage [73]: Als die Prüfung auf Vollständigkeit bezüglich der Geometrie und Attribuierung der Modellkomponenten.

Was bedeutet in diesem Kontext Vollständigkeit?

Vollständigkeit der Modellinformationen umfasst das Vorhandensein aller geforderten Informationen hinsichtlich aller geometrischen Teilkomponenten eines Bauteils entsprechend dem LOG sowie das Vorhandensein aller Attribute, welche bei den Modellkomponenten vorhanden sein sollen. Dies schließt ein, dass die Attribute im richtigen Datentyp vorliegen müssen, denn wie in 4.3.2 und 4.3.3 noch zu sehen sein wird, werden Attribute je nach Softwarelösung nur bei korrektem Datentyp erkannt. Ferner müssen die Attribute Teil der richtigen PropertySets sein.

Es erfolgt jedoch keine Prüfung, ob die Attribute einen bestimmten Wertebereich einhalten oder regel- und strukturtechnisch richtig sind. Dabei spielt es keine Rolle auf welche Weise die Properties einer Modellkomponente zugewiesen werden, auch wenn dies ebenfalls als Vollständigkeit aufgefasst werden könnte. So ist es irrelevant, ob die Zuweisung über einen Objekttyp oder eine **IfcRelDefinesByProperties** (5.3.1) erfolgt. Dies liegt in der Vielzahl der Möglichkeiten der Informationszuweisung zu den Modellkomponenten und der Nutzung von proprietären Datenformaten durch die Model Checker begründet. Ferner sind derartige Relationsbasierte Aspekte auch eher in der nachfolgend definierten Model Validation einzugliedern. Es wäre auch möglich gewesen, die Vollständigkeit ohne die Forderung des richtigen Datentyps eines Properties zu definieren. Anders als der Wertebereich, der je nach Projekt und Randbedingungen variieren kann, gehört der Datentyp zur Definition eines Properties dazu. Dies gilt umso mehr für die Modellierungssoftwarelösungen. In Revit lässt sich beispielweise der Datentyp eines Attributes nach dem Anlegen eines Attributes nicht mehr ändern. Sofern er falsch wäre, muss das Attribut also neu angelegt werden. Auch bei der Überführung von Modellen in den IFC-Standard müssen die Datentypen der Attribute festgelegt werden. Je später also ein falscher Datentyp erkannt wird desto größer wird der Korrekturaufwand. Zusätzlich wird im Rahmen dieser Arbeit noch deutlich werden, zu welchen Problemen es bezüglich Datentypen kommen kann. Dementsprechend wurde im Rahmen dieser Arbeit beschlossen, den Datentypen als Teil der Vollständigkeit zu verstehen.

Nach diesem Verständnis ist das Model Content Checking erfolgreich, wenn alle Modellkomponenten geometrisch und attributtechnisch vollständig sind. Die Prüfung gibt Fehlermeldungen aus, sofern Attribute nicht vorhanden oder vom falschen Datentyp sind oder sofern Geometrieaspekte fehlen. Wie in Kapitel 4 zu sehen sein wird, gibt es eine Vielzahl von Softwarelösungen, um Model Content Checking an BIM-Modellen durchzuführen. Im genannten Kapitel wird die Prüfung bezüglich der Attribute anhand einiger Beispiele veranschaulicht.

Darüber hinausgehende Prüfungen entfallen, sofern sie nicht Teil des Modellierungsprozesses sind, auf das „Validation Checking“. Dieses wird nachfolgend definiert als die Prüfung des BIM-Modells auf Einhaltung der vorgegebenen Regeln und Anforderungen, welche über die Vollständigkeit entsprechend obiger Definition hinausgehen.

Hierunter entfällt eine Vielzahl an möglichen Prüfungen wie z.B.:

- Die Einhaltung von Normen wie Mindestabstände oder Festigkeitsklassen
- Clash Detection
- Die Einhaltung gewisser Wertebereiche
- Die Korrektheit der Modellstruktur: Sind beispielsweise die Aggregationen korrekt ausgeführt und sinnvoll? In welcher Weise sind die Materialien zugewiesen?

Eine vollständige Beschreibung aller denkbaren Modellvalidierungen ist kaum möglich und auch nicht das Ziel dieser Arbeit. Es erfolgt jedoch im Kapitel 5 eine ausführlichere Betrachtung dieser Modellprüfung. Insbesondere wird dort versucht, die Validierung als solche zu erfassen. Zusätzlich werden noch einige Beispiele für Aspekte der Modellvalidierung durchgeführt und erläutert.

3 MODELLANFORDERUNGEN

In diesem Kapitel werden zunächst zwei Dokumente betrachtet, in denen die Anforderungen an BIM-Modelle geregelt sind. Im Anschluss daran stellen wir uns die Frage nach der Struktur und dem Aufbau von Prüfregeln. Insbesondere wird das Konzept RASE betrachtet, auf dem die nachfolgend erstellten Prüfregeln basieren.

3.1. AIA UND BAP

Der Schwerpunkt dieser Arbeit liegt auf der semantischen Prüfung von BIM-Modellen. Das heißt die BIM-Modelle sollen auf die unter den LOI fallenden Anforderungen geprüft werden. Darunter fallen beispielsweise die folgenden Fragestellungen:

- Welche Modellkomponenten sind relevant?
- Welche Attribute werden benötigt?
- Wie sollen die Modelle strukturiert sein?
- Welche Normen und Regularien müssen eingehalten werden?

Die jeweiligen Antworten hängen erheblich vom Umfang eines Projektes ab sowie von der Zielsetzung, mit der die BIM-Modelle weiter genutzt werden sollen. Entsprechend variieren die Anforderungen an die BIM-Modelle von Projekt zu Projekt. Die Erkenntnis zwingt zu dem Schluss, dass es keine allgemeingültigen Modellanforderungen für jedes Projekt geben kann. Diese müssen vielmehr für jedes Projekt individuell angepasst werden. Die Regelungen hierfür werden projektspezifisch in den Dokumenten Auftraggeberinformationsanforderung (AIA, 3.1.1) und BIM-Abwicklungsplan (BAP, 3.1.2) festgehalten.

Ein langfristiges Ziel ist es jedoch Standards, vor allem für die Mindestanforderungen festzulegen, wie dies auch später am Beispiel des „Semantischen Objektmodells“ (3.3) noch geschehen wird.

Die Notwendigkeit für diese Dokumente wurde bereits an vielen Stellen herausgestellt, beispielsweise in der Ausgabe 3/2021 des Build-Ing. Magazins [76]. Ebenso wurden sie in der Stufe 2 des Dreistufenplans zur Einführung der BIM-Methodik im Infrastrukturwesen besonders berücksichtigt [77–79]. Des Weiteren wird diesen beiden Dokumenten auch von Seiten der DB Netz AG, wie wir noch diskutieren werden, erhebliche Beachtung geschenkt.

Die nachfolgenden beiden Abschnitte beziehen sich unter anderem auf die Erkenntnisse einer vorangegangenen Projektarbeit [69] in der bereits Richtlinien und Vorschläge für die Struktur der AIA und des BAP betrachtet wurden. Deshalb werden die Dokumente im Nachfolgenden nur kurz definiert und bezüglich ihrer Modellanforderungsaspekte betrachtet.

3.1.1. AUFTRAGGEBERINFORMATIONSANFORDERUNGEN

Die Struktur der AIA wird, wie in [69] erläutert, anhand der Handreichungen der BIM4INFRA2020 [42], des VDI 2552 Blatt 10 [32] beschrieben. Zusätzlich werden Überlegungen, basierend auf Beispieldokumenten, Vorentwürfen und Besprechungen mit Mitarbeitern der DB Netz AG mit einbezogen. Bei Letzteren handelt es sich um noch nicht veröffentlichte Vorentwürfe sowie um die Planung eines Muster-AIA-Entwurfs für kommende Projekte.

Entsprechend der Handreichungen der BIM4INFRA2020 gilt für die AIA:

„Die AIA beschreiben aus der Sicht des Auftraggebers die Anforderungen, die ein Auftragnehmer im Rahmen der Leistungserbringung unter Verwendung von Building Information Modelling zu berücksichtigen hat.“ [42]

In den AIA werden also die zu erbringenden BIM-Leistungen hinreichend detailliert beschrieben (Anwendungsfälle, Detailgrade, Komponenten, etc.) sowie geklärt, in welcher Weise der Auftraggeber Zugriff auf die entsprechenden Daten erhält.

Die Beschreibung der Modellanforderungen kann laut den Handreichungen der BIM4INFRA2020 auf drei Arten erfolgen:

1. Die funktionale Beschreibung: Die Anforderungen an das Modell und seiner Komponenten erfolgt anhand der Funktionen, die mithilfe des Modells durchgeführt werden sollen. Beispiele hierfür sind Mengenermittlungen in verschiedenen Genauigkeiten und die Bestimmung von Vorzugsvarianten.
2. Bei der semi-detaillierten Beschreibung wird die funktionale Beschreibung um relevante Modellelemente und ihre zugehörigen Attribuierungen, Klassifikationen und konkrete Detailanforderungen ergänzt. Unter anderem kann es beispielhafte Komponentenbeschreibungen wie in Abbildung 4 geben die als Orientierung dienen. Auch können LOG oder LOI spezifische Anforderungen ergänzt werden wie das später noch beschriebene „Semantische Objektmodell“. Abgesehen von derartigen Ergänzungen verbleibt dem AN weiterhin ein großer Freiraum.
3. Im Falle der detaillierten Beschreibung werden sämtliche Modellkomponenten in allen Details sowohl geometrischer, als auch semantischer Natur beschrieben. Der AN kann nur ergänzende Vorschläge einbringen.

Die Wahl der Beschreibungsart hängt neben den Präferenzen des AG auch vom Umfang des zu erstellenden Modells und der Leistungsphase ab. Je mehr verschiedenartige Komponenten und je mehr Details benötigt werden, desto komplexer und aufwändiger wird

der Versuch einer detaillierten Beschreibung. Die funktionale Beschreibung erlaubt hier eine große Flexibilität, führt aber gleichzeitig zu einem Verlust an Einflussnahme.

Je nach Wahl der Beschreibungsmethode kann eine AIA um beispielhafte Darstellungen des LOD anhand einzelner Objekte (siehe Abbildung 4) erweitert werden oder muss für jede einzelne potenzielle Modellkomponente eine derartige Darstellung aufweisen. Auch können oder müssen Attribuierungsübersichten, Objektkarten oder Modellstrukturkonzepte wie das später ausführlich beschriebene „Semantische Objektmodell“ der DB Netz AG erweitert werden, um die Anforderungen der AIA zu präzisieren oder gar erst zu vervollständigen.

Im Laufe der Gespräche zur Erstellung der Musterentwürfe wurde die Problematik bei der Festlegung auf LODs deutlich, ebenso wie die Festlegung dieser in der AIA. Für Ersteres besteht die Herausforderung wie bereits in 2.1.3 beschrieben darin, den LOI und damit den LOD eindeutig abzugrenzen. Bei Letzterem erweist es sich als problematisch, dass es sich bei der AIA um ein Vertragsdokument handelt, welches vor Beginn der BIM-Modellerstellung vorliegen muss. Daher sind die hier beschriebenen Anforderungen nicht flexibel hinsichtlich Änderungen. Im Rahmen der Gespräche und auch der Neustrukturierung der SOM (siehe 3.3.4) wurde der Wunsch deutlich, die Modellanforderungen in der AIA als Ausgangs-/Mindestanforderungen an die BIM-Modelle zu verstehen, welche auf die LPH ausgerichtet sind, ab der der AIA beginnt. Die im Laufe des Projektes hinzukommenden Anforderungen und Wünsche sollten vielmehr im BAP (siehe 3.1.2) in einer gemeinschaftlichen Absprache zwischen AG und AN unter Einbeziehung von Fachplanern fortgeschrieben werden. Dies würde der Vorgehensweise in den Pilotprojekten der Stufe 1 des Dreistufenplans zur Einführung der BIM-Methodik im Infrastrukturwesen des BMVI entsprechen [79–83] und ist auch in den Konzeptionen eines LOD der DEGES [70] verankert.

3.1.2. BIM-ABWICKLUNGSPLAN

Wie die AIA wird auch der BAP hier nur bezüglich seiner modellkomponentenbezogenen Aspekte betrachtet, da er bereits in der Projektarbeit [69] untersucht wurde. Grundlage sind die Handreichungen der BIM4INFRA2020 [43] und die Richtlinie der VDI 2552 Blatt 10 [32] sowie Beispieldokumente, Vorentwürfe und Besprechungen mit Mitarbeitern der DB Netz AG. In den genannten Quellen sind Hilfestellungen für die Erstellung von BAPs sowie Musterentwürfe enthalten.

Während die AIA stets ein Bestandteil der Vertragsdokumente ist, wird dies beim BAP abhängig vom AG unterschiedlich gehandhabt.

In den AIA wird festgelegt „was“ erstellt werden soll. Der BAP beschreibt in Ergänzung dazu den Prozess der Erstellung, also das „wie“.

Im BAP werden die Rollen, Aufgaben und Prozesse geregelt, sofern dies in den AIA noch nicht geschehen ist. Dies umfasst auch wesentliche Aspekte der Modellprüfung: Im BAP

sind die Intervalle sowie der Umfang der Prüfungen festgehalten und wer die Prüfungen durchführt. Gegebenenfalls ist auch festgelegt, welche Softwarelösungen dabei zum Einsatz kommen. Wie in 1.3 beschrieben gibt es hier jedoch noch viel Nachholbedarf, denn die Prüfungen erfolgen oftmals manuell und teilweise nur stichprobenartig.

Wichtig ist zu betonen, dass der BAP als konsequent fortgeschriebenes Dokument eine Anpassung an neue Situationen oder geänderte Randbedingungen ermöglicht. Mit dieser Flexibilität bietet er ein enormes Potential zur Beschreibung der Modellanforderungen, was sich schon in der Stufe 1 des Dreistufenplans zu Einführung der BIM-Methodik im Infrastrukturwesen des BMVI als sehr günstig herausgestellt hat [80–83]. Auch in den Musterdokumentgesprächen der DB Netz AG wurde diese Flexibilität mehrfach betont. Wie in Abschnitt 3.1.1 bemängelt wurde, ist der AIA ein fixes Dokument, welches vor Beginn der Modellerstellung abgeschlossen sein muss. Viele Probleme und auch Wünsche bezüglich LOI und LOG treten jedoch erst mit Fortschreiten eines Projektes auf, beispielsweise wenn sich weitere Nutzungsmöglichkeiten abzeichnen. Daher wäre es im Sinne der BIM-Methodik wünschenswert, die Anforderungen an die Fertigstellungsgrade der Modellkomponenten (LOD, 2.1.3) im BAP festzuhalten und in den AIA nur Mindestanforderung zu beschreiben.

Manche AGs wie z.B. die DEGES, legen diese flexible Struktur bereits vertragsrechtlich fest [70, 84]. Andere, darunter auch die DB Netz AG, betrachten den BAP noch nicht als Vertragsbestandteil und verzichten damit auf die beschriebenen Vorteile. Jedoch wurde das SOM in der Neustrukturierung (siehe 3.3.4) bereits auf eine derartige BAP-Interpretation ausgelegt.

3.2. PRÜFREGELN

Bevor Modellanforderungen erstellt und Modellprüfungen durchgeführt werden können, müssen zunächst zwei grundsätzliche Fragen behandelt werden:

1. Was sind Prüfregeleln?
2. Wie sind Prüfregeleln strukturiert?

Neben den Publikationen von Prof. Hjelseth [1, 85] sei hier auch auf die Untersuchung von Sowa [86] verwiesen. Er beschreibt in seiner Arbeit „The Challenge of Knowledge Soup“ ein Ausgangsproblem, welches auch der Modellprüfung zugrunde liegt.

Der Mensch verfügt über ein ausgesprochen breites Wissen. Dieses Wissen ist teils sehr intuitiv. Um dieses Wissen an Mitmenschen und auch Maschinen weitergeben zu können muss es geordnet und strukturiert werden.

Einen Ansatz zur Gliederung stellt der in Abbildung 8 dargestellte Baum von Porphyrios aus dem 3. Jahrhundert n. Chr. dar. Dieser kategorisiert bekannte Weltaspekte anhand einer Entscheidungsbaumstruktur.

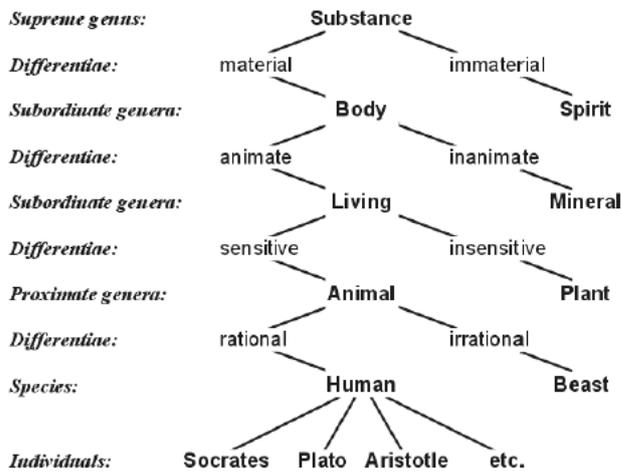


Abbildung 8: Baum des Porphyrios [86]

Dies stellt ein noch heute genutztes Verfahren dar, welches nicht nur zur Gliederung und Kategorisierung von Wissen dient, sondern wie in 2.1.4 diskutiert wurde, auch zur Strukturierung von Abläufen und Prozessen. Gleichzeitig basieren auch Vererbungsstrukturen auf derartigen Konzeptionen.

Methoden und Hilfsmittel, um Wissen zu strukturieren, sind ausgesprochen vielfältig. Beispielsweise werden dazu Bücher, Tabellenwerke und UML-Diagramme verwendet. Es wurden auch eigene Symbole und Begrifflichkeiten eingeführt, wie dies in den Wahrheitstabellen geschieht (siehe Abbildung 9). Ziel ist es dabei, logische Zusammenhänge aufzuzeigen und mit allgemeingültigen Regeln zu versehen, um dadurch Schlussfolgerungen und logische Abläufe zu modellieren.

A	B	$A \wedge B$	$A \vee B$	$A \dot{\vee} B$	$A \Leftrightarrow B$	$A \Rightarrow B$	$\neg A$
w	w	w	w	f	w	w	f
w	f	f	w	w	f	f	f
f	w	f	w	w	f	w	w
f	f	f	f	f	w	w	w

Abbildung 9: Wahrheitstabelle der Aussagenlogik [87]

Das Wissen im Bauwesen einschließlich geltender Regeln und Maßnahmen, wird in Tafelwerken wie den Schneider-Bautabellen sowie Normen und Richtlinien (siehe 1.4) festgehalten. Diese Regelwerke werden auch als Requirements, Regulations, Recommendations (RRR) bezeichnet. Sie sind für den Menschen lesbar und verständlich, jedoch nicht für Maschinen und Computer. Im Zuge der Industrie 4.0 und der zunehmenden Digitalisierung des Bauwesens liegt die Notwendigkeit klar auf der Hand, die RRR auch für Computer zugänglich zu machen.

Example Clause: (ICC IECC 2006 502.5 Moisture control)	All framed walls, floors and ceiling not ventilated to allow moisture to escape shall be			
	A	S	S	E
	provided with an approved vapour retarder having a permeance rating of 1 perm or ...			R

Legend: R – Requirement; A – Applicability; S – Selection; E – Exception; C – Check

(Adapted from: Hjelseth & Nisbet 2010)

Abbildung 11: Beispiel einer RASE-Analyse einer Anforderung nach [90]

S beschreibt alle potenziell möglichen Entitäten, A die notwendigen Bedingungen, E die Ausschlusskriterien und R alle Anforderungen an die Zielgruppe. Eine RASE-formatierte Regel besagt also, dass Objekte, welche sowohl der Applicability als auch dem Select gerecht werden und nicht einer Exception entsprechen, den Requirements genügen müssen. Dies führt, analog zu [89], zur nachfolgenden Bewertung von RASE-formatierten Regeln:

Fail: is-Applicable [A]	Pass: not is-Applicable [not(A)]
And is-Selected [S]	or not is-Selected [not(S)]
And not is-Exception [not(E)]	or is-Exception [E]
And not is-as-Required [not(R)]	or is-as-Required [R]

Bezogen auf die in 2.1.4 aufgeführten Arten der Modellprüfung ergeben sich die in Abbildung 12 dargestellten Konstellationen zwischen R und A, S, E. Es variiert vor allem die Anzahl möglicher Requirements.

Intention	Result	Rule set	Type of product *)	Generic example
Validating:	pass/fail	is related to used standard / code / regulation etc. or parts above	is related to criteria in rule set	Clause = R or not(A) or not(S) or E
Guiding:	options and advice			set of possible R given A, S, E
Adaptive:	a modified model			one R given A, S, E
Content:	a filtered norm			set of relevant R given A,S,E

*) can also be a materials, functions or other limitable subjects

R = Requirement, A = Applies, S = Select, E = Exception

Abbildung 12: RASE angewandt auf die verschiedenen Modellprüfungsansätze nach [74]

Beim Einsatz der RASE sieht man sich immer wieder mit dem Problem konfrontiert, dass RRR-Dokumente nicht vollständig RASE-konform sind. Manche sind nicht in einer einheitlichen logischen Struktur formuliert, in anderen sind die RRR unvollständig oder nur teilweise definiert. Für diese Situationen haben sich zwei Methoden als Hilfsmittel etabliert, mit denen RRR bearbeitet werden können, bevor die Umwandlung anhand der RASE erfolgt.

Die erste Methode wird „Translate, Transform, Transfer“ oder kurz Tx3 genannt [1, 88, 85]. Die Bezeichnung verweist auf die Vorgehensweise:

- Translate: Sofern die RRR bereits entsprechend einer Metrik, Regeln oder Logik verfasst sind, wird die RRR direkt in RASE übersetzt.

- Transform: Beschreibt die Umordnung, Erweiterung oder Ausbesserung unzureichender RRRs in eine RASE-kompatible Form, sofern dies möglich ist.
- Transfer: Sofern eine RRR sich nicht in einfacher Weise RASE-konform ausbessern lässt, muss die RRR von Fachpersonal neuformuliert werden, um eine weitere Verwendung zu ermöglichen.

Um verschiedene Regeln auf einheitliche metrische Bewertungsmaße zusammenzufassen können Regeln qualitativ und quantitativ betrachtet werden [85]. Hierfür bietet sich die Test-Indicator-Objectives-Methodik (TIO-Methodik) an. Ziel ist es, mehrere Regeln bezüglich ihrer Testindikationen zusammenzufassen (in der RASE werden diese zu A und S), wie dies in Abbildung 13 geschieht. Hier bestimmt die TIO Bewertungsmaße, welche betrachtet werden müssen, um die qualitativen Ziele einer Regeln zu überprüfen.

Clause	Shall/Should	Qualitative expression of goal <i>text of statement in ISO standard</i>	Test Indicator Objectives (TIO)	Quantitative metric =, <, >
<i>Minimum dimension</i>				
7.6	Shall	<i>..powered wheelchair.. If larger powered wheelchairs and scooters for outdoor use are to be considered, the outer radius of a turning space should be larger.</i>	Dimension of powered wheelchair, different types, in mm	x mm
<i>Pre-accept solution</i>				
18.3.2	Should	<i>..easy to use; open and close.. Windows should be easy to open and close. It should be possible to open and close the windows with only one hand.</i>	Pre-accepted (approved) type of window	Approved by x organization

Abbildung 13: Auszug eines TIO-bearbeiteten Regelsatzes [85]

Mit Hilfe dieser beiden Methoden können RRR derart angepasst werden, dass sie anhand der RASE-Methodik übersetzen werden können. Das Zusammenspiel der Methoden wird in Abbildung 14 veranschaulicht.

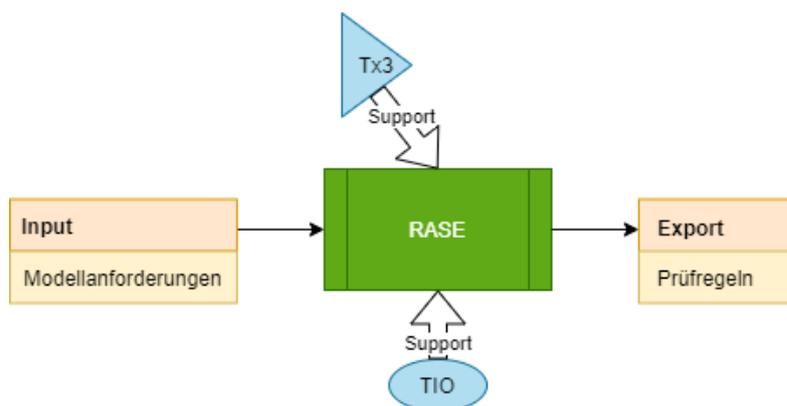


Abbildung 14: Zusammenspiel der Vorgehen für die RASE, vergleiche [1]

Untersuchungen zum Einsatz der RASE anhand mehrerer Normen zeigen, dass es sich um ein sehr effizientes Verfahren handelt [72].

IfcConstraint:

Gleichwohl gibt es noch weitere Vorgehensweisen zur Darstellung und Übersetzung von RRR in für BMC nutzbare Formate, darunter auch das Konzept **IfcConstraint**.

Das IfcConstraint-Modell wurde in IFC2x3 erstmalig eingeführt. Hierbei wird das IFC-Modell in seiner Struktur selbst anhand der RRR konzipiert und „beschränkt“. Über das Attribut *hasAssociations* kann jedem Kind von **IfcObjectDefinition** eine beliebige Anzahl von IfcConstraints zugewiesen werden.

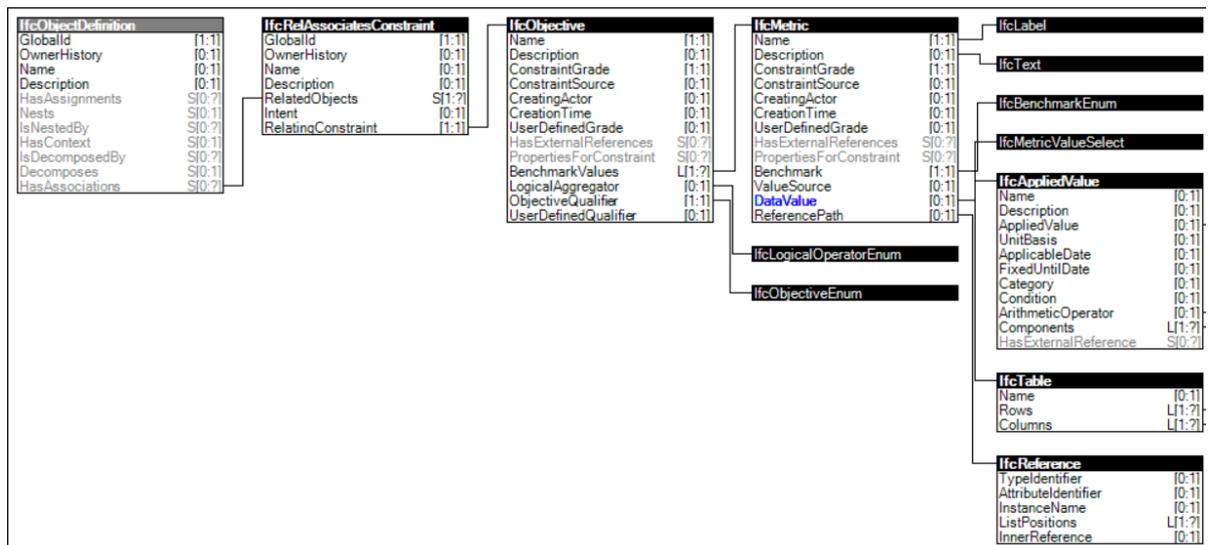


Abbildung 15: Auszug der Zuordnung eines IfcConstraint entsprechend buildingSMART [50, 53, 54]

Die Zuordnung erfolgt, wie in Abbildung 15 dargestellt, mittels der Objectified Relationship **IfcRelAssociatesConstraint** und erlaubt beispielsweise die Beschränkung der **IfcObjectDefinitions** auf gewisse Wertebereiche. **IfcConstraints** unterteilen sich in **IfcMetrics**, welche die quantitativen Beschränkungen festlegen und **IfcObjective** welche die qualitativen Einschränkungen darstellen.

Die Beschränkungen sind ferner mit einem Grad versehen, die in Abbildung 16 aufgeführt sind. Sie legen fest, welche Gewichtung die Beschränkungen haben, also ob sie erfüllt werden müssen, können oder sollen.

Constant	Description
HARD	Qualifies a constraint such that it must be followed rigidly within or at the values set.
SOFT	Qualifies a constraint such that it should be followed within or at the values set.
ADVISORY	Qualifies a constraint such that it is advised that it is followed within or at the values set.
USERDEFINED	A user-defined grade indicated by a separate attribute at the referencing entity.
NOTDEFINED	Grade has not been specified.

Abbildung 16: IfcConstraint Grade [50, 53, 54]

Ein Beispiel für die Umsetzung des IfcConstraint-Modells in IFC-Step ist in Auszügen in [72] zu finden. Hierbei wurde der Korean Building Act 34-1 in ein IfcConstraint-Modell überführt.

Formate zur Softwarenutzung von Prüfredeln sind vielfältig und im Nachfolgenden werden nur einige wenige aufgeführt, welche von buildingSMART explizit auf ihre Tauglichkeit untersucht wurden [72].

Das erste Format zur Umsetzung der Prüfredeln ist mvdXML wie es bereits in 1.5.2 beschrieben wurde. Es ist sowohl RASE-konform als auch problemlos mit IfcConstraint-Modellen kompatibel, da es ein für das IFC-Format erarbeitetes Format ist.

BuildingSMART führt unter anderem die SWRL, Java DROOLS, BIMRL als Regelprüfungsformate auf, mittels welcher die Prüfredeln implementiert werden können [72].

Unter *Procedural Code Generation* [72] beschreibt die buildingSMART die Programmierung der Prüfredeln in verschiedenen Programmiersprachen wie C++, Java, javascript, Donet, etc. Ein Beispiel ist mit Desite gegeben, welches die Prüfredeln in javascript nutzt.

Zusätzlich kann natürlich auch eine Regeltabelle angewandt werden, ein Format, welches sich wiederum schnell in weitere Formate übersetzen lässt.

Zur Prüfung von rechtlich relevanten Informationen wird durch buildingSMART explizit auf die Formate LegalDocML [91] and LegalRuleML [92] verwiesen. Hierbei handelt es sich um offene ehrenamtlich erstellte Standards für Gesetzes- und Rechtsdokumente auf Basis des XML-Formats.

3.3. DAS SEMANTISCHE OBJEKTMODELL DER DB NETZ AG

3.3.1. ERLÄUTERUNG UND ZIELSETZUNG

Wie bereits in 2.1.3 beschrieben verfügen Objekte über geometrische und nichtgeometrische Eigenschaften. Unabhängig davon, ob man auf die Begrifflichkeiten des LOD, LOG und LOI zurückgreifen will, um die Fertigstellungsgrade von Objekten zu beschreiben, ist es notwendig sich zu den mit ihnen beschriebenen Eigenschaften von BIM-Objekten auseinanderzusetzen. Als Ergebnis davon sind Anforderungen und Regeln zu formulieren, welche Charakteristika die Objekte aufweisen sollen und welche Informationen für eine hinreichende Beschreibung benötigt werden. Auch muss sich in der Modellierung der BIM-Objekte widerspiegeln, ob und wie die verschiedenen Modellkomponenten untereinander verknüpft sind. Beispielsweise besteht ein Widerlager aus einem Widerlagerfundament, einer Widerlagerwand und Flügelwänden und eine Signalbefestigung verbindet ein Signal mit einem bestimmten Objekt.

Des Weiteren stellt sich die Frage welche Modellkomponenten in welchem Fachmodell/ Gewerk vorhanden sein müssen und welche Aspekte eines Objektes in den entsprechenden Fachmodellen von Relevanz sind.

Das **Semantische Objektmodell** (SOM) der DB Netz AG stellt einen ersten Entwurf dar, die nichtgeometrischen Aspekte (LOI) von Modellkomponenten aller DB-Gewerke bezüglich der oben genannten Kriterien zu beschreiben. Das SOM stellt die Gewerke in einer strukturierten Form da und gliedert sie in verschiedene Entitäten inklusive der zugehörigen Relationen. Darüber hinaus liefert das SOM Vorgaben für die Attribuierung und für die Zuordnung der Anforderungen zu den Leistungsphasen. Diese Anforderungen können, beispielsweise über BIMQ, in Prüfregele zum BMC übersetzt werden. Es ist vorgesehen, dass das SOM Teil der AIA und des BAP wird.

Das SOM basiert auf dem Konzept einer gewerkeübergreifenden Vererbungskette, wobei die Struktur der konventionell genutzten Gewerke beibehalten wird. Entsprechend betrachtet das SOM die elf gängigen DB-Gewerke plus die gewerkeübergreifende Kategorie BIM.

Das durch die DB Netz AG erstellte SOM liegt in Form zweier Dokumente vor:

1. Ein Enterprise Architect UML Diagramm, in dem die beteiligten Klassen mit ihren wesentlichen Attributen und ihren Beziehungen untereinander dargestellt sind.
2. Eine noch unvollständige Excel-Datei, die die Klassen detailliert beschreiben soll.

Die im Nachfolgenden erstellten Modellanforderungen basieren auf der Version 1.0 dieses SOM. Dieses ist der erste Versuch einer allgemeinen Grundlage für die Modellanforderungen der Projekte der DB Netz AG. Daher gibt es hier noch einige Aspekte, welche verbessert werden können. Diese werden konsequent gesucht und das SOM entsprechend überarbeitet.

3.3.2. AUSWAHL DER GEWERKE

Das SOM bildete die Ausgangsbasis für die in dieser Arbeit formulierten Modellanforderungen. Diese sollen in zukünftigen Projekten genutzt werden, wobei zunächst der Themenkomplex der Leistungsphase 1 im Fokus steht.

In weiterführenden Schritten sollen die Anforderungen dann in Kooperation mit konkreten Projekten anhand des BAP weiterentwickelt werden, um schließlich vollständige Basiskonzepte für eine aktualisierte Version des SOM bereitzustellen. In einem ersten Schritt wurden mehrere, im Rahmen dieser Arbeit zu überarbeitende, Gewerke ausgewählt:

- Leit- und Sicherungstechnik / Stellwerkstechnik (LST)
- Oberleitungsanlagen (OLA)
- Fahrbahn (FB)
- Brücken (BR)

Motivation für diese Wahl war zum einen das breite Wissen der Mitarbeiter der DB Netz AG zu den genannten Gewerken, zum anderen die in der Projektarbeit [69] gewonnenen Kenntnisse über Brückenmodelle.

3.3.3. HERAUSFORDERUNGEN

Obwohl das SOM wie oben beschrieben in vielerlei Hinsicht sehr gute Ansätze enthält, traten bei der konkreten Verwendung auch Probleme auf. Eine erste Schwierigkeit lag in der in Abschnitt 3.3.1 genannten Unvollständigkeit der Excel-Datei, insbesondere das Fehlen der Basisklassen **Bauteil** und **Einrichtung**. Ohne diese Informationen ist es nicht möglich die Vererbungsstrukturen vollständig und schlüssig nachzuvollziehen.

Weiterhin waren Attribute nicht einheitlich als „vererbt“ oder „individuell“ gekennzeichnet und die Vererbungsstruktur nicht stringent entwickelt. So ist beispielsweise nur die Klasse **Flachgründung** als eine Kindklasse der Klasse **Gründung** aufgeführt, nicht aber, wie zu erwarten wäre, die Klasse **Tiefgründung**. Auch ist keine der beiden Entitäten als Kindklasse von **Bauteil** definiert.

Ähnliche Herausforderungen, insbesondere die Inkonsistenz der Vererbungsstrukturen, waren auch in den zugehörigen UML-Diagrammen erkennbar. Hier wurde deutlich, dass die Vererbungsstruktur einfacher war als in der Excel-Tabelle suggeriert wurde und dafür sehr viele Aspekte des Modells auf Aggregationen basieren. Die Aggregationsstruktur eines Modells beschreibt hierbei die Struktur des Modells in mit Bestandteilen. Sie fängt beispielweise an mit der Unterteilung einer Brücke in Oberbau und Unterbau und reicht bis hin zu den verschiedenen Bestandteilen eines Widerlagers oder Brückenlagers. Das Ober- und Unterbau Bestandteile einer Brücke sind und dass diese sich aus ihnen zusammensetzt beschreibt eine Aggregation.

Auch war die Nomenklatur der Klassen nur eingeschränkt hilfreich. Sie war nicht an einheitlichen Regeln ausgerichtet, so dass nicht anhand des Namens einer Klasse erkennbar war, ob es sich um eine Aggregation oder eine Vererbung handelt.

Auffällig war, dass gewerkeübergreifende Umsetzungen in der Modellierung nur wenig Berücksichtigung fanden, obwohl das Konzept der gewerkeübergreifenden Vererbungsketten ein Grundpfeiler des SOM darstellt. Es gab beispielsweise eine Konzeption für **Gleis** in OLA und eine in FB. Diese überschneiden sich teilweise in den Attribuierungen und Aggregationsstrukturen. Eine Dopplung von gleichartigen Objekten in unterschiedlicher Ausführung führt in der Regel zu Redundanz und Problemen. Eine derartige Modellierung, die beispielsweise zu mehreren konkreten Objekten **Gleis** führt, welche ein und dasselbe Gleis darstellen, sollte grundsätzlich vermieden werden.

Darüber hinaus waren die Attribute für die Anforderungen an die Leistungsphase 1 zu sehr an kaufmännischen Gesichtspunkten ausgerichtet. Die Zuordnung der Attribute zu PropertySets erfolgte auch eher vorläufig und nicht unter thematischen Gesichtspunkten.

Schließlich wies das Gewerk BR die Einschränkung auf, dass es nur für eine Rahmenbrücke verwendet werden konnte. Es war am Modell einer Brücke dieses Typs erstellt und anschließend nicht erweitert worden. Als Konsequenz waren grundlegende Komponenten wie Widerlager oder Stützen nicht vorhanden.

Entsprechend konnten im Rahmen dieser Arbeit eine Vielzahl von Korrekturen und Anpassungen unterschiedlichen Umfangs durchgeführt werden. Diese Anpassungen werden im nachfolgenden Kapitel detailliert beschrieben.

3.3.4. NEUSTRUKTURIERUNG DES SOM

Ziel der Neustrukturierung war es die zuvor beschriebenen Probleme zu lösen und das SOM auf einen in BIMQ modellierbaren Anforderungskatalog für die LPH1 zu reduzieren.

Das neustrukturierte SOM kann den Tabellen in Auszügen unter den Anhängen „Semantisches Objektmodell Basisklassen“, „Semantisches Objektmodell Bruecke“, „Semantisches Objektmodell Fahrbahn“ und „Semantisches Objektmodell Schnellübersicht“ eingesehen werden.

In einem ersten Schritt wurden die Properties nach PropertySets und Häufigkeiten sortiert und anschließend die Vererbungsstrukturen anhand der UML-Diagramme in Übereinstimmung mit der Excel-Datei gebracht.

Hierbei wurde der Umfang mancher PropertySets deutlich. Beispielsweise hatte das PropertySet **DB_AllgemeineDaten** über 240 Attribute, von denen über 100 aber jeweils nur in einer einzigen Klasse genutzt wurden. Daher mussten die PropertySets neu strukturiert werden. Ein weiterer Anlass zur Änderung bestand darin, dass BIMQ Vererbungsstrukturen nicht direkt darstellen kann. Um die Vererbungsstruktur beibehalten zu können, wurden dieser Aspekt mit der Neustrukturierung der PropertySets zusammengeführt.

Die PropertySets werden als Darstellung der Vererbungsstruktur genutzt. Das heißt, die nicht an eine Klasse vererbten Attribute werden in einem PropertySet mit dem entsprechenden Namen zusammengefasst. Einer Klasse werden anschließend alle PropertySets der Vererbungskette zugewiesen.

Zur Veranschaulichung ein kurzes Beispiel:

Die Klasse **Bohrpfahl** (*bauteilKlassifikation: br.gr.130*) des Gewerkes BR ist Kindklasse von **Bauteil**. Dieses wiederum ist Kind der später noch erläuterten SOM-Wurzelklasse **Allgemeine_Eigenschaften**.

Entsprechend werden der Klasse **Bohrpfahl** in den BIMQ-Anforderungen die PropertySets

- **Allgemeine_Eigenschaften**
- **Bauteil**
- **BP** (Kürzel für Bohrpfahl)

zugeordnet. Diese Zuordnung, wie sie in BIMQ erfolgt, ist in Abbildung 17 dargestellt.



Abbildung 17: PropertySet-Zuweisung in BIMQ

Dadurch sind die Properties zwar noch nicht thematisch sortiert, aber die Vererbungsstruktur kann abgebildet und dadurch das SOM klar strukturiert werden. Die Informationen über die Vererbungskette jedes Elements des SOM wird in den Beschreibungen in BIMQ (Abbildung 18) oder in den Exceltabellen mitgegeben.

Bohrpfahl BR.gr.130 Elternklasse: B Vererbungskette: B, AE Element IfcBuildingElementProxy,##,##.Allgemeine_Eigenschaften.bauteilKlassifikation:"br.gr.130"

Abbildung 18: Bohrpfahlelement in BIMQ

Zur Vereinfachung wurden die überarbeiteten Gewerke in einer auf die Vererbung und Aggregationen reduzierten Tabelle dargestellt (siehe Abbildung 19).

bauteilName	Kürzel	bauteilKlassifikation	Elterklasse	Vererbungskette	Bestandteile
Bruecke	BR	br	E	E, AE	BET(FB), UE, UB, GR, ASR
Ueberbau	UE	br.ue	AE	AE	BTW, KA
Brueckentragwerk	BTW	br.ue.10	AE	AE	BRP, BRT
Brueckenplatte	BRP	br.ue.11	B	B,AE	-
Rahmendecke	RD	br.ue11.100	BRP	BRP, B, AE	-
Brückenträger	BRT	br.ue12	B	B, AE	-
Kappe	KA	br.ue20	B	B, AE	KT, RK
Kapbeltrog	KT	br.ue21	B	B, AE	-
Randkappe	RK	br.ue22	B	B, AE	-

Abbildung 19: Auszug der Schnellübersicht des SOM, Kürzel: AE - Allgemeine_Eigenschaften, B - Bauteil und E - Einrichtung

Die Attributlisten wurden in Abstimmung mit Fachplanern auf die Anforderungen der LPH1 reduziert und gleichzeitig um einige fehlende Attribute erweitert. In diesem Zusammenhang wurden die Attribute

- *bauteilKlassifikation (objectType)*
- *bauteilName*
- *identitaet (GlobalId)*
- *name (Name)*
- *bemerkung (Description)*
- *KGK-Verzeichnisnummer*

in dem PropertySet **Allgemeine_Eigenschaften** zusammengefasst. In den Klammern stehen jeweils Vorschläge für mögliche Ifc-Attributpendants. Dadurch werden Vorschläge für ein direktes Mapping geboten. Dieses PropertySet wird als Pendant einer Wurzelklasse des SOM eingeführt und jeder Klasse zugewiesen. Dass dieses PropertySet größtenteils aus Pendants der IfcObject-Attribute besteht, ist dem Ziel geschuldet, eine potenzielle Anpassung der Attribuierung an den IFC-Standard möglichst einfach zu gestalten.

Die ersten beiden Attribute, *bauteilKlassifikation* und *bauteilName*, dienen zur Klassifizierung des Objekttyps, *identitaet* und *name* werden zur individuellen Benennung der Objekte verwendet. Das Attribut *bemerkung* erlaubt es Anmerkungen und Erläuterungen zu hinterlegen. *KGK-Verzeichnisnummer* dient zur Einordnung der Entitäten in den Kostenkatalogen der DB-Netz AG.

Mindestens eines der Klassifikationsattribute *bauteilKlassifikation* und *bauteilName* ist notwendig, da es sich bei den meisten Klassen um Infrastrukturkomponenten bzw. technische Komponenten handelt, die größtenteils nur als **ifcBuildingElementProxy** dargestellt werden können.

Die im SOM vorgesehene Klasse **Bauteilgruppe** wird in diesem Zusammenhang aufgelöst, da die relevanten Attribute dieser Klasse bereits in **Allgemeine_Eigenschaften** enthalten sind.

Bereits im Abschnitt 3.3.3 wurden die Vererbungsketten und die Dopplung von Klassen als Probleme des SOM identifiziert. Ursache für die bestehende SOM-Modellierung war, dass die Gewerke als unabhängige Fachmodelle konzipiert wurden. Dies wurde jetzt überholt. Die Gewerke selbst dienen nicht mehr als unabhängige Fachmodelle, sondern in einem ersten Schritt zur Einordnung der Klassen bezüglich ihrer Aggregationen untereinander. In einem zweiten Schritt erfolgt dann die Zuordnung der Anforderungen zu konkreten Fachmodellen, die sich, wie beispielsweise in 4.2 zu sehen sein wird, aus Klassen mehrerer Gewerke zusammensetzt.

Zur Vereinheitlichung wurden die Exceltabellen um die Informationen der UML-Diagramme erweitert. Anschließend wurden alle Klassen auf die Sinnhaftigkeit ihrer Vererbungsstrukturen überprüft und Unvollständigkeiten beseitigt. Zusätzlich wurden die Klassen um die fälschlicherweise als „individuell“ zugewiesenen Attribute (meist *material*, *materialEigenschaften*, *lieferant* und *inspektionsintervall*) bereinigt. Anschließend wurden die Gewerke aus einer übergeordneten Perspektive betrachtet. Dadurch konnten mehrfach auftretende Klassen wie Gleis oder Gleisabschluss identifiziert werden. Diese wurden bezüglich der Aggregationen und Nomenklatur einem einzigen Gewerk zugeordnet. Ihre Attribute und Vererbungsketten wurden zusammengeführt. In diesem Zusammenhang wurden auch Klassen bezüglich ihrer Aggregation in den Gewerken verschoben (Beispiel: **Gleisreferenzpunkt** wird von LST zu FB verschoben).

Die Gewerke LST, OLA und FB wurden, wie bereits beschrieben, verschlankt und umsortiert. Zusätzlich wurde beim Gewerk LST die Klasse **Ueberwachungssignal** ergänzt. Eine Erweiterung des Gewerkes BR um zusätzliche Klassen war notwendig, da es, wie bereits erwähnt, zu Beginn lediglich eine Rahmenbrücke beschrieb, auch wenn über Attribute wie *brueckeArt* die Grundlage für eine Verallgemeinerung auf andere Brückentypen bereits angedeutet war.

Zunächst wurde dazu die Grundklasse **Rahmenbruecke** in die Grundklasse **Bruecke** umgewandelt. Dann wurden die Teilbereiche ergänzt:

Brückentragwerk als Teil von **Ueberbau** wurde, wie in Abbildung 20 dargestellt ist, um die Klassen **Brueckenplatte** und **Brueckentraeger** ergänzt und die **Rahmendeckenplatte** als Spezialisierung (also eine Kindklasse) der **Brueckenplatte** untergeordnet.

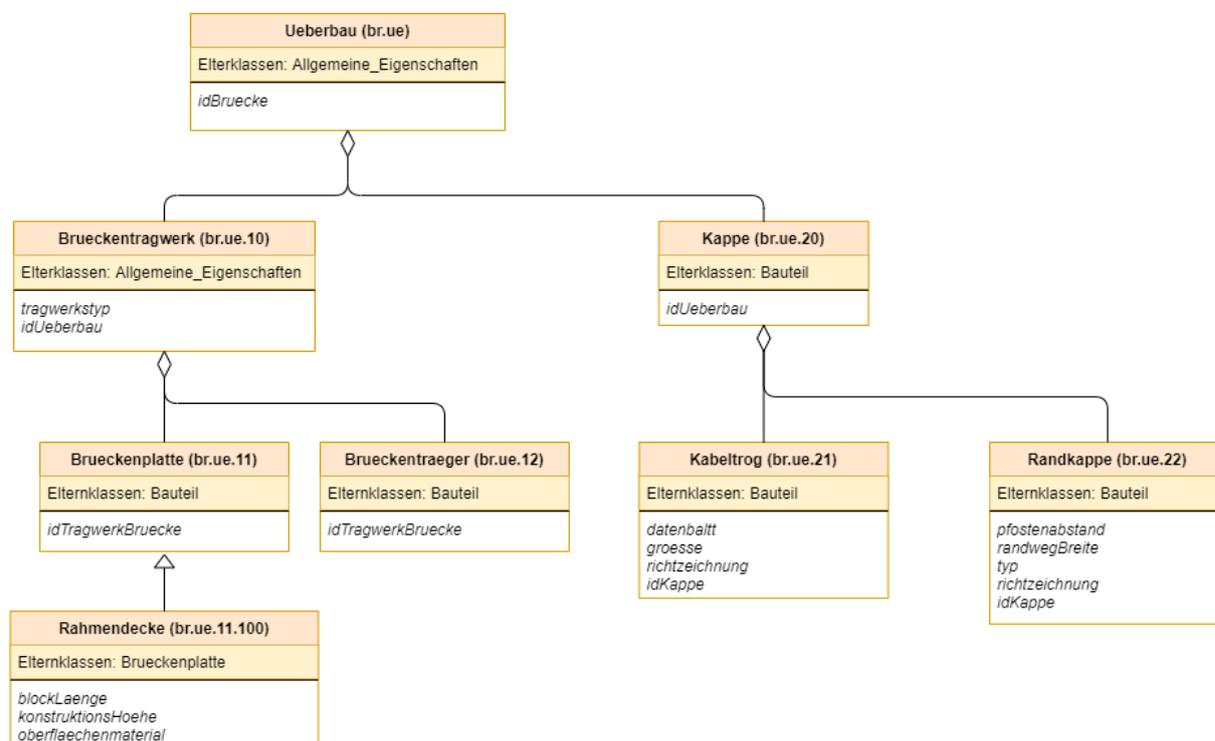


Abbildung 20: UML-Diagramm des Brueckenteilbereichs Ueberbau, eine Raute beschreibt hierbei eine Aggregation, ein Dreieck eine Vererbung

Der **Unterbau** wurde in vergleichbarer Weise neu gegliedert. Bis dahin war er für die Rahmenbrücke auf die Teilkomponenten Rahmenwand und Rahmenflügel begrenzt.

Als Erstes wurde hier die Klasse **Widerlager** mit ihren vier Bestandteilen **Widerlagerwand**, **Fluegelwand**, **Widerlagerfundament** und **Widerlagerhinterfuellung** modelliert. Die Optionen auf die Rahmenbrückenkomponenten wurden in den entsprechenden Typ-Attributen in **Widerlagerwand** und **Fluegelwand** hinterlegt. Zusätzlich wurden **Stuetzen** als Teil des Unterbaus eingeführt. Eine Übersicht hierzu bietet die Abbildung 21.

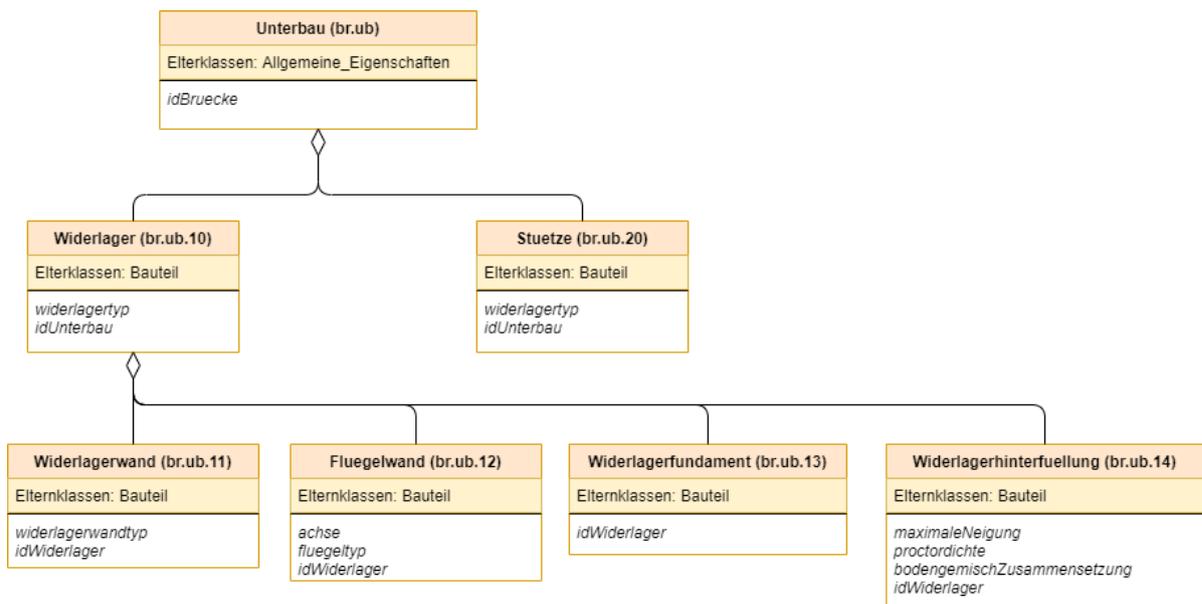


Abbildung 21: UML-Diagramm des Brueckenteilbereich Unterbau

Die weiteren Teilbereiche wurden im nächsten Gesamtschritt mit allen Gewerken angepasst. Die Vererbungsketten wurden überprüft und vereinheitlicht. So erbte bisher die **Tiefgruendung** von der **Gruendung**, die **Flachgruendung** dagegen nicht. Die **Kappe** war kein **Bauteil** und die **Mastkonsole**, als eine **Gruendung** eines anderen Gewerkes, verfügte unabhängig von dieser über die zugehörigen Attribute. Derartige Vererbungsinkonsistenzen wurden nach der Ergänzung des SOM um die entsprechenden Klassen bereinigt. In diesem Zusammenhang wurden nochmals entsprechende Attributdopplungen beseitigt und einige Attribute, wie beispielsweise die *vmax* (Maximalgeschwindigkeit), verallgemeinert.

Da das SOM in BIMQ implementiert werden sollte, musste die Struktur dementsprechend konzipiert werden. Das SOM legt einen Schwerpunkt auf Aggregationen. In IFC würde dies über die Relation **IfcRelAggregates** erfolgen. BIMQ kann jedoch keine Relationen „darstellen“. Entsprechend musste die Aggregation auf andere Art umgesetzt werden. Die Überlegungen ergaben, die Aggregation von den unteren Hierarchieebenen aus aufzurollen und jeder Klasse die *id* der Klasse der nächsthöheren Hierarchieebene zuzuweisen. Auf diese Weise können die Komponenten stets eingeordnet werden. Ein Beispiel kann Abbildung 21 entnommen werden. Es wird vorgegeben, dass alle Teilkomponenten eines **Widerlagers** die *idWiderlager* des aus ihnen zusammengesetzten **Widerlagers** zugewiesen bekommen haben. Für das **Widerlager** und die **Stuetze** gilt, dass über die *idUnterbau* die Information vorhanden ist, welchem **Unterbau** sie zugeordnet sind und Analoges besagt die *idBruecke* für den **Unterbau**.

