Technische Universität München
TUM School of Engineering and Design

# Consistent management and evaluation of building models across design phases

Jimmy A. G. Abualdenien

Vollständiger Abdruck der von der TUM School of Engineering and Design der Technischen Universität München zur Erlangung eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitz: Prof. Dr. rer. nat. Thomas H. Kolbe

Prüfer*innen der Dissertation:

1. Prof. Dr.-Ing. André Borrmann
2. Prof. Dr.-Ing. Frank Petzold
3. Prof. Robert Amor, Ph.D. (University of Auckland, New Zealand)

Die Dissertation wurde am 27.09.2022 bei der Technischen Universität München eingereicht und durch die TUM School of Engineering and Design am 23.01.2023 angenommen.

# Abstract

Construction projects are characterized by being project-specific, multidisciplinary, and collaboration-intensive. Progression through the design phases requires the consideration of various conflicting requirements and boundary conditions. From the early phases, domain experts develop designs, evaluate their performance, and exchange them with other stakeholders while handling vague, imprecise, and incomplete information. Practitioners today mitigate these uncertainties during design and analysis, implicitly with solutions from previous projects and default values, highly influencing the cost and performance of the final design.

Using Building Information Modeling (BIM), a building model is represented with geometric, semantic, and topological relationships, which facilitates managing the refinement of information from one design phase to another. The Level of Development (LOD) is a well-established concept for describing the maturity of the individual elements. It is a domain language that aims to establish a common understanding of what each level means to facilitate the formalization of detailing requirements and model maturity. This is among the most fundamental tasks in BIM-based building design.

This far, existing BIM-authoring and simulation tools lack the means for incorporating and communicating information uncertainties. Semantics and the 3D representation of BIM models appear precise and certain, conveying wrong assumptions to the project participants. This significantly influences the quality and performance of the developed solutions, especially when collaboratively developing partial models (e.g., structural and mechanical designs) and performing simulations and analysis.

Automatically validating the completeness and quality of the delivered BIM models is an essential task in BIM workflows. This far, numerous tools support validating the semantic information. However, validating the modeled geometry for compliance with detailing requirements is still a manual process, where domain experts evaluate the models manually based on their experience.

This research first formally addresses representing design requirements and the potential uncertainties through developing a multi-LOD meta-model. The meta-model provides the necessary basis for conveying the specified vagueness through multiple visualization approaches and developing a framework for preserving the consistency of building models across the design phases. Multiple intrinsic, extrinsic, animation, and walkthrough visualizations have been developed and positively evaluated for depicting the amount and impact of the uncertainties on the design.

The consistency of models was preserved by ensuring that the refinement of semantics and geometry is within the expected range of possible design options and the topological relationships are maintained within equivalency from one phase to another. The developed approach could successfully flag inconsistencies to designers when evaluated on a real-world project. To automatically validate the degree of geometric detailing of building elements, this research generated a dataset of elements on multiple LOGs (Levels of Geometry) and then extracted a set of geometric features, representing the complexity of each element on each level. The extracted features were then used to train two tree-based ensemble models capable of correctly classifying new elements with an accuracy between 83% and 85%.

After assisting decisions during the design phases, the second part of this dissertation focuses on enriching building models with information that can document design intent and the rationale behind the decisions made. This helps maintaining design knowledge and facilitating their reuse in future projects. In this regard, building models were enriched with constraints (spatial and semantic) and links to requirements and regulations (in natural language). Finally, graph representation and rewriting systems were leveraged to formally represent the captured detailing patterns and transfer them to new projects. The approaches were evaluated on multiple real-world use cases and proved their flexibility to document and reuse detailing patterns in various use cases.

# Zusammenfassung

Bauprojekte sind charakterisiert durch projektspezifische, multidisziplinäre und zusammenarbeitsintensive Abläufe. Das Durchlaufen der Entwurfsphasen erfordert die Berücksichtigung verschiedener widersprüchlicher Anforderungen und Randbedingungen. Bereits in den frühen Phasen entwickeln Fachleute Entwürfe, bewerten deren Leistung und tauschen sie mit anderen Beteiligten aus, während sie mit vagen, ungenauen und unvollständigen Informationen umgehen müssen. Fachleute reduzieren diese unklaren Faktoren während des Entwurfs und der Analysen, indem sie implizit auf Lösungen aus früheren Projekten und Standardwerte zurückgreifen, was einen starken Einfluss auf die Kosten und die Qualität des endgültigen Entwurfs haben kann.

Beim Building Information Modeling (BIM) wird ein Modell eines Gebäudes mit geometrischen, semantischen und topologischen Beziehungen dargestellt, was die Verfeinerung der Informationen von einer Planungsphase zur nächsten erleichtert. Level of Development (LOD) ist ein etabliertes Konzept zur Beschreibung des Entwicklungsstands der einzelnen Elemente. LOD ist eine Fachsprache, die darauf abzielt, ein gemeinsames Verständnis zu entwickeln, was jedes Level für die Kommunikation und die vertragliche Festlegung der Ergebnisse unter den Projektteilnehmern bedeutet. Diese Aufgabe gehört zu den grundlegendsten in der BIM-basierten Bauplanung, da sie die Anforderungen an die Detaillierung und den Entwicklungsstand des Modells formalisiert.

Bislang fehlt es den bestehenden BIM-Werkzeugen für die Erstellung und Simulation an Methoden, um Informationsungewissheiten einzubeziehen und zu kommunizieren. Semantik und 3D-Darstellung von BIM-Modellen erscheinen präzise und sicher und können den Projektbeteiligten falsche Annahmen vermitteln. Dies hat einen erheblichen Einfluss auf die Qualität und Performance der entwickelten Lösungen, insbesondere bei der kollaborativen Entwicklung von Teilmodellen (z.B. der Tragwerksplanung) und der Durchführung von Simulationen und Analysen.

