



Technische Universität Berlin
Fachgebiet Bauinformatik
Prof. Dr.-Ing. Wolfgang Huhnt

Masterarbeit

Modellierung und Prüfung von IDS-basierten Informationsanforderungen in Open BIM Planungsworkflows

Erstellt von:

Rami Bali

Matrikel Nummer: **369458**

Erster Prüfer:

Prof. Dr.-Ing. Wolfgang Huhnt

Zweiter Prüfer:

Prof. Dr. Matthias Sundermeier

Betreuer:

Dr.-Ing. Maximilian Sternal

Berlin, Januar 2024

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Berlin, den

.....

Unterschrift

Inhaltsverzeichnis

ZUSAMMENFASSUNG.....	8
ABSTRACT	9
RÉSUMÉ.....	10
1 EINLEITUNG	11
1.1 PROBLEMATIK	13
1.2 ZIEL DER ARBEIT	13
1.3 STRUKTUR DER ARBEIT	14
2 IDS-MODELLIERUNG UND VALIDIERUNG: GRUNDLAGEN	16
2.1 IDS-FORMAT.....	16
2.1.1 IDS-Metadaten.....	16
2.1.2 Spezifikationen.....	17
2.1.3 Facettenparametertyp: Einfacher Wert vs. Einschränkung.....	18
2.1.4 Facettten.....	19
2.2 OBJEKTIVE ANALYSE DER INTEGRATION VON IDS IN DAS OPENBIM-ÖKOSYSTEM VON BUILDINGSMART.....	29
2.2.1 Information Delivery Manual (IDM):	29
2.2.2 buildingSMART Data Dictionary (bSDD)	30
2.2.3 BIM Collaboration Format (BCF).....	33
2.2.4 IFC-Projekt Bibliotheken und Objekt-Vorlagen	33
2.3 WORKFLOWS.....	34
3 VERWANDTE ARBEITEN	35
3.1 LITERATUR RECHERCHE.....	35
3.1.1 „Developing a BIM-based methodology to support renewable energy assessment of buildings“	35
3.1.2 „Creating Information Delivery Specifications using Linked Data“ [39].....	36
3.1.3 „Extending information delivery specification for linking distributed model checking services“ [40]	36
3.1.4 „Interoperable validation for IFC building models using open standards“ [41]	37
3.1.5 Use Case Management von buildingSMART	37
3.2 IMPLEMENTIERUNGEN	40
3.2.1 BIMcollab.....	40

3.2.2	<i>usBIM von ACCA</i>	41
3.2.3	<i>BlenderBIM IFC Tester</i>	42
3.2.4	<i>Solibri</i>	43
3.2.5	<i>xbim Flex</i>	43
3.2.6	<i>Plannerly</i>	44
3.2.7	<i>BIMQ</i>	45
3.2.8	<i>Andere Lösungen</i>	45
4	KONZEPTION UND IMPLEMENTIERUNG	47
4.1	ANFORDERUNGEN	47
4.2	ARCHITEKTUR DES WERKZEUGS	48
4.2.1	<i>Ausgangspunkt</i>	48
4.2.2	<i>Neuerungen und Erweiterungen</i>	49
4.3	IMPLEMENTIERUNGSDetails	49
4.3.1	<i>Erste Ebene: Erstellung einer neuen IDS-Datei</i>	49
4.3.2	<i>Zweite Ebene: Bearbeitung einer bestehenden IDS-Datei</i>	51
4.3.3	<i>Dritte Ebene: Prüfung von IDS-basierten Informationsanforderungen</i>	51
4.3.4	<i>Vierte Ebene: Erstellung des Berichts der etablierten Prüfung von IDS-basierten Informationsanforderungen</i>	52
5	BEISPIEL WORKFLOWS	53
5.1	INSTALLATION UND START DER ANWENDUNG:	53
5.2	ERSTE EBENE: ERSTELLUNG EINER NEUEN IDS-DATEI	55
5.3	ZWEITE EBENE: BEARBEITUNG EINER BESTEHENDEN IDS-DATEI	60
5.4	DRITTE EBENE: PRÜFUNG VON IDS-BASIERTEN INFORMATIONSANFORDERUNGEN	61
5.5	VIERTE EBENE: ERSTELLUNG DES BERICHTS DER ETABLIERTEN PRÜFUNG VON IDS-BASIERTEN INFORMATIONSANFORDERUNGEN	70
6	FAZIT	71
6.1	GRENZEN DER IMPLEMENTIERUNG	72
6.2	AUSBLICK	72
	QUELLEN	74
	ANHANG	79
	IDS-basierter IFC-Validierungsbericht	79

Verzeichnis der Abbildungen

ABBILDUNG 1. UNTERSCHIEDUNG VON BIM-ANWENDUNGSFORMEN: PUBLIZIERTE (OPEN) VS. PROPRIETÄRE FORMATE (CLOSED) UND (BIG) INTEGRALE VS. ISOLIERTE (LITTLE) LÖSUNG [3]	12
ABBILDUNG 2. DIREKTE ZUWEISUNG EINER MERKMALSGRUPPE ZU EINER ENTITÄTS-INSTANZ IN IFC 4.....	21
ABBILDUNG 3. DIREKTE ZUWEISUNG EINER MENGEN-GRUPPE ZU EINER ENTITÄTS-INSTANZ IN IFC 4	21
ABBILDUNG 4. ZUWEISUNG EINER MERKMALSGRUPPE ZU EINER ENTITÄTS-INSTANZ ÜBER OBJECTTYPE IN IFC 4..	21
ABBILDUNG 5. ZUWEISUNG EINER MENGEN-GRUPPE ZU EINER ENTITÄTS-INSTANZ ÜBER OBJEKTTYPE IN IFC 4	21
ABBILDUNG 6. BEISPIEL DER KLASSIFIKATION EINER INNENWAND NACH "UNICLASS" IN IFC 4	23
ABBILDUNG 7. ZUWEISUNG EINES EINZELNEN MATERIAL ZU EINER ENTITÄT IN IFC 4	24
ABBILDUNG 8. ZUWEISUNG GESCHICHTETER MATERIALIEN ZU EINER ENTITÄTS-INSTANZ IN IFC 4	24
ABBILDUNG 9. ZUWEISUNG PROFILIERTER MATERIALIEN ZU EINER ENTITÄTS-INSTANZ IN IFC	25
ABBILDUNG 10. ZUWEISUNG ZUSAMMENGESETZTER MATERIALIEN ZU EINER ENTITÄTS-INSTANZ IN IFC 4.....	25
ABBILDUNG 11. ATTRIBUTEN EINER KLASSE "IFCCOLUMN" NACH IFC 4 ADDENDUM 2	26
ABBILDUNG 12. AGGREGATION VON GESCHOSSEN IN EINEM GEBÄUDE MIT IFCRELAGGREGATES IN IFC4	27
ABBILDUNG 13. ZUWEISUNG MEHRERER ENTITÄTS-INSTANZEN ZU EINER GRUPPE IN IFC 4.....	27
ABBILDUNG 14. BEISPIEL EINER RÄUMLICHEN ZUORDNUNG IN IFC4	28
ABBILDUNG 15. ERSTELLUNG EINER VERBINDUNG VON PHYSISCHEN OBJEKTEN ANHAND "IFCRELNESTS" IN IFC4	28
ABBILDUNG 16. ANWENDUNG VON „IFCRELVOIDSELEMENT“ UND „IFCRELFFILLSELEMENT“ ZUR INTEGRATION EINES FENSTERS IN EINER WAND IN IFC4.....	28
ABBILDUNG 17. IDS: DIE NEUE ERGÄNZUNG IM OPENBIM-ÖKOSYSTEM.....	29
ABBILDUNG 18. ZWEI MÖGLICHE BSDD-WORKFLOWS FÜR EINE EFFIZIENTE NUTZUNG DES IFC-SCHEMAS.....	31
ABBILDUNG 19. BEISPIEL FÜR DIE DEFINITION EINES MERKMALS IN BSDD: „COLOUR TEMPERATURE“	31
ABBILDUNG 20. NUTZUNG DER IN DER BSDD-DATENBANK GESPEICHERTEN INFORMATIONEN BEI DER ERSTELLUNG EINER IDS.....	32
ABBILDUNG 21. BEISPIEL FÜR DIE DEFINITION EINER KLASSIFIKATION IN BSDD: „INTERNAL COMBUSTION ENGINES“	32
ABBILDUNG 22. MÖGLICHER WORKFLOW FÜR DIE NUTZUNG VON BSDD, IDS, IFC UND BCF GEMEINSAM	33
ABBILDUNG 23. KONZEPTIONELLER WORKFLOW FÜR DIE ENTWICKLUNG VON IFC-KONFORMEN SIMULATIONSMODELLEN [38]	35
ABBILDUNG 24. MÖGLICHE SYNTHAX FÜR DIE NUTZUNG DES ATTRIBUTS „XLINK“	36
ABBILDUNG 25. WEBSEITENGESTALTUNG VOM BIMCOLLAB-TOOL FÜR DIE ERSTELLUNG EINER IDS-DATEI.....	40
ABBILDUNG 26. AUSFÜHRUNG EINER IDS-BASIERTEN IFC VALIDIERUNG IN BIMCOLLAB ZOOM	41
ABBILDUNG 27. ERSTELLUNG EINER NEUEN IDS-DATEI MIT USBIM.IDSEEDITOR	41
ABBILDUNG 28. AUSFÜHRUNG EINER IDS-BASIERTEN IFC VALIDIERUNG ANHAND USBIM.BROWSER	42
ABBILDUNG 29. POSITIONIERUNG DES FELDES DER ZU VERWENDENDEN SCHALTFLÄCHEN IN BLENDERBIM ZUR VALIDIERUNG EINER IFC-DATEI MIT EINER IDS-DATEI	42

ABBILDUNG 30. AUSFÜHRUNG EINER IDS-BASIERTEN VALIDIERUNG EINER IDS DATEI IN SOLIBRI OFFICE	43
ABBILDUNG 31. ERSTELLUNG EINER NEUEN IDS DATEI MIT „XBIM FLEX“	44
ABBILDUNG 32. ERSTELLUNG EINER NEUEN IDS IN PLANNERLY	45
ABBILDUNG 33. ERSTELLUNG EINER NEUEN IDS IN BIMQ.....	45
ABBILDUNG 34. MIT TOMCZAKS IDS-EDITOR ERSTELLTER CODEAUSZUG	46
ABBILDUNG 35. INTERAKTIONEN IN DER ANGULAR-ANWENDUNG: ANSICHTS- UND ARCHITEKTURKOMPONENTEN	49
ABBILDUNG 36. GESTALTUNG DES IDS-MANAGER-SEITENDESIGNS	50
ABBILDUNG 37. GESTALTUNG DES HAUPT-SEITENDESIGNS.....	51
ABBILDUNG 38. DARSTELLUNG DER VORSEITENLADE-SEITE DER ANGULAR-ANWENDUNG	54
ABBILDUNG 39. DARSTELLUNG DER STARTSEITE DER ANGULAR-ANWENDUNG	54
ABBILDUNG 40. ZUGRIFF AUF DEN IDS-MANAGER.....	55
ABBILDUNG 41. AUSWAHLMENÜ ZUR ERSTELLUNG EINES IDS.....	55
ABBILDUNG 42. FORMULAR ZUR EINGABE VON METADATEN-INFORMATIONEN EINES IDS	57
ABBILDUNG 43. STANDORT DER '+'-SCHALTFLÄCHE IM IDS-MANAGER.....	57
ABBILDUNG 44. UNTERKLASSEN VON "IFCWALL" NACH IFC4	58
ABBILDUNG 45. GESTALTUNG EINES FELDES MIT EINSCHRÄNKUNGSMÖGLICHKEIT.....	58
ABBILDUNG 46. AUSWAHL FÜR HINZUZUFÜGENDE EINSCHRÄNKUNGEN	58
ABBILDUNG 47. MÖGLICHE ENTITÄTSNAMEN FÜR „WAND“-ELEMENTE PER AUZÄHLUNG (ENUMERATION).....	58
ABBILDUNG 48. MERKMALSFACETTE FÜR DAS ATTRIBUT „FEUERWIDERSTANDSKLASSE F“ BEI WAND-BAUTEILEN	59
ABBILDUNG 49. ECHTZEITANSICHT DER SPEZIFIKATIONSFUNKTIONALITÄT	59
ABBILDUNG 50. DEFINITION DER MERKMALSFACETTE FÜR DAS ATTRIBUT „KOSTENGRUPPE DIN276“ SPEZIFISCH FÜR TÜRELEMENTE	60
ABBILDUNG 51. ÖFFNEN EINER IDS-DATEI IN DER IDS-PALETTE	62
ABBILDUNG 52. FENSTER ZUM ÖFFNEN DER DATEI RACBASICSAMPLEPROJECT.IFC INNERHALB DER ANGULAR- ANWENDUNG.....	62
ABBILDUNG 53. ERGEBNIS DER DURCHGEFÜHRTEN IDS-BASIERTEN VALIDIERUNG.....	62
ABBILDUNG 54. ANZEIGE DER LISTE DER ANFORDERUNGEN EINER SPEZIFIKATION IN DER IDS-PALETTE.....	64
ABBILDUNG 55. DARSTELLUNG DER ERGEBNISSE NACH EXPRESS-IDS IN EINER TABELLE.....	64
ABBILDUNG 56. DEFINITION DER MERKMALSGRUPPENVORLAGE FÜR TÜREN.....	66
ABBILDUNG 57. ANZEIGE DER DATEN DER ZU VERWENDENDEN MERKMALSGRUPPENVORLAGE	66
ABBILDUNG 58. ERFASSUNG DER MERKMALSWERTE DER „TÜREN FAMILIE 1“	67
ABBILDUNG 59. ANZEIGE DER MIT DER FAMILIE „TÜREN FAMILIE 1" VERKNÜPFTEN MERKMALEN	68
ABBILDUNG 60. HINZUFÜGEN DER MERKMALSGRUPPE ZUR FAMILIE „TÜREN FAMILIE 1“	68
ABBILDUNG 61. FENSTER ZUM HINZUFÜGEN VON ELEMENTEN ZUR FAMILIE	68
ABBILDUNG 62. ANZEIGE DER ERGEBNISSE NACH ÄNDERUNGEN IN SPEZIFIKATION 2	69
ABBILDUNG 63. ERGEBNISSE IN DER IDS-PALETTE NACH EINFÜHRUNG DER VERBLEIBENDEN FAMILIEN	69
ABBILDUNG 64. ERGEBNISSE IN SPEZIFIKATION 2 NACH EINFÜHRUNG DER VERBLEIBENDEN FAMILIEN	69
ABBILDUNG 65. POSITION DER SCHALTFLÄCHE ZUM HERUNTERLADEN DES PDF-BERICHTS	70

Verzeichnis der Tabellen

TABELLE 1. IDS-SPEZIFIKATIONEN: METADATEN UND IHRE BESCHREIBUNGEN	17
TABELLE 2. DEFINITION DER OPTIONALITÄT EINER IDS-SPEZIFIKATION	18
TABELLE 3. PARAMETER DER ENTITÄTS-FACETTE: EIGENSCHAFTEN UND BEISPIELE	20
TABELLE 4. PARAMETER DER MERKMALS-FACETTE: EIGENSCHAFTEN UND BEISPIELE	22
TABELLE 5. PARAMETER DER KLASSIFIKATION-FACETTE: EIGENSCHAFTEN UND BEISPIELE	23
TABELLE 6. PARAMETER DER MATERIAL-FACETTE: EIGENSCHAFTEN UND BEISPIELE	25
TABELLE 7. PARAMETER DER ATTRIBUT-FACETTE: EIGENSCHAFTEN UND BEISPIELE	26
TABELLE 8. PARAMETER DER TEIL-VON-FACETTE (PARTOF FACET): EIGENSCHAFTEN UND BEISPIELE	29
TABELLE 9. OBJEKTVORLAGEN ANFORDERUNGEN VON BIM DEUTSCHLAND (FÜR „IFCWALL“ UND „IFCDOOR“)	56
TABELLE 10. BEDEUTUNG DER FÜR DIE VALIDIERUNG BENUTZTE SYMBOLE	63
TABELLE 11. BEDEUTUNG DER ANZAHL DER AUSGEWÄHLTE ELEMENTE HINTERGRUNDFARBE	63
TABELLE 12. ANZEIGE DER WERTE DER FÜR DIESES BEISPIEL VERWENDETEN ENTITÄTSFAMILIEN	65

Zusammenfassung

In dieser Masterarbeit wird die Entwicklung und Prüfung von IDS-basierten Informationsanforderungen in Open BIM Planungsworkflows und deren Integration in eine Web-Applikation untersucht. Der Schwerpunkt liegt auf der Erstellung eines direkten Zusammenhangs zwischen maschinenlesbaren Anforderungen und der IFC-basierten Definition von Typen, Merkmalen und Eigenschaftsvorlagen. Ziel ist es, eine effiziente und präzise Methodik für anforderungskonforme Modellierungen im BIM-Umfeld zu ermöglichen.

Die Arbeit gliedert sich in sechs Hauptkapitel, die beginnend mit einer Einführung in die Grundlagen von BIM, insbesondere die Unterscheidung zwischen Open und Closed BIM, bis hin zur detaillierten Betrachtung der IDS-Modellierung und -Validierung reichen. Ein zentrales Element bildet die umfangreiche Literaturanalyse, die den aktuellen Stand der Forschung im Bereich BIM und IDS beleuchtet.

Das Kernstück der Arbeit ist die Konzeption und Implementierung der Web-Applikation. Hierbei wird die Architektur der Applikation und die einzelnen Implementierungsphasen, von der Erstellung und Bearbeitung von IDS-Dateien bis zur Prüfung von IDS-Anforderungen und Berichterstellung, ausführlich dargestellt. Anhand praktischer Beispiel-Workflows wird die Anwendung und der Nutzen der entwickelten Applikation in realen Szenarien demonstriert.

Die Arbeit schließt mit einer Zusammenfassung und einem Ausblick. Die Ergebnisse unterstreichen die Bedeutung der Web-Applikation für die Steigerung der Effizienz und Transparenz in Planungs- und Genehmigungsprozessen in der Bauindustrie. Die Implementierung zeigt das Potenzial von IDS in Open BIM-Umgebungen und bietet wertvolle Ansätze für die weitere Forschung und Entwicklung im Bereich der digitalen Bauanträge und der Optimierung von BIM-Strategien. Die Arbeit leistet somit einen signifikanten Beitrag zur Digitalisierung der Baubranche und adressiert sowohl technische als auch organisatorische Herausforderungen, die mit der Implementierung von BIM-Strategien verbunden sind.

Abstract

This master's thesis investigates the development and validation of IDS-based information requirements in Open BIM planning workflows and their integration into a web application. The focus is on establishing a direct correlation between machine-readable requirements and the IFC-based definition of types, features, and property templates. The objective is to provide an efficient and precise methodology for compliance-driven modeling in the BIM environment. The thesis is structured into six main chapters, starting with an introduction to the fundamentals of BIM, particularly distinguishing between Open and Closed BIM, and culminating in a detailed examination of IDS modeling and validation. A key element is the comprehensive literature analysis that illuminates the current state of research in the BIM and IDS domains.

The centerpiece of the thesis is the conception and implementation of the web application. It elaborately presents the architecture of the application and the various implementation phases, from the creation and editing of IDS files to the verification of IDS requirements and report generation. Practical example workflows are used to demonstrate the application and benefits of the developed application in real scenarios.

The thesis concludes with a summary and outlook. The results highlight the significance of the web application in enhancing efficiency and transparency in planning and approval processes in the construction industry. The implementation showcases the potential of IDS in Open BIM environments and offers valuable approaches for further research and development in the area of digital construction applications and the optimization of BIM strategies. Thus, the work makes a significant contribution to the digitalization of the construction industry and addresses both technical and organizational challenges associated with the implementation of BIM strategies.

Résumé

Cette thèse de master étudie le développement et la validation des exigences d'information basées sur IDS dans les workflows de planification BIM ouverts et leur intégration dans une application web. L'accent est mis sur l'établissement d'une corrélation directe entre les exigences lisibles par machine et la définition basée sur l'IFC des types, caractéristiques et modèles de propriétés. L'objectif est de fournir une méthodologie efficace et précise pour la modélisation conforme aux exigences dans l'environnement BIM.

La thèse est structurée en six chapitres principaux, commençant par une introduction aux fondamentaux du BIM, en particulier la distinction entre BIM ouvert et fermé, et aboutissant à un examen détaillé de la modélisation et de la validation IDS. Un élément clé est l'analyse littéraire complète qui éclaire l'état actuel de la recherche dans les domaines BIM et IDS.

La pièce maîtresse de la thèse est la conception et la mise en œuvre de l'application web. Elle présente de manière approfondie l'architecture de l'application et les différentes phases de mise en œuvre, de la création et l'édition des fichiers IDS à la vérification des exigences IDS et à la génération de rapports. Des workflows d'exemple pratiques sont utilisés pour démontrer l'application et les avantages de l'application développée dans des scénarios réels.

La thèse se conclut par un résumé et une perspective. Les résultats soulignent l'importance de l'application web pour améliorer l'efficacité et la transparence dans les processus de planification et d'approbation dans l'industrie de la construction. La mise en œuvre démontre le potentiel des IDS dans les environnements BIM ouverts et offre des approches précieuses pour de futures recherches et développements dans le domaine des demandes de construction numériques et l'optimisation des stratégies BIM. Ainsi, le travail apporte une contribution significative à la numérisation de l'industrie de la construction et aborde à la fois les défis techniques et organisationnels associés à la mise en œuvre des stratégies BIM.

1 Einleitung

Building Information Modeling (BIM) wird als eine kollaborative Arbeitsmethode definiert, bei der Daten, die für die Planung, Errichtung und den Betrieb eines Gebäudes relevant sind, erfasst, verwaltet und mit Hilfe von digitalen Modellen weitergegeben werden. Diese digitalen Modelle bieten, unter anderem, eine virtuelle dreidimensionale Planungsdarstellung des Projekts und können mit zusätzlichen Daten wie Bauteil-Merkmalen, zeitlichen Planungsdaten und Fristen oder Kosten verknüpft werden. BIM wird international weit verbreitet für die Realisierung von Bauprojekten genutzt und stellt traditionelle Formen der Zusammenarbeit in Planung und Bau in Frage [1].

In der modernen Baubranche zeichnet sich daher eine signifikante Verschiebung hin zur Digitalisierung und Integration von BIM-Verfahren ab. Diese Entwicklung wird durch nationale Initiativen wie BIM Deutschland und international durch buildingSMART vorangetrieben, welche sich auf die Standardisierung und Förderung von BIM-Praktiken und die Bereitstellung von Plattformen konzentrieren. Die Notwendigkeit einer effizienten und effektiven Informationsweitergabe wird durch die zunehmende Komplexität von Bauprojekten und die Vielfalt der beteiligten Fachdisziplinen unterstrichen. In diesem Kontext gewinnen open-source Web-Anwendungen und open-BIM Prozesse an Bedeutung, da sie die Möglichkeit bieten, Planungs- und Bauabläufe durch strukturierte Anforderungen und schnellere Datenverfügbarkeit zu optimieren [2].

Der wesentliche Unterschied zwischen „Open“ und „Closed“ BIM liegt in ihrem Ansatz zur Softwareinteroperabilität und zur Datenweitergabe. „Closed“ BIM beschränkt die Projektteilnehmer darauf, dieselbe Software-Suite zu verwenden, was eine nahtlosere Daten-Integration gewährleistet, jedoch die Flexibilität einschränkt. Im Gegensatz dazu verwendet Open BIM neutrale, nicht-proprietäre Dateiformate, die es den Teilnehmern erlauben, Daten unabhängig von einem bestimmten Modellierungswerkzeug oder eines bestimmten Herstellers auszutauschen. Dieser Ansatz ist flexibler und umfassender und ermöglicht es jedem Fachplaner, seine bevorzugte Software zu verwenden und bereits erstellte Daten anderer Planer zu nutzen, ohne diese erneut zu erstellen. Open BIM kann die Zusammenarbeit zwischen verschiedenen Spezialisten, die unterschiedliche Softwareprodukte verwenden, erleichtern, und ist daher für Projekte mit unterschiedlichen Teams und Tools von Vorteil, sofern die Anwendungsfälle und Anforderungen an die weiterzugebenden Daten im Vorfeld des Projektes klar und prüfbar definiert werden.

Vor diesem Hintergrund konzentriert sich die vorliegende Arbeit auf die Entwicklung eines Validierungs-Workflows für ein Web-basiertes Software Werkzeug, das auf der Basis von IFC-Modellen (Industry Foundation Classes) die Erstellung, Bearbeitung und automatisierten Prüfung von PropertySets sowie Templates ermöglicht. Der Fokus liegt dabei auf der semantischen Ebene der Informationsverarbeitung, unter Ausschluss von geometrischen und topologischen Daten des Bauwerksmodells. Ziel ist es, einen Beitrag zur Verbesserung der Validierung, Beschleunigung der Planungs- und Genehmigungsabläufe sowie zur Entwicklung hin zum digitalen Bauantrag zu leisten.

Die **Abbildung 1** [3] illustriert diesen Unterschied und ferner ergänzen die Konzepte von „Little“ und „Big“ BIM das Verständnis von Open und Closed BIM. Während Little BIM sich oft auf die begrenzte Nutzung von BIM in einzelnen Projektbereichen konzentriert und limitierte Funktionalitäten anbietet, bezieht sich Big BIM auf eine umfangreichere, integrative Nutzung, die eine höhere Ebene der Zusammenarbeit und Informationsaustausch zwischen allen Projektbeteiligten über verschiedene Phasen und Disziplinen hinweg ermöglicht [4].

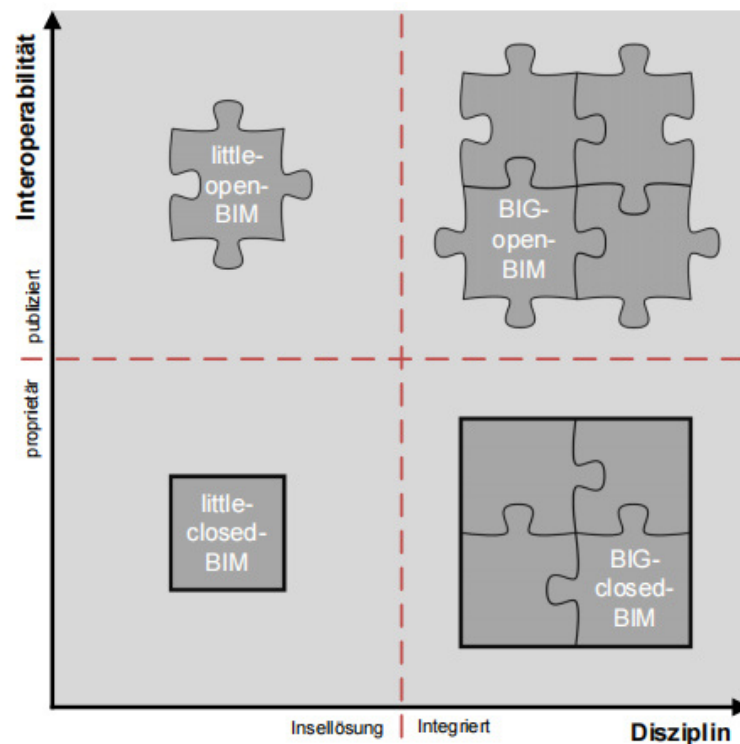


Abbildung 1. Unterscheidung von BIM-Anwendungsformen: Publierte (Open) vs. Proprietäre Formate (Closed) und (Big) Integrale vs. Isolierte (Little) Lösung [3]

1.1 Problematik

In der heutigen Architektur-, Ingenieurs- und Bauindustrie (AEC) präsentiert sich Building Information Modeling (BIM), besonders in Open BIM-Umgebungen, als eine Herausforderung durch seine Transformation traditioneller Arbeitsweisen. Individuelle Informationsanforderungen, die von Projekt zu Projekt variieren, kollidieren oft mit etablierten Betriebsprozessen, was zu Ineffizienzen führt [5]. Hinzu kommt eine wachsende Flut globaler BIM-Richtlinien, Standards und Leitfäden spezifischer Fachbereiche, die einerseits wertvolle Orientierung bieten, andererseits jedoch zu inkonsistenter Terminologie und Methodik führen. Dies erschwert sowohl die Entwicklung als auch die Anwendung dieser Dokumente [6]. Angesichts dieser weltweiten Diversität wird die Integration von adaptiven Validierung von Datenmodellen wie der „Information Delivery Specification“ (IDS) in Open BIM-Workflows unerlässlich. Diese Modelle müssen nicht nur standardisierte Anforderungen erfüllen, sondern auch flexibel die spezifischen Bedürfnisse einzelner Projekte berücksichtigen können. Sie sind entscheidend, um die dynamischen Informationsbedarfe effizient zu managen und die Diskrepanz zwischen traditionellen Methoden und den Anforderungen moderner Bauprojekte zu überbrücken. In diesem Kontext offenbart sich ein erhebliches Potenzial für Innovationen im Informationsaustauschmanagement des Bausektors, so dass digitale Projektabwicklungen nicht zu Mehraufwand sondern einer echten Effizienzsteigerung führen kann.