In einem letzten Schritt wurde ein einheitliches Konzept für die Nomenklatur der Klassen erarbeitet. Die Nomenklatur bezeichnet die Werte des Attributes *bauteilKlassifikation*. Bis

dahin gab es dafür keine stringente Regel, so dass die Nomenklatur selbst innerhalb eines Gewerkes variierte.

Das Resultat sieht wie folgt aus:

Der Anfang der Nomenklatur gibt das Gewerk an, in welchem die Objekte eingegliedert sind:

- br: Brücke
- ola: Oberleitungsanlage
- Ist: Leit- und Sicherungstechnik / Stellwerkstechnik
- ob: Fahrbahn/Oberbau

Es folgt das Kürzel des Teilbereichs. Die daran anschließenden Hunderter Zahlen bedeuten Vererbung, 10er und 1er Zahlen Aggregation.

Eine Beispielkette:

Der **Ueberbau** ist definiert als „br.ue“, „ue“ ist das Überbaukürzel, und besteht aus **Tragwerk_Bruecke** „br.ue.10“. Das Tragwerk besteht aus den Klassen **Brueckenplatte** „br.ue.11“ und **Brueckentraeger** „br.ue.12“. Die **Brueckenplatte** kann als Rahmendecke in Form ihrer Kindklasse **Rahmendecke** „br.ue.11.100“ usw. auftreten.

Ursprünglich war geplant die Klassen weitestgehend an den IFC-Standard anzupassen. Im IFC 4.3 wurden bereits die Bereiche IfcBridge und IfcRail eingeführt. Da BIMQ nur bis IFC 4Add2 modellieren kann hätte hier lediglich die Attribuierung der Klassen angepasst werden können. Darauf wurde allerdings verzichtet da die meisten IFC Entitäten in ihrer Grundstruktur hauptsächlich indirekte Attribute aufweisen (Das Beispiel IfcRail ist in Abbildung 22 gegeben), also aus Relationen entstehende Attribute [50], welche BIMQ wie bereits beschrieben darstellen kann.

Attribute inheritance

#	Attribute	Type
<i>IfcRoot</i>		
1	GlobalId	IfcGloballyUniqueId
2	OwnerHistory	IfcOwnerHistory
3	Name	IfcLabel
4	Description	IfcText
<i>IfcObjectDefinition</i>		
	<i>HasAssignments</i>	IfcRelAssigns @RelatedObjects
	<i>Nests</i>	IfcRelNests @RelatedObjects
	<i>IsNestedBy</i>	IfcRelNests @RelatingObject
	<i>HasContext</i>	IfcRelDeclares @RelatedDefinitions
	<i>IsDecomposedBy</i>	IfcRelAggregates @RelatingObject
	<i>Decomposes</i>	IfcRelAggregates @RelatedObjects
	<i>HasAssociations</i>	IfcRelAssociates @RelatedObjects

Abbildung 22: Auszug der Attribute von IfcRail [50]

Die anderen Attribute sind weitestgehend vergleichbar mit den Attributen der Klasse **Allgemeine_Eigenschaften**. Beispielsweise entspricht *identitaet* einer *GlobalID*. Die individuellen Attribute der im IFC 4.3 eingeführten Entitäten sind *PredefinedType* womit ein passender Typus aber ohne wirkliche Attribuierung zugewiesen wird. Die bedarfsgerechte Attribuierung erfolgt entsprechend des IFC-Konzepts erst über PropertySet-Zuweisungen. Eine Anpassung ergibt hier also wenig Sinn, da wir den Proxylementen ohnehin individuelle PropertySets zuweisen.

Anmerkung: Um einen Vergleich zum IFC dennoch nicht gänzlich auszuschließen, wurden den Entitäten im SOM wie es in den Anhängen „Semantisches Objektmodell Basisklassen“, „Semantisches Objektmodell Bruecke“ und „Semantisches Objektmodell Fahrbahn“ dargestellt ist, noch nach IFC4.3 passende IFC-Entitäten zugewiesen. Das vollständige neustrukturierte SOM kann im digitalen Anhang eingesehen werden.

3.3.5. DIE BIMQ-IMPLEMENTIERUNG

Die Implementierung des SOM in BIMQ erfolgte entsprechend der BIMQ-Strukturen. Die Attribute werden gemäß des SOM in „Gruppen“ gegliedert, welche die PropertySets repräsentieren. Die Festlegung der Datentypen wurde in der Verfeinerung, welche BIMQ anbietet, in Rücksprache mit Fachplanern, getroffen. An dieser Stelle muss betont

werden, dass das Attribut *material* in BIMQ mit dem Datentyp „Text“ belegt wurde. Im IFC-Standard kann die Zuweisung eines oder mehrerer Materialien zu einem Objekt auf mehreren Wegen erfolgen, wie beispielsweise über eine Typzuweisung oder durch die Relation ***IfcRelAssociatesMaterial*** (5.2). In allen Fällen findet die Zuweisung über eine Relation statt. Materialien sind in IFC als eigene Objekte inklusive Eigenschaften abgebildet und nicht nur als einfacher Text. BIMQ kann Relationen nur eingeschränkt wiedergeben, da die meisten damit kompatiblen Model Checker Relationen in ihren proprietären Datenformaten nicht exakt wiedergeben. Wie später auch noch im Falle der Aggregationen muss hier eine pragmatische Lösung angewendet werden, um den Softwarebeschränkungen Rechnung zu tragen. Deshalb fiel die Wahl auf den Datentyp Text.

In vergleichbarer Weise wie die Attribute, wurden die Elemente, welche die Klassen repräsentieren, entsprechend der Gewerkstrukturen in Gruppen und Untergruppen gegliedert angelegt. Das IFC-Mapping spielt hierbei eine wichtige Rolle. Wie bereits erwähnt (3.3.4), bietet der IFC-Standard bis zur genutzten Version IFC4Add2 kaum geeignete Entitäten [54] für infrastrukturenspezifische Anwendungen. Es werden daher hauptsächlich ***IfcBuildingElementProxy***s genutzt. Diese müssen näher spezifiziert werden, auch damit bei den späteren Prüfungen klar ist, welche Anforderungen an die jeweiligen Objekte gestellt werden. Es ist nicht das Ziel, dass sämtliche ***IfcBuildingElementProxys*** auf sämtliche Regeln hin geprüft werden. Hier kann nun in BIMQ eine Regel anhand beliebiger Properties aufgestellt werden. In diesem Fall fiel die Wahl auf die *bauteilKlassifikation*.

Für jedes Element wird definiert, dass es sich um ein ***IfcBuildingElementProxy*** handelt mit dem PropertySet **Allgemeine_Eigenschaften** mit dem Wert der entsprechenden Nomenklatur (siehe 3.3.4). Ein Beispiel hierfür ist in Abbildung 18 zu sehen. Dort wird ein ***Bohrpfahl*** als ein ***IfcBuildingElementProxy*** mit dem Propertyset **Allgemeine_Eigenschaften** und dem Attributwert „br.gr.130“ für das Attribut *bauteilKlassifikation* definiert.

Für die Realisierung der Fachmodelle boten sich verschiedene Möglichkeiten an. Eine naheliegende Option bestand darin, das SOM als Ganzes in ein einziges Fachmodell einzuspeisen und dann anwendungsfallspezifischen Leistungsbildern zuzuordnen. Diese Vorgehensweise ermöglicht eine gebündelte Zuweisung, bei der auch vergessene Attribute und Objekte schnell ergänzt werden können. Allerdings wird ein derartiges Fachmodell mit zunehmendem Umfang schnell unübersichtlich. Alternativ könnte für jeden Anforderungskatalog ein eigenes Fachmodell erstellt werden. Diese Methode schafft eine schnelle Übersicht, bringt aber den Nachteil mit sich, dass Änderungen etwas komplizierter werden, insbesondere wenn eigentlich Aspekte mehrerer Fachmodelle getestet werden müssten. Als Resultat dieser Überlegungen wurden beide Konzepte in Teilen genutzt. Für alle bearbeiteten Gewerke wurden jeweils eigene Fachmodelle erstellt, in denen die Klassen in ihrer Attribuierung und als Konsequenz auch in ihrer Vererbungsstruktur dargestellt wurden. Sie dienen hier in erster Linie als Basis zur Weiterführung des SOM in zukünftigen Projekten sowie zur Erstellung von Objektkarten wie sie in Abbildung 23 für ein

Brueckentragwerk zu sehen ist. Diese finden vor allem zur Beschreibung der Objekte für Auftragnehmer Verwendung. Die Objektkarten fassen die Informationen der Modellkomponenten zusammen. Es werden neben den Kommentaren die benötigten PropertySets und die zugehörigen Attribute mit Datentypen, diskreten Werten und ihren Mappings dargestellt.

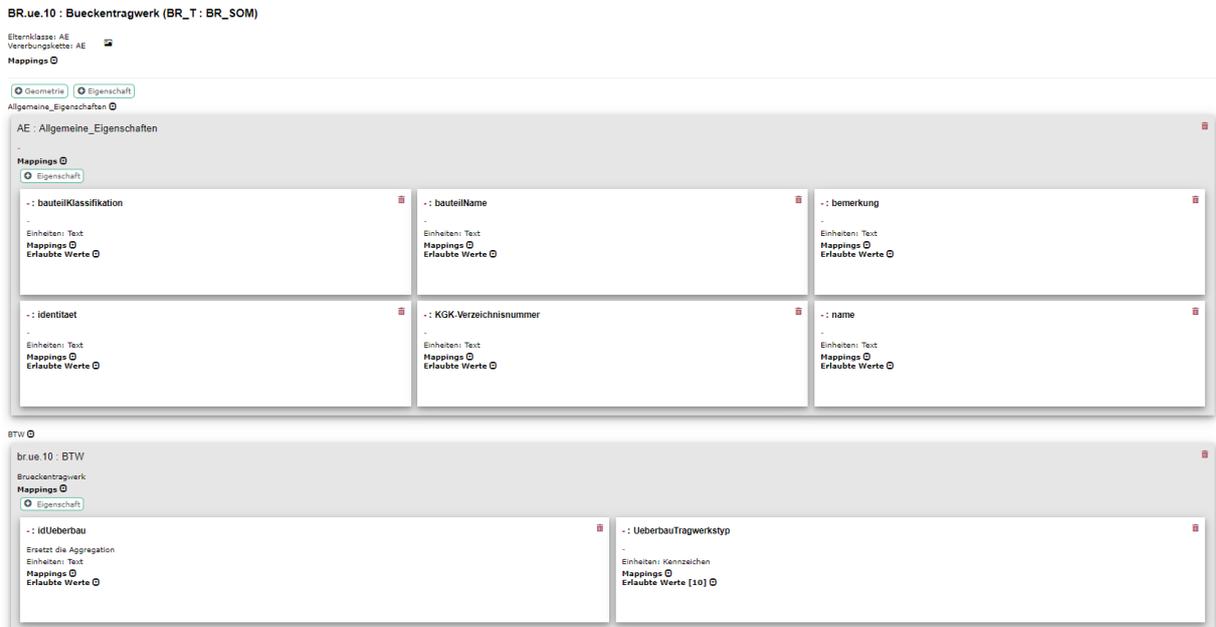


Abbildung 23: Objektkarte eines Brueckentragwerk in BR Gewerk

Um die erstellten Anforderungen zu testen und darzustellen hätte ein großes Modell, welches alle erarbeiteten Modellkomponenten enthalten hätte, genutzt werden können und die Prüfregele in mehreren Model Checkern ausgeführt werden können. Die zugehörigen Resultate wären allerdings umfangreich und unübersichtlich und maßgebende Aspekte würden nicht deutlich werden. Daher werden in Kapitel 4 nicht sämtliche Anforderungen an Modellen geprüft, sondern nur eine gezielte Auswahl aus den Gewerken Brücke und Fahrbahn/Oberbau. Dadurch bleibt der Überblick gewahrt und wichtige Aspekte treten deutlich hervor.

In Abbildung 24 werden die ausgewählten Gewerke mit ihren Teilbereichen auszugsweise dargestellt. Für jedes Test-IFC-Modell wurde ein eigener Anwendungsfall definiert und der LPH 1 zugewiesen (Lph-1 BRM_1/2/3). Diesen werden die Anforderungen zugewiesen. Dies erfolgt, indem die Properties der Modellkomponenten über die Haken als für den Anwendungsfall gefordert definiert werden.

Brueckenmodell_1 (SOM_Testlaufplaner)	Code	Beschreibung	Typ	IFC 4 Add2	Lph-1-BRM_1	Lph-1-BRM_2	Lph-1-BRM_3
Bruecken	-	-	Gruppe	-	☐	☐	☐
Ausruestung	-	-	Gruppe	-			☐
Gruendung	-	-	Gruppe	-			☐
Ueberbau	-	-	Gruppe	-	☐	☐	☐
Ueberbau	BR.ue	Elternklasse: AE Vererbungskette: AE	Element	IfcBuildingElementProxy.*	☐		
Brueckenplatte	BR.ue.11	Elternklasse: B Vererbungskette: B, AE	Element	IfcBuildingElementProxy.*		☐	☐
Allgemeine_Eigenschaften	AE	-	Gruppe	Allgemeine_Eigenschaften		☐	☐
BRP	br.ue.11	Brueckenplatte	Gruppe	BRP		☐	☐
idTragwerkBruecke	-	Ersetzt die Aggregation	Eigenschaft	#.idTragwerkBruecke	-	☑	☑
Bauteil	BT	-	Gruppe	Bauteil		☐	☐
Brueckentraeger	BR.ue.12	Elternklasse: B Vererbungskette: B, AE	Element	IfcBuildingElementProxy.*		☐	☐
Kappe	BR.ue.20	Elternklasse: B Vererbungskette: B, AE	Element	IfcBuildingElementProxy.*		☐	☐
Unterbau	-	-	Gruppe	-	☐	☐	☐
Fahrbahn/Oberbau	-	-	Gruppe	-		☐	☐
FesteFahrbahn	Ob.bet.100	Elternklasse: BET, B Vererbungskette: BET, AE, B	Element	IfcBuildingElementProxy.*		☐	☐
Tragschicht	Ob.bet.140	Elternklasse: B Vererbungskette: B, AE	Element	IfcBuildingElementProxy.*			☐
Seitenfuehrungsblock	Ob.bet.170	Elternklasse: B Vererbungskette: B, AE	Element	IfcBuildingElementProxy.*			☐

Abbildung 24: Auszug des BIMQ-Anforderungskataloges an die Testmodelle

Der spätere Export der Modellanforderungen in Form von Prüfredeln erfolgt entsprechend des RASE-Konzepts (siehe 3.2). Hierbei entspricht die Auswahl der **IfcBuildingElementProxys** dem Select. Die Beschreibung der Komponenten erfolgt über das Property-Sets **Allgemeine_Eigenschaften** und die Identifizierung in Form des Attributs *bauteilklassifikation*. Diese Kombination entspricht der Applicability. Die geforderten der Property-Sets und zugehörigen Attribute entsprechen den Requirements.

In der Beschreibung der Testmodelle (4.1) werden die Modellkomponenten mit ihrer Klassifikation und einer Liste der zugehörigen geforderten PropertySets aufgelistet.

Beispiel:

Wenn ein **IfcBuildingElementProxy** mit dem *bauteilklassifikation*-Wert *br.ue.11* vorhanden ist, müssen die PropertySets **Allgemeine_Eigenschaften**, **Bauteil** und **BRP** zugewiesen sein inklusive ihrer zugewiesenen Attribute. Sofern beispielsweise das Attribut *name* (**Allgemeine_Eigenschaften**) nicht vorhanden oder statt dem vorgegebenen Datentyp „String“ mit einem „boolean“ belegt ist, sollte eine Fehlermeldung ausgelöst werden. Die Modellkomponenten werden also auf Vollständigkeit ihrer Attribuierung nach 2.1.5 geprüft.

Wie in Abbildung 24 durch den Haken bei „Lph-1 BRM_3“ (rechte Spalte) dargestellt wird, wurde dem dritten Anwendungsfall die **FesteFahrbahn** (*ob.bet.100*) zugewiesen, obwohl wie in 4.1.3 zu sehen sein wird, kein entsprechend attribuiertes Objekt im Modell vorhanden ist. Hintergrund dafür ist, dass gezeigt werden soll, dass diesbezüglich keine Fehlermeldung entsteht, da die Select-Exception-Auswahl der RASE-Methodik nicht greift. Es ist nämlich kein Objekt vom Typ **IfcBuildingElementProxy** mit dem Attribut *bauteilklassifikation* mit Wert „*ob.bet.100*“ vorhanden. Daher wird nicht geprüft ob einer derart klassifizierten Modellkomponente die geforderten PropertySets zugewiesen wurden. Folglich ist es vertretbar, nur eine Auswahl der Entitäten in das Beispielmmodell einzufügen, da weitere Entitäten anhand der genutzten IFC-Modelle nicht ausgewertet werden würden.

4 MODEL CONTENT CHECKING

Im Nachfolgenden wurde auf Basis des in BIMQ implementierten, neustrukturierten SOM ein „Model Content Checking“ (2.1.5) durchgeführt. Da das SOM nur die semantischen Aspekte des Modells beschreibt, wurde auch die Model Content Prüfung auf die Attribuierung beschränkt.

Es wurden mehrere Brückenmodelle auf das Vorhandensein aller gemäß des SOM geforderten Attribute geprüft. Darüber hinaus wurden mehrere Model Checker bezüglich der Prüfergebnisse miteinander verglichen. Um die unterschiedlichen Funktionsweisen der Model Checker zu verdeutlichen wurde - über das reine „Model Content Checking“ hinausgehend - im Beispielmmodell auch der diskrete Wertebereich eines Attributes geprüft (4.1.2, 4.3.2).

4.1. ZU PRÜFENDE MODELLE

Die oben beschriebenen Anforderungen sind noch nicht in bestehenden Projekten realisiert. Es mussten also zunächst Modelle erstellt oder bestehende angepasst werden, an denen anschließend die Vollständigkeitsanforderungen getestet werden konnten. Um dies in einem angemessenen Rahmen zu halten, beschränkten sich die untersuchten Model Content Checkings auf die Gewerke BR und FB. Durch den Umstand begünstigt, dass bereits mehrere Brückenmodelle zur Verfügung standen, die günstiger Weise auch noch mit Straßen oder Gleisanlagen kombiniert waren, mussten diese lediglich noch attribuiert werden. Die Modellierung und Attribuierung erfolgte in Revit und die Modelle wurden anschließend ins IFC-Format exportiert.

Um die Regel im Falle von Fehlern effektiv anpassen und zusätzlich die Resultate in den genutzten Softwarelösungen vergleichen zu können, wurden drei Modelle unterschiedlicher Komplexität und unterschiedlicher Anzahl und Art von Komponenten betrachtet. Die Modelle werden im Nachfolgenden kurz beschrieben.

4.1.1. MODELL 1: BRÜCKENRAMPE

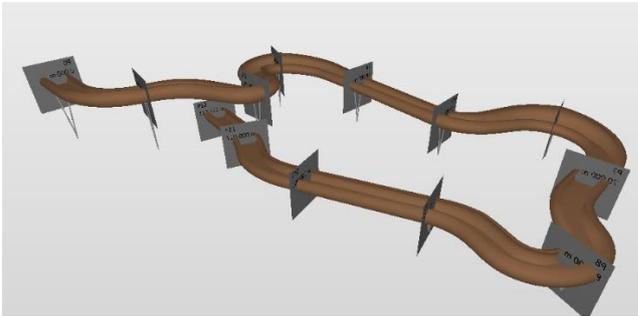


Abbildung 25: Brückenmodell 1

Beim Brückenmodell 1 handelt es sich um das Modell einer Fußgängerbrücke, die optisch an eine Marmelbahn angelehnt ist [69]. Die Struktur des Modells ist simpel gehalten und besteht ausschließlich aus **IfcBuildingElementProxys**. Diese sind lediglich in zwei Typen unterteilt: **Stuetzen** und **Ueberbau**. Folglich betrachten wir nur zwei Arten von Entitäten des SOM (3.3.4). In Klammern sind die zugehörigen PropertySets angegeben:

- br.ue: **Ueberbau** (Allgemeine_Eigenschaften, UE)
- br.ub.20: **Stuetze** (Allgemeine_Eigenschaften, Bauteil, STU)

Bei keiner der **Stuetzen** wurden die Attribute *bemerkung* und *KGK-Verzeichnisnummer* attribuiert. Das Attribut *identitaet* weist nur für eine Stütze einen Wert auf.

Weiterführende Hintergründe zu Modellierung und Struktur des Modells können [69] entnommen werden.

4.1.2. MODELL 2: STAHLHOHLTRÄGERBRÜCKE

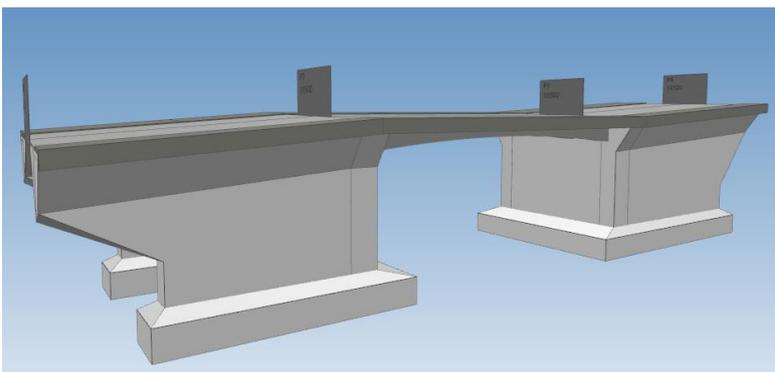


Abbildung 26: Brückenmodell 2

Das Brückenmodell 2 stellt eine schlichte Brücke dar, bestehend aus einer Brückenplatte und zugehörigen Brückenträgern. Es setzt sich aus mehr Komponenten als das erste und durch die Fahrbahn ist auch eine Komponente eines zweiten Gewerks enthalten. Die verwendeten Komponenten sind:

- br.ub.10: **Widerlager** (Allgemeine_Eigenschaften, Bauteil, WL)
- br.ue.11: **Brueckenplatte** (Allgemeine_Eigenschaften, Bauteil, BRP)
- br.ue.12: **Brueckentraeger** (Allgemeine_Eigenschaften, Bauteil, BRT)
- br.ue.20: **Kappe** (Allgemeine_Eigenschaften, Bauteil, KA)
- ob.bet.100: **FesteFahrbahn** ((Allgemeine_Eigenschaften, Bauteil, BET, FF)

Die Attribuierung wurde möglichst genau anhand des SOM durchgeführt, wobei aus Testzwecken jedoch eine kleinere Abweichung eingebaut wurde: In BIMQ wurde festgelegt, dass das Attribut *widerlagerart* des PropertySets **WL** nur die Werte „rahmenbruecke“ und „standard“ annehmen kann. Für das Widerlager 1 des Modells wurde dieser Attributwert jedoch mit „sonderanfertigung“ attribuiert.

4.1.3. MODELL 3: EISENBAHNÜBERFÜHRUNG INGWEILER

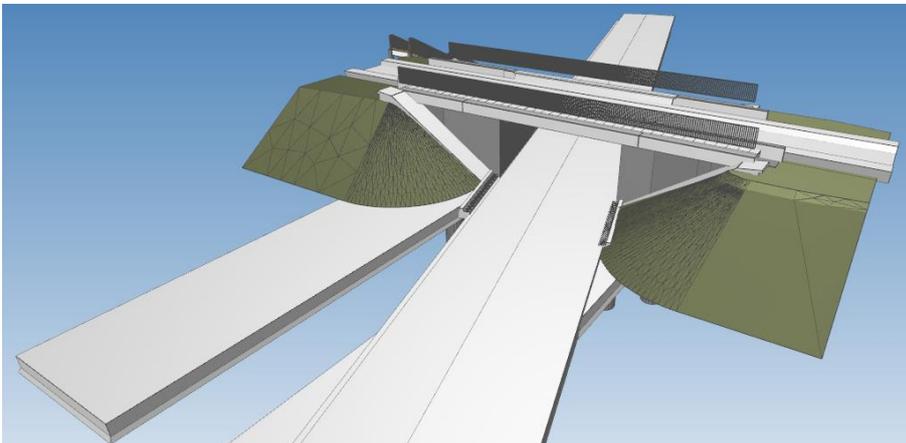


Abbildung 27: Brückenmodell 3

Das Brückenmodell 3 ist ein Eisenbahnüberführungsmodell der DB Netz AG. Es diente ferner als Orientierungshilfe für die Erweiterung des Gewerks BR. Es besteht aus einer Vielzahl von Modellkomponenten:

- br.ub: **Unterbau** (Allgemeine_Eigenschaften, UE)
- br.ub.10: **Widerlager** (Allgemeine_Eigenschaften, Bauteil, WL)
- br.ub.12: **Fluegelwand** (Allgemeine_Eigenschaften, Bauteil, FW)
- br.ub.13: **Widerlagerfundament** (Allgemeine_Eigenschaften, Bauteil, WDF)
- br.ub.14: **Widerlagerhinterfuellung** (Allgemeine_Eigenschaften, Bauteil, WHF)
- br.ue.11: **Brueckenplatte** (Allgemeine_Eigenschaften, Bauteil, BRP)
- br.ue.20: **Kappe** (Allgemeine_Eigenschaften, Bauteil, KA)
- br.gr.130: **Bohrpfahl** (Allgemeine_Eigenschaften, Bauteil, BP)
- br.asr.10: **Gelaender** (Allgemeine_Eigenschaften, Bauteil, GEL, *hierbei handelt es sich um kein Mapping auf ein IfcBuildingElementProxy sondern auf IfcRailing*)
- br.asr.60: **Lager** (Allgemeine_Eigenschaften, Bauteil, LG)
- ob.bet.100: **FesteFahrbahn** (Allgemeine_Eigenschaften, Bauteil, BET, FF)

- ob.bet.140: **Tragschicht** (Allgemeine_Eigenschaften, Bauteil, TS)
- ob.bet.170: **Seitenfuehrungsblock** (Allgemeine_Eigenschaften, Bauteil, SFB)

Die Attribuierung orientiert sich möglichst genau am SOM. Absichtliche Fehler wurden beispielhaft nur an den **Gelaendern** eingebaut. Dies ist der Tatsache geschuldet, dass in Revit die Zuordnung von Attributen, welche von mehreren Objekttypen genutzt werden, über „geteilte Parameter“ geschehen muss. In einer ersten Attribuierung wurde dies nicht berücksichtigt und das Gelände konnte entsprechend nicht vollständig attribuiert werden. Daher bot es sich an, an dieser Stelle die Gewichtung von Fehlern zu überprüfen (siehe 4.3.3). Den **Geländern** fehlen die PropertySets **Allgemeine_Eigenschaften** und **Bauteil**, sowie das Attribut *idAusruistung* des PropertySets **GEL**. Dementsprechend sollten die Fehlermeldungen unterschiedlich gewichtet werden (fehlendes PropertySet und fehlendes Attribut).

Des Weiteren wurde das Modell einmal vollständig über geteilte Parameter attribuiert, um eine fehlerfreie Prüfung ausführen zu können.

Anmerkung: Bevor die Prüfergebnisse betrachtet werden können, muss bereits einmal vorgegriffen werden. Mehrere Objekte verfügen über Attribute vom Datentyp Integer. Es handelt sich dabei um die **Flügelwaende** mit dem Attribut *achse*, die **Lager** mit dem Attribut *achse*, die **Widerlagerhinterfuellungen** mit dem Attribut *proktordichte* und die **Bohrpfahle** mit dem Attribut *einbauWinkel*.

Dies wurde auch bei der Attribuierung des Modells in Revit, den Übersetzungen in IFC und den BIMQ-Prüfregeln berücksichtigt. Eine Überprüfung des IFC-Step-Codes mittels „notepad++“ zeigte, dass die Attribute auch als IFCInteger aus Revit exportiert wurden, wie die nachfolgenden Codezeilen aus dem IFC-Step-Code zeigen:

```
#306= IFCPROPERTY SINGLEVALUE('einbauWinkel',$,IFCINTEGER(0),$);
```

```
#1465= IFCPROPERTY SINGLEVALUE('achse',$,IFCINTEGER(1),$);
```

```
#7494= IFCPROPERTY SINGLEVALUE('proktordichte',$,IFCINTEGER(100),$);
```

Dies sollte für die Ergebnisse der Prüfung dieses Modells im Hinterkopf behalten werden.

4.2. DIE PRÜFUNGSVORBEREITUNG

Um die Modelle überprüfen zu können, müssen sie zunächst ins IFC-Format exportiert werden, da die Model Checker das Revit zugrundeliegende Format .rvt nicht öffnen können. Für den zugehörigen Export muss eine PropertySet-Beschreibung als .txt-Datei für die Übersetzung der Revit-Bauteillisten in IFC-PropertySets erstellt werden [93]. Nach diesem Schritt liegt die erfolgte Attribuierung auch im IFC-Modell vor.

In der .txt-Datei müssen die einzelnen PropertySets mit all ihren Attributen und den zugehörigen Datentypen den entsprechenden IFC-Entitäten zugewiesen werden. Unter anderem können Attribute hierbei neuen Namen zugeordnet werden. So kann die *IfcGUID* aus Revit in das Attribut *identitaet* übersetzt werden. Zusätzlich müssen noch die Datentypen der Attribute festgelegt werden. In der Abbildung 28 ist die Übersetzung für die PropertySets **Allgemeine_Eigenschaften** und **Bauteil** aufgeführt.

```
PropertySet:  Allgemeine_Eigenschaften      I      IfcBuildingElement
bauteilKlassifikation  Text
bauteilName            Text
bemerkung              Text
KGK-Verzeichnisnummer Text
name                   Text
identitaet             Text

PropertySet:  Bauteil I      IfcBuildingElement
betriebszustand Text
technischerPlatz  Text
technischerPlatzBezeichnung Text
material          Text
materialEigenschaften Text
```

Abbildung 28: Auszug der IFC-Übersetzungs-txt.-Datei

Da die Modelle fast ausschließlich aus „Allgemeinen Modellen“ bestehen, gestaltet sich die Zuordnung der PropertySets entsprechend schwierig. In Revit werden vorerst allen Allgemeinen Modellen alle Properties zugeordnet, wodurch die Modelle in Revit unnötig aufgebläht werden. Beim Export werden nicht attribuierte Attribute jedoch nicht exportiert, so dass die IFC-Modelle im Gegensatz zu den Revit-Modellen mit Ausnahme von „doppelten“ PropertySets keine Redundanzen aufweisen. Mit „doppelten“ ist gemeint, dass PropertySets mit gleichen Attributen allen Objekten zugewiesen werden, die diese Attribute besitzen.

Die in BIMQ erstellten Anforderungen wurden bei Bedarf noch angepasst und korrigiert.

Der Export der Prüfredeln aus BIMQ in die benötigten Formate erfolgte automatisch.

Der Import der Dateien in die entsprechenden Softwarelösungen erfolgte wie in 1.6 beschrieben und mit unterschiedlichem Aufwand. In *Desite* und *BIMcollab ZOOM* musste nur der Import durchgeführt werden, in *Solibri* hingegen müssen die Regeln explizit erstellt werden, wobei der BIMQ-Export nur als Ergänzung dient.

Die Regelsätze und ihre Einstellungen werden nachfolgend kurz erläutert:

Einstellungen der Regelsätze in Solibri:

The screenshot shows the 'PARAMETER' window in Solibri. The title bar includes 'PARAMETER' and 'Schweregradparameter'. Below the title bar, there is a section 'Überprüfte Komponenten' with a table of checked components. Below that, there is a table of property sets from an Excel file 'Eigenschaftensets (C:\Users\dsonn\Documents\TU Dresden\10.Semester\Diplomarbeit\Prüfregeln\Solibri\BRM2_SOM_V3.xlsx)'. The table lists components, property sets, properties, and their values and conditions.

Status	Komponente	Eigenschaft	Funktion	Wert
Einschließen	Beliebige	Allgemeine_Eigenschaften.ba...	Stimmt überein	br.ub.10

Komponente	Eigenschaftenset	Eigenschaft	Wert ist vorhanden	Wertbedingungen	Visualisierung
<input type="radio"/> Objekt	Allgemeine_Eigenschaften	bauteilKlassifikation	Muss vorhanden sein	X = Text	
<input type="radio"/> Objekt	Allgemeine_Eigenschaften	bauteilName	Muss vorhanden sein	X = Text	
<input type="radio"/> Objekt	Allgemeine_Eigenschaften	bemerkung	Muss vorhanden sein	X = Text	
<input type="radio"/> Objekt	Allgemeine_Eigenschaften	identitaet	Muss vorhanden sein	X = Text	
<input type="radio"/> Objekt	Allgemeine_Eigenschaften	KGK-Verzeichnisnum...	Muss vorhanden sein	X = Text	

Abbildung 29: Einstellungen einer Prüfregel in Solibri

Die Erstellung eines Prüfregelsatzes erfolgt im *Ruleset Manager*. Ausgangspunkt für die Regelsatz-Erstellung in Solibri ist die Wahl der Prüfungsart. Die Prüfungsarten sind Vorlagen an Regelsetzung für verschiedene Aspekte des BMC. In diesem Fall wird die Prüfungsart „Erforderlichen Attribute“ gewählt. Für jeden Komponententyp, der geprüft werden soll, muss dieser Prüfregeltyp einmal in den zu erstellenden Regelsatz eingefügt werden.

Im Anschluss daran muss jeder dieser Regelsatztypen präzisiert werden, d.h. es muss festgelegt werden, welche Art von Komponenten unter welchen Voraussetzungen geprüft werden muss. In diesem Fall bedeutet das stets, dass „Beliebige Komponenten“ (keine Einschränkungen) genau dann mit einem entsprechenden Regelsatz geprüft werden, wenn bei diesen das Attribut *bauteilKlassifikation* aus dem PropertySet **Allgemeine_Eigenschaften** mit der Nomenklatur des entsprechenden Bauteils übereinstimmt, beispielsweise „br.ub.10“ für ein Widerlager wie dies in Abbildung 29 zu sehen ist. Diese Festlegung was ein Wiederlager ist, wurde bereits in BIMQ getroffen und wird auch in den anderen Prüfertools problemlos übernommen. In Solibri hingegen muss dies neu eingestellt werden.

Sofern die zu prüfenden Modellkomponenten gewählt sind (A, S und E) können die die zugehörigen Requirements für diese festgelegt werden. Diese können entweder manuell erstellt oder aus einem Excel-Sheet-Import ausgewählt werden. Ein derartiges Excel-Sheet wird auch beim Solibri-Export der Prüfregeln aus BIMQ erzeugt. Die in Abbildung 29 dargestellten Attribute und die Prüfregel „Wert ist vorhanden“ wurden aus den exportierten Excel-Sheets ausgelesen. Daher ergänzt BIMQ hier nur die Prüfregelerstellung und stellt die Prüfregeln nicht direkt.

Anmerkung: Der BIMQ-Export für Solibri für die Anforderungen an das erste Modell ist als Beispiel unter „BIMQ-Export BRM1 Solibri“ zu finden. Die BIMQ-Exporte für die anderen Modelle sind im Digitalen Anhang zu finden.

Einstellung der Prüfredeln in Desite:



Abbildung 30: Darstellung von Regelsätzen in Desite

Die Einstellung der Regelsätze erfolgt hier denkbar unkompliziert. Es muss lediglich im Slot „Modellprüfung“ der Import angestoßen und der entsprechende Regelsatz aufgerufen werden.

Anschließend kann noch ausgewählt werden welche der Prüfredeln durchgeführt werden sollen. Bei Bedarf können die Regeln, welche als „*java script*“-Code vorliegen, ergänzt oder korrigiert werden. Ein Auszug einer Prüfredelauswahl ist in Abbildung 30 (links) zu sehen. Die rechte Seite der Abbildung zeigt beispielhaft einen Auszug der Regeln für eine Brückenplatte. Zu sehen ist zum einen die Auswahl der Komponenten als **IfcBuildingElementProxys** mit dem Wert „br.ue.11“ für das Attribut *bauteilKlassifikation* sowie die Prüfung auf das Vorhandensein des entsprechenden dieses Attributs.

Anmerkung: Der BIMQ-Export für Destie für die Anforderungen an das erste Brückenmodell ist als Beispiel unter „BIMQ-Export BRM1 Desite“ zu finden. Die BIMQ-Exporte für die anderen Modelle sind im Digitalen Anhang zu finden.

Einstellung der Prüfreden in BIMcollab ZOOM:

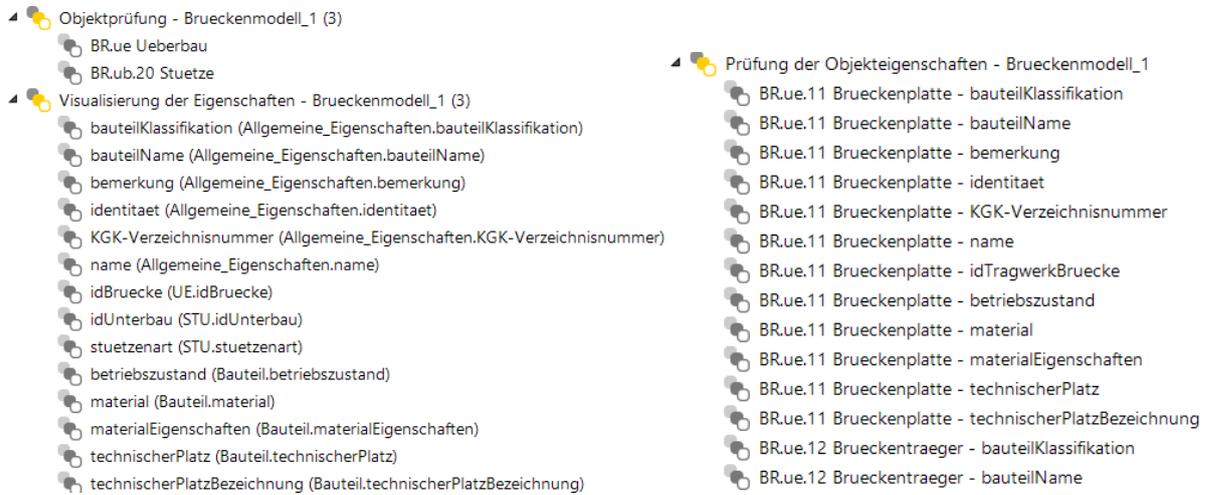


Abbildung 31: Darstellung der Regeln/Smart-Views in BIMcollab ZOOM

Wie in 1.6.4 beschrieben, werden in BIMcollab ZOOM sogenannte Smart-Views genutzt. Bei diesen handelt es sich um eine farbliche Darstellung der geprüften Modellkomponenten. Im Standardfall hat eine grün eingefärbte Modellkomponente die Prüfung bestanden, eine rot eingefärbte Komponente weist auf einen Fehler hin. Wie in *Desite* kann einfach ein Import angestoßen und die entsprechenden Smart-Views ausgeführt werden. Die in BIMQ erstellten Regelsätze für einen **Ueberbau** und eine **Stuetze** sind in Form von Smart-Views in Abbildung 31 unter „Objektprüfung – Brückenmodell_1“ zu sehen. Anders als bei *Solibri* und *Desite* werden aus den Prüfreden zusätzlich Teilregeln generiert. Diese sind in Abbildung 31 unter „Visualisierung der Eigenschaften“ und „Prüfung der Objekteigenschaften“ zu finden. Mittels dieser zusätzlich erzeugten Regeln werden zum einen alle Modellkomponenten auf das Vorhandensein einzelner Attribute untersucht. Zum anderen werden durch die automatisch aus dem BIMQ-Import erzeugten Smart-Views „Objekteigenschaftsprüfung“ die einzelnen Eigenschaften der Modellkomponenten auf die Einhaltung der geforderten Eigenschaften zu prüfen. Hierunter fallen beispielsweise diskrete Wertebereiche und Datentypen. Diese beiden Arten der Abweichung werden orange dargestellt.

Anmerkung: Der BIMQ-Export für BIMcollab ZOOM für die Anforderungen an das erste Brückenmodell ist als Beispiel unter „BIMQ-Export BRM1 BIMcollab ZOOM“ zu finden. Die BIMQ-Exporte für die anderen Modelle sind im Digitalen Anhang zu finden.

4.3. RESULTATE DER MODELLPRÜFUNG

Nachfolgend werden die Resultate der Prüfungen der Modelle in den verschiedenen Softwarelösungen Solibri, Desite md Pro und BIMcollab ZOOM beschrieben und verglichen. Vor der Prüfung wird stets eine Prognose der Ergebnisse aufgestellt und diese dann mit den Resultaten abgeglichen.

4.3.1. MODELL 1

Wie bereits in 4.1.1 beschrieben wurde, sind die **Stuetzen** nicht vollständig attribuiert. Bei den **Allgemeine_Eigenschaften** fehlen mehrere Attribute. Dementsprechend sollte es für jede der **Stuetzen** mehrere Fehlermeldungen aufgrund fehlender Attribute geben. Der **Ueberbau** wiederum sollte die Prüfung fehlerfrei bestehen.

Solibri:

ERGEBNISÜBERSICHT					
Problemanzahl	0	3	0	0	0
Problemdichte	0	0.41	0	0	0

ERGEBNISSE	
Fehlende Eigenschaften (0/3)	
Objekt Komponenten (0/3)	
Allgemeine_Eigenschaften - KGK-Verzeichnisnummer	
Allgemeine_Eigenschaften - bemerkung	
Allgemeine_Eigenschaften - identitaet	

Abbildung 32: Auswertung des Brückenmodells 1 in Solibri

In Abbildung 32 werden die zu prüfenden Regelsätze und die akkumulierten Prüfergebnisse gezeigt. Wie prognostiziert ist die Eigenschaftsprüfung des **Ueberbaus** fehlerfrei (OK). Für die **Stuetzen** gibt es wie vorhergesagt mehrere Fehlermeldungen wegen der nicht vorhandenen Attribute:

- *KGK-Verzeichnisnummer*
- *bemerkung*
- *identitaet*

Da es keine weiteren Abweichungen gibt, ist die Prognose korrekt.

Desite:



Abbildung 33: Auswertung des Brückenmodells 1 in Desite

Wie Abbildung 33 auszugsweise zeigt, ergibt die Auswertung in Desite ein gleichartiges Ergebnis wie Solibri. Der **Ueberbau** weist keine Fehler auf (passed), die **Stuetzen** werden alle mit einem „Warning“ gekennzeichnet. Die Fehler sind die gleichen wie in Solibri. Interessant ist, dass in Desite und nachfolgend auch in BIMcollab ZOOM die Groß- und Kleinschreibung beachtet wird. In einem ersten Versuch gab es die Fehlermeldung, dass *Name* nicht vorhanden war, obwohl dies, wie in Abbildung 33 rechts zu sehen ist, doch mit „Brueckenrampe“ attribuiert ist. Entsprechend musste das Attribut-Mapping in BIMQ von *Name* zu *name* geändert werden. Nach dieser Änderung entsprach die Prognose den Resultaten.

Die blau markierten Modellkomponenten wurden im Prüfverlauf ignoriert, da sie die Voraussetzungen zur Prüfung (A, S, E) nicht erfüllten. Bei diesen Komponenten handelt es sich um die Achse. Diese ist ein Resultat des Modellierungs-Plug-Ins mit dem die Rampe in Revit modelliert wurde. Für die weitere Nutzung hat sie aber keine Relevanz. Es handelt sich bei der Achse zwar um ein **IfcBuildingElementProxy** (S), aber ohne das PropertySet **Allgemeine_Eigenschaften** und ohne das Attribut *bauteilKlassifikation* (A).

BIMcollab Zoom:

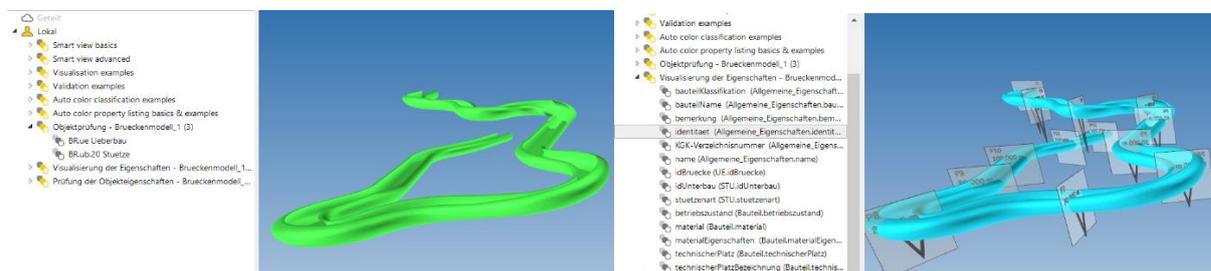


Abbildung 34: Auswertung des Brückenmodells 1 in Bimcollab Zoom (l.) und Attributprüfung(r.)

Wie bereits in 1.6.4 erläutert, gibt es in BIMcollab ZOOM keine direkte Modellprüfung mit einem anschließenden Bericht oder einer Fehlerbewertung. Stattdessen bietet BIMcollab ZOOM Smart-Views an. Die importierten Smart-Views stellen jeweils die Prüfung eines Entitätstyps bzw. Regelsatzes dar. Im gegebenen Fall werden alle Objekte die als **Ueberbau** klassifiziert sind geprüft und einmal alle Objekte, welche als **Stuetze** klassifiziert sind.

Abbildung 34 zeigt durch die Grünfärbung an, dass der **Ueberbau** die Regeln erfüllt. Die **Stuetzen** dagegen werden rot gezeigt (ohne Abbildung), erfüllen also nicht alle Anforderungen. Für dieses Ergebnis musste die Attribuierung bezüglich der Groß- und Kleinschreibung angepasst werden, wie auch zuvor bei Desite.

Interessant ist, dass man durch die Smart-Views auch auf einzelne Attribute prüfen kann. In Abbildung 34 (rechts) werden alle Objekte eingefärbt, welche über das Attribut *identitaet* verfügen. Als Ergebnis werden der **Ueberbau** und die erste **Stuetze** eingefärbt, die anderen Objekte nicht.

4.3.2. MODELL 2

Im zweiten Modell wurden eine größere Zahl von Objekten und ein simplifiziertes Zusammenspiel der zwei Gewerke FB und BR modelliert. Da die Attribuierung - abgesehen von dem absichtlich falsch gewählten Attributwert bei einem Widerlager - so akkurat wie möglich durchgeführt wurde, sollten alle Modellkomponenten die Regelprüfung fehlerfrei bestehen. Lediglich für das fehlerhafte **Widerlager** sollte es zu einer Warnung oder Fehlermeldung führen.

Solibri

The screenshot displays the Solibri software interface during a rule check. The main window is titled 'ÜBERPRÜFEN' (Check) and shows a tree view of rules under 'Regelsatz: Überprüftes Modell'. The tree view includes a folder 'BRM2' with several rules, all of which are marked as 'OK'. Below the tree view is a summary table showing the results of the check:

Problemanzahl	0	1	0	0	0
Problemdichte	0	0.14	0	0	0

Below the summary table is a section for 'ERGEBNISSE' (Results) showing a list of results. The first result is 'Nicht akzeptable Eigenschaftswerte (0/1)' with a sub-entry 'Objekt Komponenten (0/1)'. The second result is 'WL.widerlagerart.sonderanfertigung: Aufzählung' with a sub-entry 'Objekt.0.6'. The 'INFORMATIONEN' (Information) section at the bottom shows details for 'Objekt.0.6', including its identification, position, material, and classification.

Abbildung 35: Auswertung des 2. Brückenmodells in Solibri

Die Ergebnisse der Regelprüfung in Solibri für das zweite Brückenmodell wird in Abbildung 35 gezeigt. Die im Voraus getroffenen Überlegungen wurden voll und ganz bestätigt. Abgesehen von besagtem **Widerlager** erfüllten alle geprüften Modellkomponenten die Regelsätze. Beim Widerlager tritt der bereits vorhergesagte Fehler auf. Der Attributwert „sonderanfertigung“ für *widerlagerart* entspricht keinem der Werte der Aufzählung.

Desite

No.	Rule	Text	Created User
1	Lph-1-BRM_2:Widerlager Eigenschaft *widerlagerart -> WL:widerlagerart* besitzt nicht den richtigen Wert!	2021-07-14T12:59:12	dsonn
2	Lph-1-BRM_2:Widerlager 1 von 13 Eigenschaft(en) nicht oder nicht wie gefordert vorhanden!	2021-07-14T12:59:12	dsonn

Abbildung 36: Auszug der Auswertung des 2 .Brückenmodells in Desite

In obiger Abbildung 36 wird ein Auszug der Modellprüfung in Desite gezeigt. Als Gesamtergebnis ergab sich eine Warnung. Wie prognostiziert ist, bestanden abgesehen von einem **Widerlager** alle Modellkomponenten die Regelprüfung. Für das **Widerlager** wurde wie erwartet die Warnung ausgegeben, dass das Attribut *widerlagerart* nicht den richtigen Wert besitzt, also weder „rahmenbruecke“ oder „standard“.

Bei dem ignorierten Objekt (blau markiert) handelt es sich wie beim ersten Modell um Überreste des Modellierungstools.

BIMcollab ZOOM

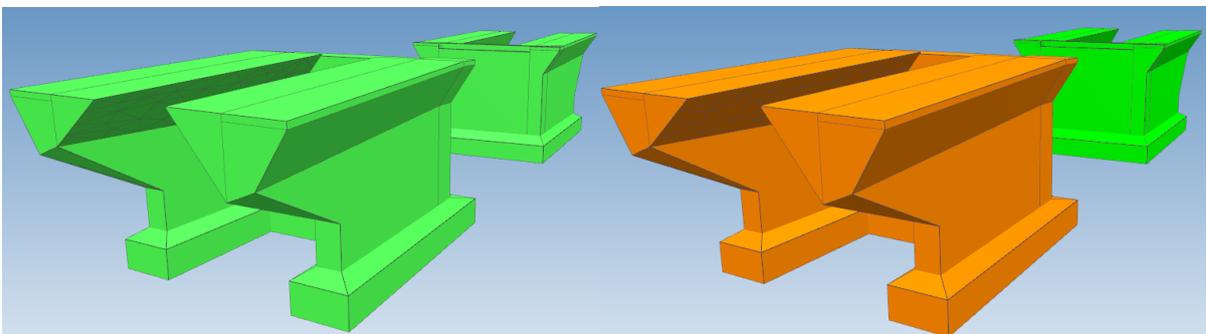


Abbildung 37: Auswertung der Widerlager in BIMcollab ZOOM Komponentenprüfung (l.) Objekteigenschaftsprüfung (r.)

Mit Ausnahme der **Widerlager** ergibt Auswertung der Modellkomponenten in BIMcollab ZOOM die gleichen korrekten Ergebnisse wie Desite. Deshalb wird in Abbildung 37 nur die

Auswertung der zugehörigen Smart-Views für die **Widerlager** gezeigt. In diesem Fall wird die Prognose aber auf den ersten Blick (l.) überraschenderweise nicht erfüllt, denn die Prüfung wird wider Erwarten als erfolgreich angezeigt.

Der Grund hierfür konnte in den Smart-View-Konfigurationen ausgemacht werden.

Name BR.ub.10 Widerlager					
Elementtyp	Eigenschaft	Operator	Wert	Aktion	
Any element	IFC Element	Ist	IfcBuildingElementProxy	Und...	
	bauteilKlassifikation	Ist	br.ub.10	Hinzufügen	
Any element	None			Farbig einstellen	
Any element	bauteilKlassifikation	Ist nicht definiert		Farbig einstellen	
Any element	bauteilName	Ist nicht definiert		Farbig einstellen	
Any element	bemerkung	Ist nicht definiert		Farbig einstellen	
Any element	identitaet	Ist nicht definiert		Farbig einstellen	
Any element	KGK-Verzeichnisnummer	Ist nicht definiert		Farbig einstellen	
Any element	name	Ist nicht definiert		Farbig einstellen	
Any element	idUnterbau	Ist nicht definiert		Farbig einstellen	
Any element	widerlagerart	Ist nicht definiert		Farbig einstellen	
Any element	betriebszustand	Ist nicht definiert		Farbig einstellen	
Any element	material	Ist nicht definiert		Farbig einstellen	
Any element	materialEigenschaften	Ist nicht definiert		Farbig einstellen	
Any element	technischerPlatz	Ist nicht definiert		Farbig einstellen	
Any element	technischerPlatzBezeichnung	Ist nicht definiert		Farbig einstellen	

Abbildung 38: Smart-View-Einstellungen

Die aus dem BIMQ importierten Smart-Views für Modellkomponenten prüfen lediglich, wie aus Abbildung 38 hervor geht, auf das Vorhandensein von Attributen, nicht aber ob ein Wert oder Datentyp eingehalten ist.

Die in BIMQ konzipierte und beispielsweise in Desite geprüfte Beschränkung der *widerlagerart* auf zwei diskrete Werte oder gar einen Datentyp erfolgt in BIMcollab ZOOM über die „Prüfung der Objekteigenschaften“ genannten automatisch aus BIMQ exportierten Smart Views. Hier kann mittels der Objekteigenschaftsprüfung „widerlagerart“ erfolgreich der falsche Attributwert identifiziert werden (Abbildung 37 (r.)).

4.3.3. MODELL 3

Wie aus der Modellbeschreibung in 4.1.3 hervorgeht wurde das Modell möglichst akkurat attribuiert. Daher sollten sich die Fehlermeldungen auf die **Gelaender** beschränken, da diesen nicht alle PropertySets zugewiesen wurden und manche PropertySets unvollständig waren. Des Weiteren sollte deutlich werden, dass die **FesteFahrbahn** nicht geprüft wird, da keine der Modellkomponenten die Voraussetzungen zur Durchführung einer derartigen Prüfung erfüllt.