Die automatisierte Validierung der Vollständigkeit und Qualität der gelieferten BIM-Modelle ist eine wesentliche Aufgabe in BIM-Workflows. Bislang unterstützen zahlreiche Werkzeuge die Validierung der semantischen Informationen. Die Validierung der modellierten Geometrie auf Übereinstimmung mit den Detaillierungsanforderungen ist jedoch nach wie vor ein manueller Prozess, bei dem Fachexperten die Modelle auf der Grundlage ihrer Erfahrung persönlich bewerten.

Diese Forschung beschäftigte sich zunächst mit der formalen Darstellung von Entwurfsanforderungen und den potenziellen Unklarheiten durch die Entwicklung eines Multi-LOD-Metamodells. Das Metamodell lieferte die notwendige Grundlage für die Vermittlung der definierten Unschärfe durch mehrere Visualisierungsansätze und die Entwicklung eines Rahmens für die Konsistenz der Gebäudemodelle über die Entwurfsphasen hinweg. Es wurden mehrere intrinsische, extrinsische, animierte und begehbare Visualisierungen entwickelt und

positiv evaluiert, um den Umfang und die Auswirkungen der Ungewissheiten auf den Entwurf darzustellen.

Die Konsistenz der Modelle wurde sichergestellt, indem die Verfeinerung der Semantik und der Geometrie innerhalb der erwarteten Bandbreite möglicher Entwurfsoptionen liegt und die topologischen Beziehungen von einer Phase zur anderen gleichwertig bleiben. Bei der Evaluierung eines realen Projekts konnte der entwickelte Ansatz den Designer erfolgreich auf Inkonsistenzen hinweisen. Um den Grad der geometrischen Detaillierung von Gebäudeelementen automatisch zu validieren, wurde im Rahmen dieser Forschungsarbeit ein Datensatz von Elementen auf mehreren LOGs ( Level of Geometry) erstellt und anschließend eine Reihe von geometrischen Merkmalen extrahiert, die die Komplexität jedes Elements auf jeder Ebene darstellen. Die extrahierten Merkmale wurden dann verwendet, um zwei baum-basierte Ensemble-Modelle zu trainieren, die in der Lage sind, neue Elemente mit einer Genauigkeit zwischen 83% und 85% korrekt zu klassifizieren.

Nach der Unterstützung von Entwurfsentscheidungen während der Entwurfsphasen konzentriert sich der zweite Teil dieser Dissertation auf die Anreicherung von Gebäudemodellen mit Informationen, die die Entwurfsabsicht und die Gründe für die getroffenen Entscheidungen dokumentieren können. Dies trägt dazu bei, das Entwurfswissen zu erhalten und die Wiederverwendung in zukünftigen Projekten zu erleichtern. In diesem Zusammenhang wurden die Gebäudemodelle mit (räumlichen und semantischen) Einschränkungen und Links zu Anforderungen und Vorschriften (in natürlicher Sprache) angereichert. Schließlich wurden Graphdarstellungs- und Rewriting-Systeme eingesetzt, um die erfassten Detaillierungsmuster formal darzustellen und sie auf neue Projekte zu übertragen. Die Ansätze wurden an mehreren realen Anwendungsfällen evaluiert und bewiesen ihre Flexibilität bei der Dokumentation und Wiederverwendung von Detaillierungsmustern in verschiedenen Anwendungsfällen.

# Acknowledgements

This thesis is the outcome of my research at the chair of Computational Modeling and Simulation at Technical University of Munich. This experience has significantly influenced my personal and professional development, providing me with all the courage, knowledge, and skills to design, develop, and innovate with new open-minded ideas.

My sincere gratitude goes to Prof. André Borrmann for his valuable guidance and supervision, supporting me in following a clear path and overcoming the different challenges. André, thank you for providing the perfect environment for self-development and growth!

Besides the collaboration throughout the joint research projects, Prof. Frank Petzold supported my journey in this Ph.D. in many ways, including his valuable advice as a mentor and in the final examination. Thank you!

A big thank you also to Prof. Robert Amor for the valuable discussions on the scientific avenues and also for taking over the task of the second examiner for my Ph.D.!

Next, I want to thank Prof. Thomas Kolbe for the interesting discussions during my research at TUM and also for accepting to chair the examination board of my Ph.D.!

Thanks to my colleagues in the research group for the continuous exchange, support, and good times. Each one of you is special and has contributed to my knowledge.

Last but not least, I want to thank my friends and especially my parents for supporting me in making decisions outside my comfort zone. Big thanks to my wife Angelina and my son Tony for the continuous support, motivation, and joy you bring to my life. I love you.

# Contributions

This cumulative dissertation is based on four published, peer-reviewed research papers, which are presented in chapters 2 to 7.

### Paper I

**Abualdenien, J.**; Borrmann, A. (2022): *Levels of detail, development, definition, and information need: a critical literature review*, Journal of Information Technology in Construction 27, pp. 363-392, 2022, DOI: 10.36680/j.itcon.2022.018

Contributions:
Jimmy Abualdenien designed and conducted the systematic literature review for reviewing the most common concepts for describing design maturity and requirements. André Borrmann supervised this study, contributed to the conceptualization, and reviewed the manuscript. All authors approved the final version.