1.2 Ziel der Arbeit

Das Hauptziel dieser Masterarbeit ist es, einen neuartigen Ansatz zur Modellierung und Prüfung von IDS-basierten Informationsanforderungen in Open BIM Planungsworkflows zu entwickeln und zu evaluieren. Im Gegensatz zu existierenden Arbeiten, die sich häufig primär auf die technischen Aspekte von BIM oder die Implementierung von Closed BIM-Strategien konzentrieren, fokussiert diese Arbeit auf die Integration und Validierung semantischer Informationsanforderungen innerhalb offener BIM-Umgebungen in Kombination von Eigenschaftssätzen und deren Vorlagen. Dies impliziert die Entwicklung eines effizienten Validierungs-Workflows, der speziell auf die Bedürfnisse der semantischen Datenverarbeitung zugeschnitten ist, ohne dabei geometrische oder topologische Daten des Bauwerksmodells zu berücksichtigen, die Ergebnisse aber nachvollziehbar für den Nutzer mit einer dreidimensionalen Darstellung verbindet.

Ein weiteres Ziel ist es, durch die Anwendung von IDS-Standards die Lücke zwischen den vielfältigen und oft inkompatiblen Informationsanforderungen verschiedener Projektpartner zu überbrücken. Die Arbeit strebt danach, eine Brücke zwischen der theoretischen Modellierung

und der praktischen Anwendung von IDS in realen Bauprojekten zu schlagen. Dies umfasst die Entwicklung eines Online-Tools, das auf Basis eines IFC-Modells die Erstellung und Bearbeitung von „PropertySets“ und Templates ermöglicht, wobei der Schwerpunkt auf der semantischen Präzision und der Kompatibilität mit verschiedenen BIM-Softwarelösungen liegt. Anforderungen sollen mittels IDS Dateien beschrieben und automatisiert prüfbar gemacht werden, ohne hierzu zusätzliche Software zu benötigen. „PropertyTemplates“ und „Typen“ von Objekten können, ausgerichtet an den jeweiligen Projektanforderungen im Browser erstellt und modelliert werden, ohne dass ein Modellierungstool benötigt wird

Darüber hinaus zielt die Arbeit darauf ab, durch die Implementierung von maschinenlesbaren und standardisierten Anforderungen, wie sie in AIA und LOIN definiert sind, die Planungs- und Genehmigungsprozesse effizienter zu gestalten. Dies soll zu einer Beschleunigung der Validierungsprozesse und der Realisierung des digitalen Bauantrags beitragen. Letztendlich wird angestrebt, einen Beitrag zur Diskussion über die Vorteile von Open BIM gegenüber Closed BIM zu leisten, welches essentiell für öffentliche Bauprojekte ist, und die Potenziale von webbasierten Anwendungen in der Baubranche aufzuzeigen.

1.3 Struktur der Arbeit

Die vorliegende Masterarbeit ist grundlegend in sechs Hauptkapitel gegliedert, die zusammen eine systematische Untersuchung des Konzepts, der Entwicklung und Analyse eines Validierungswerkzeugs für IDS-basierte Anforderungen in Open BIM Workflows darstellen. Jedes Kapitel ist darauf ausgerichtet, spezifische Aspekte der Thematik zu beleuchten mit dem Ziel, Stück für Stück zum umfassenden Verständnis des Gesamtprojekts und der web-basierten Validierung und Bearbeitung beizutragen. Im ersten Kapitel wurde dazu das Projektziel definiert.

IDS-Modellierung und Validierung - Grundlagen: Das zweite Kapitel legt den Grundstein für das Verständnis der IDS-Modellierung und ihrer Validierung. Es bietet eine detaillierte Betrachtung der grundlegenden Aspekte des IDS-Formats, seiner Spezifikationen und der technischen Voraussetzungen. Hier wird die Komplexität und die Feinheiten von IDS eingehend analysiert, wobei auch die Synergien zwischen IDS und anderen Bestandteilen des Open BIM-Ökosystems von buildingSMART, insbesondere des buildingSMART Data Dictionaries (bSDD) und des BIM Collaboration Formats (BCF) sowie Property Templates, beleuchtet werden. Dieses Kapitel ist entscheidend, um die theoretische Basis für die nachfolgenden praktischen Anwendungen zu legen.

Verwandte Arbeiten: Das dritte Kapitel widmet sich einer umfassenden Literaturanalyse und diskutiert relevante Forschungsarbeiten und Implementierungen im Bereich BIM und IDS. Durch den Vergleich und die kritische Analyse bestehender Ansätze wird ein Kontext für die eigene Forschungsarbeit geschaffen und bestehende Forschungslücken werden identifiziert.

Konzeption und Implementierung: Im vierten Kapitel erfolgt der Übergang von der Theorie zur Praxis. Beginnend mit der Definition der Anforderungen für das zu entwickelnde Werkzeug, wird dessen Architektur umfassend dargestellt. Die Implementierung wird in vier Phasen unterteilt: 1) Erstellung und 2) Bearbeitung von IDS-Dateien, 3) Prüfung von IDS-Anforderungen und 4) Berichterstellung. Dieses ist das zentrale Kapitel der Arbeit und illustriert die konkrete Umsetzung der theoretischen Grundlagen in praktische Anwendungen.

Beispiel-Workflows: Im fünften Kapitel werden verschiedene Anwendungsbeispiele und Workflows präsentiert. Diese reichen von der Installation und dem Start der Anwendung bis hin zur Durchführung der verschiedenen Erstellungs- und Validierungsebenen. Ziel ist es, die Anwendung des entwickelten Werkzeugs in realen Szenarien zu demonstrieren.

Fazit: Das abschließende sechste Kapitel fasst die Ergebnisse der Arbeit zusammen und reflektiert über die erzielten Fortschritte und Erkenntnisse. Es werden auch die Grenzen und Einschränkungen der Implementierung diskutiert. Zudem werden zukünftige Entwicklungsmöglichkeiten und Forschungsansätze in diesem Bereich aufgezeigt, um den Weg für weitere Untersuchungen und Innovationen der automatisierten Modellierung und Prüfung von Informationsanforderungen bis hin zum digitalen Bauantrag zu ebnen.

2 IDS-Modellierung und Validierung: Grundlagen

In diesem Kapitel erfolgt eine eingehende Analyse des IDS-Formats, einschließlich seiner Spezifikationen und aller relevanten Aspekte, die für ein profundes Verständnis des Konzepts unerlässlich sind. Es ist dabei wesentlich zu berücksichtigen, dass das IDS-Format, als ein relativ neues Konzept, sich noch in der Entwicklungsphase befindet und zum aktuellen Zeitpunkt noch nicht offiziell freigegeben ist [7]. Die Untersuchung konzentriert sich somit auf die derzeitige Version des IDS-Formats, um ein umfassendes Bild seines aktuellen Entwicklungsstandes zu erlangen. Dieses vertiefte Verständnis ist entscheidend, um eine fundierte Basis für die korrekte Implementierung in den nachfolgenden Schritten der Arbeit zu schaffen.

2.1 IDS-Format

IDS (Abkürzung für „Information Delivery Specification“) ist ein Standard von buildingSMART zur Spezifikation und Überprüfung einfacher Informationsanforderungen in IFC-Modellen [8]. Eine IDS Datei ist eine XML-Datei [7], die so konzipiert, dass es sowohl von Menschen als auch von Computern gelesen werden kann [9], und deren Schema in XSD definiert ist [7]. Eine vorhandene IDS-Datei könnte geöffnet und deren Inhalt untersucht werden, um ein Verständnis dafür zu bekommen, wie eine IDS strukturiert ist [7].

Laut buildingSMART, IDS erleichtert die Überprüfung von IFC für Kunden, Modellierer und Software-Tools, die (automatisierte) Analysen durchführen. Als wesentlicher Bestandteil kann es als vertragliches Mittel dienen, um die Bereitstellung genauer Informationen zu gewährleisten, vergleichbar mit einem „Modell als rechtliches Dokument“. Darüber hinaus verfügt es über die Fähigkeit, auf spezifische lokale Kontexte und Anwendungsfälle zugeschnittene Anforderungen zu generieren [10].

2.1.1 IDS-Metadaten

Jede IDS-Datei verfügt über grundlegende Metadaten, die für alle im IDS aufgelisteten Spezifikationen gemeinsam sind. Dazu gehören der „Titel des Dokuments“ als Referenz für das gesamte IDS, das „Urheberrecht des IDS-Inhabers“ (English: Copyright owner), die „Version“ des IDS zur Nachverfolgung von Änderungen, der „Autor des IDS“, das „Veröffentlichungsdatum“ und eine „detaillierte Beschreibung“, die erläutert, für wen das IDS bestimmt ist, warum es erstellt wurde und auf welche Projekte es anwendbar ist. Des Weiteren umfasst es eine kurze Zusammenfassung des „Zwecks des IDS“ und zu welchem „Projektmeilenstein“ das IDS erfüllt sein sollte.

2.1.2 Spezifikationen

Spezifikationen sind so in einer IDS-Datei strukturiert, dass sie von Computerprogrammen automatisch und eindeutig auf die Erfüllung von Informationsbedürfnissen überprüft werden können. Sie gliedern sich in drei Hauptbereiche [11]:

a) Spezifikations-Metadaten

Jede Spezifikation verfügt über Metadaten, die dabei helfen, die Ziele zu beschreiben und Anleitungen zur Erreichung dieser Ziele zu geben. Hier könnte der Benutzer erklären, warum die Spezifikation für Ihr Projekt von Bedeutung ist und wie sie umgesetzt werden kann. Dieser Abschnitt ist so gestaltet, dass Personen die Hintergründe und Gründe für die Informationsanforderung verstehen können [9,12]:

Metadata	Beschreibung
Name	Eine Kurzbezeichnung der spezifizierten Informationen
Beschreibung (Description)	Dabei wird beschrieben, warum die Anforderung für das Projekt wichtig ist. Die Person, die die Beschreibung liest, sollte verstehen, warum die Informationen einen Wert liefern, welche Arbeitsabläufe dadurch unterstützt werden, und welche Projektvorteile verloren gehen würden, wenn die Anforderungen nicht erfüllt werden.
Anweisungen (Instructions)	Geben Anweisungen darüber, wer für die Bereitstellung der Informationen verantwortlich ist, und Einzelheiten darüber, wie dies erreicht wird, wie zum Beispiel Richtlinien darüber, wie eine Namenskonvention funktioniert, wie man einen geeigneten Wert auswählt oder was in Ausnahmefällen zu tun ist.
Spezifikations- Identifikator (Identifier)	Der Autor der IDS kann der Spezifikation einen Identifikator zuweisen. Dies soll ein maschinenlesbarer Identifikator sein.

Tabelle 1. IDS-Spezifikationen: Metadaten und ihre Beschreibungen

b) Anwendbarkeit (English: Applicability):

Dieser Teil definiert, für welche Arten von Objekten die Spezifikation gilt. In IFC-Modellen existieren unterschiedlichste Objekttypen wie Wände, Türen und Fenster oder sogar die sogenannte „ObjectTypes“. Jedoch richtet sich jede Spezifikation nur an bestimmte Objektarten [11].

c) Anforderungen (English: Requirements):

In diesem Abschnitt wird festgelegt, welche spezifischen Informationen für die im vorherigen Teilgenannten Objekte (Anwendbarkeit) notwendig sind, beispielsweise bestimmte Merkmalen oder Klassifikationen [11].

Spezifikationen und Anforderungen können eine „Optionalität“ aufweisen, die als „Erforderlich“, „Optional“ oder „Verboten“ festgelegt wird. Dies wird mit den Attributen „minOccurs“ und „maxOccurs“ von XSD dargestellt. Falls keine Werte für „minOccurs“ und „maxOccurs“ vorgegeben wurden, wird die Optionalität standardmäßig als „Erforderlich“ angenommen. Die Zustände werden wie folgt repräsentiert:

Optionalität	minOccurs	maxOccurs
Erforderlich	1	unbounded
Optional	0	unbounded
Verboten	0	0

Tabelle 2. Definition der Optionalität einer IDS-Spezifikation

Andere Konfigurationen von „minOccurs“ und „maxOccurs“ sind laut buildingSMART derzeit nicht zulässig [7].

In diesem Zusammenhang ist es besonders hervorzuheben, dass Spezifikationen, die als „Verboten“ klassifiziert sind, gemäß den aktuellen festgelegten Standards keine Anforderungen enthalten dürfen [13].

2.1.3 Facettenparametertyp: Einfacher Wert vs. Einschränkung

Ein Facetten-Parameter wird entweder durch einen einfachen Wert (*English: Simple value*) oder eine komplexe Einschränkung (*English: Restriction*) definiert. Ein einfacher Wert könnte Text, Zahl oder ein Boolescher Wert (TRUE / FALSE) sein. Komplexe Einschränkungen könnten im Gegensatz dazu eine Auswahlliste gültiger Werte, Benennungsschemata und -konventionen, numerische Bereiche und andere beinhalten.

Eine Einschränkung kann auf vier verschiedene Weisen spezifiziert werden: Aufzählung (*English: Enumeration*), Muster (*English: Pattern*), Grenzen und Länge;

- Aufzählung:

Eine Aufzählung-Einschränkung bedeutet, dass der Wert aus einer Liste erlaubter Zahlen- oder Text basieren Werte stammen muss.

Boolesche Werte wie „TRUE“ oder „FALSE“ können ebenfalls festgelegt werden. Allerdings ist es meist nicht sinnvoll, boolesche Werte als Aufzählung zu definieren, da sie naturgemäß nur zwei Optionen bieten.

- Muster (Pattern):

In der Anwendung von Namenskonventionen wird eine Muster-Einschränkung verwendet, um Kennzeichnungen zu standardisieren, wie beispielsweise das Benennen von Tür-Entitäten als "TT01", "TT02" usw. Diese kann durch die Verwendung von Regulären Ausdrücken (Regex) weiter verfeinert werden, welche spezielle Symbole zur Darstellung von Zahlen, Buchstaben oder anderen Zeichen nutzen, um eine präzisere Definition dieser Muster zu ermöglichen.

- Grenzen (bounds):

Grenzen ermöglichen die Einschränkung von Werten auf einen bestimmten Bereich von Zahlen. Dabei können sowohl eine obere als auch eine untere Grenze definiert werden. Mithilfe der mathematischen Notationen „>“, „<“, „≥“, „≤“ lässt sich ein Zahlenintervall bestimmen.

- Länge:

Es ist möglich, eine maximale, minimale oder genaue Zeichenlänge für einen Wert zu definieren. Beispielsweise wäre „Klasse“ ein gültiger Wert, wenn eine Länge von sechs Zeichen festgelegt wird.

2.1.4 Facetten

Jede Spezifikation setzt sich aus einer Reihe von Anwendbarkeits- und Anforderungsfacetten zusammen. Es ist essenziell, dass jede Anwendbarkeit einer Spezifikation mindestens eine spezifische Facette umfasst, wobei die Anzahl der Facetten nach oben hin unbegrenzt ist. Hinsichtlich der Anforderungsfacetten kann eine Spezifikation entweder keine oder eine beliebig hohe Zahl dieser Facetten beinhalten. Eine Ausnahme bilden hier die als „Verboten“ klassifizierten Spezifikationen, die, wie zuvor erwähnt (in **2.1.2 c**), keine Anforderungsfacetten enthalten dürfen.

- Entitäts-Facette (English: Entity facet)

Jede Entität in einem IFC-Modell hat einen "IFC-Klassen"-Namen. Zum Beispiel haben Wand-Entitäten die IFC-Klasse „IfcWall“ und Tür-Entitäten die IFC-Klasse „IfcDoor“. Eine IFC-Datei besteht auch aus Entitäten, die keine einzelnen Bauelemente darstellen, aber haben ebenfalls eine Klasse; wie beispielsweise hat das Projekt die Klasse „IfcProject“ und Fenstertypen die Klasse „IfcWindowType“ [14].

In dieser Facette werden nicht nur die Unterklassen von „IfcElement“ als Entitäten angenommen, sondern alle Entitäten des IFC-Schemas wie; „IfcProject“, „IfcWallType“, „IfcWindowType“, „IfcGrid“, „IfcLine“, usw...

Klassen dienen nicht nur der Kategorisierung von Entitäten. Sie geben auch an, welche Arten von Merkmalen und Beziehungen erlaubt sind. Zum Beispiel kann logischerweise eine Entität der Klasse „IfcDoor“ ein Merkmal für die Feuerfestigkeit haben, eine Entität der Klasse „IfcGrid“ jedoch nicht [14].

Einige Entitäten können optional auch einen vordefinierten Typ (durch ihre Attribute „PredefinedType“) haben. Dies ist eine weitere Ebene der Entitätskategorisierung zusätzlich zum IFC-Klassennamen. Zum Beispiel kann eine Entität „IfcFurniture“ einen vordefinierten Typ wie „CHAIR“, „TABLE“, „DESK“, „BED“, „FILECABINET“, „SHELF“, „SOFA“, „NOTDEFINED“ oder „USERDEFINED“ haben [15]. Falls der Wert des vordefinierten Typs auf „USERDEFINED“ (benutzerdefinierte) gesetzt ist, wird der Wert des Attributs „ObjectType“ berücksichtigt [16].

Die folgende Tabelle bietet eine Übersicht über die Parameter, die mittels der Entitäts-Facette definiert werden können bzw. müssen [14]:

Parameter	Erforderlich	Einschränkungen erlaubt	Beispiel
Name	✓	✓	IFCFURNITURE
PredefinedType	✗	✓	TABLE

Tabelle 3. Parameter der Entitäts-Facette: Eigenschaften und Beispiele

Typischerweise wird jede einzelne Spezifikation in ihrem Anwendungsbereich eine Entitäts-Facette verwenden [14].

Laut des XSD-Schemas dürfen Entitäts-Facetten **keine** Werte für „minOccurs“ und „maxOccurs“ erhalten (gilt auch wenn sie als Anforderungen benutzt sind).

- Merkmals-Facette (English: Property facet)

IFC bietet die Möglichkeit, benutzerdefinierte Metadaten, die an Objekte angehängt sind, anhand der relationalen Klasse „IfcRelDefinesByProperties“ [17,18] zu definieren, die als „Merkmalen“ („IfcProperty“) bezeichnet werden. Merkmalen haben einen Namen (z. B. „Festigkeitsklasse“), einen vom Benutzer bereitgestellten Wert (z. B. „8.8“) und ähnliche Merkmalen werden in „Merkmalsgruppen“ („IfcPropertySet“) gruppiert [19].

Analogisch sind „Mengen“ (Unterklassen von „IfcPhysicalQuantity“ [20]) als Art Merkmalen betrachtet und sie werden zu der Entität „IfcElementQuantity“ [17] durch seiner Vordefinierten Attribut „Quantities“ zugewiesen.

Die folgenden Abbildungen verdeutlichen, wie beides zu einer Komponente verbunden werden können:

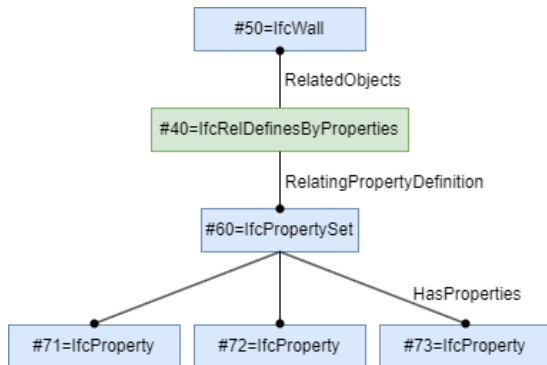


Abbildung 2. Direkte Zuweisung einer Merkmalsgruppe zu einer Entitäts-Instanz in IFC 4

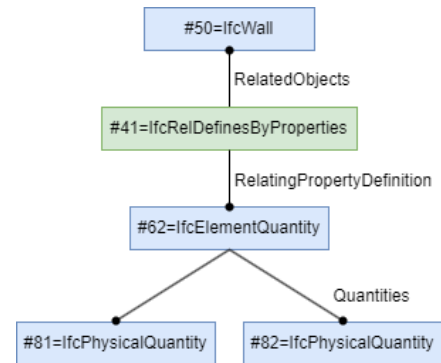


Abbildung 3. Direkte Zuweisung einer Mengen-Gruppe zu einer Entitäts-Instanz in IFC 4

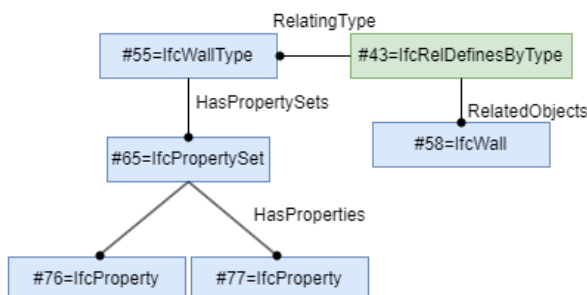


Abbildung 4. Zuweisung einer Merkmalsgruppe zu einer Entitäts-Instanz über ObjectType in IFC 4

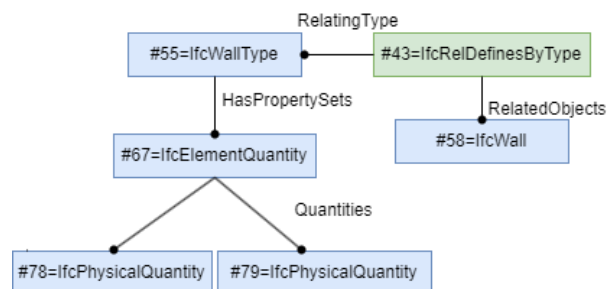


Abbildung 5. Zuweisung einer Mengen-Gruppe zu einer Entitäts-Instanz über ObjectType in IFC 4

„IfcProperty“ und „IfcPhysicalQuantity“, Abbildung 3 und 4, sind beide abstrakte Klassen die eine Menge von Konkrete einfache und komplexe Merkmalen (Unterklassen) enthalten können [18,20].

Einige Merkmalsgruppen und Merkmalen werden von buildingSMART festgelegt, um eine Standardisierung von häufig in weltweiten Projekten vorkommenden Merkmalen zu erreichen.

Beispielsweise ist das „FireRating“- Merkmal, die zur Merkmalsgruppe „Pset_WallCommon“ gehört, eine von buidingSMART definierten Merkmal. Nutzern haben zudem die Möglichkeit,

eigenen Merkmalen und Merkmalsgruppen zu erstellen, die entweder spezifisch für ihr Projekt sind oder mittels der Merkmalsgruppe-Vorlagen-Funktion in IFC verbreitet werden.

Es wird aber empfohlen, zunächst die von buildingSMART standardisierten Merkmalen zu nutzen, bevor daran denken, neuen eigenen Merkmalen zu erstellen. Alle standardisierten Merkmalen sind Teil einer Merkmalsgruppe, dessen Name mit „Pset_“ oder „Qto_“ beginnt [19]. Die Zuordnung von Merkmalen variiert je nach Art der Entität. Zum Beispiel sind Merkmalen wie die Tragfähigkeit meistens für Bauelemente wie Wände und Träger relevant, jedoch nicht für Einrichtungsgegenstände, Luftkanäle oder Kabel geeignet. Diese spezifischen Zuordnungen werden als geeignete Entitäten bezeichnet. In einer IDS ist es wichtig, die Anwendbarkeit von Merkmalen auf verschiedene Objekte zu bedenken. Merkmalen können einer breiten Palette von Objekten zugeordnet werden, nicht nur konkreten Objekten wie Türen, Fenstern und Bodenplatten, sondern auch abstrakten wie Aufgaben, Materialien, Querschnitten von Bauprofilen oder Arbeitskräften.[19]

Die nachstehende Tabelle 4 zeigt eine Aufstellung der Parameter, die durch die Merkmals-Facette festgelegt werden können bzw. müssen [19]:

Parameter	Erforderlich	Einschränkungen erlaubt	Beispiel
Property Set	✓	✓	Pset_WallCommon
Name	✓	✓	FireRating
Value	✗	✓	5400
Data Type	✓	✗	IfcTimeMeasure
URI	✗	✗	(*)

Tabelle 4. Parameter der Merkmals-Facette: Eigenschaften und Beispiele

(*) URI („Uniform Resource Identifier“) ist ein einzigartiger Identifikator gemäß der **ISO 23386** Norm, der eine präzise Referenzieren gewährleistet [19,21].

Beispiel: <https://identifier.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/FireRating>

Jenseits des Datentyps selbst, der zur Definition des Merkmals verwendet wird, legt dieser Parameter auch die Einheit fest, die mit dem betreffenden Merkmal verknüpft ist. In der Praxis verlangt buildingSMART, dass jeder numerische Wert von physikalischen Größen eines Datentyps eine internationale Einheit zugewiesen bekommt [22]. In der Dokumentation zu IDS findet sich eine Tabelle, die mögliche Datentypen und ihre zugehörigen Einheiten auflistet ¹.

¹ <https://github.com/buildingSMART/IDS/blob/master/Documentation/units.md> [zuletzt 10.01.2024]

- Klassifikationsfacette (English: Classification facet)

Ein Klassifizierungssystem strukturiert Elemente in einer festgelegten Hierarchie; „Omniclass“, „uniclass“, „Masterformat“ und „DIN27“ sind Beispiele für ein solches System [23]. Dabei werden kurze Referenzcodes verwendet, wie „EF_25_10“ für Wände und „EF_25_10_40“ für Innenwände [24] in „uniclass“, um Elemente auf verschiedenen Detailebenen einzuordnen. In IFC-Modellen kann jedes Objekt einer solchen Klassifizierung zugeordnet werden. Das Klassifizierungs-Facette unterscheidet sich vom Entitäts-Facette; Während die Entitäts-Facette sich auf vorgegebene IFC-Klassen und Typen beschränkt, die auch zur Klassifizierung dienen können, bezieht sich die Klassifizierungs-Facette auf Systeme außerhalb des IFC-Standards [23].

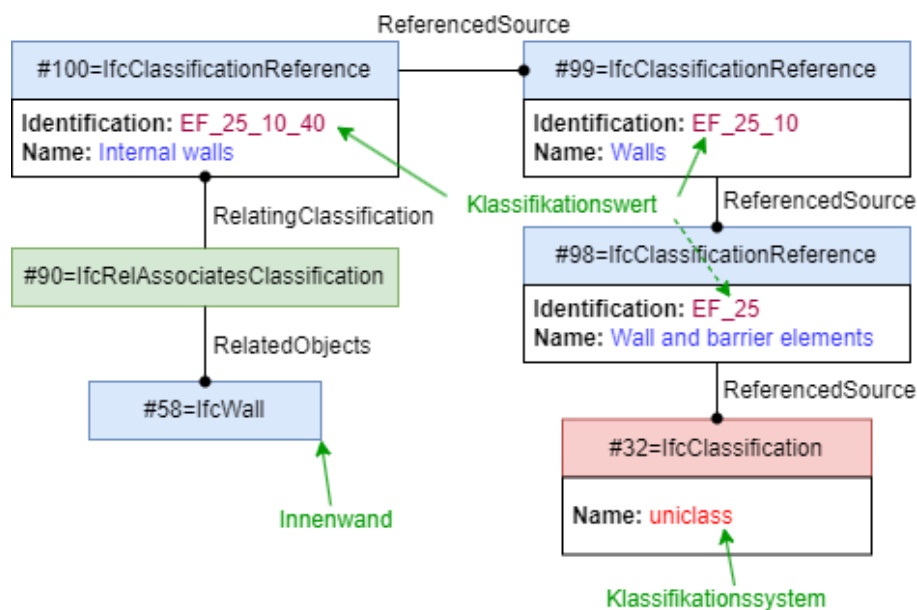


Abbildung 6. Beispiel der Klassifikation einer Innenwand nach "uniclass" in IFC 4

Die untenstehende Tabelle 5 bietet eine Übersicht über die Parameter, die durch die Klassifikation-Facette festgelegt werden können [23]:

Parameter	Erforderlich	Einschränkungen erlaubt	Beispiel
System	✗	✓	uniclass
Value	✗	✓	EF_25_10_40
URI	✗	✗	(siehe: Merkmals-Facette)

Tabelle 5. Parameter der Klassifikation-Facette: Eigenschaften und Beispiele

- Materialfacette (English: Material facet)

In IFC können Elemente wie Wände, Türen und Fenster sowie deren Typen mit bestimmten Materialien (in der Regel Baustoffe) verbunden werden, Abbildung 7. So werden beispielsweise verschiedene Betonsorten als unterschiedliche Materialien angesehen [25].