Solibri

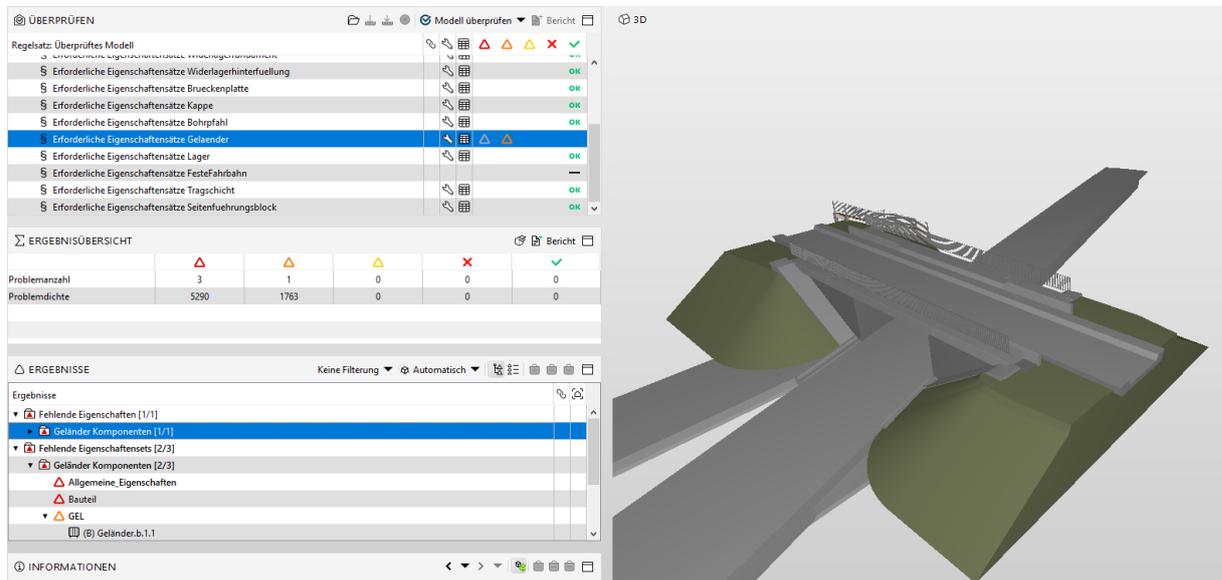


Abbildung 39: Auswertung der Modellprüfung des 3.Brueckenmodells in Solibri

Anders als bei den anderen betrachteten Softwarelösungen, welche die Fehler nicht gewichten, sollte das Fehlen der PropertySets in Solibri zu einer schwereren Fehlerbewertung als das Fehlen von Attributen führen. Die Prüfungsergebnisse des Modells passten genau zu den getroffenen Vorhersagen. Nahezu alle Modellkomponenten erfüllten die Anforderungen. Passenderweise wurden auch das Fehlen ganzer PropertySets (rotes Warnschild) und das Fehlen einzelner Attribute (oranges Warnschild) für die **Geländer**-komponenten unterschiedlich gewichtet, wie Abbildung 39 in Auszügen entnommen werden kann. Dies entsprach der Vorhersage ebenso wie die Tatsache, dass die Prüfung der **Feste Fahrbahn** entfiel, was durch einen Querstrich in der entsprechenden Zeile gekennzeichnet wurde.

Desite

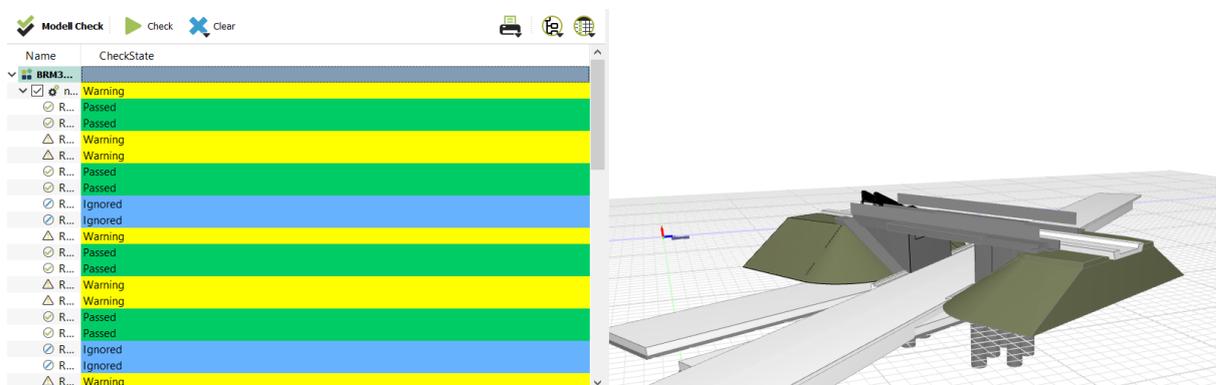


Abbildung 40: Auszug der Auswertung des 3. Brueckenmodells in Desite

Die Resultate der Prüfung in Desite überraschten: Im Gegensatz zu der erfolgreich bestätigten Prognose in Solibri wichen die Regelprüfungsergebnisse in Desite signifikant von den Erwartungen ab.

Zunächst werden kurz die Übereinstimmungen betrachtet. Die **FesteFahrbahn** wurde wie erwartet nicht geprüft. Abgesehen von den **Fluegelwaenden, Bohrpfaehlen** und **Gelaendern** bestanden alle Modellkomponenten die Prüfung. Wie Abbildung 41 beispielhaft aufzeigt, wurden nicht vorhandene Attribute korrekterweise bemängelt.

No.	Rule	Eigenschaft	Text	Created User
1	Lph-1-BRM_3:Gelaender	*Allgemeine_Eigenschaften:bauteilKlassifikation* nicht vorhanden!	2021-07-14T15:27:15	dsonn
2	Lph-1-BRM_3:Gelaender	*Allgemeine_Eigenschaften:bauteilName* nicht vorhanden!	2021-07-14T15:27:15	dsonn

Abbildung 41: Beispiel für die Fehlermeldungen der Gelaenderkomponenten

Hier nochmals der Verweis auf 4.1.3, also die Tatsache, dass alle geforderten Integer-Attribute auch als solche attribuiert und vorhanden sind. Zusätzlich wird nachfolgend in Abbildung 42 gezeigt, dass die entsprechenden Attribute auch in der Darstellung des Brückenmodells in Desite vorlagen und nicht beim Import verloren gingen.

BP

: einbauWinkel	0	xs:int
: idTiefgruendung	TG_WL_2	xs:string
: pfahlArt	bohrpfahl	xs:string

FW

: achse	1	xs:int
---------	---	--------

Abbildung 42: Attributtypen von achse und einbauWinkel

Entsprechend überraschend waren die in Abbildung 43 beispielhaft aufgeführten Fehlermeldungen.

No.	Rule	Eigenschaft	Text	Created User
1	Lph-1-BRM_3:Bohrpfahl	*BP:einbauWinkel* nicht vorhanden!	2021-07-14T15:27:15	dsonn
2	Lph-1-BRM_3:Bohrpfahl	1 von 14 Eigenschaft(en) nicht oder nicht wie gefordert vorhanden!	2021-07-14T15:27:15	dsonn

No.	Rule	Eigenschaft	Text	Created User
1	Lph-1-BRM_3:Fluegelwand	*FW:achse* nicht vorhanden!	2021-07-14T15:27:15	dsonn
2	Lph-1-BRM_3:Fluegelwand	1 von 14 Eigenschaft(en) nicht oder nicht wie gefordert vorhanden!	2021-07-14T15:27:15	dsonn

Abbildung 43: Beispiele für die Fehlermeldungen für Bohrpfaehle und Fluegelwaende

Laut Desite waren die entsprechenden Attribute nicht vorhanden, was zuvor aber in mehrfacher Weise widerlegt wurde und zwar sowohl im IFC-Step-Code als auch in der Desite-Interpretation des Modells. Die gleichen Fehlermeldungen ergaben sich ferner für die *achse* bei den Lagern und die *proktordichte* bei den Widerlagerhinterfuellungen.

Interessanterweise gibt es bezüglich des Double-Attributs *maximaleNeigung* der Widerlagerhinterfüllungen keine Probleme.

Der Schluss lag nahe, dass Desite die Integer-Attribute bei der Prüfung nicht als solche erkennen konnte.

Tatsächlich stellte sich heraus, dass die Übersetzung der BIMQ-Regeln in die Java-Script-Codes, die in Desite genutzt werden, Ursache für die fehlerhaften Prüfergebnisse waren. Wie wir bereits in Abbildung 42 sehen konnten, werden die Integervariablen als xs:int gespeichert, d.h. als auf 32 Bit beschränkte Integerzahl. Eine Analyse der Java-Script-Codes für die Prüfregele zeigte, wie in Abbildung 44 zu sehen ist, dass die Suche nach xs:integer erfolgt, also auf 64 Bit beschränkte Integer, und nicht nach xs:int.

```
desiteAPI.getPropertyValue(id,svalue10,'xs:integer');  
if (value10 == undefined) { checkfailed++;
```

Abbildung 44: Integerprüfung im exportierten Java-Script-Code

Auch wenn die xs:int eine Teilmenge der xs:integer sind, werden die Attribute nicht erkannt, obwohl sie mit Werten zwischen 0 und 2 auch vom Wertebereich passen. Von Seiten des IFC-Standards ist der Datentyp Integer als xs:integer vorgesehen, wie dies auch im Mapping in Desite erfolgt. Desite hingegen verwendet ein anderes Mapping auf xs:int. Zur erfolgreichen Prüfung in Desite muss entweder das Mapping für den Export der Prüfregele aus BIMQ angepasst werden oder eine manuelle Anpassung der Prüfcodes erfolgen. Bei Letzterer müsste in den entsprechenden Codes die Suche auf xs:int geändert werden. Eine derart modifizierte Modellprüfung ergibt das in Abbildung 45 abgebildete Prüfergebnis, das den Prognosen entspricht.



Abbildung 45: Prüfergebnis des 3. Brückenmodells nach einer Codeanpassung

Die Fehlermeldungen sind, wie vorhergesagt, auf die Gelaenderkomponenten beschränkt.

Der Vollständigkeit halber zeigt Abbildung 46 das Prüfergebnis des korrekt attribuierten Modells in Desite mit einer korrigierten Prüfregele. Es überrascht nicht, dass alle Prüfungen fehlerfrei durchlaufen wurden.

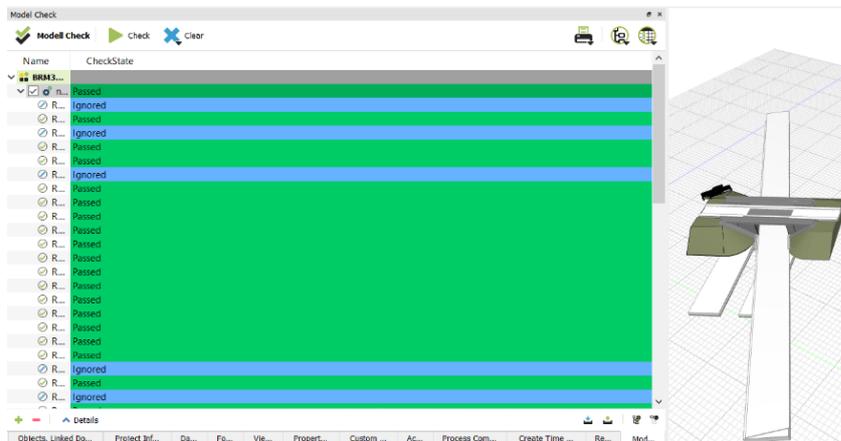


Abbildung 46: Die fehlerfreie Prüfung des 3. Brückenmodells

BIMcollab ZOOM

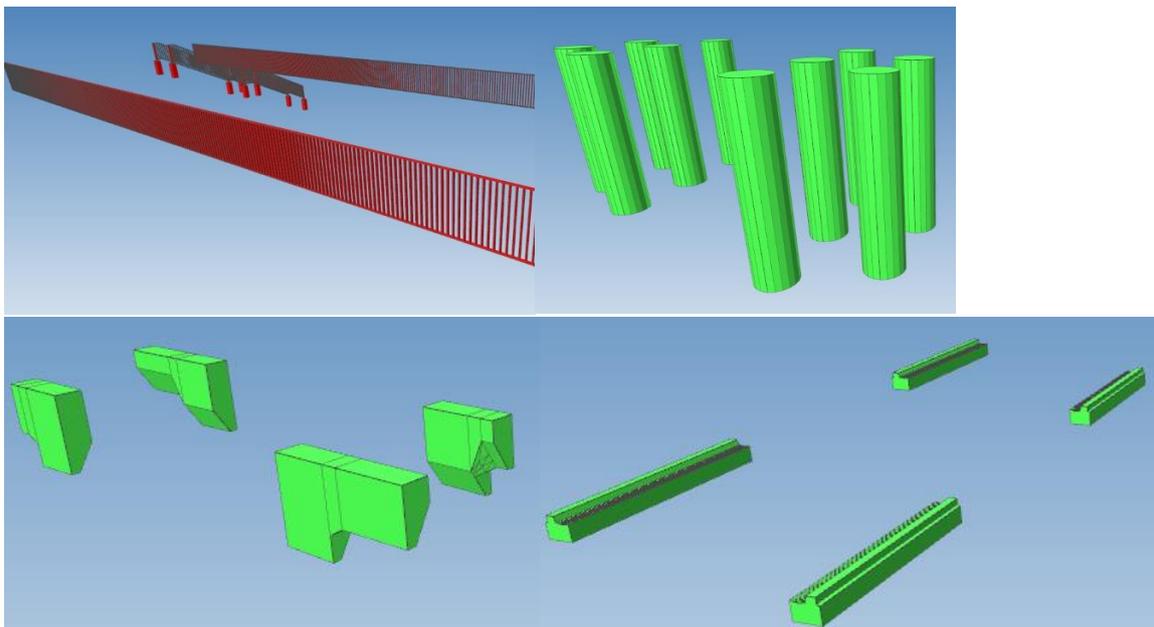


Abbildung 47: Beispiele der Smart-View-Ergebnisse für die Gelaender, Bohrpfahle, Fluegelwaende und Seitenfuehrungsbloেকে

Im Gegensatz zu den Ergebnissen in Desite erfüllen die Smart-Views die Vorhersagen. Fehlermeldungen gibt es lediglich bei den **Gelaendern**. Wie in Abbildung 47 zu sehen ist sind diese allesamt rot eingefärbt. Für die **FesteFahrbahn** zeigt der Smart-View keine Komponenten an. Probleme oder Überraschungen wie bei Desite treten weder bei den **Bohrpfahlen** noch bei den **Fluegelwaenden** auf, was an der IFC-konformen Umsetzung des Datentyps Integer liegt.

4.3.4. BEWERTUNG DER RESULTATE

Bei einer Gesamtbetrachtung der dargestellten Auswertungen wird deutlich, wie unterschiedlich die untersuchten Softwarelösungen funktionieren. Ruft man sich ins Gedächtnis, dass Desite und BIMcollab ZOOM die Exporte aus BIMQ direkt importieren, wohingegen die Regeln in Solibri unter Einbeziehung des BIMQ-Exports neu erzeugt werden, so ist es durchaus verwunderlich, dass die Modellprüfungen ausgerechnet für das BIMQ-kompatible Tool Desite md Pro fehleranfälliger ist. Die Problematik bei der Übereinstimmung des JavaScript-Codes von Desite mit der Software, wie wir es für die Integer-Variablen sahen, darf durchaus kritisch beurteilt werden. Auch wenn diese Fehler schnell behoben werden können, sollte gerade der Datentyp nicht zu Problemen führen. Dies wirft aber die Frage nach der Interpretation ein und derselben Information auf. Es geht lediglich um ein unterschiedliches Mapping von IFCInteger. An dieser Stelle sei vermerkt, dass auch Revit und Solibri Integer auf xs:int beschränken, anders als dies im IFC-Standard vorgesehen ist.

Dieses Beispiel verdeutlicht, wie schnell die Verwendung unterschiedlicher Datenformate zu Problemen führen kann. Der Schluss liegt also nahe, der Nutzung einheitlicher Standards und Austauschformate den Vorzug vor der Verwendung lösungsspezifischer Formate zu geben. Dies gilt umso mehr, als mit mvdXML (1.5.2) ein offener Regelstandard vorliegt.

Da die Regeln in Solibri „neu erstellt“ werden müssen, werden entsprechende Interpretationsabweichungen in den Prüfregelein nicht auftreten. Allerdings erhöht sich der Aufwand in der Prüfregerstellung und die Frage stellt sich, warum Solibri in der Regelerstellung einen eigenen Weg eingeschlagen hat, anstatt den offenen Standard mvdXML zu verwenden. Dies gilt insbesondere, da Solibri auf den IFC-Standard ausgelegt ist. Auch ist unklar, warum Solibri Integer auf xs:int reduziert.

Im Falle von BIMcollab ZOOM kann die Besonderheit, dass zur Überprüfung von diskreten Wertebereichen und Datentypen für jedes Attribut ein eigener Smart View ausgeführt werden muss, zumindest aufwandstechnisch hinterfragt werden.

In der nachfolgenden Tabelle werden die Ergebnisse einmal vergleichend dargestellt.

Tabelle 7: Bewertung der Model Checker

Softwarelösung/Prüfung	Solibri	Desite md Pro BIMcollab ZOOM
Vorhandensein von Attributen	Green	Green
Datentypen von Attributen	Green	Yellow
Wertebereiche von Attributen	Green	Green
BIMQ-Anforderungen werden genutzt	Yellow	Green
Erhält Relationen	Yellow	Red
Für Model Content Checking geeignet	Green	Green
Proprietärer Datentyp ist nah am IFC	Yellow	Yellow

Legende	
Farbe	Erfüllungsgrad
Green	voll
Yellow	eingeschränkt
Red	kaum/ gar nicht

Im dritten Brückenmodell wird darüber hinaus mittels der Regelprüfung für die **Feste-Fahrbahn** gezeigt, dass die reduzierte Auswahl an Regeln, welche individuell für die Testmodelle getroffen wurde, für die Prüfung ausreicht. Ursache hierfür ist die Regelstruktur: Die Regeln greifen nach dem RASE-Prinzip erst, wenn bestimmte Voraussetzungen (Applicability, Select, not Exception) erfüllt sind. Weitere Prüfregeln für zusätzliche Modellkomponententypen wären nicht ausgeführt worden, denn es waren keine Modellkomponenten vorhanden, welche von diesen Regeln betroffen wären.

Insgesamt wird deutlich, dass eine Kombination aus BIMQ und den vorgestellten Model Checkern (Solibri, Desite oder BIMcollab ZOOM) geeignet ist die Erstellung von Modellanforderungen und Prüfregeln sowie einer anschließenden Modellprüfung entsprechend der Anforderungen an das Model Content Checking, wie es im Rahmen dieser Arbeit definiert wurde, zu voll erfüllen.

Der letzte Aspekt aus Tabelle 7 wird an dieser Stelle kurz erläutert. Er soll verdeutlichen, dass keiner der Model Checker den IFC-Standard im vollen Umfang wiedergibt. Jeder von ihnen übersetzt die IFC-Modelle beim Import in ein proprietäres Datenformat. Deutlich wurde dies bisher nur für die unterschiedlichen Interpretationen des IFCInteger. Es gibt jedoch weitere Auswirkungen. Insbesondere Relationen werden durch die Softwarelösungen nur eingeschränkt wiedergegeben und zum Teil grob übersetzt. So werden sie in Desite in teilweise fälschliche Vererbungen umgewandelt. In Solibri und BIMcollab ZOOM werden Relationen zwar eingeschränkt für die Struktur erhalten, können jedoch nicht eingesehen oder überprüft werden.

5 VALIDATION CHECKING

Im vorangegangenen Kapitel wurde ein BMC im Sinne einer „Model Content Prüfung“ nach 2.1.5 durchgeführt. Die BIM-Modelle wurden auf das Vorhandensein aller Attribute sowie die Korrektheit der Datentypen geprüft. Teilweise erfolgten darüber hinaus auch Prüfungen auf die Einhaltung gewisser diskreter Wertemengen. Hat ein Modell dieses Content Checking erfolgreich abgeschlossen, so ist sichergestellt, dass alle relevanten Informationen vorhanden sind. In darauffolgenden Schritt kann dann die Validierung erfolgen.

In 2.1.5 wurde Validation Checking als die Prüfung aller Modellanforderungen eingeführt, welche über das bloße Vorhandensein hinausgehen, sofern diese Aspekte nicht Teil des Modellierungsprozesses sind (siehe 2.1.4). Es gilt nunmehr festzulegen, wie diese weiteren Anforderungen korrekt aussehen und was Validierung in diesem Kontext bedeutet.

Die Möglichkeiten zur Validierung sind äußerst vielfältig und ihre vollständige Darstellung würde den Rahmen der vorliegenden Arbeit sprengen. Deshalb seien hier nur einige wesentliche Aspekte aufgeführt, welche innerhalb eines Projektes validiert werden.

Tabelle 8: Auswahl von Aspekten der Model Validation

Geometrisch	Semantisch	Sonstige
Clash Detection Platzierung Normen und Richtlinien	Ist die Attribuierung sinnvoll Wertebereiche Normen und Richtlinien Strukturelle Integrität Nachweisführung	Gestalterische Prüfung Rechtliche Randbedingungen

Geometrisch:

- **Clash Detection:** Das Ziel der Clash Detection ist zu prüfen, ob sich die Modellkomponenten geometrisch überschneiden. Ist dies nicht der Fall oder beschränken sich die Überschneidungen auf ein den Anforderungen entsprechendes Toleranzniveau, so ist die Validierung erfolgreich.
- **Platzierung:** Hierbei wird sichergestellt, dass die Modellkomponenten richtig platziert sind. Beispielsweise müssen sich die Teilkomponenten eines Widerlagers an den richtigen Positionen innerhalb der Geometrie des Widerlagers befinden
- **Normen und Richtlinien:** Es muss gewährleistet sein, dass die Normen und Richtlinien eingehalten werden. Hierunter fallen unter anderem Mindestabstände, Grenzwerte der Abmessungen oder das Freihalten von Sichtlinien und Lichträumen.

Semantisch:

- **Sinnhaftigkeit der Attribuierung:** Die Attribuierung muss sinnvoll erfolgen. Nur weil durch das Model Content Checking sichergestellt wurde, dass ein Attribut vorhanden ist, heißt das nicht, dass es mit einem sinnvollen Wert attribuiert wurde. Beispielsweise sollte einem Betonbauteil als Material Beton zugewiesen sein, nicht aber Stahl oder Glas.
- **Wertebereiche:** Die Attribuierung soll innerhalb der geforderten Wertebereiche erfolgen. Wenn ein Beton der Klasse C30/37 gefordert ist, sollte dieser auch der Modellkomponente zugewiesen sein. In diese Kategorie fallen auch Informationen wie vorgegebene Klassifikationsmöglichkeiten und die Nichtüberschreitung von Extremwerten.
- **Normen und Richtlinien:** Die Modellkomponenten sollen gültige Normen und angesetzte Richtlinien einhalten wie beispielsweise Festigkeitsklassen, Mindestbewehrungen oder Umweltschutzanforderungen.
- **Strukturelle Integrität:** Die Struktur der BIM-Modelle soll korrekt anhand der gegebenen RRR erfolgen. Hierunter fallen beispielsweise:
 - Die korrekte Unterteilung der BIM-Modelle in die Fachmodelle.
 - Eine anforderungskonforme Gliederung der Modelle wie die Korrektheit von Aggregationen. Bspw. ist ein Widerlager Teil des Unterbaus einer Brücke und nicht Teil der Fahrbahn. Noch weiter gedacht: Ist das Widerlager innerhalb der Modellstruktur dem Unterbau der richtigen Brücke zugewiesen?
 - Sind die Modellkomponenten untereinander richtig verknüpft? So sollten z.B. die richtigen Messgeräte den richtigen Signalen zugeordnet sein.
 - Ferner muss die anforderungskonforme Zuweisung von Informationen zu den Modellkomponenten sichergestellt werden. Erfolgt dies über einzelne PropertySets oder gebündelt über Typenzuweisungen?
 - **Nachweisführung:** Die Modellkomponenten müssen unter anderem die Nachweise der Tragsicherheit und Gebrauchstauglichkeit erfüllen. Es wird hier allerdings nochmals betont, dass die BIM-Methodik für die Nachweisführung durch die BIM4INFRA noch nicht empfohlen wird [46].
- Ferner können bauplanerische Gesichtspunkte wie Bauablaufplanungs- und zeitliche Planungsaspekte anhand der BIM-Modelle validiert werden.

Darüber hinaus gibt es noch weitere Bereiche, die nicht sofort mit BMC in Verbindung gebracht werden, aber dennoch zu validieren sind. So erhält Bauwerke möglicherweise nur eine Genehmigung, wenn sie vorgegebene gestalterische und rechtliche Anforderungen erfüllen, welche auch im BIM-Modell hinterlegt sein können. Als Beispiel sei der Fall erwähnt ein Bauwerk sich in das Umgebungsgesamtbild eingliedern muss. Dies könnte die Formen von Dachgauben oder die Farbe von Dachschindeln einschränken.

Rechtliche Aspekte sind nicht zwangsläufig Teil des BMC, da das Modell nicht mit dem Projekt gleichzusetzen ist. Dennoch könnten derartige Informationen und Regeln im Modell hinterlegt sein und daher für eine Prüfung zur Verfügung stehen.

5.1. AUSWAHL DER VALIDIERUNGSASPEKTE

Wie bereits beschrieben kann in dieser Arbeit keine umfassende Darstellung aller Aspekte eine Model Validierung erreicht werden. Ziel ist es vielmehr, einen Einblick in die Validierung zu geben. Dementsprechend beschränkt sich diese Arbeit darauf eine Grundlage für die Modellvalidierungen zu erarbeiten. Dies erfolgt über die Entwicklung einer Grundlage zur Prüfung und Validierungen mithilfe von Relationen.

Relationen sind ein grundlegender Bestandteil des IFC-Standards. Mittels Relationen erfolgt die Attribuierung und Verknüpfung von Modellkomponenten, die Gliederung des Modells, die Platzierung von Modellkomponenten sowie die Verknüpfung von semantischen Daten und geometrischen Komponenten. Entsprechend können anhand von Relationen viele der zuvor beschriebenen Modellanforderungen geprüft werden.

Trotz ihres grundlegenden Charakters sind Relationen in einigen der proprietären Datenformate der bekannten Model Checking Softwarelösungen (Desite, BIMcollab ZOOM, Bexel) nicht mehr originär vorhanden, sondern werden in andere Entitäten transformiert. In Desite werden die Aggregationen und Strukturgliederungen sogar fälschlicherweise in Vererbungen überführt. Wie in 3.3.4 bereits diskutiert wurde, können Relationen in Softwarelösungen wie BIMQ nicht direkt modelliert werden. Die Aggregationsinformationen in der auf BIMQ ausgerichteten Neustrukturierung des SOM, konnten teilweise dadurch erhalten werden, das den Modellkomponenten *id*-Attribute der ihnen Übergeordneten Modellkomponenten zugewiesen wurden (siehe 3.3.5). Es bleiben jedoch nur die Informationen der unteren Hierarchieebenen erhalten.

Da viele Model Checker die Prüfung von Relationen trotz ihrer offensichtlichen Bedeutung nicht unterstützen, besteht die Möglichkeit, die Prüfung mittels mvdXML (1.5.2) durchzuführen, dem von buildingSMART empfohlenen offenen Standard für BMC. Im Rahmen dieser Arbeit werden nun einige Relationen ausgewählt, für welche mvdXML-Vorlagen zur Modellprüfung erstellt werden.

Im Abschluss dieses Kapitels (5.3.4) wird ein vollständiger BMC-Ablauf beschrieben. Anhand einer reduzierten Form des SOM werden Anforderungen an ein Modell erstellt. Diese werden mittels der RASE-Methodik in Prüfregele überführt welche mit Hilfe der nachfolgend erarbeiteten Templates in eine mvdXML-Datei transformiert werden. Abschließend wird das Testmodell einem Model Content Checking und einer Validation unterworfen.

5.2. DIE AUSGEWÄHLTEN RELATIONEN

Die Auswahl der zu untersuchenden Relationen erfolgte im Wesentlichen anhand zweier Aspekte: Zum einen sollten die Relationen eine große praktische Relevanz haben, zum anderen auch gut modellierbar sein. Der letztere Gesichtspunkt wird vor allem durch die Modellierungstools beeinflusst. Die am weitesten verbreitete BIM-Modellierungslösung ist Revit und so bot es sich an, sich bei der Festlegung der zu untersuchenden Relationen an der Funktionalität von Revit zu orientieren. Es ergaben sich damit allerdings auch Einschränkungen, denn aus Revit exportierte IFC-Modelle weisen meist nur die folgenden drei Relationen auf:

- ***IfcRelDefinesByProperties* / *IfcRelDefinesByType***
- ***IfcRelAssociatesMaterial***
- ***IfcRelContainedInSpatialStructure***

Da die letzte der drei Relationen immer nur das Stockwerk zuweist und die betrachteten Modelle als Infrastrukturprojekte immer aus einem Stockwerk bestehen, wurde diese Relation nicht weiter untersucht.

IfcRelDefinesByProperties

IfcRelDefinesByProperties ist eine der grundlegendsten Relationen des IFC-Standards, da den Entitäten mittels dieser Relation die PropertySets zugewiesen werden. Eine Alternative hierzu ist die Zuweisung über ***IfcRelDefinesByType***. Um die Prüfregeleln jedoch in einem angemessenen Umfang zu halten, beschränkt sich diese Arbeit auf ***IfcRelDefinesByProperties***. Die Relation ist für alle Kindklassen von ***IfcObject*** möglich und erfolgt über das Attribut *isDefinedBy*. Der Relation können nun PropertySets (sowohl vorgefertigte als auch individuelle) oder QuantitySets zugeordnet werden. Die Auswahl der möglichen Properties unterliegt keinen Einschränkungen. Sie können individuell zugewiesen werden und in der Theorie auch beschränkter Natur sein oder mit ***IfcConstraints*** verknüpft werden. Die Struktur der Zuordnung eines PropertySet über die Relation ist in Abbildung 48 dargestellt.

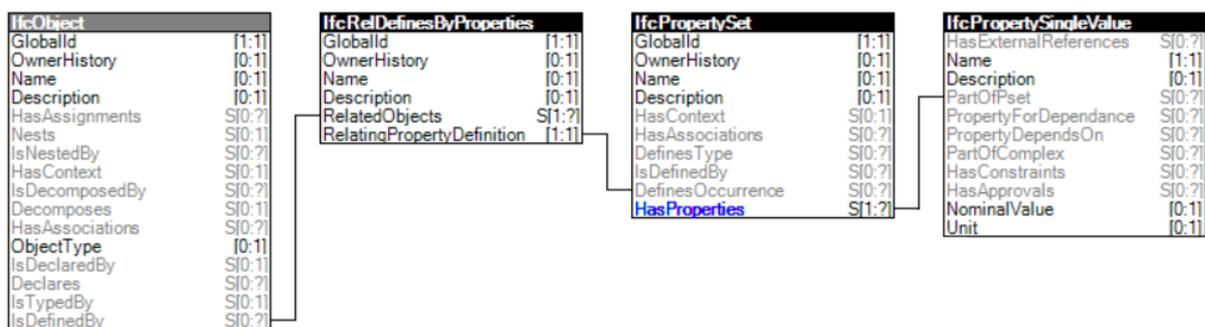


Abbildung 48: *IfcRelDefinesByProperties* Struktur [50]

Die vollständige Attribuierung wird zwar bereits durch die Attributprüfungen im Model Content Checking abgedeckt. Dies erfolgte aber bezogen auf die entsprechenden Datenformate der Prüftools und nicht bezogen auf die IFC-Struktur über Relationen. Wie bereits mehrfach erwähnt wurde, werden die IFC-Strukturen und insbesondere die Relationen in den Prüftools teilweise stark reduziert oder ganz aufgelöst. Daher ist eine Validierung der IFC-konformen Modellstruktur sinnvoll. Zusätzlich ist diese Relation in nahezu jedem Modell vorhanden und entsprechend bietet sie einen idealen Einstieg in die mvdXML Strukturen von Relationsprüfungen.

IfcRelAssociatesMaterial

So grundlegend Materialien für die AEC-Industrie sind, so vielfältig sind die Zuweisungen von Materialien und Modellkomponenten. Materialien sind wesentlich mehr als nur ein Name in Form eines Textes. Je nach Softwarelösung ist man unterschiedlichen Beschränkungen unterworfen. In 3.3 mussten Materialien auf die Textbezeichnung reduziert werden, da eine komplexere Propertystruktur hier nicht möglich war. Im IFC-Standard und den meisten Formaten der BIM-Softwarelösungen werden Materialien korrekterweise als eigene Entitäten angelegt. Dadurch ist es möglich, Materialien detailliert mit Properties zu versehen und sie unter anderem bezüglich ihrer physikalischen Eigenschaften zu präzisieren.

Wenn Materialien wie eigene Entitäten abgebildet werden, erfolgt die Zuweisung zu einer anderen Entität über eine Relation. Damit lässt sich auf sehr natürliche Art abbilden, dass ein Material mehreren verschiedenen Objekten zugewiesen werden kann. Wie bei PropertySets kann die Zuweisung entweder über den Typen erfolgen oder alternativ über eine direkte Relation, die in dieser Arbeit genauer betrachtete Relation **IfcRelAssociatesMaterial**.

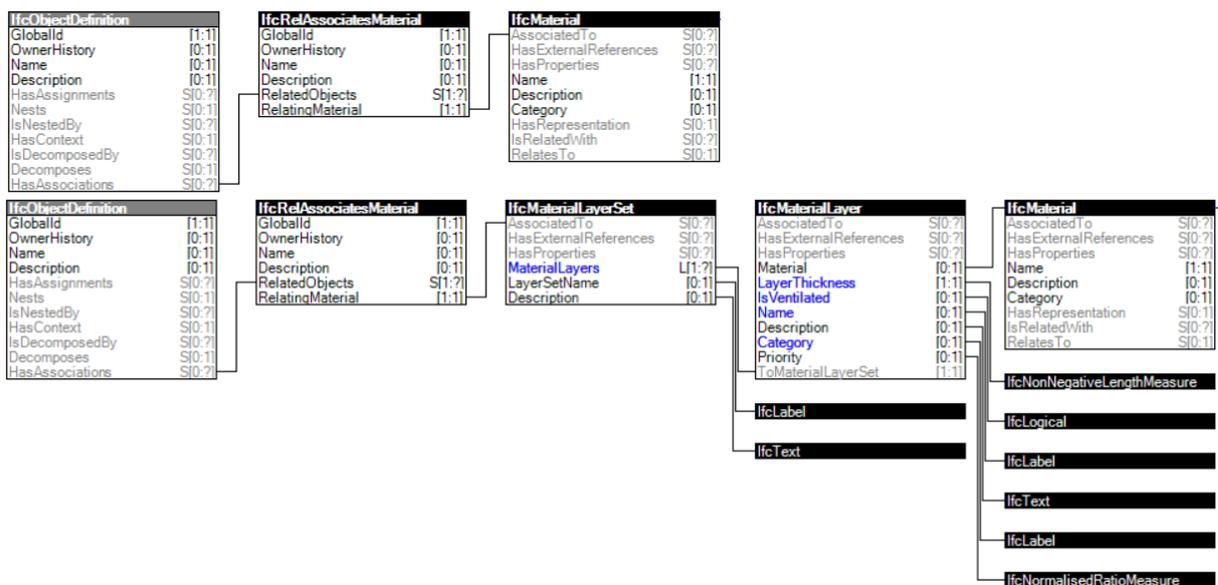


Abbildung 49: IfcRelAssociatesMaterial für ein Material (oben) und Materialschichten (unten)

Wie in Abbildung 49 dargestellt, erfolgt die Zuweisung über das Attribut *hasAssociation*. Zugleich kann mehr als nur ein Material zugewiesen werden. Unter anderem können, wie in Abbildung 49 dargestellt, ganze Schichtsysteme inklusive der Schichtabmessungen zugewiesen werden.

Materialien können in jedem BIM-Modellierungstool zugewiesen werden. Daher ist diese Relation in softwarespezifischer Form in den Modellierungstools enthalten und wird beim Export in IFC-Dateien geeignet übersetzt. Wie die Überprüfung der Modelle zeigte, ist diese Relation in vielen IFC-Modellen vorhanden und kann entsprechend geprüft und validiert werden.

IfcRelAggregates

Da insbesondere die Aggregationsstruktur der BIM-Modelle im SOM und auch im überarbeiteten SOM von großer Bedeutung ist, wird ferner die zugehörige Relation untersucht.

IfcRelAggregates ist die IFC-Version der Aggregation und erlaubt es, die Struktur von Modellen und Modellkomponenten abzubilden. Die Relation **IfcRelAggregates** kann aus zwei unterschiedlichen Blickwinkeln betrachtet werden.

Ein Beispiel zu diesem Sachverhalt: Zum einen besteht ein Widerlager aus der Widerlagerwand, den Flügelwänden und dem Fundament. Gleichzeitig bilden diese Komponenten zusammen ein Widerlager. Die Relation kann also von beiden her Seiten verstanden werden. Die Alternativen werden in Abbildung 50 beispielhaft dargestellt.

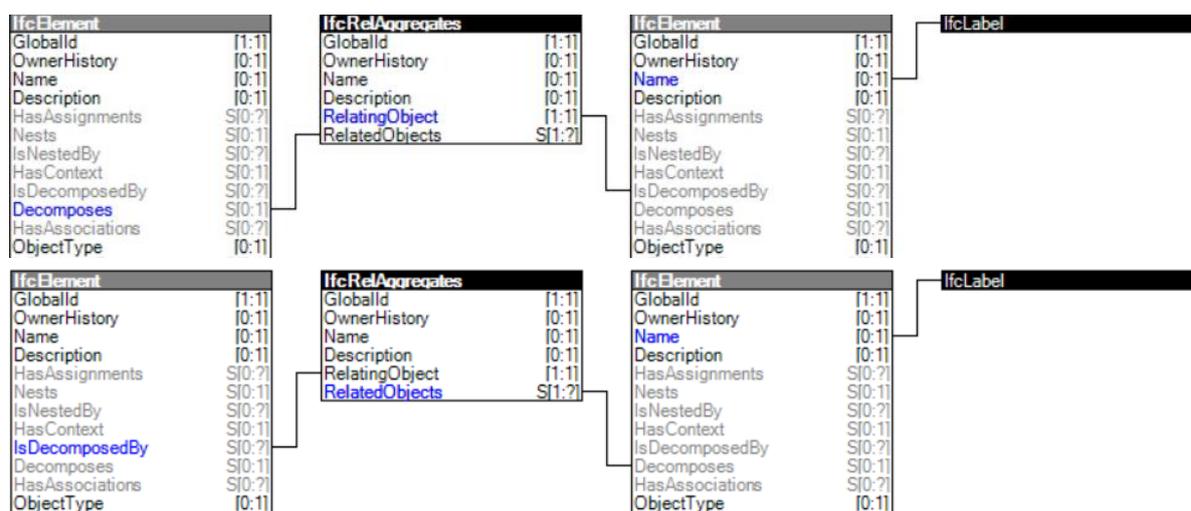


Abbildung 50: IfcRelAggregates von unten (oberes Bild) und oben (unteres Bild) in der Hierarchie [50]

Eine Aggregationsstruktur ist bis auf wenige Aspekte nicht eindeutig normativ festgelegt. Beispielsweise kann weder eine maximale Tiefe der Aggregationsstrukturen eindeutig bestimmt werden noch ist die Gliederung an sich eindeutig. Beides muss immer an die Erfordernisse in den Projekten individuell angepasst werden.

Eine Brücke bspw. besteht aus Überbau, Unterbau, der Gründung und der Ausrüstung. Der Unterbau besteht aus dem Widerlager und gegebenenfalls Stützen. Das Widerlager wiederum besteht aus den Widerlagerwänden, den Flügelwänden und dem Fundament. Dies kann beliebig fortgeführt werden, beispielsweise bis zum Beton, der aus Gestein und Zement besteht.

Im Gegensatz zu dieser Gliederung kann die Gründung aber auch als Teil des Unterbaus betrachtet werden und das Widerlagerfundament als Teil der Gründung. Die Ausrüstung kann wiederum auf den Unterbau und Überbau aufgeteilt werden. Dies ist nur eine mögliche alternative Struktur.

Da das SOM in seiner ursprünglichen Form einen Schwerpunkt auf die Aggregationsstruktur legt und auf dieser Basis die verschiedenen Detaillierungen eines Fachmodells gut ausgearbeitet werden können, wird im Rahmen dieser Arbeit auch eine Grundlage zur Prüfung hinsichtlich dieser Relation ausgearbeitet.

5.3. RELATIONSPRÜFUNG

Zunächst werden für eine Beispielvalidierung mvdXML-Dateien zur Prüfung der in 5.2 beschriebenen Relationen erstellt. Dabei wird die Erstellung von mvdXML-Dateien und deren Aufbau erläutert. Um die Modellprüfung und insbesondere die Modellvalidierung an einem kleinen Beispiel darzustellen, werden abschließend Anforderungen an ein Modell aufgestellt, ein Model Content Checking durchgeführt und das Modell anhand der Anforderungen validiert. Der Schwerpunkt der Anforderungen liegt hierbei auf Modellstrukturen, welche sich auf Relationen beschränken. Dazu werden die zuvor unabhängig voneinander erstellten mvdXML-Dateien zu einer einzigen zusammengeführt.

Zur Erstellung der mvdXML-Dateien wurde das Ifc Documentation Generator Tool (siehe 1.6.5) genutzt und die damit erzeugten Dateien manuell angepasst und erweitert. Ferner wurden in manchen Fällen die mvdXML-Exporte der BIMQ-Anforderungen als Ausgangsbasis für die mvdXML-Dateien genutzt. Die Prüfung der Modelle anhand der mvdXML-Dateien erfolgt mittels XBIM Explorer (1.6.7) und FZKViewer (1.6.6).

5.3.1. IFCRELDEFINESBYPROPERTIES

Zunächst wird die Relation ***IfcRelDefinesByProperties*** untersucht. Ziel ist es, in einem Modell zu prüfen, ob das PropertySet **Allgemeine_Eigenschaften** den Modellkomponenten anhand der besagten Relation zugewiesen wurde. Ferner wird geprüft, ob das Attribut *bauteilKlassifikation* Teil dieses PropertySets ist.

Ausgangspunkt der Erstellung der zugehörigen mvdXML ist der Ifc Documentation Generator. Am Beispiel dieser mvdXML werden ferner mehrere Arbeitsschritte beschrieben, welche beim Arbeiten mit dem Ifc Documentation Generator beachtet werden müssen.

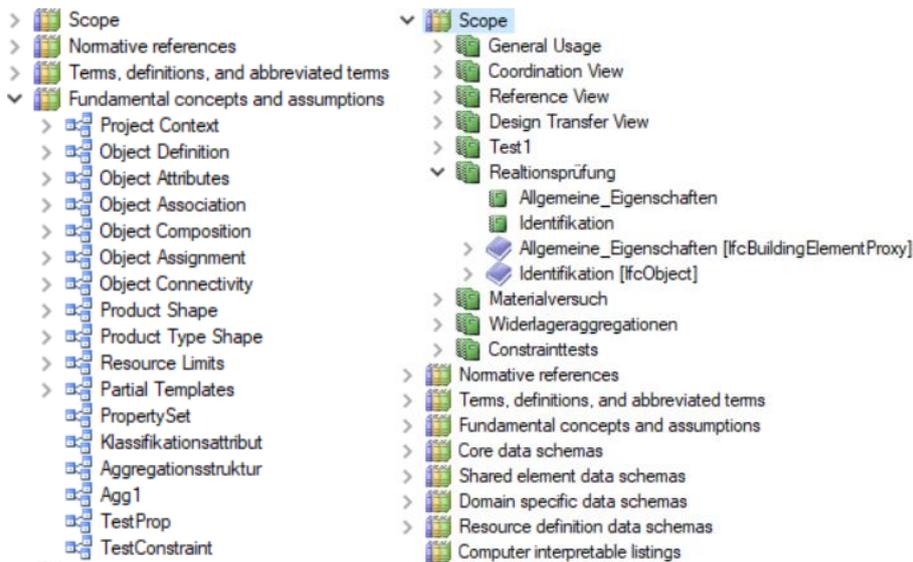


Abbildung 51: IFC Documentation Generator: Fundamental concepts and assumptions (links) und MVDs (rechts)

Der erste Schritt besteht in der Erstellung eines Templates für die Zuweisung eines PropertySets unter ‚Fundamental concepts and assumptions‘ wie es in Abbildung 51 (links) zu sehen ist. Dieses Template wird später in eine konkrete MVD importiert (siehe Abbildung 51 rechts) und dort mit konkreten Werten versehen.

Für das Template muss zunächst unter ‚Query‘ die Entität ausgewählt werden, auf welche das Template angewandt werden soll. Im gegebenen Fall trifft dies auf das **IfcObject** zu. Für jedes Attribut können die weiteren Verknüpfungen ausgewählt werden.

Maßgebende Attribute, welche „Regeln“ der späteren Prüfungen festlegen, müssen als ‚default‘ festgelegt werden. Abbildung 52 zeigt das Template der Propertyzuweisung. Die Festlegung des Attributs *Name* auf ‚default‘ des PropertySets wird durch die blaue Färbung gekennzeichnet.

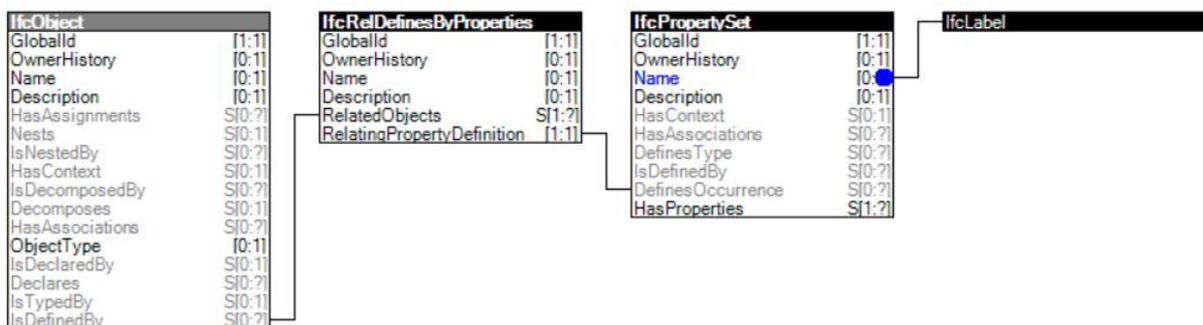


Abbildung 52: Schema der Propertyzuweisung

Für die Erstellung der mvdXML muss als nächster Schritt eine MVD unter ‚Scope‘ angelegt werden. Die MVD wird in diesem Beispiel „Realtionsprüfung“ genannt. Wie Abbildung 53 zeigt, besteht die MVD aus mehreren Unterkomponenten: dem als grüne Mappe

dargestellten ‚Exchange Requirement‘ und der als lila Buch dargestellten ‚Table Definition‘. Diesen sind in der Abbildung bereits konkrete Namen zugewiesen.

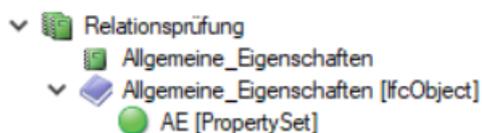


Abbildung 53: MVD Komponenten

Die Table Definition beschreibt die Auswahl der zu prüfenden Entitäten. Der Table Definition wird nun eine ‚Column Definition‘ (grüner Punkt) zugewiesen. Diese erlaubt es, den zu prüfenden Entitäten eigene Prüferegeln für die einzelnen Attribute zuzuweisen.

In die Column Definition wird anschließend unter ‚Parameter‘ das erstellte Template für die Attribute importiert. Dadurch können die im Template als ‚default‘ gekennzeichneten Attribute benannt werden. Im gegebenen Fall wird in Abbildung 54 der „Regel“ Name (zweite Spalte) als geforderter Wert der Name des PropertySets „Allgemeine_Eigenschaften“ zugewiesen.

Usage	Name	Description
Required	Allgemeine_Eigenschaften	

Abbildung 54: Regelwertzuweisung im IFC Documentation Generator

Unter ‚Requirements‘ muss abschließend noch der Column Definition das Exchange Requirement zugewiesen werden. Die derart erstellte MVD kann nun als mvdXML exportiert werden. Hat man den letzten Schritt nicht durchgeführt, ist die mvdXML leer. Auf die Exchange Requirements wird in den späteren Kapiteln noch eingegangen.

Die exportierte mvdXML besteht aus dem Bereich „Templates“ (Abbildung 55) und dem Bereich „View“ (Abbildung 56). Unter „Templates“ werden alle erstellten Templates aufgeführt und können im „View“ referenziert werden. Im gegebenen Fall kann unter Templates nur ein Template gefunden werden. Dieses beschreibt den Pfad vom **IfcObject** zum Attribut *Name* des PropertySets:

Das Attribut *IsDefinedBy* eines **IfcObject** soll verknüpft sein mit der Entität **IfcRelDefinesByProperties**. Das zugehörige Attribut *RelatingPropertyDefinition* dieser Entität soll wiederum verknüpft sein mit einem **IfcPropertySet**. Das Attribut *Name* des **IfcPropertySets** soll ein **IfcLabel** sein.

```

<ConceptTemplate uuid="12a40e69-a0be-47a6-807a-ef976566c1a8" name="PropertySet" status="sample" applicableSchema="IFC4" applicableEntity="IfcObject">
  <Rules>
    <AttributeRule RuleID="IsDefinedBy" AttributeName="IsDefinedBy">
      <EntityRules>
        <EntityRule EntityName="IfcRelDefinesByProperties">
          <AttributeRules>
            <AttributeRule RuleID="RelatingPropertyDefinition" AttributeName="RelatingPropertyDefinition">
              <EntityRules>
                <EntityRule EntityName="IfcPropertySet">
                  <AttributeRules>
                    <AttributeRule RuleID="Name" AttributeName="Name">
                      <EntityRules>
                        <EntityRule EntityName="IfcLabel" />
                      </EntityRules>
                    </AttributeRule>
                  </AttributeRules>
                </EntityRule>
              </EntityRules>
            </AttributeRule>
          </AttributeRules>
        </EntityRule>
      </EntityRules>
    </AttributeRule>
  </Rules>
</ConceptTemplate>

```

Abbildung 55: mvdXML Allgemeine_Eigenschaften Templates

Im „View“ (Abbildung 56) werden die konkreten Anforderungen festgelegt. Die Exchange Requirements bieten eine Möglichkeit zur Gliederung der Prüfungen. Im Falle der Materialprüfung und Aggregationen wird dies noch ausführlicher beschrieben.

Die „ConceptsRoots“ bündeln die einzelnen „Concepts“. Diese referenzieren die zuvor aufgelisteten Templates und versehen sie mit konkreten Anforderungen. Die „ConceptRoots“ können ferner auch noch mit einer „Applicability“ konkretisiert werden. Dies wird in 5.3.3 detailliert beschrieben. Für das gegebene Beispiel legt die „ConceptRoot“ fest, dass sämtliche **IfcObjects** (applicableRootEntity = „IfcObject“) und entsprechende Kindklassen eines Modells geprüft werden. Im Concept wird das Template referenziert und es wird festgelegt, dass alle relevanten Attribute der Kette vom **IfcObject** zum PropertySet vorhanden sein müssen ([Exists]=TRUE). Zusätzlich wird explizit gefordert, dass der Regel „Name“ der Wert „Allgemeine_Eigenschaften“ zugewiesen werden muss. Dementsprechend ist die Prüfung eines **IfcBuildingElementProxys** erfolgreich, sofern ihm ein PropertySet mit Namen „Allgemeine_Eigenschaften“ zugewiesen ist.

```

<Views>
  <ModelView uuid="182f7189-c63f-4dce-98c8-c02be69df025" name="Relationsprüfung" status="sample" applicableSchema="IFC4">
    <ExchangeRequirements>
      <ExchangeRequirement uuid="f053dc5c-58f0-4f0b-8eae-71d60287de71" name="Allgemeine_Eigenschaften" status="sample" applicability="both" />
    </ExchangeRequirements>
    <Roots>
      <ConceptRoot uuid="425e3c19-7e9c-4ba6-86d4-6cb3bfcc2909" name="Allgemeine_Eigenschaften" status="sample" applicableRootEntity="IfcObject">
        <Applicability uuid="00000000-0000-0000-0000-000000000000" status="sample">
          <Template ref="6cb91ac0-32eb-4d68-a905-c4b4d7dfb1ce" />
          <TemplateRules operator="and" />
        </Applicability>
        <Concepts>
          <Concept uuid="37a49301-59ef-4ff3-a3a0-8c1da940a6da" name="AE" status="sample" override="false">
            <Template ref="12a40e69-a0be-47a6-807a-ef976566c1a8" />
            <Requirements>
              <Requirement applicability="export" requirement="mandatory" exchangeRequirement="f053dc5c-58f0-4f0b-8eae-71d60287de71" />
            </Requirements>
            <TemplateRules operator="and">
              <TemplateRule Parameters="IsDefinedBy[Exists]=TRUE AND RelatingPropertyDefinition[Exists]=TRUE AND Name[Value]='Allgemeine_Eigenschaften'" />
            </TemplateRules>
          </Concept>
        </Concepts>
      </ConceptRoot>
    </Roots>
  </ModelView>
</Views>

```

Abbildung 56: mvdXML Allgemeine_Eigenschaften View

Das Resultat der Prüfung des Brückenmodells 1 im xBIM Xplorer (1.6.7) anhand dieser mvdXML Datei ist auszugsweise in Abbildung 57 dargestellt. Der Achse und einem

einzelnen Bauteil ist das PropertySet **Allgemeine_Eigenschaften** nicht zugewiesen. Allen weiteren Modellkomponenten ist hingegen ein PropertySet mit dem geforderten Namen entsprechend der beschriebenen Anforderung zugewiesen worden. Dies stimmt mit der Modellierung überein und validiert, dass die Attributzuweisung entsprechend der geforderten Relationsstruktur erfolgt.

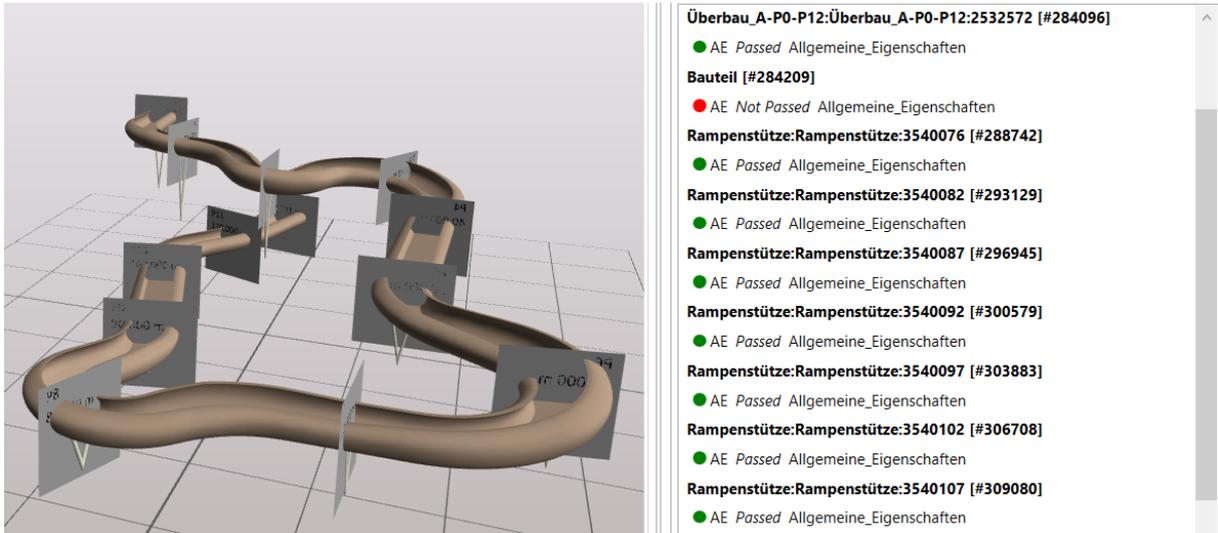


Abbildung 57: Auszug der Prüfung des Brückenmodells 1

Nun gilt es, die mvdXML-Anforderungen detaillierter zu gestalten. Man könnte beispielsweise fordern, dass die konkrete Instanz der **IfcRelDefinesByProperties** einen gewünschten Namen hat oder dass dem **IfcPropertySet** eine bestimmte Beschreibung zugewiesen ist. Um derartige Überlegungen nachzugehen, wurde das erstellte Template im Nachfolgenden erweitert. Das Ziel ist zu prüfen, ob den Modellkomponenten das Attribut *bauteil-Klassifikation* als Teil des PropertySets **Allgemeine_Eigenschaften** zugewiesen wurde.