### Paper II

**Abualdenien, J.**; Borrmann, A. (2019): *A meta-model approach for formal specification and consistent management of multi-LOD building models*, Advanced Engineering Informatics 40 (1474-0346), pp. 135-153, 2019, DOI: 10.1016/j.aei.2019.04.003

Contributions:
Jimmy Abualdenien and André Borrmann conceptualized the idea of the Multi-LOD model. Jimmy Abualdenien designed the Multi-LOD model and developed a concept for checking the consistency of building models. André Borrmann supervised this study and reviewed the manuscript. All authors approved the final version.

### Paper III

**Abualdenien, J.**; Borrmann, A. (2020): *Vagueness visualization in building models across different design stages*, Advanced Engineering Informatics 45, pp. 101-1018, 2020, DOI: 10.1016/j.aei.2020.101107

Contributions:
Jimmy Abualdenien developed the vagueness visualization approaches and evaluated their effectiveness on different scales and use cases. André Borrmann supervised this study, contributed to the conceptualization, and reviewed the manuscript. All authors approved the final version.

### Paper IV

**Abualdenien, J.**; Borrmann, A. (2022): *Ensemble-learning approach for the classification of Levels Of Geometry (LOG) of building elements*, Advanced Engineering Informatics 51 (1474-0346), pp. 10149, 2022, DOI: 10.1016/j.aei.2021.101497

Contributions:
Jimmy Abualdenien extracted a representative set of geometric features describing the geometric complexity of building elements. Jimmy Abualdenien developed a framework for classifying building elements according to their Level of Geometry. André Borrmann supervised this study, contributed to the conceptualization, and reviewed the manuscript. All authors approved the final version.

### Paper V

Zahedi, A.; **Abualdenien, J.**; Petzold, F.; Borrmann, A. (2022): *BIM-based design decisions documentation using design episodes, explanation tags, and constraints*, Journal of Information Technology in Construction 27, pp. 756-780, 2022, DOI: https://dx.doi.org/10.36680/j.itcon.2022.037

Contributions:
Ata Zahedi and Jimmy Abualdenien developed the underlying method for documenting design decisions based on BIM methodology. Ata Zahedi conceptualized the design episodes and explanation tags notions for documenting design decisions' intent and rationale and developed the list of explanation tags together with the demonstrative use cases. Jimmy Abualdenien developed an approach for using both constraints and Natural Language Processing (NLP) for linking requirements and building regulations and explicitly describing design intent. Frank Petzold and André Borrmann supervised this study, contributed to the conceptualization, and reviewed the manuscript. All authors approved the final version.

### Paper VI

**Abualdenien, J.**; Borrmann, A. (2021): *PBG: A parametric building graph capturing and transferring detailing patterns of building models*, In: Proc. of the CIB W78 Conference 2021, Luxembourg, 2021

Contributions:
Jimmy Abualdenien developed a methodology for capturing and transferring detailing patterns using graph representations and rewriting systems. André Borrmann supervised this study, contributed to the conceptualization, and reviewed the manuscript. All authors approved the final version.

# Further Related Scientific Contributions

## Book chapters

- Abualdenien, J.; Borrmann, A.; König, M.: Ausarbeitungsgrade von BIM-Modellen. In: Borrmann, A.; König, M.; Koch, C.; Beetz, J. (eds): Building Information Modeling, Springer, 2021
- Amann, J.; Esser, S.; Krijnen, T.; Abualdenien, J.; Preidel, C.; Borrmann, A.: BIM-Programmierschnittstellen. In: Borrmann, A.; König, M.; Koch, C.; Beetz, J. (eds): Building Information Modeling, Springer, 2021

## Peer reviewed journal papers

- Abualdenien, J.; Schneider-Marin, P.; Zahedi, A.; Harter, H.; Exner, H.; Steiner, D.; Mahan Singh, M.; Borrmann, A.; Lang, W.; Petzold, F.; König, M.; Geyer, P.; Schnellenbach-Held, M.: Consistent management and evaluation of building models in the early design stages, Journal of Information Technology in Construction 25, pp. 212-232, 2020, DOI: 10.36680/j.itcon.2020.013

## Peer reviewed conference papers

- Abualdenien, J.; Borrmann, A.: Formal analysis and validation of Levels of Geometry (LOG) in building information models, In: Proc. of the 27th International Workshop on Intelligent Computing in Engineering, Berlin, Germany, 2020
- Exner, H.; Abualdenien, J.; König, M.; Borrmann, A.: Managing Building Design Variants at Multiple Development Levels, In: Proc. of the 36th International Council for Research and Innovation in Building and Construction (CIB W78), Newcastle, UK, 2019
- Abualdenien, J.; Pfuhl, S.; Braun, A.: Development of an MVD for checking fire-safety and pedestrian simulation requirements, In: Proc. of the 31th Forum Bauinformatik, Berlin, Germany, 2019
- Schneider-Marin, P.; Abualdenien, J.: A framework to facilitate an interdisciplinary design process using BIM In: Proc. of the 31th Forum Bauinformatik, Berlin, Germany, 2019
- Abualdenien, J.; Borrmann, A.: A multi-LOD model for visualizing building information models' vagueness, In: Proceedings of The 2019 ASCE International Conference on Computing in Civil Engineering, Atlanta, U.S.A, 2019

- Zahedi, A.; Abualdenien, J.; Petzold, F.; Borrmann, A.: Minimized communication protocol based on a multi-LOD meta-model for adaptive detailing of BIM models, In: Proc. of the 26th International workshop on intelligent computing in engineering, Leuven, Belgium, 2019

- Abualdenien, J.; Borrmann,A.: Multi-LOD model for describing uncertainty and checking requirements in different design stages, In: eWork and eBusiness in Architecture, Engineering and Construction: Proceedings of the 11th European Conference on Product and Process Modelling (ECPPM 2018), Copenhagen, Denmark, 2018