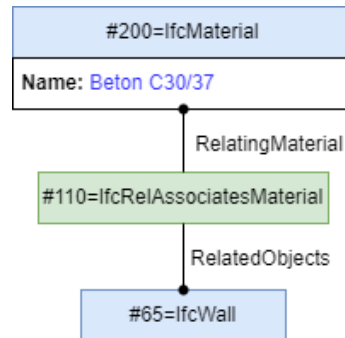


Abbildung 7. Zuweisung eines einzelnen Material zu einer Entität in IFC 4

Ein Element kann einfach aus einem Material bestehen, wie ein Stuhl, der aus Holz gefertigt ist. Mit dem Materialfacetten-Konzept können Elemente nach ihrem Material gefiltert werden. Es gibt auch Fälle, in denen ein Element mehrere Materialien aufweist, und zwar in drei Varianten unterteilt [25] :

- Bei **geschichteten Materialien** wird ein Element, wie eine Wand, durch die Kombination verschiedener Materialschichten mit jeweiliger Dicke, wie Ziegelstein und Isolation aus Glaswolle, definiert [25].

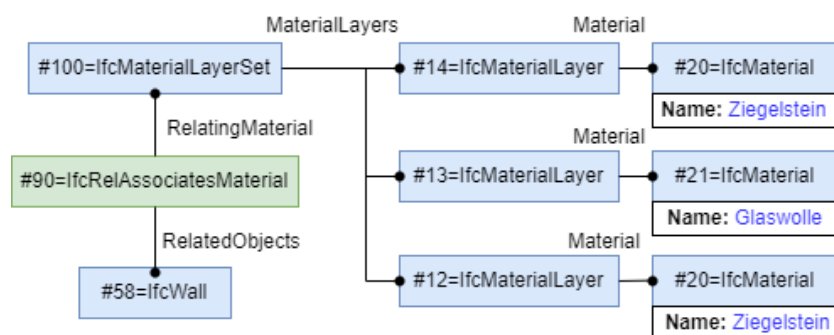


Abbildung 8. Zuweisung geschichteter Materialien zu einer Entitäts-Instanz in IFC 4

- Bei **profilierten Materialien** wird ein Element (z. B. ein Balken) anhand eines bestimmten Profils (z.B. C-, H- oder I-Profil) beschrieben, das entlang eines Weges

geformt wird. Solche Elemente können aus mehreren Materialien zusammengesetzt sein [25].

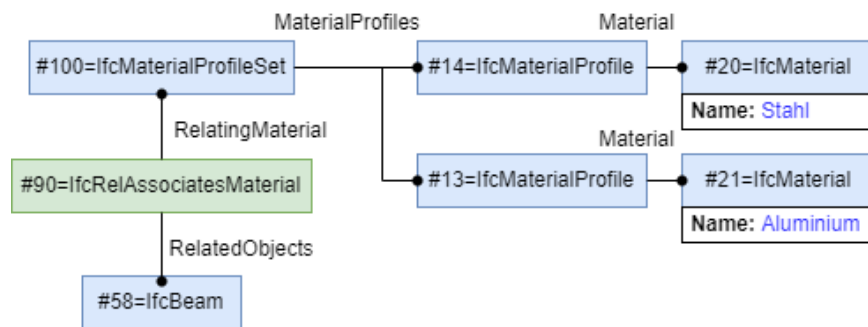


Abbildung 9. Zuweisung profilierter Materialien zu einer Entitäts-Instanz in IFC

- Bei **zusammengesetzten Materialien** besteht ein Element, beispielsweise ein Fenster oder eine Decke, aus verschiedenen Materialteilen (etwa Glas und Rahmen des Fensters) oder aus einer Mischung verschiedener Stoffe (wie bei einem Balken mit Anteilen von Eisen und Kohlenstoff, Abbildung 10) [25].

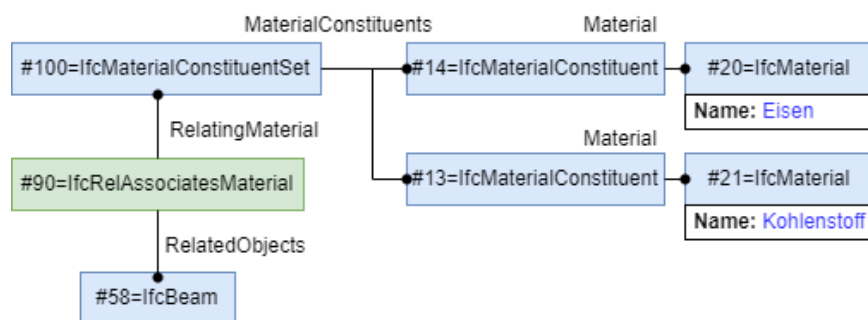


Abbildung 10. Zuweisung zusammengesetzter Materialien zu einer Entitäts-Instanz in IFC 4

Die vorliegende Tabelle 6 stellt eine Auflistung der Parameter dar, die durch die Material-Facette definiert werden können [25]:

Parameter	Erforderlich	Einschränkungen erlaubt	Beispiel
Material	✗	✓	Aluminium
URI	✗	✗	(siehe: Merkmals-Facette)

Tabelle 6. Parameter der Material-Facette: Eigenschaften und Beispiele

- Attribut-Facette (English: Attribute facet)

In IFC-Modellen besitzt jede Entität eine Reihe von festgelegten Attributen; Attribute sind grundlegende Daten, die allen IFC-Entitäten zugeordnet sind. Sie sind im IFC-Standard von buildingSMART definiert und lassen sich nicht individuell anpassen [26].

Attribute sind daher ideal, um die Anwendbarkeit auf spezielle Elemente festzulegen oder um zu bestimmen, dass bestimmte Elemente auf spezifische Weise identifiziert, benannt oder beschrieben werden sollen [26].

Um herauszufinden, welche Attribute für eine IFC-Klasse verfügbar sind und welche Werte sie annehmen können, sollte die IFC-Dokumentation konsultiert werden. Die Vorgehensweise ist ähnlich für alle IFC-Versionen [26].

Laut des XSD-Schemas dürfen Attribut-facetten **keine** Werte für „minOccurs“ und „maxOccurs“ erhalten (gilt auch wenn sie als Anforderungen benutzt sind).

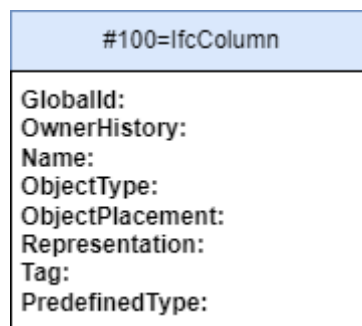


Abbildung 11. Attributen einer Klasse "IfcColumn" nach IFC 4 Addendum 2

Die folgende Tabelle zeigt eine Liste der Parameter, die in der Attributs-Facette festgelegt werden können bzw. müssen:

Parameter	Erforderlich	Einschränkungen erlaubt	Beispiel
Name	✓	✓	Name
Value	✗	✓	Tür-1-1

Tabelle 7. Parameter der Attribut-Facette: Eigenschaften und Beispiele

- Teil-von-Facette (English: PartOf facet)

Diese Facette bestimmt, ob Objekte als Teil eines anderen Objekts gelten (Anwendbarkeit) oder als solche erforderlich sind (Anforderung).

Für die Feststellung, ob eine Entität ein Teil eines anderen ist, wendet diese Facette eine **rekursive** Durchsuchung verschiedener relationale Typen an: [27]

- „IfcRelAggregates“ legt fest, wie mehrere kleinere Objekte zu einem größeren Objekt zusammengefasst werden. Beispielsweise formen mehrere Stockwerke ein Gebäude, oder Balken und Bretter bilden gemeinsam eine Platte [27].

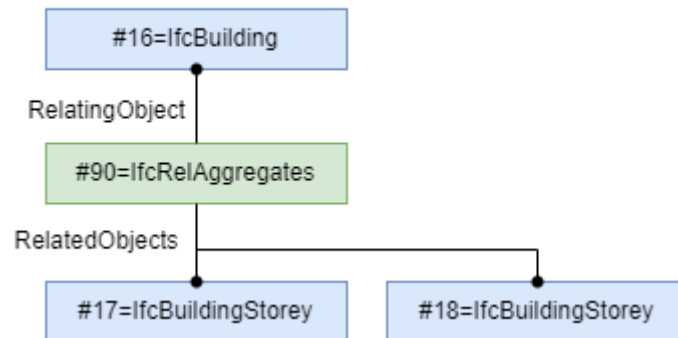


Abbildung 12. Aggregation von Geschossen in einem Gebäude mit IfcRelAggregates in IFC4

- „IfcRelAssignsToGroup“ beschreibt die Gruppierung mehrerer Objekte zu einer Sammlung für spezifische Zwecke. Beispielsweise können Lüftungskanäle und Ventilatoren als Teil eines Verteilungssystems gruppiert werden [27].

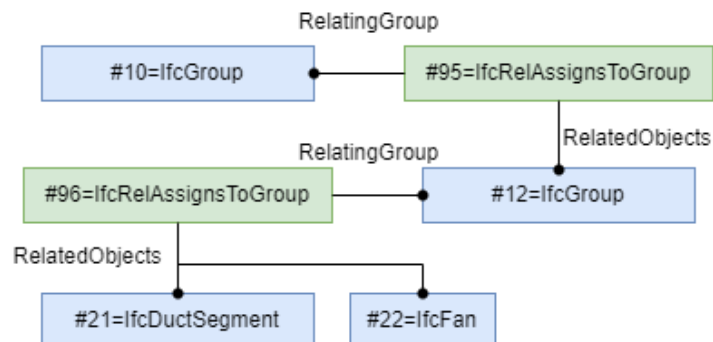


Abbildung 13. Zuweisung mehrerer Entitäts-Instanzen zu einer Gruppe in IFC 4

- „IfcRelContainedInSpatialStructure“ bestimmt die räumliche Zuordnung von Objekten [27], wie eine Pumpe in einem Raum oder eine Säule auf einer bestimmten Etage.

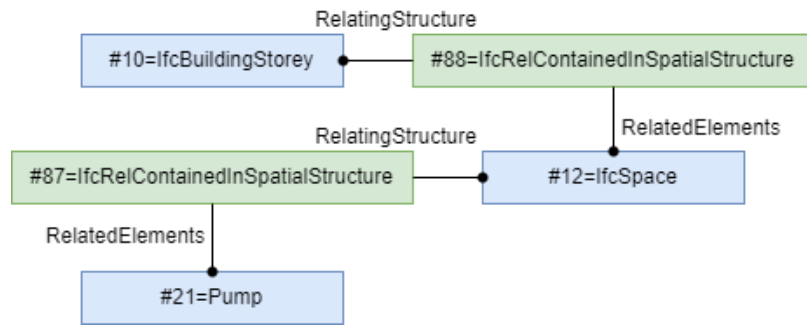


Abbildung 14. Beispiel einer räumlichen Zuordnung in IFC4

- „IfcRelNests“ beschreibt die Verbindung von physischen Objekten mit einem größeren Trägerobjekt, wobei sich die verbundenen Objekte mit dem Träger bewegen [27].

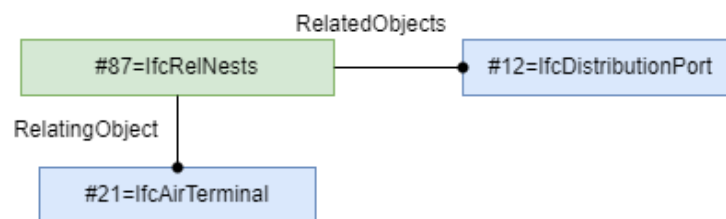


Abbildung 15. Erstellung einer Verbindung von physischen Objekten anhand "IfcRelNests" in IFC4

- „IfcRelVoidsElement“ und „IfcRelFillsElement“ definieren, wie Hohlräume zu Elementen gehören [27].

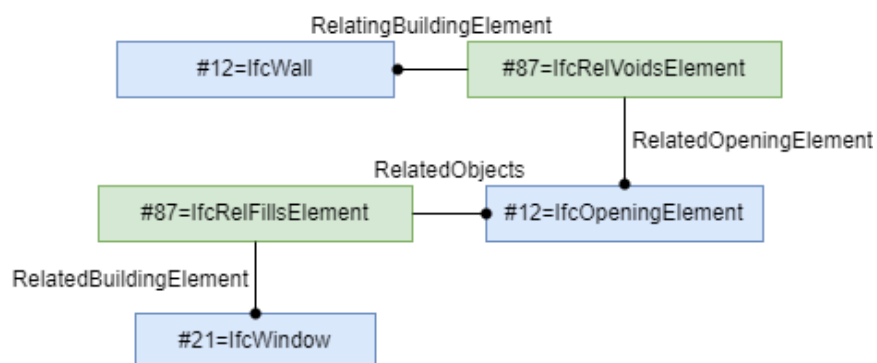


Abbildung 16. Anwendung von „IfcRelVoidsElement“ und „IfcRelFillsElement“ zur Integration eines Fensters in einer Wand in IFC4

Die folgende Tabelle bietet eine Übersicht über die Parameter, die mittels dieser Facette definiert werden können bzw. müssen [27,28]:

Parameter	Erforderlich	Einschränkungen erlaubt	Beispiel
Entity	✓	✓	IFCBUILDINGSTOREY
Relation	✗	✗	IfcRelAggregates

Tabelle 8. Parameter der Teil-von-Facette (PartOf facet): Eigenschaften und Beispiele

2.2 Objektive Analyse der Integration von IDS in das OpenBIM-Ökosystem von buildingSMART

IDS repräsentiert die neueste Ergänzung im OpenBIM-Ökosystem von buildingSMART und ist Gegenstand einer eingehenden Betrachtung in diesem Kapitel. Ziel ist es, die Beziehungen zwischen IDS und den bestehenden Komponenten des OpenBIM-Ökosystems zu untersuchen, um ein klares Bild von den möglichen Synergien und Integrationspotentialen zu erhalten. Diese Analyse soll beleuchten, wie IDS in die bestehende Struktur eingebettet sein und welche Auswirkungen es auf die derzeitigen Arbeitsabläufe und Prozesse haben könnte. Anstatt voreilige Schlussfolgerungen zu ziehen, wird eine objektive Betrachtung angestrebt, um die Rolle von IDS innerhalb des OpenBIM-Ansatzes zu verstehen und zu bewerten. Die Diskussion konzentriert sich auf die realen und potenziellen Beiträge von IDS im Kontext der aktuellen und zukünftigen Anforderungen in der Bauindustrie.

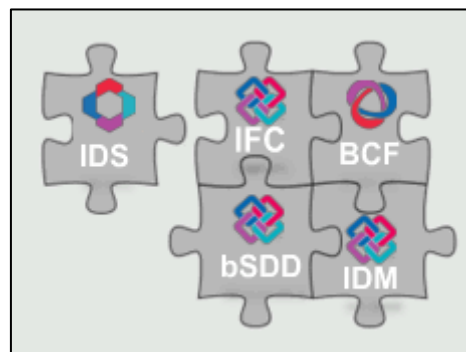


Abbildung 17. IDS: die neue Ergänzung im OpenBIM-Ökosystem

2.2.1 Information Delivery Manual (IDM):

Das „Information Delivery Manual“ (IDM) ist als **ISO 29481-1:2010** Standard anerkannt [29] und wurde von buildingSMART, zur Definition, Dokumentation und Standardisierung von Prozessen und Informationsflüssen über den Lebenszyklus von Bauprojekten, entwickelt. IDM beabsichtigt, die Kommunikation und Zusammenarbeit zwischen den verschiedenen Beteiligten eines Bauprojekts zu erleichtern, besonders im Kontext des Building Information Modeling (BIM). Durch die Klärung und Standardisierung von Informationsbedürfnissen gewährleistet IDM, dass alle Teilnehmer eines Projekts ein klares Verständnis der erforderlichen Daten in jeder Phase haben, was die Effizienz und Entscheidungsfindung optimiert [30,31].

Die IDS hingegen besteht aus einer Menge von technischeren und detaillierten Spezifikationen (2.1), die sich sehr wohl auf die in der IDM identifizierten Anforderungen stützen können. Sie beschreibt genau, wie Informationen strukturiert und formatiert werden müssen, um effizient zwischen den verschiedenen Beteiligten und Softwaresystemen in einem BIM-Projekt ausgetauscht zu werden. Die IDS konzentriert sich auf die Definition von Datenformaten und deren Strukturen für den Informationsaustausch und legt den Schwerpunkt auf die Interoperabilität der Daten im Rahmen des BIM.

Mit anderen Worten, die IDM legt fest, „was“ in Bezug auf das Informationsmanagement getan werden muss, während die IDS definiert, „wie“ dies auf technischer Ebene umgesetzt werden soll. Die IDS könnte sich auf die von der IDM bereitgestellten Richtlinien stützen und diese technisch umsetzen.

Zusammengefasst kümmert sich die IDM mehr um Geschäftsprozesse und Informationsanforderungen in einem BIM-Projektkontext, während sich die IDS auf die technischen Aspekte der Strukturierung und des Informationsaustauschs gemäß diesen Anforderungen konzentriert. Beide sind somit komplementär und können zusammen eine wichtige Rolle bei der effektiven Implementierung von BIM im Bausektor spielen.

2.2.2 buildingSMART Data Dictionary (bSDD)

Der buildingSMART Data Dictionary (bSDD) ist laut Herr Léon van Berlo ein Dienst zum Veröffentlichen und Teilen von Baudaten im Internet [32]. Er bietet in einer standardisierten Weise den Institutionen die Möglichkeit Klassen, Merkmalen (aus Bibliotheken „Projekt Libraries“ ebenfalls), Klassifikationen, Übersetzungen und Beziehungen zu teilen, um die Datenqualität und Interoperabilität zu gewährleisten. Genutzt von BIM-Profis, unterstützt er sowohl internationale Standards als auch firmenspezifische Anforderungen, im Einklang mit ISO 12006-3 und ISO 23386 [33].

bSDD dient als ein wesentliches Instrument, das darauf ausgelegt ist, den Prozess der Integration von semantischen Informationen in IFC-Datensätze zu beschleunigen und gleichzeitig die Konsistenz der BIM-Informationen zu gewährleisten. Dieses Werkzeug erleichtert eine nahtlose Verbindung zwischen Objekten und etablierten Standards und fördert daher einen standardisierten Arbeitsablauf beim Erstellen von IFC-Daten sowie IDS-Dateien. Dabei wird sichergestellt, dass Informationen genau und systematisch in der IFC-Datei organisiert werden, was die Datenintegrität und Nutzbarkeit erhöht [32].

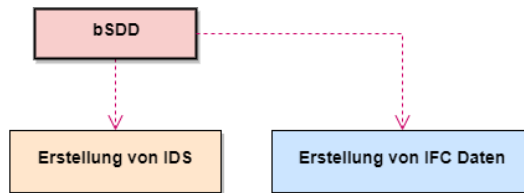


Abbildung 18. Zwei mögliche bSDD-Workflows für eine effiziente Nutzung des IFC-Schemas

Die Komponenten im bSDD, einschließlich Klassen und Merkmale, werden neben ihrem eindeutigen Namen et Datentyp mit einer zusätzlichen Kennzeichnungssequenz versehen, die als URI (Uniform Resource Identifier) bezeichnet wird. Diese URI dient als Bezugsquelle beim Eintrag innerhalb einer IFC-Datei und könnte beim Erstellen einer Spezifikations-Facette in IDS als ein URI-Parameter-Wert eingetragen werden.

Der folgende Schnappschuss zeigt beispielhaft die mit dem Merkmal „Colour Temperature“ verknüpften Daten in der bSDD-Datenbank:

Property	
Colour Temperature English	
Code	ColourTemperature
URI	https://identifier.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/ColourTemperature
Definition	The colour temperature of any source of radiation is defined as the temperature (in Kelvin) of a black-body or Planckian radiator whose radiation has the same chromaticity as the source of radiation. Often the values are only approximate colour temperatures as the black-body radiator cannot emit radiation of every chromaticity value. The colour temperatures of the commonest artificial light sources range from less than 3000K (warm white) to 4000K (intermediate) and over 5000K (daylight).
Description	Technical note: in IFC this property takes 'IfcThermodynamicTemperatureMeasure' as value. Such objects are not included in bSDD for simplicity reason. IFC also doesn't enforce particular units, but recommends using metric SI units (metre, kilogram, etc.). Read the IFC documentation for more information.
Dictionary	IFC
Dictionary version	4.3
Dictionary license	CC BY-ND 4.0
Owner	buildingSMART International
DataType	Real
IsDynamic	False
Property/ValueKind	Single
Status	Active
VersionDateUtc	2024-01-12
Classes	
Compact Fluorescent	A fluorescent lamp having a compact form factor produced by shaping the tube.
Fluorescent	A typically tubular discharge lamp in which most of the light is emitted by one or several layers of phosphors excited by ultraviolet radiation from the discharge.
Halogen	An incandescent lamp in which a tungsten filament is sealed into a compact transport envelope filled with an inert gas and a small amount of halogen such as iodine or bromine.
High Pressure Mercury	A discharge lamp in which most of the light is emitted by exciting mercury at high pressure.
High Pressure Sodium	A discharge lamp in which most of the light is emitted by exciting sodium at high pressure.
Lamp	A lamp is an artificial light source such as a light bulb or tube.
LED	A solid state lamp that uses light-emitting diodes as the source of light.
Metalhalide	A discharge lamp in which most of the light is emitted by exciting a metal halide.
OLED	A solid state lamp that uses light-emitting diodes as the source of light whose emissive electroluminescent layer is composed of a film of organic compounds.
Property Set: Lamp Type Common	A lamp is a component within a light fixture that is designed to emit light.

Abbildung 19. Beispiel für die Definition eines Merkmals in bSDD: „Colour Temperature“

Im aktuellen Zustand bietet bSDD nicht die Möglichkeit, direkt den entsprechenden IFC-Datentyp für das betreffende Merkmal einzutragen, was bei der Erstellung einer Merkmals-Facette erforderlich ist. Der Nutzer muss daher selbst die Informationen in der Definition des betreffenden Merkmals durchlesen, um den Datentyp auswählen zu können, der der korrekt verwendeten Einheit für das Merkmal entspricht.

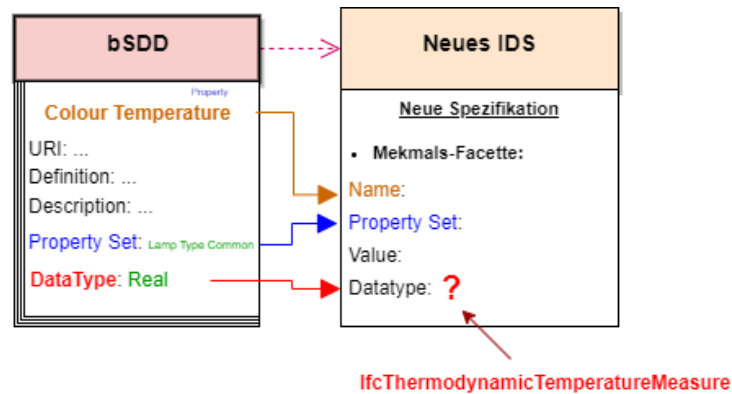


Abbildung 20. Nutzung der in der bSDD-Datenbank gespeicherten Informationen bei der Erstellung einer IDS

Class	English
Internal combustion engines	
Code	Pr_60_70_65_41
Class type	Class
URI	https://identifier.buildingsmart.org/uri/nbs/uniclass2015/1/class/Pr_60_70_65_41
Dictionary	Uniclass 2015
Dictionary version	1
Dictionary license	Creative Commons Attribution-NoDerivatives 4.0 International
Dictionary state	Preview
Owner	NBS
Parent class	Power generators, engines and packaged combined heat and power (CHP) units
Status	Preview

Abbildung 21. Beispiel für die Definition einer Klassifikation in bSDD: „Internal combustion engines“

2.2.3 BIM Collaboration Format (BCF)

Das „BIM Collaboration Format“ (BCF) optimiert die Kommunikation von BIM-Daten und adressiert Herausforderungen im standardisierten Austausch von IFC-Daten sowie in der Diskussion über Modellobjekte [34]. BCF ersetzt ein ungenaues E-Mail- und Anrufsystem durch ein System, das Probleme mit präzisen Modellkoordinaten verknüpft, wodurch Klarheit und Effizienz verbessert werden. Diese Methode funktioniert durch den Austausch von IFC-Daten zwischen den verschiedenen Beteiligten eines Bauprojekts und ermöglicht die Identifikation und das Teilen von modellbasierten Problemen über verschiedene BIM-Softwareplattformen hinweg, ohne auf proprietäre Formate zurückgreifen zu müssen. Die Beteiligten eines Projekts können das BCF-Format verwenden, um über Model-basierten Problemen [35] bzw. Korrekturvorschlägen zu kommunizieren, in dem koordinierte Objekte und mit Metadaten angereicherte Unterlagen (z. B. Fotos, PDF usw..) zur Detaillierung und zum Teilen eingefügt werden können, was eine effiziente Problemlösung unterstützt [36].

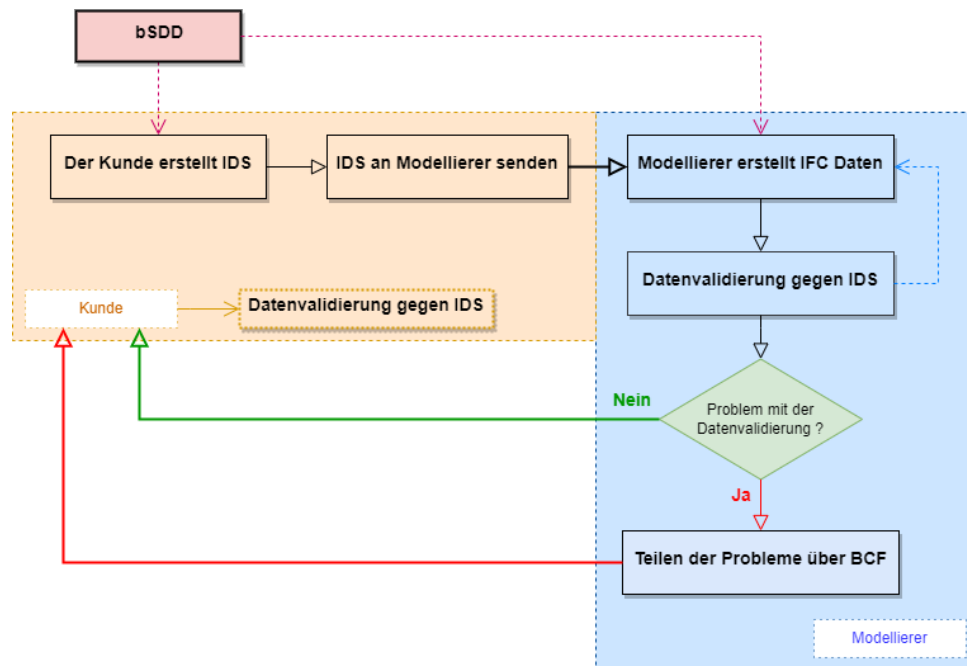


Abbildung 22. Möglicher Workflow für die Nutzung von bSDD, IDS, IFC und BCF gemeinsam

2.2.4 IFC-Projekt Bibliotheken und Objekt-Vorlagen

„IfcProjectLibrary“ [37] sammelt die folgende Bibliothekselemente in einem referenzierten Datensatz:

- Untertypen von „IfcTypeObject“ (Objekt-Vorlagen) [z. B. IfcWallType]
- „IfcPropertySetTemplate“ Elemente

Bei der Anwendung solche Informationen im Kontext von IDS, können die Entitäten

„IfcTypeObject“ aus einer „Projekt Bibliothek“ zur Definition der Anwendbarkeit herangezogen

werden. Für die Festlegung der Parameter in einer Anforderungs-Merkmalfacette sind die Entitäten „IfcPropertySetTemplate“ dafür geeignet; Die Werte der Namen, Datentypen gelten als Parameter für die Erstellung neuer Merkmals-Facetten.