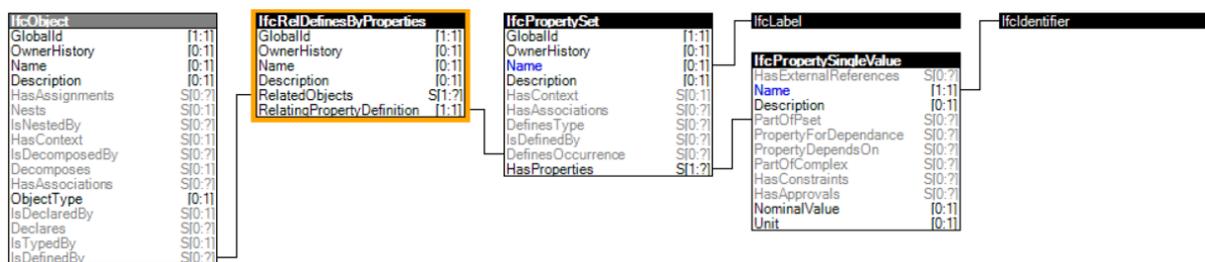


Abbildung 58: Schema der Bauteilklassifizierung

Es wurde ein passendes Schema im „Ifc Documentation Generator“ erstellt. Dieses ist in Abbildung 58 visualisiert. Wie deutlich wird, ist der **IfcPropertySingleValue** Teil des PropertySets. Dementsprechend sollte dieser und die Festlegung als **IfcIdentifier** in der mvdXML gestaffelt eingefügt werden. Da dies nicht der Fall war, musste die mvdXML angepasst werden. Auch musste die konkrete Benennung der Properties im „View“ in einer Regel zusammengefasst werden. Diese wurden im Export in zwei getrennte Regeln exportiert.

Das Resultat ist das Template aus Abbildung 59. Der Pfad lautet: Ein *IfcObject* verfügt über das Attribut *IsDefinedBy*. Dieses verweist auf *IfcRelDefinesByProperties*. Das Attribut *RelatingPropertyDefinition* der Relation verweist auf ein *IfcPropertySet* mit dem Attribut *Name* (RuleID = PsetName) vom Typ *IfcLabel*. Ferner ist das Attribut *HasProperties* (RuleID = HasProperties) des PropertySets verknüpft mit einem *IfcPropertySingleValue*. Dessen Attribut *Name* (RuleID = AttributName) ist vom Typ *IfcIdentifier*.

```
<ConceptTemplate uuid="69cb72cb-fd55-4e49-bef4-14ca32cd568b" name="Klassifikationsattribut" applicableSchema="IPC4" applicableEntity="IfcObject">
  <Rules>
    <AttributeRule RuleID="IsDefinedBy" AttributeName="IsDefinedBy">
      <EntityRules>
        <EntityRule EntityName="IfcRelDefinesByProperties">
          <AttributeRules>
            <AttributeRule RuleID="RelatingPropertyDefinition" AttributeName="RelatingPropertyDefinition">
              <EntityRules>
                <EntityRule EntityName="IfcPropertySet">
                  <AttributeRules>
                    <AttributeRule RuleID="PsetName" AttributeName="Name">
                      <EntityRules>
                        <EntityRule EntityName="IfcLabel" />
                      </EntityRules>
                    </AttributeRule>
                    <AttributeRule RuleID="HasProperties" AttributeName="HasProperties">
                      <EntityRules>
                        <EntityRule EntityName="IfcPropertySingleValue">
                          <AttributeRules>
                            <AttributeRule RuleID="AttributName" AttributeName="Name">
                              <EntityRules>
                                <EntityRule EntityName="IfcIdentifier" />
                              </EntityRules>
                            </AttributeRule>
                          </AttributeRules>
                        </EntityRule>
                      </EntityRules>
                    </AttributeRule>
                  </AttributeRules>
                </EntityRule>
              </EntityRules>
            </AttributeRule>
          </AttributeRules>
        </EntityRule>
      </EntityRules>
    </AttributeRule>
  </Rules>
</ConceptTemplate>
```

Abbildung 59: Template für die Zuweisung einer bauteilKlassifikation

Im „View“ (Abbildung 60) werden den beiden Properties die vorgegebenen Bezeichnungen als eine TemplateRule zugewiesen.

```
<ConceptRoot uuid="befebec6-7f7a-4e17-802e-d02e7b4a198c" name="Klassifikation" status="sample" applicableRootEntity="IfcObject">
  <Applicability uuid="00000000-0000-0000-0000-000000000000" status="sample">
    <Template ref="7a5bdb56-fe4e-4561-9864-22e6979dec2e" />
    <TemplateRules operator="and" />
  </Applicability>
  <Concepts>
    <Concept uuid="8a1a5102-b88e-4ff9-ba11-aa75bb54d070" name="Klassifikation" status="sample" override="false">
      <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b" />
      <Requirements>
        <Requirement applicability="export" requirement="mandatory" exchangeRequirement="f70671a4-42e4-461a-83d1-6782b489e452" />
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="IsDefinedBy[Exists]=TRUE AND RelatingPropertyDefinition[Exists]=TRUE AND PsetName[Value]='Allgemeine_Eigenschaften' AND HasProperties[Exists]=TRUE AND AttributName[Value]='bauteilKlassifikation'" />
      </TemplateRules>
    </Concept>
  </Concepts>
</ConceptRoot>
```

Abbildung 60: View für die bauteilKlassifikation

Um sicherzustellen, dass die Anforderungen des richtigen PropertySets und das Vorhandensein des Attributs *bauteilKlassifikation* nicht getrennt, sondern als Einheit geprüft

werden, wurde im Brückenmodell 1 das Attribut **bauteilKlassifikation** einer der **Stuetzen** in das PropertySet **Bauteil** verschoben. Sofern die Regel richtig angewandt würde, müsste es folglich zu einer Fehlermeldung für diese **Stuetze** kommen. Die Ergebnisse bei der Durchführung der Prüfung im Xbim Xplorer (Abbildung 61) und im FZKViewer passten zu dieser Prognose. In beiden Softwarelösungen wird das Fehlen der **bauteilKlassifikation** im PropertySet **Allgemeine_Eigenschaften** bemängelt.

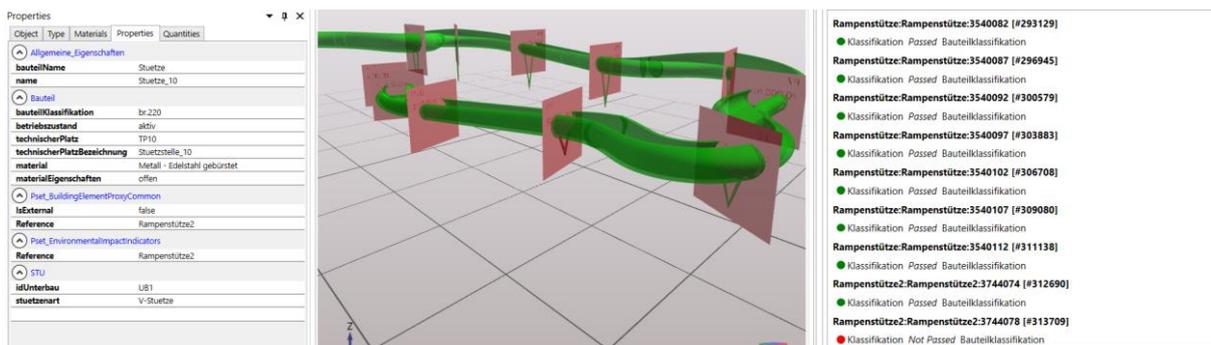


Abbildung 61: Auswertung der bauteilKlassifikations-Probe

Da der FZKViewer anders als der Xbim Xplorer einen Fehlerbericht ausgibt, kann der Misserfolg der Prüfregel genau identifiziert werden (Abbildung 62). Das PropertySet **Allgemeine_Eigenschaften** ist für die ausgewählte **Stuetze** zwar zugewiesen, aber das Attribut **bauteilKlassifikation** ist wie erwartet nicht in diesem zu finden.

Failed	Type: IfcBuildingElementProxy Name: Rampenstütze2:Rampenstütze2:3744078 OID: #313709
Successful	Check condition: PsetName [Value] == 'Allgemeine_Eigenschaften' (EXPRESS: IfcRoot.Name, Value: "Allgemeine_Eigenschaften")
Failed	Check condition: AttributName [Value] == 'bauteilKlassifikation' (EXPRESS: IfcProperty.Name, Value: NOT EXISTS)

Abbildung 62: Fehlermeldungen bei der mvdXML-Prüfung im FZKViewer

Wichtig ist es hier, darauf hinzuweisen, dass für den FZKViewer der View etwas gekürzt werden musste. Der FZKViewer ist nicht in der Lage, einen Wechsel in der Abfrage, ob ein Attribut existiert und ob ein Attribut einen Wert hat zu interpretieren. Entweder man prüft bei mehreren Abfragen nur nach „[Exists]“ oder nur nach „[Value]“. Dementsprechend wurde in der FZKViewer Abfrage der Verknüpfungskette im „View“ (Abbildung 60) abgeändert auf:

PsetName[Value]='Allgemeine_Eigenschaften' AND AttributName[Value]='bauteilKlassifikation'

Es wurden also alle [Exists]-Abfragen beseitigt. Sofern nichts anderes vermerkt ist, werden gleichartige Vereinfachungen bei der Nutzung des FZKViewers auch in den kommenden Kapiteln genutzt.

5.3.2. IFCREASSOCIATESMATERIAL

Die Untersuchung der **IfcRelAssociatesMaterial** erfolgt, wie zuvor für die **IfcRelDefinesByProperties**, in zwei Schritten. Die buildingSMART bietet Konzeptvorlagen für die Zuweisung von Materialien, welche auch im Ifc Documentation Generator vorimplementiert sind. Die Exporte dieser als mvdXML-Dateien sind jedoch nicht voll funktionsfähig. Daher muss die mvdXML-Datei ebenfalls modifiziert werden. Gleiches gilt für die Vorlagen aus BIMQ. Wie unter anderem in 5.2 bereits beschrieben, werden Materialien hierbei als eigene Objekte zugewiesen. Entsprechend wird in den Vorlagen auf ein Template für die Materialien referenziert, da die verwendeten Prüftools mehrere gekoppelte Referenzen zwischen Templates nicht korrekt interpretieren. Entsprechend müssen diese Referenzen aufgelöst werden.

Die mvdXML, welche in den Anhängen „mvdXML Materialzuweisungstemplate“ und „mvdXML Materialzuweisungsvue“ zu finden ist, besteht aus zwei Prüfungen in Form von zwei „Exchange Requirements“: „Materialzuordnung“ und „Stahlbeton“.

Für die beiden Prüfungen wurden innerhalb einer ConceptRoot für **IfcObjectDefinitions** zwei Concepts, „Material vorhanden“ und „Material der Kategorie Stahlbeton“, angelegt und diese jeweils einer der beiden „Exchange Requirements“ zugewiesen.

Zunächst wird überprüft, ob einem Objekt vom Typ **IfcObjectDefinition** über **IfcRelAssociatesMaterial** eine **IfcMaterial** zugewiesen wurde. Die Verknüpfungskette ist die folgende: Ein **IfcObjectDefinition** muss das Attribut *HasAssociation* haben. Dieses verweist auf **IfcRelAssociatesMaterial**. Dort wird das Attribut *RelatingMaterial* verknüpft mit einem **IfcMaterial**. Die Probe der mvdXML erfolgt anhand des Brückenmodells 2. Die Ergebnisse der Prüfung sind in Abbildung 63 dargestellt: Genau diejenigen Modellkomponenten, denen ein Material zugewiesen wurde, bestehen die Prüfung erfolgreich.

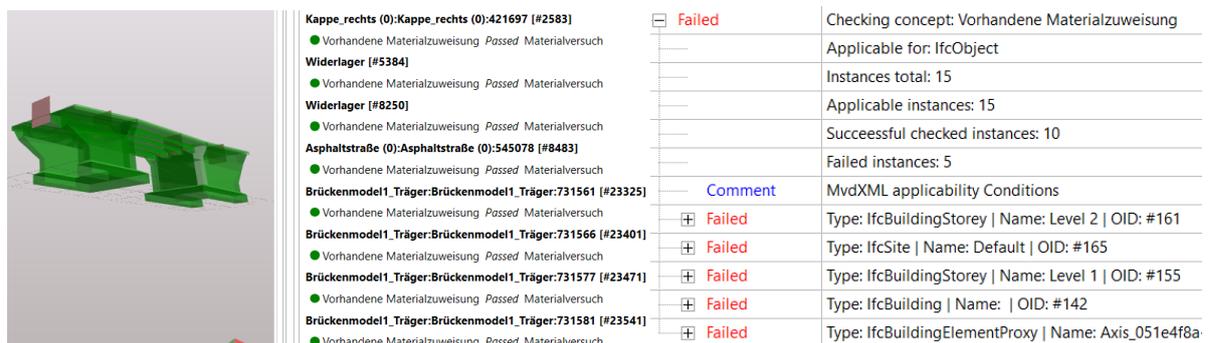


Abbildung 63: Auswertung der Probe auf Materialvorkommen im Xbim Explorer (l.) und im FZKViewer (r.)

Der zweite Schritt betrachtet die Struktur der **IfcMaterials** genauer. Ziel ist es eine Basis für eine Validierung der Materialien zu bieten. Dazu wird noch einmal der Aufbau eines **IfcMaterials** nach IFC-Schema und im Brückenmodell 2 betrachtet. Diese sind in Abbildung 64 dargestellt. Ein **IfcMaterial** verfügt über die drei Attribute *Name*, *Description* und *Category*. Diese können, wie dies für das Beispielmateriale geschieht, verschiedene Werte

annehmen. Im Rahmen der Versuche, eine mvdXML Prüfung für Materialien zu entwerfen, wurde entschieden, anhand des Concepts „Material der Kategorie Stahlbeton“ beispielhaft zu überprüfen, ob einer Modellkomponente ein Material der Kategorie Stahlbeton zugewiesen wurde.

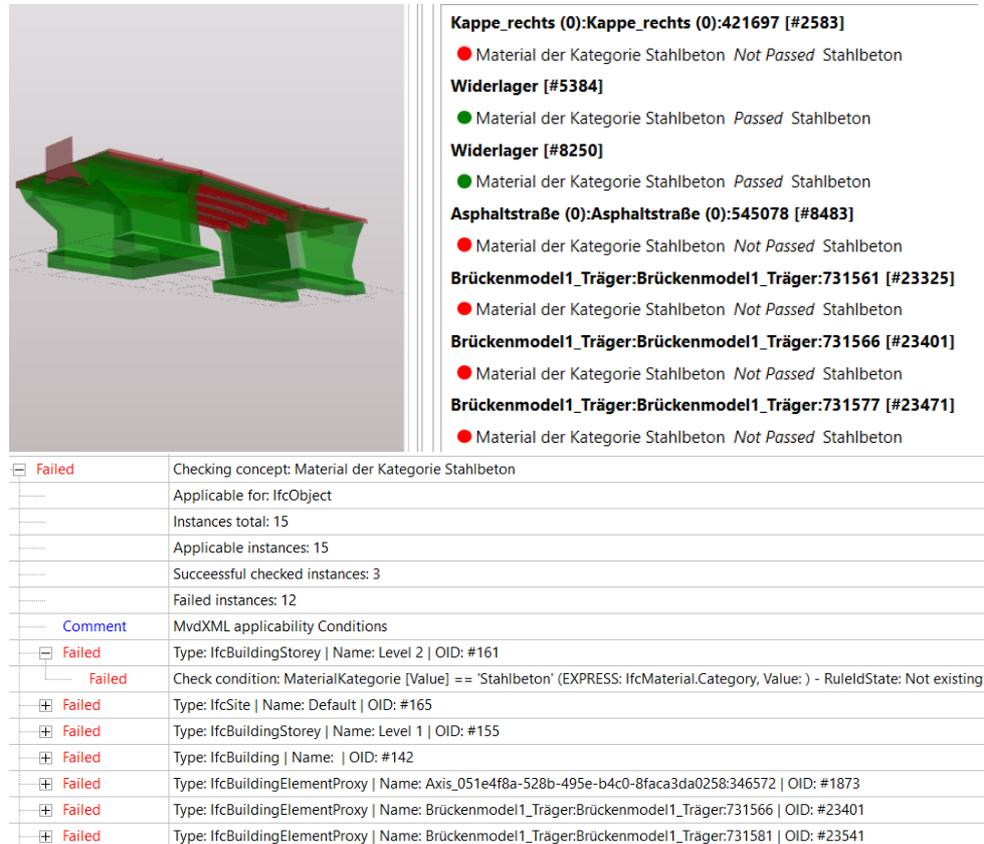
#	Attribute	Type
1	Name	IfcLabel
2	Description	IfcText
3	Category	IfcLabel

Ifc Label	#17964
Type	IfcMaterial
Name	C30/37
Description	Ortbeton
Category	Stahlbeton
AssociatedTo[0]	IfcRelAssociatesMaterial ('3KhwiXjilBERi\$Kxdg0saK' # 24520

Abbildung 64: IfcMaterial nach [54] (l.) und ein Material im Brückenmodell 2(r.)

Das Resultat dieser Untersuchung der Modellkomponenten kann in Abbildung 65 eingesehen werden. Den Widerlagern und der Brückenplatte wurden Materialien der entsprechenden Kategorie zugewiesen, den anderen Modellkomponenten hingegen nicht. Dies entsprach der Erwartung.

Wie noch gezeigt werden wird, können auf dieser Basis Prüfungen erstellt werden, welche einzelne Modellkomponenten bezüglich beliebig vieler Materialaspekte untersuchen.



Kappe_rechts (0):Kappe_rechts (0):421697 [#2583]
 ● Material der Kategorie Stahlbeton *Not Passed* Stahlbeton

Widerlager [#5384]
 ● Material der Kategorie Stahlbeton *Passed* Stahlbeton

Widerlager [#8250]
 ● Material der Kategorie Stahlbeton *Passed* Stahlbeton

Asphaltstraße (0):Asphaltstraße (0):545078 [#8483]
 ● Material der Kategorie Stahlbeton *Not Passed* Stahlbeton

Brückenmodell1_Träger:Brückenmodell1_Träger:731561 [#23325]
 ● Material der Kategorie Stahlbeton *Not Passed* Stahlbeton

Brückenmodell1_Träger:Brückenmodell1_Träger:731566 [#23401]
 ● Material der Kategorie Stahlbeton *Not Passed* Stahlbeton

Brückenmodell1_Träger:Brückenmodell1_Träger:731577 [#23471]
 ● Material der Kategorie Stahlbeton *Not Passed* Stahlbeton

Failed	Checking concept: Material der Kategorie Stahlbeton
—	Applicable for: IfcObject
—	Instances total: 15
—	Applicable instances: 15
—	Successful checked instances: 3
—	Failed instances: 12
—	Comment MvdXML applicability Conditions
⊖ Failed	Type: IfcBuildingStorey Name: Level 2 OID: #161
— Failed	Check condition: MaterialKategorie [Value] == 'Stahlbeton' (EXPRESS: IfcMaterial.Category, Value:) - RuleIdState: Not existing
⊕ Failed	Type: IfcSite Name: Default OID: #165
⊕ Failed	Type: IfcBuildingStorey Name: Level 1 OID: #155
⊕ Failed	Type: IfcBuilding Name: OID: #142
⊕ Failed	Type: IfcBuildingElementProxy Name: Axis_051e4f8a-528b-495e-b4c0-8faca3da0258:346572 OID: #1873
⊕ Failed	Type: IfcBuildingElementProxy Name: Brückenmodell1_Träger:Brückenmodell1_Träger:731566 OID: #23401
⊕ Failed	Type: IfcBuildingElementProxy Name: Brückenmodell1_Träger:Brückenmodell1_Träger:731581 OID: #23541

Abbildung 65: „Material der Kategorie Stahlbeton“ im Xbim Explorer und im FZKViewer

5.3.3. IFCRELAGGREGATES

Wie bereits diskutiert, beschreibt **IfcRelAggregates** die Aggregationsstrukturen eines Modells. Im Gegensatz zu den bereits beschriebenen Relationen ist diese Relation nur in wenigen Modellen zu finden (abgesehen davon, dass ein **IfcBuilding** aus Stockwerken besteht.). Stattdessen musste in einem ersten Schritt ein Testmodell erstellt werden, anhand dessen diese Relation untersucht werden konnte.

Hierfür wurde ein Brückenwiderlager gewählt. Dieses ist im SOM klar in seiner Aggregationsstruktur gegeben. Ein Widerlager ist Teil (*Decomposes*) eines **Unterbaus** und besteht (*IsDecomposedBy*) aus einer **Widerlagerwand**, **Fluegelwaenden**, einem **Widerlagerfundament** und gegebenenfalls einer **Widerlagerhinterfuellung**.

Entsprechend wurde ein Widerlager in Revit modelliert. Hierbei wurde für die Bestandteile des Widerlagers auf geteilte Komponenten zurückgegriffen, d.h. die Widerlagerkomponente setzt sich aus den Teilkomponenten zusammen, welche aber dennoch als zusätzliche Komponenten im Modell enthalten sind. Dementsprechend ist eine Aggregationsstruktur in Revit enthalten. Die beispielhafte Attribuierung der Widerlagerkomponenten ist in Abbildung 66 dargestellt.

identitaet	bauteilName	bauteilKlassifikation	KGK-Verzeichnisnummer	bemerkung	name	idWiderlager
TWF	Widerlager	br.210	TWF	TestWiderlager	Testwiderlager	
WF	Widerlagerfundament	br.213	WF	Widerlagerfundament	Testfundament	TWF
FWR	Fluegelwand_Rechts	br.212	FW	TestfluegelwandRechts	Testfluegelwand_rechts	TWF
FWL	Fluegelwand_Links	br.212	FW	TestfluegelwandLinks	Testfluegelwand_links	TWF
WW	Widerlagerwand	br.211	WW	Testwiderlagerwand	Testwiderlagerwand	TWF

Abbildung 66: Attribuierung der Widerlagerkomponenten in Revit

Überraschenderweise geht die Aggregation beim Export in eine IFC-Datei verloren. Genau genommen verschwindet das Widerlager als übergeordnete Modellkomponente und nur die Bestandteile bleiben erhalten. Drei **IfcRelAggregates** bleiben jedoch nach wie vor vorhanden: Das **IfcProject** besteht aus einem **IfcSite** und dieses aus einem **IfcBuildingStorey**. Dementsprechend ist Revit grundsätzlich in der Lage, die Relation darzustellen und zu exportieren. Warum diese aber für andere Komponenten verloren geht, konnte im Rahmen dieser Arbeit nicht geklärt werden.

Das Modell muss aufgrund dieser Problematik angepasst werden, indem ein Metaobjekt Widerlager eingeführt wird. Hierfür wird der IFC-STEP Code des Modells gemäß Abbildung 67 manuell erweitert. Es wird also ein **IfcBuildingElementProxy** eingeführt, welches über das Property *bauteilKlassifikation* des PropertySets **Allgemeine_Eigenschaften** und auch über das IFC-Pendant *ObjektType* eindeutig als Widerlager nach SOM gekennzeichnet ist, also mit dem Attributwert „br.ub.10“. Es gibt zwei Gründe für diese Dopplung. Der erste ist der Bezug auf die Vorschläge für Ifc-Attributpendants wie sie in 3.3.4 beschrieben

wurden. Es geht darum die Klassifikation der Bauteile eindeutig zu regeln und zu zeigen, dass es dafür mehrere Möglichkeiten gibt.

```
#4000= IFCBUILDINGELEMENTPROXY ('WL', #42, 'WL', $, 'br.210', #190, #435, '346569', .NOTDEFINED.);
#4010= IFCPROPERTYSINGLEVALUE ('bauteilKlassifikation', $, IFCTEXT ('br.210'), $);
#4011= IFCPROPERTYSINGLEVALUE ('bauteilName', $, IFCTEXT ('Widerlager'), $);
#4012= IFCPROPERTYSINGLEVALUE ('bemerkung', $, IFCTEXT ('Widerlager_Test'), $);
#4013= IFCPROPERTYSINGLEVALUE ('KGK-Verzeichnisnummer', $, IFCTEXT ('WL'), $);
#4014= IFCPROPERTYSINGLEVALUE ('name', $, IFCTEXT ('Testwiderlager'), $);
#4015= IFCPROPERTYSINGLEVALUE ('identitaet', $, IFCTEXT ('WL'), $);
#4020= IFCPROPERTYSET ('2MktZ0JZf9uxFEFWSLwRDE', #42, 'Allgemeine_Eigenschaften', '', (#4010, #4011, #4012, #4013, #4014, #4015));
#4021= IFCRELDEFINESBYPROPERTIES ('2XCRET6g19BhpuaSQAABd', #42, $, $, (#4000, #4020));
#4001= IFCRELAGGREGATES ('AGGTEST', $, $, $, #4000, (#3041, #1684, #2626, #626));
```

Abbildung 67: Metaobjekt Widerlager in IFC-STEP

Der zweite Grund liegt in der Annäherung der Aggregation, wie in den nachfolgenden Abschnitten noch erläutert wird. In gleicher Weise werden auch die Teilkomponenten des Widerlagers doppelt klassifiziert.

Anschließend muss eine **IfcRelAggregates** erstellt werden (#4001 in Abbildung 67), in welchem einem **IfcElement** mehrere **IfcElements** als Bestandteile zugewiesen werden. Es wird deutlich, dass eine einzige Zeile ausreicht, um beide Wirkrichtungen der **IfcRelAggregates** innerhalb des IFC-Modells zu erstellen.

Für die Aggregation wird zunächst ein Ansatz gewählt, welcher die Aggregation nach IFC-Standard in unveränderter Weise wiedergibt. Das zugehörige Template ist in „mvdXML Aggregation Templates“ dargestellt. Das Template legt fest, dass das Attribut *IsDecomposedBy* eines **IfcElement** mit **IfcRelAggregates** verknüpft ist. Dessen Attribut *RelatedObjects* ist verknüpft mit einem weiteren **IfcElement** und für dieses wird der Datentyp des Attributes *ObjectType* festgelegt.

```
<Rules>
  <AttributeRule RuleID="IsDecomposedBy" AttributeName="IsDecomposedBy">
    <EntityRules>
      <EntityRule EntityName="IfcRelAggregates">
        <AttributeRules>
          <AttributeRule RuleID="RelatedObjects" AttributeName="RelatedObjects">
            <EntityRules>
              <EntityRule EntityName="IfcElement">
                <AttributeRules>
                  <AttributeRule RuleID="ObjectType" AttributeName="ObjectType">
                    <EntityRules>
                      <EntityRule EntityName="IfcLabel"/>
                    </EntityRules>
                  </AttributeRule>
                </AttributeRules>
              </EntityRule>
            </EntityRules>
          </AttributeRule>
        </AttributeRules>
      </EntityRule>
    </EntityRules>
  </AttributeRule>
</Rules>
```

Abbildung 68: Template einfache Aggregation (Bestandteile ohne AE)

Bevor auf die „Concepts“ des „Views“ eingegangen werden, muss die grundlegende Problematik geklärt werden, wie man eine „ConceptRoot“ auf konkrete Modellkomponenten beschränkt und wie dies mit dem Konzept der RASE zusammenhängt.

Die vorherigen mvdXML-Dateien prüfen immer alle **IfcObjects** oder **IfcObjectDefinitions** eines Modells auf die korrekte Zuweisung ausgewählter Attribute. Es ergibt jedoch wenig Sinn, stets alle Modellkomponenten eines Typs auf alle möglichen Anforderungen zu prüfen, beispielsweise ob alle **IfcElements** über **IfcRelAggregates** eine Verknüpfung mit beliebigen **IfcElements** haben. Ebenso entsteht nur ein geringer Nutzen, wenn sämtliche **IfcElements** dahingehend geprüft werden, ob sie die Anforderungen an **Widerlager** erfüllen.

Die Lösung dieses Problems erfolgt im mvdXML-Format sehr konform zum Prinzip der RASE. Die schon erwähnte „ConceptRoot“ verfügt über den Aspekt „applicableRootEntity“, welche dem Select der RASE entspricht. Sie gibt für jede „ConceptRoot“ an, auf welche Entitäten und zugehörigen Kindentitäten die darin gebündelten „Concepts“ angewandt werden können. Eine darüber hinaus gehende Konkretisierung kann unter „Applicability“ erfolgen. Dies entspricht genau Applicability der RASE. In den vorangegangenen Relationen wurde hier jeweils die obere Variante aus Abbildung 69 genutzt. In dieser wird auf ein Template verwiesen und ein leerer Templateregelsatz aufgelistet, d.h. es gibt keine weiteren Einschränkungen. Es wurden im gegebenen Beispiel schlicht alle **IfcObjects** und Objekte der Kindklassen geprüft.

```
<ConceptRoot uuid="5875979d-2f09-4290-979b-d7ad341596f9" name="Materialzuweisung" status="sample" applicableRootEntity="IfcObject">
  <Applicability uuid="00000000-0000-0000-0000-000000000000" status="sample">
    <Template ref="7d4442be-15b3-4677-83d9-32bc23f05582" />
    <TemplateRules operator="and" />
  </Applicability>
</ConceptRoot>

<ConceptRoot uuid="00000036-0936-0936-0000-000002907057" name="br.ub.10 Widerlager" applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
      <TemplateRule Parameters="PsetName[Value]='Allgemeine_Eigenschaften' AND AttributeName[Value]='bauteilKlassifikation' AND Wert[Value]='br.ub.10'"/>
    </TemplateRules>
  </Applicability>
</ConceptRoot>
```

Abbildung 69: Die Spezifizierung der zu prüfenden Modellkomponenten minimal (o.) und präzise (u.)

Im unteren der dargestellten Beispiele wird eine Applicability präzisiert. Es werden konkrete Eignungskriterien festgelegt. Das Select („applicableRootEntity“ der „ConceptRoot“) wählt noch alle **IfcBuildingElements** aus. Zusätzlich wird anschließend jedoch noch festgelegt, dass nur jene **IfcBuildingElements** geeignet sind welche entsprechend dem referenzierten Template das PropertySet **Allgemeine_Eigenschaften** zugewiesen bekommen haben und dort über das Attribut *bauteilKlassifikation* mit dem Wert „br.ub.10“ belegt sind. Es wird also festgelegt, dass nur Modellkomponenten, welche die Kriterien eines Widerlagers nach SOM erfüllen, untersucht werden.

Die Concepts, welche die konkreten Wertbelegungen der Templates bündeln und den zugehörigen ConceptRoots mit ihren „Applicabilities“ untergeordnet sind, entsprechend den Requirements der RASE.

Die Belegung mit Werten im „View“ ist in Abbildung 70 dargestellt. Diese Regel wird der Applicability aus Abbildung 69 zugeordnet. Das bedeutet im konkreten Beispiel, dass die Überprüfung dieser Requirements nur auf Widerlager nach SOM angewandt wird. Für

diese werden drei Regeln geprüft. In jeder von diesen wird anhand des Templates aus Abbildung 68 geprüft, ob *IsDecomposedBy* existiert genauso wie *RelatedObject*. Ferner wird der Wert von *ObjectType* jeweils auf Übereinstimmung mit einer Wertzuweisung der *bauteilKlassifikationen* für eine **Widerlagerwand**, eine **Fluegelwand** und ein **Widerlagerfundamentes** nach SOM geprüft.

```
<Concept uuid="00000036-0936-0936-0000-000002907145" name="Bestandteile ohne AE" code="">
  <Template ref="AggregationIsDecomposedOhneAE"/>
  <Requirements>
    <Requirement applicability="export" exchangeRequirement="EinfacheAggregation" requirement="mandatory"/>
  </Requirements>
  <TemplateRules operator="and">
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND RelatedObjects[Exists]=TRUE AND ObjectType[Value]='br.ub.11'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND RelatedObjects[Exists]=TRUE AND ObjectType[Value]='br.ub.12'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND RelatedObjects[Exists]=TRUE AND ObjectType[Value]='br.ub.13'"/>
  </TemplateRules>
</Concept>
```

Abbildung 70: Aggregation *IsDecomposedBy* eines Widerlagers im View

Dementsprechend besteht ein Widerlager nur die Prüfung, wenn ihm über **IfcRelAggregates** mindestens ein Bestandteil von jedem dieser Typen zugewiesen ist.

Es fällt auf, dass hierbei nur eine Wirkrichtung der **IfcRelAggregates** dargestellt wird. In 5.2 wurde bereits beschrieben, dass **IfcRelAggregates** in zwei Richtungen wirkt: Einmal von Seiten des übergeordneten, zusammengesetzten Elements und einmal von Seiten der Bestandteile. Dementsprechend muss ein weiteres Template „Ist Teil von ohne AE“ angelegt werden, um prüfen zu können ob **Widerlagerwaende**, **Fluegelwaende** und **Widerlagerfundamente** ein **Widerlager** erzeugen. Die entsprechenden Templates und der „View“ sind im Anhang unter „mvdXML Aggregation Templates“ und „mvdXML Aggregation View“ zu finden. Sie beschreiben die Kette: *Decomposes[Exists]* und *RelatingObject[Exists]* und *ObjectType[Value]*="br.ub.10". Diese ist als Concept den einzelnen zu prüfenden Modellkomponenten zugewiesen.

Eine Probe der mvdXML im Xbim Xplorer wird in Abbildung 71 dargestellt. Der Versuch einer Probe im FZKViewer scheiterte daran, dass der FZKViewer nach einer [Exists]-Probe keine [Value]-Probe durchführen kann. Dementsprechend kann er die Regeln nicht erfolgreich prüfen.



Abbildung 71: Ergebnis der einfachen Aggregationsprüfung

In einem nächsten Schritt werden die erstellten Templates erweitert, um sie dem SOM anzunähern. Ziel ist es die Modellkomponenten nicht mehr über das Attribut *ObjectType*, sondern über das Attribut *bauteilklassifikation* als Property des PropertySets **Allgemeine_Eigenschaften** zu klassifizieren. Die erweiterten Templates „Bestandteile mit AE“ und „Ist Bestandteil mit AE“ sind ebenfalls unter „mvdXML Aggregation Templates“ und „mvdXML Aggregation View“ zu finden.

In die Aggregationstemplates wurde das Template zur Klassifizierung aus 5.3.1 integriert. Dadurch kann der Pfad von *Decomposes/IsDecomposedBy* bis zur *bauteilklassifikation* geprüft werden. Dementsprechend können auch die Regeln ergänzt werden. Abbildung 72 zeigt dies beispielhaft für das Widerlager. Der bisher erstellte Verknüpfungspfad wird erweitert, indem das verknüpfte Element auf das Attribut *IsDefinedBy* untersucht wird. Von diesem Attribut aus erweitert sich die Prüfung analog zu 5.3.1 bis hin zur Wertabfrage des Attributes.

```
<TemplateRules operator="and">
  <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND RelatedObjects[Exists]=TRUE AND IsDefinedBy[Exists]=TRUE AND PSetName[Value]='Allgemeine_Eigenschaften'
    AND PropertyName[Value]='bauteilklassifikation' AND AttributWert[Value]='br.ub.11'"/>
  <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND RelatedObjects[Exists]=TRUE AND IsDefinedBy[Exists]=TRUE AND PSetName[Value]='Allgemeine_Eigenschaften'
    AND PropertyName[Value]='bauteilklassifikation' AND AttributWert[Value]='br.ub.12'"/>
  <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND RelatedObjects[Exists]=TRUE AND IsDefinedBy[Exists]=TRUE AND PSetName[Value]='Allgemeine_Eigenschaften'
    AND PropertyName[Value]='bauteilklassifikation' AND AttributWert[Value]='br.ub.13'"/>
</TemplateRules>
```

Abbildung 72: Anforderungen des Widerlagers

Das Ergebnis ist wie im vorherigen Fall eine erfolgreiche Prüfung des **Widerlagers** (siehe Abbildung 73). Es besteht aus den geforderten drei Komponentenarten und diese wiederum sind einer als **Widerlager** gekennzeichneten Modellkomponente zugeordnet.



Abbildung 73: Widerlagerprüfung mit der Klassifikation im PropertySet

Es sei noch angemerkt, dass wie auch in 5.3.2 beide Prüfungen der Aggregationsstruktur in der gleichen mvdXML eingearbeitet wurden. Es wurden wieder zwei unterschiedliche „Exchange Requirements“ angelegt. Die „Concepts“, welche auf das simplifizierte Aggregationstemplate zugreifen, wurden dem ersten dieser Exchange Requirements zugewiesen, während das zweite Exchange Requirement mit den Concepts verbunden wurde, welche das komplexere Aggregationstemplate verwenden. Die Exchange Requirements können unabhängig voneinander ausgeführt werden. Dementsprechend können in einer mvdXML mehrere unterschiedliche Prüfungen eingearbeitet werden, unter anderem ein Model Content Checking und ein Validation Checking.

5.3.4. VOLLSTÄNDIGE PRÜFUNG EINES BEISPIELMODELLS

Nachdem nun Vorlagen für Templates zur Prüfung der einzelnen Relationen erstellt und getestet wurden, kann der vollständige Ablauf eines BMC beispielhaft anhand des Beispiels einer erweiterten Version des Brückenmodells 2 (4.1.2) erfolgen. Die hierfür zu durchlaufende Prozesskette wird in Abbildung 74 visualisiert.

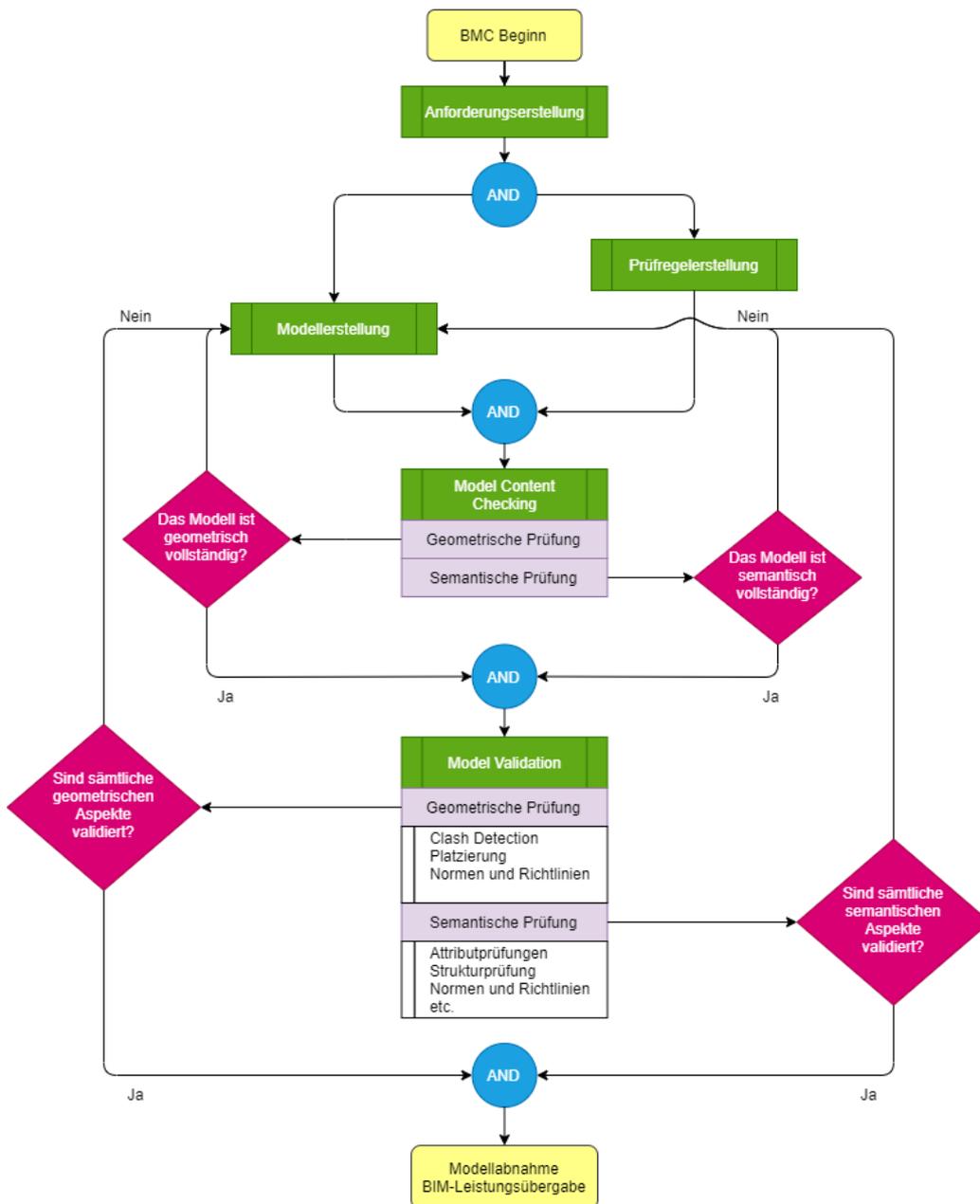


Abbildung 74: Prozesskette eines BMC

Anhand der Modellanforderungen wird zum einen das Modell erzeugt, zum anderen werden die Anforderungen in Prüfregeln überführt. Sofern beide Schritte abgeschlossen sind, erfolgt ein Model Content Checking, in welchem überprüft wird, ob ein Modell geometrisch und semantisch vollständig ist. Wenn dadurch sichergestellt ist, dass alle für die

Validierung relevanten Informationen im Modell existieren, wird die Model Validation durchgeführt.

Das nachfolgende Beispiel beschränkt sich auf die semantischen Anforderungen. Unter Berücksichtigung der bisherigen Ergebnisse dieser Arbeit ergibt sich der folgende Ablauf:

1. Die Anforderungen an das Modell aufstellen.
2. Das Modell bedarfsgerecht erweitern.
3. Die Modellanforderungen anhand der RASE-Methode in Prüfregelein umwandeln.
4. Über ein Model Content Checking das Vorhandensein aller relevanten Informationen gewährleisten.
5. Anhand eines Validation Checking validieren, dass das Modell auch den aufgestellten Anforderungen gerecht wird.

Die Prüfregelein werden vollständig in mvdXML implementiert und das BMC wird für beide Teilbereiche im Xbim Xplorer durchgeführt.

Schritt 1: Anforderungserstellung

Ziel ist es ein Brückenmodell anhand einer reduzierten Form des SOM zu prüfen. Es wird gefordert, dass das Modell auf dem neustrukturierten SOM (3.3.4) basiert, wobei die Modellkomponenten und ihre Aggregationsstruktur wie nachfolgend dargestellt (Tabelle 9) gegliedert sein sollen. Die Aggregationsstruktur soll hierbei in beiden Wirkrichtungen vollständig vorhanden sein.

Tabelle 9: Die geforderte Aggregationsstruktur der Brücke

Bruecke	Ebene 1: Bestandteile der Brücke	Ebene 2: Bestandteile der Ebene 1	Ebene 3: Bestandteile der Ebene 2
	Ueberbau	Brueckentragwerk	Brueckenplatte Brueckentraeger
		Kappe	
	Unterbau	Widerlager	Widerlagerwand Fluegelwand Widerlagerfundament
	FesteFahrbahn		

Für die Attribuierung gilt:

- Sämtliche Modellkomponenten müssen über das PropertySet **Allgemeine_Eigenschaften** und das zugehörige Attribut *bauteilKlassifikation* eindeutig klassifiziert sein.
- Sämtliche Modellkomponenten müssen über das PropertySet **Allgemeine_Eigenschaften** und das zugehörige Attribut *name* verfügen.
- Sämtliche Modellkomponenten der Ebene 3 sowie die **FesteFahrbahn** müssen über das PropertySet **Bauteil** ein Material über die Relation

IfcRelAssociatesMaterial zugewiesen bekommen. Die zu verwenden Materialien sind in Tabelle 10 aufgeführt:

Tabelle 10: Materialanforderungen der einzelnen Modellkomponenten

Modellkomponente	Materialname	Materialbeschreibung	Materialkategorie
Brueckenplatte	C40/50		Stahlbeton
Brueckentraeger	S455		Stahl
Kappe	C20/25	Ortbeton	Beton
Widerlagerwand	C30/37	Ortbeton	Stahlbeton
Fluegelwand	C30/37	Ortbeton	Stahlbeton
Widerlagerfundament	C30/37	Ortbeton	Stahlbeton
FesteFahrbahn	Asphalt	Gussasphalt	Bitumenmaterial

- Der **Bruecke** sollen über das PropertySet **BR** die Attribute *spannweite* und *lichteHoehe* zugewiesen werden, wobei *spannweite* mit einem Wert größer 0 Meter und kleiner 30 Meter und die *lichteHoehe* mit einem Wert größer 6 Meter attribuiert sein sollen.

Schritt 2: Modellanpassung

Um die beschriebenen Anforderungen erfüllen zu können, musste das Brückenmodell 2 um die meisten Modellkomponenten höherer Aggregationsebenen erweitert werden, also die **Bruecke**, den **Ueberbau**, das **Brueckentragwerk**, den **Unterbau** und die **Widerlager**. Sämtliche Modellkomponenten mussten anschließend über die **IfcRelAggregates** gegliedert werden.

Ferner waren die Materialien entsprechend der Anforderungen anzupassen, also in fast allen Fällen die Materialbeschreibung (*Description*) und Materialkategorie (*Category*) zu attribuieren. Schlussendlich wurde die korrekte Attribuierung der Modellkomponenten sichergestellt. Unter „IFC-STEP Anpassung des Brückenmodells 2“ kann die Erweiterung des Brückenmodells um die Metaobjekte und die Aggregationen eingesehen werden.

Schritt 3: Erstellung der Prüfregele

Für die Modellprüfung müssen die Anforderungen zunächst in Prüfregele umgeformt und diese anschließend in einer mvdXML-Datei umgesetzt werden. Die vollständige mvdXML-Datei kann unter „BMC mvdXML“ eingesehen werden.

In dieser mvdXML-Datei wurden zwei „Exchange Requirements“ angelegt:

- Model Content Checking
- Model Validation

Dadurch können beide Prüfungen in einer einzigen Datei zusammengefasst, aber dennoch getrennt ausgeführt werden.

Für das Model Content Checking müssen drei Regeln geprüft werden:

1. Allen **IfcBuildingElements** (stellen die Modellkomponenten dar) müssen über die Attribute *bauteilKlassifikation* und *name* als Teile des PropertySets **Allgemeine_Eigenschaften** zugewiesen sein.
2. Allen Nicht-Metamodellkomponenten muss ein Material zugewiesen sein.
3. Der **Bruecke** sollen die zwei Attribute *spannweite* und *lichteHoehe* als Teil des PropertySets **BR** zugewiesen sein.

Für die erste Regel reicht es aus, alle **IfcBuildingElements** (Select) ohne weitere Präzisierungen (Applicability) oder Ausnahmen (Exception) zu prüfen. Die Requirements in Form der zugewiesenen Attribute als Teil des PropertySets **Allgemeine_Eigenschaften** werden über die Wertbelegung des Templates aus 5.3.1 geprüft.

Für die Materialprüfung muss die Auswahl der zu prüfenden Komponenten anhand des Applicability-Ansatzes aus 5.3.3 erfolgen, um alle Metaobjekte auszuschließen. Das bedeutet, es werden alle **IfcBuildingElementProxys** geprüft (Select), bei denen das Attribut *bauteilKlassifikation* mit einem Wert der Menge {br.ue.11, br.ue.12, br.ue.20, br.ub.11, br.ub.12, br.ub.13, ob.bet.100} (Applicability) attribuiert ist. Zur Prüfung der Zuweisung eines Materials werden das erste Template und Concept aus 5.3.2 genutzt.

Die konkrete Prüfung der **Bruecke** auf die zwei geforderten Attribute erfolgt bezüglich Select und Applicability analog zur Materialprüfung. Die Requirements nutzen als Concept die Wertbelegung des Templates aus 5.3.1.

Für die Model Validation sind ebenfalls mehrere Anforderungen einzuhalten:

1. Die Bauteile müssen die Aggregationsstrukturen einhalten. An dieser Stelle wird nur ein Beispiel aufgeführt:

Eine als „Widerlager_1“ bestimmte Modellkomponente:

- Select: Bei der Modellkomponente muss es sich um ein **IfcBuildingElementProxy** handeln.
- Applicability: *bauteilKlassifikation* = „br.ub10“ und *name* = „Widerlager_1“

muss die folgenden Requirements erfüllen:

- Es muss Bestandteil des Unterbaus (**IfcBuildingElementProxy**, *bauteilKlassifikation* = “br.ub”) sein.
- Es muss aus den zugehörigen Teilkomponenten bestehen:
 - „Widerlagerwand_WL1“ (**IfcBuildingElementProxy**, *bauteilKlassifikation* = “br.ub.11“, *name* = “Widerlagerwand_WL1“)
 - „Fluegelwand_links_WL1“ (**IfcBuildingElementProxy**, *bauteilKlassifikation* = “br.ub.12“, *name* = “Fluegelwand_links_WL1“)

- „Fluegelwand_rechts_WL1“ (**IfcBuildingElementProxy**, *bauteilKlassifikation* = “br.ub.12”, *name* = “Fluegelwand_rechts_WL1”)
 - „Widerlagerfundament_WL1“ (**IfcBuildingElementProxy**, *bauteilKlassifikation* = “br.ub.13”, *name* = “Widerlagerfundament_WL1”)
 - Diese Aggregationsstruktur muss in beiden Wirkrichtungen vorliegen.
2. Die Materialien der Nicht-Metaobjekte (**IfcBuildingElementProxy**, *bauteilKlassifikation* in {br.ue.11, br.ue.12, br.ue.20, br.ub.11, br.ub.12, br.ub.13, ob.bet.100}) müssen den Materialanforderungen aus Tabelle 10 genügen.
 3. Für die Bruecke (**IfcBuildingElementProxy**, *bauteilKlassifikation* = “br”) müssen die Attribute *Spannweite* und *Lichte_Höhe* die vorgegebenen Grenzwerte einhalten.

Es wird deutlich, dass die Regeln nun für einen Großteil der Modellkomponenten einzeln angelegt werden müssen, da die Prüfung der korrekten Aggregation für die meisten Komponenten variieren. Eine Ausnahme bilden beispielsweise die **Brueckentraeger**. Diese müssen allesamt Teil desselben **Brueckentragwerks** sein. Dementsprechend muss jede Modellkomponente eindeutig ausgewählt werden.

Für die Prüfregele werden stets **IfcBuildingElementProxys** (Select) ausgewählt. Sie müssen für die Applicability jeweils einen gewissen Wert der Attribuierung von *bauteilKlassifikation* aufweisen. Sofern mehrere Modellkomponenten eines Typs im Modell vorhanden sind, für die unterschiedliche Regeln gelten (z.B. die Widerlager), muss ferner noch das Attribut *name* einen bestimmten Wert aufweisen, um die Modellkomponenten eindeutig voneinander zu unterscheiden.

Für die Aggregation werden jeweils erweiterte Formen der Templates “Besteht aus mit AE” und „Bestandteile mit AE“ aus 5.3.3 genutzt. Hierbei wurde in beiden Templates die „AttributeRule“ *HasProperties* erweitert. Es wurde ergänzt, dass das PropertySet ein zweites Attribut vom Typ **IfcSingleValue** hat. Dies erlaubt das gleichzeitige Überprüfen zweier Properties eines PropertySets in einem Concept.

Dadurch kann in den Concepts die Anforderungen an die Attribuierungskette erweitert werden. Im Falle des „Widerlagerfundament_1“ wird die bisherige Attributkette

```
Decomposes[Exists]=TRUE AND RelatingObject[Exists]=TRUE AND IsDecomposedBy[Exists]=TRUE AND PsetName[Value]='Allgemeine_Eigenschaften' AND PropertyName[Value]='bauteilKlassifikation' AND PropertyWert[Value]='br.ub.10'
```

erweitert um

```
PropertyzweiName[Value]='name' AND PropertyzweiWert[Value]='Widerlager_1'.
```

Es wird also zusätzlich zur korrekten *bauteilKlassifikation* des übergeordneten **Widerlagers** gefordert, dass dieses den richtigen Individualnamen hat. Hier hätte alternativ auch die *identitaet* als zweites Attribut zur individuellen Auswahl genutzt werden können.

In gleicher Weise werden für alle weiteren Modellkomponenten die zugehörigen Concepts in der mvdXML-Datei erstellt.

Bei den Materialien werden die Anforderungen an die entsprechenden Modellkomponenten um ein weiteres Concept erweitert. Für dieses wird das Concept „Material der Kategorie Stahlbeton“ aus 5.3.2 ergänzt. Das zugehörige Template zur Zuweisung eines „Single-Material“ an ein **IfcObject** kann unverändert übernommen werden. Die Erweiterung des Concepts nutzt die in diesem Template bereits angelegten Attributregeln „Materialname“ und „Materialbeschreibung“. Diese werden in den jeweiligen Concepts mit den Werten entsprechend den Anforderungen belegt.

Die Grenzwertprüfung der **Bruecke** greift auf die Templates und Concepts der für die *bauteilklassifizierung* zurück, nur dass im Concept der „PsetName“ mit „BR“ und der „Attributname“ mit „spannweite“ und „lichtehoehe“ belegt werden. Die Grenzwertbeschränkungen können im Concept über einfache Abfragen, wie sie in Abbildung 75 zu finden sind, überprüft werden.

```
<Concept uuid="00000036-0936-0936-0000-000002907142" name="Grenzwerteinhaltung" code="">
  <Definitions>
    <Definition>
      <Body lang="en"></Body>
    </Definition>
  </Definitions>
  <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
  <Requirements>
    <Requirement applicability="export" exchangeRequirement="00000036-0936-0936-0000-000000000000" requirement="mandatory"/>
  </Requirements>
  <TemplateRules operator="and">
    <TemplateRule Parameters="PsetName[Value]='BR' AND AttributName[Value]='Lichte_Hoehe' AND Wert[Value]>6"/>
    <TemplateRule Parameters="PsetName[Value]='BR' AND AttributName[Value]='Spannweite' AND Wert[Value]<30 AND Wert[Value]>0"/>
  </TemplateRules>
</Concept>
```

Abbildung 75: Concept der Grenzwertprüfung der Brückenattribute

Schritt 4: Model Content Checking

Nachdem die Anforderungen in Prüfregeln und aus diesen eine mvdXML-Datei erstellt wurde, kann das BMC erfolgen.