# Contents

# Acronyms

| | |
|---|---|
| **AEC** | Architecture, Engineering, and Construction |
| **AIA** | American Institute of Architects |
| **API** | Application Programming Interface |
| **BDL** | Building Development Level |
| **BIM** | Building Information Modeling |
| **bsDD** | buildingSmart Data Dictionary |
| **BxP** | Project execution Plan |
| **CAD** | Computer-Aided Design |
| **CBD** | Case-Based Design |
| **CDS** | Computational Design Synthesis |
| **DE** | Design Episode |
| **DFG** | Deutsche Forschungsgemeinschaft |
| **DPs** | Detailing Patterns |
| **EGHG** | Embedded GreenHouse Gases |
| **ET** | Explanation Tag |
| **GRS** | Graph Rewriting Systems |
| **HVAC** | Heating, Ventilation, and Air Conditioning |
| **IFC** | Industry Foundation Classes |
| **LCA** | Life Cycle Assessment |
| **LOA** | Level of Accuracy |
| **LoD** | Level of Detail |
| **LOD** | Level of Development |
| **LOG** | Level of Geometry |
| **LOI** | Level of Information |
| **LOIN** | Level of Information Need |
| **MDG** | Modelldetaillierungsgrad |
| **ML** | Machine Learning |
| **MVD** | Model View Definition |
| **NLP** | Natural Language Processing |
| **PBG** | Parametric Building Graph |
| **PCC** | Pearson correlation coefficient |
| **REST** | REpresentational State Transfer |
| **RF** | Random Forest |
| **RFP** | Request for Proposal |
| **SDF** | Shape-Diameter Function |

| | |
|---|---|
| **SHAP** | Shapley Additive exPlanations |
| **XGBoost** | eXtreme Gradient Boosting |

# Chapter 1

# Introduction & background

## 1.1 Design process in the AEC industry

Projects in the Architecture, Engineering, and Construction (AEC) industry are characterized by being progressive, multi-disciplinary and collaborative. Conventionally and using Building Information Modeling (BIM), a model is developed through multiple phases involving various disciplines, including architectural, structural, energy analysis experts, and more. With the increasing digitization in the AEC industry, design processes and workflows are becoming more dynamic and specialized for every project. During each design phase, the attention of domain experts oscillates between understanding the problem and developing a solution. On average, 58% of the time during the design process is spent on managing information (Flager et al., 2009). As a result, the design concept and information are collaboratively refined to satisfy various design and engineering requirements and boundary conditions. For example, developing a structural design requires architectural design information as input. At the same time, the design of a Heating, Ventilation, and Air Conditioning (HVAC) system has to be coordinated with the structural system to take into account the required voids in slabs and structural members. Facilitating such collaboration among the involved disciplines requires establishing an agreement on (what) information should be available at what time (when). Based on the available information, it can be decided which evaluations and calculations can be performed using the model (purpose), which makes it possible to determine what model deliverables are expected from the actors involved (who) (Beetz et al., 2018).

BIM is a well-established methodology for cross-disciplinary building design based on the creation, management, and exchange of semantically rich 3D models (C. M. Eastman et al., 2011). BIM models comprise descriptions of geometric and semantic information as well as their topological relationships and functional dependencies. The 3D representation of objects is a fundamental perspective for numerous domains, from computer graphics to BIM. Especially in BIM, the 3D representation of building elements is the primary way of defining the shape of a building and its components. It is also a fundamental aspect for performing a variety of tasks, including clash detection or even exploring the reliability of the building information across the design phases (Borrmann et al., 2018a). BIM improves the process' efficiency and quality by promoting the early exchange of 3D building models.

Through the phases of a construction project, a BIM model is gradually refined from a rough conceptual design to highly detailed individual elements. In this process, each expert, such as the architect and structural engineer, uses different authoring tools and requires specific information to be present in the model to perform a particular type of simulations and analysis. With the increasing specialization of the stakeholders, the building industry requires a high level of interoperability. For this purpose, the international standardization organization, buildingSMART, promotes Industry Foundation Classes (IFC), a vendor-neutral data exchange format specialized for information in the AEC industry (Liebich, 2013).

## 1.2 BIM-based design process

Overall, the costs involved in design are 20% of construction costs. However, the quality of the constructed designs has a high impact on maintenance and operation costs, which are as much as 200 times construction costs (Flager et al., 2009). The process of designing a building requires several iterations that are collaboratively realized to adapt the developed solutions towards the project goals, deeming the design phases, especially the early ones, complex to manage (Knotten et al., 2015).