Die Festlegung der Optionalität (erforderlich, optional oder verboten) obliegt aber dem Anwender und sollte basierend auf den spezifischen Charakteristika des jeweiligen Projekts erfolgen.

2.3 Workflows

In dieser Rubrik geht es darum, die verschiedenen Arbeitsabläufe aufzulisten, die es ermöglichen, die für das laufende Projekt erforderlichen Tests durchzuführen:

Erste Ebene: Erstellung einer neuen IDS-Datei.

- Eintragen von IDS-Metadaten.
- Erstellung von IDS-Spezifikationen: Metadaten, Anwendbarkeit und Anforderungen.
- Die neu erstellten IDS exportieren.

Zweite Ebene: Bearbeitung einer bestehenden IDS-Datei.

- Eine bestehende IDS importieren.
- Die betreffenden Spezifikationen ändern.
- Die neu bearbeitete IDS exportieren.

Dritte Ebene: Import und Validierung auf Basis von IDS-basierten Informationsanforderungen.

- Importieren von IFC-Dateien und die Darstellung des 3D-Modells.
- Importieren der erstellten IDS-Datei.
- Anzeige der Ergebnisse der IDS-Validierung.
- Korrektur von semantischen Informationen der IFC-Datei auf Basis der IDS-Spezifikationen.

Vierte Ebene: Erstellung des Berichts der etablierten Prüfung von IDS-basierten Informationsanforderungen.

3 Verwandte Arbeiten

In diesem Kapitel wird das Ziel verfolgt, einen sorgfältigen Überblick über die interessanten bestehenden wissenschaftlichen Forschungen, die sich direkt oder indirekt auf dem Konzept von IDS beziehen, zu präsentieren. Eine umfassende Betrachtung der aktuellen Produkte, die im Zusammenhang mit IDS stehen, wird anschließend vorgenommen.

3.1 Literatur Recherche

3.1.1 „Developing a BIM-based methodology to support renewable energy assessment of buildings“²

Diese Forschung untersucht die Verwendung eines IFC-konformen Simulationsmodells zur Durchführung einer Solar-PV-Simulation. Dabei wird speziell eine Validierung des Eintrags der notwendigen Informationen für die Solar-PV-Simulation in die IFC-Datei vorgenommen. Diese Vorgehensweise wird im Workflow als IDR (Input Data Requirements) bezeichnet (siehe Abbildung) und ist in Einklang mit den Funktionalitäten des IDS (Information Delivery Specification) [38].

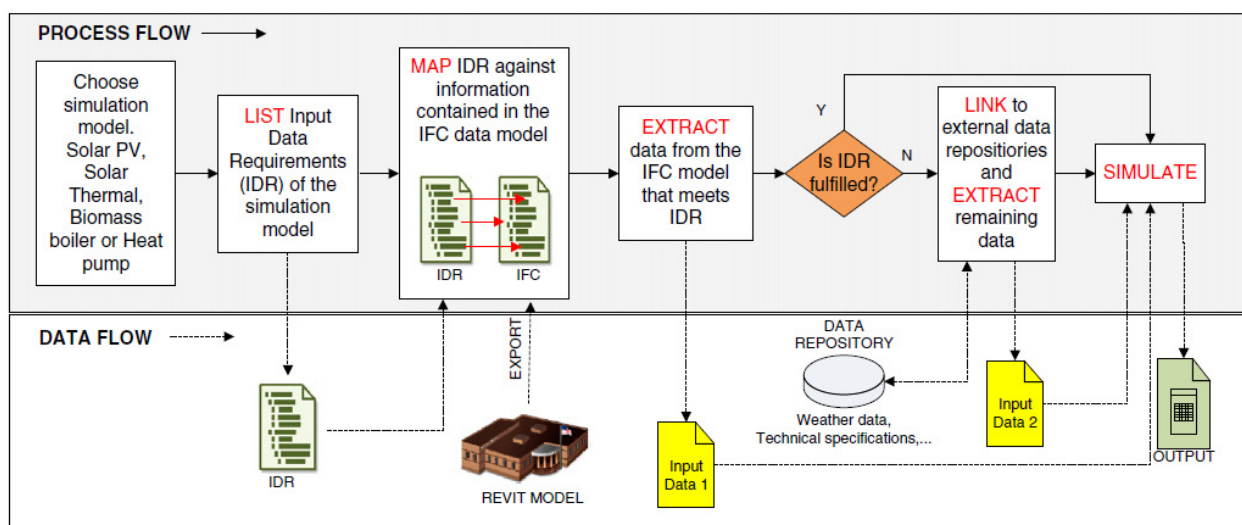


Abbildung 23. Konzeptioneller Workflow für die Entwicklung von IFC-konformen Simulationsmodellen [38]

² <https://www.sciencedirect.com/science/article/pii/S0926580513001623> [zuletzt 10.01.2024]

3.1.2 „Creating Information Delivery Specifications using Linked Data“ [39]

Das Papier befasst sich mit Fortschritten im Bereich des Gebäudeinformationsmanagements (BIM) in Bezug auf die Verwendung von verknüpften Daten (*English: Linked Data*) in Informationslieferungsspezifikationen (IDS). Die angesprochenen Probleme beziehen sich auf den Datenaustausch sowie die Dokumentation von Informationsanforderungen, die den Bereich Bausektor betreffen. Dabei wird einen Ansatz mit verknüpften Daten vorgeschlagen, um IDSs zu generieren, und diskutiert die Grenzen aktueller Praktiken wie die Nutzung von PDF-Dokumente sowie das Potenzial von Technologien wie mvdXML, bSDD und Verknüpfte Gebäudedaten. Es stellt einen Systemprototyp vor, der die Definition, das Teilen und die Wiederverwendung von Eigenschaftsdefinitionen und IDSs ermöglicht, wodurch die Interoperabilität im Sektor verbessert wird. Das Dokument überprüft auch diese Methode und hebt ihre Vorteile zur Förderung des Austauschs und der Standardisierung von Informationen im Bauwesen hervor [39].

3.1.3 „Extending information delivery specification for linking distributed model checking services“ [40]

Dieses Papier schlägt vor und beschreibt eine mögliche Erweiterung der Information Delivery Specification (IDS) für Building Information Modeling (BIM), die es durch die Verwendung des Attributs „xlink“, das mit einer URL („href“) versehen ist, ermöglicht, die Verlinkung mit externen Funktionen zu integrieren. Diese Funktionen erfordern Parameter und eine Datenbasis, auf der eine Berechnung durchgeführt werden kann. Diese Parameter enthalten notwendige Informationen für die Berechnung, sie werden bereitgestellt und als Parameter in einer dynamischen URL an den externen Berechnungsdienst übertragen (siehe Abbildung). Diese Erweiterung hat das Ziel, mehr Flexibilität und Effizienz des IDS-Formats im OpenBIM-Rahmen zu bieten, indem sie externe Ressourcen und umfassendere Modellprüfungen ermöglicht [40]. Dadurch können die notwendige Informationsanforderungen besser abgedeckt werden, da ein IDS so ein breiteres Spektrum an Möglichkeiten umfassen kann.

```

1 <ids:extension minOccurs="1" maxOccurs="1"
  measure="IfcBoolean">
2   <ids:resource
3     xlink:href="https://www.localhost.de/
      touches/ifcelementadd=IFCSITE&
      ifcelementor1=1C_xd4xVP5R8Y8yHywtYQL"
    />
4   <ids:value>
5     <ids:simpleValue>True</ids:simpleValue>
6   </ids:value>
7 </ids:extension>

```

Abbildung 24. Mögliche Syntax für die Nutzung des Attributs „xlink“

Die Idee ist in der Tat sehr interessant. Die Anwendung einer solchen Methodologie könnte es Nutzern ermöglichen, die aktuellen relative limitierte Fähigkeiten der „Information Delivery Specification“ (IDS) zu erweitern, indem sie externe Funktionen integrieren. Diese könnten beispielsweise die Auswahl von Elementen auf der Basis spezifischer geometrischer Merkmale ermöglichen oder fordern, dass bestimmte Elemente genau definierte geometrische Kriterien erfüllen. Dies würde einen bedeutenden Fortschritt in der Personalisierung und Präzision von Spezifikationen in BIM-Projekten darstellen.

3.1.4 „Interoperable validation for IFC building models using open standards“ [41]

Diese Recherche präsentiert ein Validierungstool von IFC-Modellen. Das Tool wird auf der Plattform bimserver.org implementiert und verwendet „mvdXML“, ein offener Standard, zur Strukturierung von Validierungsregeln und das BIM Collaboration Format (BCF) zur Berichterstellung. Für die Validierung wird die niederländische „Rgd“ BIM-Norm und das norwegische „Statsbygg“ BIM-Handbuch verwendet.

Der Inhalt fokussiert auf die Etablierung standardisierter, interoperabler Methoden zur Validierung von Gebäudemodellen im Kontext von OpenBIM. Die Verwendung offener Standards wie mvdXML und BCF entspricht den Zielen von OpenBIM, transparente, kollaborative Prozesse im Gebäudeinformationsmodellierung zu fördern [41].

3.1.5 Use Case Management von buildingSMART

Der Use Case Management Dienst von buildingSMART International bietet ein effektives Werkzeug zur Erfassung und Spezifizierung bewährter Methoden in der Bauindustrie [42].

Durch die Veröffentlichung von Informationsanforderungen für reale Bauprojekte, organisiert nach spezifischen Anwendungsfällen, ist diese Plattform besonders nützlich, um die Relevanz und praktische Anwendung von Information Delivery Specifications (IDS) in der Bauindustrie zu begreifen.

- **„Modellbasierte Bewehrung verlegen“ [43]**

Dieser Anwendungsfall wurde von buildingSMART Switzerland erstellt und beschreibt die notwendigen Anforderungen und Umsetzung des Verlegens von Bewehrungen auf der Baustelle ab Ausführungsmodell. Dabei wurden mehrere Schlüsselziele festgelegt, wie beispielweise ein Durchgängiger Informationsfluss und weniger Fehlinterpretationen, um die Effizienz und Genauigkeit in Baupraktiken zu verbessern. Er beschäftigt sich damit, wie Bewehrungsdesignmodelle effektiv auf der Baustelle genutzt werden können.

Die Unterlagen geben zunächst einen Überblick über die festgelegten Anforderungen für die geometrische sowie die alphanumerischen Informationen. Die Erfüllung der Anforderungen im

Bezug der alphanumerischen Informationen sind mittels einer IDS-Datei³ automatisch nachzuprüfen, welches die erforderlichen Informationen und den Datentypen, die im IFC-Dateiformat zu inkludieren sind, enthält.

- „OpenBIM für Tageslichtplanung und -analyse im Hinblick auf die Energieeffizienz von Gebäuden“ [44]

Dieser Anwendungsfall stellt einen Workflow für die Durchführung einer Tageslichtanalyse dar. Dieser wurde als kritischer Prozess qualifiziert, der bei Bauprojekten in der realen Welt wie dem Gebäude der Sozialeinrichtungen der Meffert AG in Bad Kreuznach (Deutschland) angewendet wurde; Es werden innerhalb dieses Anwendungsfalls die notwendigen Schritte zur Durchführung von Tageslichtsimulationen in den frühen Designphasen der erwähnten Bauprojekten veranschaulicht. Hauptziel dieser Simulationen ist es, die optimale Positionierung und Größe von Fenstern, Oberlichtern und Vorrichtungen für Schattierung und Blendungsschutz, die entscheidend sind, um den Einfluss von Designentscheidungen auf den Energieverbrauch und den Komfort der Bewohner zu bewerten und fundierte Entscheidungen zu ermöglichen. Der Workflow beinhaltet mehrere Beteiligte, darunter Auftraggeber, Architekten, Nachhaltigkeitsberater, Lichtdesigner, Energieaudatoren und BIM-Manager.

Der Prozess beginnt, sobald ein Konsens unter allen Beteiligten erreicht ist. Der Architekt leitet die Phase ein, indem er ein Referenz-IFC-Modell an den Designer sendet, welches die Grundlage für die Analyse bildet. Der Designer überprüft daraufhin gründlich die Zielkriterien und Auftraggeber-Anforderungen, um die Ziele der Tageslichtanalyse vollständig zu verstehen. Etwaige Unstimmigkeiten oder Probleme werden über das „BIM Collaboration Format“ (BCF) an den Architekten zurückgemeldet. Für die Tageslichtsimulation werden die erforderlichen Anforderungen in einer IDS-Datei⁴ festgelegt und Klimadaten wie Wetterdaten für die Simulation beschafft. Die Ergebnisse der Tageslichtsimulation werden sorgfältig überprüft und dokumentiert, einschließlich Zertifikaten und Empfehlungen, die dem Architekten und Auftraggeber zur Verfügung gestellt werden. Dieser umfassende Workflow stellt sicher, dass jeder Aspekt der Tageslichtanalyse gründlich bewertet und kommuniziert wird, was die Gesamtqualität und Nachhaltigkeit des Gebäudedesigns verbessert.

³ bit.ly/IDS-Modellbasierte-Bewehrung-verlegen [zuletzt 10.01.2024]

⁴ <https://bit.ly/IDS-Daylighting-Design-and-Analyses> [zuletzt 10.01.2024]

- „Durchführung eines modellbasierten Aufgabenmanagements aus Sicht von bauausführenden Unternehmen“ [45]

Der beschriebene Anwendungsfall befasst sich mit der Durchführung eines modellbasierten Aufgabenmanagements aus der Perspektive von Bauunternehmen. Der Prozess beginnt mit der Identifikation einer Aufgabe durch einen Aufgabenerfasser. Anschließend wird diese Aufgabe zur weiteren Bearbeitung an eine zuständige Person oder Abteilung, den sogenannten „Aufgabenbearbeiter“, weitergeleitet. Dieser prüft die Aufgabe auf verschiedene Parameter, insbesondere darauf, ob sie in seinen Verantwortungsbereich fällt und sachlich begründet ist. Bei positiver Bewertung bearbeitet der Aufgabenbearbeiter die Aufgabe und meldet den Abschluss zurück. Im Falle einer negativen Bewertung wird die Aufgabe zur Neuuzuweisung an den Aufgabenerfasser zurückgegeben. Der Anwendungsfall umfasst die Nutzung eines Bauwerksdatenmodells nach festgelegten Modellierungsrichtlinien und erfordert die Definition von Datenaustauschformaten sowie den Einsatz geeigneter Software-Anwendungen zur Ableitung und Verarbeitung von Modellinformationen sowie zur Erstellung von Kommunikationsereignissen, beispielsweise mit dem „BIM Collaboration Format“ (BCF). Das Hauptziel dieses Ansatzes ist die Verbesserung der Steuerung und Kontrolle von Unternehmens- und Organisationsabläufen, insbesondere in der Bauausführung, wobei die Erstellung des erforderlichen Bauwerksdatenmodells nicht Teil des Anwendungsfalls ist. Für einen effizienten Arbeitsablauf im Rahmen des modellgestützten Aufgabenmanagements wurde eine Excel-Datei ⁵ öffentlich zugänglich gemacht. Diese Datei demonstriert die verwendeten Anforderungen in Bezug auf die notwendigen alphanumerischen Informationen. Dieser Ansatz ermöglicht ein klares und strukturiertes Beispiel für eine Darstellung der erforderlichen Anforderungen, die für eine effiziente und zielgerichtete Aufgabenverwaltung im Rahmen von BIM-Projekten nötig sein können.

- „Gebäudehülle“ [46]

Der „Gebäudehülle“-Anwendungsfall beschreibt die Nutzung von BIM für die effektive Planung und Ausführung von Gebäudehüllen. Er betont die Wichtigkeit des Austauschs von Informationen und geometrischen Daten unter verschiedenen Beteiligten wie Planern, Spezialisten und Bauunternehmern. Diese Koordination ist entscheidend für das Design der Gebäudehülle. Der Anwendungsfall fördert die interdisziplinäre Zusammenarbeit in Bereichen wie Fassadenplanung, Architektur, Bauphysik und HVAC-Planung. Er geht auf die Anwendung von BIM in diesen Bereichen ein und behandelt die Informationsanforderungen für

⁵ <https://bit.ly/modellbasierten-Aufgabenmanagements> [zuletzt 10.01.2024]

verschiedene Projektphasen. Eine PDF-Datei ⁶, die die benötigten Anforderungen in Bezug auf die notwendigen alphanumerischen Informationen für jede Projektphase veranschaulicht, online verfügbar.

3.2 Implementierungen

In diesem Abschnitt werden Implementierungen vorgestellt, die für das Thema dieser Masterarbeit von besonderer Relevanz sind.

3.2.1 BIMcollab

BIMcollab ist ein umfassendes Lösungsset für BIM-basierte Gebäude- und Infrastrukturdaten, das aus drei Hauptanwendungen besteht [47]:

- a) BIMcollab Zoom: Ein Modellprüfer für Daten- und Geometrieprüfungen sowie Mengenermittlungen, integriert in Issue-Management-Workflows.
- b) BIMcollab Nexus: Eine BIM-Koordinationslösung, die BIM-Anforderungen (IDS), Modellzusammenarbeit und Issue-Management in der Cloud zentralisiert.
- c) BIMcollab Twin: Ein Dokumentenmanagementsystem für den gesamten Lebenszyklus von Projekten, das eine einheitliche Informationsquelle bietet und zur Erstellung eines Digitalen Zwillings dient.

Ein Webinar von BIMcollab ⁷ sowie ein Webinar von buildingSMART ⁸ zeigen eine Demonstration der Nutzung dieses Werkzeugs bei der Erstellung eines IDS und bei der Durchführung einer auf IDS basierenden IFC-Validierung über ihr Revit-Plugin.

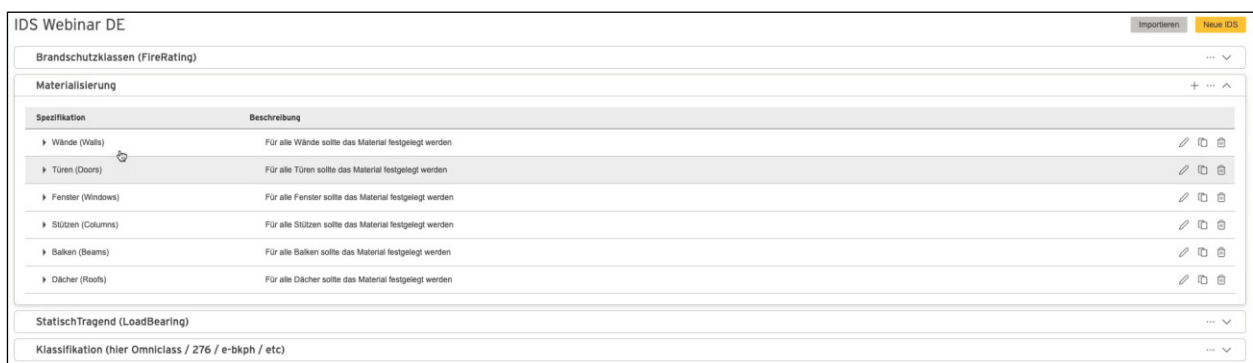


Abbildung 25. Webseitengestaltung vom BIMcollab-Tool für die Erstellung einer IDS-Datei.

⁶ <https://bit.ly/Informationsanforderungen-Gebaudehuelle> [zuletzt 10.01.2024]

⁷ <https://bit.ly/webinar-BIMcollab-IDS-demo> [zuletzt 10.01.2024]

⁸ https://www.youtube.com/watch?v=kNTKt_C0wQ4 [zuletzt 10.01.2024]

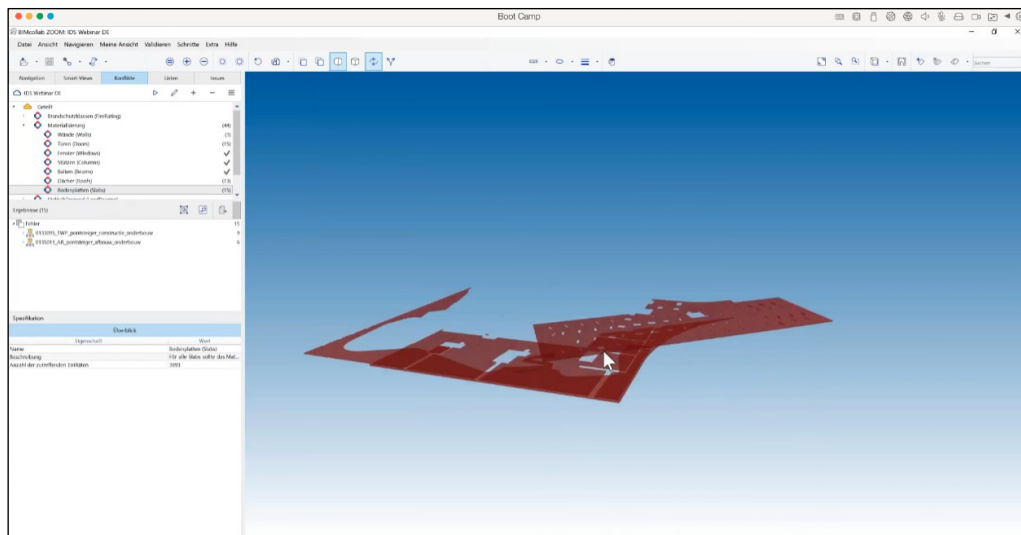


Abbildung 26. Ausführung einer IDS-basierten IFC Validierung in BIMcollab ZOOM

3.2.2 usBIM von ACCA

ACCA bietet eine umfangreiche Palette an Softwarelösungen im Bereich BIM an [48]. Dank usBIM.ids ermöglicht es die Erstellung von IDS-Dateien basierend auf Benutzerspezifikationen, die anschließend als Validierungswerkzeug in der usBIM-Anwendung eingesetzt werden können. Dies ermöglicht die Darstellung der Art von Problemen, die während des Validierungsprozesses auftreten, und deren Export und gemeinsames Teilen im BCF-Format [49].

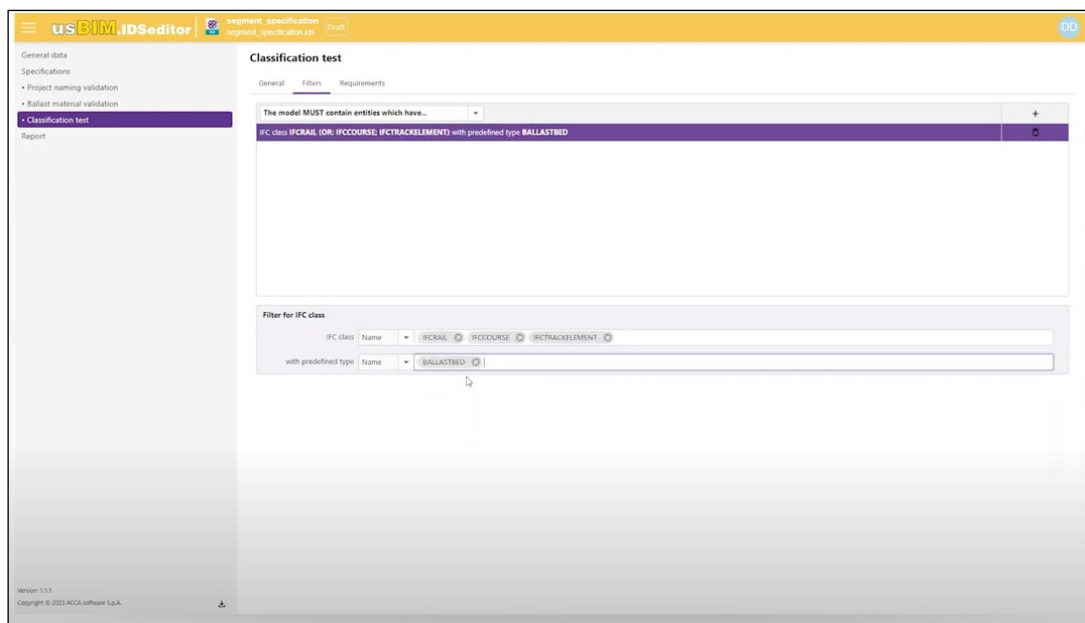


Abbildung 27. Erstellung einer neuen IDS-Datei mit usBIM.IDSeditor

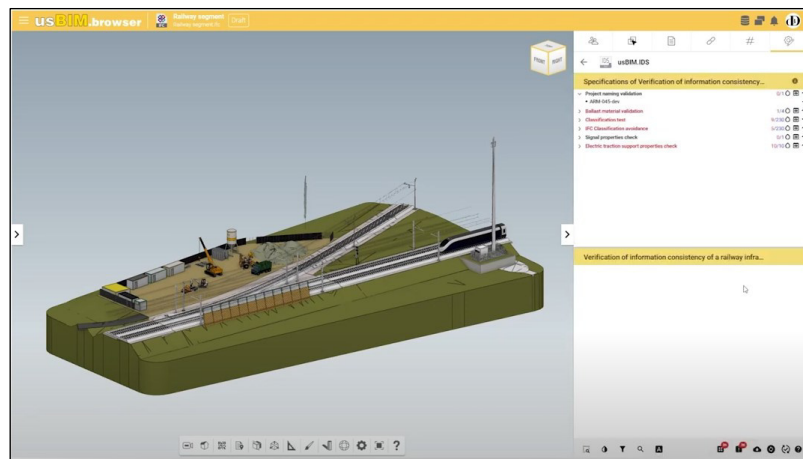


Abbildung 28. Ausführung einer IDS-basierten IFC Validierung anhand usBIM.browser

3.2.3 BlenderBIM IFC Tester

BlenderBIM verwendet IfcOpenShell, eine Open-Source-Softwarelösung für den Umgang mit Industry Foundation Classes [50], um eine Erweiterung für Blender bereitzustellen. Diese Erweiterung ermöglicht es dem Nutzer, IFC-Dateien innerhalb der grafischen Oberfläche von Blender zu importieren, zu erstellen, zu bearbeiten und zu exportieren [51].

Laut der Dokumentation von BlenderBIM ist IfcTester ein Hilfsprogramm in BlenderBIM, das dem Benutzer das Erstellen und Lesen von Information Delivery Specification (IDS)-Dateien ermöglicht [52]. Es erlaubt ihm, IFC-Modelle gemäß IDS zu validieren und Berichte zu generieren.

Dieses Tool erlaubt es aber noch nicht, IDS-Dateien zu erstellen, ermöglicht jedoch bereits die Nutzung einer vorhandenen IDS-Datei, um die Validierung einer IFC-Datei gemäß den in IDS enthaltenen Informationsanforderungen durchzuführen [53].

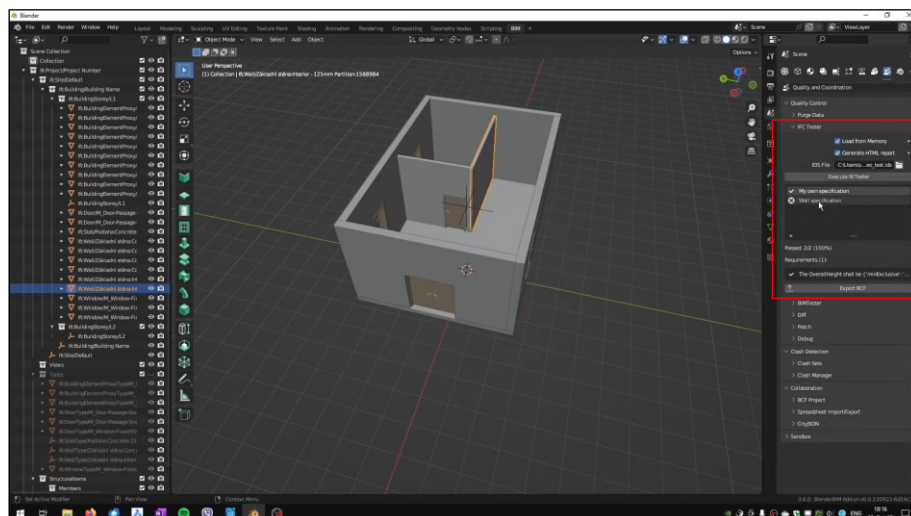


Abbildung 29. Positionierung des Feldes der zu verwendenden Schaltflächen in BlenderBIM zur Validierung einer IFC-Datei mit einer IDS-Datei

3.2.4 Solibri

Solibri, eine Marke von Nemetschek Group, bietet Tools für BIM-Validierung, Konformitätskontrolle, Koordination des Entwurfsprozesses, Designprüfung, Analyse und Code-Überprüfung an. Das Ziel des Unternehmens ist es, Qualitätssicherungslösungen zu entwickeln und zu vermarkten, die die Qualität des BIM-basierten Designs verbessern und den gesamten Entwurfs- und Bauprozess produktiver und kosteneffektiver machen [54]. Zwar hat Solibri ein IDS-Validierungstool online zur Verfügung gestellt, das es Benutzern ermöglicht, ihre BIM-Modelle anhand der in den IDS-Dateien gespeicherten Informationsanforderungen zu überprüfen [55,56], jedoch hat das Unternehmen bislang noch kein Werkzeug bereitgestellt, um IDS-Dateien zu erstellen. Ein im Internet verfügbares Video ⁹ zeigt, wie dieses Tool für eine auf IDS basierende Validierung verwendet wird.