Um im Rahmen des Model Content Checking zu überprüfen, ob alle relevanten Aspekte vorhanden sind, wird über den Exchange Requirement „Model Content Checking“ der mvdXML im Xbim Xplorer durchgeführt. Das Ergebnis dieser Prüfung ist in Abbildung 76 auszugsweise dargestellt. Das Model Content Checking ergibt, dass sämtliche Modellkomponenten über die benötigten Attribute verfügen.

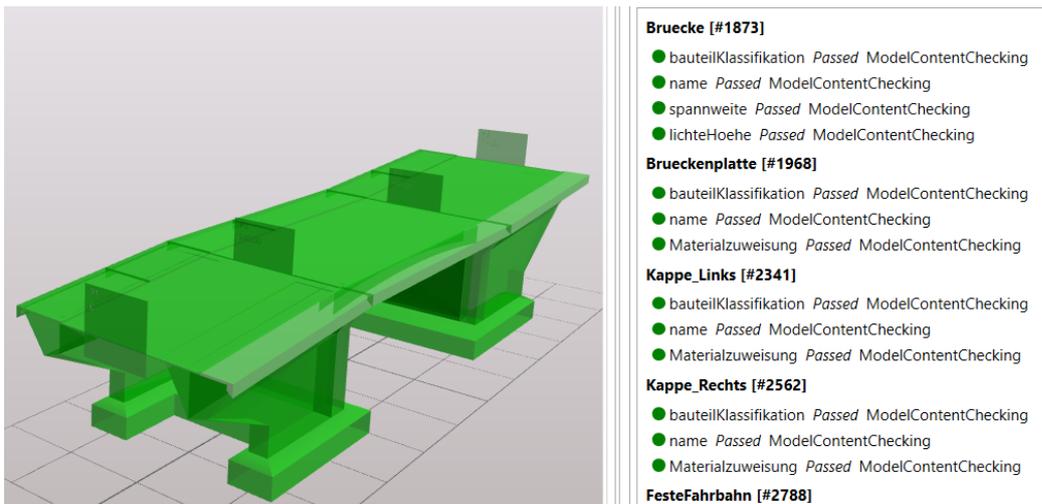


Abbildung 76: Ergebnisse des Model Content Checking

Schritt 5: Model Validation

Nachdem im vorangegangenen Schritt sichergestellt wurde, dass alle für die Validierung benötigten Informationen wie die Attribute *bauteilKlassifikation* und *name* für alle Modellkomponenten attribuiert sind, wird das Modell validiert und so abschließend sichergestellt, dass diese Informationen den Anforderungen entsprechen. Auch die Model Validation wird im Xbim Xplorer anhand der in Schritt 3 erzeugten mvdXML durchgeführt. Hierbei wird die Exchange Requirement „Model Validation“ verwendet. Ein Auszug der zugehörigen Prüfungsergebnisse ist in Abbildung 77 dargestellt. Unter anderem ist zu sehen, dass die Grenzwerte der Abmessungen der Brücke eingehalten werden und dieser die passenden Bestandteile zugewiesen sind. Für die **Brueckenplatte** wurde neben der Zuordnung zum **Brueckentragwerk** validiert, dass das zugewiesene Material den Vorgaben entspricht. Entsprechend der „BMC mvdXML“ bedeutet dies, dass es sich bei dem Material um ein **IfcMaterial** mit Namen „C40/50“ der Kategorie „Stahlbeton“ handelt.

Demnach ist das gegebene Modell bezüglich der erstellten Anforderungen validiert.

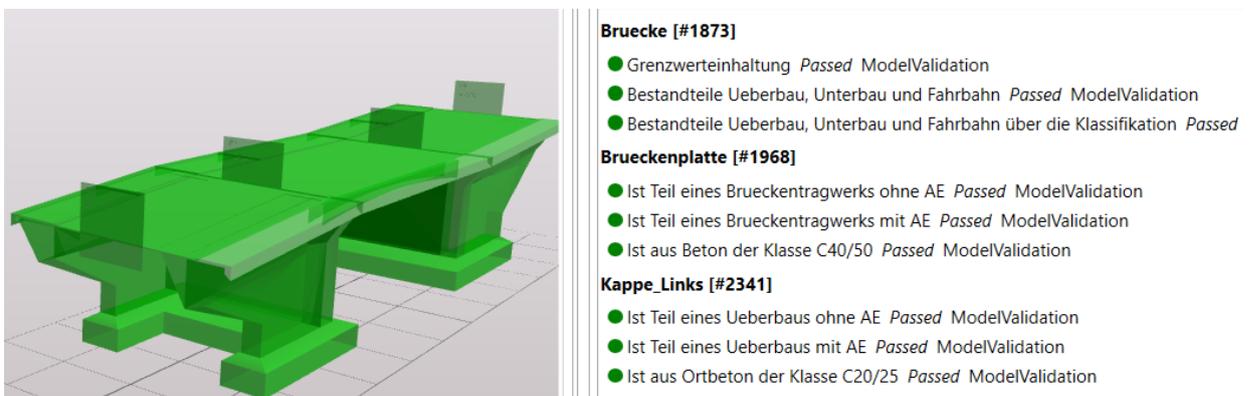


Abbildung 77: Ergebnisse der Model Validation

6 FAZIT UND AUSBLICK

Um die Ergebnisse dieser Arbeit zu bewerten, sollten zunächst die gesetzten Ziele in Erinnerung gerufen werden:

- Die Erarbeitung eines Überblicks über das BMC
- Die Entwicklung von theoretischen und praktischen Grundlagen des BMC
- Die Validierung dieser Ergebnisse anhand mehrerer Beispiele

Zur Erfüllung dieser Ziele wurden maßgebende Begrifflichkeiten des BMC aufgearbeitet sowie verschiedene geeignete Softwarelösungen vorgestellt. Ferner wurden Ansätze zur Definition und Gliederung des BMC diskutiert. Als Ergebnis wurden mit dem Model Content Checking und der Model Validation zwei Teilbereiche des BMC bestimmt, die in der vorliegenden Arbeit näher untersucht wurden.

Das Model Content Checking beschreibt als der erste dieser Teilbereiche die Überprüfung eines Modells auf die Vollständigkeit aller relevanten Informationen. Diese Vollständigkeit dient auch als Grundlage für den zweiten Teilbereich des BMC (Model Validation). Es wurde ferner gezeigt, dass die gängigen BMC-Softwarelösungen (Solibri, Desite, BIMcollab ZOOM) den Anforderungen eines Model Content Checking gerecht werden, insbesondere wenn sie durch BIMQ unterstützt werden.

Mit der Model Validation wurde ein zweiter Teilbereich des BMC mit seinen Grundlagen beleuchtet. Aus dem sehr breiten Spektrum von Modellaspekten, die einer Modellvalidierung unterzogen werden könnten, wurden wegen ihres grundlegenden Charakters und ihrer hohen Praxisrelevanz beispielhaft einige Relationen (***IfcRelDefinesByProperties***, ***IfcRelAssociatesMaterial*** und ***IfcRelAggregates***) vertieft betrachtet und anhand dieser eine Validierungsgrundlage geschaffen.

Ferner untersuchte diese Arbeit anhand des Beispiels des Semantischen Objektmodells der DB Netz AG (SOM), wie Modellanforderungen erstellt werden können und welche Herausforderungen dabei zu bewältigen sind, unter anderem auch im Zusammenhang mit Vertragsregelungen. Auch wurden die Probleme in der Vertragsregelungen hierzu berücksichtigt. Bezüglich der Anforderungen wurde weiterhin betrachtet, wie diese in Prüfreden übersetzt werden können, welche den Ansprüchen eines BMC genügen. Das hierbei diskutierte Prinzip der RASE konnte erfolgreich auf das SOM angewendet und diese Regeln bezogen auf das Model Content Checking in BIMQ implementiert werden. Für dieses wurde die Eignung zur Erstellung von Modellanforderungen und Prüfreden nochmals deutlich.

Mithilfe der in BIMQ erzeugten Regeln konnten drei Beispielmodelle ein Model Content Checking durchlaufen. Dabei kristallisierten sich Unterschiede in der Arbeitsweise und der angebotenen Funktionalität mehrerer gängiger Model Checker heraus. Zusätzlich wurde

die Problematik proprietärer Datenformate der verschiedenen Softwarelösungen aufgezeigt. Diese unterscheiden sich in ihrer Informationsdarstellung und ihrem Informationsgehalt, was die Implementierung und Durchführung von Prüfungen erschwert.

Für das Validation Checking wurden die wesentlichen Grundlagen für die Validierung von Materialien und Aggregationsstrukturen dargestellt. Dabei wurden die Erstellung und Struktur einer mvdXML-Datei erörtert und das enorme Potential dieses Formates deutlich. Insbesondere der hohe Grad der Wiederverwertbarkeit und Flexibilität eines einmal erstellten Templates wurde unterstrichen. Gleichzeitig zeigte sich die Beschränkung dieses Vorgehens bezüglich geeigneter Softwarelösungen welche mvdXML-Dateien einlesen und verarbeiten können.

Die abschließende Durchführung eines vollständigen BMC-Ablaufs veranschaulichte die Logik und die Struktur der in dieser Arbeit vorgeschlagenen Prozesskette an einem konkreten Beispiel.

Die vorliegende Untersuchung zeigt deutlich, dass das BMC in seiner Relevanz kaum überbewertet werden kann. Die RASE-Methodik und Softwarelösungen wie BIMQ schaffen notwendige Grundlagen für die Erstellung von Prüfregelein anhand von Modellanforderungen und damit für die praktische Durchführung der Prüfungen für ein Model Content Checking. Im Hinblick auf die Model Validation wurde offensichtlich, dass die Nutzung proprietärer Datenformate die Implementierung und Durchführung von Prüfungen stark einschränkt. Gerade diese Art der Prüfung ist jedoch langfristig maßgebend. Die im Rahmen dieser Arbeit gezeigte Eignung des mvdXML-Formats zur Beschreibung und Implementierung von Prüfregelein für IFC-Modelle fordert geradezu, dass Softwarelösungen, die heute noch nicht mit diesem Format kompatibel sind, entsprechend erweitert werden sollen. Als alternative Option dazu bietet es sich an zu untersuchen, ob sich mvdXML-Dateien in geeigneter Weise und ohne wesentlichen Informationsverlust in proprietäre Prüfformate übersetzen lassen.

Zusammenfassend kann man also sagen, dass mit den gängigen, kommerziellen Softwarelösungen ein Modell Content Checking weitestgehend automatisiert durchgeführt werden kann. Einschränkungen sind vor allem für eine Model Validation gegeben. Probleme sind vor allem, dass man mvdXML-Prüfregelein weitestgehend manuell erstellen muss sowie das Fehlen vollumfänglicher Softwarelösungen zur Prüfung anhand des mvdXML-Formats (u.a. Instabilität, eingeschränkte Unterstützung von mvdXML, unzureichende Fehleranalyse).

VI LITERATURVERZEICHNIS

- [1] Hjelseth E. Foundation for BIM-based model checking systems [PhD Thesis]. Ås, Norwegen: Norwegian University of Life Science; 2015.
- [2] BMVI. Stufenplan Digitales Bauen; 2015.
- [3] Borrmann A, König M, Koch C, Beetz J. Building Information Modeling. Cham: Springer International Publishing; 2018.
- [4] Nemetschek Group. Nemetschek Group: Nemetschek Group; 2021. Available from: URL: <https://www.nemetschek.com/>.
- [5] thinkproject. Thinkproject: Desite-BIM: <https://thinkproject.com/products/desite-bim/>.
- [6] BIMcollab ZOOM. BIMcollab ZOOM. Available from: URL: <https://www.bimcollab.com/de/products/bimcollab-zoom>.
- [7] DIN. DIN EN ISO 16739: Industry Foundation Classes (IFC) für den Datenaustausch in der Bauwirtschaft und im Anlagenmanagement.
- [8] DIN. DIN SPEC 91391-2: Gemeinsame Datenumgebungen (CDE) für BIM-Projekte – Funktionen und offener Datenaustausch zwischen Plattformen unterschiedlicher Hersteller – Teil 2: Offener Datenaustausch mit Gemeinsamen Datenumgebungen.
- [9] DIN. DIN EN ISO 29481-2: Bauwerksinformationsmodelle - Handbuch der Informationslieferungen - Teil 2: Interaktionsframework.
- [10] DIN. DIN EN ISO 29481-1: Bauwerksinformationsmodelle - Handbuch der Informationslieferungen - Teil 1: Methodik und Format.
- [11] DIN. DIN EN ISO 23387: Bauwerksinformationsmodellierung (BIM) – Datenvorlagen für Bauobjekte während des Lebenszyklus eines baulichen Vermögensgegenstandes - Konzepte und Grundsätze.
- [12] DIN. DIN EN ISO 23386: Bauwerksinformationsmodellierung und andere digitale Prozesse im Bauwesen - Methodik zur Beschreibung, Erstellung und Pflege von Merkmalen in miteinander verbundenen Datenkatalogen.
- [13] DIN. DIN EN ISO 21597-2: Informationscontainer zur Datenübergabe - Austauschspezifikation - Teil 2: Dynamische Semantik.
- [14] DIN. DIN EN ISO 19650-5: Organisation und Daten zu Bauwerken - Informationsmanagement mit BIM - Teil 5: Spezifikation für Sicherheitsbelange von BIM, der digitalisierten Bauwerke und des smarten Assetmanagements.

- [15] DIN. DIN EN ISO 21597-1: Informationscontainer zur Datenübergabe - Austausch-Spezifikation - Teil 1: Container.
- [16] DIN. DIN EN 17412: Building Information Modelling – BIM-Definitionsgrade - Konzepte und Definitionen.
- [17] DIN. DIN EN 19650-2.
- [18] DIN. DIN EN 19737.
- [19] DIN. DIN EN ISO 12006-2: Hochbau - Organisation des Austausches von Informationen über die Durchführung von Hoch- und Tiefbauten – Teil 2: Struktur für die Klassifizierung.
- [20] DIN. DIN EN ISO 12006-3: Bauwesen - Organisation von Daten zu Bauwerken - Teil 3 Struktur für den objektorientierten Informationsaustausch.
- [21] DIN. DIN EN ISO 16757-2: Datenstrukturen für elektronische Produktkataloge der Technischen Gebäudeausrüstung: Teil 2 Geometrie.
- [22] DIN. DIN EN ISO 16757-1: Datenstrukturen für elektronische Produktkataloge der Technischen Gebäudeausrüstung - Teil 1: Konzepte, Architektur und Modelle.
- [23] DIN. DIN EN ISO 19650-2: Organisation und Digitalisierung von Informationen zu Bauwerken und Ingenieurleistungen, einschließlich Bauwerksinformationsmodellierung (BIM) - Informationsmanagement mit BIM - Teil 2: Planungs-, Bau- Inbetriebnahme-phase.
- [24] DIN. DIN EN ISO 19650-1: Organisation und Digitalisierung von Informationen zu Bauwerken und Ingenieurleistungen, einschließlich Bauwerksinformationsmodellierung (BIM) - Teil 1: Begriffe und Grundsätze.
- [25] DIN. DIN EN ISO 19650-3: Organisation und Informationen zu Bauwerken -Informationsmanagement und Bauwerksinformationsmodellierung - Teil 3: Betriebsphase der Assets.
- [26] BSI. BSI-PAS 1192-2: Specification for information management for the capital/delivery phase of construction projects using building information modelling. London: BSI British Standards.
- [27] BSI. BSI-PAS 1192-3: Specification for information management for the operational phase of assets using building information modelling. London: BSI British Standards.
- [28] BSI. BSI-PAS 1192-4: Fulfilling employer's information exchange requirements using COBie - Code of practice.
- [29] BSI. BSI-PAS 1192-5: Specification for security-minded building information modelling, digital built environments and smart asset management.

- [30] BSI. BSI-PAS 1192-6: Specification for collaborative sharing and use of structured Health and Safety information using BIM.
- [31] VDI. VDI 2552 Blatt 1: Grundlagen.
- [32] VDI. VDI 2552 Blatt 10: AIA und BAP.
- [33] VDI. VDI 2552 Blatt 2: Begriffe.
- [34] VDI. VDI 2552 Blatt 3: Modellbasierte Mengenermittlung zur Kostenplanung, Terminplanung, Vergabe und Abrechnung.
- [35] VDI. VDI 2552 Blatt 4: Anforderungen an den Datenaustausch.
- [36] VDI. VDI 2552 Blatt 5: Datenmanagement.
- [37] VDI. VDI 2552 Blatt 7: Prozesse.
- [38] VDI. VDI 2552 Blatt 8.1: Qualifikation: Basiskenntnisse.
- [39] VDI. VDI 2552 Blatt 9: Klassifikationssysteme.
- [40] BIM4INFRA2020. BIM4INFRA2020_AP4_Glossar.
- [41] BIM4INFRA2020. BIM4INFRA2020_AP4_Teil1: Grundlagen und BIM-Gesamtprozess.
- [42] BIM4INFRA2020. BIM4INFRA2020_AP4_Teil2: Leitfaden und Muster für Auftraggeberinformationsanforderungen (AIA).
- [43] BIM4INFRA2020. BIM4INFRA2020_AP4_Teil3: Leitfaden und Muster für den BIM-Abwicklungsplan (BAP).
- [44] BIM4INFRA2020. BIM4INFRA2020_AP4_Teil4: Leitfaden zur Leistungsbeschreibung.
- [45] BIM4INFRA2020. BIM4INFRA2020_AP4_Teil5: Muster Besondere Vertragsbedingungen BIM (BIM-BVB).
- [46] BIM4INFRA2020. BIM4INFRA2020_AP4_Teil6: Steckbriefe der wichtigsten BIM-Anwendungsfälle.
- [47] BIM4INFRA2020. BIM4INFRA2020_AP4_Teil7: Handreichungen BIM Fachmodelle und Ausarbeitungsgrad.
- [48] BIM4INFRA2020. BIM4INFRA2020_AP4_Teil8: Neutraler Datenaustausch im Überblick.
- [49] BIM4INFRA2020. BIM4INFRA2020_AP4_Teil9: Datenaustausch mit Industry Foundation Classes (IFC).
- [50] building smart ev. IFC specifications: IFC 4.3; 2021. Available from: URL: https://standards.buildingsmart.org/IFC/DEV/IFC4_3/RC3/HTML/.
- [51] IFC Bridge Team. <https://ifcinfra.de/ifc-bridge/bridge-abschluss/>; 2020. Available from: URL: <https://ifcinfra.de/ifc-bridge/bridge-abschluss/>.

- [52] IFC Infra Team. http://ifcinfra.de/wp-content/uploads/2019/05/2019-03-26_IFC-Bridge_Overview_.pdf; 2019. Available from: URL: http://ifcinfra.de/wp-content/uploads/2019/05/2019-03-26_IFC-Bridge_Overview_.pdf.
- [53] building smart ev. IFC specifications: IFC 2x3: building smart ev.; seit 2008. Available from: URL: <https://standards.buildingsmart.org/IFC/RELEASE/IFC2x3/FINAL/HTML/>.
- [54] building smart ev. IFC specifications. IFC 4Add2. Available from: URL: <https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/>.
- [55] building smart ev. mvdXML: building smart ev.; 2021. Available from: URL: <https://technical.buildingsmart.org/standards/ifc/mvd/mvdxml/>.
- [56] Chi Zhang, Jakob Beetz, Matthias Weise. Interoperable validation for IFC building models using open standards 2014.
- [57] GitBook. mvdXML. Available from: URL: https://hujb2000.gitbooks.io/ifcspecification/content/en/specifications/mvdXML_overview.html.
- [58] Weise M. mvdXML 1.1. Available from: URL: <https://github.com/buildingSMART/mvdXML/tree/master/mvdXML1.1>.
- [59] Chipman T, Liebich T, Weise M. Specification of a standardized format to define and exchange Model View Definitions with Exchange Requirements and Validation Rules: building smart ev.; 2014. Available from: URL: https://standards.buildingsmart.org/MVD/RELEASE/mvdXML/v1-0/mvdXML_V1-0.pdf.
- [60] Carsten Klempin. BIMQ und Modelcheck mit DESITE MD. (BIMQ) [cited 2021 Apr 20]. Available from: URL: <https://www.youtube.com/watch?v=oaBDIt4cswc>.
- [61] Fliegel F, Welte F. BIMQ und Modelcheck mit Solibri; 2020. Available from: URL: <https://www.youtube.com/watch?v=T0TCRI2GJEY>.
- [62] Carsten Klempin. BIMQ und Modelcheck mit BIMcollab ZOOM; 2020. Available from: URL: <https://www.youtube.com/watch?v=YRVEuHKOnQM>.
- [63] building smart ev. IFC Documentation Generator: GitHub. Available from: URL: <https://github.com/BuildingSMART/IfcDoc>.
- [64] KIT. FZKViewer: KIT; 2021. Available from: URL: <https://www.iai.kit.edu/english/1309.php>.
- [65] XBIM Team. XBIM Xplorer. Available from: URL: <https://docs.xbim.net/>.
- [66] buildingSMART alliance® - a council of the National Institute of Building Sciences. National BIM Standard - United States® Version 3 - 4.2 Construction Operation Building information exchange (COBie).
- [67] Jernigan FE. National BIM Standard - United States Version 3: Terms and Definitions. 2nd ed. Salisbury MD: 4Site Press; 2008.

[68] Wegbreit B. Property extraction in well-founded property sets. IEEE Trans. Software Eng. 1975; SE-1(3):270–85.

[69] Sonnek DN. Untersuchungen zum Einsatz der BIM-Methode im konstruktiven Ingenieurbau anhand des Projektes "Neubau Brücke im Zuge der Verlängerung der Wiesenstraße-Nord in Jena über die Anlagen der Deutschen Bahn": Projektarbeit [Projektarbeit]. Dresden: Technische Universität Dresden; 2021.

[70] Lewerenz. BIM-LOD-Konzept V16. <https://www.deges.de/building-information-modeling-bim/>; DEGES. Available from: URL: <https://www.deges.de/building-information-modeling-bim/>.

[71] Hjelseth E. Exploring Semantic Based Model Checking. Ås, Norwegen: Norwegian University of Life Science.

[72] Øivind Rooth IK. Report on Open Standards for Regulations, Requirements and Recommendations Content.

[73] Hjelseth E. Classification of BIM-based model-checking concepts. ITcon 2016; 2016.

[74] Eilif Hjelseth, Nisbet N. Overview of concepts for model checking [Paper]. Ås, Norwegen: Norwegian University of Life Science; 2010.

[75] Autodesk. Revit Dokumentation; 2020. Available from: URL: <https://knowledge.autodesk.com/de/support/revit-products/learn/caas/qsguides/revit-quick-start-guide.html?Model/files/GUID-6E0ECA27-AF40-4B1D-9E0B-1DE5FBBBD45F2-htm.html>.

[76] Tim Westphal. Architekten im Interview: "Ein BAP ist nicht in Stein gemeißelt". Build-Ing. Mai:7–10. Available from: URL: https://www.build-ing.de/benutzerbereich/e-papper/?tx_hussyumpu_archiv%5Btitle%5D=BIM%2003%2F21&tx_hussyumpu_archiv%5Bcover%5D=http%3A%2F%2Fimg.yumpu.com%2F65624835%2F1%2F500x640%2Fbim-03-21.jpg%3FAWSAccessKeyId%3DAKIAICNEWSPSEKTJ5M3Q%26Expires%3D1623056400%26Signature%3Dvr%252FQCoT9PssozhZ3LPRFKTwR8ig%253D&tx_hussyumpu_archiv%5Bdescription%5D=&tx_hussyumpu_archiv%5Bdate%5D=12.05.2021&tx_hussyumpu_archiv%5Bembed%5D=https%3A%2F%2Fwww.yumpu.com%2Fde%2Fembed%2Fview%2Fhy1N2rYS8qDR79uZ&tx_hussyumpu_archiv%5Baction%5D=SingleShow&tx_hussyumpu_archiv%5Bcontroller%5D=Archiv&cHash=952a36949e83ea169f8b8d7db8b0da5a.

[77] BMVI. Umsetzung der Stufenplans Digitales Planen und Bauen: Erster Fortschrittsbericht; 2017.

[78] Borrmann A, König M, Hochmuth M, Liebich T, Elixmann R. Wissenschaftliche Begleitung der BMVI Pilotprojekte zur Anwendung von BIM im Infrastrukturbereich: Endbericht - Handlungsempfehlung 4.

- [79] André Borrmann, Markus König, Julian Amann, Matthias Braun, Robert Elixmann, Klaus Eschenbruch, Kerstin Hausknecht, Markus Hochmut, Thomas Lieblich, Markus Scheffer, Simon Vilgertshofer. Umsetzung des Stufenplans "Digitalen Planen und Bauen"; 2018.
- [80] André Borrmann, Markus König, Julian Amann, Matthias Braun, Robert Elixmann, Klaus Eschenbruch, Kerstin Hausknecht, Markus Hochmut, Thomas Lieblich, Markus Scheffer, Simon Vilgertshofer. Endbericht: Wissenschaftliche Begleitung Pilotprojekte Talbrücke Auenbach; 017.
- [81] André Borrmann, Markus König, Julian Amann, Matthias Braun, Robert Elixmann, Klaus Eschenbruch, Kerstin Hausknecht, Markus Hochmut, Thomas Lieblich, Markus Scheffer, Simon Vilgertshofer. Endbericht: Wissenschaftliche Begleitung Brücke über den Petersdorfer See; 2017.
- [82] André Borrmann, Markus König, Julian Amann, Matthias Braun, Robert Elixmann, Klaus Eschenbruch, Kerstin Hausknecht, Markus Hochmut, Thomas Lieblich, Markus Scheffer, Simon Vilgertshofer. Endbericht: Wissenschaftliche Begleitung Pilotprojekt EÜ Filstal; 2017.
- [83] André Borrmann, Markus König, Julian Amann, Matthias Braun, Robert Elixmann, Klaus Eschenbruch, Kerstin Hausknecht, Markus Hochmut, Thomas Lieblich, Markus Scheffer, Simon Vilgertshofer. Endbericht: Wissenschaftliche Begleitung Pilotprojekt Tunnel Rastatt; 2017.
- [84] Lewerenz. Muster-BAP Deges. <https://www.deges.de/building-information-modeling-bim/>: DEGES. Available from: URL: <https://www.deges.de/building-information-modeling-bim/>.
- [85] Hjelseth E. Experiences on converting interpretative regulations into computable rules [Paper]. Ås, Norwegen: Norwegian University of Life Science; 2013.
- [86] Sowa JF. The Challenge of Knowledge Soup. Research Trends in Science, Technology and Mathematics Education 2006.
- [87] Skript Aussagenlogikmengen [Vorlesungsskript]. Karlsruhe: KIT; 2021.
- [88] Hjelseth E, Nisbet N. Capturing Normative Constraints by Use of the Semantic Markup RASE Methodology [Paper]. Ås, Norwegen: Norwegian University of Life Science; 2011.
- [89] Nisbet N, Wix J, Conover D. Virtual futures for design, construction & procurement. Oxford, Malden MA: Blackwell Pub; 2008.
- [90] Shih S-Y, Sher W, Giggins H. Assessment of the Building Code of Australia to Inform the Development of BIM-enabled Codechecking Systems 2013; 2013.

[91] OASIS-open. LegalDocumentML TC: LegalDocML. Available from: URL: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=legaldocml.

[92] OASIS-open. LegalRuleML: LegalRuleML. Available from: URL: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=legalruleml.

[93] BIM me up! Export any Revit property to IFC using custom Property Sets. Available from: URL: <https://www.youtube.com/watch?v=SswHKtcM3ml>.

VII DIGITALES ANLAGEVERZEICHNIS

Diplomarbeit_Sonnek.pdf

SOM:

Neustrukturiertes_Semantisches_Objektmodell

Anforderungen_Brückenmodelle-BIMQ

BIMQ-Exporte:

BIMQ-Export_BRM1_Solibri

BIMQ-Export_BRM2_Solibri

BIMQ-Export_BRM3_Solibri

BIMQ-Export_BRM1_Desite.QA

BIMQ-Export_BRM2_Desite.QA

BIMQ-Export_BRM3_Desite.QA

BIMQ-Export_BRM3_Desite_korrigiert.QA

BIMQ-Export_BRM1_BIMcollabZOOM.bcsv

BIMQ-Export_BRM2_BIMcollabZOOM.bcsv

BIMQ-Export_BRM3_BIMcollabZOOM.bcsv

Solibri PrüfregeIn:

BRM1.cset

BRM2.cset

BRM3.cset

IFC-Modelle:

Brueckenmodell_1.ifc

Brueckenmodell_2.ifc

Brueckenmodell_3.ifc

Brueckenmodell_3_fehlerfrei.ifc

Widerlager.ifc

BMC-Modell.ifc

mvdXML-PrüfregeIn:

RelDefinesByProperties.mvdXML

RelAssociatesMaterial.mvdXML

RelAggregates.mvdXML

BMC.mvdXML

VIII ANLAGENVERZEICHNIS

Anlage 1	Semantisches Objektmodell Basisklassen	x
Anlage 2	Semantisches Objektmodell Bruecke	xi
Anlage 3	Semantisches Objektmodell Fahrbahn	xviii
Anlage 4	Semantisches Objektmodell Schnellübersicht	xxxiii
Anlage 5	BIMQ-Export BRM1 Solibri	xxxvi
Anlage 6	BIMQ-Export BRM1 Desite	xxxvii
Anlage 7	BIMQ-Export BRM1 BIMcollab ZOOM	xli
Anlage 8	mvdXML Materialzuweisungstemplate	lxxxvi
Anlage 9	mvdXML Materialzuweisungsview	lxxxvii
Anlage 10	mvdXML Aggregation Templates.....	lxxxviii
Anlage 11	mvdXML Aggregation View.....	xcii
Anlage 12	BMC mvdXML	xcv
Anlage 13	IFC-STEP Anpassung des Brückenmodells 2.....	cxxiv

ANLAGE 1 SEMANTISCHES OBJEKTMODELL BASISKLASSEN

Basisklassen		Allgemeine Eigenschaften	
name	AE		
kurzel	AE		
Elternklasse	-		Potentielle IFC-Attribute
Attribut	Alternativname	Bemerkungen	Description
<i>bemerkung</i>			
<i>kgk\verzeichnisnummer</i>		KGK\Verzeichnisnummer	
<i>idendat</i>		Identitaet	Globallid
<i>name</i>		Bauwerkname	Name
<i>bauteilklassifikation</i>		Klassifikation der Bauwerkteilgruppe	ObjectType
<i>bauteilname</i>		Name der Bauteilwerkgruppe	oder ObjectType
name	Bauteil		
kurzel	B		
Elternklasse	B		
Attribut	Alternativname		
<i>betriebszustand</i>		Betriebszustand	String
<i>material</i>		Material	Material
<i>materialeigenschaften</i>		Materialeigenschaften	String
<i>technischerplatz</i>		Technischer Platz	String
<i>technischerplatzbezeichnung</i>		Technischer Platz Bezeichnung	String
name	Einrichtung		
kurzel	E		
Elternklasse	E		
Attribut	Alternativname		
<i>ab</i>		Ist die Anlage im Bau erfasst	Boolean
<i>abbezeichnung</i>		Anlage im Bau - Bezeichnung	String
<i>abnummer</i>		Anlage im Bau - Nummer	Integer
<i>modellname</i>		Modellname	String
<i>netzbezirk</i>		Netzbezirk-Kuerzel	String
<i>netzsegment</i>		Netzsegment-OEKuerzel	Integer
<i>standort</i>		Ort des Projekts	String
<i>standortkennzahl</i>		Standortwerk Kennzahl	Integer
<i>strecke</i>			String

ANLAGE 2 SEMANTISCHES OBJEKTMODELL BRUECKE

name:	Bruecke	br
Kürzel	BR	
Elternklasse	E	
Besteht aus:	BET(FB), UE, UB, GR, ASR	
Attribut	Alternativname	lfoBridge
<i>ausrichtung</i>	Ausrichtung	String
<i>bauhoehe</i>	Bauhoehe	Double
<i>bauweise</i>	Bauweise	String
<i>breiteBahnLinks</i>	Breite Bahn_Links	Double
<i>breiteBahnRechts</i>	Breite Bahn_Rechts	Double
<i>breiteZwGelaender</i>	Breite zw Gelaender	Double
<i>brueckeTyp</i>	Brueckenbauart	String
<i>gleisAnzahl</i>	Anzahl der Gleise	Integer
<i>gleisRadiusMin</i>	Minimaler Gleisradius	Double
<i>kennung</i>	Bauwerkskennung	String
<i>konstruktionshoehe</i>	Konstruktionshoehe	Double
<i>lichteHoeheMin</i>	Kleinste Lichte-Hoehe	Double
<i>lichteWeite</i>	Lichte_Weite	Double
<i>tragwerkTyp</i>	Tragwerksart	String
<i>ueberschuetzungHoehe</i>	Ueberschuetzungshoehe	Double
<i>Spannweite</i>	Spannweite	Double

name:	Ueberbau	brue
Kürzel	UE	
Elementklasse	AE	
Bezieht aus:	BTy, KÄ	
Attribut	Alternativname	IceBuildingElementProxy
ID: Eruicke	String	

name	Brueckenragwerk	brue10
Kürzel	BTy	
Elementklasse	AE	
Bezieht aus:	RD	
Attribut	Alternativname	IceBuildingElementProxy
<i>ragwerk_Zoe</i>	Ueberbaueigenschaften	String
ID: Ueberbau	String	

name	Brueckenplatte	brue11	name	Brueckenträger	brue12
Kürzel	BRP		Kürzel	BRT	
Elementklasse	B		Elementklasse	B	
Bezieht aus:	Alternativname	IceBuildingElementProxy / IcePlate	Bezieht aus:	Alternativname	IceBuildingElementProxy / IceBeam
ID: ITraewerk	String		ID: ITraewerk	String	

name	RahmenDecke	brue11100
Kürzel	RD	
Elementklasse	BRP	
Bezieht aus:		
Attribut	Alternativname	IceBuildingElementProxy / IcePlate
<i>Aerock_aeroge</i>	Blocklaenge	Double
<i>Aerock_koerkkoekke</i>	Konstruktionshoehe	Double
<i>schlueckenmaterial</i>	Schluessenbezeichnung	String

name:	Unterbau	brub		
Kürzel	UB			
Elementklasse	AE			
Besteht aus:	W/L			
Attribut	Alternativname	IsBuildingElementProxy		
id/Erbecke	Id Erbecke	String		
name:	Widerlager	brub.10		
Kürzel	W/L			
Elementklasse	B			
Besteht aus:	W/L, FV			
Attribut	Alternativname	IsBuildingElementProxy		
art	WiderlagerTyp	String		
id/Unterbau	Id Unterbau	String		
name:	Widerlagerwand	brub.11		
Kürzel	W/W			
Elementklasse	B			
Besteht aus:	-			
Attribut	Alternativname	IsBuildingElementProxy / IsWall		
art	WiderlagerwandTyp	String		
id/Widerlager	Id Widerlager	String		
name:	Flügelwand	brub.12		
Kürzel	F/W			
Elementklasse	B			
Besteht aus:	-			
Attribut	Alternativname	IsBuildingElementProxy / IsWall		
art	Achse / Flügelart	Integer / String		
id/Widerlager	Id Widerlager	String		
name:	Widerlagerfundament	brub.13		
Kürzel	W/DF			
Elementklasse	B			
Besteht aus:	-			
Attribut	Alternativname	IsBuildingElementProxy		
art	Widerlager	String		
id/Widerlager	Id Widerlager	String		

name:	Gründung	br.gr		
Kürzel:	GR			
Elementklasse:	AE			
Besteht aus:	TG, FQ, SAS			
Attribut:	Alternativname	lfcBuildingElementProxy		
lfcBauwerk:	ID_Bauwerk			
abmessungKopf:	abmessungKopf	Double		
abmessungStufe:	abmessungStufe	Double		
ankerbodenAnordnung:	ankerbodenAnordnung	String		
ankerbodenDurchmesser:	ankerbodenDurchmesser	Integer		
ankerbodenLänge:	ankerbodenLänge	Double		
ankerbodenUeberstand:	ankerbodenUeberstand	Double		
eMasse:	Mast; SO ueber F, O.K. e+M;e	Double		
gruendungTyp:	gruendungTyp	String		
rampfStahlLänge:	rampfStahlLänge	Double		
rampfStahlProfil:	rampfStahlProfil	String		
rampfStahlSchlossAnzahl:	rampfStahlSchlossAnzahl	Integer		
rampfStahlStellung:	rampfStahlStellung	String		
stahlrohrDurchmesser:	stahlrohrDurchmesser	Integer		
stahlrohrLänge:	stahlrohrLänge	Double		
stahlrohrWandstaerke:	stahlrohrWandstaerke	Integer		
eMasse:	Gründung Vergrößerung;	Double		
eMasse:	eMasse	Double		
zeichnungsNummer:	Ebs	Integer		
name:	Teilgründung	br.gr.100		
Kürzel:	TG			
Elementklasse:	GR, B			
Besteht aus:	SW, MIP, EP			
Attribut:	Alternativname	lfcBuildingElementProxy / lfcDeepFoundation		
lfcGründung:	ID_Gruendung	String		
name:	Spundwand	br.gr.110		
Kürzel:	SW			
Elementklasse:	B			
Besteht aus:	-			
Attribut:	Alternativname	lfcBuildingElementProxy / lfcClassonFoundation		
anordnung:	Anordnung	DE; Allgemein-Daten		
lfcTeilgründung:	ID_Teilgründung	String		
name:	Mikropfahl	br.gr.120		
Kürzel:	MIP			
Elementklasse:	B			
Besteht aus:	-			
Attribut:	Alternativname	lfcBuildingElementProxy / lfcPile		
eMasse/ObjektTyp:	Einbauelement	Integer		
ObjektTyp:	Mikropfahltyp	String		
zuweisung:	Zuweisung	String		
lfcTeilgründung:	ID_Teilgründung	String		
name:	Echtpfahl	br.gr.130		
Kürzel:	EP			
Elementklasse:	B			
Besteht aus:	-			
Attribut:	Alternativname	lfcBuildingElementProxy / lfcPile		
eMasse/ObjektTyp:	Einbauelement	Integer		
ObjektTyp:	Pfahlart	String		
lfcTeilgründung:	ID_Teilgründung / lfcStuetze; ID_Teilgründung / lfc Stuetze	String		

name:	Fachquendung	br.gr:200
Kürzel	FG	
Elementklasse	GR, B	
Bezieht aus:	FP, PP	
Attribut	Alternativname	ICBuildingElementProxy /ICF-coing
id/quendung	ID_Gruendung	String
name:	Rahmenfundament	br.gr:210
Kürzel	FP	
Elementklasse	B	
Bezieht aus:	-	
Attribut	Alternativname	ICBuildingElementProxy
zsd/obersaustausch	Bodenaustausch	Boolean
zsd/verkehlung	Richtzeichnung	String
id/fachquendung	ID_Fachquendung	String

name:	Prätkopplatte	br.gr:220
Kürzel	FP	
Elementklasse	B	
Bezieht aus:	-	
Attribut	Alternativname	ICBuildingElementProxy
id/fachquendung	ID_Fachquendung	String

name:	Saubertisch	br.gr:10
Kürzel	SAS	
Elementklasse	B	
Bezieht aus:	-	
Attribut	Alternativname	ICBuildingElementProxy
id/quendung	ID_Gruendung	String

ANLAGE 3 SEMANTISCHES OBJEKTMODELL FAHRBAHN

name	Oberbau	ob	bet
Kürzel	OBB		
Elementklasse	E		
Besteht aus:	BET, G, W, Hoch, 100, ob, 200, ob, 300		
Alternativname		IceBuildingElementProxy	
Attribut	Alternativname	Fahrbahnbreite	Double
<i>Abstrakte Eigenschaft</i>		istGeschlosseneBauweise	Boolean
<i>Abstrakte Eigenschaft</i>		Oberbauart	String
<i>Abstrakte Eigenschaft</i>		Schienenrip	String
<i>Abstrakte Eigenschaft</i>		Ueberschneidungshöhe	Double

name	Beitrag	ob	bet
Kürzel	BET		
Elementklasse	AE		
Besteht aus:	.	Indirekt	
Alternativname		IceBuildingElementProxy	
Attribut	Alternativname	ID_Oberbau	String

name	Feste Fahrbahn	ob	bet, 100
Kürzel	FF		
Elementklasse	BET, B		
Besteht aus:	DF, AP, GTP, TS, GTS, HSP, SFB, TE, SA, SL		
Alternativname		IceBuildingElementProxy	
<i>Abstrakte Eigenschaft</i>		Lieferant	String
<i>Abstrakte Eigenschaft</i>		Feste-Fahrbahn-System	String

name	Dilatationsfuge	ob	bet, 110
Kürzel	DF		
Elementklasse	B		
Besteht aus:	.	IceBuildingElementProxy	
Attribut	Alternativname	ID_FesteFahrbahn	String

name	Ausgleichsplatte	ob	bet, 120
Kürzel	AP		
Elementklasse	B		
Besteht aus:	.	IceBuildingElementProxy	
Attribut	Alternativname	ID_FesteFahrbahn	String

name			
Kürzel			
Elementklasse			
Besteht aus:			
Attribut			

name	GlasTafelPlate	objekt:130	name	Tagessticht	objekt:140
Kürzel	GTP		Kürzel	TS	
Elementklasse	B		Elementklasse	B	
Bezieht aus:	-		Bezieht aus:	-	
Attribut	Alternativname	InBuildingElementProxy	Attribut	Alternativname	InBuildingElementProxy
ausgabe	Aufgabe	String	Herstellerkennung	Herstellername	String
abschreibungsgegenstand	Glasbauungssystem	String	Kennung	Kennung	String
Herstellerkennung	Herstellername	String	Herstellerkennung	Herstellername	String
Abkürzung	Plate	String	Herstellerkennung	Herstellername	String
Herstellerkennung	ID_Feststoffbahn	String	Herstellerkennung	Herstellername	String

name	Tagesschicht	objekt:140	name	GlattTrennschicht	objekt:150	name	HartschaumPlatte	objekt:180
Kurzref	TS		Kurzref	GT/S		Kurzref	HSP	
Elementklasse	E		Elementklasse	E		Elementklasse	E	
Bezieht aus:	-		Bezieht aus:	-		Bezieht aus:	-	
Attribut	Alternativname	lfdBuildingElementFrom	Attribut	Alternativname	lfdBuildingElementFrom	Attribut	Alternativname	lfdBuildingElementFrom
AccessModus	Herstellungsart	String	Attribut	ID_Faserfaehlein	String	Attribut	ID_Faserfaehlein	String
AccessModus	Koernding	String	Attribut	ID_Faserfaehlein	String	Attribut	ID_Faserfaehlein	String
ID_Faserfaehlein	ID_Faserfaehlein	String	ID_Faserfaehlein	ID_Faserfaehlein	String	ID_Faserfaehlein	ID_Faserfaehlein	String

name	HanschaumPlatte	objekt:160	name	SelbsterkennungBlock	objekt:170	name	TrennEbene	objekt:180
Kürzel	HSP		Kürzel	SFB		Kürzel	TE	
Elementklasse	B		Elementklasse	B		Elementklasse	B	
Bezeichnet aus:	-		Bezeichnet aus:	-		Bezeichnet aus:	-	
Attribut	Alternativname	IceBuildingElementProxy	Attribut	Alternativname	IceBuildingElementProxy	Attribut	Alternativname	IceBuildingElementProxy
idFestFallschein	ID_FestFallschein	String	idFestFallschein	ID_FestFallschein	String	idFestFallschein	ID_FestFallschein	String

name	Schleppentanzzug	abbet:190	name	Seitenführungslager	abbet:195
Kürzel	SA		Kürzel	SL	
Elementklasse	AE		Elementklasse	AE	
Besteht aus:	-		Besteht aus:	-	
Attribut	Alternativname	IdBuildingElementFroui	Attribut	Alternativname	IdBuildingElementFroui
<i>material</i>	Material		<i>material</i>	Material	
IdElementAnbahn	IdElementAnbahn	String	<i>materialEigenschaften</i>	MaterialEigenschaften	Material
			IdElementAnbahn	IdElementAnbahn	String

name	SchoterDierbau	ob bet 200
Kürzel	SO	
Elementklasse	BET B	
Bereit aus:	R, S, SH, USM, PS, SPS	
Attribut	Alternativname	ICBuildingElementProxy
-system	SchoterDierbausystem	String

name	Randweg	ob bet 210
Kürzel	R	
Elementklasse	B	
Bereit aus:	-	
Attribut	Alternativname	ICBuildingElementProxy
Ausprägung	Koernung	String
Ausprägung/Winkel	Nachung Winkel	Integer
ID SchoterDierbau	ID SchoterDierbau	String

name	Schoter	ob bet 220
Kürzel	S	
Elementklasse	B	
Bereit aus:	-	
Attribut	Alternativname	ICBuildingElementProxy
Ausprägung	Koernung	String
ID SchoterDierbau	ID SchoterDierbau	String

name	Schoterhaltung	ob bet 230
Kürzel	SH	
Elementklasse	B	
Bereit aus:	-	
Attribut	Alternativname	ICBuildingElementProxy
ID SchoterDierbau	ID SchoterDierbau	String

name	Unternehmensmatri	ob.bet.240	name	PlanungsSchwachsicht	ob.bet.250	name	Sperrschicht	ob.bet.260
Kürzel	USM		Kürzel	PS		Kürzel	SPS	
Elternklasse	B		Elternklasse	B		Elternklasse	B	
Bezieht aus:	-		Bezieht aus:	-		Bezieht aus:	-	
Attribut	Alternativname	IsBuildingElementProxy	Attribut	Alternativname	IsBuildingElementProxy	Attribut	Alternativname	IsBuildingElementProxy
<i>ZeitungsKnoten</i>	Modul der Beratung	String	<i>adokKurzStr</i>	Abteilungsart	String	<i>qnr</i>	Sperrschicht typ	String
<i>IsSchonDerbau</i>	ID_SchonDerbau	String	<i>Kennung</i>	Kennung	String	<i>IsSchonDerbau</i>	ID_SchonDerbau	String
			<i>Komplexisch</i>	Komplexisch	String			
			<i>Schutzbedingtheit</i>	Schutzbedingtheit	Ineger			
			<i>IsSchonDerbau</i>	ID_SchonDerbau	String			

name	Gleis	obj:gl			
Kürzel	GL				
Elementklasse	AE				
Bezieht aus:	SW, SCH, E, FS, GA, GS, SB, GQ, SIS, GRV, UV				
Attribut	Alternativname	ICBuildingElementProxy / ICFFall			
<i>spursalter</i>	Spurhalter	String			
<i>zwischenlage</i>	Zwischenlage	String			
<i>zahl</i>	YMax	Integer			
<i>idOberbau</i>	ID_Oberbau	String			
<i>adZuweisungsanlage</i>	ID_Zuweisungsanlage	String			
<i>schienenErldung</i>	Erldungsschiene	String			
<i>gleisnummer</i>	Gleisnummer	Integer			
name	Schiene	obj:gl5			
Kürzel	SW				
Elementklasse	E				
Bezieht aus:	-				
Attribut	Alternativname	ICBuildingElementProxy			
<i>schweifeTip</i>	Schweife_Tip	String			
<i>idGleis</i>	ID_Gleis	String			
<i>wahlAnlage</i>	wahl_anlage	String			
<i>idGleis</i>	ID_Gleis	String			
name	Schiene	obj:gl5			
Kürzel	SCH				
Elementklasse	E				
Bezieht aus:	-				
Attribut	Alternativname	ICBuildingElementProxy			
<i>schienenForm</i>	Form der Schiene	String			
<i>schienenStoessArt</i>	Schiene Stoessarten	String			
<i>wahlAnlage</i>	wahl_anlage	Double			
<i>idGleis</i>	ID_Gleis	String			
name	Eindeckung	obj:gl20			
Kürzel	E				
Elementklasse	E				
Bezieht aus:	-				
Attribut	Alternativname	ICBuildingElementProxy			
<i>erdeckungTip</i>	Tip der Eindeckung	String			
<i>erdeckbar</i>	Erdeckbar	Boolean			
<i>idGleis</i>	ID_Gleis	String			

name	Fangschlere	obj.gj.25	name	Gleisabschluss	obj.gj.30	name	GleisSperr	obj.gj.35
Kürzel	FS		Kürzel	GA		Kürzel	GS	
Etenkklasse	B		Etenkklasse	B		Etenkklasse	B	
Bereikt aus:	-		Bereikt aus:	-		Bereikt aus:	-	
Attribut	Alternativname	IceBuildingElementProxy	Attribut	Alternativname	IceBuildingElementProxy	Attribut	Alternativname	IceBuildingElementProxy
OBIS	ID OBIS	String	OBIS	ID OBIS	String	OBIS	ID OBIS	String

name	Sohlenbefestigung	ob.gj.40	name	Gliedsequenz	ob.gj.45	name	Sohlenstosserstoss	ob.gj.50
Kürzel	SB		Kürzel	GIQ		Kürzel	SIS	
Elementklasse	B		Elementklasse	B		Elementklasse	B	
Bereicht aus:	.		Bereicht aus:	.		Bereicht aus:	.	
Attribut	Alternativname	IsBuildingElementProxy	Attribut	Alternativname	IsBuildingElementProxy	Attribut	Alternativname	IsBuildingElementProxy
sohnenbefestigungstyp	sohnenbefestigungstyp	String	sohnenbefestigungstyp	sohnenbefestigungstyp	String	sohnenbefestigungstyp	sohnenbefestigungstyp	String
idGleis	ID_Gleis	String	idGleis	ID_Gleis	String	idGleis	ID_Gleis	String

name	Gleisrostverstaerkung	ob:gl:55	name	Unterhaltsweg	ob:gl:60	name	GleisReferenzPunkt	ob:gl:55
Kürzel	GRV		Kürzel	UW		Kürzel	GRP	
Elementklasse	E		Elementklasse	E		Elementklasse	AE	
Besteht aus:	-		Besteht aus:	-		Besteht aus:	-	
Attribut	Alternativname	IceBuildingElementProxy	Attribut	Alternativname	IceBuildingElementProxy	Attribut	Alternativname	IceBuildingElementProxy
gdb:source/Verstaerkung_Zyp	Gleisrostverstaerkung	String	gdb:source/Verstaerkung_Zyp	Gleisrostverstaerkung	String	gdb:source/Verstaerkung_Zyp	GleisBezeichnung	String
IdGleis	IdGleis	String	IdGleis	IdGleis	String	IdGleis	km	Integer
						km	Koordinaten	Koordinaten
						Radius	Radius des Gleises	Double
						Abkennung	Richtung	String
						IdGleis	IdGleis	String

name	Welche	ob.w	name	WelcherBefestigung	ob.w 15	name	Welcherschwellen	ob.w 20
name	Welche	ob.w	name	WelcherBefestigung	ob.w 15	name	Welcherschwellen	ob.w 20
Kürzel	AE		Kürzel	WB		Kürzel	VS	
Elementklasse	SUVS, WB, WS, ZUV, ZV, WG, SG, ZG, ZST, ET	IEBuildingElementProxy	Elementklasse	B		Elementklasse	B	
Attribut	Alternativname		Attribut	Alternativname	IEBuildingElementProxy	Attribut	Alternativname	IEBuildingElementProxy
abzweigRichtung	abzweigRichtung	String	abzweigRichtung	abzweigRichtung	String	abzweigRichtung	abzweigRichtung	String
abzweigRichtung_senige	Laenge des abzweigenden Gleises	Double	abzweigRichtung_senige	Laenge des abzweigenden Gleises	Double	abzweigRichtung_senige	Laenge des abzweigenden Gleises	Double
abschl_schraubeAnkennung	Achslastverschraenkungen	String	abschl_schraubeAnkennung	Achslastverschraenkungen	String	abschl_schraubeAnkennung	Achslastverschraenkungen	String
antrieb	antrieb	String	antrieb	antrieb	String	antrieb	antrieb	String
durchdringung_senige	Laenge des durchdringenden Gleises	Double	durchdringung_senige	Laenge des durchdringenden Gleises	Double	durchdringung_senige	Laenge des durchdringenden Gleises	Double
endbleibung	endbleibung	Double	endbleibung	endbleibung	Double	endbleibung	endbleibung	Double
istRueckfallbleiche	istRueckfallbleiche	Boolean	istRueckfallbleiche	istRueckfallbleiche	Boolean	istRueckfallbleiche	istRueckfallbleiche	Boolean
schleneBefestigung_Typ	schleneBefestigung_Typ	String	schleneBefestigung_Typ	schleneBefestigung_Typ	String	schleneBefestigung_Typ	schleneBefestigung_Typ	String
schmierStoff	schmierStoff	String	schmierStoff	schmierStoff	String	schmierStoff	schmierStoff	String
stammgleisRadius	stammgleisRadius	Double	stammgleisRadius	stammgleisRadius	Double	stammgleisRadius	stammgleisRadius	Double
vZu	vZu	Double	vZu	vZu	Double	vZu	vZu	Double
vZuabzweig	vZuabzweig	Double	vZuabzweig	vZuabzweig	Double	vZuabzweig	vZuabzweig	Double
welcherForm	welche Form	String	welcherForm	welche Form	String	welcherForm	welche Form	String
welcheSkizzeW	welcheSkizzeW	Integer	welcheSkizzeW	welcheSkizzeW	Integer	welcheSkizzeW	welcheSkizzeW	Integer
welcheVerlegetechnik	welcheVerlegetechnik	Integer	welcheVerlegetechnik	welcheVerlegetechnik	Integer	welcheVerlegetechnik	welcheVerlegetechnik	Integer
zweiggleisRadius	zweiggleisRadius	Double	zweiggleisRadius	zweiggleisRadius	Double	zweiggleisRadius	zweiggleisRadius	Double
ID:barbau	ID:barbau		ID:barbau	ID:barbau		ID:barbau	ID:barbau	
name	StellundVerschlussystem	ob.w 10	name	WelcherBefestigung	ob.w 15	name	Welcherschwellen	ob.w 20
Elementklasse	SUVS		Elementklasse	WB		Elementklasse	VS	
Attribut	Alternativname	IEBuildingElementProxy	Attribut	Alternativname	IEBuildingElementProxy	Attribut	Alternativname	IEBuildingElementProxy
ID:weiche	ID:weiche	String	ID:weiche	ID:weiche	String	ID:weiche	ID:weiche	String
name	StellundVerschlussystem	ob.w 10	name	WelcherBefestigung	ob.w 15	name	Welcherschwellen	ob.w 20
Elementklasse	AE		Elementklasse	B		Elementklasse	B	
Attribut	Alternativname	IEBuildingElementProxy	Attribut	Alternativname	IEBuildingElementProxy	Attribut	Alternativname	IEBuildingElementProxy
ID:weiche	ID:weiche	String	ID:weiche	ID:weiche	String	ID:weiche	ID:weiche	String
name	StellundVerschlussystem	ob.w 10	name	WelcherBefestigung	ob.w 15	name	Welcherschwellen	ob.w 20
Elementklasse	SUVS		Elementklasse	WB		Elementklasse	VS	
Attribut	Alternativname	IEBuildingElementProxy	Attribut	Alternativname	IEBuildingElementProxy	Attribut	Alternativname	IEBuildingElementProxy
ID:weiche	ID:weiche	String	ID:weiche	ID:weiche	String	ID:weiche	ID:weiche	String
name	StellundVerschlussystem	ob.w 10	name	WelcherBefestigung	ob.w 15	name	Welcherschwellen	ob.w 20
Elementklasse	SUVS		Elementklasse	WB		Elementklasse	VS	
Attribut	Alternativname	IEBuildingElementProxy	Attribut	Alternativname	IEBuildingElementProxy	Attribut	Alternativname	IEBuildingElementProxy
ID:weiche	ID:weiche	String	ID:weiche	ID:weiche	String	ID:weiche	ID:weiche	String

name	Zungenlaufvorrichtung	ob.w.25	name	Zungenvorrichtung	ob.w.30	name	WeichheitGrenzzeichen	ob.w.35
Kürzel	ZUV		Kürzel	ZV		Kürzel	WG	
Elementklasse	AE	-	Elementklasse	AE	-	Elementklasse	B	-
Bezeichnet aus:	-		Bezeichnet aus:	-		Bezeichnet aus:	-	
Attribut	Alternativname	ISOBuildingElementProxy	Attribut	Alternativname	ISOBuildingElementProxy	Attribut	Alternativname	ISOBuildingElementProxy
Id/Weiche	ID/Weiche	String	Id/Weiche	ID/Weiche	String	Id/Weiche	ID/Weiche	String

name	Stammglets	ob.w.40	name	Zweiggleis	ob.w.45	name	Zwischenschienenstell	ob.w.50
Eitenklasse Kürzel	SG AE		Eitenklasse Kürzel	ZG AE		Eitenklasse Kürzel	ZST AE	
Besteht aus:	-		Besteht aus:	-		Besteht aus:	-	
Attribut	Alternativname id/Weiche String	IceBuildingElementProxy	Attribut	Alternativname id/Weiche String	IceBuildingElementProxy	Attribut	Alternativname id/Weiche String	IceBuildingElementProxy
id/Weiche	id/Weiche String		id/Weiche	id/Weiche String		id/Weiche	id/Weiche String	

name	Endteil	obw.55
Kürzel	ET	
Elementklasse	AE	
Bezeichnet aus:	-	
Attribut	Alternativname	!toBuildingElementFrom
IdVerbleibe	!ID_Verbleibe	String