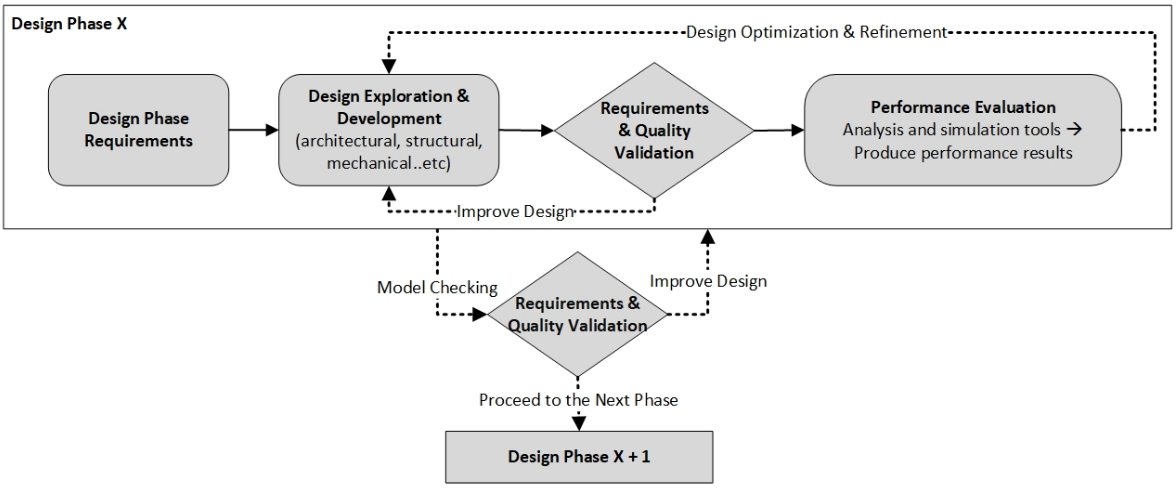
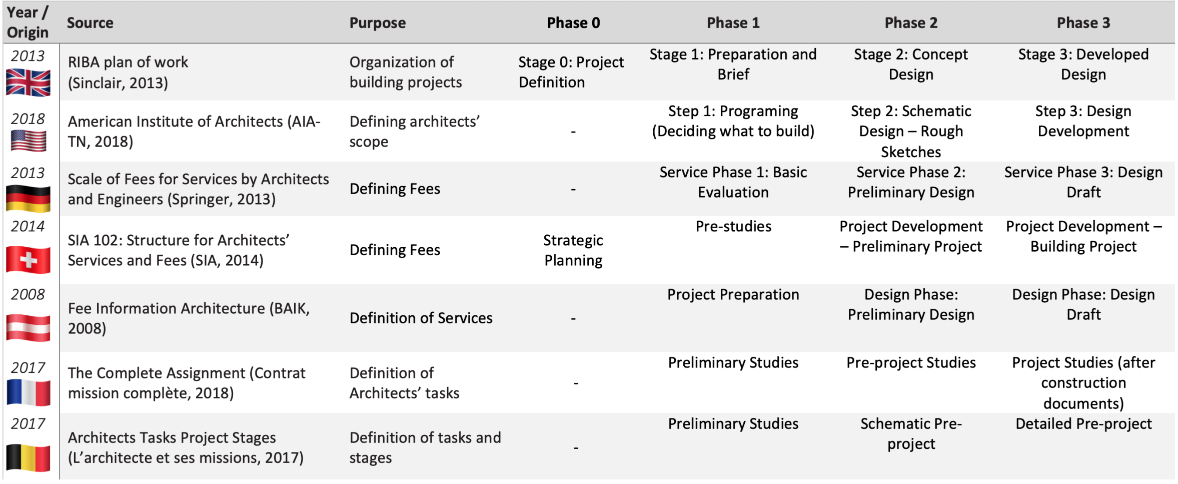


Figure 1.1: Overview of the interactions and collaboration during the design phases.

Figure 1.1 illustrates the flow of developing a design throughout the design phases. In each phase, owner requirements, regulations and boundary conditions are considered. To develop the design further by a different domain expert, BIM information is validated for completeness and compliance to exchange requirements. Afterwards, discipline-specific partial models (e.g., mechanical, electrical, etc.) are developed, and multiple simulations as well as calculations are performed. In this context, performance results inform the design process to support optimizing the design towards the intended goals. Once the deliverables of a design phase are produced, they are checked against the project requirements, providing a gateway for proceeding to the next design phase.

Internationally, the design process is aligned to the activities that architects and engineers consecutively perform along with their costs. Hence, each country follows standards for managing the design phases. Table 1.1 shows an overview of the definitions followed in the different countries for the early design phases (before construction documents are prepared).

Table 1.1: Selected list of guidelines for managing the design phases - focused on the early phases (Schneider-Marin & Abualdenien, 2019)



| Year / Origin | Source | Purpose | Phase 0 | Phase 1 | Phase 2 | Phase 3 |
|---|---|---|---|---|---|---|
| *2013* 🇬🇧 | RIBA plan of work (Sinclair, 2013) | Organization of building projects | Stage 0: Project Definition | Stage 1: Preparation and Brief | Stage 2: Concept Design | Stage 3: Developed Design |
| *2018* 🇺🇸 | American Institute of Architects (AIA-TN, 2018) | Defining architects' scope | - | Step 1: Programing (Deciding what to build) | Step 2: Schematic Design – Rough Sketches | Step 3: Design Development |
| *2013* 🇩🇪 | Scale of Fees for Services by Architects and Engineers (Springer, 2013) | Defining Fees | - | Service Phase 1: Basic Evaluation | Service Phase 2: Preliminary Design | Service Phase 3: Design Draft |
| *2014* 🇨🇭 | SIA 102: Structure for Architects' Services and Fees (SIA, 2014) | Defining Fees | Strategic Planning | Pre-studies | Project Development – Preliminary Project | Project Development – Building Project |
| *2008* 🇦🇹 | Fee Information Architecture (BAIK, 2008) | Definition of Services | - | Project Preparation | Design Phase: Preliminary Design | Design Phase: Design Draft |
| *2017* 🇫🇷 | The Complete Assignment (Contrat mission complète, 2018) | Definition of Architects' tasks | - | Preliminary Studies | Pre-project Studies | Project Studies (after construction documents) |
| *2017* 🇧🇪 | Architects Tasks Project Stages (L'architecte et ses missions, 2017) | Definition of tasks and stages | - | Preliminary Studies | Schematic Pre-project | Detailed Pre-project |

Although existing guidelines provide means for estimating the effort and cost involved (cost-oriented), they are rigid for supporting the milestones' deliverables. Each milestone involves exchanging BIM models and performing different calculations and simulations (for example, quantity take-off and evacuation analysis). Hence, a more flexible concept that is information-maturity oriented, acting as a container for the design requirements, is necessary for supporting the collaborative workflows across a project's phases.