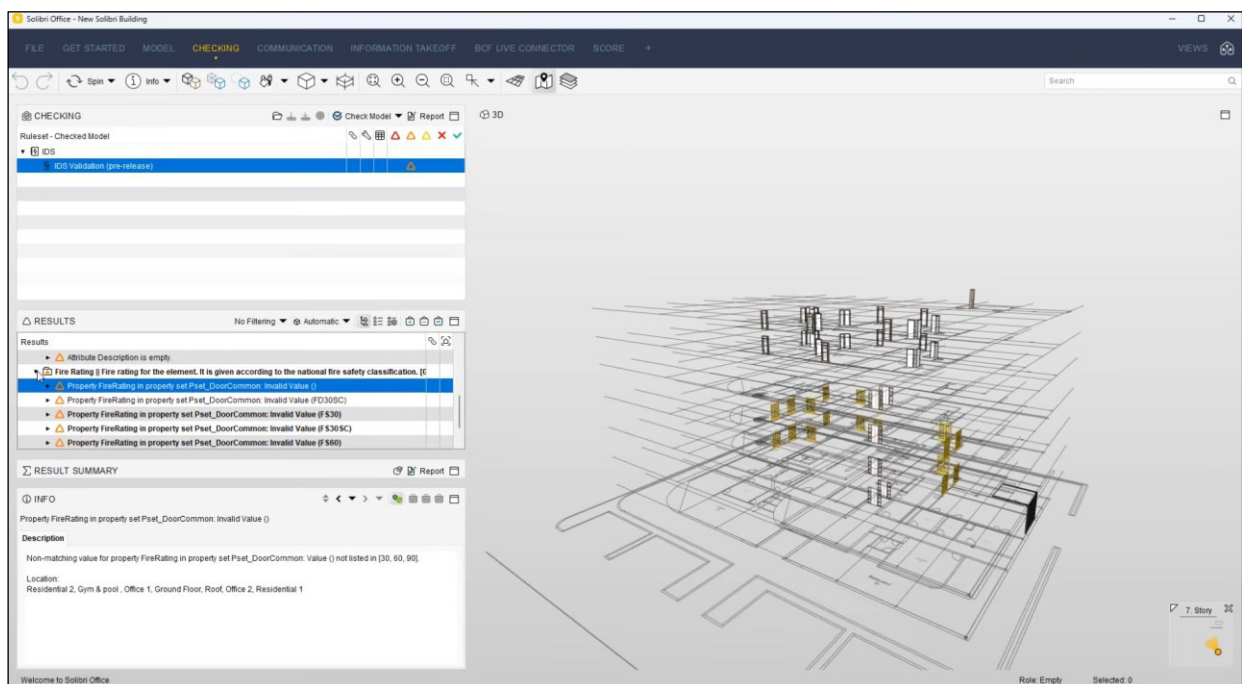


Abbildung 30. Ausführung einer IDS-basierten Validierung einer IDS Datei in Solibri Office

3.2.5 xbim Flex

„xbim“ ist ein Team von Softwareingenieuren, die sich auf die Entwicklung von BIM-Anwendungen für die Bauindustrie spezialisiert haben [57]. „xbim“ bietet mit ihrem Produkt „Flex Flow“ die Möglichkeit, IDS-Dateien zu erstellen und Validierungen auf Basis von IDS für IFC-Modelle durchzuführen [58,59].

⁹ <https://www.youtube.com/watch?v=VoUuiNZMTPs> [zuletzt 10.01.2024]

Das Tool zur Erstellung von IDS-Dateien befindet sich noch in der Beta-Version und ist kostenlos über den Link xbim.it/xids zugänglich.

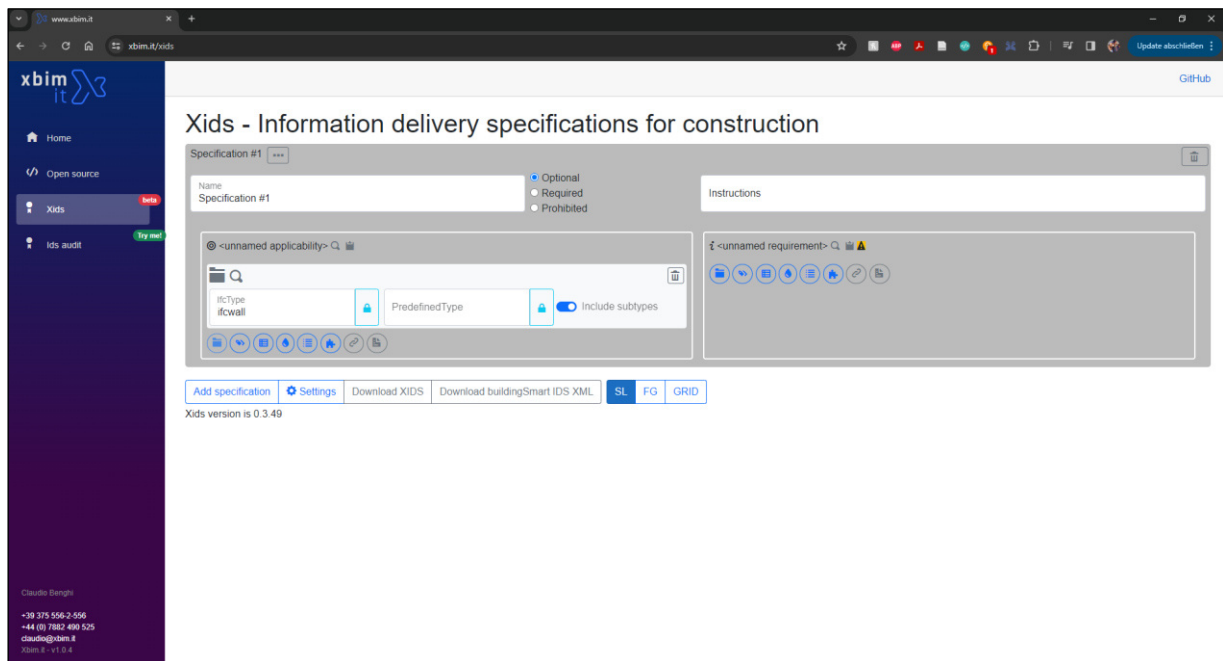


Abbildung 31. Erstellung einer neuen IDS Datei mit „xbid Flex“

3.2.6 Plannerly

Plannerly stellt eine Option dar, maßgeschneidert für die Bedürfnisse von Architekten, Ingenieuren und Bauträgern. Ziel ist es, das Projektmanagement spielend leicht und intuitiv zu gestalten, dank einer benutzerfreundlichen Oberfläche. Ein besonderes Merkmal ist die Integration von BIM-Anforderungen und Tracking, wodurch die Einhaltung der ISO 19650-Norm sichergestellt wird. Die Plattform vereinfacht die Zusammenarbeit, indem sie die gemeinsame Erstellung von Verträgen ermöglicht und gleichzeitig die Überprüfung von BIM-Modelllieferungen automatisiert [60]. Zusätzlich zu diesen Features unterstützt Plannerly umfangreich die Erstellung von Information Delivery Specifications (IDS). Diese Unterstützung basiert unter anderem auf dem COBie-Standard und dem buildingSMART Data Dictionary (bSDD). Ebenso ermöglicht die Plattform eine effektive Validierung von IFC-Modellen, die auf diesen IDS basieren [61].

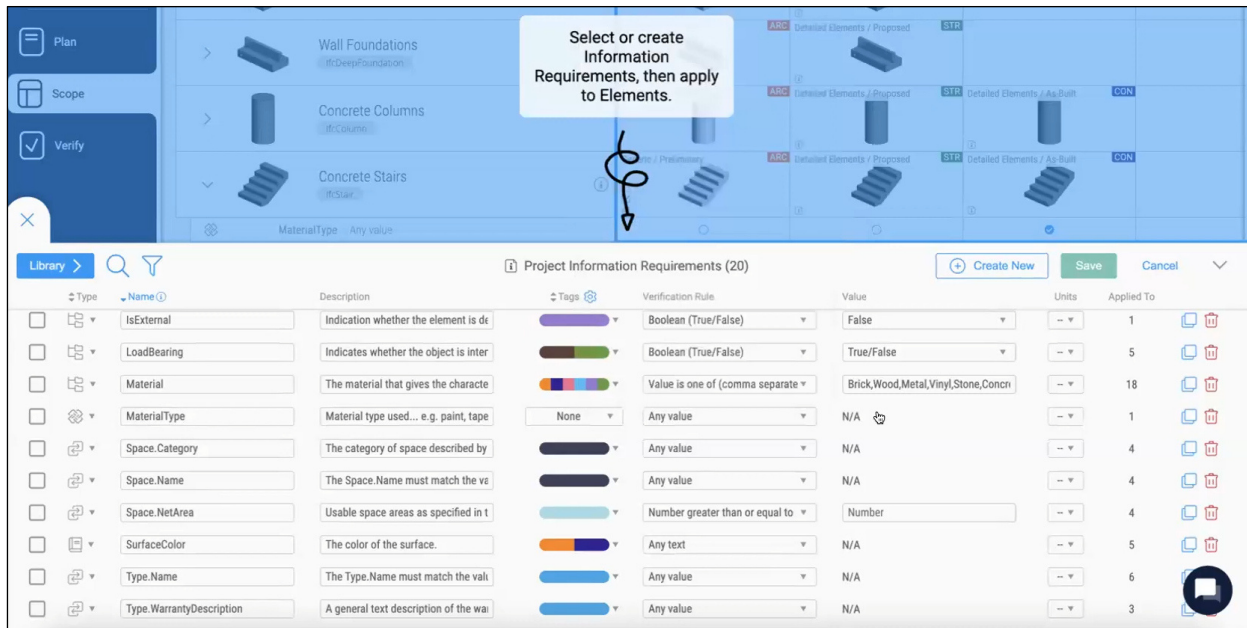


Abbildung 32. Erstellung einer neuen IDS in Plannerly

3.2.7 BIMQ

BIMQ dient als eine Informationsverwaltungsplattform, die speziell auf BIM-Projekte zugeschnitten ist [62]. Sie ermöglicht die Erstellung von IDS-Dateien, die von anderen Anwendungen genutzt werden können [63,64].

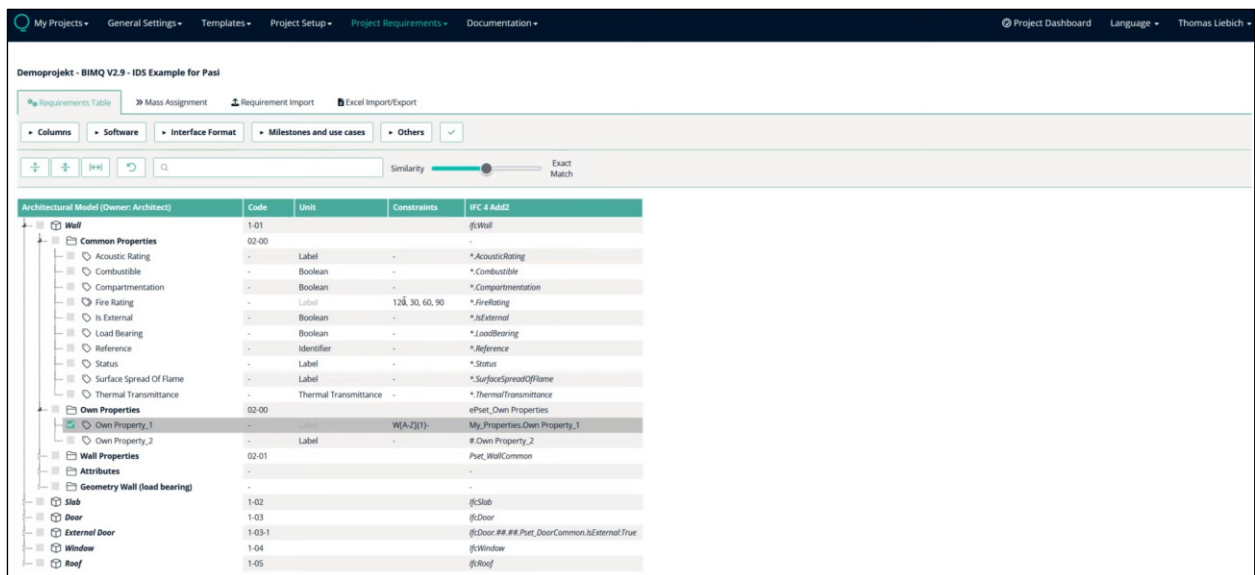


Abbildung 33. Erstellung einer neuen IDS in BIMQ

3.2.8 Andere Lösungen

Die vorherige erwähnten Softwareprodukte zeigen eine Auswahl der von Bauinformatik-Unternehmen in Verbindung mit IDS durchgeführten Implementierungen. Es ist auch möglich, an diesem Punkt das Vorhandensein von Implementierungen zu erwähnen, die dank persönlicher bzw. gemeinschaftlicher Initiativen realisiert wurden:

- Artur Tomczaks IDS-Editor:

Dieses Projekt wurde am 14.10.2022 entwickelt [65]. Seitdem ist das Format der durch dieses Werkzeug erstellten IDS nicht mehr aktuell. Es verwendet beispielsweise das Attribut „location“ sowie das Attribut „necessity“, wobei letzteres dazu dient, die Optionalität einer Spezifikation zu definieren. Dies entspricht nicht den Anforderungen, die bereits bei der Erstellung einer IDS-Datei diskutiert wurden (Im Unterkapitel **2.1.2 Spezifikationen**).

```
<specification name="Test specification (IfcWall)" necessity="required">
  <applicability>
    <entity>
      <name>
        <simpleValue>IfcWall</simpleValue>
      </name>
      <predefinedtype>
        <simpleValue>TEST</simpleValue>
      </predefinedtype>
    </entity>
    <classification location="any">
      <value>
        <simpleValue>TEST</simpleValue>
      </value>
      <system>
        <simpleValue>TEST</simpleValue>
      </system>
    </classification>
    <property location="any">
      <propertyset>
        <simpleValue>TEST</simpleValue>
      </propertyset>
      <name>
        <simpleValue>TEST</simpleValue>
      </name>
    </property>
  </applicability>
</specification>
```

Abbildung 34. Mit Tomczaks IDS-Editor erstellter Codeauszug

- IDS-Converter von Carlos Dias:

IDS-Converter ist ein Open-Source-Projekt, das die Generierung einer IDS-Datei erleichtern soll, indem Informationen aus einer Excel-Datei mit den zu erstellenden Spezifikationen-Attributen extrahiert werden. Eine zugängliche Excel-Vorlagendatei steht zur Verfügung, um diese Datei zu erstellen, die dann in eine IDS-Datei umgewandelt werden kann [66].

4 Konzeption und Implementierung

4.1 Anforderungen

Im Rahmen einer wissenschaftlichen Forschung, und trotz der Verfügbarkeit von kommerziellen Lösungen auf dem Markt, die die Erstellung und Nutzung von IDS zur Validierung von IFC-Modellen ermöglichen, könnte es aus objektiver Sicht für einen Bauinformatiker eine wertvolle Gelegenheit darstellen, sich in die Entwicklung einer Anwendung mit diesen Funktionalitäten einzubringen. Dieser Ansatz beschränkt sich nicht nur darauf, vorhandene Kapazitäten zu duplizieren, sondern strebt danach, neue Möglichkeiten zu erforschen, wodurch die aktuellen Grenzen der Nutzung von BIM-Technologien erweitert werden können. Darüber hinaus könnte eine Entwicklung den Weg für zukünftige innovative Implementierungen ebnen, die erweiterte Funktionen umfassen könnten, wie beispielsweise die im Unterkapitel **„3.1.3 Extending information delivery specification for linking distributed model checking services“** diskutiert wurden.

Das Engagement in dieser Art von Entwicklung könnte die wissenschaftliche Gemeinschaft nur bereichern und bietet neue Perspektiven und Werkzeuge, die unsere Herangehensweise an BIM signifikant transformieren können.

Die geplante Implementierung wird auf einer Angular-Anwendung basieren, die im Rahmen eines vorherigen Projekts für das Studienarbeit „Nutzung von Libraries und Templates zur Definition von Attributen in aktuellen Open-Source Modellierungs-Werkzeugen“ entwickelt wurde. Diese Anwendung zeichnet sich durch einen webbasierten Ansatz aus und hat das Ziel, eine Lösung anzubieten, die nicht nur eine IFC4-Datei (Version 4 - Addendum 2 von buildingSMART ¹⁰) lesen kann, sondern auch die an jedem Bauteil angehängten Merkmalsgruppen darstellt und die Möglichkeit bietet, neue Merkmalsgruppen mit oder ohne Vorlagen zu erstellen. Somit dient sie als Ausgangspunkt für die Implementierung der notwendigen Funktionalitäten zur Untersuchung des aktuellen Themas dieser Masterarbeit.

¹⁰ IFC4 Add2 - Addendum 2 [Offizielle Dokumentation]:

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/>

Das Thema dieser Forschung kann in zwei Schlüsselfunktionalitäten unterteilt werden:

- Die Erstellung und Bearbeitung von IDS-Dateien zur effektiven Modellierung von Informationsanforderungen
- Eine Prüffunktion, die die Übereinstimmung von IFC-Modellen mit diesen Anforderungen sicherstellt.

Zusätzlich sollte die Anwendung verschiedene Export- und Import-Funktionalitäten für IDS und IFC-4-Datenformate unterstützen, was die Durchführung der notwendigen Nachweise fördert. Die modulare Architektur der Angular-Anwendung ermöglicht Erweiterungen und Integrationen, was eine flexible Anpassung an sich ändernde zukünftige Anforderungen und Technologien in der BIM-Branche ermöglicht.

Um eine breite Nutzerbasis anzusprechen, wurde die Anwendung unter besonderer Berücksichtigung von Benutzerfreundlichkeit und intuitiver Bedienung entwickelt, sodass Anwender verschiedenster technischer Hintergründe effizient arbeiten können, wobei die technische Browserkompatibilität die Zugänglichkeit erhöht und eine reibungslose Funktion auf verschiedenen Geräten garantiert.

4.2 Architektur des Werkzeugs

4.2.1 Ausgangspunkt

Zunächst soll ein Überblick über die bestehende Architektur der Angular-Anwendung gegeben werden, bevor die neuen Funktionalitäten eingeführt werden. In der ursprünglichen Entwicklung einer Anwendung wurde ein Webworker (`ifc.worker.ts`) eingesetzt, um rechenintensive IFC-bezogene Aufgaben wie Lesen, Modifizieren und Exportieren zu bewältigen, wodurch Browserabstürze vermieden werden. Die `ifc.service.ts`-Datei steuert diese Aktionen und kommuniziert asynchron über einen speziellen Kanal mit `ifc.worker.ts`, der im Hintergrund operiert und Ergebnisse zurückmeldet. Dieser Ansatz ermöglicht es, komplexe Rechenoperationen im Hintergrund durchzuführen, ohne die Haupt-UI-Leistung zu beeinträchtigen und somit eine reibungslose Benutzererfahrung zu gewährleisten. Die folgende Abbildung zeigt einen Überblick über die Struktur der Architektur der Anwendung, die als Ausgangspunkt für die durchzuführende Implementierung dient.

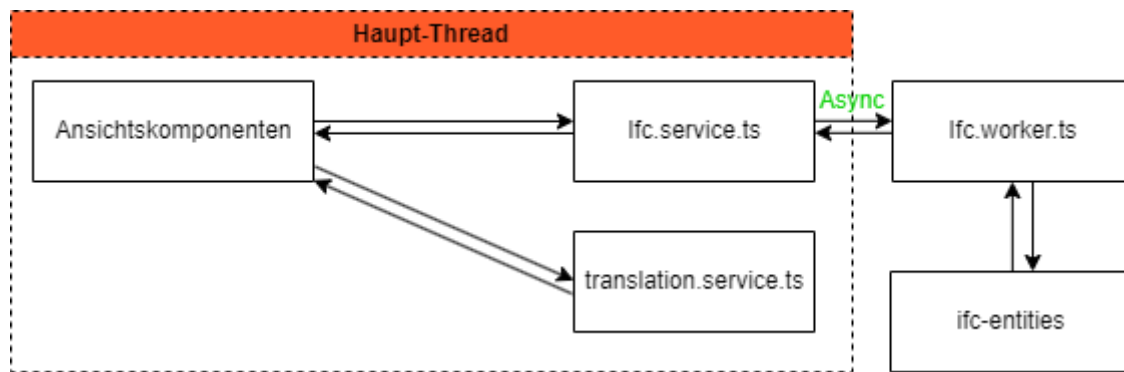


Abbildung 35. Interaktionen in der Angular-Anwendung: Ansichts- und Architekturkomponenten

Mehr Einzelheiten bezüglich der bereits existierenden Implementierung sind über die folgende Datei zugänglich: „Docs/Studienarbeit/Bericht/Bericht-Studienarbeit_Rami-Bali_369458.pdf“.

4.2.2 Neuerungen und Erweiterungen

Die Integration der neuen Funktionalitäten in die bestehende Architektur erfolgt durch die Entwicklung neuer Ansichtskomponenten, die für die Erstellung, Bearbeitung und Nutzung von IDS sowie zur Validierung von IFC-Dateien erforderlich sind.

Die Funktionen, die eine intensive Berechnung erfordern, sowie das Speichern der Daten aus importierten oder aktuell genutzten Dateien, werden weiterhin im Webworker ausgeführt.

4.3 Implementierungsdetails

4.3.1 Erste Ebene: Erstellung einer neuen IDS-Datei

Aus grafischer Sicht wurde ein Bereich eingerichtet, der als „IDS-Manager“ bezeichnet wird und dem Benutzer das Erstellen von IDS-Dateien ermöglicht.

Die Gestaltung der „IDS-Manager“-Seite ähnelt der Hauptseite des Dashboards, wobei jedoch unterschiedliche Komponenten zum Einsatz kommen (siehe Abbildung).

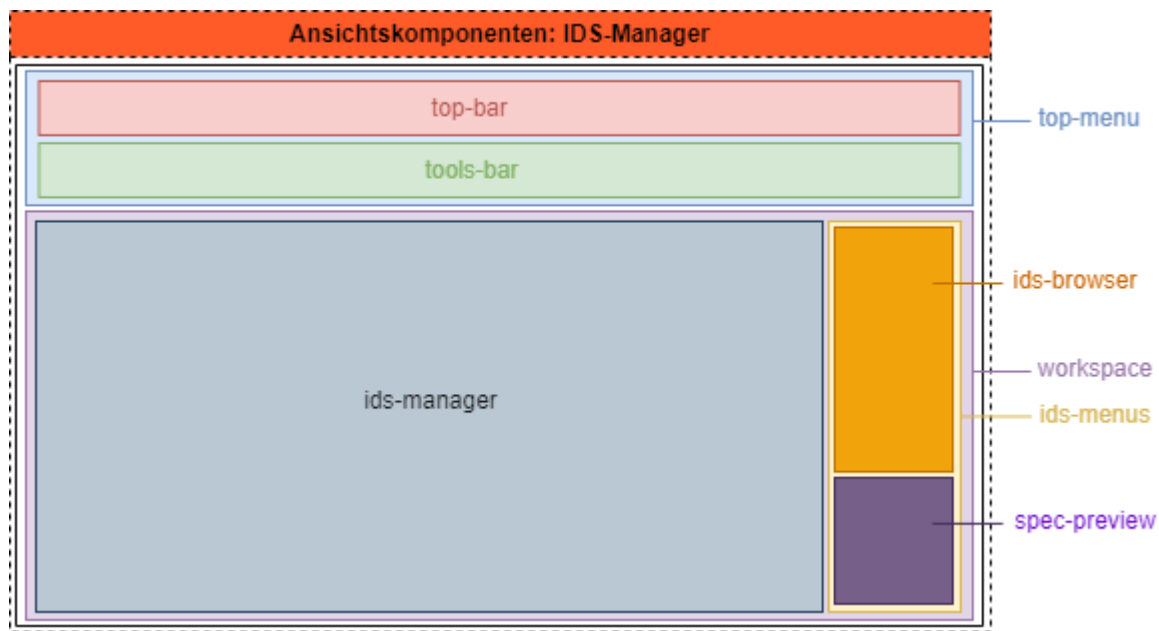


Abbildung 36. Gestaltung des IDS-Manager-Seitendesigns

Die Komponente „ids-manager“ ist für die Anzeige verantwortlich und sollte dem Benutzer die Bearbeitung des Inhalts eines IDS ermöglichen, einschließlich der Spezifikationen. Die Komponente „ids-browser“ sollte die Navigation durch die verschiedenen Spezifikationen innerhalb eines IDS erleichtern. Schließlich sollte die Komponente „ids-preview“ die Bedeutungen jeder erstellten Spezifikation präsentieren.

Bei der Erstellung einer neuen IDS-Datei wird jeder Datei eine einzigartige ID zugewiesen, und ihre Daten werden in der Variablen „IDS_Manager“ gespeichert, die wie folgt definiert ist:

```
var IDS_Manager: { [idsID: number]: IDS }
```

Codeausschnitt 1. Definition der Variable für die IDS-Manager-Datenspeicherung

Die Klasse „IDS“ wurde so definiert, dass sie eine objektorientierte Verarbeitung der Daten ermöglicht. Diese Klasse erlaubt es, alle Daten jeder IDS zu speichern und zu manipulieren, einschließlich ihrer verschiedenen Metadaten sowie der Liste der Spezifikationen, die ebenfalls durch die Klasse „Ids_specification“ definiert sind.

Zur Erweiterung der Funktionalitäten der bereits erwähnten Elemente wurden zusätzliche Funktionen namens „AddSpec“, „saveIDSitem“ und „deleteSpec“ in die Webworker-Datei integriert. „AddSpec“ dient dem Hinzufügen einer Spezifikation zu einem bestimmten IDS. „saveIDSitem“ unterstützt das Speichern des Inhalts einer spezifischen Spezifikation bzw IDS-Metadaten, und „deleteSpec“ ist für das Entfernen einer bestimmten Spezifikation aus einem IDS vorgesehen.

Der Webworker verwendet die Funktion „getIdsPageContent“, wobei „IDSid“, als Attribut der Funktion, das jeweilige IDS spezifiziert und „pageNB“, ein weiteres Attribut, entscheidet, welcher Inhalt zurückgegeben wird: Bei einem Wert von 0 wird der Metadaten-Inhalt des IDS zurückgegeben, bei einem Wert ungleich 0 der Inhalt einer zugehörigen Spezifikation.

Die Komponente „ids-value-input“ gewährleistet die korrekte Eingabe eines Parameters einer Facette mit entsprechenden zulässigen Einschränkungen.

Die Funktion „specPreviewText“ wird im Haupt-UI von der Komponente „ids-preview“ aufgerufen, um die Bedeutung jeder ausgewählten Spezifikation zu generieren.

Beim Export einer IDS-Datei kommt die Funktion „exportIDS“ zum Einsatz, die die ID des spezifischen IDS als Eingabe verwendet. Während des Durchlaufens der Attributmethoden innerhalb des IDS ruft sie die Funktion „encodeXML“ auf, um sicherzustellen, dass das Endergebnis den Anforderungen an das XML-Dateiformat entspricht.