ANLAGE 4 SEMANTISCHES OBJEKTMODELL SCHNELLÜBERSICHT

Name	Kürzel	bauteilKlassifikation	Elterklasse	Vererbungskette	Bestandteile
Allgemeine Eigenschaft	AE	-	-	-	-
Bauteil	B	-	AE	AE	-
Einrichtung	E	-	AE	AE	-
bauteilName	Kürzel	bauteilKlassifikation	Elterklasse	Vererbungskette	Bestandteile
Brücke	BR	br	E	E, AE	BET(FB), UE, UB, GR, ASR
Ueberbau	UE	br.ue	AE	AE	BTW, KA
Brueckentragwerk	BTW	br.ue.10	AE	AE	BRP, BRT
Brueckenplatte	BRP	br.ue.11	B	B, AE	-
Rahmendecke	RD	br.ue11.100	BRP	BRP, B, AE	-
Brückenträger	BRT	br.ue12	B	B, AE	-
Kappe	KA	br.ue20	B	B, AE	KT, FK
Kapbeltrog	KT	br.ue21	B	B, AE	-
Randkappe	FK	br.ue22	B	B, AE	-
Unterbau	UB	br.ub	AE	AE	WL
Widerlager	WL	br.ub.10	B	B, AE	WW, FW, WLF, WHF
Widerlagerwand	WW	br.ub.11	B	B, AE	-
Fluegelwand	FW	br.ub.12	B	B, AE	-
Widerlagerfundament	WLF	br.ub.13	B	B, AE	-
Widerlagerhinterfuellung	WHF	br.ub.14	B	B, AE	-
Stuetze	STU	br.ub.20	B	B, AE	-
Gruendung	GR	br.gr	AE	AE	TG, FG, SAS
Tiefgruendung	TG	br.gr.100	B, GR	B, AE, GR	SW, MP, BP
Spundwand	SW	br.gr.110	B	B, AE	-
Mikropfahl	MP	br.gr.120	B	B, AE	-
Bohrpfahl	BP	br.gr.130	B	B, AE	-
Flachgruendung	FG	br.gr.200	B, GR	B, AE, GR	RF, PP
Rahmenfundament	RF	br.gr.210	B	B, AE	-
Pfahlkopfplatte	PP	br.gr.220	B	B, AE	-
Sauberkeitschicht	SAS	br.gr.10	B	B, AE	-
Ausruestung	ASR	br.asr	AE	AE	GEL, ABD, LSW, BFE, EWS, STS, BWS, FUE, MK, APS, LG, ASB
Gelaender	GEL	br.asr.10	B	B, AE	-
Abdichtung	ABD	br.asr.15	B	B, AE	-
Laermeschutzwand	LSW	br.asr.20	B	B, AE	-
Befestigungselement	BFE	br.asr.25	B	B, AE	-
Entwaesserungseinrichtung	EWS	br.asr.30	B	B, AE	-
Steuerstab	STS	br.asr.35	B	B, AE	-
Bewehrungsschutz	BWS	br.asr.40	B	B, AE	-
Fahrbahnuebergang	FUE	br.asr.45	B	B, AE	-
Anprallschutz	APS	br.asr.50	B	B, AE	-
Lager	LG	br.asr.55	B	B, AE	-
Ausruestungsbalken	ASB	br.asr.60	B	B, AE	-
Baubeheif	BBH	br.bbh	AE	AE	VS, VB
Verschubsystem	VS	br.bbh.10	B	B, AE	-
Verbau	VB	br.bbh.20	B	B, AE	-

Name	Kürzel	bauteilKlassifikation	Elterklasse	Vererbungskette	Bestandteile
Oberleitungsbauteil	OB	-	B	B, AE	-
Oberleitungsanlage	OLA	ola	E	E, AE	TW, SUNG, FL, ERD, KL, WE, GL(ob.gl)
Tragwerk	TW	ola.tw	AE	AE	-
Trageinrichtung	TE	ola.tw.100	TW	TW, AE	-
Mast	M	ola.tw.100.100	TE, OB	TE, TW, AE, OB, B	GR, TR
Mastkonsole	MK	ola.tw.100.110	OB, GP(br.gr)	OB, B, AE, GR	-
Traverse	TR	ola.tw.100.120	OB	OB, B, AE	-
BauwerksAufhaengung	BA	ola.tw.100.200	TE, OB	TE, TW, AE, OB, B	HS
Haengesaeule	HS	ola.tw.100.210	OB	OB, B, AE	-
Quertrageinrichtung	QE	ola.tw.200	TW	TW, AE	-
Mehrgleislausleger	MAG	ola.tw.200.100	QE, OB	QE, TW, AE, OB, B	-
Joch	JO	ola.tw.200.200	QE, OB	QE, TW, AE, OB, B	-
Querfeld	QF	ola.tw.200.300	QE, OB	QE, TW, AE	IQ, QT, RA
IsolationQuerfeld	IQ	ola.tw.200.310	OB	OB, B, AE	-
QuerfeldTragsseil	QT	ola.tw.200.320	OB	OB, B, AE	-
Richtseilabschnitt	RA	ola.tw.200.330	OB	OB, B, AE	-
Schaltung	SUNG	ola.sung	AE	AE	ST, OS, VBE
Schalter	ST	ola.sung.10	OB	OB, B, AE	-
Oberleitungsseparator	OS	ola.sung.20	OB	OB, B, AE	-
Verbinder	VBE	ola.sung.30	OB	OB, B, AE	-
Fahrleitung	FL	ola.fl	AE	AE	AL, QSP, BA, QH, AV, OS, QF
Aussteiger	AL	ola.fl.10	OB	OB, B, AE	-
QuerfeldStuetzpunkt	QSP	ola.fl.20	OB	OB, B, AE	-
BogenAbzug	BA	ola.fl.30	OB	OB, B, AE	-
QuerfeldHaenger	QH	ola.fl.40	OB	OB, B, AE	-
AbspannVorrichtung	AV	ola.fl.50	OB	OB, B, AE	-
OberleitungsSegment	OS	ola.fl.60	OB	OB, B, AE	-
OberleitungsFuehrung	QF	ola.fl.70	OB	OB, B, AE	-
Erdung	ERD	ola.erd	AE	AE	-
Kabelleitungen	KL	ola.kl	AE	AE	K, L
Kabel	K	ola.kl.10	OB	OB, B, AE	-
Leitung	L	ola.kl.20	OB	OB, B, AE	-
Warneinrichtung	WE	ola.we	AE	AE	SD, ELS
Schild	SD	ola.we.10	OB, WE	OB, B, AE, WE	-
ElSignal	ELS	ola.we.20	OB, WE	OB, B, AE, WE	-

Name	Kürzel	bauteilKlassifikation	Elterklasse	Vererbungskette	Bestandteile
Oberbau	OBB	ob	E	E, AE	BET, GL, W
Bettung	BET	ob.bet	AE	AE	-
FesteFahrbahn	FF	ob.bet.100	BET, B	BET, AE, B	DF, AP, GTP, TS, GTS, HSP, SFB, TE, SA, SL
Dilatationsfuge	DF	ob.bet.110	B	B, AE	-
AusgleichsPlatte	AP	ob.bet.120	B	B, AE	-
GleisTragPlatte	GTP	ob.bet.130	B	B, AE	-
Tragschicht	TS	ob.bet.140	B	B, AE	-
Gleitrennschicht	GTS	ob.bet.150	B	B, AE	-
Hartschaumplatte	HSP	ob.bet.160	B	B, AE	-
Seitenfuehrungsblock	SFB	ob.bet.170	B	B, AE	-
Trennebene	TE	ob.bet.180	B	B, AE	-
Schieneauszug	SA	ob.bet.190	AE	AE	-
Seitenfuehrungslager	SL	ob.bet.195	AE	AE	-
Schotteroberbau	SO	ob.bet.200	BET, B	BET, AE, B	R, S, SH, USM, PS, SPS
Randweg	R	ob.bet.210	B	B, AE	-
Schotter	S	ob.bet.220	B	B, AE	-
Schotterhalterung	SH	ob.bet.230	B	B, AE	-
Unterschottermatte	USM	ob.bet.240	B	B, AE	-
Planumsschutzschicht	PS	ob.bet.250	B	B, AE	-
Sperrschicht	SPS	ob.bet.260	B	B, AE	-
Gleis	GL	ob.gl	AE	AE	SW, SCH, E, FS, GA, GS, SB, GQ, SIS, GRV, UW
Schwelle	SW	ob.gl.10	B	B, AE	-
Schiene	SCH	ob.gl.15	B	B, AE	-
Eindeckung	EI	ob.gl.20	B	B, AE	-
Fangschiene	FS	ob.gl.25	B	B, AE	-
Gleisabschluss	GA	ob.gl.30	B, LBO	B, AE, LBO	-
Gleissperre	GS	ob.gl.35	B	B, AE	-
Schienebefestigung	SB	ob.gl.40	B	B, AE	-
Gleisquerung	GQ	ob.gl.45	B	B, AE	-
Schieneisolierstoss	SIS	ob.gl.50	B	B, AE	-
Gleisrostverstaerkung	GRV	ob.gl.55	B	B, AE	-
Unterhaltsweg	UW	ob.gl.60	B	B, AE	-
Gleisreferenzpunkt	GRF	ob.gl.65	AE	AE	-
Weiche	W	ob.w	AE	AE	SUVS, WB, WS, ZUV, ZV, WG, SG, ZG, ZST, ET
StellUndVerschlussystem	SUVS	ob.w.10	AE	AE	-
Weichenbefestigung	WB	ob.w.15	B	B, AE	-
Weichenschwellen	WS	ob.w.20	B	B, AE	-
Zungeumlaufvorrichtung	ZUV	ob.w.25	AE	AE	-
Zungenvorrichtung	ZV	ob.w.30	AE	AE	-
WeicheGrenzzeichen	WG	ob.w.35	B	B, AE	-
Stammgleis	SG	ob.w.40	AE	AE	-
Zweiggleis	ZG	ob.w.45	AE	AE	-
Zwischenschienenteil	ZST	ob.w.50	AE	AE	-
Endteil	ET	ob.w.55	AE	AE	-

Name	Kürzel	bauteilKlassifikation	Elterklasse	Vererbungskette	Bestandteile
LSTBasisObjekt	LBO	-	AE	AE	-
LSTBereichsobjekt	LBerO	-	LBO	LBO, AE	-
LSTPunktobjekt	LPO	-	LBO	LBO, AE	-
LSTAnlage	LST	Ist	AE	AE	-
Bahnsteige	BS	Ist.bs	AE	AE	BSA
Bahnsteiganlage	BSA	Ist.bs.10	LBO, LST	LBO, AE, LST	BSZ
Bahnsteigzugang	BSZ	Ist.bs.11	B, LPO	B, AE, LPO, LBO	-
Signale	SGE	Ist.sge	AE	AE	SIG, SBF
Signal	SIG	Ist.sge.10	LPO, LST	LPO, LBO, AE, LST	SIR
Signalrahmen	SIR	Ist.sge.11	B, LBO	B, AE, LBO	-
Signalbefestigung	SBF	Ist.sge.20	B, LPO, BFE(aus.125)	B, AE, LPO, LBO, BFE	-
PZB	PZB	Ist.pzb	AE	AE	PZBE
PZBElement	PZBE	Ist.pzb.10	LPO	LPO, LBO, AE	-
PZBElementGM	PZBEGM	Ist.pzb.10.100	PZBE	PZBE, LPO, LBO, AE	-
PZBElementGUE	PZBEGUE	Ist.pzb.10.200	PZBE	PZBE, LPO, LBO, AE	-
Block	BLC	Ist.blc	AE	AE	BLCA
Blockanlage	BLCA	Ist.blc.10	LST, LBO	LST, AE, LBO	BLCE
Blockelement	BLCE	Ist.blc.11	B, LBO	B, AE, LBO	-
Ortung	ORT	Ist.ort	AE	AE	FMAA, ZEW
FMAAnlage	FMAA	Ist.ort.10	LST, LBO	LST, AE, LBO	FMAK
FMAKomponente	FMAK	Ist.ort.11	LPO	LPO, LBO, AE	-
Achszählpunkt	AZP	Ist.ort.11.100	FMAK, B	FMAK, LPO, LBO, AE, B	-
Zugewirkung	ZEW	Ist.ort.20	B, LPO	B, AE, LPO, LBO	-
Schlüsselabhängigkeit	SAH	Ist.sah	AE	AE	SCHL, SLS, SLK
Schloss	SCHL	Ist.sah.10	B, LBO	B, AE, LBO	-
Schlüsselsperre	SLS	Ist.sah.20	B, LBO	B, AE, LBO	-
Schlosskomponente	SCHLK	Ist.sah.30	B, LBO	B, AE, LBO	-
Bedienung	BED	Ist.bed	AE	AE	BEDOE
Bedienungseinrichtung	BEDOE	Ist.bed.10	LST, LBO	LST, AE, LBO	BEDAE
Bedienanzeigeelement	BEDAE	Ist.bed.11	B, LBO	B, AE, LBO	-
Ansteuerungselement	ASE	Ist.ase	AE	AE	UBR
Unterbringung	UBR	Ist.ase.10	B, LBO	B, AE, LBO	-
WeichenUndGleissperrerr	WUG	Ist.wug	AE	AE	WKA, GA (ob.230)
WKAAnlage	WKA	Ist.wug.10	LST, LBO	LST, AE, LBO	WKGE
WKAElement	WKGE	Ist.wug.11	LBO	LBO, AE	WKGK
WKAKomponente	WKGK	Ist.wug.11.10	LPO	LPO, LBO, AE	-
Kreuzung	KRZ	Ist.wug.11.10.100	B, WKGK	B, AE, WKGK, LPO, LBO	-
Zungenpaar	ZP	Ist.wug.11.10.200	B, WKGK	B, AE, WKGK, LPO, LBO	-
Ertgleisungsschuh	EGS	Ist.wug.11.10.300	B, WKGK	B, AE, WKGK, LPO, LBO	-
besonderesFahrwerk	BF	Ist.wug.11.10.400	B, WKGK	B, AE, WKGK, LPO, LBO	-
Bahnuebergaenge	BUG	Ist.bug	AE	AE	BUEA
BUEAnlage	BUEA	Ist.bug.10	LST, LPO	LST, AE, LPO, LBO	BUEAS, VKZ, SRA, GFRA, BUES
BUEAnlageStrasse	BUEAS	Ist.bug.11	B, LBO	B, AE, LBO	-
Verkehrszeichen	VKZ	Ist.bug.12	B, LBO	B, AE, LBO	-
Schrankenbetrieb	SRA	Ist.bug.13	B, LBO	B, AE, LBO	-
GFRAAnlage	GFRA	Ist.bug.14	LBO	LBO, AE	GFRE, GFRT
GFRElement	GFRE	Ist.bug.14.10	B, LBO	B, AE, LBO	-
GFRTripelspiegel	GFRT	Ist.bug.14.20	B, LBO	B, AE, LBO	-
Ueberwachungssignal	UES	Ist.bug.15	LST, LPO	LST, AE, LPO, LBO	SIG
BUESchnittstelle	BUES	Ist.bug.16	B, LBO	B, AE, LBO	-
MedienUndTrassen	MUT	Ist.mut	AE	AE	TKA, TKO, KAB, KVP
TrasseKante	TKA	Ist.mut.10	B, LBO	B, AE, LBO	-
TrasseKnoten	TKO	Ist.mut.20	B, LBO	B, AE, LBO	-
LSTKabel	KAB	Ist.mut.30	B, LBO	B, AE, LBO	-
KabelVerteilpunkt	KVP	Ist.mut.40	B, LBO	B, AE, LBO	-
TOPKante	TOPKante	Ist.geo.10	LBO	LBO, AE	-
GEOKante	GEOKante	Ist.geo.20	LBO	LBO, AE	-
Strecke	Strecke	Ist.geo.30	LBerO	LBerO, LBO, AE	-
TOPKnoten	TOPKnoten	Ist.geo.40	LBO	LBO, AE	-
GEOKnoten	GEOKnoten	Ist.geo.50	LBO	LBO, AE	-
StreckePunkt	StreckePunkt	Ist.geo.60	LBO	LBO, AE	-
GEOPunkt	GEOPunkt	Ist.geo.70	LBO	LBO, AE	-
Oertlichkeit	Oertlichkeit	Ist.geo.80	LBO	LBO, AE	-

ANLAGE 5 BIMQ-EXPORT BRM1 SOLIBRI

Component	Property Set	Property	Value Exists	Value Type
Objekt	Allgemeine_Eigenschaften	bauteilKlassifikation	Must exist	Text
Objekt	Allgemeine_Eigenschaften	bauteilName	Must exist	Text
Objekt	Allgemeine_Eigenschaften	bemerkung	Must exist	Text
Objekt	Allgemeine_Eigenschaften	identitaet	Must exist	Text
Objekt	Allgemeine_Eigenschaften	KGK-Verzeichnisnummer	Must exist	Text
Objekt	Allgemeine_Eigenschaften	name	Must exist	Text
Objekt	UE	idBruecke	Must exist	Text
BR.ue	Ueberbau			
Objekt	Allgemeine_Eigenschaften	bauteilKlassifikation	Must exist	Text
Objekt	Allgemeine_Eigenschaften	bauteilName	Must exist	Text
Objekt	Allgemeine_Eigenschaften	bemerkung	Must exist	Text
Objekt	Allgemeine_Eigenschaften	identitaet	Must exist	Text
Objekt	Allgemeine_Eigenschaften	KGK-Verzeichnisnummer	Must exist	Text
Objekt	Allgemeine_Eigenschaften	name	Must exist	Text
Objekt	STU	idUnterbau	Must exist	Text
Objekt	STU	stuetzenart	Must exist	Text
Objekt	Bauteil	betriebszustand	Must exist	Text
Objekt	Bauteil	material	Must exist	Text
Objekt	Bauteil	materialEigenschaften	Must exist	Text
Objekt	Bauteil	technischerPlatz	Must exist	Text
Objekt	Bauteil	technischerPlatzBezeichnung	Must exist	Text
BR.ub.20	Stuetze			

ANLAGE 6 BIMQ-EXPORT BRM1 DESITE

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<qExport xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" user="Dominik Nils Sonnek"  
date="2021-09-02 12:45:53 +0200">
```

```
<elementSection>  
  <checkrun ID="95691d58-67b9-4fbb-b65e-e30def5af4aa"  
    name="neue Richtlinie 4g1_HDWB - 20210312" active="true" user="" date="" state="0"  
    objectsOnly="1" partsOfComposites="0" createFailed="true" createWarnings="true"  
    createIgnored="true" createPassed="true" createUndefined="true">  
    <rule type="Attributes">  
      <attributeRuleList>  
        <ruleScript name="Lph-1-BRM_1:Ueberbau" active="true" resume="false">  
          <code><![CDATA[  
var id = desiteThis.ID();  
var isContainer = desiteAPI.getPropertyValue(id, 'cpIsContainer', 'xs:boolean');  
var isComposite = desiteAPI.getPropertyValue(id, 'cpIsComposite', 'xs:boolean');  
  
if (isContainer == true && isComposite == false) {  
  desiteResult.setCheckState('ignored');  
  desiteResult.addMessage('Container was ignored.');} else {  
  var valueIfcType = desiteAPI.getPropertyValue(id, 'ifcType', 'xs:string');  
  var propFilter1 = desiteAPI.getPropertyValue(id, 'Allgemeine_Eigenschaften:bauteilKlassifikation', 'xs:string');  
  if (valueIfcType == 'IfcBuildingElementProxy' && propFilter1 == 'br.ue') {  
    var checkfailed = 0;  
    var text1 = 'Eigenschaft *';  
    var text2 = '* nicht vorhanden!';  
    var text3 = '* besitzt nicht den richtigen Wert!';  
    var text4 = ' -> ';  
  
    var sBIMQConceptName1 = 'bauteilKlassifikation';  
    var svalue1 = 'Allgemeine_Eigenschaften:bauteilKlassifikation';  
    var value1 = desiteAPI.getPropertyValue(id, svalue1, 'xs:string');  
    if (value1 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue1 + text2);  
  }  
  
    var sBIMQConceptName2 = 'bauteilName';  
    var svalue2 = 'Allgemeine_Eigenschaften:bauteilName';  
    var value2 = desiteAPI.getPropertyValue(id, svalue2, 'xs:string');  
    if (value2 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue2 + text2);  
  }  
  
    var sBIMQConceptName3 = 'bemerkung';  
    var svalue3 = 'Allgemeine_Eigenschaften:bemerkung';  
    var value3 = desiteAPI.getPropertyValue(id, svalue3, 'xs:string');  
    if (value3 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue3 + text2);  
  }  
  
    var sBIMQConceptName4 = 'identitaet';  
    var svalue4 = 'Allgemeine_Eigenschaften:identitaet';  
    var value4 = desiteAPI.getPropertyValue(id, svalue4, 'xs:string');  
    if (value4 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue4 + text2);  
  }  
  
    var sBIMQConceptName5 = 'KKG-Verzeichnisnummer';  
    var svalue5 = 'Allgemeine_Eigenschaften:KKG-Verzeichnisnummer';  
    var value5 = desiteAPI.getPropertyValue(id, svalue5, 'xs:string');  
    if (value5 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue5 + text2);  
  }  
  
    var sBIMQConceptName6 = 'name';  
    var svalue6 = 'Allgemeine_Eigenschaften:name';  
    var value6 = desiteAPI.getPropertyValue(id, svalue6, 'xs:string');  
    if (value6 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue6 + text2);  
  }  
  
    var sBIMQConceptName7 = 'idBruecke';
```

```

var svalue7 = 'UE:idBruecke';
var value7 = desiteAPI.getPropertyValue(id,svalue7,'xs:string');
if (value7 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue7 + text2);
}

var maxAtt = 7;

if (checkfailed == 0) desiteResult.setCheckState('passed');

if (maxAtt == checkfailed) {
    desiteResult.setCheckState('failed');
    desiteResult.addMessage('Keine der geforderten Eigenschaften vorhanden!');
}

if (checkfailed < maxAtt && checkfailed != 0) {
    desiteResult.setCheckState('warning');
    desiteResult.addMessage(checkfailed + ' von ' + maxAtt + ' Eigenschaft(en) nicht oder
nicht wie gefordert vorhanden!');
}
} else {
    desiteResult.setCheckState('ignored');
}
}
]]></code>
</ruleScript>
<ruleScript name="Lph-1-BRM_1:Stuetze" active="true" resume="false">
<code><![CDATA[
var id = desiteThis.ID();
var isContainer = desiteAPI.getPropertyValue(id,'cpIsContainer','xs:boolean');
var isComposite = desiteAPI.getPropertyValue(id,'cpIsComposite','xs:boolean');

if (isContainer == true && isComposite == false) {
    desiteResult.setCheckState('ignored');
    desiteResult.addMessage('Container was ignored. ');
} else {
    var valueIfcType = desiteAPI.getPropertyValue(id,'ifcType','xs:string');
    var propFilter1 = desiteAPI.getPropertyValue(id, 'Allgemeine_Eigenschaften:bauteilKlassifikation', 'xs:string');
    if (valueIfcType == 'IfcBuildingElementProxy' && propFilter1 == 'br.ub.20') {
        var checkfailed = 0;
        var text1 = 'Eigenschaft *';
        var text2 = '* nicht vorhanden!';
        var text3 = '* besitzt nicht den richtigen Wert!';
        var text4 = ' -> ';

        var sBIMQConceptName1 = 'bauteilKlassifikation';
        var svalue1 = 'Allgemeine_Eigenschaften:bauteilKlassifikation';
        var value1 = desiteAPI.getPropertyValue(id,svalue1,'xs:string');
        if (value1 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue1 + text2);
    }

    var sBIMQConceptName2 = 'bauteilName';
    var svalue2 = 'Allgemeine_Eigenschaften:bauteilName';
    var value2 = desiteAPI.getPropertyValue(id,svalue2,'xs:string');
    if (value2 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue2 + text2);
    }

    var sBIMQConceptName3 = 'bemerkung';
    var svalue3 = 'Allgemeine_Eigenschaften:bemerkung';
    var value3 = desiteAPI.getPropertyValue(id,svalue3,'xs:string');
    if (value3 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue3 + text2);
    }

    var sBIMQConceptName4 = 'identitaet';
    var svalue4 = 'Allgemeine_Eigenschaften:identitaet';
    var value4 = desiteAPI.getPropertyValue(id,svalue4,'xs:string');
    if (value4 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue4 + text2);
    }

    var sBIMQConceptName5 = 'KGK-Verzeichnisnummer';
    var svalue5 = 'Allgemeine_Eigenschaften:KGK-Verzeichnisnummer';
    var value5 = desiteAPI.getPropertyValue(id,svalue5,'xs:string');

```

```

    if (value5 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue5 + text2);
}

var sBIMQConceptName6 = 'name';
var svalue6 = 'Allgemeine_Eigenschaften:name';
var value6 = desiteAPI.getPropertyValue(id,svalue6,'xs:string');
if (value6 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue6 + text2);
}

var sBIMQConceptName7 = 'idUnterbau';
var svalue7 = 'STU:idUnterbau';
var value7 = desiteAPI.getPropertyValue(id,svalue7,'xs:string');
if (value7 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue7 + text2);
}

var sBIMQConceptName8 = 'stuetzenart';
var svalue8 = 'STU:stuetzenart';
var value8 = desiteAPI.getPropertyValue(id,svalue8,'xs:string');
if (value8 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue8 + text2);
}

var sBIMQConceptName9 = 'betriebszustand';
var svalue9 = 'Bauteil:betriebszustand';
var value9 = desiteAPI.getPropertyValue(id,svalue9,'xs:string');
if (value9 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue9 + text2);
}

var sBIMQConceptName10 = 'material';
var svalue10 = 'Bauteil:material';
var value10 = desiteAPI.getPropertyValue(id,svalue10,'xs:string');
if (value10 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue10 +
text2); }

var sBIMQConceptName11 = 'materialEigenschaften';
var svalue11 = 'Bauteil:materialEigenschaften';
var value11 = desiteAPI.getPropertyValue(id,svalue11,'xs:string');
if (value11 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue11 +
text2); }

var sBIMQConceptName12 = 'technischerPlatz';
var svalue12 = 'Bauteil:technischerPlatz';
var value12 = desiteAPI.getPropertyValue(id,svalue12,'xs:string');
if (value12 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue12 +
text2); }

var sBIMQConceptName13 = 'technischerPlatzBezeichnung';
var svalue13 = 'Bauteil:technischerPlatzBezeichnung';
var value13 = desiteAPI.getPropertyValue(id,svalue13,'xs:string');
if (value13 == undefined) { checkfailed++; desiteResult.addMessage(text1 + svalue13 +
text2); }

var maxAtt = 13;

if (checkfailed == 0) desiteResult.setCheckState('passed');

if (maxAtt == checkfailed) {
    desiteResult.setCheckState('failed');
    desiteResult.addMessage('Keine der geforderten Eigenschaften vorhanden!');
}

if (checkfailed < maxAtt && checkfailed != 0) {
    desiteResult.setCheckState('warning');
    desiteResult.addMessage(checkfailed + ' von ' + maxAtt + ' Eigenschaft(en) nicht oder
nicht wie gefordert vorhanden!');
}
} else {
    desiteResult.setCheckState('ignored');
}
}
]]></code>
    </ruleScript>
</attributeRuleList>
</rule>

```

```
</checkrun>  
</elementSection>  
</qaExport>
```

ANLAGE 7 BIMQ-EXPORT BRM1 BIMCOLLAB ZOOM

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<bimcollabsmartviewfile>  
  <version>5</version>  
  <applicationversion>Win - Version: 3.2 (build 3.2.13.312)</applicationversion>  
</bimcollabsmartviewfile>
```

```
<SMARTVIEWSETS>
```

```
  <SMARTVIEWSET>
```

```
    <TITLE>Objektprüfung - Brueckenmodell_1</TITLE>
```

```
    <DESCRIPTION>Prüfung für folgende Einstellungen:
```

```
  - Lph-1-BRM_1 BRM_1
```

```
</DESCRIPTION>
```

```
  <GUID>9b5710a2-f61f-4199-9e07-240ce2de12da</GUID>
```

```
  <MODIFICATIONDATE>2021-07-12 11:36:21 +0200</MODIFICATIONDATE>
```

```
  <SMARTVIEWS>
```

```
    <SMARTVIEW>
```

```
      <TITLE>BR.ue Ueberbau</TITLE>
```

```
      <DESCRIPTION>Elternklasse: AE
```

```
Vererbungskette: AE : Ueberbau : Bruecken</DESCRIPTION>
```

```
      <CREATOR>Dominik Nils Sonnek</CREATOR>
```

```
      <CREATIONDATE>2021-07-12 11:30:17 +0200</CREATIONDATE>
```

```
      <MODIFIER>Dominik Nils Sonnek</MODIFIER>
```

```
      <MODIFICATIONDATE>2021-09-02 11:14:25 +0200</MODIFICATIONDATE>
```

```
      <GUID>40cc0f20-17f6-4614-b8f0-4ef5cddfeda</GUID>
```

```
      <RULES>
```

```
        <RULE>
```

```
          <IFCTYPE>Any</IFCTYPE>
```

```
          <PROPERTY>
```

```
            <NAME>IFC Element</NAME>
```

```
            <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
```

```
            <TYPE>Summary</TYPE>
```

```
            <VALUETYPE>StringValue</VALUETYPE>
```

```
            <UNIT>None</UNIT>
```

```
          </PROPERTY>
```

```
          <CONDITION>
```

```
            <TYPE>Is</TYPE>
```

```
            <VALUE>IfcBuildingElementProxy</VALUE>
```

```
          </CONDITION>
```

```
          <ACTION>
```

```
            <TYPE>And...</TYPE>
```

```
          </ACTION>
```

```
        </RULE>
```

```
      </RULE>
```

```
      <IFCTYPE>Any</IFCTYPE>
```

```
      <PROPERTY>
```

```
        <NAME>bauteilKlassifikation</NAME>
```

```
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
```

```
        <TYPE>PropertySet</TYPE>
```

```
        <VALUETYPE>StringValue</VALUETYPE>
```

```
        <UNIT>None</UNIT>
```

```
      </PROPERTY>
```

```
      <CONDITION>
```

```
        <TYPE>Is</TYPE>
```

```
        <VALUE>br.ue</VALUE>
```

```
      </CONDITION>
```

```
      <ACTION>
```

```
        <TYPE>Add</TYPE>
```

```
      </ACTION>
```

```
    </RULE>
```

```
  </SMARTVIEWSET>
```

```
</SMARTVIEWSETS>
```

```
  <IFCTYPE>Any</IFCTYPE>
```

```
  <PROPERTY>
```

```
    <NAME>None</NAME>
```

```
    <PROPERTYSETNAME>None</PROPERTYSETNAME>
```

```
    <TYPE>None</TYPE>
```

```
    <VALUETYPE>None</VALUETYPE>
```

```
    <UNIT>None</UNIT>
```

```

</PROPERTY>
<CONDITION>
  <TYPE>Is</TYPE>
  <VALUE/>
</CONDITION>
<ACTION>
  <TYPE>SetColored</TYPE>
  <R>78</R>
  <G>255</G>
  <B>83</B>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bauteilKlassifikation</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColored</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bauteilName</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColored</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bemerkung</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColored</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>

```

```

<IFCTYPE>Any</IFCTYPE>
<PROPERTY>
  <NAME>identitaet</NAME>
  <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
  <TYPE>PropertySet</TYPE>
  <VALUETYPE>StringValue</VALUETYPE>
  <UNIT>None</UNIT>
</PROPERTY>
<CONDITION>
  <TYPE>StringIsNotDefined</TYPE>
  <VALUE/>
</CONDITION>
<ACTION>
  <TYPE>SetColor</TYPE>
  <R>255</R>
  <G>0</G>
  <B>0</B>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>KGK-Verzeichnisnummer</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>name</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>idBruecke</NAME>
    <PROPERTYSETNAME>UE</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>

```

```

        <TYPE>SetColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ub.20 Stuetze</TITLE>
    <DESCRIPTION>Elternklasse: B
Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:29:59 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-09-02 11:17:33 +0200</MODIFICATIONDATE>
    <GUID>4b93c70d-96b7-4a86-aac4-14c7d91da84b</GUID>
    <RULES>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>IFC Element</NAME>
            <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
            <TYPE>Summary</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE>IfcBuildingElementProxy</VALUE>
        </CONDITION>
        <ACTION>
            <TYPE>And...</TYPE>
        </ACTION>
    </RULE>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>bauteilKlassifikation</NAME>
            <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
            <TYPE>PropertySet</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE>br.ub.20</VALUE>
        </CONDITION>
        <ACTION>
            <TYPE>Add</TYPE>
        </ACTION>
    </RULE>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>None</NAME>
            <PROPERTYSETNAME>None</PROPERTYSETNAME>
            <TYPE>None</TYPE>
            <VALUETYPE>None</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE/>
        </CONDITION>
        <ACTION>
            <TYPE>SetColored</TYPE>
            <R>78</R>
            <G>255</G>
            <B>83</B>
        </ACTION>
    </RULE>
</RULES>

```

```

<IFCTYPE>Any</IFCTYPE>
<PROPERTY>
  <NAME>bauteilKlassifikation</NAME>
  <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
  <TYPE>PropertySet</TYPE>
  <VALUETYPE>StringValue</VALUETYPE>
  <UNIT>None</UNIT>
</PROPERTY>
<CONDITION>
  <TYPE>StringIsNotDefined</TYPE>
  <VALUE/>
</CONDITION>
<ACTION>
  <TYPE>SetColored</TYPE>
  <R>255</R>
  <G>0</G>
  <B>0</B>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bauteilName</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColored</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bemerkung</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColored</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>identitaet</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>

```

```

        <TYPE>SetColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>K GK-Verzeichnisnummer</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>name</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>idUnterbau</NAME>
        <PROPERTYSETNAME>STU</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>stuetzenart</NAME>
        <PROPERTYSETNAME>STU</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>

```

```

    <UNIT>None</UNIT>
  </PROPERTY>
</CONDITION>
  <TYPE>StringIsNotDefined</TYPE>
  <VALUE/>
</CONDITION>
<ACTION>
  <TYPE>SetColor</TYPE>
  <R>255</R>
  <G>0</G>
  <B>0</B>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>betriebszustand</NAME>
    <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>material</NAME>
    <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>materialEigenschaften</NAME>
    <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>

```

```

<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>technischerPlatz</NAME>
    <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>technischerPlatzBezeichnung</NAME>
    <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
</SMARTVIEWS>
</SMARTVIEWSET>
<SMARTVIEWSET>
  <TITLE>Visualisierung der Eigenschaften - Brueckenmodell_1</TITLE>
  <DESCRIPTION>Prüfung für folgende Einstellungen:
- Lph-1-BRM_1 BRM_1
</DESCRIPTION>
  <GUID>99b683c0-566a-4b5d-9788-5ad53df55acf</GUID>
  <MODIFICATIONDATE>2021-07-12 11:36:21 +0200</MODIFICATIONDATE>
  <SMARTVIEWS>
    <SMARTVIEW>
      <TITLE>bauteilKlassifikation (Allgemeine_Eigenschaften.bauteilKlassifikation)</TITLE>
      <DESCRIPTION/>
      <CREATOR>Dominik Nils Sonnek</CREATOR>
      <CREATIONDATE>2021-07-12 11:38:07 +0200</CREATIONDATE>
      <MODIFIER>Dominik Nils Sonnek</MODIFIER>
      <MODIFICATIONDATE>2021-07-12 11:38:07 +0200</MODIFICATIONDATE>
      <GUID>b25d9803-56f0-4ab7-a5c5-62d3301f1f33</GUID>
      <RULES>
        <RULE>
          <IFCTYPE>Any</IFCTYPE>
          <PROPERTY>
            <NAME>None</NAME>
            <PROPERTYSETNAME>None</PROPERTYSETNAME>
            <TYPE>None</TYPE>
            <VALUETYPE>None</VALUETYPE>
            <UNIT>None</UNIT>
          </PROPERTY>
          <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE/>
          </CONDITION>
        </RULE>
      </RULES>
    </SMARTVIEW>
  </SMARTVIEWS>
</SMARTVIEWSET>

```

```

</CONDITION>
<ACTION>
  <TYPE>AddSetColored</TYPE>
  <R>255</R>
  <G>255</G>
  <B>255</B>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bauteilKlassifikation</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetAutoColored</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bauteilKlassifikation</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetTransparent</TYPE>
  </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>bauteilName (Allgemeine_Eigenschaften.bauteilName)</TITLE>
  <DESCRIPTION/>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:38:13 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:38:13 +0200</MODIFICATIONDATE>
  <GUID>cc7d3f02-6e7e-4246-bd29-0292b88a0c18</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>None</NAME>
        <PROPERTYSETNAME>None</PROPERTYSETNAME>
        <TYPE>None</TYPE>
        <VALUETYPE>None</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>AddSetColored</TYPE>
        <R>255</R>
        <G>255</G>

```

```

        <B>255</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>bauteilName</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetAutoColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>bauteilName</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetTransparent</TYPE>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>bemerkung (Allgemeine_Eigenschaften.bemerkung)</TITLE>
    <DESCRIPTION/>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:38:16 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:38:16 +0200</MODIFICATIONDATE>
    <GUID>346cd628-b8e2-4b52-982b-942b7361c5aa</GUID>
    <RULES>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>None</NAME>
                <PROPERTYSETNAME>None</PROPERTYSETNAME>
                <TYPE>None</TYPE>
                <VALUETYPE>None</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE/>
            </CONDITION>
            <ACTION>
                <TYPE>AddSetColored</TYPE>
                <R>255</R>
                <G>255</G>
                <B>255</B>
            </ACTION>
        </RULE>
    </RULES>
    <IFCTYPE>Any</IFCTYPE>

```

```

<PROPERTY>
  <NAME>bemerkung</NAME>
  <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
  <TYPE>PropertySet</TYPE>
  <VALUETYPE>StringValue</VALUETYPE>
  <UNIT>None</UNIT>
</PROPERTY>
<CONDITION>
  <TYPE>StringIsDefined</TYPE>
  <VALUE/>
</CONDITION>
<ACTION>
  <TYPE>SetAutoColored</TYPE>
  <R>255</R>
  <G>0</G>
  <B>0</B>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bemerkung</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetTransparent</TYPE>
  </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>identitaet (Allgemeine_Eigenschaften.identitaet)</TITLE>
  <DESCRIPTION/>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:38:21 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:38:21 +0200</MODIFICATIONDATE>
  <GUID>3cdae709-c749-48d9-8cd3-e02b3fc5bd55</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>None</NAME>
        <PROPERTYSETNAME>None</PROPERTYSETNAME>
        <TYPE>None</TYPE>
        <VALUETYPE>None</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>AddSetColored</TYPE>
        <R>255</R>
        <G>255</G>
        <B>255</B>
      </ACTION>
    </RULE>
  </RULES>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>identitaet</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
    </PROPERTY>
  </RULE>

```

```

    <UNIT>None</UNIT>
  </PROPERTY>
</CONDITION>
  <TYPE>StringIsDefined</TYPE>
  <VALUE/>
</CONDITION>
<ACTION>
  <TYPE>SetAutoColored</TYPE>
  <R>255</R>
  <G>0</G>
  <B>0</B>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>identitaet</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetTransparent</TYPE>
  </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>KGK-Verzeichnisnummer (Allgemeine_Eigenschaften.KGK-Verzeichnisnummer)</TITLE>
  <DESCRIPTION/>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:38:19 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-13 09:39:16 +0200</MODIFICATIONDATE>
  <GUID>8d42ffd5-0d94-4f79-a949-0470c8b4822e</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>None</NAME>
        <PROPERTYSETNAME>None</PROPERTYSETNAME>
        <TYPE>None</TYPE>
        <VALUETYPE>None</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>AddSetColored</TYPE>
        <R>255</R>
        <G>255</G>
        <B>255</B>
      </ACTION>
    </RULE>
  </RULES>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>KGK-Verzeichnisnummer</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsDefined</TYPE>
      <VALUE/>
    </CONDITION>
  </RULE>

```

```

</CONDITION>
<ACTION>
  <TYPE>SetAutoColored</TYPE>
  <R>255</R>
  <G>0</G>
  <B>0</B>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>KGK-Verzeichnisnummer</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetTransparent</TYPE>
  </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>name (Allgemeine_Eigenschaften.name)</TITLE>
  <DESCRIPTION/>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:38:24 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:38:24 +0200</MODIFICATIONDATE>
  <GUID>4d24cfcd-0560-4979-b674-acbc28784e51</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>None</NAME>
        <PROPERTYSETNAME>None</PROPERTYSETNAME>
        <TYPE>None</TYPE>
        <VALUETYPE>None</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>AddSetColored</TYPE>
        <R>255</R>
        <G>255</G>
        <B>255</B>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>name</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>SetAutoColored</TYPE>
        <R>255</R>
        <G>0</G>

```

```

        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>name</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetTransparent</TYPE>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>idBruecke (UE.idBruecke)</TITLE>
    <DESCRIPTION>Ersetzt die Aggregation</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:39:14 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:39:14 +0200</MODIFICATIONDATE>
    <GUID>2c3261d4-9186-4d9a-909d-1bc47b29e93b</GUID>
    <RULES>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>None</NAME>
                <PROPERTYSETNAME>None</PROPERTYSETNAME>
                <TYPE>None</TYPE>
                <VALUETYPE>None</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE/>
            </CONDITION>
            <ACTION>
                <TYPE>AddSetColored</TYPE>
                <R>255</R>
                <G>255</G>
                <B>255</B>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>idBruecke</NAME>
                <PROPERTYSETNAME>UE</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>StringIsDefined</TYPE>
                <VALUE/>
            </CONDITION>
            <ACTION>
                <TYPE>SetAutoColored</TYPE>
                <R>255</R>
                <G>0</G>
                <B>0</B>
            </ACTION>
        </RULE>
    </RULES>
    <IFCTYPE>Any</IFCTYPE>

```

```

    <PROPERTY>
      <NAME>idBruecke</NAME>
      <PROPERTYSETNAME>UE</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsNotDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetTransparent</TYPE>
    </ACTION>
  </RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>idUnterbau (STU.idUnterbau)</TITLE>
  <DESCRIPTION>Ersetzt die Aggregation</DESCRIPTION>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
  <GUID>1e791ca3-877a-4957-8549-abe519d3f144</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>None</NAME>
        <PROPERTYSETNAME>None</PROPERTYSETNAME>
        <TYPE>None</TYPE>
        <VALUETYPE>None</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>AddSetColor</TYPE>
        <R>255</R>
        <G>255</G>
        <B>255</B>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>idUnterbau</NAME>
        <PROPERTYSETNAME>STU</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>SetAutoColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>idUnterbau</NAME>
        <PROPERTYSETNAME>STU</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>

```

```

        <UNIT>None</UNIT>
    </PROPERTY>
</CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
</CONDITION>
<ACTION>
    <TYPE>SetTransparent</TYPE>
</ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>stuetzenart (STU.stuetzenart)</TITLE>
    <DESCRIPTION/>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
    <GUID>5893a24e-d6cf-4267-8f07-5fafdf8bc5a7</GUID>
    <RULES>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>None</NAME>
                <PROPERTYSETNAME>None</PROPERTYSETNAME>
                <TYPE>None</TYPE>
                <VALUETYPE>None</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE/>
            </CONDITION>
            <ACTION>
                <TYPE>AddSetColored</TYPE>
                <R>255</R>
                <G>255</G>
                <B>255</B>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>stuetzenart</NAME>
                <PROPERTYSETNAME>STU</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>StringIsDefined</TYPE>
                <VALUE/>
            </CONDITION>
            <ACTION>
                <TYPE>SetAutoColored</TYPE>
                <R>255</R>
                <G>0</G>
                <B>0</B>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>stuetzenart</NAME>
                <PROPERTYSETNAME>STU</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>StringIsNotDefined</TYPE>
                <VALUE/>

```

```

        </CONDITION>
        <ACTION>
            <TYPE>SetTransparent</TYPE>
        </ACTION>
    </RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>betriebszustand (Bauteil.betriebszustand)</TITLE>
    <DESCRIPTION>Betriebszustandstyp</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
    <GUID>f2402620-1e75-419d-8815-944d0213cfeb</GUID>
    <RULES>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>None</NAME>
                <PROPERTYSETNAME>None</PROPERTYSETNAME>
                <TYPE>None</TYPE>
                <VALUETYPE>None</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE/>
            </CONDITION>
            <ACTION>
                <TYPE>AddSetColor</TYPE>
                <R>255</R>
                <G>255</G>
                <B>255</B>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>betriebszustand</NAME>
                <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>StringIsDefined</TYPE>
                <VALUE/>
            </CONDITION>
            <ACTION>
                <TYPE>SetAutoColored</TYPE>
                <R>255</R>
                <G>0</G>
                <B>0</B>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>betriebszustand</NAME>
                <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>StringIsNotDefined</TYPE>
                <VALUE/>
            </CONDITION>
            <ACTION>
                <TYPE>SetTransparent</TYPE>
            </ACTION>
        </RULE>
    </RULES>

```

```

</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>material (Bauteil.material)</TITLE>
  <DESCRIPTION/>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
  <GUID>fc065a76-1c4b-46a1-bb62-63fd791932c5</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>None</NAME>
        <PROPERTYSETNAME>None</PROPERTYSETNAME>
        <TYPE>None</TYPE>
        <VALUETYPE>None</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>AddSetColored</TYPE>
        <R>255</R>
        <G>255</G>
        <B>255</B>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>material</NAME>
        <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>SetAutoColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>material</NAME>
        <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>SetTransparent</TYPE>
      </ACTION>
    </RULE>
  </RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>materialEigenschaften (Bauteil.materialEigenschaften)</TITLE>
  <DESCRIPTION/>

```

```

<CREATOR>Dominik Nils Sonnek</CREATOR>
<CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
<MODIFIER>Dominik Nils Sonnek</MODIFIER>
<MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
<GUID>4ba63e19-05dd-4d47-bf78-18331d11c3c4</GUID>
<RULES>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>None</NAME>
      <PROPERTYSETNAME>None</PROPERTYSETNAME>
      <TYPE>None</TYPE>
      <VALUETYPE>None</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>Is</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>AddSetColor</TYPE>
      <R>255</R>
      <G>255</G>
      <B>255</B>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>materialEigenschaften</NAME>
      <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetAutoColored</TYPE>
      <R>255</R>
      <G>0</G>
      <B>0</B>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>materialEigenschaften</NAME>
      <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsNotDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetTransparent</TYPE>
    </ACTION>
  </RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>technischerPlatz (Bauteil.technischerPlatz)</TITLE>
  <DESCRIPTION/>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
  <GUID>8a6944cf-01b3-4f73-8b20-6c8b8b104095</GUID>

```

```

<RULES>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>None</NAME>
      <PROPERTYSETNAME>None</PROPERTYSETNAME>
      <TYPE>None</TYPE>
      <VALUETYPE>None</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>Is</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>AddSetColor</TYPE>
      <R>255</R>
      <G>255</G>
      <B>255</B>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>technischerPlatz</NAME>
      <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetAutoColored</TYPE>
      <R>255</R>
      <G>0</G>
      <B>0</B>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>technischerPlatz</NAME>
      <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsNotDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetTransparent</TYPE>
    </ACTION>
  </RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>technischerPlatzBezeichnung (Bauteil.technischerPlatzBezeichnung)</TITLE>
  <DESCRIPTION/>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
  <GUID>40563e86-0172-4dcc-a053-4acd217918ab</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>None</NAME>

```

```

        <PROPERTYSETNAME>None</PROPERTYSETNAME>
        <TYPE>None</TYPE>
        <VALUETYPE>None</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>AddSetColored</TYPE>
        <R>255</R>
        <G>255</G>
        <B>255</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>technischerPlatzBezeichnung</NAME>
        <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetAutoColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>technischerPlatzBezeichnung</NAME>
        <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetTransparent</TYPE>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
</SMARTVIEWS>
</SMARTVIEWSET>
<SMARTVIEWSET>
    <TITLE>Prüfung der Objekteigenschaften - Brueckenmodell_1</TITLE>
    <DESCRIPTION>Prüfung für folgende Einstellungen:
    - Lph-1-BRM_1 BRM_1
</DESCRIPTION>
    <GUID>ba48f84d-4474-4ac7-8d00-f037665a48cb</GUID>
    <MODIFICATIONDATE>2021-07-12 11:36:21 +0200</MODIFICATIONDATE>
    <SMARTVIEWS>
        <SMARTVIEW>
            <TITLE>BR.ue Ueberbau - bauteilKlassifikation</TITLE>
            <DESCRIPTION>bauteilKlassifikation : Allgemeine Eigenschaften : Elternklasse: AE
            Vererbungskette: AE : Ueberbau : Bruecken</DESCRIPTION>
            <CREATOR>Dominik Nils Sonnek</CREATOR>
            <CREATIONDATE>2021-07-12 11:38:07 +0200</CREATIONDATE>
            <MODIFIER>Dominik Nils Sonnek</MODIFIER>
            <MODIFICATIONDATE>2021-07-12 11:38:07 +0200</MODIFICATIONDATE>
        </SMARTVIEW>
    </SMARTVIEWS>
</SMARTVIEWSET>

```

```

<GUID>b826a30a-2104-4597-b0c5-d700f1c97bf1</GUID>
<RULES>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>IFC Element</NAME>
      <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
      <TYPE>Summary</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>Is</TYPE>
      <VALUE>IfcBuildingElementProxy</VALUE>
    </CONDITION>
    <ACTION>
      <TYPE>And...</TYPE>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>bauteilKlassifikation</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>Is</TYPE>
      <VALUE>br.ue</VALUE>
    </CONDITION>
    <ACTION>
      <TYPE>Add</TYPE>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>bauteilKlassifikation</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsNotDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetColor</TYPE>
      <R>255</R>
      <G>0</G>
      <B>0</B>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>bauteilKlassifikation</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetColor</TYPE>
      <R>0</R>
      <G>255</G>
    </ACTION>
  </RULE>

```

```

        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ue Ueberbau - bauteilName</TITLE>
    <DESCRIPTION>bauteilName : Allgemeine_Eigenschaften : Elternklasse: AE
Vererbungskette: AE : Ueberbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:38:13 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:38:13 +0200</MODIFICATIONDATE>
    <GUID>3259a6b7-593c-47c6-9b56-6b6d0054cdfc</GUID>
    <RULES>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>IFC Element</NAME>
                <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
                <TYPE>Summary</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE>IfcBuildingElementProxy</VALUE>
            </CONDITION>
            <ACTION>
                <TYPE>And...</TYPE>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>bauteilKlassifikation</NAME>
                <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE>br.ue</VALUE>
            </CONDITION>
            <ACTION>
                <TYPE>Add</TYPE>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>bauteilName</NAME>
                <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>StringIsNotDefined</TYPE>
                <VALUE/>
            </CONDITION>
            <ACTION>
                <TYPE>SetColor</TYPE>
                <R>255</R>
                <G>0</G>
                <B>0</B>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>bauteilName</NAME>

```

```

        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColor</TYPE>
        <R>0</R>
        <G>255</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ue Ueberbau - bemerkung</TITLE>
    <DESCRIPTION>bemerkung : Allgemeine Eigenschaften : Elternklasse: AE
Vererbungskette: AE : Ueberbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:38:16 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:38:16 +0200</MODIFICATIONDATE>
    <GUID>17bc5ad3-b7db-4fe8-85b7-6efc9b515b6b</GUID>
    <RULES>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>IFC Element</NAME>
                <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
                <TYPE>Summary</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE>IfcBuildingElementProxy</VALUE>
            </CONDITION>
            <ACTION>
                <TYPE>And...</TYPE>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>bauteilKlassifikation</NAME>
                <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE>br.ue</VALUE>
            </CONDITION>
            <ACTION>
                <TYPE>Add</TYPE>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>bemerkung</NAME>
                <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>StringIsNotDefined</TYPE>