### 1.2.1 Early design phases

The decisions made throughout the design phases, especially the early ones, steer a project's success and results (Gilbertson, 2006; Howell, 2016). The impact of the decisions made in the early design phases (conceptual and preliminary phases) is significant, as they form the basis of the following phases (Kraft & Nagl, 2007; Steinmann, 1997). The focus in the early phases is on the building's overall structural system, outer form, and interior organization (Joedicke, 1993; Steinmann, 1997; Struck et al., 2007). In these phases, the uncertainty on how the design may evolve is high, as many decisions have not yet been made (Knotten et al., 2015). As the design evolves through the phases, domain experts tend to leverage those uncertainties as a degree of freedom to explore the potential design options. Then, the performance of these designs is compared for making decisions that refine the concept further towards fulfilling the desired project goals.

For instance, as illustrated in Figure 1.2, architects could evaluate different building shapes (e.g., rectangular or L-Shape), facade materials (curtain or concrete walls), or openings percentages. At the same time, structural engineers could investigate the load distribution by varying the quantities, material, and thicknesses of columns and slabs based on the architectural model provided by the architect. From a different perspective, experts in crowd and energy analysis could perform multiple calculations and simulations to evaluate the impact of the individual design options on the design performance, informing the decision-making process. Hence, several researchers have emphasized the advantages of integrating performance analysis and simulations early by incorporating the information uncertainty (Hopfe & Hensen, 2011; Struck et al., 2007).

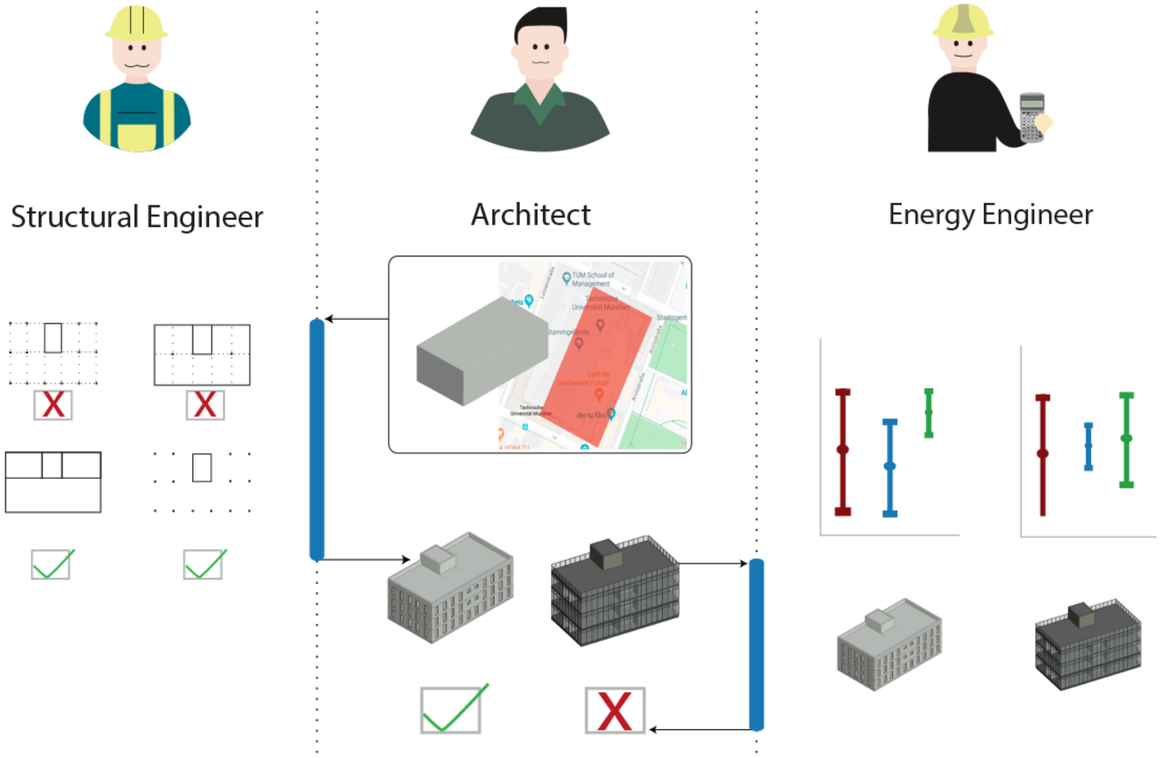


Figure 1.2: Design process in the early design phases - Simulation driven design: an example of the interaction and collaboration between the different disciplines to support design decisions in a way that fulfills the project requirements.

However, although domain experts employ analysis tools to support decisions, a well-reported gap exists between the predicted and actual building performance (De Wilde, 2014; Menezes et al., 2012). One reason for this gap is the unknown or missing information in the early phases. Practitioners quantify uncertainties in the model's inputs, such as geometric and material attributes, using information from literature, experience, or default values (De Wilde, 2014; Van Dronkelaar et al., 2016). Therefore, at every phase, the information needed along with its uncertainties must be defined and communicated to the project participants to alleviate the uncertainties' impact on simulation results and improve the quality of the decision-making (Sanguinetti et al., 2009; Tian et al., 2018; Van Gelder et al., 2014).

### 1.2.2 Information uncertainty & vagueness

Information uncertainty is complex, multidimensional, and has many interpretations. The terms uncertainty, fuzziness, and vagueness are used in various domains and application contexts (Raskin & Taylor, 2014). Most commonly, uncertainty is an umbrella term that describes a lack of knowledge or information, causing the occurrence of an uncertain future state (Hawer et al., 2018). A fundamental definition of the term *uncertainty* encompasses multiple concepts: liability to chance or accident, lack of knowledge, lack of information, or lack of trust in knowledge (Murray et al., 1961; Wynn et al., 2011).