4.3.2 Zweite Ebene: Bearbeitung einer bestehenden IDS-Datei

Um den Inhalt einer bestehenden IDS-Datei zu bearbeiten, verwendet die Anwendung die Funktionen, die bereits im Unterkapitel 4.3.1 beschrieben wurden. Zudem kommt die Funktion „readIDS“ zum Einsatz, welche die „XMLValidator“- und „XMLParser“-Methoden aus dem npm-Modul „fast-xml-parser“ nutzt, um die Korrektheit des XML-Formats der importierten Datei zu überprüfen und den XML-Inhalt in Objekte zu konvertieren.

4.3.3 Dritte Ebene: Prüfung von IDS-basierten Informationsanforderungen

Die Funktionalität der Durchführung einer IDS-basierten IFC-Validierung wird durch die „IDS-Palette“ sichergestellt. Diese wird, wie in der folgenden Abbildung dargestellt, in das Haupt-Dashboard integriert.

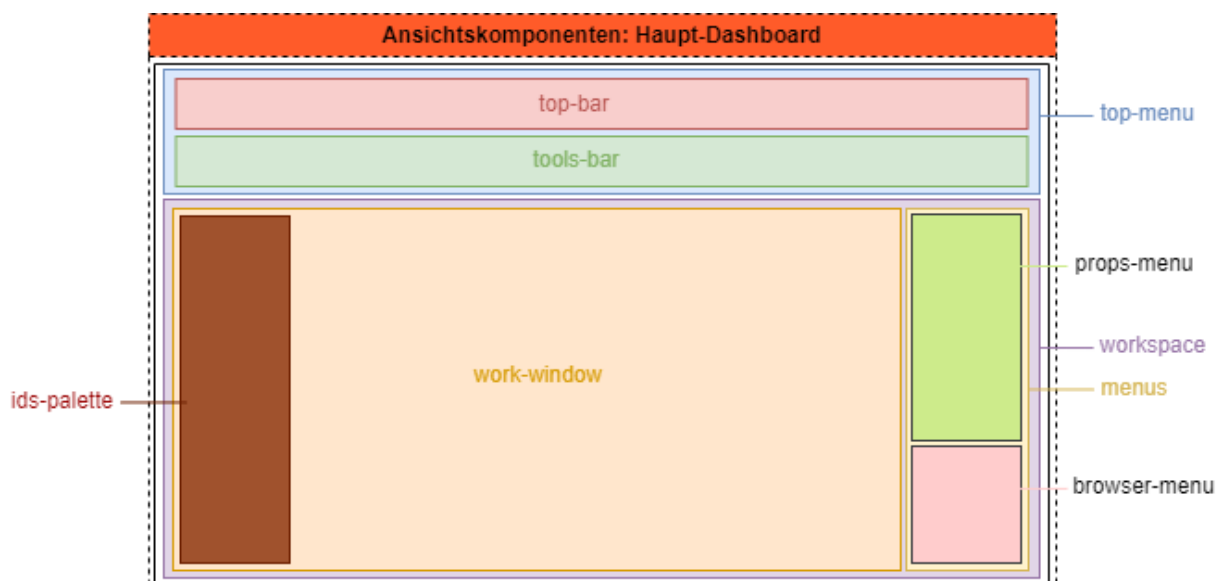


Abbildung 37. Gestaltung des Haupt-Seitendesigns

Die IDS-Palette dient dem Lesen einer IDS-Datei mittels der zuvor erläuterten „readIDS“-Funktion. Die dabei gelesene IDS-Datei wird in diesem Fall in das Objekt „IDS_Palette“ gespeichert, welches wie Folgendes definiert ist:

```
var IDS_Palette: { [idsID: number]: IDS }
```

Codeausschnitt 2. Definition der Variable für die IDS-Palette-Datenspeicherung

Beim Hochladen einer IDS-Datei, sofern eine IFC-Datei geöffnet ist, wird automatisch die Funktion „validation_Ids“ aktiviert. Sie nimmt die ID der geöffneten IFC-Datei sowie die ID der betreffenden IDS-Datei als Parameter und liefert das Ergebnis des Validierungsprozesses zurück. Zusätzlich wird diese Funktion bei jeglichen Änderungen in der IFC-Datei ausgelöst, um sicherzustellen, dass die Validierungsergebnisse kontinuierlich den aktuellen Zustand des IFC-Modells abbilden.

buildingSMART hat den Entwicklern über seine Online-IDS-Dokumentation auf „Github“ eine Liste von Testfällen zur Verfügung gestellt, die ausgeführt werden müssen, um die ordnungsgemäße Funktion des Programms zu überprüfen. Die meisten dieser Tests sind nicht mehr aktuell, einige enthalten zum Beispiel Werte für „minOccurs“ und „maxOccurs“, die nicht mehr dem aktuellen Format entsprechen. Daher wurden die meisten dieser Testdateien aktualisiert, um später getestet werden zu können.

Diese aktualisierte Testfälle sind ist verfügbar unter [Docs/Masterarbeit/Anhänge/Testcases](#)

4.3.4 Vierte Ebene: Erstellung des Berichts der etablierten Prüfung von IDS-basierten Informationsanforderungen

Ein in das Element „IDS-Palette“ integrierte Schaltfläche ermöglicht das Auslösen des Validierungsergebnisses zum Erzeugen einer PDF-Datei durch die Funktion „generateReportContent“ im Webworker. Dieser Vorgang verwendet die IDs der entsprechenden IDS- und IFC-Dateien und führt zur Erstellung des Inhalts für den PDF-Bericht. Der Bericht selbst wird in der Komponente „ids-palette-content“ gefertigt, wobei das npm-Modul „pdfmake“ zur Erzeugung der herunterzuladenden Datei verwendet wird.

5 Beispiel Workflows

Dieses Kapitel widmet sich zunächst den notwendigen Schritten zum Starten der Anwendung. Nachfolgend wird eine detaillierte Untersuchung der Umsetzung der in **Kapitel 2.3** beschriebenen Workflows vorgenommen, basierend auf der für dieses Projekt entwickelten Angular-Anwendungsoberfläche.

5.1 Installation und Start der Anwendung:

1. Installieren Sie Node v18.13

<https://nodejs.org/download/release/v18.13.0/>

2. Laden Sie den App-Ordner von: <https://git.tu-berlin.de/mysterieux20/bimfuge-masterarbeit>

Nutzen Sie dafür eine Git-Anwendung und den Befehl:

```
git clone https://git.tu-berlin.de/mysterieux20/bimfuge-masterarbeit.git
```

Anmerkung: Diese 2. Schritt ist nur dann zu berücksichtigen, wenn der Quellcode aus dem Git der TU Berlin heruntergeladen werden muss.

3. Installieren Sie Angular CLI v15.1.4 mit Visual Studio Code:

```
npm install -g @angular/cli@15.1.4
```

4. Im App-Ordner (wo sich die Datei package.json befindet), verwenden Sie den Befehl:

```
npm install
```

5. Öffnen Sie „node_modules/web-ifc-three/IFCLoader.js“ mit einem Texteditor wie Notepad++ oder Wordpad und bearbeiten Sie die Zeile 12:

von `this.modelID = IFCModel.modelIdCounter++;`

zu `this.modelID = 0;`

6. Öffnen Sie „node_modules/fast-xml-parser/src/fix.d.ts“ mit einem Texteditor wie Notepad++ oder Wordpad und bearbeiten Sie die Zeile 93:

von `parse(xmlData: string | Buffer, validationOptions?: validationOptionsOptional | boolean): any;`

zu `parse(xmlData: string ,validationOptions?: validationOptionsOptional | boolean): any;`

7. Um die Anwendung zu starten, verwenden Sie `ng serve`

Anmerkung: Falls Sie auf den Fehler "Execution policy" stoßen, führen Sie bitte PowerShell als Administrator aus und geben Sie den folgenden Befehl ein:

`Set-ExecutionPolicy RemoteSigned`

Die Anwendung sollte üblicherweise mittels eines Webbrowsers unter der Adresse `localhost:4200` erreichbar sein. Im Rahmen dieses Verfahrens kommt der Google Chrome-Browser zum Einsatz. Es ist darauf hinzuweisen, dass der 3D-Engine von IFC.js in einigen Browsern, wie beispielsweise Mozilla Firefox, möglicherweise Anzeichen von Verzögerungen aufzeigen könnte.

Nach dem Laden der Anwendung erscheint eine Seite, die fast identisch mit der im zuvor erwähnten Studienarbeitsbericht ist, ausgestattet mit einer Benutzeroberfläche.

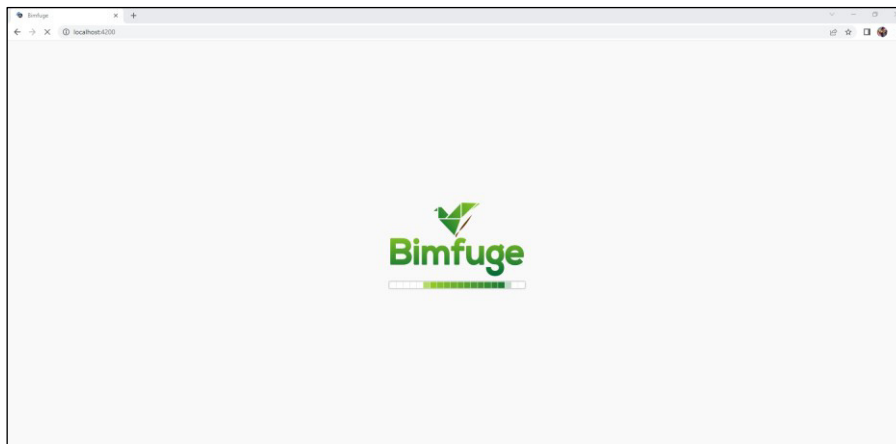


Abbildung 38. Darstellung der Vorseitenlade-Seite der Angular-Anwendung



Abbildung 39. Darstellung der Startseite der Angular-Anwendung

5.2 Erste Ebene: Erstellung einer neuen IDS-Datei

Vom Haupt-Dashboard aus beginnt der erste Schritt damit, den Mauszeiger auf das „Werkzeuge“-Menü zu setzen, wodurch der „IDS-Manager“-Schaltfläche erscheint und den Zugriff auf den IDS-Manager ermöglicht.



Abbildung 40. Zugriff auf den IDS-Manager

Anschließend wird die Option „Create a new IDS“ (Neues IDS erstellen) ausgewählt.

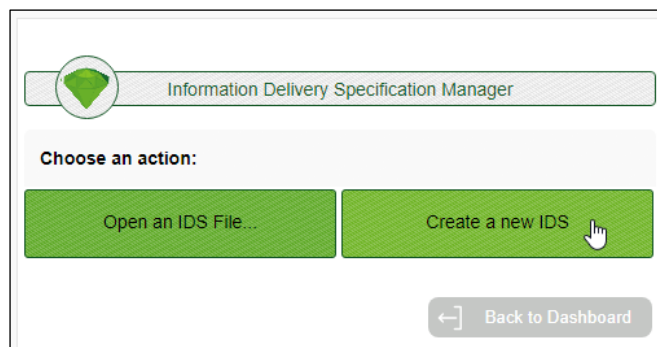


Abbildung 41. Auswahlmenü zur Erstellung eines IDS

Für diesen Schritt werden die von BIM Deutschland online gestellten Objektvorlagen [67] benutzt, die die Zuordnung der in der folgenden Tabelle aufgeführten Merkmalen zu Bauteilen des Typs „IfcWall“¹¹ und „IfcDoor“¹² verlangen:

¹¹ <https://via.bund.de/bim/assets/static-content/objekte/Bauteil-Wand.ifcxml>

¹² <https://via.bund.de/bim/assets/static-content/objekte/Bauteil-Tuer.ifcxml>

Anwendbarkeit	Merkmalsgruppe	Merkmals-Name	Merkmals-Beschreibung	Einschränkungs-Werte (Für Aufzählung Datentyp)
IfcWall	Wand	Außen	Angabe, ob dieses Bauteil ein Außenbauteil ist oder ein Innenbauteil. Als Außenbauteil grenzt es an den Außenraum (Erdreich oder Wasser).	
		Betonfestigkeitsklasse	Die Betonfestigkeitsklassen dienen zur Bestimmung der charakteristischen Festigkeit und Zuordnung zu den Festigkeitsklasse nach DIN 1045-2 Tragwerke aus Beton, Stahlbeton und Spannbeton, Teil 2: Beton.	C8/10, C12/15, C16/20, C20/25, C25/30, C30/37, C35/45, C40/50, C45/55, C50/60
		Bewehrungsstahlsorte	Zum Einsatz kommt in Deutschland Betonstahl mit einer charakteristischen Streckgrenze von 500 N/mm ² zur Anwendung. Die erforderlichen Eigenschaften sind beispielsweise in der DIN 488 oder in der Europäischen Norm EN 10080 geregelt.	B500A, B500B
		Feuerwiderstandsklasse F	Die Feuerwiderstandsklasse F gilt für Wände, Decken, Stützen, Unterzüge und Treppen	F30, F60, F90, F120, F180
		Hauptbaustoff	Der Hauptbaustoff beschreibt das Hauptbaumaterial eines Bauteils bzw. Bauwerks.	
		Kostengruppe DIN276		
		Tragend		
IfcDoor	Tuer	Außen	Angabe, ob dieses Bauteil ein Außenbauteil ist oder ein Innenbauteil. Als Außenbauteil grenzt es an den Außenraum (Erdreich oder Wasser).	
		Feuerwiderstandsklasse T	Feuerwiderstandsklasse T gilt für Türen, Tore und Klappen	F30, F60, F90, F120, F180
		Hauptbaustoff	Der Hauptbaustoff beschreibt das Hauptbaumaterial eines Bauteils bzw. Bauwerks.	
		Kostengruppe DIN276		

Tabelle 9. Objektvorlagen Anforderungen von BIM Deutschland (für „IfcWall“ und „IfcDoor“)

Nachdem der Name der zu erstellenden IDS-Datei eingegeben wurde, führt ein einfacher Klick auf „Create“ zur nächsten Seite, die Felder zum Ausfüllen der Metadaten der IDS-Datei enthält (Siehe Abbildung).

Abbildung 42. Formular zur Eingabe von Metadaten-Informationen eines IDS

Die „+“-Schaltfläche auf der linken Seite ermöglicht das Hinzufügen einer neuen Spezifikation zur IDS-Datei (siehe Abbildung).

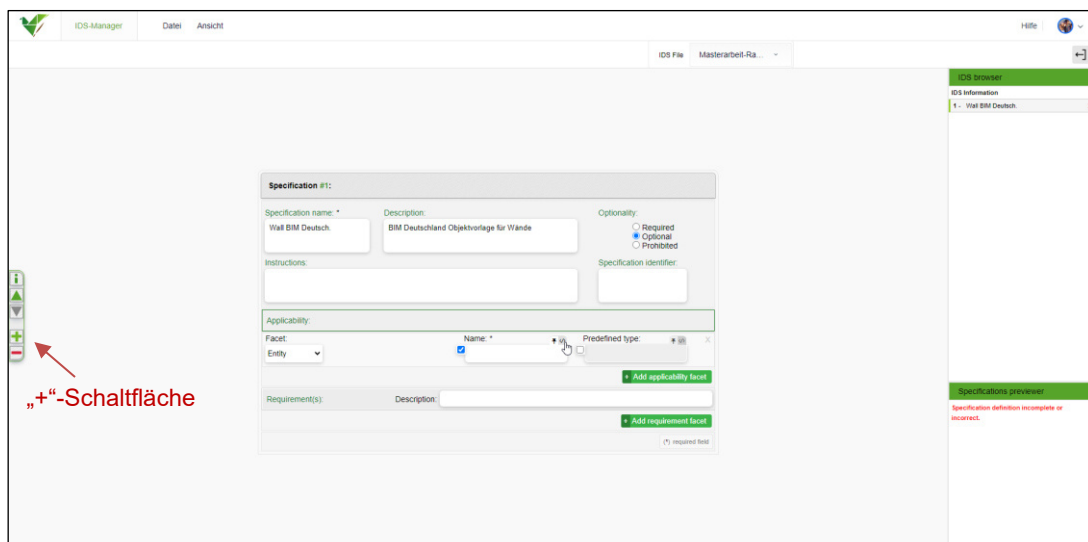


Abbildung 43. Standort der '+-Schaltfläche im IDS-Manager

Für die Eingabe der Daten der ersten Spezifikation ist zu erwähnen, dass die Entität „IfcWall“, gemäß dem verwendeten Standard IFC4 ADD2 [68], über zwei Unterklassen verfügt, die bei der Auswahl der Elemente, die für die Anwendbarkeit relevant sind, berücksichtigt werden müssen.

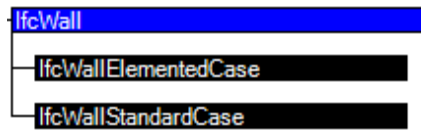


Abbildung 44. Unterklassen von "IfcWall" nach IFC4

Für die Erfassung der relevanten Entitäten wird eine Facette vom Typ „Entity“ (Entität) unter dem Feld „Applicability“ (Anwendbarkeit) erstellt.

Für jedes Textfeld, das eine Einschränkung erlaubt, wurde eine Komponente erstellt, die es dem Benutzer ermöglicht, zwischen der Art der einzugebenden Information zu wählen: Einfacher Wert oder Einschränkung.



Abbildung 45. Gestaltung eines Feldes mit Einschränkungsmöglichkeit

Beim Eintragen des Feldes, das dem Attribut „Name“ entspricht, hat man durch einen Klick auf den „</>“-Button die Möglichkeit, eine unbegrenzte Anzahl von Einschränkungen hinzuzufügen. In diesem Fall würde eine Einschränkung vom Typ „Aufzählung“ (Enumeration) hinzugefügt, in der die drei verschiedenen möglichen Entitätsnamen eines Elements vom Typ „Wand“ aufgeführt sind; „IFCWALL“, „IFCWALLELEMENTEDCASE“ und „IFCWALLSTANDARDCASE“. Die Namen der IFC-Entitäten müssen immer in Großbuchstaben eingeführt werden [69].

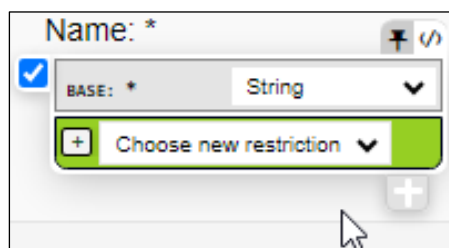


Abbildung 46. Auswahl für hinzuzufügende Einschränkungen

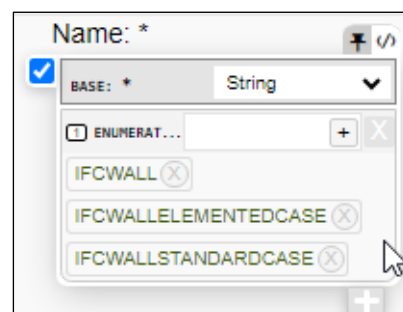


Abbildung 47. Mögliche Entitätsnamen für „Wand“-Elemente per Aufzählung (Enumeration)

Die Facetten für die entsprechenden „Wand“-Elemente werden dann unter dem Feld „Anforderungen“ (Requirements) hinzugefügt, basierend auf der zuvor erwähnten Tabelle. Die nachstehende Abbildung illustriert exemplarisch die Merkmalsfacette für das Attribut „Feuerwiderstandsklasse F“ bei Wand-Bauteilen.

The screenshot shows a configuration window for a facet. It includes the following fields and options:

- Facet:** A dropdown menu set to 'Property'.
- Instructions:** A text box containing the instruction: 'Die Feuerwiderstandsklasse F gilt für Wände, Decken, Stützen, Unterzüge und Treppen'.
- Property set: ***: A dropdown menu set to 'Wand'.
- Property name: ***: A dropdown menu set to 'Feuerwiderstandsklasse F'.
- Property value:** A dropdown menu set to 'String'. Below it is an 'ENUMERAT...' section with a list of values: F30, F60, F90, F120, and F180, each with a small 'x' icon next to it.
- Datatype: ***: A dropdown menu set to 'IfcIdentifier'.
- URI:**: An empty text input field.
- Optionality:** Three radio buttons: 'Required' (selected), 'Optional', and 'Prohibited'.

Abbildung 48. Merkmalsfacette für das Attribut „Feuerwiderstandsklasse F“ bei Wand-Bauteilen

Eine Beschreibung der definierten Spezifikationsfunktion wird in Echtzeit rechts im kleinen Fenster „Specifications previewer“ (Spezifikationenvorschau) angezeigt (Siehe Abbildung).

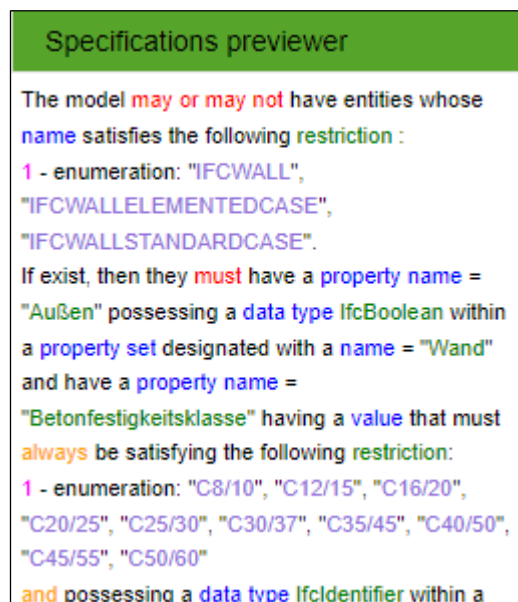


Abbildung 49. Echtzeitansicht der Spezifikationsfunktionalität

Abschließend kann die neu erstellte IDS-Datei durch Auswahl des Menüs „Datei → Export IDS“ heruntergeladen werden

Die generierte Datei ist verfügbar unter EXPORT->Masterarbeit -> Masterarbeit-Rami_Bali.ids

5.3 Zweite Ebene: Bearbeitung einer bestehenden IDS-Datei

In diesem spezifischen Szenario genügt es, die Funktion „Open an IDS File...“ (Eine IDS-Datei öffnen...) auszuwählen (Siehe Abbildung), welche wieder über das Hauptmenü unter „Datei → Neu“ erreichbar ist. Nach der Auswahl öffnet sich ein neues Fenster, in dem der Benutzer die auszuführende IDS-Datei selektieren kann.

An diesem Punkt und am Beispiel wird vorgegangen, die im vorherigen Schritt erstellte Datei zu öffnen und ihr eine neue Spezifikation für Elemente des Typs „Tür“ hinzuzufügen.

Im Gegensatz zur vorherigen Spezifikation für die „Wände“, bei der alle Merkmalsfacetten als „erforderlich“ definiert wurden, wird in der neuen Spezifikation eine Ausnahme gemacht: Die „Kostengruppe DIN276“ wird als „optional“ betrachtet; Dieser spezifische Unterschied ist für einen nachfolgenden Test vorgesehen, der im nächsten Unterkapitel eine Überprüfung unterzogen wird.

Die nachfolgende Abbildung veranschaulicht beispielhaft die Merkmalsfacette für das Attribut „Kostengruppe DIN276“ spezifisch für Türelemente.


The screenshot shows a software window titled 'Facet: Property' with a dropdown menu. The 'Instructions' field is empty. The 'Datatype: *' is set to 'IfoIdentifier'. The 'Property set: *' is set to 'Tuer' with a checked checkbox. The 'Property value:' field is empty. The 'Property name: *' is set to 'Kostengruppe DIN276' with a checked checkbox. The 'Optionality' section has three radio buttons: 'Required' (unselected), 'Optional' (selected), and 'Prohibited' (unselected). The 'URI:' field is empty.

Abbildung 50. Definition der Merkmalsfacette für das Attribut „Kostengruppe DIN276“ spezifisch für Türelemente

Die erhaltene IDS-Datei wird exportiert (analog zum Export im vorherigen **Unterkapitel 5.2**) mit insgesamt zwei Spezifikationen: Eine für die Wände und die zweite für die Türen.

Die generierte Datei ist verfügbar unter **EXPORT->Masterarbeit -> Masterarbeit-Rami_Bali-2.ids**

5.4 Dritte Ebene: Prüfung von IDS-basierten Informationsanforderungen

Über die oben rechts auf der Seite positionierte Schaltfläche  kann der IDS-Manager verlassen und zum Haupt-Dashboard zurückgekehrt werden.

Für die Zwecke dieser Demonstration wird die Datei „racbasicsampleproject.ifc“ verwendet, die ursprünglich im Revit-Format von der offiziellen Autodesk-Webseite [70] heruntergeladen wurde.

Aufgrund der Tatsache, dass die Angular-Anwendung in einem vorherigen Studienarbeit entwickelt wurde, um ausschließlich mit IFC4-Dateien kompatibel zu sein, erfolgte anschließend die notwendige Konvertierung der Datei in das IFC4-Format. Diese Konvertierung wurde mittels der Revit-Anwendung durchgeführt. Diese Anpassung gewährleistet eine effektive Integration der Datei in die Anwendungsumgebung. Die konvertierte Datei ist unter dem Verzeichnis „src/assets/ifc-models/racbasicsampleproject.ifc“ zugänglich. Sie ist in der Benutzeroberfläche der Anwendung referenziert und kann folgendermaßen aufgerufen werden:

Wählen Sie „Datei“ → „Öffne IFC-Datei...“ → „aus Modellen“ → „racbasicsampleproject.ifc“ → „Modell öffnen“ (Siehe Abbildung)

Standardmäßig ist die IDS-Palette deaktiviert. Für ihre Aktivierung kann im oberen Menü der Seite über „Ansicht“ → „IDS-Palette“ navigiert werden.

Die erste Schaltfläche links in der IDS-Palette dient dazu, eine IDS-Datei zu importieren und die IDS-basierte Validierung der zuvor geöffneten IFC-Datei zu initiieren (Siehe Abbildung).



Abbildung 52. Fenster zum Öffnen der Datei *racbasicsampleproject.ifc* innerhalb der Angular-Anwendung



Abbildung 51. Öffnen einer IDS-Datei in der IDS-Palette

Für diesen Prozess wird die im Unterkapitel „5.3 Zweite Ebene: Bearbeitung einer bestehenden IDS-Datei“ erstellte IDS-Datei verwendet.

Nach der Einführung der für die Validierung vorgesehenen IDS-Datei beginnt die IDS-basierte Validierung der IFC-Datei automatisch.

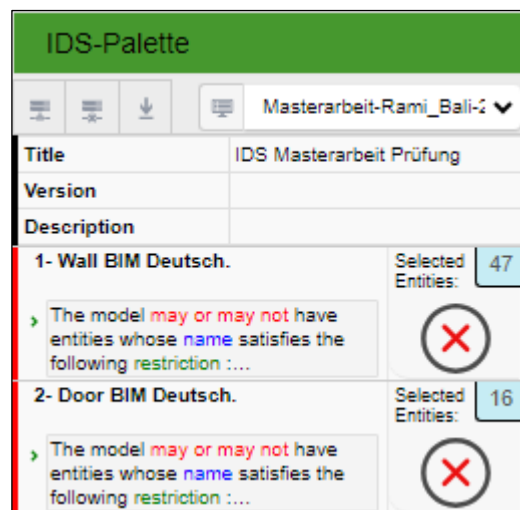


Abbildung 53. Ergebnis der durchgeführten IDS-basierten Validierung

Die Validierung kann je nach der Größe der betreffenden IFC und IDS-Datei einige Sekunden in Anspruch nehmen, bevor das Ergebnis sichtbar wird. Vor jeder Spezifikation erscheint eines

von zwei möglichen Symbolen. Die beiden möglichen Symbole sowie ihre Bedeutungen sind in der folgenden Tabelle aufgelistet.



Symbol	Bedeutung
	Die IFC-Datei erfüllt vollständig die durch die IDS-Spezifikation festgelegten Anforderungen.
	Die IFC-Datei erfüllt die durch die IDS-Spezifikation festgelegten Anforderungen nur teilweise / nicht vollständig.