```

```

        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>bemerkung</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColored</TYPE>
        <R>0</R>
        <G>255</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ue Ueberbau - identitaet</TITLE>
    <DESCRIPTION>identitaet : Allgemeine_Eigenschaften : Elternklasse: AE
Vererbungskette: AE : Ueberbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:38:21 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:38:21 +0200</MODIFICATIONDATE>
    <GUID>f3504a3a-d1fc-4ba1-8963-b90fe2079465</GUID>
    <RULES>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>IFC Element</NAME>
            <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
            <TYPE>Summary</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE>IfcBuildingElementProxy</VALUE>
        </CONDITION>
        <ACTION>
            <TYPE>And...</TYPE>
        </ACTION>
    </RULE>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>bauteilKlassifikation</NAME>
            <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
            <TYPE>PropertySet</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE>br.ue</VALUE>
        </CONDITION>
        <ACTION>
            <TYPE>Add</TYPE>

```

```

    </ACTION>
  </RULE>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>identitaet</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>identitaet</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>0</R>
    <G>255</G>
    <B>0</B>
  </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>BR.ue Ueberbau - KGK-Verzeichnisnummer</TITLE>
  <DESCRIPTION>KGK-Verzeichnisnummer : Allgemeine_Eigenschaften : Elternklasse: AE
  Vererbungskette: AE : Ueberbau : Bruecken</DESCRIPTION>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:38:19 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-13 09:39:16 +0200</MODIFICATIONDATE>
  <GUID>b03aefde-3587-4786-ae9b-9f85bb887367</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>IFC Element</NAME>
        <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
        <TYPE>Summary</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>IfcBuildingElementProxy</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>And...</TYPE>
      </ACTION>
    </RULE>
  </RULES>
  <IFCTYPE>Any</IFCTYPE>

```

```

    <PROPERTY>
      <NAME>bauteilKlassifikation</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>Is</TYPE>
      <VALUE>br.ue</VALUE>
    </CONDITION>
    <ACTION>
      <TYPE>Add</TYPE>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>K GK-Verzeichnisnummer</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsNotDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetColor</TYPE>
      <R>255</R>
      <G>0</G>
      <B>0</B>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>K GK-Verzeichnisnummer</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetColor</TYPE>
      <R>0</R>
      <G>255</G>
      <B>0</B>
    </ACTION>
  </RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>BR.ue Ueberbau - name</TITLE>
  <DESCRIPTION>name : Allgemeine_Eigenschaften : Elternklasse: AE
  Vererbungskette: AE : Ueberbau : Bruecken</DESCRIPTION>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:38:24 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:38:24 +0200</MODIFICATIONDATE>
  <GUID>663c0f07-f8e8-40d0-aeb1-52677e763e03</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>IFC Element</NAME>
        <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
        <TYPE>Summary</TYPE>

```

```

        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>IfcBuildingElementProxy</VALUE>
    </CONDITION>
    <ACTION>
        <TYPE>And...</TYPE>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>bauteilKlassifikation</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>br.ue</VALUE>
    </CONDITION>
    <ACTION>
        <TYPE>Add</TYPE>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>name</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColor</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>name</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColor</TYPE>
        <R>0</R>
        <G>255</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ue Ueberbau - idBruecke</TITLE>
    <DESCRIPTION>Ersetzt die Aggregation : Ueberbau : Elternklasse: AE

```

```

Vererbungskette: AE : Ueberbau : Bruecken</DESCRIPTION>
<CREATOR>Dominik Nils Sonnek</CREATOR>
<CREATIONDATE>2021-07-12 11:39:14 +0200</CREATIONDATE>
<MODIFIER>Dominik Nils Sonnek</MODIFIER>
<MODIFICATIONDATE>2021-07-12 11:39:14 +0200</MODIFICATIONDATE>
<GUID>2e03ea39-b336-49ca-b989-765e8cdbdca8</GUID>
<RULES>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>IFC Element</NAME>
      <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
      <TYPE>Summary</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>Is</TYPE>
      <VALUE>IfcBuildingElementProxy</VALUE>
    </CONDITION>
    <ACTION>
      <TYPE>And...</TYPE>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>bauteilKlassifikation</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>Is</TYPE>
      <VALUE>br.ue</VALUE>
    </CONDITION>
    <ACTION>
      <TYPE>Add</TYPE>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>idBruecke</NAME>
      <PROPERTYSETNAME>UE</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsNotDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetColor</TYPE>
      <R>255</R>
      <G>0</G>
      <B>0</B>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>idBruecke</NAME>
      <PROPERTYSETNAME>UE</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsDefined</TYPE>
      <VALUE/>
  </RULE>

```

```

</CONDITION>
<ACTION>
  <TYPE>SetColored</TYPE>
  <R>0</R>
  <G>255</G>
  <B>0</B>
</ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>BR.ub.20 Stuetze - bauteilKlassifikation</TITLE>
  <DESCRIPTION>bauteilKlassifikation : Allgemeine_Eigenschaften : Elternklasse: B
  Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
  <GUID>b270ee09-150a-4d97-abae-4b1c61edb6d5</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>IFC Element</NAME>
        <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
        <TYPE>Summary</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>IfcBuildingElementProxy</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>And...</TYPE>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>bauteilKlassifikation</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>br.ub.20</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>Add</TYPE>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>bauteilKlassifikation</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>SetColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
      </ACTION>

```

```

</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bauteilKlassifikation</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColored</TYPE>
    <R>0</R>
    <G>255</G>
    <B>0</B>
  </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>BR.ub.20 Stuetze - bauteilName</TITLE>
  <DESCRIPTION>bauteilName : Allgemeine_Eigenschaften : Elternklasse: B
  Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
  <GUID>170bele4-d1ca-48d7-a791-9feb211cc83f</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>IFC Element</NAME>
        <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
        <TYPE>Summary</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>IfcBuildingElementProxy</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>And...</TYPE>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>bauteilKlassifikation</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>br.ub.20</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>Add</TYPE>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>bauteilName</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>

```

```

        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColor</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>bauteilName</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColor</TYPE>
        <R>0</R>
        <G>255</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ub.20 Stuetze - bemerkung</TITLE>
    <DESCRIPTION>bemerkung : Allgemeine_Eigenschaften : Elternklasse: B
Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
    <GUID>d2ca3a50-836d-4cf8-8280-ddf13eb05a8b</GUID>
    <RULES>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>IFC Element</NAME>
                <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
                <TYPE>Summary</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE>IfcBuildingElementProxy</VALUE>
            </CONDITION>
            <ACTION>
                <TYPE>And...</TYPE>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>bauteilKlassifikation</NAME>
                <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>

```

```

        <TYPE>Is</TYPE>
        <VALUE>br.ub.20</VALUE>
    </CONDITION>
    <ACTION>
        <TYPE>Add</TYPE>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>bemerkung</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>bemerkung</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColored</TYPE>
        <R>0</R>
        <G>255</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ub.20 Stuetze - identitaet</TITLE>
    <DESCRIPTION>identitaet : Allgemeine_Eigenschaften : Elternklasse: B
Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
    <GUID>2c882cd0-e5b2-46e9-b8a1-617e10872e19</GUID>
    <RULES>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>IFC Element</NAME>
                <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
                <TYPE>Summary</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE>IfcBuildingElementProxy</VALUE>
            </CONDITION>
            <ACTION>

```

```

        <TYPE>And...</TYPE>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>bauteilKlassifikation</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>br.ub.20</VALUE>
    </CONDITION>
    <ACTION>
        <TYPE>Add</TYPE>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>identitaet</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColor</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>identitaet</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColor</TYPE>
        <R>0</R>
        <G>255</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ub.20 Stuetze - KGK-Verzeichnisnummer</TITLE>
    <DESCRIPTION>KGK-Verzeichnisnummer : Allgemeine_Eigenschaften : Elternklasse: B
    Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-13 09:39:16 +0200</MODIFICATIONDATE>
    <GUID>8a802dcd-25f7-4250-adc3-8e9a41273a2b</GUID>
</RULES>
<RULE>

```

```

<IFCTYPE>Any</IFCTYPE>
<PROPERTY>
  <NAME>IFC Element</NAME>
  <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
  <TYPE>Summary</TYPE>
  <VALUETYPE>StringValue</VALUETYPE>
  <UNIT>None</UNIT>
</PROPERTY>
<CONDITION>
  <TYPE>Is</TYPE>
  <VALUE>IfcBuildingElementProxy</VALUE>
</CONDITION>
<ACTION>
  <TYPE>And...</TYPE>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bauteilKlassifikation</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>Is</TYPE>
    <VALUE>br.ub.20</VALUE>
  </CONDITION>
  <ACTION>
    <TYPE>Add</TYPE>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>KGK-Verzeichnisnummer</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColored</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>KGK-Verzeichnisnummer</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColored</TYPE>
    <R>0</R>
    <G>255</G>
    <B>0</B>
  </ACTION>
</RULE>

```

```

</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>BR.ub.20 Stuetze - name</TITLE>
  <DESCRIPTION>name : Allgemeine_Eigenschaften : Elternklasse: B
Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
  <GUID>3ed699da-624d-4286-808a-dddf79214244</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>IFC Element</NAME>
        <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
        <TYPE>Summary</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>IfcBuildingElementProxy</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>And...</TYPE>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>bauteilKlassifikation</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>br.ub.20</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>Add</TYPE>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>name</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
      </CONDITION>
      <ACTION>
        <TYPE>SetColor</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>name</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>

```

```

        <UNIT>None</UNIT>
    </PROPERTY>
</CONDITION>
    <TYPE>StringIsDefined</TYPE>
    <VALUE/>
</CONDITION>
<ACTION>
    <TYPE>SetColor</TYPE>
    <R>0</R>
    <G>255</G>
    <B>0</B>
</ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ub.20 Stuetze - idUnterbau</TITLE>
    <DESCRIPTION>Ersetzt die Aggregation : Stuetze : Elternklasse: B
Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
    <GUID>a6bbf3f4-3ef4-4302-a57a-8a1fe2e39b7d</GUID>
    <RULES>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>IFC Element</NAME>
            <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
            <TYPE>Summary</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE>IfcBuildingElementProxy</VALUE>
        </CONDITION>
        <ACTION>
            <TYPE>And...</TYPE>
        </ACTION>
    </RULE>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>bauteilKlassifikation</NAME>
            <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
            <TYPE>PropertySet</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE>br.ub.20</VALUE>
        </CONDITION>
        <ACTION>
            <TYPE>Add</TYPE>
        </ACTION>
    </RULE>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>idUnterbau</NAME>
            <PROPERTYSETNAME>STU</PROPERTYSETNAME>
            <TYPE>PropertySet</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>StringIsNotDefined</TYPE>
            <VALUE/>
        </CONDITION>
        <ACTION>

```

```

        <TYPE>SetColored</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>idUnterbau</NAME>
        <PROPERTYSETNAME>STU</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColored</TYPE>
        <R>0</R>
        <G>255</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ub.20 Stuetze - stuetzenart</TITLE>
    <DESCRIPTION>stuetzenart : Stuetze : Elternklasse: B
Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
    <GUID>2dc227f3-91dc-4cc0-a4d0-bcd5f0ce0b7a</GUID>
    <RULES>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>IFC Element</NAME>
                <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
                <TYPE>Summary</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE>IfcBuildingElementProxy</VALUE>
            </CONDITION>
            <ACTION>
                <TYPE>And...</TYPE>
            </ACTION>
        </RULE>
        <RULE>
            <IFCTYPE>Any</IFCTYPE>
            <PROPERTY>
                <NAME>bauteilKlassifikation</NAME>
                <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
                <TYPE>PropertySet</TYPE>
                <VALUETYPE>StringValue</VALUETYPE>
                <UNIT>None</UNIT>
            </PROPERTY>
            <CONDITION>
                <TYPE>Is</TYPE>
                <VALUE>br.ub.20</VALUE>
            </CONDITION>
            <ACTION>
                <TYPE>Add</TYPE>
            </ACTION>
        </RULE>
    </RULES>

```

```

<IFCTYPE>Any</IFCTYPE>
<PROPERTY>
  <NAME>stuetzenart</NAME>
  <PROPERTYSETNAME>STU</PROPERTYSETNAME>
  <TYPE>PropertySet</TYPE>
  <VALUETYPE>StringValue</VALUETYPE>
  <UNIT>None</UNIT>
</PROPERTY>
<CONDITION>
  <TYPE>StringIsNotDefined</TYPE>
  <VALUE/>
</CONDITION>
<ACTION>
  <TYPE>SetColored</TYPE>
  <R>255</R>
  <G>0</G>
  <B>0</B>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>stuetzenart</NAME>
    <PROPERTYSETNAME>STU</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColored</TYPE>
    <R>0</R>
    <G>255</G>
    <B>0</B>
  </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>BR.ub.20 Stuetze - betriebszustand</TITLE>
  <DESCRIPTION>Betriebszustandstyp : Bauteil : Elternklasse: B
  Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
  <GUID>a7eadb4d-2cbf-433a-b1d7-a023a8fcb0ff</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>IFC Element</NAME>
        <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
        <TYPE>Summary</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>IfcBuildingElementProxy</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>And...</TYPE>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>bauteilKlassifikation</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>

```

```

        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>br.ub.20</VALUE>
    </CONDITION>
    <ACTION>
        <TYPE>Add</TYPE>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>betriebszustand</NAME>
        <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsNotDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColor</TYPE>
        <R>255</R>
        <G>0</G>
        <B>0</B>
    </ACTION>
</RULE>
<RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
        <NAME>betriebszustand</NAME>
        <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
        <TYPE>StringIsDefined</TYPE>
        <VALUE/>
    </CONDITION>
    <ACTION>
        <TYPE>SetColor</TYPE>
        <R>0</R>
        <G>255</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ub.20 Stuetze - material</TITLE>
    <DESCRIPTION>material : Bauteil : Elternklasse: B
Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
    <GUID>e5a8a7c4-541a-40b1-bfab-a42838c92b4d</GUID>
    <RULES>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>IFC Element</NAME>
            <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
            <TYPE>Summary</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>

```

```

<CONDITION>
  <TYPE>Is</TYPE>
  <VALUE>IfcBuildingElementProxy</VALUE>
</CONDITION>
<ACTION>
  <TYPE>And...</TYPE>
</ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>bauteilKlassifikation</NAME>
    <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>Is</TYPE>
    <VALUE>br.ub.20</VALUE>
  </CONDITION>
  <ACTION>
    <TYPE>Add</TYPE>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>material</NAME>
    <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsNotDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>255</R>
    <G>0</G>
    <B>0</B>
  </ACTION>
</RULE>
<RULE>
  <IFCTYPE>Any</IFCTYPE>
  <PROPERTY>
    <NAME>material</NAME>
    <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
    <TYPE>PropertySet</TYPE>
    <VALUETYPE>StringValue</VALUETYPE>
    <UNIT>None</UNIT>
  </PROPERTY>
  <CONDITION>
    <TYPE>StringIsDefined</TYPE>
    <VALUE/>
  </CONDITION>
  <ACTION>
    <TYPE>SetColor</TYPE>
    <R>0</R>
    <G>255</G>
    <B>0</B>
  </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>BR.ub.20 Stuetze - materialEigenschaften</TITLE>
  <DESCRIPTION>materialEigenschaften : Bauteil : Elternklasse: B
  Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>

```

```

<MODIFIER>Dominik Nils Sonnek</MODIFIER>
<MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
<GUID>8c8d89ed-e82d-4e66-a53d-befe885df429</GUID>
<RULES>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>IFC Element</NAME>
      <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
      <TYPE>Summary</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>Is</TYPE>
      <VALUE>IfcBuildingElementProxy</VALUE>
    </CONDITION>
    <ACTION>
      <TYPE>And...</TYPE>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>bauteilKlassifikation</NAME>
      <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>Is</TYPE>
      <VALUE>br.ub.20</VALUE>
    </CONDITION>
    <ACTION>
      <TYPE>Add</TYPE>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>materialEigenschaften</NAME>
      <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsNotDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetColor</TYPE>
      <R>255</R>
      <G>0</G>
      <B>0</B>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>materialEigenschaften</NAME>
      <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetColor</TYPE>

```

```

        <R>0</R>
        <G>255</G>
        <B>0</B>
    </ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
    <TITLE>BR.ub.20 Stuetze - technischerPlatz</TITLE>
    <DESCRIPTION>technischerPlatz : Bauteil : Elternklasse: B
Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
    <CREATOR>Dominik Nils Sonnek</CREATOR>
    <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
    <MODIFIER>Dominik Nils Sonnek</MODIFIER>
    <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
    <GUID>8ca14572-87de-45fa-a104-1cbf74cd25d7</GUID>
    <RULES>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>IFC Element</NAME>
            <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
            <TYPE>Summary</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE>IfcBuildingElementProxy</VALUE>
        </CONDITION>
        <ACTION>
            <TYPE>And...</TYPE>
        </ACTION>
    </RULE>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>bauteilKlassifikation</NAME>
            <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
            <TYPE>PropertySet</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>Is</TYPE>
            <VALUE>br.ub.20</VALUE>
        </CONDITION>
        <ACTION>
            <TYPE>Add</TYPE>
        </ACTION>
    </RULE>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>
        <PROPERTY>
            <NAME>technischerPlatz</NAME>
            <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
            <TYPE>PropertySet</TYPE>
            <VALUETYPE>StringValue</VALUETYPE>
            <UNIT>None</UNIT>
        </PROPERTY>
        <CONDITION>
            <TYPE>StringIsNotDefined</TYPE>
            <VALUE/>
        </CONDITION>
        <ACTION>
            <TYPE>SetColor</TYPE>
            <R>255</R>
            <G>0</G>
            <B>0</B>
        </ACTION>
    </RULE>
    <RULE>
        <IFCTYPE>Any</IFCTYPE>

```

```

<PROPERTY>
  <NAME>technischerPlatz</NAME>
  <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
  <TYPE>PropertySet</TYPE>
  <VALUETYPE>StringValue</VALUETYPE>
  <UNIT>None</UNIT>
</PROPERTY>
<CONDITION>
  <TYPE>StringIsDefined</TYPE>
  <VALUE/>
</CONDITION>
<ACTION>
  <TYPE>SetColor</TYPE>
  <R>0</R>
  <G>255</G>
  <B>0</B>
</ACTION>
</RULE>
</RULES>
</SMARTVIEW>
<SMARTVIEW>
  <TITLE>BR.ub.20 Stuetze - technischerPlatzBezeichnung</TITLE>
  <DESCRIPTION>technischerPlatzBezeichnung : Bauteil : Elternklasse: B
Vererbungskette: B, AE : Unterbau : Bruecken</DESCRIPTION>
  <CREATOR>Dominik Nils Sonnek</CREATOR>
  <CREATIONDATE>2021-07-12 11:42:08 +0200</CREATIONDATE>
  <MODIFIER>Dominik Nils Sonnek</MODIFIER>
  <MODIFICATIONDATE>2021-07-12 11:42:08 +0200</MODIFICATIONDATE>
  <GUID>06292f28-d008-455f-9c00-003f6e8362ff</GUID>
  <RULES>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>IFC Element</NAME>
        <PROPERTYSETNAME>Summary</PROPERTYSETNAME>
        <TYPE>Summary</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>IfcBuildingElementProxy</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>And...</TYPE>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>bauteilKlassifikation</NAME>
        <PROPERTYSETNAME>Allgemeine_Eigenschaften</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>
      <CONDITION>
        <TYPE>Is</TYPE>
        <VALUE>br.ub.20</VALUE>
      </CONDITION>
      <ACTION>
        <TYPE>Add</TYPE>
      </ACTION>
    </RULE>
    <RULE>
      <IFCTYPE>Any</IFCTYPE>
      <PROPERTY>
        <NAME>technischerPlatzBezeichnung</NAME>
        <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
        <TYPE>PropertySet</TYPE>
        <VALUETYPE>StringValue</VALUETYPE>
        <UNIT>None</UNIT>
      </PROPERTY>

```

```

    <CONDITION>
      <TYPE>StringIsNotDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetColor</TYPE>
      <R>255</R>
      <G>0</G>
      <B>0</B>
    </ACTION>
  </RULE>
  <RULE>
    <IFCTYPE>Any</IFCTYPE>
    <PROPERTY>
      <NAME>technischerPlatzBezeichnung</NAME>
      <PROPERTYSETNAME>Bauteil</PROPERTYSETNAME>
      <TYPE>PropertySet</TYPE>
      <VALUETYPE>StringValue</VALUETYPE>
      <UNIT>None</UNIT>
    </PROPERTY>
    <CONDITION>
      <TYPE>StringIsDefined</TYPE>
      <VALUE/>
    </CONDITION>
    <ACTION>
      <TYPE>SetColor</TYPE>
      <R>0</R>
      <G>255</G>
      <B>0</B>
    </ACTION>
  </RULE>
</RULES>
</SMARTVIEW>
</SMARTVIEWS>
</SMARTVIEWSET>
</SMARTVIEWSETS>

```

ANLAGE 8 MVXML MATERIALZUWEISUNGSTEMPLATE

```
<Templates>
  <ConceptTemplate uuid="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54"
    name="Material Single" applicableSchema="IFC4"
    applicableEntity="IfcObjectDefinition">
    <AttributeRule RuleID="HasAssociations"
      AttributeName="HasAssociations">
      <EntityRules>
        <EntityRule EntityName="IfcRelAssociatesMaterial">
          <AttributeRules>
            <AttributeRule RuleID="RelatingMaterial"
              AttributeName="RelatingMaterial">
              <EntityRules>
                <EntityRule EntityName="IfcMaterial">
                  <AttributeRules>
                    <AttributeRule RuleID="MaterialName" AttributeName="Name">
                      <EntityRules>
                        <EntityRule EntityName="IfcLabel"/>
                      </EntityRules>
                    </AttributeRule>
                    <AttributeRule RuleID="MaterialBeschreibung"
                      AttributeName="Description">
                      <EntityRules>
                        <EntityRule EntityName="IfcText"/>
                      </EntityRules>
                    </AttributeRule>
                    <AttributeRule RuleID="MaterialKategorie"
                      AttributeName="Category">
                      <EntityRules>
                        <EntityRule EntityName="IfcLabel"/>
                      </EntityRules>
                    </AttributeRule>
                  </AttributeRules>
                </EntityRule>
              </EntityRules>
            </AttributeRule>
          </AttributeRules>
        </EntityRule>
      </EntityRules>
    </AttributeRule>
  </Rules>
</ConceptTemplate>
<Views>
```

ANLAGE 9 MVDXML MATERIALZUWEISUNGSVIEW

```
. <ModelView uuid="6d7dc3fb-fc35-4040-b564-2b250da2c1de"
  name="Materialversuch" applicableSchema="IFC4">
  <ExchangeRequirements>
    <ExchangeRequirement uuid="b66682f3-d593-49c1-9584-183eb6699824"
      name="Materialversuch" status="sample" applicability="both" />
    <ExchangeRequirement uuid="b66682f3-d593-49c1-9584-183eb6699825"
      name="Stahlbeton" status="sample" applicability="both" />
  </ExchangeRequirements>
  <Roots>
    <ConceptRoot uuid="5875979d-2f09-4290-979b-d7ad341596f9"
      name="Materialzuweisung" applicableRootEntity="IfcObject">
      <Applicability uuid="00000000-0000-0000-0000-000000000000"
        status="sample">
        <Template ref="7d4442be-15b3-4677-83d9-32bc23f05582" />
        <TemplateRules operator="and" />
      </Applicability>
      <Concepts>
        <Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce09bd"
          name="Vorhandene Materialzuweisung">
          <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
          <Requirements>
            <Requirement applicability="import" requirement="mandatory"
              exchangeRequirement="b66682f3-d593-49c1-9584-183eb6699824" />
          </Requirements>
          <TemplateRules>
            <TemplateRule Parameters="HasAssociations[Exists]=TRUE AND
              RelatingMaterial[Exists]=TRUE"/>
          </TemplateRules>
        </Concept>
        <Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0DDB"
          name="Material der Kategorie Stahlbeton">
          <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
          <Requirements>
            <Requirement applicability="import" requirement="mandatory"
              exchangeRequirement="b66682f3-d593-49c1-9584-183eb6699825" />
          </Requirements>
          <TemplateRules>
            <TemplateRule Parameters="HasAssociations[Exists]=TRUE AND
              RelatingMaterial[Exists]=TRUE AND
              MaterialKategorie[Value]='Stahlbeton'"/>
          </TemplateRules>
        </Concept>
      </Concepts>
    </ConceptRoot>
  </Roots>
</ModelView>
</Views>
```

ANLAGE 10 MVDXML AGGREGATION TEMPLATES

```
<Templates>
  <ConceptTemplate uuid="5098cd13-bf4b-473a-a846-a60f69e9b738"
    name="Aggregationen" applicableSchema="IFC4"
    applicableEntity="IfcObjectDefinition">
    <SubTemplates>
      <ConceptTemplate uuid="AggregationIsDecomposedOhneAE"
        name="Bestandsteile" applicableSchema="IFC4"
        applicableEntity="IfcElement">
        <Rules>
          <AttributeRule RuleID="IsDecomposedBy" AttributeName="IsDecomposedBy">
            <EntityRules>
              <EntityRule EntityName="IfcRelAggregates">
                <AttributeRules>
                  <AttributeRule RuleID="RelatedObjects" AttributeName="RelatedObjects">
                    <EntityRules>
                      <EntityRule EntityName="IfcElement">
                        <AttributeRules>
                          <AttributeRule RuleID="ObjectType" AttributeName="ObjectType">
                            <EntityRules>
                              <EntityRule EntityName="IfcLabel"/>
                            </EntityRules>
                          </AttributeRule>
                        </AttributeRules>
                      </EntityRule>
                    </EntityRules>
                  </AttributeRule>
                </AttributeRules>
              </EntityRule>
            </EntityRules>
          </AttributeRule>
        </Rules>
      </ConceptTemplate>
      <ConceptTemplate uuid="AggregationDecomposesOhneAE" name="IstBestandteil"
        applicableSchema="IFC4" applicableEntity="IfcElement">
        <Rules>
          <AttributeRule RuleID="Decomposes" AttributeName="Decomposes">
            <EntityRules>
              <EntityRule EntityName="IfcRelAggregates">
                <AttributeRules>
                  <AttributeRule RuleID="RelatingObject" AttributeName="RelatingObject">
                    <EntityRules>
                      <EntityRule EntityName="IfcElement">
                        <AttributeRules>
                          <AttributeRule RuleID="IstTeilVon" AttributeName="ObjectType">
                            <EntityRules>
                              <EntityRule EntityName="IfcLabel"/>
                            </EntityRules>
                          </AttributeRule>
                        </AttributeRules>
                      </EntityRule>
                    </EntityRules>
                  </AttributeRule>
                </AttributeRules>
              </EntityRule>
            </EntityRules>
          </AttributeRule>
        </Rules>
      </ConceptTemplate>
      <ConceptTemplate uuid="AggregationDecomposesmitAE" name="Ist Bestandteil AE"
        applicableSchema="IFC4" applicableEntity="IfcElement">
        <Rules>
          <AttributeRule RuleID="Decomposes" AttributeName="Decomposes">
            <EntityRules>
              <EntityRule EntityName="IfcRelAggregates">
                <AttributeRules>
                  <AttributeRule RuleID="RelatingObject" AttributeName="RelatingObject">
                    <EntityRules>
                      <EntityRule EntityName="IfcElement">
                    </EntityRules>
                  </AttributeRule>
                </AttributeRules>
              </EntityRule>
            </EntityRules>
          </AttributeRule>
        </Rules>
      </ConceptTemplate>
    </SubTemplates>
  </ConceptTemplate>

```

```

<AttributeRules>
  <AttributeRule RuleID="IsDefinedBy" AttributeName=
    "IsDefinedBy">
    <EntityRules>
      <EntityRule EntityName="IfcRelDefinesByProperties">
        <AttributeRules>
          <AttributeRule AttributeName=
            "RelatingPropertyDefinition">
            <EntityRules>
              <EntityRule EntityName= "IfcPropertySet">
                <AttributeRules>
                  <AttributeRule RuleID= "PSetName"
                    AttributeName="Name">
                    <EntityRules>
                      <EntityRule EntityName="IfcLabel" />
                    </EntityRules>
                  </AttributeRule>
                </AttributeRules>
              </EntityRule>
            </EntityRules>
          </AttributeRule>
          <AttributeRule AttributeName= "HasProperties">
            <EntityRules>
              <EntityRule EntityName=
                "IfcPropertySingleValue">
                <AttributeRules>
                  <AttributeRule RuleID="PropertyName"
                    AttributeName="Name">
                    <EntityRules>
                      <EntityRule EntityName=
                        "IfcIdentifier" />
                    </EntityRules>
                  </AttributeRule>
                  <AttributeRule RuleID="AttributWert"
                    AttributeName="NominalValue">
                    <EntityRules>
                      <EntityRule EntityName=
                        "IfcText" />
                    </EntityRules>
                  </AttributeRule>
                </AttributeRules>
              </EntityRule>
            </EntityRules>
          </AttributeRule>
        </AttributeRules>
      </EntityRule>
    </EntityRules>
  </AttributeRule>
</AttributeRules>
</ConceptTemplate>
<ConceptTemplate uuid="AggregationIsDecomposedMitAE" name="Bestandteile mit AE"
  applicableSchema="IFC4" applicableEntity="IfcElement">
  <Rules>
    <AttributeRule RuleID="IsDecomposedBy" AttributeName="IsDecomposedBy">
      <EntityRules>
        <EntityRule EntityName="IfcRelAggregates">
          <AttributeRules>
            <AttributeRule RuleID="RelatedObjects" AttributeName="RelatedObjects">
              <EntityRules>
                <EntityRule EntityName="IfcElement">
                  <AttributeRules>
                    <AttributeRule RuleID="IsDefinedBy" AttributeName=
                      "IsDefinedBy">
                      <EntityRules>
                        <EntityRule EntityName="IfcRelDefinesByProperties">

```

```

<AttributeRules>
  <AttributeRule AttributeName=
    "RelatingPropertyDefinition">
    <EntityRules>
      <EntityRule EntityName="IfcPropertySet">
        <AttributeRules>
          <AttributeRule RuleID="PSetName"
            AttributeName="Name">
            <EntityRules>
              <EntityRule EntityName="IfcLabel" />
            </EntityRules>
          </AttributeRule>
          <AttributeRule AttributeName="HasProperties">
            <EntityRules>
              <EntityRule EntityName=
                "IfcPropertySingleValue">
                <AttributeRules>
                  <AttributeRule RuleID="PropertyName"
                    AttributeName="Name">
                    <EntityRules>
                      <EntityRule EntityName=
                        "IfcIdentifier" />
                    </EntityRules>
                  </AttributeRule>
                  <AttributeRule RuleID="AttributWert"
                    AttributeName="NominalValue">
                    <EntityRules>
                      <EntityRule EntityName=
                        "IfcText" />
                    </EntityRules>
                  </AttributeRule>
                </AttributeRules>
              </EntityRule>
            </EntityRules>
          </AttributeRule>
        </AttributeRules>
      </EntityRule>
    </EntityRules>
  </AttributeRule>
</AttributeRules>
</ConceptTemplate>
</SubTemplates>
</ConceptTemplate>
<ConceptTemplate uuid="00000000-0000-0000-0000-000000000000" name="Realtionsprüfung"
  applicableSchema="IFC4" applicableEntity="IfcObject">
  <Rules>
    <AttributeRule AttributeName="IsDefinedBy">
      <EntityRules>
        <EntityRule EntityName="IfcRelDefinesByProperties">
          <AttributeRules>
            <AttributeRule AttributeName="RelatingPropertyDefinition">
              <EntityRules>
                <EntityRule EntityName="IfcPropertySet">
                  <AttributeRules>
                    <AttributeRule RuleID="PsetName" AttributeName="Name">
                      <EntityRules>
                        <EntityRule EntityName="IfcLabel" />
                      </EntityRules>
                    </AttributeRule>
                    <AttributeRule AttributeName="HasProperties">
                      <EntityRules>

```

```

        <EntityRule EntityName="IfcPropertySingleValue">
            <AttributeRules>
                <AttributeRule RuleID="AttributName" AttributeName="Name">
                    <EntityRules>
                        <EntityRule EntityName="IfcIdentifier" />
                    </EntityRules>
                </AttributeRule>
            </AttributeRules>
        </EntityRule>
    </EntityRules>
</AttributeRule>
</AttributeRules>
</EntityRule>
</EntityRules>
</AttributeRule>
</AttributeRules>
</EntityRule>
</EntityRules>
</AttributeRule>
</AttributeRules>
</EntityRule>
</EntityRules>
</AttributeRule>
</Rules>
</ConceptTemplate>
<ConceptTemplate uuid="69cb72cb-fd55-4e49-bef4-14ca32cd568b"
name="Klassifikationsattribut" applicableSchema="IFC4" applicableEntity="IfcObject">
    <Rules>
        <AttributeRule RuleID="IsDefinedBy" AttributeName="IsDefinedBy">
            <EntityRules>
                <EntityRule EntityName="IfcRelDefinesByProperties">
                    <AttributeRules>
                        <AttributeRule RuleID="RelatingPropertyDefinition" AttributeName=
"RelatingPropertyDefinition">
                            <EntityRules>
                                <EntityRule EntityName="IfcPropertySet">
                                    <AttributeRules>
                                        <AttributeRule RuleID="PsetName" AttributeName="Name">
                                            <EntityRules>
                                                <EntityRule EntityName="IfcLabel" />
                                            </EntityRules>
                                        </AttributeRule>
                                        <AttributeRule RuleID="HasProperties" AttributeName="HasProperties">
                                            <EntityRules>
                                                <EntityRule EntityName="IfcPropertySingleValue">
                                                    <AttributeRules>
                                                        <AttributeRule RuleID="AttributName" AttributeName="Name">
                                                            <EntityRules>
                                                                <EntityRule EntityName="IfcIdentifier" />
                                                            </EntityRules>
                                                        </AttributeRule>
                                                        <AttributeRule RuleID="Wert" AttributeName="NominalValue">
                                                            <EntityRules>
                                                                <EntityRule EntityName="IfcSingleValue" />
                                                            </EntityRules>
                                                        </AttributeRule>
                                                    </AttributeRules>
                                                </EntityRule>
                                            </EntityRules>
                                        </AttributeRule>
                                    </AttributeRules>
                                </EntityRule>
                            </EntityRules>
                        </AttributeRule>
                    </AttributeRules>
                </EntityRule>
            </EntityRules>
        </AttributeRule>
    </Rules>
</ConceptTemplate>
</Templates>

```

ANLAGE 11 MVXML AGGREGATION VIEW

```
<ModelView uuid="00000036-0936-0000-0000-000000000000" name="Aggregationprüfung" applica-
bleSchema="IFC4">
  <ExchangeRequirements>
    <ExchangeRequirement uuid="EinfacheAggregation" name="EinfacheAggregation"
applicability="export"></ExchangeRequirement>
    <ExchangeRequirement uuid="SOMkonformeAggregation" name="SOMkonformeAggregation"
applicability="export"></ExchangeRequirement>
  </ExchangeRequirements>
  <Roots>
    <ConceptRoot uuid="00000036-0936-0936-0000-000002907057" name="br.ub.10 Widerlager"
applicableRootEntity="IfcBuildingElementProxy">
      <Applicability>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
        <TemplateRules operator="and">
          <TemplateRule Parameters="PsetName[Value]='Allgemeine_Eigenschaften'
AND AttributName[Value]='bauteilKlassifikation' AND Wert[Value]='br.ub.10'"/>
        </TemplateRules>
      </Applicability>
    </ConceptRoot>
    <Concepts>
      <Concept uuid="00000036-0936-0936-0000-000002907145" name="Bestandteile ohne AE">
        <Template ref="AggregationIsDecomposedOhneAE"/>
        <Requirements>
          <Requirement applicability="export" exchangeRequirement="EinfacheAggregation"
requirement="mandatory"/>
        </Requirements>
        <TemplateRules operator="and">
          <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
RelatedObjects[Exists]=TRUE AND ObjectType[Value]='br.ub.11'"/>
          <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
RelatedObjects[Exists]=TRUE AND ObjectType[Value]='br.ub.12'"/>
          <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
RelatedObjects[Exists]=TRUE AND ObjectType[Value]='br.ub.13'"/>
        </TemplateRules>
      </Concept>
      <Concept uuid="00000036-0936-0936-0000-000002907MMM" name="Bestandteile mit AE">
        <Template ref="AggregationIsDecomposedMitAE"/>
        <Requirements>
          <Requirement applicability="export" exchangeRequirement=
"SOMkonformeAggregation" requirement="mandatory"/>
        </Requirements>
        <TemplateRules operator="and">
          <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
RelatedObjects[Exists]=TRUE AND IsDefinedBy[Exists]=TRUE AND
PSetName[Value]='Allgemeine_Eigenschaften' AND PropertyName[Value]=
'bauteilKlassifikation' AND AttributWert[Value]='br.ub.11'"/>
          <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
RelatedObjects[Exists]=TRUE AND IsDefinedBy[Exists]=TRUE AND
PSetName[Value]='Allgemeine_Eigenschaften' AND PropertyName[Value]=
'bauteilKlassifikation' AND AttributWert[Value]='br.ub.12'"/>
          <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
RelatedObjects[Exists]=TRUE AND IsDefinedBy[Exists]=TRUE AND
PSetName[Value]='Allgemeine_Eigenschaften' AND PropertyName[Value]=
'bauteilKlassifikation' AND AttributWert[Value]='br.ub.13'"/>
        </TemplateRules>
      </Concept>
    </Concepts>
  </ConceptRoot>
  <ConceptRoot uuid="00000036-0936-0936-0000-000002907194" name="br.ub.12 Fluegelwand"
applicableRootEntity="IfcBuildingElementProxy">
    <Applicability>
      <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
      <TemplateRules operator="and">
        <TemplateRule Parameters="PsetName[Value]='Allgemeine_Eigenschaften' AND
AttributName[Value]='bauteilKlassifikation' AND Wert[Value]='br.ub.12'"/>
      </TemplateRules>
    </Applicability>
  </ConceptRoot>
  <Concepts>
    <Concept uuid="00000036-0936-0936-0000-000002907RRR" name="Ist Teil eines
```

```

    Widerlagers ohne AE">
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
      <Requirement applicability="export" exchangeRequirement="EinfacheAggregation"
        requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
      <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
        RelatingObject[Exists]=TRUE AND IstTeilVon[Value]='br.ub.10'"/>
    </TemplateRules>
  </Concept>
  <Concept uuid="00000036-0936-0936-0000-000002907GHH" name="Ist Teil eines
Widerlagers mit AE">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
      <Requirement applicability="export" exchangeRequirement=
        "SOMkonformeAggregation" requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
      <TemplateRule Parameters="Decomposes[Exists]=TRUE AND #
        RelatingObject[Exists]=TRUE AND IsDefinedBy[Exists]=TRUE AND
        PSetName[Value]='Allgemeine_Eigenschaften' AND PropertyName[Value]=
        'bauteilKlassifikation' AND AttributWert[Value]='br.ub.10'"/>
    </TemplateRules>
  </Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907367" name="br.ub.13
Widerlagerfundament" applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
      <TemplateRule Parameters="PSetName[Value]='Allgemeine_Eigenschaften' AND
        AttributName[Value]='bauteilKlassifikation' AND Wert[Value]='br.ub.13'"/>
    </TemplateRules>
  </Applicability>
  <Concepts>
    <Concept uuid="00000036-0936-0936-0000-000002907SSS" name="Ist Teil eines
Widerlagers ohne AE" code="">
      <Template ref="AggregationDecomposesOhneAE"/>
      <Requirements>
        <Requirement applicability="export" exchangeRequirement="EinfacheAggregation"
          requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND IstTeilVon[Value]='br.ub.10'"/>
      </TemplateRules>
    </Concept>
    <Concept uuid="00000036-0936-0936-0000-000002907JJI" name="Ist Teil eines
Widerlagers mit AE">
      <Template ref="AggregationDecomposesmitAE"/>
      <Requirements>
        <Requirement applicability="export" exchangeRequirement=
          "SOMkonformeAggregation" requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND IsDefinedBy[Exists]=TRUE AND
          PSetName[Value]='Allgemeine_Eigenschaften' AND PropertyName[Value]=
          'bauteilKlassifikation' AND AttributWert[Value]='br.ub.10'"/>
      </TemplateRules>
    </Concept>
  </Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907BBB" name="br.ub.11
Widerlagerwand" applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
      <TemplateRule Parameters="PSetName[Value]='Allgemeine_Eigenschaften' AND
        AttributName[Value]='bauteilKlassifikation' AND Wert[Value]='br.ub.11'"/>
    </TemplateRules>

```

```

</Applicability>
<Concepts>
  <Concept uuid="00000036-0936-0936-0000-000002907STT" name="Ist Teil eines
  Widerlagers ohne AE" code="">
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
      <Requirement applicability="export" exchangeRequirement="EinfacheAggregation"
      requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
      <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
      RelatingObject[Exists]=TRUE AND IstTeilVon[Value]='br.ub.10'"/>
    </TemplateRules>
  </Concept>
  <Concept uuid="00000036-0936-0936-0000-000002907GGG" name="Ist Teil eines
  Widerlagers mit AE" code="">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
      <Requirement applicability="export" exchangeRequirement=#
      "SOMkonformeAggregation" requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
      <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
      RelatingObject[Exists]=TRUE AND IsDefinedBy[Exists]=TRUE AND
      PSetName[Value]='Allgemeine_Eigenschaften' AND PropertyName[Value]=
      'bauteilKlassifikation' AND AttributWert[Value]='br.ub.10'"/>
    </TemplateRules>
  </Concept>
</Concepts>
</ConceptRoot>
</Roots>
</ModelView></Views>

```

ANLAGE 12 BMC MVDXML

```
<mvdXML uuid="BMC" name="BMC" >
  <Templates>
    <ConceptTemplate uuid="4b385464-b99a-42bc-a17a-a0a647a1ab5b"
      name="Object Association Material" applicableSchema="IFC4"
      applicableEntity="IfcObjectDefinition">
      <SubTemplates>
        <ConceptTemplate uuid="ea2a63be-fd0d-426d-863b-52c3970ca585"
          name="Material Association" applicableSchema="IFC4"
          applicableEntity="IfcObjectDefinition">
          <Rules>
            <AttributeRule AttributeName="HasAssociations">
              <EntityRules>
                <EntityRule EntityName="IfcRelAssociatesMaterial">
                  <AttributeRules>
                    <AttributeRule AttributeName="RelatingMaterial"/>
                  </AttributeRules>
                </EntityRule>
              </EntityRules>
            </AttributeRule>
          </Rules>
          <SubTemplates>
            <ConceptTemplate uuid="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54"
              name="Material Single" applicableSchema="IFC4"
              applicableEntity="IfcObjectDefinition">
              <Definitions>
                <Definition>
                  <Body></Body>
                </Definition>
                <Definition>
                  <Body lang="tt"></Body>
                  <Link lang="tt" category="definition" title="AUTOMATIC"
                    href=""></Link>
                </Definition>
              </Definitions>
              <Rules>
                <AttributeRule RuleID="Assoziation" AttributeName="HasAssociations">
                  <EntityRules>
                    <EntityRule EntityName="IfcRelAssociatesMaterial">
                      <AttributeRules>
                        <AttributeRule RuleID="VerknuepftesMaterial"
                          AttributeName="RelatingMaterial">
                          <EntityRules>
                            <EntityRule EntityName="IfcMaterial">
                              <AttributeRules>
                                <AttributeRule RuleID="MaterialName" AttributeName="Name">
                                  <EntityRules>
                                    <EntityRule EntityName="IfcLabel"/>
                                  </EntityRules>
                                </AttributeRule>
                                <AttributeRule RuleID="MaterialBeschreibung"
                                  AttributeName="Description">
                                  <EntityRules>
                                    <EntityRule EntityName="IfcText"/>
                                  </EntityRules>
                                </AttributeRule>
                                <AttributeRule RuleID="MaterialKategorie"
                                  AttributeName="Category">
                                  <EntityRules>
                                    <EntityRule EntityName="IfcLabel"/>
                                  </EntityRules>
                                </AttributeRule>
                              </AttributeRules>
                            </EntityRule>
                          </EntityRules>
                        </AttributeRule>
                      </AttributeRules>
                    </EntityRule>
                  </EntityRules>
                </AttributeRule>
              </Rules>
            </ConceptTemplate>
          </SubTemplates>
        </ConceptTemplate>
      </SubTemplates>
    </ConceptTemplate>
  </Templates>
</mvdXML>
```



```

        <EntityRules>
            <EntityRule EntityName="IfcLabel" />
        </EntityRules>
    </AttributeRule>
</AttributeRules>
</EntityRule>
</EntityRules>
</AttributeRule>
</AttributeRules>
</EntityRule>
</EntityRules>
</AttributeRule>
</Rules>
</ConceptTemplate>
<ConceptTemplate uuid="AggregationDecomposesmitAE"
    name="IstBestandteilMitKlassifikation"
    applicableSchema="IFC4" applicableEntity="IfcElement">
<Rules>
    <AttributeRule RuleID="Decomposes" AttributeName="Decomposes">
        <EntityRules>
            <EntityRule EntityName="IfcRelAggregates">
                <AttributeRules>
                    <AttributeRule RuleID="RelatingObject"
                        AttributeName="RelatingObject">
                        <EntityRules>
                            <EntityRule EntityName="IfcElement">
                                <AttributeRules>
                                    <AttributeRule RuleID="PsetZuweisung"
                                        AttributeName="IsDefinedBy">
                                        <EntityRules>
                                            <EntityRule EntityName="IfcRelDefinesByProperties">
                                                <AttributeRules>
                                                    <AttributeRule AttributeName=
                                                        "RelatingPropertyDefinition">
                                                        <EntityRules>
                                                            <EntityRule EntityName="IfcPropertySet">
                                                                <AttributeRules>
                                                                    <AttributeRule
                                                                        RuleID= "PsetName"
                                                                        AttributeName="Name">
                                                                        <EntityRules>
                                                                            <EntityRule EntityName="IfcLabel" />
                                                                        </EntityRules>
                                                                    </AttributeRule>
                                                                <AttributeRule
                                                                    AttributeName="HasProperties">
                                                                    <EntityRules>
                                                                        <EntityRule
                                                                            EntityName="IfcPropertySingleValue">
                                                                            <AttributeRules>
                                                                                <AttributeRule
                                                                                    RuleID="PropertyName"
                                                                                    AttributeName="Name">
                                                                                    <EntityRules>
                                                                                        <EntityRule
                                                                                            EntityName="IfcIdentifier" />
                                                                                    </EntityRules>
                                                                                </AttributeRule>
                                                                            <AttributeRule
                                                                                RuleID="PropertyWert"
                                                                                AttributeName="NominalValue">
                                                                                <EntityRules>
                                                                                    <EntityRule
                                                                                        EntityName="IfcText" />
                                                                                </EntityRules>
                                                                            </AttributeRule>
                                                                        </AttributeRules>
                                                                    </EntityRule>
                                                                </EntityRules>
                                                            </AttributeRule>
                                                        </EntityRules>
                                                    </AttributeRule>
                                                </EntityRules>
                                            </AttributeRule>
                                        </EntityRules>
                                    </AttributeRule>
                                </EntityRules>
                            </EntityRule>
                        </AttributeRules>
                    </EntityRule>
                </EntityRules>
            </AttributeRule>
        </EntityRules>
    </AttributeRule>
</Rules>

```



```

"RelatingPropertyDefinition">
  <EntityRules>
    <EntityRule EntityName="IfcPropertySet">
      <AttributeRules>
        <AttributeRule RuleID="PsetName"
          AttributeName="Name">
          <EntityRules>
            <EntityRule EntityName="IfcLabel" />
          </EntityRules>
        </AttributeRule>
        <AttributeRule
          AttributeName="HasProperties">
          <EntityRules>
            <EntityRule
              EntityName="IfcPropertySingleValue">
              <AttributeRules>
                <AttributeRule
                  RuleID="PropertyName"
                  AttributeName="Name">
                  <EntityRules>
                    <EntityRule
                      EntityName="IfcIdentifier" />
                  </EntityRules>
                </AttributeRule>
                <AttributeRule
                  RuleID="PropertyWert"
                  AttributeName="NominalValue">
                  <EntityRules>
                    <EntityRule
                      EntityName=
                        "IfcPropertySingleValue" />
                  </EntityRules>
                </AttributeRule>
              </AttributeRules>
            </EntityRule>
          </EntityRules>
        </AttributeRule>
        <AttributeRule
          AttributeName="HasProperties">
          <EntityRules>
            <EntityRule
              EntityName="IfcPropertySingleValue">
              <AttributeRules>
                <AttributeRule
                  RuleID="PropertyzweiName"
                  AttributeName="Name">
                  <EntityRules>
                    <EntityRule
                      EntityName="IfcIdentifier" />
                  </EntityRules>
                </AttributeRule>
                <AttributeRule
                  RuleID="PropertyzweiWert"
                  AttributeName="NominalValue">
                  <EntityRules>
                    <EntityRule
                      EntityName=
                        "IfcPropertySingleValue" />
                  </EntityRules>
                </AttributeRule>
              </AttributeRules>
            </EntityRule>
          </EntityRules>
        </AttributeRule>
      </AttributeRules>
    </EntityRule>
  </EntityRules>
</AttributeRules>
</EntityRule>
</EntityRules>
</AttributeRule>
</AttributeRules>

```

```

        </EntityRule>
      </EntityRules>
    </AttributeRule>
  </AttributeRules>
</EntityRule>
</EntityRules>
</AttributeRule>
</Rules>
</ConceptTemplate>
</SubTemplates>
</ConceptTemplate>
<ConceptTemplate uuid="00000000-0000-0000-0000-000000000000"
  name="Realtionsprüfung" applicableSchema="IFC4"
  applicableEntity="IfcObject">
  <Rules>
    <AttributeRule AttributeName="IsDefinedBy">
      <EntityRules>
        <EntityRule EntityName="IfcRelDefinesByProperties">
          <AttributeRules>
            <AttributeRule AttributeName="RelatingPropertyDefinition">
              <EntityRules>
                <EntityRule EntityName="IfcPropertySet">
                  <AttributeRules>
                    <AttributeRule RuleID="PsetName"
                      AttributeName="Name">
                      <EntityRules>
                        <EntityRule EntityName="IfcLabel" />
                      </EntityRules>
                    </AttributeRule>
                  </AttributeRules>
                </EntityRule>
              </EntityRules>
            </AttributeRule>
          </AttributeRules>
        </EntityRule>
      </EntityRules>
    </AttributeRule>
  </Rules>
</ConceptTemplate>
<ConceptTemplate uuid="69cb72cb-fd55-4e49-bef4-14ca32cd568b"
  name="Klassifikationsattribut" applicableSchema="IFC4" applicableEntity="IfcObject">
  <Rules>
    <AttributeRule RuleID="IsDefinedBy" AttributeName="IsDefinedBy">
      <EntityRules>
        <EntityRule EntityName="IfcRelDefinesByProperties">
          <AttributeRules>
            <AttributeRule RuleID="RelatingPropertyDefinition"
              AttributeName="RelatingPropertyDefinition">
              <EntityRules>
                <EntityRule EntityName="IfcPropertySet">
                  <AttributeRules>
                    <AttributeRule RuleID="PsetName"
                      AttributeName="Name">
                      <EntityRules>
                        <EntityRule EntityName="IfcLabel" />
                      </EntityRules>
                    </AttributeRule>
                  </AttributeRules>
                </EntityRule>
              </EntityRules>
            </AttributeRule>
          </AttributeRules>
        </EntityRule>
      </EntityRules>
    </AttributeRule>
  </Rules>
</ConceptTemplate>

```

```

<AttributeRule RuleID="HasProperties"
  AttributeName="HasProperties">
  <EntityRules>
    <EntityRule EntityName=
      "IfcPropertySingleValue">
      <AttributeRules>
        <AttributeRule RuleID="AttributName"
          AttributeName="Name">
          <EntityRules>
            <EntityRule EntityName=
              "IfcIdentifier" />
          </EntityRules>
        </AttributeRule>
        <AttributeRule RuleID="Wert" AttributeName=
          "NominalValue">
          <EntityRules>
            <EntityRule EntityName=
              "IfcSingleValue" />
          </EntityRules>
        </AttributeRule>
      </AttributeRules>
    </EntityRule>
  </EntityRules>
</AttributeRule>
</EntityRules>
</ConceptTemplate>
</Templates>

<Views><ModelView uuid="00000036-0936-0000-0000-000000000000" name="BMC"
  applicableSchema="IFC4">
  <ExchangeRequirements>
    <ExchangeRequirement uuid="ModelContentChecking" name=
      "ModelContentChecking" applicability="export">
    </ExchangeRequirement>
    <ExchangeRequirement uuid="ModelValidation" name="ModelValidation"
      applicability="export">
    </ExchangeRequirement>
  </ExchangeRequirements>
  <Roots>
    <ConceptRoot uuid="befebec6-7f7a-4e17-802e-d02e7b4a198c"
      name="bauteilKlassifikation" status="sample"
      applicableRootEntity="IfcBuildingElement">
    <Applicability uuid="00000000-0000-0000-0000-000000000000"
      status="sample">
      <Template ref="00000000-0000-0000-0000-000000000000" />
      <TemplateRules operator="and" />
    </Applicability>
    <Concepts>
      <Concept uuid="8a1a5102-b88e-4ff9-ba11-aa75bb54d070"
        name="bauteilKlassifikation" status="sample"
        override="false">
      <Template ref="00000000-0000-0000-0000-000000000000" />
      <Requirements>
        <Requirement applicability="import" requirement="mandatory"
          exchangeRequirement="ModelContentChecking" />
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="PsetName[Value]=
          'Allgemeine Eigenschaften' AND AttributName[Value]=
          'bauteilKlassifikation'" />
      </TemplateRules>
    </Concept>
  </Concepts>
</ConceptRoot>