Generally, there are two main sources of uncertainty, *Aleatory* and *Epistemic* (Mullins et al., 2016). Aleatory uncertainty is irreducible as it represents the natural variation of inputs and is commonly treated with probability theory. On the other hand, Epistemic uncertainty results from a lack of knowledge about a system, including parameter uncertainty, solution approximation errors, measurement uncertainty, and imprecise data. Hence, epistemic uncertainty can be reduced by obtaining additional information, supporting decisions about data collection and model improvement (Mullins et al., 2016).

The AEC industry encounters both sources of uncertainty during an asset's life cycle. However, domain experts collaborate on reducing epistemic uncertainty during the design phases by gradually evolving the design solution. In this regard, the assumptions made due to the lack of information or knowledge throughout the design phases are a primary cause of information uncertainty (Nilsen & Aven, 2003). The presence of uncertainty influences the produced designs and their performance, impacting the decisions made (Raskin & Taylor, 2014).

From an epistemic uncertainty point of view, vagueness is related to a specific state of a specific object, which refers to having imprecise or inaccurate information (Hawer et al., 2018; Klir, 1987). In the context of Computer-Aided Design (CAD) modeling, Steinmann, 1997 described fuzziness (a synonym of vagueness) as the distance from the complete and exact description (Steinmann, 1997). When considering the uncertainties incorporated into the design process, there are four main categories:

- *Requirements uncertainty*: The main intentions of the building design, including its usage, environmental impact, and cost. Understanding the client's requirements and decisions is important for an efficient design process (Sujan et al., 2019). Kometa et al., 1996 explain the client's influence on the successful execution of construction projects (Kometa et al., 1996). Additionally, the source of requirements' uncertainty can be regulations and other boundary conditions.

- Design uncertainty: Significant decisions in construction projects are reliant on heuristic processes where assumptions are developed from past experience (Kerzner & Kerzner, 2017). Typically, the process involves choosing among design options and variants while fulfilling the project goals and requirements. This kind of uncertainty has a wide range of combinations in the early design phases and becomes narrower as more decisions are made in the subsequent phases. Design uncertainty and decisions have an impact on the information flow and latency (Sujan et al., 2019).
- Interaction uncertainty: Design decisions are built on the continuous feedback and exchange of information among the participating domain experts. Architects typically evaluate various requirements, including functional, operational, and architectural requirements, to make design decisions. Architects are usually concerned with *what the building is*, rather than *how the building performs* (Rezaee et al., 2015). Therefore, with the presence of requirements and design uncertainties, the interaction among the project participants is necessary to agree on the model content and incorporate the building performance in making subjective estimations and decisions.
- Performance uncertainty: Performing model analysis utilizes the design information as an input. Accordingly, the uncertainty of the design and requirements, such as material properties, a scenario of use, or other boundary conditions, propagate into the analysis results, producing a range or a set of outcomes. This kind of uncertain results assists in making design decisions for developing an optimized solution, fulfilling the project's requirements.

In this thesis, vagueness describes the reliability of the building elements' attributes and their refinement throughout the design phases, for example, the exact material of load-bearing elements or the percentage of the facade openings.

### 1.2.3 Level of Development (LOD)

The Level of Detail (LoD) concept is an old topic that existed in computer graphics for bridging the graphical complexity and performance by regulating the amount of detail used to visualize the virtual world (Luebke, Reddy, Cohen, et al., 2003). In 2005, VicoSoftware (Trimble, 2013; VicoSoftware, 2005) has introduced the LoD concept for the AEC industry, describing the necessary semantic and geometric information with a set of five levels.

In 2008, the American Institute of Architects (AIA) built upon the LoD definitions and introduced the Level of Development (LOD), which also comprises five levels starting from LOD

100 and reaching LOD 500. From 2013, the BIMForum working group has investigated the AIA definitions in detail and introduced LOD 350 (BIMForum, 2019). The BIMForm subsequently has published updated versions of the *Level of Development Specification* in a yearly cycle with the aim of providing a common understanding of the expected information at every LOD. It starts with a conceptual model at LOD 100, which is limited to a generic representation of the building elements, meaning no shape information or geometric representation. The second level, LOD 200 (approximate geometry), consists of generic elements as placeholders with approximate geometric and semantic information. At LOD 300 (precise geometry), all the elements are modeled with their quantity, size, shape location, and orientation. Next, to enable the detailed coordination between the different disciplines, such as clash detection and avoidance, LOD 350 (construction documentation) is introduced, including the interfaces between all the building systems. Reaching LOD 400, the model incorporates additional information about detailing, fabrication, assembly, and installation. Lastly, at LOD 500 (as built), the model elements are a field verified representation in terms of size, shape, location, quantity, and orientation.