Tabelle 10. Bedeutung der für die Validierung benutzte Symbole

Vor jeder Spezifikation ist ein kleiner Bereich vorgesehen, um die Anzahl der Entitäten anzuzeigen, die die Bedingungen für die Anwendbarkeit der betreffenden Spezifikation erfüllen. Die Hintergrundfarbe hinter der Anzahl der ausgewählten Elemente ist so gestaltet, dass sie die Notwendigkeit der jeweiligen Spezifikation wie folgt anzeigt:

Hintergrundfarbe	Bedeutung
Rot	symbolisiert die Natur der Spezifikation als „ Verbot “, was darauf hinweist, dass bestimmte Elemente / Kriterien strikt untersagt sind.
Grün	symbolisiert die Natur der Spezifikation als „ Erforderlich “, was darauf hinweist, dass bestimmte Elemente / Kriterien unbedingt einzuhalten sind.
Blau	symbolisiert die Natur der Spezifikation als „ Optional “, was darauf hinweist, dass bestimmte Elemente / Kriterien nach Ermessen einbezogen oder ausgelassen werden können

Tabelle 11. Bedeutung der Anzahl der ausgewählte Elemente Hintergrundfarbe

Beim Anklicken einer beliebigen Spezifikation entfaltet sich das Design, um weitere Details zur Spezifikation zu enthüllen. Dem Nutzer wird dadurch die Funktionalität der Spezifikation sowie die betroffenen Entitäten ersichtlich gemacht. Ebenso wird ein detaillierteres Ergebnis des Validierungsprozesses dargestellt.

In diesem Beispiel wird das detailliertere Ergebnis betrachtet, das durch die zweite Spezifikation für „Tür“-Elemente angezeigt wird. Die beiden folgenden Schnappschüsse können wie folgt beschrieben werden:

Im ersten Schnappschuss wird eine Liste der Anforderungen präsentiert, die von der Spezifikation festgelegt wurden und entweder obligatorisch oder optional zu erfüllen sind. Diese Anforderungen sind nummeriert und werden in einem 1-zu-1-Verhältnis in einer Tabelle angeordnet. Für jede Anforderung wird von dem Algorithmus in kontrolliert zufällige Weise eine unterschiedliche Farbe generiert. Es ist jedoch wichtig zu beachten, dass diese generierten

Farben für die Anforderungen nicht mit den früher erwähnten Farben, die die Notwendigkeit der Spezifikationen symbolisieren, verwechselt werden sollten. Die hier genutzten Farben dienen ausschließlich der Differenzierung der einzelnen Anforderungen.

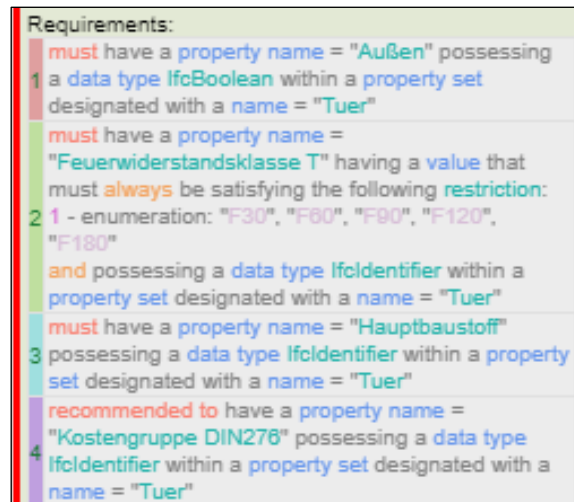


Abbildung 54. Anzeige der Liste der Anforderungen einer Spezifikation in der IDS-Palette.

Im zweiten Schnappschuss wird eine Tabelle dargestellt, in der jedem Express-Id (Entität) die verschiedenen von der Spezifikation geforderten Anforderungen zugeordnet sind, zusammen mit dem entsprechenden Ergebnis. Jedes Express-Id wird als klickbar dargestellt, was die Auswahl der betreffenden Entität im 3D-Engine bzw. Anwendung erlaubt. Zusätzlich können die dieser Entität zugeordneten Informationen und Eigenschaften eingesehen werden.

Validation result:		
Entities	Req.	Result
#9831, #9957, #28044, #28081,	1	✗
#29096, #103085, #104166, #104183,	2	✗
#10419, #11087, #12921, #12970,	3	✗
#12985, #12999, #13274, #18356	4	✗

Abbildung 55. Darstellung der Ergebnisse nach Express-IDs in einer Tabelle

Durch diese Art der Interaktiven Darstellung wird eine klare und strukturierte Übersicht über die Anforderungen und deren Erfüllung ermöglicht, was für die Genauigkeit und Effizienz im Spezifikationsmanagement wesentlich ist.

Bei diesem Beispiel der zu den „Tür“-Elementen gehörenden Ergebnisse ist es auffällig, dass laut der zwei vorherigen Illustrationen sämtliche gewählten Elemente die festgelegten Anforderungen, ob obligatorisch oder optional, der Spezifikation nicht erfüllen. Ein auffälliger

Unterschied ist jedoch auch zu erkennen: Die ersten drei roten Kreuze bedeuten, dass die Nichterfüllung der betreffenden Anforderung verboten ist, während das letzte blaue Kreuz anzeigt, dass die entsprechende Anforderung zwar nicht erfüllt wurde, aber dennoch als optional betrachtet wird.

Daher fördert diese Präsentation eine umfassende Analyse und Bewertung der Spezifikation, was für eine präzise und effiziente Arbeitsweise im Bauwesen von großer Bedeutung ist.

Jetzt wird der Fokus auf die Behebung der bei der zweiten Spezifikation in Bezug auf die Bauteile „IfcDoor“ aufgetretenen Fehler gelegt. Dabei wird angenommen, dass die betroffenen Elemente in verschiedene Familien kategorisiert sind, was im Bauwesen ein häufig anzutreffender Fall sein könnte. In diesem Szenario werden drei verschiedene Familien berücksichtigt, von denen jede ihren eigenen einzigartigen Merkmalen aufweist, wie in der folgenden Tabelle dargestellt:

Familie	Entitäten („IfcDoor“)	Merkmals-Werte (Merkmalsgruppe: „Tuer“)			
		„Außen“	„Feuerwiderstandsklasse T“	„Hauptbaustoff“	„Kostengruppe DIN276“
Türen Familie 1	#9957, #28044, #28081, #29096, #10419, #11087	TRUE	F30	Holz	300
Türen Familie 2	#9831, #103085, #104166, #104183, #12921, #12970	FALSE	F60	Glas	300
Türen Familie 3	#12985, #12999, #13274, #18356	FALSE	F30	Holz	(Kein Wert)

Tabelle 12. Anzeige der Werte der für dieses Beispiel verwendeten Entitätsfamilien

Da die Merkmalsdefinition für unterschiedliche Entitätsfamilien, bei denen einen gemeinsamen Typ von Elementen „IfcDoor“, benötigt wird, empfiehlt es sich, eine Vorlage für die zu erstellenden Merkmalsgruppen anzulegen. Dies verhindert, dass Benutzer die zu einzugebenden Daten für jede Familie innerhalb der Merkmalsgruppe wiederholt neu definieren müssen. Für diesen Zweck kann die Option „Bearbeiten“ → „Selbstdefiniertes Template erstellen“ verwendet werden, um Namen und Beschreibungen für die erforderlichen Merkmalsgruppen sowie die zugehörigen Merkmalsdefinitionen effektiv zu erfassen.

Selbstdefiniertes Template erstellen

Eine neue Eigenschaften-Set-Vorlage erstellen:

Name: Tuer

Beschreibung: BIM Deutschland Objektvorlage für Türen

Eigenschaft hinzufügen

Außen	Angabe, ob dieses Bauteil	IfcPropertySingleValue	IfcBoolean	Keine Einheit	Einheit hinzufügen	Eigenschaft löschen
Feuerwiderstandsklasse T	Feuerwiderstandsklasse T	IfcPropertySingleValue	IfcIdentifier	Keine Einheit	Einheit hinzufügen	Eigenschaft löschen
Hauptbaustoff	Der Hauptbaustoff beschrei	IfcPropertySingleValue	IfcIdentifier	Keine Einheit	Einheit hinzufügen	Eigenschaft löschen
Kostengruppe DIN276	Beschreibung 4	IfcPropertySingleValue	IfcIdentifier	Keine Einheit	Einheit hinzufügen	Eigenschaft löschen

← Zurück Eigenschaftens-Set-Vorlage speichern

* Pflichtfeld

Abbildung 56. Definition der Merkmalsgruppenvorlage für Türen

Durch einen Klick auf „Eigenschaften-Set-Vorlage speichern“ wird die neu definierte Vorlage in das IFC-Modell eingefügt, und dem Benutzer eine Bestätigungsnachricht übermittelt. Anschließend erfolgt die Navigation zum oberen Menüpunkt „Bearbeiten“ → „Eigenschaftensatz-Manager“ → „Neuer Eigenschaftensatz“. In dem Dropdown-Feld „Eigenschaftensatzdefinition“ wird die Option „Mit Vorlage“ ausgewählt, und im anschließenden Feld wird die zuvor eingeführte Vorlage gewählt. Auf diese Weise wird eine Zusammenfassung dieser Vorlage angezeigt (Siehe Abbildung).

Eigenschaftensatz-Manager

Neuen Eigenschaftensatz erstellen:

Eigenschaftensatzdefinition: Mit Vorlage

Vorlage: Tuer (#125407)

Express Id: #125407

Name: Tuer

Beschreibung: BIM Deutschland Objektvorlage für Türen

Name	Beschreibung	Typ	Maßeinheitentyp	Einheit
Außen	Angabe, ob dieses Bauteil ein Außenbauteil ist oder ein Innenbauteil. Als Außenbauteil grenzt es an den Außenraum (Erdreich oder Wasser).	IfcPropertySingleValue	IfcBoolean	
Feuerwiderstandsklasse T	Feuerwiderstandsklasse T gilt für Türen, Tore und Klappen	IfcPropertySingleValue	IfcIdentifier	
Hauptbaustoff	Der Hauptbaustoff beschreibt das Hauptbaumaterial eines Bauteils bzw. Bauwerks.	IfcPropertySingleValue	IfcIdentifier	
Kostengruppe DIN276	Beschreibung 4	IfcPropertySingleValue	IfcIdentifier	

← Zurück Weiter

Abbildung 57. Anzeige der Daten der zu verwendenden Merkmalsgruppenvorlage

Nach Betätigung der Schaltfläche „Weiter“ wird die entsprechende Seite dargestellt, auf der die Eingabe der Merkmalswerte für die erste Merkmalsgruppe möglich ist. Hiermit wird ein Überblick über die nachfolgenden Schritte gegeben: ein Schnappschuss zur Erfassung der Merkmalswerte der „Türen Familie 1“:

Eigenschaftensatz-Manager

Neuen Eigenschaftensatz erstellen (Mit Vorlage):

Name: Tür

Beschreibung: BIM Deutschland Objektvorlage für Türen

Außen	Angabe, ob dieses Bauteil €	IfcPropertySingleValue	IfcBoolean	TRUE	Keine Einheit	Einheit hinzufügen
Feuerwiderstandsklasse T	Feuerwiderstandsklasse T €	IfcPropertySingleValue	IfcIdentifier	F30	Keine Einheit	Einheit hinzufügen
Hauptbaustoff	Der Hauptbaustoff beschrei	IfcPropertySingleValue	IfcIdentifier	Holz	Keine Einheit	Einheit hinzufügen
Kostengruppe DIN276	Beschreibung 4	IfcPropertySingleValue	IfcIdentifier	300	Keine Einheit	Einheit hinzufügen

« Zurück Eigenschaftensatz speichern

* Pflichtfeld

Abbildung 58. Erfassung der Merkmalswerte der „Türen Familie 1“

Nachdem „Eigenschaftensatz speichern“ betätigt wurde, wird die neue Merkmalsgruppe generiert und dem IFC-Modell hinzugefügt, erkennbar an einer Benachrichtigung der Anwendung.

Es folgt dann die Erstellung der Familie, durch welche die neu erstellte Merkmalsgruppe allen Entitäten, die zur „Türen Familie 1“ gehören, zugewiesen werden können: Durch das Anklicken des oberen Menüs und das Befolgen der Klickfolge: „Bearbeiten“ → „Familien Manager“ → „Neue Familie“, erfolgt unmittelbar die Erfassung des Namens und Typs der Familie („IfcDoorType“), wodurch die Erstellung der neuen Familie vollzogen wird.

Die Zuordnung der Merkmalsgruppe erfolgt durch die Auswahl der gerade neu erstellten Familie im Familien-Manager, das Auswählen des Tabs „Eigenschaftensätze“ und anschließendes Klicken auf die numerische Schaltfläche „Eigenschaftensatz hinzufügen/entfernen“. Daraufhin erscheint eine Liste aller Merkmalsgruppen, die in der IFC-Datei enthalten sind. Da die neu erstellten Merkmalsgruppen immer am Ende der Liste positioniert sind, muss man zur letzten Seite der existierenden Merkmalsgruppenliste navigieren, um die betreffende Merkmalsgruppe (die gerade erstellt wurde) zu finden und sie mit einem einfachen Klick auf den mit einem Pfeil nach links dargestellte Schaltfläche der betreffenden Familie hinzuzufügen.

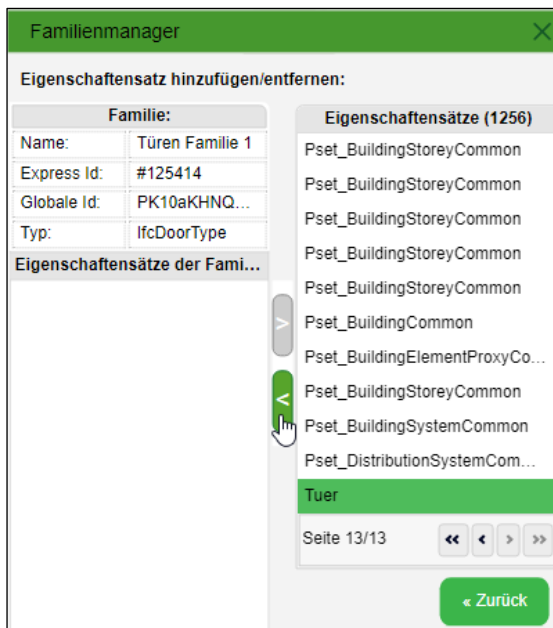


Abbildung 60. Hinzufügen der Merkmalsgruppe zur Familie „Türen Familie 1“

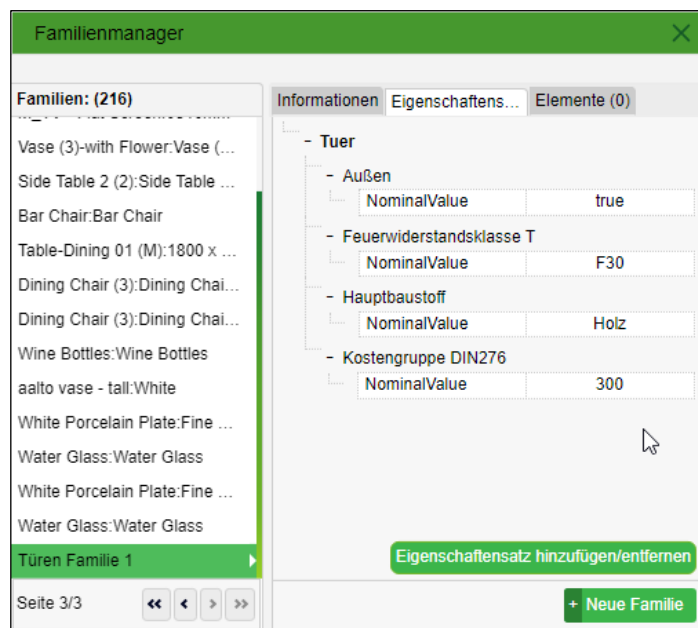


Abbildung 59. Anzeige der mit der Familie „Türen Familie 1“ verknüpften Merkmalen

Der Prozess der Zuweisung der Bauteile zu der Familie ist fast analog zu dem zuvor beschriebenen für das Hinzufügen der Merkmalsgruppe. Diesmal jedoch muss man von der Seite der betreffenden Familie aus zum Tab „Elemente“ oben navigieren, auf die Schaltfläche „Elemente hinzufügen/entfernen“ klicken, um dort eine Liste aller existierenden Elemente vom Typ „IfcDoor“ zu finden und die Elemente auszuwählen, die zur „Türen Familie 1“ gehören.

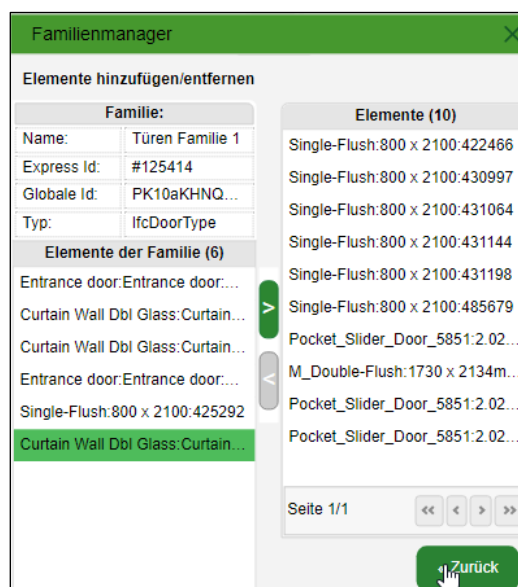


Abbildung 61. Fenster zum Hinzufügen von Elementen zur Familie

Die Anwendung startet den Validierungsprozess nach jeder Aktion des Benutzers, die eine Änderung der Struktur des IFC-Modells verursacht, und führt einen Vergleich des erhaltenen Ergebnisses mit dem bereits vorhandenen Inhalt durch: Sollte das erhaltene Ergebnis unterschiedlich sein, wird der Inhalt der IDS-Palette aktualisiert, um sicherzustellen, dass der Status der Validierung auf dem neuesten Stand angezeigt wird. Auf dieser Ebene ist es bereits möglich, eine Veränderung in den Ergebnissen zu sehen, die in der IDS-Palette angezeigt werden:

Validation result:		
Entities	Req.	Result
#9831, #12921, #12970, #12985, #12999, #13274, #18356, #103085, #104166, #104183	1	✗
	2	✗
	3	✗
	4	✗
#9957, #10419, #11087, #28044, #28081, #29096	1	✓
	2	✓
	3	✓
	4	✓

Abbildung 62. Anzeige der Ergebnisse nach Änderungen in Spezifikation 2

Der gleiche Prozess wird dann für die Elemente der beiden verbleibenden Familien unter Verwendung derselben Merkmalsgruppenvorlage durchgeführt. Am Ende wird das Ergebnis der zweiten Spezifikation in der IDS-Palette wie folgt dargestellt:

Validation result:		
Entities	Req.	Result
#9831, #9957, #104166, #104183, #10419, #11087, #12921, #12970, #28044, #28081, #29096, #103085	1	✓
	2	✓
	3	✓
	4	✓
#12985, #12999, #13274, #18356	1	✓
	2	✓
	3	✓
	4	✗

Abbildung 64. Ergebnisse in Spezifikation 2 nach Einführung der verbleibenden Familien

1- Wall BIM Deutsch.		Selected Entities: 47
> The model may or may not have entities whose name satisfies the following restriction :...		✗
2- Door BIM Deutsch.		Selected Entities: 16
> The model may or may not have entities whose name satisfies the following restriction :...		✓

Abbildung 63. Ergebnisse in der IDS-Palette nach Einführung der verbleibenden Familien

Das IFC-Datei wird dann als gültig im Hinblick auf die zweite Spezifikation betrachtet, obwohl einige Elemente nicht einer der Anforderungen entsprechen, die als optional (Empfehlung) definiert wurde; Und genau deshalb beeinflusst dies das Endergebnis der Validierung nicht.

Die aktualisierte IFC-Datei ist verfügbar unter **EXPORT -> Masterarbeit -> racbasicsampleproject (2) (1).ifc**

5.5 Vierte Ebene: Erstellung des Berichts der etablierten Prüfung von IDS-basierten Informationsanforderungen

Um den Vorgang zur Erstellung eines IDS-basierten Prüfberichts für eine IFC-Datei zu starten, wird einfach auf die dritte Schaltfläche von links („Download IDS report“) geklickt. Die eigentliche Erstellung dauert nur wenige Sekunden, bevor sich ein neues Fenster öffnet, in dem der Speicherort für die gerade erstellte Datei vom Benutzer ausgewählt werden kann. Eine Kopie der in diesem Beispiel generierten PDF-Datei ist im Anhang A1 enthalten.

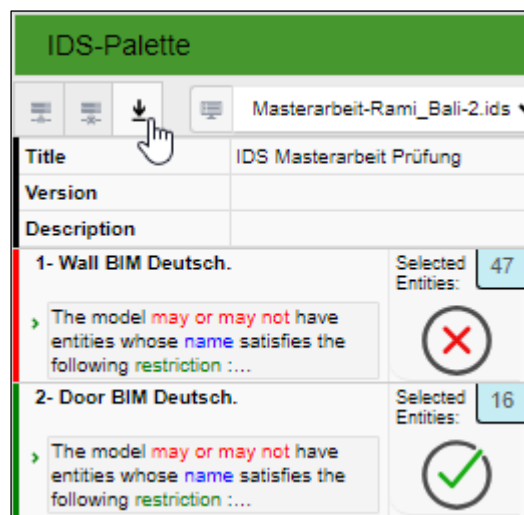


Abbildung 65. Position der Schaltfläche zum Herunterladen des PDF-Berichts

6 Fazit

In dieser Masterarbeit wurde erfolgreich eine Web-Applikation entwickelt, beschrieben und getestet, die nicht nur die Erzeugung von Property Sets und Templates ermöglicht, sondern auch die Beschreibung von Anforderungen mittels IDS und die Überprüfung realer IFC-Modelle unterstützt. Diese Innovation repräsentiert einen erstmaligen, direkt implementierten Zusammenhang zwischen maschinenlesbaren Anforderungen und der IFC-basierten Definition von Typen, Merkmalen und Eigenschaftsvorlagen. Dieser Fortschritt ist von entscheidender Bedeutung, um anforderungskonforme Modellierungen im Kontext von Open BIM zu ermöglichen und trägt maßgeblich zur Verbesserung der Planungs- und Validierungsprozesse bei.

Basierend auf den Grundlagen des zweiten Kapitels wurden aktuelle Implementierungen eines IDS-basierten Workflows in Kapitel drei untersucht. Dabei wurde die Integration von IDS in aktuellen, kostenpflichtigen sowie open-source Software Werkzeugen untersucht und die Ergebnisse im Konzept der zu erstellenden Implementierung berücksichtigt.

Im vierten Kapitel wurde umfangreich beschrieben, von welchem Ausgangspunkt aus der Implementierung dieser Masterarbeit begonnen hat und welche Funktionalitäten und Anforderungen im Zuge der Arbeit umgesetzt wurden. Dies wurde mit Implementierungsdetails und der beschriebenen Struktur der vier Ebenen, Erstellung, Bearbeitung, Validierung und Report, strukturiert umgesetzt. Der Beispiel-Workflow im fünften Kapitel zeigt umfassend, welche Bearbeitungsmöglichkeiten das Werkzeug bietet und wie diese für den Nutzer umgesetzt wurden. Dabei wurden Anforderungen und Ergebnisse der Validierung sowohl graphisch als auch umfassend in Textform ausgegeben, was eine übersichtliche Beurteilung der Modelqualität ermöglicht.

Die Implementierung der Web-Applikation markiert einen wichtigen Schritt in Richtung eines effizienteren und transparenteren Planungsprozesses in der Bauindustrie. Durch die Fähigkeit, IDS-basierte Anforderungen zu erzeugen, verarbeiten und zu validieren, leistet die open-source Applikation einen wesentlichen Beitrag zur Überwindung der bestehenden Herausforderungen in der Informationsverarbeitung und -überprüfung, wozu häufig umfassende Lizenzen von Modellierungs- und Prüfprogrammen notwendig sind.

6.1 Grenzen der Implementierung

Trotz der erfolgreichen Implementierung des Ansatzes gibt es dennoch Grenzen, die berücksichtigt werden müssen. Die aktuelle Version der Web-Applikation konzentriert sich ausschließlich auf die semantische Ebene und lässt geometrische und topologische Aspekte außer Acht. Zudem sind die Interoperabilität und die Integration in bestehende Systeme und Workflows kontinuierlich zu verbessern, um eine breitere Anwendbarkeit und Benutzerakzeptanz zu erreichen.

Der Bericht (siehe Anhang) wird aktuell noch als PDF geniert, nicht im BCF Format. Hierbei gibt es noch deutliche Potentiale für eine bessere Übersichtlichkeit und Darstellung der Ergebnisse in Kombination mit dem 3D Modell.

Der webworker, welcher in der Implementierung genutzt wird, verfügt nur über begrenzte Leistung und ist nicht parallelisiert, wodurch die Anwendung teilweise langsamer reagiert als eine vergleichbare Desktop Anwendung.

Eine direkte Schnittstelle zu Eigenschaften des bSDD oder anderer Herstellerdatenbanken würde eine enorme Verbesserung des Workflows bei der Erstellung von Property Templates und IDS Anforderungen ergeben. Gleiches gilt für das BIM Portal und dessen Merkmale, welche von BIM Deutschland zur Verfügung gestellt werden.

6.2 Ausblick

Für die Zukunft ergibt sich aus dieser Arbeit ein vielversprechender Forschungsansatz. Es ist notwendig, die Applikation weiterzuentwickeln, um eine umfassendere Abdeckung aller relevanten BIM-Aspekte durch Anforderungen zu gewährleisten. Insbesondere die Integration geometrischer und topologischer Daten könnte die Funktionalität und Anwendungsbreite signifikant erweitern. Ebenso ist die kontinuierliche Anpassung und Aktualisierung der Software an neue Standards und Entwicklungen im BIM-Sektor entscheidend, um die Relevanz und Wirksamkeit der Applikation zu gewährleisten. Da IDS ein relativ junger Standard ist, welcher sich noch in der Entwicklung befindet, werden hierbei häufig Erweiterungen und Anpassungen vorgenommen, welche integriert werden sollten.

Weitere Arbeiten sollten darauf abzielen, eine noch direktere und automatisierte Methode zur Generierung von Typen und Property Set Templates aus IDS und umgekehrt zu entwickeln. Dies würde die Effizienz und Genauigkeit der Modellierungsprozesse erheblich steigern und die Anpassungsfähigkeit an unterschiedliche Projektanforderungen verbessern.

Ein weiterer wichtiger Forschungsbereich ist die Weiterentwicklung einer API, die eine reibungslose Kombination der Web-Applikation mit dem buildingSMART Data Dictionary (bSDD) und Hersteller-spezifischen Daten wie Product Data Sheets ermöglicht. Dies würde die Interoperabilität und Präzision der Datenverarbeitung im BIM-Umfeld weiter verbessern.

Ein innovatives Konzept, das in Zukunft untersucht werden soll, ist die Integration eines "Property-Copilot" in Modellierungstools. Dieses Tool würde Nutzern dabei helfen, relevante Eigenschaften und Standards effizient zu identifizieren und anzuwenden, was die Modellierungsprozesse weiter optimieren würde.

Die Schaffung einer transparenten Verbindung zwischen allen Beteiligten eines Bauprojekts, von Planungsteams über Auftraggeber bis hin zu Genehmigungsinstanzen, ist ein weiteres wichtiges Ziel. Dies würde die Kommunikation und Koordination zwischen den verschiedenen Parteien verbessern und zu effizienteren Genehmigungsprozessen führen.

Schließlich sollte die Kombination der semantischen Web-Applikation mit erweiterten Prüfregeleln zur Nutzung topologischer und geometrischer Daten erforscht werden. Dies würde eine umfassendere und genauere Analyse und Validierung von BIM-Modellen ermöglichen, insbesondere in komplexen Bauprojekten.

Abschließend stellt diese Arbeit einen bedeutenden Beitrag zur Digitalisierung der Baubranche dar und ebnet den Weg für weitere Forschungen und Entwicklungen im Bereich Open BIM, insbesondere im Hinblick auf die Umsetzung öffentlicher Bauvorhaben, die Implementierung digitaler Bauanträge und die Optimierung von Planungs- und Genehmigungsprozessen.