```

```

<ConceptRoot uuid="befebec6-7f7a-4e17-802e-d02e7b4a198g"
  name="name" status="sample" applicableRootEntity=
  "IfcBuildingElement">
  <Applicability uuid="00000000-0000-0000-0000-000000000000"
    status="sample">
    <Template ref="00000000-0000-0000-0000-000000000000" />
    <TemplateRules operator="and" />
  </Applicability>
</ConceptRoot>
<ConceptRoot uuid="8a1a5102-b88e-4ff9-ba11-aa75bb54d0UU"
  name="name" status="sample" override="false">
  <Template ref="00000000-0000-0000-0000-000000000000" />
  <Requirements>
  <Requirement applicability="import" requirement="mandatory"
    exchangeRequirement="ModelContentChecking" />
  </Requirements>
  <TemplateRules operator="and">
  <TemplateRule Parameters="PsetName[Value]=
    'Allgemeine_Eigenschaften' AND AttributName[Value]=
    'name' " />
  </TemplateRules>
</ConceptRoot>
<ConceptRoot uuid="befebec6-7f7a-4e17-802e-d02e7b4a198t"
  name="Materialzuweisung" status="sample"
  applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
  <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
  <TemplateRules operator="or">
  <TemplateRule Parameters="PsetName[Value]=
    'Allgemeine_Eigenschaften' AND AttributName[Value]=
    'bauteilKlassifikation' AND Wert[Value]='br.ue.11'"/>
  <TemplateRule Parameters="PsetName[Value]=
    'Allgemeine_Eigenschaften' AND AttributName[Value]=
    'bauteilKlassifikation' AND Wert[Value]='br.ue.12'"/>
  <TemplateRule Parameters="PsetName[Value]=
    'Allgemeine_Eigenschaften' AND AttributName[Value]=#
    'bauteilKlassifikation' AND Wert[Value]='br.ue.20'"/>
  <TemplateRule Parameters="PsetName[Value]=
    'Allgemeine_Eigenschaften' AND AttributName[Value]=
    'bauteilKlassifikation' AND Wert[Value]='br.ub.11'"/>
  <TemplateRule Parameters="PsetName[Value]=
    'Allgemeine_Eigenschaften' AND AttributName[Value]=
    'bauteilKlassifikation' AND Wert[Value]='br.ub.12'"/>
  <TemplateRule Parameters="PsetName[Value]=
    'Allgemeine_Eigenschaften' AND AttributName[Value]=
    'bauteilKlassifikation' AND Wert[Value]='br.ub.13'"/>
  <TemplateRule Parameters="PsetName[Value]=
    'Allgemeine_Eigenschaften' AND AttributName[Value]=
    'bauteilKlassifikation' AND Wert[Value]='ob.bet.100'"/>
  </TemplateRules>
</Applicability>
</ConceptRoot>
<ConceptRoot uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce09b1"
  name="Materialzuweisung">
  <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
  <Requirements>
  <Requirement applicability="import" requirement="mandatory"
    exchangeRequirement="ModelContentChecking" />
  </Requirements>
  <TemplateRules>
  <TemplateRule Parameters="MaterialName[Exists]=TRUE"/>
  </TemplateRules>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907557"
  name="BR Bruecke" applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
  <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
  <TemplateRules operator="and">
  <TemplateRule Parameters="PsetName[Value]=

```

```

        'Allgemeine Eigenschaften' AND AttributName[Value]=
        'bauteilKlassifikation' AND Wert[Value]='br'"/>
    <TemplateRule Parameters="PsetName[Value]='BR' AND
        AttributName[Value]='Spannweite' AND Wert[Value]&lt;30"/>
</TemplateRules>
</Applicability>
<Concepts>
    <Concept uuid="8a1a5102-b88e-4ff9-ba11-aa75bb54d070"
        name="spannweite" status="sample" override="false">
        <Template ref="00000000-0000-0000-0000-000000000000" />
        <Requirements>
            <Requirement applicability="import" requirement="mandatory"
                exchangeRequirement="ModelContentChecking" />
        </Requirements>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PsetName[Value]='BR' AND
                AttributName[Value]='Spannweite'" />
        </TemplateRules>
    </Concept>
    <Concept uuid="8a1a5102-b88e-4ff9-ba11-aa75bb54d070"
        name="lichteHoehe" status="sample" override="false">
        <Template ref="00000000-0000-0000-0000-000000000000" />
        <Requirements>
            <Requirement applicability="import" requirement="mandatory"
                exchangeRequirement="ModelContentChecking" />
        </Requirements>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PsetName[Value]='BR' AND
                AttributName[Value]='Lichte_Hoehe'" />
        </TemplateRules>
    </Concept>
    <Concept uuid="00000036-0936-0936-0000-000002907142"
        name="Grenzwerteinhaltung" code="">
        <Definitions>
            <Definition>
                <Body lang="en"></Body>
            </Definition>
        </Definitions>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
        <Requirements>
            <Requirement applicability="export"
                exchangeRequirement="ModelValidation"
                requirement="mandatory"/>
        </Requirements>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PsetName[Value]='BR' AND
                AttributName[Value]='Lichte_Hoehe' AND
                Wert[Value]>6"/>
            <TemplateRule Parameters="PsetName[Value]='BR' AND
                AttributName[Value]='Spannweite' AND Wert[Value]&lt;30
                AND Wert[Value]>0"/>
        </TemplateRules>
    </Concept>
    <Concept uuid="00000036-0936-0936-0000-000002907145"
        name="Bestandteile Ueberbau, Unterbau und Fahrbahn" code="">
        <Definitions>
            <Definition>
                <Body lang="en"></Body>
            </Definition>
        </Definitions>
        <Template ref="AggregationIsDecomposedOhneAE"/>
        <Requirements>
            <Requirement applicability="export"
                exchangeRequirement="ModelValidation"
                requirement="mandatory"/>
        </Requirements>
        <TemplateRules operator="and">
            <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
                AND RelatedObjects[Exists]=TRUE AND
                PartKlassifikation[Value]='br.ue'"/>
            <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
                AND RelatedObjects[Exists]=TRUE AND
                PartKlassifikation[Value]='br.ub'"/>
        </TemplateRules>

```

```

        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
            AND RelatedObjects[Exists]=TRUE AND
            PartKlassifikation[Value]='ob.bet.100'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907MMM"
    name="Bestandteile Ueberbau, Unterbau und Fahrbahn über die
    Klassifikation" code="">
    <Definitions>
        <Definition>
            <Body lang="en"></Body>
        </Definition>
    </Definitions>
    <Template ref="AggregationIsDecomposedMitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PSetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ue'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PSetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ub'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PSetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='ob.bet.100'"/>
    </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907657"
    name="br.ue Ueberbau" applicableRootEntity=
    "IfcBuildingElementProxy">
    <Applicability>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PSetName[Value]=#
                'Allgemeine_Eigenschaften' AND
                AttributName[Value]='bauteilKlassifikation'
                AND Wert[Value]='br.ue'"/>
        </TemplateRules>
    </Applicability>
</Concepts>
<Concept uuid="00000036-0936-0936-0000-000002907145"
    name="Bestandteile Brueckentragwerk und Kappe" code="">
    <Definitions>
        <Definition>
            <Body lang="en"></Body>
        </Definition>
    </Definitions>
    <Template ref="AggregationIsDecomposedOhneAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
            AND RelatedObjects[Exists]=TRUE AND
            PartKlassifikation[Value]='br.ue.10'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE

```

```

        AND RelatedObjects[Exists]=TRUE AND
        PartKlassifikation[Value]='br.ue.20'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907MMM"
    name="Bestandteile Brueckentragwerk und Kappe über die
    Klassifikation" code="">
    <Definitions>
        <Definition>
            <Body lang="en"></Body>
        </Definition>
    </Definitions>
    <Template ref="AggregationIsDecomposedMitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PsetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ue.10' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Brueckentragwerk'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PsetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ue.20' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Kappe_L'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PsetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ue.20' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Kappe_R'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907SSS"
    name="Ist Teil einer Bruecke ohne AE" code="">
    <Definitions>
        <Definition>
            <Body lang="en"></Body>
        </Definition>
    </Definitions>
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            IstTeilVon[Value]='br'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907JJI"
    name="Ist Teil eine Bruecke mit AE" code="">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">

```

```

    <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
      RelatingObject [Exists]=TRUE AND
      PSetZuweisung [Exists]=TRUE AND
      PsetName [Value]='Allgemeine_Eigenschaften' AND
      PropertyName [Value]='bauteilKlassifikation' AND
      PropertyWert [Value]='br' AND
      PropertyzweiName [Value]='name' AND
      PropertyzweiWert [Value]='Testbruecke' "/>
  </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907757"
  name="br.ue.10 Brueckentragwerk"
  applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
      <TemplateRule Parameters="PsetName [Value]=
        'Allgemeine_Eigenschaften' AND AttributName [Value]=
        'bauteilKlassifikation' AND Wert [Value]='br.ue.10' "/>
    </TemplateRules>
  </Applicability>
  <Concepts>
    <Concept uuid="00000036-0936-0936-0000-000002907145"
      name="Bestandteile Brueckenplatte und Brueckentraeger"
      code="">
      <Template ref="AggregationIsDecomposedOhneAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="IsDecomposedBy [Exists]=TRUE
          AND RelatedObjects [Exists]=TRUE AND
          PartKlassifikation [Value]='br.ue.11' "/>
        <TemplateRule Parameters="IsDecomposedBy [Exists]=TRUE
          AND RelatedObjects [Exists]=TRUE AND
          PartKlassifikation [Value]='br.ue.12' "/>
      </TemplateRules>
    </Concept>
    <Concept uuid="00000036-0936-0936-0000-000002907MMM"
      name="Bestandteile Brueckentragwerk und Brueckentraeger über
      die Klassifikation" code="">
      <Template ref="AggregationIsDecomposedMitAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="IsDecomposedBy [Exists]=TRUE AND
          RelatedObjects [Exists]=TRUE AND
          PSetZuweisung [Exists]=TRUE AND
          PsetName [Value]= 'Allgemeine_Eigenschaften' AND
          PropertyName [Value]='bauteilKlassifikation' AND
          PropertyWert [Value]='br.ue.11' AND
          PropertyzweiName [Value]='name' AND
          PropertyzweiWert [Value]='Brueckenplatte' "/>
        <TemplateRule Parameters="IsDecomposedBy [Exists]=TRUE AND
          RelatedObjects [Exists]=TRUE AND
          PSetZuweisung [Exists]=TRUE AND
          PsetName [Value]='Allgemeine_Eigenschaften' AND
          PropertyName [Value]='bauteilKlassifikation' AND
          PropertyWert [Value]='br.ue.12' AND
          PropertyzweiName [Value]='name' AND
          PropertyzweiWert [Value]='Stahltraeger 1' "/>
        <TemplateRule Parameters="IsDecomposedBy [Exists]=TRUE AND
          RelatedObjects [Exists]=TRUE AND
          PSetZuweisung [Exists]=TRUE AND
          PsetName [Value]='Allgemeine_Eigenschaften' AND
          PropertyName [Value]='bauteilKlassifikation' AND

```

```

        PropertyWert[Value]='br.ue.12' AND
        PropertyzweiName[Value]='name' AND
        PropertyzweiWert[Value]='Stahltraeger_2'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
        RelatedObjects[Exists]=TRUE AND
        PSetZuweisung[Exists]=TRUE AND
        PsetName[Value]='Allgemeine_Eigenschaften' AND
        PropertyName[Value]='bauteilKlassifikation' AND
        PropertyWert[Value]='br.ue.12' AND
        PropertyzweiName[Value]='name' AND
        PropertyzweiWert[Value]='Stahltraeger_3'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
        RelatedObjects[Exists]=TRUE AND
        PSetZuweisung[Exists]=TRUE AND
        PsetName[Value]='Allgemeine_Eigenschaften' AND
        PropertyName[Value]='bauteilKlassifikation' AND
        PropertyWert[Value]='br.ue.12' AND
        PropertyzweiName[Value]='name' AND
        PropertyzweiWert[Value]='Stahltraeger_4'"/>
</TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907SSS"
    name="Ist Teil eines Ueberbaus ohne AE" code="">
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            IstTeilVon[Value]='br.ue'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907JJI"
    name="Ist Teil eines Ueberbaus mit AE" code="">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PsetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ue' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Ueberbau'"/>
    </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907857"
    name="br.ue.11 Brueckenplatte"
    applicableRootEntity="IfcBuildingElementProxy">
    <Applicability>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PsetName[Value]=
                'Allgemeine_Eigenschaften' AND AttributeName[Value]=
                'bauteilKlassifikation' AND Wert[Value]='br.ue.11'"/>
        </TemplateRules>
    </Applicability>
</Concepts>
<Concept uuid="00000036-0936-0936-0000-000002907SSS"
    name="Ist Teil eines Brueckentragwerks ohne AE" code="">
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
        <Requirement applicability="export"

```

```

        exchangeRequirement="ModelValidation"
        requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
            RelatingObject [Exists]=TRUE AND
            IstTeilVon [Value]='br.ue.10'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907JJI"
    name="Ist Teil eines Brueckentragwerks mit AE" code="">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
            RelatingObject [Exists]=TRUE AND
            PSetZuweisung [Exists]=TRUE AND
            PsetName [Value]='Allgemeine_Eigenschaften' AND
            PropertyName [Value]='bauteilKlassifikation' AND
            PropertyWert [Value]='br.ue.10' AND
            PropertyzweiName [Value]='name' AND
            PropertyzweiWert [Value]='Brueckentragwerk'"/>
    </TemplateRules>
</Concept>
<Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0b9D"
    name="Ist aus Beton der Klasse C40/50">
    <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
    <Requirements>
        <Requirement applicability="import"
            requirement="mandatory"
            exchangeRequirement="ModelValidation" />
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Assoziation [Exists]=TRUE AND
            VerknuepftesMaterial [Exists]=TRUE AND
            MaterialName [Value]='C40/50' AND
            MaterialKategorie [Value]='Stahlbeton'"/>
    </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907957"
    name="br.ue.12 Brueckentraeger"
    applicableRootEntity="IfcBuildingElementProxy">
    <Applicability>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PsetName [Value]=
                'Allgemeine_Eigenschaften' AND AttributName [Value]=
                'bauteilKlassifikation' AND Wert [Value]='br.ue.12'"/>
        </TemplateRules>
    </Applicability>
    <Concepts>
        <Concept uuid="00000036-0936-0936-0000-000002907SSS"
            name="Ist Teil eines Brueckentragwerks ohne AE" code="">
            <Template ref="AggregationDecomposesOhneAE"/>
            <Requirements>
                <Requirement applicability="export"
                    exchangeRequirement="ModelValidation"
                    requirement="mandatory"/>
            </Requirements>
            <TemplateRules operator="and">
                <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
                    RelatingObject [Exists]=TRUE AND
                    IstTeilVon [Value]='br.ue.10'"/>
            </TemplateRules>
        </Concept>
    </Concepts>
    <Concept uuid="00000036-0936-0936-0000-000002907JJI"
        name="Ist Teil eines Brueckentragwerks mit AE" code="">

```

```

<Template ref="AggregationDecomposesmitAE"/>
<Requirements>
  <Requirement applicability="export"
    exchangeRequirement="ModelValidation"
    requirement="mandatory"/>
</Requirements>
<TemplateRules operator="and">
  <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
    RelatingObject[Exists]=TRUE AND
    PSetZuweisung[Exists]=TRUE AND
    PsetName[Value]='Allgemeine_Eigenschaften' AND
    PropertyName[Value]='bauteilKlassifikation' AND
    PropertyWert[Value]='br.ue.10' AND
    PropertyzweiName[Value]='name' AND
    PropertyzweiWert[Value]='Brueckentragwerk'"/>
</TemplateRules>
</Concept>
<Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0BBB"
  name="Ist aus Stahl der Klasse S455">
  <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
  <Requirements>
    <Requirement applicability="import"
      requirement="mandatory"
      exchangeRequirement="ModelValidation" />
  </Requirements>
  <TemplateRules operator="and">
    <TemplateRule Parameters="Assoziation[Exists]=TRUE AND
      VerknuepftesMaterial[Exists]=TRUE AND
      MaterialName[Value]='S455' AND
      MaterialKategorie[Value]='Stahl'"/>
  </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907157"
  name="br.ue.20 Kappe"
  applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
      <TemplateRule Parameters="PsetName[Value]=
        'Allgemeine_Eigenschaften' AND AttributName[Value]=
        'bauteilKlassifikation' AND Wert[Value]='br.ue.20'"/>
    </TemplateRules>
  </Applicability>
  <Concepts>
    <Concept uuid="00000036-0936-0936-0000-000002907SSS"
      name="Ist Teil eines Ueberbaus ohne AE" code="">
      <Template ref="AggregationDecomposesOhneAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND
          IstTeilVon[Value]='br.ue'"/>
      </TemplateRules>
    </Concept>
    <Concept uuid="00000036-0936-0936-0000-000002907JJI"
      name="Ist Teil eines Ueberbaus mit AE" code="">
      <Template ref="AggregationDecomposesmitAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND
          PSetZuweisung[Exists]=TRUE AND
          PsetName[Value]='Allgemeine_Eigenschaften' AND

```

```

        PropertyName[Value]='bauteilKlassifikation' AND
        PropertyWert[Value]='br.ue' AND
        PropertyzweiName[Value]='name' AND
        PropertyzweiWert[Value]='Ueberbau'"/>
    </TemplateRules>
</Concept>
<Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0DDD"
    name="Ist aus Ortbeton der Klasse C20/25">
    <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
    <Requirements>
        <Requirement applicability="import" requirement="mandatory"
            exchangeRequirement="ModelValidation" />
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Assoziation[Exists]=TRUE AND
            VerknuepftesMaterial[Exists]=TRUE AND
            MaterialName[Value]='C20/25' AND
            MaterialBeschreibung[Value]='Ortbeton' AND
            MaterialKategorie[Value]='Beton'"/>
    </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907257"
    name="br.ub Unterbau" applicableRootEntity=
    "IfcBuildingElementProxy">
    <Applicability>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PsetName[Value]=
                'Allgemeine_Eigenschaften' AND AttributName[Value]=
                'bauteilKlassifikation' AND Wert[Value]='br.ub'"/>
        </TemplateRules>
    </Applicability>
    <Concepts>
        <Concept uuid="00000036-0936-0936-0000-000002907145"
            name="Bestandteile Widerlager und/oder Stütze" code="">
            <Template ref="AggregationIsDecomposedOhneAE"/>
            <Requirements>
                <Requirement applicability="export"
                    exchangeRequirement="ModelValidation"
                    requirement="mandatory"/>
            </Requirements>
            <TemplateRules operator="or">
                <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
                    AND RelatedObjects[Exists]=TRUE AND
                    PartKlassifikation[Value]='br.ub.10'"/>
                <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
                    AND RelatedObjects[Exists]=TRUE AND
                    PartKlassifikation[Value]='br.220'"/>
            </TemplateRules>
        </Concept>
        <Concept uuid="00000036-0936-0936-0000-000002907MMM"
            name="Bestandteile Widerlager und/oder Stütze über die
            Klassifikation" code="">
            <Template ref="AggregationIsDecomposedMitAE"/>
            <Requirements>
                <Requirement applicability="export"
                    exchangeRequirement="ModelValidation"
                    requirement="mandatory"/>
            </Requirements>
            <TemplateRules operator="and">
                <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
                    RelatedObjects[Exists]=TRUE AND
                    PSetZuweisung[Exists]=TRUE AND
                    PSetName[Value]='Allgemeine_Eigenschaften' AND
                    PropertyName[Value]='bauteilKlassifikation' AND
                    PropertyWert[Value]='br.ub.10' AND
                    PropertyzweiName[Value]='name' AND
                    PropertyzweiWert[Value]='Widerlager_1'"/>
                <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
                    RelatedObjects[Exists]=TRUE AND
                    PSetZuweisung[Exists]=TRUE AND

```

```

        PsetName[Value]= 'Allgemeine_Eigenschaften' AND
        PropertyName[Value]='bauteilKlassifikation' AND
        PropertyWert[Value]='br.ub.10' AND
        PropertyzweiName[Value]='name' AND
        PropertyzweiWert[Value]='Widerlager_2'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907SSS"
    name="Ist Teil einer Bruecke ohne AE" code="">
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            IstTeilVon[Value]='br'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907JJI"
    name="Ist Teil einer Bruecke mit AE" code="">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            PsetZuweisung[Exists]=TRUE AND
            PsetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Testbruecke'"/>
    </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907357"
    name="br.ub.10 Widerlager 1" applicableRootEntity=
    "IfcBuildingElementProxy">
    <Applicability>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PsetName[Value]=
                'Allgemeine_Eigenschaften' AND AttributName[Value]=
                'bauteilKlassifikation' AND Wert[Value]='br.ub.10'"/>
            <TemplateRule Parameters="PsetName[Value]=
                'Allgemeine_Eigenschaften' AND AttributName[Value]=
                'name' AND Wert[Value]='Widerlager_1'"/>
        </TemplateRules>
    </Applicability>
</Concepts>
<Concept uuid="00000036-0936-0936-0000-000002907145"
    name="Bestandteile des Widerlagers" code="">
    <Template ref="AggregationIsDecomposedOhneAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="or">
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
            AND RelatedObjects[Exists]=TRUE AND
            PartKlassifikation[Value]='br.ub.11'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
            AND RelatedObjects[Exists]=TRUE AND
            PartKlassifikation[Value]='br.ub.12'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE

```

```

        AND RelatedObjects[Exists]=TRUE AND
        PartKlassifikation[Value]='br.ub.13'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
    AND RelatedObjects[Exists]=TRUE AND
    PartKlassifikation[Value]='br.ub.14'"/>
</TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907MMM"
    name="Bestandteile des Widerlagers über die Klassifikation"
    code="">
    <Template ref="AggregationIsDecomposedMitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PSetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ub.11' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Widerlagerwand_WL1'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PSetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ub.12' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Fluegelwand_rechts_WL1'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PSetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ub.12' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Fluegelwand_links_WL1'"/>
        <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
            RelatedObjects[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PSetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ub.13' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Widerlagerfundament_WL1'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907SSS"
    name="Ist Teil eines Unterbaus ohne AE" code="">
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            IstTeilVon[Value]='br.ub'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907JJI"
    name="Ist Teil eines Unterbaus mit AE" code="">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>

```

```

<TemplateRules operator="and">
  <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
    RelatingObject[Exists]=TRUE AND
    PSetZuweisung[Exists]=TRUE AND
    PSetName[Value]= 'Allgemeine_Eigenschaften' AND
    PropertyName[Value]='bauteilKlassifikation' AND
    PropertyWert[Value]='br.ub' AND
    PropertyzweiName[Value]='name' AND
    PropertyzweiWert[Value]='Unterbau'"/>
</TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907357"
  name="br.ub.10 Widerlager 2"
  applicableRootEntity="IfcBuildingElementProxy">
<Applicability>
  <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
  <TemplateRules operator="and">
    <TemplateRule Parameters="PSetName[Value]=
      'Allgemeine_Eigenschaften' AND AttributName[Value]=
      'bauteilKlassifikation' AND Wert[Value]='br.ub.10'"/>
    <TemplateRule Parameters="PSetName[Value]=
      'Allgemeine_Eigenschaften' AND AttributName[Value]='name'
      AND Wert[Value]='Widerlager_2'"/>
  </TemplateRules>
</Applicability>
</Concepts>
<Concept uuid="00000036-0936-0936-0000-000002907145"
  name="Bestandteile des Widerlagers" code="">
  <Template ref="AggregationIsDecomposedOhneAE"/>
  <Requirements>
    <Requirement applicability="export"
      exchangeRequirement="ModelValidation" requirement="mandatory"/>
  </Requirements>
  <TemplateRules operator="or">
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
      AND RelatedObjects[Exists]=TRUE AND
      PartKlassifikation[Value]='br.ub.11'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
      AND RelatedObjects[Exists]=TRUE AND
      PartKlassifikation[Value]='br.ub.12'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
      AND RelatedObjects[Exists]=TRUE AND
      PartKlassifikation[Value]='br.ub.13'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE
      AND RelatedObjects[Exists]=TRUE AND
      PartKlassifikation[Value]='br.ub.14'"/>
  </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907MMM"
  name="Bestandteile des Widerlagers über die Klassifikation"
  code="">
  <Template ref="AggregationIsDecomposedMitAE"/>
  <Requirements>
    <Requirement applicability="export"
      exchangeRequirement="ModelValidation"
      requirement="mandatory"/>
  </Requirements>
  <TemplateRules operator="and">
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
      RelatedObjects[Exists]=TRUE AND
      PSetZuweisung[Exists]=TRUE AND
      PSetName[Value]= 'Allgemeine_Eigenschaften' AND
      PropertyName[Value]='bauteilKlassifikation' AND
      PropertyWert[Value]='br.ub.11' AND
      PropertyzweiName[Value]='name' AND
      PropertyzweiWert[Value]='Widerlagerwand WL2'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
      RelatedObjects[Exists]=TRUE AND
      PSetZuweisung[Exists]=TRUE AND
      PSetName[Value]= 'Allgemeine_Eigenschaften' AND
      PropertyName[Value]='bauteilKlassifikation' AND

```

```

        PropertyWert[Value]='br.ub.12' AND
        PropertyzweiName[Value]='name' AND
        PropertyzweiWert[Value]='Fluegelwand rechts WL2'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
        RelatedObjects[Exists]=TRUE AND
        PSetZuweisung[Exists]=TRUE AND
        PsetName[Value]= 'Allgemeine_Eigenschaften' AND
        PropertyName[Value]='bauteilKlassifikation' AND
        PropertyWert[Value]='br.ub.12' AND
        PropertyzweiName[Value]='name' AND
        PropertyzweiWert[Value]='Fluegelwand links WL2'"/>
    <TemplateRule Parameters="IsDecomposedBy[Exists]=TRUE AND
        RelatedObjects[Exists]=TRUE AND
        PSetZuweisung[Exists]=TRUE AND
        PsetName[Value]='Allgemeine_Eigenschaften' AND
        PropertyName[Value]='bauteilKlassifikation' AND
        PropertyWert[Value]='br.ub.13' AND
        PropertyzweiName[Value]='name' AND
        PropertyzweiWert[Value]='Widerlagerfundament_WL2'"/>
</TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907SSS"
    name="Ist Teil eines Unterbaus ohne AE" code="">
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            IstTeilVon[Value]='br.ub'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907JJI"
    name="Ist Teil eines Unterbaus mit AE" code="">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PsetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ub' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Unterbau'"/>
    </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907BBB"
    name="br.ub.11 Widerlagerwand 1"
    applicableRootEntity="IfcBuildingElementProxy">
    <Applicability>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
        <TemplateRule Parameters="PsetName[Value]=
            'Allgemeine_Eigenschaften' AND AttributeName[Value]=
            'bauteilKlassifikation' AND Wert[Value]='br.ub.11'"/>
        <TemplateRule Parameters="PsetName[Value]='Bauteil' AND
            AttributeName[Value]='technischerPlatz' AND
            Wert[Value]='WLP1'"/>
    </TemplateRules>
</Applicability>
<Concepts>
    <Concept uuid="00000036-0936-0936-0000-000002907STT"
        name="Ist Teil eines Widerlagers ohne AE" code="">

```

```

<Template ref="AggregationDecomposesOhneAE"/>
<Requirements>
  <Requirement applicability="export"
    exchangeRequirement="ModelValidation"
    requirement="mandatory"/>
</Requirements>
<TemplateRules operator="and">
  <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
    RelatingObject[Exists]=TRUE AND
    IstTeilVon[Value]='br.ub.10'"/>
</TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907GGG"
  name="Ist Teil eines Widerlagers mit AE" code="">
<Template ref="AggregationDecomposesmitAE"/>
<Requirements>
  <Requirement applicability="export"
    exchangeRequirement="ModelValidation"
    requirement="mandatory"/>
</Requirements>
<TemplateRules operator="and">
  <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
    RelatingObject[Exists]=TRUE AND
    PSetZuweisung[Exists]=TRUE AND
    PsetName[Value]='Allgemeine_Eigenschaften' AND
    PropertyName[Value]='bauteilKlassifikation' AND
    PropertyWert[Value]='br.ub.10' AND
    PropertyzweiName[Value]='name' AND
    PropertyzweiWert[Value]='Widerlager_1'"/>
</TemplateRules>
</Concept>
<Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0FF"
  name="Ist aus Ortbeton der Klasse C30/37">
<Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
<Requirements>
  <Requirement applicability="import"
    requirement="mandatory"
    exchangeRequirement="ModelValidation" />
</Requirements>
<TemplateRules operator="and">
  <TemplateRule Parameters="Assoziation[Exists]=TRUE AND
    VerknuepftesMaterial[Exists]=TRUE AND
    MaterialName[Value]='C30/37' AND
    MaterialBeschreibung[Value]='Ortbeton' AND
    MaterialKategorie[Value]='Stahlbeton'"/>
</TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907BBB"
  name="br.ub.11 Widerlagerwand 2"
  applicableRootEntity="IfcBuildingElementProxy">
<Applicability>
  <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
  <TemplateRules operator="and">
    <TemplateRule Parameters="PsetName[Value]=
      'Allgemeine_Eigenschaften' AND AttributName[Value]=
      'bauteilKlassifikation' AND Wert[Value]='br.ub.11'"/>
    <TemplateRule Parameters="PsetName[Value]='Bauteil' AND
      AttributName[Value]='technischerPlatz' AND
      Wert[Value]='WLP2'"/>
  </TemplateRules>
</Applicability>
</Concepts>
<Concept uuid="00000036-0936-0936-0000-000002907STT"
  name="Ist Teil eines Widerlagers ohne AE" code="">
<Template ref="AggregationDecomposesOhneAE"/>
<Requirements>
  <Requirement applicability="export"
    exchangeRequirement="ModelValidation"
    requirement="mandatory"/>
</Requirements>
<TemplateRules operator="and">

```

```

        <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
            RelatingObject [Exists]=TRUE AND
            IstTeilVon [Value]='br.ub.10'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907GGG"
    name="Ist Teil eines Widerlagers mit AE" code="">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
            RelatingObject [Exists]=TRUE AND
            PSetZuweisung [Exists]=TRUE AND
            PSetName [Value]='Allgemeine_Eigenschaften' AND
            PropertyName [Value]='bauteilKlassifikation' AND
            PropertyWert [Value]='br.ub.10' AND
            PropertyzweiName [Value]='name' AND
            PropertyzweiWert [Value]='Widerlager_2'"/>
    </TemplateRules>
</Concept>
<Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0FF"
    name="Ist aus Ortbeton der Klasse C30/37">
    <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
    <Requirements>
        <Requirement applicability="import"
            requirement="mandatory"
            exchangeRequirement="ModelValidation" />
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Assoziation [Exists]=TRUE AND
            VerknuepftesMaterial [Exists]=TRUE AND
            MaterialName [Value]='C30/37' AND
            MaterialBeschreibung [Value]='Ortbeton' AND
            MaterialKategorie [Value]='Stahlbeton'"/>
    </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907194"
    name="br.ub.12 Fluegelwand links 1"
    applicableRootEntity="IfcBuildingElementProxy">
    <Applicability>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PSetName [Value]=
                'Allgemeine_Eigenschaften' AND
                AttributName [Value]='bauteilKlassifikation' AND
                Wert [Value]='br.ub.12'"/>
            <TemplateRule Parameters="PSetName [Value]=
                'Allgemeine_Eigenschaften' AND
                AttributName [Value]='name' AND
                Wert [Value]='Fluegelwand_links_WL1'"/>
        </TemplateRules>
    </Applicability>
</Concepts>
<Concept uuid="00000036-0936-0936-0000-000002907RRR"
    name="Ist Teil eines Widerlagers" code="">
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
            RelatingObject [Exists]=TRUE AND
            IstTeilVon [Value]='br.ub.10'"/>
    </TemplateRules>
</Concept>

```

```

<Concept uuid="00000036-0936-0936-0000-000002907GHH"
  name="Ist Teil eines Widerlagers mit AE" code="">
  <Template ref="AggregationDecomposesmitAE"/>
  <Requirements>
    <Requirement applicability="export"
      exchangeRequirement="ModelValidation"
      requirement="mandatory"/>
  </Requirements>
  <TemplateRules operator="and">
    <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
      RelatingObject[Exists]=TRUE AND
      PsetZuweisung[Exists]=TRUE AND
      PsetName[Value]='Allgemeine_Eigenschaften' AND
      PropertyName[Value]='bauteilKlassifikation' AND
      PropertyWert[Value]='br.ub.10' AND
      PropertyzweiName[Value]='name' AND
      PropertyzweiWert[Value]='Widerlager_1'"/>
  </TemplateRules>
</Concept>
<Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0EEE"
  name="Ist aus Ortbeton der Klasse C30/37">
  <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
  <Requirements>
    <Requirement applicability="import" requirement="mandatory"
      exchangeRequirement="ModelValidation" />
  </Requirements>
  <TemplateRules operator="and">
    <TemplateRule Parameters="Assoziation[Exists]=TRUE AND
      VerknuepftesMaterial[Exists]=TRUE AND
      MaterialName[Value]='C30/37' AND
      MaterialBeschreibung[Value]='Ortbeton' AND
      MaterialKategorie[Value]='Stahlbeton'"/>
  </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907194"
  name="br.ub.12 Fluegelwand rechts 1"
  applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
      <TemplateRule Parameters="PsetName[Value]=
        'Allgemeine_Eigenschaften' AND
        AttributName[Value]='bauteilKlassifikation' AND
        Wert[Value]='br.ub.12'"/>
      <TemplateRule Parameters="PsetName[Value]=
        'Allgemeine_Eigenschaften' AND
        AttributName[Value]='name' AND
        Wert[Value]='Fluegelwand_rechts_WL1'"/>
    </TemplateRules>
  </Applicability>
</Concepts>
<Concept uuid="00000036-0936-0936-0000-000002907RRR"
  name="Ist Teil eines Widerlagers" code="">
  <Template ref="AggregationDecomposesOhneAE"/>
  <Requirements>
    <Requirement applicability="export"
      exchangeRequirement="ModelValidation"
      requirement="mandatory"/>
  </Requirements>
  <TemplateRules operator="and">
    <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
      RelatingObject[Exists]=TRUE AND
      IstTeilVon[Value]='br.ub.10'"/>
  </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907GHH"
  name="Ist Teil eines Widerlagers mit AE" code="">
  <Template ref="AggregationDecomposesmitAE"/>
  <Requirements>
    <Requirement applicability="export"
      exchangeRequirement="ModelValidation"

```

```

        requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
            RelatingObject [Exists]=TRUE AND
            PSetZuweisung [Exists]=TRUE AND
            PsetName [Value]='Allgemeine_Eigenschaften' AND
            PropertyName [Value]='bauteilKlassifikation' AND
            PropertyWert [Value]='br.ub.10' AND
            PropertyzweiName [Value]='name' AND
            PropertyzweiWert [Value]='Widerlager_1'"/>
    </TemplateRules>
</Concept>
<Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0EEE"
    name="Ist aus Ortbeton der Klasse C30/37">
    <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
    <Requirements>
        <Requirement applicability="import"
            requirement="mandatory"
            exchangeRequirement="ModelValidation" />
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Assoziation [Exists]=TRUE AND
            VerknuepftesMaterial [Exists]=TRUE AND
            MaterialName [Value]='C30/37' AND
            MaterialBeschreibung [Value]='Ortbeton' AND
            MaterialKategorie [Value]='Stahlbeton'"/>
    </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907194"
    name="br.ub.12 Fluegelwand links 2"
    applicableRootEntity="IfcBuildingElementProxy">
    <Applicability>
        <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
        <TemplateRules operator="and">
            <TemplateRule Parameters="PsetName [Value]=
                'Allgemeine_Eigenschaften' AND
                AttributName [Value]='bauteilKlassifikation' AND
                Wert [Value]='br.ub.12'"/>
            <TemplateRule Parameters="PsetName [Value]=
                'Allgemeine_Eigenschaften' AND
                AttributName [Value]='name' AND
                Wert [Value]='Fluegelwand_links_WL2'"/>
        </TemplateRules>
    </Applicability>
    <Concepts>
        <Concept uuid="00000036-0936-0936-0000-000002907RRR"
            name="Ist Teil eines Widerlagers" code="">
            <Template ref="AggregationDecomposesOhneAE"/>
            <Requirements>
                <Requirement applicability="export"
                    exchangeRequirement="ModelValidation"
                    requirement="mandatory"/>
            </Requirements>
            <TemplateRules operator="and">
                <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
                    RelatingObject [Exists]=TRUE AND
                    IstTeilVon [Value]='br.ub.10'"/>
            </TemplateRules>
        </Concept>
        <Concept uuid="00000036-0936-0936-0000-000002907GHH"
            name="Ist Teil eines Widerlagers mit AE" code="">
            <Template ref="AggregationDecomposesmitAE"/>
            <Requirements>
                <Requirement applicability="export"
                    exchangeRequirement="ModelValidation"
                    requirement="mandatory"/>
            </Requirements>
            <TemplateRules operator="and">
                <TemplateRule Parameters="Decomposes [Exists]=TRUE AND
                    RelatingObject [Exists]=TRUE AND

```

```

        PSetZuweisung[Exists]=TRUE AND
        PSetName[Value]='Allgemeine_Eigenschaften' AND
        PropertyName[Value]='bauteilKlassifikation' AND
        PropertyWert[Value]='br.ub.10' AND
        PropertyzweiName[Value]='name' AND
        PropertyzweiWert[Value]='Widerlager_2'"/>
    </TemplateRules>
</Concept>
<Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0EEE"
    name="Ist aus Ortbeton der Klasse C30/37">
    <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
    <Requirements>
        <Requirement applicability="import"
            requirement="mandatory"
            exchangeRequirement="ModelValidation" />
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Assoziation[Exists]=TRUE AND
            VerknuepftesMaterial[Exists]=TRUE AND
            MaterialName[Value]='C30/37' AND
            MaterialBeschreibung[Value]='Ortbeton' AND
            MaterialKategorie[Value]='Stahlbeton'"/>
    </TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907194"
    name="br.ub.12 Fluegelwand rechts 2"
    applicableRootEntity="IfcBuildingElementProxy">
<Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
        <TemplateRule Parameters="PSetName[Value]=
            'Allgemeine_Eigenschaften' AND AttributName[Value]=
            'bauteilKlassifikation' AND Wert[Value]='br.ub.12'"/>
        <TemplateRule Parameters="PSetName[Value]=
            'Allgemeine_Eigenschaften' AND
            AttributName[Value]='name' AND
            Wert[Value]='Fluegelwand_rechts_WL2'"/>
    </TemplateRules>
</Applicability>
<Concepts>
<Concept uuid="00000036-0936-0936-0000-000002907RRR"
    name="Ist Teil eines Widerlagers" code="">
    <Template ref="AggregationDecomposesOhneAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            IstTeilVon[Value]='br.ub.10'"/>
    </TemplateRules>
</Concept>
<Concept uuid="00000036-0936-0936-0000-000002907GHH"
    name="Ist Teil eines Widerlagers mit AE" code="">
    <Template ref="AggregationDecomposesmitAE"/>
    <Requirements>
        <Requirement applicability="export"
            exchangeRequirement="ModelValidation"
            requirement="mandatory"/>
    </Requirements>
    <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
            RelatingObject[Exists]=TRUE AND
            PSetZuweisung[Exists]=TRUE AND
            PSetName[Value]='Allgemeine_Eigenschaften' AND
            PropertyName[Value]='bauteilKlassifikation' AND
            PropertyWert[Value]='br.ub.10' AND
            PropertyzweiName[Value]='name' AND
            PropertyzweiWert[Value]='Widerlager_2'"/>
    </TemplateRules>
</Concept>

```

```

    </TemplateRules>
  </Concept>
  <Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0EEE"
    name="Ist aus Ortbeton der Klasse C30/37">
    <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
    <Requirements>
      <Requirement applicability="import"
        requirement="mandatory"
        exchangeRequirement="ModelValidation" />
    </Requirements>
    <TemplateRules operator="and">
      <TemplateRule Parameters="Assoziation[Exists]=TRUE AND
        VerknuepftesMaterial[Exists]=TRUE AND
        MaterialName[Value]='C30/37' AND
        MaterialBeschreibung[Value]='Ortbeton' AND
        MaterialKategorie[Value]='Stahlbeton' "/>
    </TemplateRules>
  </Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907367"
  name="br.ub.13 Widerlagerfundament_1"
  applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
      <TemplateRule Parameters="PsetName[Value]=
        'Allgemeine_Eigenschaften' AND
        AttributName[Value]='bauteilKlassifikation'
        AND Wert[Value]='br.ub.13' "/>
      <TemplateRule Parameters="PsetName[Value]=
        'Allgemeine_Eigenschaften' AND
        AttributName[Value]='name' AND
        Wert[Value]='Widerlagerfundament_WL1' "/>
    </TemplateRules>
  </Applicability>
  <Concepts>
    <Concept uuid="00000036-0936-0936-0000-000002907SSS"
      name="Ist Teil eines Widerlagers ohne AE" code="">
      <Template ref="AggregationDecomposesOhneAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory" />
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND
          IstTeilVon[Value]='br.ub.10' "/>
      </TemplateRules>
    </Concept>
    <Concept uuid="00000036-0936-0936-0000-000002907JJI"
      name="Ist Teil eines Widerlagers mit AE" code="">
      <Template ref="AggregationDecomposesmitAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory" />
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND
          PSetZuweisung[Exists]=TRUE AND
          PSetName[Value]='Allgemeine_Eigenschaften' AND
          PropertyName[Value]='bauteilKlassifikation' AND
          PropertyWert[Value]='br.ub.10' AND
          PropertyzweiName[Value]='name' AND
          PropertyzweiWert[Value]='Widerlager_1' "/>
      </TemplateRules>
    </Concept>
  </Concepts>
  <Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0GGG"
    name="Ist aus Ortbeton der Klasse C30/37">
    <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />

```

```

<Requirements>
  <Requirement applicability="import"
    requirement="mandatory"
    exchangeRequirement="ModelValidation" />
</Requirements>
<TemplateRules operator="and">
  <TemplateRule Parameters="Assoziation[Exists]=TRUE AND
    VerknuepftesMaterial[Exists]=TRUE AND
    MaterialName[Value]='C30/37' AND
    MaterialBeschreibung[Value]='Ortbeton' AND
    MaterialKategorie[Value]='Stahlbeton'"/>
</TemplateRules>
</Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907367"
  name="br.ub.13 Widerlagerfundament 2"
  applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
      <TemplateRule Parameters="PsetName[Value]=
        'Allgemeine_Eigenschaften' AND
        AttributName[Value]='bauteilKlassifikation' AND
        Wert[Value]='br.ub.13'"/>
      <TemplateRule Parameters="PsetName[Value]=
        'Allgemeine_Eigenschaften' AND
        AttributName[Value]='name' AND
        Wert[Value]='Widerlagerfundament_WL2'"/>
    </TemplateRules>
  </Applicability>
  <Concepts>
    <Concept uuid="00000036-0936-0936-0000-000002907SSS"
      name="Ist Teil eines Widerlagers ohne AE" code="">
      <Template ref="AggregationDecomposesOhneAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND
          IstTeilVon[Value]='br.ub.10'"/>
      </TemplateRules>
    </Concept>
    <Concept uuid="00000036-0936-0936-0000-000002907JJI"
      name="Ist Teil eines Widerlagers mit AE" code="">
      <Template ref="AggregationDecomposesmitAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND
          PSetZuweisung[Exists]=TRUE AND
          PSetName[Value]='Allgemeine_Eigenschaften' AND
          PropertyName[Value]='bauteilKlassifikation' AND
          PropertyWert[Value]='br.ub.10' AND
          PropertyzweiName[Value]='name' AND
          PropertyzweiWert[Value]='Widerlager_2'"/>
      </TemplateRules>
    </Concept>
    <Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0GGG"
      name="Ist aus Ortbeton der Klasse C30/37">
      <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
      <Requirements>
        <Requirement applicability="import"
          requirement="mandatory"
          exchangeRequirement="ModelValidation" />
      </Requirements>
  </Concepts>

```

```

    <TemplateRules operator="and">
      <TemplateRule Parameters="Assoziation[Exists]=TRUE AND
        VerknuepftesMaterial[Exists]=TRUE AND
        MaterialName[Value]='C30/37' AND
        MaterialBeschreibung[Value]='Ortbeton' AND#
        MaterialKategorie[Value]='Stahlbeton' "/>
    </TemplateRules>
  </Concept>
</Concepts>
</ConceptRoot>
<ConceptRoot uuid="00000036-0936-0936-0000-000002907457"
  name="ob.bet.100 FesteFahrbahn"
  applicableRootEntity="IfcBuildingElementProxy">
  <Applicability>
    <Template ref="69cb72cb-fd55-4e49-bef4-14ca32cd568b"/>
    <TemplateRules operator="and">
      <TemplateRule Parameters="PsetName[Value]=
        'Allgemeine_Eigenschaften' AND
        AttributName[Value]='bauteilKlassifikation' AND
        Wert[Value]='ob.bet.100' "/>
    </TemplateRules>
  </Applicability>
  <Concepts>
    <Concept uuid="00000036-0936-0936-0000-000002907SSS"
      name="Ist Teil einer Bruecke ohne AE" code="">
      <Template ref="AggregationDecomposesOhneAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND
          IstTeilVon[Value]='br' "/>
      </TemplateRules>
    </Concept>
    <Concept uuid="00000036-0936-0936-0000-000002907JJI"
      name="Ist Teil einer Bruecke mit AE" code="">
      <Template ref="AggregationDecomposesmitAE"/>
      <Requirements>
        <Requirement applicability="export"
          exchangeRequirement="ModelValidation"
          requirement="mandatory"/>
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Decomposes[Exists]=TRUE AND
          RelatingObject[Exists]=TRUE AND
          PSetZuweisung[Exists]=TRUE AND
          PsetName[Value]='Allgemeine_Eigenschaften' AND
          PropertyName[Value]='bauteilKlassifikation' AND
          PropertyWert[Value]='br' AND
          PropertyzweiName[Value]='name' AND
          PropertyzweiWert[Value]='Testbruecke' "/>
      </TemplateRules>
    </Concept>
    <Concept uuid="5b650aa3-6de8-4d5e-81a8-7285a3ce0MMM"
      name="Ist aus Gussasphalt">
      <Template ref="5f39a814-44a0-4fb3-8dc6-ddaa0560fc54" />
      <Requirements>
        <Requirement applicability="import"
          requirement="mandatory"
          exchangeRequirement="ModelValidation" />
      </Requirements>
      <TemplateRules operator="and">
        <TemplateRule Parameters="Assoziation[Exists]=TRUE AND
          VerknuepftesMaterial[Exists]=TRUE AND
          MaterialName[Value]='Asphalt' AND
          MaterialBeschreibung[Value]='Gussasphalt' AND
          MaterialKategorie[Value]='Bitumenmaterialien' "/>
      </TemplateRules>
    </Concept>
  </Concepts>

```

```
</ConceptRoot>  
</Roots>  
</ModelView></Views>  
</mvdXML>
```

ANLAGE 13 IFC-STEP ANPASSUNG DES BRÜCKENMODELLS 2

```
#30001= IFCBUILDINGELEMENTPROXY('Widerlager_1',#42,'Widerlager_1',$,,'br.ub.10',#90003,#90000,'1076496',.NOTDEFINED.);
#30002= IFCPROPERTYSINGLEVALUE('bauteilKlassifikation',$,IFCTEXT('br.ub.10'),$);
#30003= IFCPROPERTYSINGLEVALUE('bauteilName',$,IFCTEXT('Widerlager'),$);
#30004= IFCPROPERTYSINGLEVALUE('bemerkung',$,IFCTEXT('Metaobjekt'),$);
#30005= IFCPROPERTYSINGLEVALUE('KGK-Verzeichnisnummer',$,IFCTEXT('W1'),$);
#30006= IFCPROPERTYSINGLEVALUE('name',$,IFCTEXT('Widerlager_1'),$);
#30007= IFCPROPERTYSINGLEVALUE('identitaet',$,IFCTEXT('W1'),$);
#30008= IFCPROPERTYSET('AE_W1',#42,'Allgemeine_Eigenschaften','',(#30002,#30003,#30004,#30005,#30006,#30007));
#30009= IFCPROPERTYSINGLEVALUE('betriebszustand',$,IFCTEXT('aktiv'),$);
#30010= IFCPROPERTYSINGLEVALUE('technischerPlatz',$,IFCTEXT('WLP1'),$);
#30011= IFCPROPERTYSINGLEVALUE('technischerPlatzBezeichnung',$,IFCTEXT('Widerlagerplatz1'),$);
#30012= IFCPROPERTYSINGLEVALUE('material',$,IFCTEXT('Ortbeton'),$);
#30013= IFCPROPERTYSINGLEVALUE('materialEigenschaften',$,IFCTEXT('Ortbeton'),$);
#30014= IFCPROPERTYSET('B_W1',#42,'Bauteil','',(#30009,#30010,#30011,#30012,#30013));
#30015= IFCRELDEFINESBYPROPERTIES('AEW1',#42,$,$,($30001),#30008);
#30016= IFCRELDEFINESBYPROPERTIES('BWL1',#42,$,$,($30001),#30014);
#30017= IFCRELAGGREGATES('Widerlager_1_Zusammensetzung',#42,$,$,($30001,($20899,#19392,#20594,#17976));
#40001= IFCBUILDINGELEMENTPROXY('Widerlager_2',#42,'Widerlager_2',$,,'br.ub.10',#24366,#24360,'1430889',.NOTDEFINED.);
#40002= IFCPROPERTYSINGLEVALUE('bauteilKlassifikation',$,IFCTEXT('br.ub.10'),$);
#40003= IFCPROPERTYSINGLEVALUE('bauteilName',$,IFCTEXT('Widerlager'),$);
#40004= IFCPROPERTYSINGLEVALUE('bemerkung',$,IFCTEXT('Metaobjekt'),$);
#40005= IFCPROPERTYSINGLEVALUE('KGK-Verzeichnisnummer',$,IFCTEXT('W2'),$);
#40006= IFCPROPERTYSINGLEVALUE('name',$,IFCTEXT('Widerlager_2'),$);
#40007= IFCPROPERTYSINGLEVALUE('identitaet',$,IFCTEXT('W2'),$);
#40008= IFCPROPERTYSET('AE_W2',#42,'Allgemeine_Eigenschaften','',($40002,#40003,#40004,#40005,#40006,#40007));
#40009= IFCPROPERTYSINGLEVALUE('betriebszustand',$,IFCTEXT('aktiv'),$);
#40010= IFCPROPERTYSINGLEVALUE('technischerPlatz',$,IFCTEXT('WLP2'),$);
#40011= IFCPROPERTYSINGLEVALUE('technischerPlatzBezeichnung',$,IFCTEXT('Widerlagerplatz2'),$);
#40012= IFCPROPERTYSINGLEVALUE('material',$,IFCTEXT('Ortbeton'),$);
#40013= IFCPROPERTYSINGLEVALUE('materialEigenschaften',$,IFCTEXT('Ortbeton'),$);
#40014= IFCPROPERTYSET('B_W2',#42,'Bauteil','',($40009,#40010,#40011,#40012,#40013));
#40015= IFCRELDEFINESBYPROPERTIES('AEW2',#42,$,$,($40001),#40008);
#40016= IFCRELDEFINESBYPROPERTIES('BWL2',#42,$,$,($40001),#40014);
#40017= IFCRELAGGREGATES('Widerlager_2_Zusammensetzung',#42,$,$,($40001,($24367,#22885,#24064,#21091));
#50001= IFCBUILDINGELEMENTPROXY('Unterbau',#42,'Unterbau',$,,'br.ub',#17613,#17607,'731561',.NOTDEFINED.);
#50002= IFCPROPERTYSINGLEVALUE('bauteilKlassifikation',$,IFCTEXT('br.ub'),$);
#50003= IFCPROPERTYSINGLEVALUE('bauteilName',$,IFCTEXT('Unterbau'),$);
#50004= IFCPROPERTYSINGLEVALUE('bemerkung',$,IFCTEXT('Metaobjekt'),$);
#50005= IFCPROPERTYSINGLEVALUE('KGK-Verzeichnisnummer',$,IFCTEXT('UB'),$);
#50006= IFCPROPERTYSINGLEVALUE('name',$,IFCTEXT('Unterbau'),$);
#50007= IFCPROPERTYSINGLEVALUE('identitaet',$,IFCTEXT('UB'),$);
#50008= IFCPROPERTYSET('AE_UB',#42,'Allgemeine_Eigenschaften','',($50002,#50003,#50004,#50005,#50006,#50007));
#50009= IFCRELDEFINESBYPROPERTIES('AEUB',#42,$,$,($50001),#50008);
#50010= IFCRELAGGREGATES('Unterbau_Zusammensetzung',#42,$,$,($50001,($30001,#40001));
#60001= IFCBUILDINGELEMENTPROXY('Brueckentragwerk',#42,'Brueckentragwerk',$,,'br.ue.10',#1967,#1961,'386895',.NOTDEFINED.);
#60002= IFCPROPERTYSINGLEVALUE('bauteilKlassifikation',$,IFCTEXT('br.ue.10'),$);
#60003= IFCPROPERTYSINGLEVALUE('bauteilName',$,IFCTEXT('Brueckentragwerk'),$);
#60004= IFCPROPERTYSINGLEVALUE('bemerkung',$,IFCTEXT('Metaobjekt'),$);
#60005= IFCPROPERTYSINGLEVALUE('KGK-Verzeichnisnummer',$,IFCTEXT('BTW'),$);
#60006= IFCPROPERTYSINGLEVALUE('name',$,IFCTEXT('Brueckentragwerk'),$);
#60007= IFCPROPERTYSINGLEVALUE('identitaet',$,IFCTEXT('BTW'),$);
#60008= IFCPROPERTYSET('AE_BTW',#42,'Allgemeine_Eigenschaften','',($60002,#60003,#60004,#60005,#60006,#60007));
#60009= IFCRELDEFINESBYPROPERTIES('AEBTW',#42,$,$,($60001),#60008);
#60010= IFCRELAGGREGATES('Zusammensetzung_Brueckentragwerk',#42,$,$,($60001,($1968,#17615,#17677,#17733,#17789));
#70001= IFCBUILDINGELEMENTPROXY('Ueberbau',#42,'Ueberbau',$,,'br.ue',#1967,#1961,'386895',.NOTDEFINED.);
#70002= IFCPROPERTYSINGLEVALUE('bauteilKlassifikation',$,IFCTEXT('br.ue'),$);
#70003= IFCPROPERTYSINGLEVALUE('bauteilName',$,IFCTEXT('Ueberbau'),$);
#70004= IFCPROPERTYSINGLEVALUE('bemerkung',$,IFCTEXT('Metaobjekt'),$);
#70005= IFCPROPERTYSINGLEVALUE('KGK-Verzeichnisnummer',$,IFCTEXT('UEB'),$);
#70006= IFCPROPERTYSINGLEVALUE('name',$,IFCTEXT('Ueberbau'),$);
#70007= IFCPROPERTYSINGLEVALUE('identitaet',$,IFCTEXT('UEB'),$);
#70008= IFCPROPERTYSET('AE_UE',#42,'Allgemeine_Eigenschaften','',($70002,#70003,#70004,#70005,#70006,#70007));
#70009= IFCRELDEFINESBYPROPERTIES('AEUE',#42,$,$,($70001),#70008);
#70010= IFCRELAGGREGATES('Zusammensetzung_Ueberbau',#42,$,$,($70001,($60001,#2341,#2562));
#80000= IFCRELAGGREGATES('Zusammensetzung_Bruecke',#42,$,$,($1873,($70001,#50001,#2788));
```