Quellen

- [1] André Borrmann, Markus König, Christian Koch, Jakob Beetz. 2018. Building Information Modeling: Why? What? How? - (1.5 Summary). https://link.springer.com/chapter/10.1007/978-3-319-92862-3_1. [Letztes Besuch:10.01.2024]
- [2] Kathrin Köhler. 2019 Open versus ClosedBIM – Different Planning Approaches and Their Importance for the Planning Process of Municipalities n.d. https://pefnet.mendelu.cz/wcd/w-rek-pefnet/pefnet19-extended_abstracts_web.pdf. [Letztes Besuch:10.01.2024]
- [3] Reinhard Wimmer. 2020. BIM-Informationsmanagement bei der thermisch-energetischen Simulation von gebäudetechnischen Anlagen (2.1 BIM-Methode) n.d. <https://publications.rwth-aachen.de/record/784956/>. [Letztes Besuch:10.01.2024]
- [4] André Borrmann, Markus König, Christian Koch, Jakob Beetz. 2018. Building Information Modeling: Why? What? How? - (1.3.1 Little BIM vs. BIG BIM, Closed BIM vs. Open BIM) 2024. https://link.springer.com/chapter/10.1007/978-3-319-92862-3_1. [Letztes Besuch:10.01.2024]
- [5] Hamid Abdirad. 2020. Disruptive Information Exchange Requirements in Construction Projects: Perception and Response Patterns https://www.researchgate.net/publication/341117003_Disruptive_information_exchange_requirements_in_construction_projects_perception_and_response_patterns. [Letztes Besuch:10.01.2024]
- [6] Hasan Burak Cavka, Sheryl Staub-French, Erik A. Poirier. Automation in Construction (6. Conclusion) <https://www.sciencedirect.com/science/article/pii/S0926580517307227>. [Letztes Besuch:10.01.2024]
- [7] buildingSMART IDS Documentation. 2023 Developer guide n.d. <https://github.com/buildingSMART/IDS/blob/master/Documentation/developer-guide.md>. [Letztes Besuch:10.01.2024]
- [8] buildingSMART IDS Documentation. 2023 Information Delivery Specification Documentation n.d. <https://github.com/buildingSMART/IDS/blob/master/Documentation/README.md>. [Letztes Besuch:12.01.2024]
- [9] buildingSMART IDS Documentation. 2023 Guide to effective IDS metadata n.d. <https://github.com/buildingSMART/IDS/blob/master/Documentation/ids-metadata.md>.
- [10] buildingSMART International. 2023 Information Delivery Specifications (IDS) n.d. <https://www.buildingsmart.org/standards/bsi-standards/information-delivery-specifications-ids/>. [Letztes Besuch:12.01.2024]
- [11] buildingSMART IDS Documentation. 2023 How do specifications work? n.d. <https://github.com/buildingSMART/IDS/blob/master/Documentation/specifications.md>. [Letztes Besuch:12.01.2024]
- [12] buildingSMART IDS Issues. 2023 Setting the identifier for specification as a requirement n.d. <https://github.com/buildingSMART/IDS/issues/118>. [Letztes Besuch:12.01.2024]

[13] buildingSMART IDS Documentation. 2023 buildingSMART IDS issue #219 n.d.

<https://github.com/buildingSMART/IDS/issues/219>. [Letztes Besuch:12.01.2024]

[14] buildingSMART IDS Documentation. 2023 Entity facet n.d.

<https://github.com/buildingSMART/IDS/blob/master/Documentation/entity-facet.md>. [Letztes Besuch:12.01.2024]

[15] buildingSMART IFC4 ADD2 Documentation. 2023 IfcFurnitureTypeEnum n.d.

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/schema/ifcsharedfacilitieselements/lexical/ifcfurnituretypeenum.htm>. [Letztes Besuch:12.01.2024]

[16] buildingSMART IFC4 ADD2 Documentation. 2023 IfcFurniture n.d.

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/schema/ifcsharedfacilitieselements/lexical/ifcfurniture.htm>. [Letztes Besuch:12.01.2024]

[17] buildingSMART IFC4 ADD2 Documentation. 2023 IfcElementQuantity n.d.

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/schema/ifcproductextension/lexical/ifcelementquantity.htm>. [Letztes Besuch:12.01.2024]

[18] buildingSMART IFC4 ADD2 Documentation. 2023 IfcPropertySet n.d.

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/schema/ifckernel/lexical/ifcpropertyset.htm>. [Letztes Besuch:12.01.2024]

[19] buildingSMART IDS Documentation. 2023 Property facet n.d.

<https://github.com/buildingSMART/IDS/blob/master/Documentation/property-facet.md>. [Letztes Besuch:12.01.2024]

[20] buildingSMART IFC4 ADD2 Documentation. 2023 IfcPhysicalQuantity n.d.

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/schema/ifcquantityresource/lexical/ifcphysicalquantity.htm>. [Letztes Besuch:12.01.2024]

[21] buildingSMART International. 2023 bSDD - Property Fire Rating n.d.

<https://search.bsdd.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/FireRating>. [Letztes Besuch:15.01.2024]

[22] buildingSMART IDS Documentation. 2023 Units n.d.

<https://github.com/buildingSMART/IDS/blob/master/Documentation/units.md>. [Letztes Besuch:15.01.2024]

[23] buildingSMART IFC4 ADD2 Dokumentation. 2023 IfcClassification n.d.

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/schema/ifcexternalreferenceresource/lexical/ifclassification.htm>. [Letztes Besuch:15.01.2024]

[24] NBS. Uniclass EF_25_10 Walls 2021 n.d. https://uniclass.thenbs.com/taxon/ef_25_10. [Letztes

Besuch:15.01.2024]

[25] buildingSMART IDS Documentation. 2023 Material facet n.d.

<https://github.com/buildingSMART/IDS/blob/master/Documentation/material-facet.md>. [Letztes Besuch:15.01.2024]

- [26] buildingSMART IDS Documentation. 2023 Attribute facet n.d.
<https://github.com/buildingSMART/IDS/blob/master/Documentation/attribute-facet.md>. [Letztes Besuch:15.01.2024]
- [27] buildingSMART IDS Documentation. 2023 PartOf facet n.d.
<https://github.com/buildingSMART/IDS/blob/master/Documentation/partof-facet.md>. [Letztes Besuch:15.01.2024]
- [28] buildingSMART IDS Documentation. 2023 Partof testcases n.d.
<https://github.com/buildingSMART/IDS/blob/master/Documentation/testcases-partof.md>. [Letztes Besuch:15.01.2024]
- [29] buildingSMART International. 2023 Information Delivery Manual (IDM) n.d.
<https://technical.buildingsmart.org/standards/information-delivery-manual/>. [Letztes Besuch:15.01.2024]
- [30] J. Wix. Quick Guide: Business Process Modeling Notation (BPMN) n.d.
https://standards.buildingsmart.org/documents/IDM/IDM_guide-QuickGuideToBPMN-2007_01.pdf. [Letztes Besuch:15.01.2024]
- [31] J. Wix, J. Karlshøj. 2007 IDM: Guide to Components and Development Methods n.d.
https://standards.buildingsmart.org/documents/IDM/IDM_guide-CompsAndDevMethods-IDMC_004-v1_2.pdf. [Letztes Besuch:15.01.2024]
- [32] buildingSMART International. 2023 buildingSMART Data Dictionary (bSDD) session at InfraBIM Open 2023 n.d.
<https://www.youtube.com/watch?v=gieOryGD3NQ>. [Letztes Besuch:15.01.2024]
- [33] buildingSMART International. 2023 buildingSMART Data Dictionary n.d.
<https://www.buildingsmart.org/users/services/buildingsmart-data-dictionary/>. [Letztes Besuch:15.01.2024]
- [34] buildingSMART International. 2023 BIM Collaboration Format (BCF) n.d.
<https://www.buildingsmart.org/standards/bsi-standards/bim-collaboration-format-bcf/>. [Letztes Besuch:15.01.2024]
- [35] buildingSMART International. 2023 About openBIM® workflows n.d. <https://user.buildingsmart.org/knowledge-base/openbim-workflows-explained/>. [Letztes Besuch:15.01.2024]
- [36] buildingSMART International. 2023 What is BCF? n.d. <https://user.buildingsmart.org/knowledge-base/what-is-bcf/>. [Letztes Besuch:15.01.2024]
- [37] BuildingSMART. Industry Foundation Classes 4.0.2.1 2023.
<https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/link/ifcprojectlibrary.htm>. [Letztes Besuch:15.01.2024]
- [38] Apeksha Gupta. 2013Developing a BIM-based methodology to support renewable energy assessment of buildings n.d. <https://orca.cardiff.ac.uk/id/eprint/52917/1/2013GuptaAPhD.pdf>. [Letztes Besuch:15.01.2024]
- [39] Léon van Berlo, Peter Willems, Pieter Pauwels. 2023 Creating information delivery specifications using linked data n.d. <https://biblio.ugent.be/publication/8633671>. [Letztes Besuch:15.01.2024]

- [40] Noemi Kremer, Jakob Beetz. 2023 Extending information delivery specification for linking distributed model checking services n.d. https://ec-3.org/publications/conferences/EC32023/papers/EC32023_266.pdf. [Letztes Besuch:15.01.2024]
- [41] Zhang, C., Beetz, J., Weise, M. 2015 Interoperable validation for IFC building models using open standards n.d. <https://research.tue.nl/files/3969218/2763599124870.pdf>. [Letztes Besuch:15.01.2024]
- [42] buildingSMART International. Use Case Management service n.d. <https://ucm.buildingsmart.org/>. [Letztes Besuch:16.01.2024]
- [43] Use Case Management - a service by buildingSMART International. 2023 Use Case «Modellbasierte Bewehrung verlegen» n.d. <https://ucm.buildingsmart.org/use-case-details/2275/de>. [Letztes Besuch:16.01.2024]
- [44] Use Case Management - a service by buildingSMART International. 2023 Use Case «openBIM for Daylighting Design & Analyses in respect for Building Energy Performance» n.d. <https://ucm.buildingsmart.org/use-case-details/2525/en>. [Letztes Besuch:16.01.2024]
- [45] Use Case Management - a service by buildingSMART International. 2023 Use Case «Durchführung eines modellbasierten Aufgabenmanagements aus Sicht von bauausführenden Unternehmen» n.d. <https://ucm.buildingsmart.org/use-case-details/2459/de>. [Letztes Besuch:16.01.2024]
- [46] Use Case Management - a service by buildingSMART International. 2023 Use Case «Gebäudehülle» n.d. <https://ucm.buildingsmart.org/use-case-details/1587/de>. [Letztes Besuch:16.01.2024]
- [47] BIMcollab Webseite 2024 <https://www.bimcollab.com/de/>. [Letztes Besuch:16.01.2024]
- [48] ACCA Softwares - BIM-Lösungen 2023 <https://www.accasoftware.com/de/bim-software>. [Letztes Besuch:16.01.2024]
- [49] ACCA software. IDS BIM - Information Delivery Specification | usBIM.IDS | ACCA n.d. <https://www.youtube.com/watch?v=45TNdDgTZ-Y>. [Letztes Besuch:16.01.2024]
- [50] IfcOpenShell 0.7.0 documentation 2023. <https://blenderbim.org/docs-python/ifcopenshell.html>. [Letztes Besuch:16.01.2024]
- [51] BlenderBIM Add-on 0.0.231104 documentation 2023 https://blenderbim.org/docs/users/introduction_to_bim.html. [Letztes Besuch:16.01.2024]
- [52] BlenderBIM.org. IfcTester 2023 <https://blenderbim.org/docs-python/ifctester.html>. [Letztes Besuch:16.01.2024]
- [53] SPB Production. 2023 Create IDS specification with usBIM and validate IFC with it using BlenderBIM n.d. <https://www.youtube.com/watch?v=wkdwzTAB8FE>. [Letztes Besuch:16.01.2024]
- [54] Nemetschek 2023. Solibri n.d. <https://www.nemetschek.com/en/brand/solibri>. [Letztes Besuch:16.01.2024]

- [55] Solibri. 2023. Release Notes 9-13-6. <https://www.solibri.com/news/solibri-9-13-6-release-notes>. [Letztes Besuch:16.01.2024]
- [56] Solibri. What's new in Solibri Office – December 2023. <https://www.solibri.com/news/release-highlights-release-9-13-7>. [Letztes Besuch:16.01.2024]
- [57] xbim Ltd. 2024. <https://xbim.net/>. [Letztes Besuch:16.01.2024]
- [58] xbim Ltd. 2023. BIM Verification Tool - Flex Flow In Use n.d. <https://www.youtube.com/watch?v=5050BMUMXq0>. [Letztes Besuch:16.01.2024]
- [59] xbim Ltd. 2024. Improving and Adding Value in The Client Handover Process - xbim Webinar Catalogue n.d. https://6110118.hs-sites.com/webinar-catalogue?utm_campaign=webinars. [Letztes Besuch:17.01.2024]
- [60] Plannerly - About us. 2024. <https://plannerly.com/about/>. [Letztes Besuch:17.01.2024]
- [61] buildingSMART International. 2023. How to implement bSDD and IDS with Plannerly: A Step-by-Step Guide n.d. <https://www.youtube.com/watch?v=Zz0PZNkMKXM>. [Letztes Besuch:17.01.2024]
- [62] BIMQ. 2024. <https://www.bimq.de/en/>. [Letztes Besuch:17.01.2024]
- [63] BIMQ. 2023 The pre-release version of the IDS interface is now available! n.d. <https://www.bimq.de/en/the-pre-release-version-of-the-ids-interface-is-now-available/>. [Letztes Besuch:17.01.2024]
- [64] Solibri Inc. Verifying BIM models with IDS in Solibri- Everything you need to know n.d. <https://www.youtube.com/watch?v=VoUuiNZMTPs>. [Letztes Besuch:17.01.2024]
- [65] Artur Tomczaks. 2023. IDS-Editor n.d. <https://artomczak.pythonanywhere.com/create/>. [Letztes Besuch:17.01.2024]
- [66] Carlos Dias. 2023. IDS Converter. <https://idsconverter.streamlit.app/>. [Letztes Besuch:17.01.2024]
- [67] BIM Deutschland. 2023. Objektvorlagen. <https://via.bund.de/bim/objekte/landing>. [Letztes Besuch:17.01.2024]
- [68] buildingSMART. 2020. IFC4 Addendum 2. <https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2/HTML/>. [Letztes Besuch:17.01.2024]
- [69] buildingSMART. 2023. IDS Dokumentation - Entities must be specified as uppercase strings n.d. <https://github.com/buildingSMART/IDS/blob/master/Documentation/testcases-entity.md#fail-entities-must-be-specified-as-uppercase-strings>. [Letztes Besuch:17.01.2024]
- [70] Autodesk. 2022 Revit Sample Project Files. <https://help.autodesk.com/view/RVT/2023/ENU/?guid=GUID-61EF2F22-3A1F-4317-B925-1E85F138BE88>. [Letztes Besuch:17.01.2024]

Anhang

IDS-basierter IFC-Validierungsbericht

Information delivery validation REPORT

Created: 22/01/2024 14:53:45



TABLE OF CONTENT

1. IFC file information	3
2. IDS file information	4
3. IDS specifications metadata	5
4. IDS specifications definition	6
5. IDS specifications validation	7



1. IFC file information

Filename	Basic Sample
Timestamp	2023-02-24T00:07:48+01:00
Author	Autodesk
Organization	Autodesk
Preprocessor version	ODA SDAI 22.12
Originating system	23.0.11.19 - Exporter 23.0.11.19 - Alternativ-UI 23.0.11.19
Authorization	
Description	ViewDefinition [ReferenceView_V1.2]
Implementation level	2;1
Schema identifier	IFC4

2. IDS file information

Title	IDS Masterarbeit Prüfung
Copyright	Technisches Universität Berlin
Version	
Description	
Author	bali.rami911@gmail.com
Date	
Purpose	
Milestone	

3. IDS specifications metadata

Specification N°	Name	Description	Identifier	Optionality
1	Wall BIM Deutsch.	BIM Deutschland Objektvorlage für Wände		optional
2	Door BIM Deutsch.	BIM Deutschland Objektvorlage für Türen		optional

4. IDS specifications definition

Specification N°	Name	Information delivery requirements
1	Wall BIM Deutsch.	<p>The model may or may not have entities whose name satisfies the following restriction :</p> <p>1 - enumeration: "IFCWALL", "IFCWALLELEMENTEDCASE", "IFCWALLSTANDARDCASE".</p> <p>If exist, then they must have a property name = "Außen" possessing a data type IfcBoolean within a property set designated with a name = "Wand" and have a property name = "Betonfestigkeitsklasse" having a value that must always be satisfying the following restriction:</p> <p>1 - enumeration: "C8/10", "C12/15", "C16/20", "C20/25", "C25/30", "C30/37", "C35/45", "C40/50", "C45/55", "C50/60"</p> <p>and possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Bewehrungsstahlsorte" having a value that must always be satisfying the following restriction:</p> <p>1 - enumeration: "B500A", "B500B"</p> <p>and possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Feuerwiderstandsklasse F" having a value that must always be satisfying the following restriction:</p> <p>1 - enumeration: "F30", "F60", "F90", "F120", "F180"</p> <p>and possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Hauptbaustoff" possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Kostengruppe DIN276" possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Tragend" possessing a data type IfcBoolean within a property set designated with a name = "Wand".</p>
2	Door BIM Deutsch.	<p>The model may or may not have entities whose name satisfies the following restriction :</p> <p>1 - enumeration: "IFCDOOR", "IFCDOORSTANDARDCASE".</p> <p>If exist, then they must have a property name = "Außen" possessing a data type IfcBoolean within a property set designated with a name = "Tuer" and have a property name = "Feuerwiderstandsklasse T" having a value that must always be satisfying the following restriction:</p> <p>1 - enumeration: "F30", "F60", "F90", "F120", "F180"</p> <p>and possessing a data type IfcIdentifier within a property set designated with a name = "Tuer" and have a property name = "Hauptbaustoff" possessing a data type IfcIdentifier within a property set designated with a name = "Tuer".</p> <p>It is recommended for them to have a property name = "Kostengruppe DIN276" possessing a data type IfcIdentifier within a property set designated with a name = "Tuer".</p>

5. IDS specifications validation

Specification N°	Information delivery validation																									
1	<div><p>The model may or may not have entities whose name satisfies the following restriction :</p><p>1 - enumeration: "IFCWALL", "IFCWALLELEMENTEDCASE", "IFCWALLSTANDARDCASE".</p><p>If exist, then they must have a property name = "Außen" possessing a data type IfcBoolean within a property set designated with a name = "Wand" and have a property name = "Betonfestigkeitsklasse" having a value that must always be satisfying the following restriction:</p><p>1 - enumeration: "C8/10", "C12/15", "C16/20", "C20/25", "C25/30", "C30/37", "C35/45", "C40/50", "C45/55", "C50/60" and possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Bewehrungsstahlsorte" having a value that must always be satisfying the following restriction:</p><p>1 - enumeration: "B500A", "B500B" and possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Feuerwiderstandsklasse F" having a value that must always be satisfying the following restriction:</p><p>1 - enumeration: "F30", "F60", "F90", "F120", "F180" and possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Hauptbaustoff" possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Kostengruppe DIN276" possessing a data type IfcIdentifier within a property set designated with a name = "Wand" and have a property name = "Tragend" possessing a data type IfcBoolean within a property set designated with a name = "Wand".</p></div>	<div><div></div></div>																								
Requirements:																										
1	must have a property name = "Außen" possessing a data type IfcBoolean within a property set designated with a name = "Wand"																									
2	must have a property name = "Betonfestigkeitsklasse" having a value that must always be satisfying the following restriction : 1 - enumeration: "C8/10", "C12/15", "C16/20", "C20/25", "C25/30", "C30/37", "C35/45", "C40/50", "C45/55", "C50/60" and possessing a data type IfcIdentifier within a property set designated with a name = "Wand"																									
3	must have a property name = "Bewehrungsstahlsorte" having a value that must always be satisfying the following restriction : 1 - enumeration: "B500A", "B500B" and possessing a data type IfcIdentifier within a property set designated with a name = "Wand"																									
4	must have a property name = "Feuerwiderstandsklasse F" having a value that must always be satisfying the following restriction : 1 - enumeration: "F30", "F60", "F90", "F120", "F180" and possessing a data type IfcIdentifier within a property set designated with a name = "Wand"																									
5	must have a property name = "Hauptbaustoff" possessing a data type IfcIdentifier within a property set designated with a name = "Wand"																									
6	must have a property name = "Kostengruppe DIN276" possessing a data type IfcIdentifier within a property set designated with a name = "Wand"																									
7	must have a property name = "Tragend" possessing a data type IfcBoolean within a property set designated with a name = "Wand"																									
Validation result:																										
<table><tr><th>Entities</th><th>Req.</th><th>Result</th></tr><tr><td>#1032 (IfcWall), #1217 (IfcWall), #5277 (IfcWall), #5775 (IfcWall), #9699 (IfcWall), #10323 (IfcWall), #10435 (IfcWall), #10478 (IfcWall), #10560 (IfcWall), #10732 (IfcWall), #12447 (IfcWall), #12504 (IfcWall), #12684 (IfcWall), #12727 (IfcWall), #12771 (IfcWall), #12814 (IfcWall), #13495 (IfcWall), #13522 (IfcWall), #13545 (IfcWall), #13577 (IfcWall), #13684 (IfcWall), #18398 (IfcWall), #18442 (IfcWall), #19222 (IfcWall), #19403 (IfcWall), #19429 (IfcWall), #19451 (IfcWall), #19473 (IfcWall), #19495 (IfcWall), #19553 (IfcWall), #19619 (IfcWall), #32217 (IfcWall), #32239 (IfcWall), #32261 (IfcWall), #32284 (IfcWall), #32307 (IfcWall), #37658 (IfcWall), #37680 (IfcWall), #64173 (IfcWall), #101805 (IfcWall), #102920 (IfcWall), #103002 (IfcWall), #103934 (IfcWall), #104008 (IfcWall), #104051 (IfcWall), #105251 (IfcWall), #105294 (IfcWall)</td><td>1</td><td><div></div></td></tr><tr><td></td><td>2</td><td><div></div></td></tr><tr><td></td><td>3</td><td><div></div></td></tr><tr><td></td><td>4</td><td><div></div></td></tr><tr><td></td><td>5</td><td><div></div></td></tr><tr><td></td><td>6</td><td><div></div></td></tr><tr><td></td><td>7</td><td><div></div></td></tr></table>			Entities	Req.	Result	#1032 (IfcWall), #1217 (IfcWall), #5277 (IfcWall), #5775 (IfcWall), #9699 (IfcWall), #10323 (IfcWall), #10435 (IfcWall), #10478 (IfcWall), #10560 (IfcWall), #10732 (IfcWall), #12447 (IfcWall), #12504 (IfcWall), #12684 (IfcWall), #12727 (IfcWall), #12771 (IfcWall), #12814 (IfcWall), #13495 (IfcWall), #13522 (IfcWall), #13545 (IfcWall), #13577 (IfcWall), #13684 (IfcWall), #18398 (IfcWall), #18442 (IfcWall), #19222 (IfcWall), #19403 (IfcWall), #19429 (IfcWall), #19451 (IfcWall), #19473 (IfcWall), #19495 (IfcWall), #19553 (IfcWall), #19619 (IfcWall), #32217 (IfcWall), #32239 (IfcWall), #32261 (IfcWall), #32284 (IfcWall), #32307 (IfcWall), #37658 (IfcWall), #37680 (IfcWall), #64173 (IfcWall), #101805 (IfcWall), #102920 (IfcWall), #103002 (IfcWall), #103934 (IfcWall), #104008 (IfcWall), #104051 (IfcWall), #105251 (IfcWall), #105294 (IfcWall)	1	<div></div>		2	<div></div>		3	<div></div>		4	<div></div>		5	<div></div>		6	<div></div>		7	<div></div>
Entities	Req.	Result																								
#1032 (IfcWall), #1217 (IfcWall), #5277 (IfcWall), #5775 (IfcWall), #9699 (IfcWall), #10323 (IfcWall), #10435 (IfcWall), #10478 (IfcWall), #10560 (IfcWall), #10732 (IfcWall), #12447 (IfcWall), #12504 (IfcWall), #12684 (IfcWall), #12727 (IfcWall), #12771 (IfcWall), #12814 (IfcWall), #13495 (IfcWall), #13522 (IfcWall), #13545 (IfcWall), #13577 (IfcWall), #13684 (IfcWall), #18398 (IfcWall), #18442 (IfcWall), #19222 (IfcWall), #19403 (IfcWall), #19429 (IfcWall), #19451 (IfcWall), #19473 (IfcWall), #19495 (IfcWall), #19553 (IfcWall), #19619 (IfcWall), #32217 (IfcWall), #32239 (IfcWall), #32261 (IfcWall), #32284 (IfcWall), #32307 (IfcWall), #37658 (IfcWall), #37680 (IfcWall), #64173 (IfcWall), #101805 (IfcWall), #102920 (IfcWall), #103002 (IfcWall), #103934 (IfcWall), #104008 (IfcWall), #104051 (IfcWall), #105251 (IfcWall), #105294 (IfcWall)	1	<div></div>																								
	2	<div></div>																								
	3	<div></div>																								
	4	<div></div>																								
	5	<div></div>																								
	6	<div></div>																								
	7	<div></div>																								
2	<div><p>The model may or may not have entities whose name satisfies the following restriction :</p><p>1 - enumeration: "IFCDOOR", "IFCDOORSTANDARDCASE".</p><p>If exist, then they must have a property name = "Außen" possessing a data type IfcBoolean within a property set designated with a name = "Tuer" and have a property name = "Feuerwiderstandsklasse T" having a value that must always be satisfying the following restriction:</p><p>1 - enumeration: "F30", "F60", "F90", "F120", "F180" and possessing a data type IfcIdentifier within a property set designated with a name = "Tuer" and have a property name = "Hauptbaustoff" possessing a data type IfcIdentifier within a property set designated with a name = "Tuer".</p><p>It is recommended for them to have a property name = "Kostengruppe DIN276" possessing a data type IfcIdentifier within a property set designated with a name = "Tuer".</p></div>	<div><div></div></div>																								

Specification N°	Information delivery validation																							
	Requirements:																							
	1	must have a property name = "Außen" possessing a data type IfcBoolean within a property set designated with a name = "Tuer"																						
	2	must have a property name = "Feuerwiderstandsklasse T" having a value that must always be satisfying the following restriction: 1 - enumeration: "F30", "F60", "F90", "F120", "F180" and possessing a data type IfcIdentifier within a property set designated with a name = "Tuer"																						
	3	must have a property name = "Hauptbaustoff" possessing a data type IfcIdentifier within a property set designated with a name = "Tuer"																						
	4	recommended to have a property name = "Kostengruppe DIN276" possessing a data type IfcIdentifier within a property set designated with a name = "Tuer"																						
	Validation result:																							
	<table><tr><th>Entities</th><th>Req.</th><th>Result</th></tr><tr><td rowspan="4">#9831 (IfcDoor), #9957 (IfcDoor), #10419 (IfcDoor), #11087 (IfcDoor), #12921 (IfcDoor), #12970 (IfcDoor), #28044 (IfcDoor), #28081 (IfcDoor), #29096 (IfcDoor), #103085 (IfcDoor), #104166 (IfcDoor), #104183 (IfcDoor)</td><td>1</td><td>✓</td></tr><tr><td>2</td><td>✓</td></tr><tr><td>3</td><td>✓</td></tr><tr><td>4</td><td>✓</td></tr><tr><td rowspan="4">#12985 (IfcDoor), #12999 (IfcDoor), #13274 (IfcDoor), #18356 (IfcDoor)</td><td>1</td><td>✓</td></tr><tr><td>2</td><td>✓</td></tr><tr><td>3</td><td>✓</td></tr><tr><td>4</td><td>✗</td></tr></table>			Entities	Req.	Result	#9831 (IfcDoor), #9957 (IfcDoor), #10419 (IfcDoor), #11087 (IfcDoor), #12921 (IfcDoor), #12970 (IfcDoor), #28044 (IfcDoor), #28081 (IfcDoor), #29096 (IfcDoor), #103085 (IfcDoor), #104166 (IfcDoor), #104183 (IfcDoor)	1	✓	2	✓	3	✓	4	✓	#12985 (IfcDoor), #12999 (IfcDoor), #13274 (IfcDoor), #18356 (IfcDoor)	1	✓	2	✓	3	✓	4	✗
	Entities	Req.	Result																					
	#9831 (IfcDoor), #9957 (IfcDoor), #10419 (IfcDoor), #11087 (IfcDoor), #12921 (IfcDoor), #12970 (IfcDoor), #28044 (IfcDoor), #28081 (IfcDoor), #29096 (IfcDoor), #103085 (IfcDoor), #104166 (IfcDoor), #104183 (IfcDoor)	1	✓																					
		2	✓																					
3		✓																						
4		✓																						
#12985 (IfcDoor), #12999 (IfcDoor), #13274 (IfcDoor), #18356 (IfcDoor)	1	✓																						
	2	✓																						
	3	✓																						
	4	✗																						