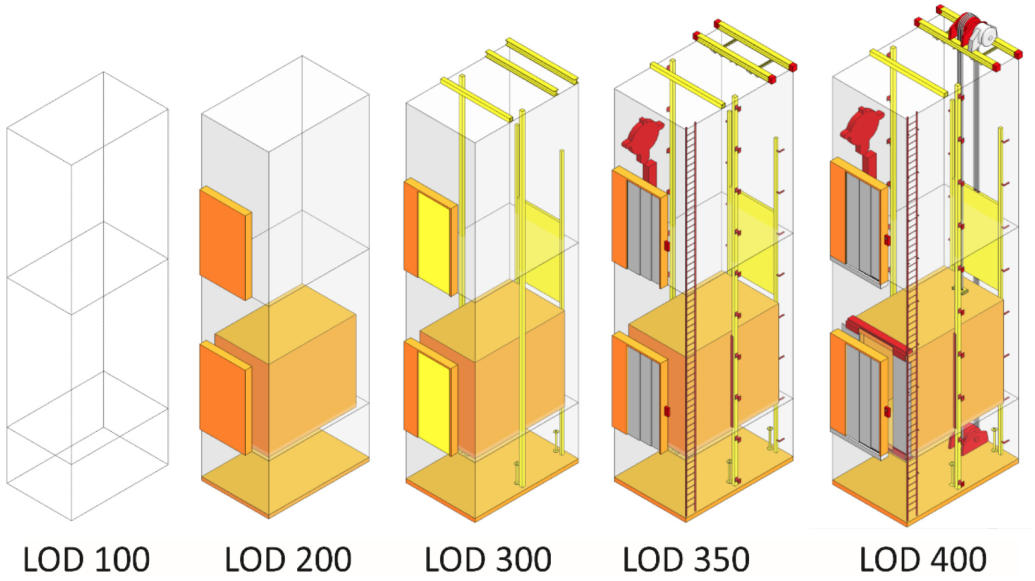


Figure 1.3: Illustrating the progression of design across the LODs with an example of Elevator. This example was modeled according the BIMForum's specification (Abualdenien & Borrmann, 2022; BIMForum, 2019)

Figure 2.3 clarifies the progression of the design across the LODs with an elevator example. This example was modeled according to the BIMForum's specification (Abualdenien & Borrmann, 2022; BIMForum, 2019). At LOD 100, the elevator's material, dimensions, and even location are still flexible. LOD 200 provides a generic envelope representation and travel paths. At LOD 300, any associated equipment and structural support are modeled. Sizing, tracks, rails, and access zones are modeled and fixed at LOD 350. Finally, all connections, supports, framing, and other supplementary components are modeled at LOD 400 (BIMForum, 2019).

Since then, many efforts have been invested worldwide to develop a suitable and practical standard assisting the delivery of the industry projects. As a results, various guidelines were published providing comprehensive spreadsheets, descriptions, and illustrations of the expected information at every level. Chapter 2 reviews the existing guidelines worldwide in detail.

### 1.2.4 Vagueness visualization

When designing a building, each discipline understands and evaluates the proposed design from a different perspective; for example, the architect is concerned with the building's footprint, facade, and interior layout, energy engineers look at the heat loss and gain of the used materials, and structural engineers are interested in the effectiveness of the structural system. In this process, visualization is an essential component of the established workflows and exchange scenarios, including communicating design intent, checking the integrity of partial designs, and evaluating design options. The interactive visualization of 3D building models provides great support for many building design and engineering tasks.

However, despite the insufficient information in the early design phases, a BIM model appears precise and certain as existing BIM authoring tools lack methods for depicting vagueness simultaneously with building models and interacting with those depictions in an understandable way. The current visualization would wrongly suggest that the design is more elaborate than it actually is, which can lead to false assumptions and model evaluations, affecting the decisions made throughout the design phases. Hence, it is crucial to communicate the incorporated vagueness along the representation of building models.

Visually representing information vagueness encourages using domain experts' knowledge, which assists with carrying out tasks more effectively (Card, 1999; Munzner, 2014). Conveying the quantity of vagueness in the information is crucial for making rational conclusions (Deitrick, 2007; Griethe & Schumann, 2005). This particularly applies to the architectural design and engineering of buildings. Visual communication of information has advantages over verbal description, as humans process visual information with high-efficiency (Smith Mason et al., 2017), which can improve the estimates made (Greis et al., 2018).

Conventional construction planning relies heavily on using different drawing scales to represent geometric information on a suitable level of detail and degree of preciseness (Farrelly, 2008). The produced drawings evolve from sketches depicting the rough shape of the building and the floor plans to detailed workshop drawings presenting the precise design of individual components, connection points, etc. Accordingly, a drawing's scale directly implies the degree of abstraction, vagueness, and maturity of the design information conveyed, and typically, specific scales are requested in specific design phases. As the concept of scale cannot be applied to 3D digital building models, an analogue concept is necessary for communicating a design's reliability and intentions.

Developing vagueness visualization approaches for the AEC industry is challenging and requires understanding how individual domain experts perceive building information. This is crucial for understanding how the knowledge of the vagueness would influence the decisions taken.

### 1.2.5 Design knowledge & detailing decisions

The building design process is rich with numerous implicit design decisions and domain knowledge. Design concepts are usually realized through modeling and detailing the individual elements while considering their relative context, including their internal location and connections, as well as their relative location to the surrounding environment. Detailing decisions could involve one or multiple elements, such as deciding on the windows' material or junction types between walls, columns, and slabs (Schneider-Marin & Abualdenien, 2019).

Such detailing decisions significantly influence the performance of the resultant building design from various aspects, including energy consumption, cost, and comfort. Hence, designers typically explore and evaluate the potential design options during the different phases (Châteauvieux-Hellwig et al., 2022; Exner et al., 2019). Furthermore, although requirements and boundary conditions vary from one construction project to another, architects and engineers steer the designs' development based on the knowledge gained from previous successful projects. In this context, similar building information and dependencies are evaluated, aiming to achieve similar functions or performance.

Detailing rationale includes the context information necessary to apply such a detailing pattern, such as the element's relative position to the storey's entrance and the building's orientation (considering its sun path during the different seasons). However, detailing decisions are currently embedded in building models, and detailing rationale is implicit in the designers' minds, hindering their proper management and reuse.

## 1.3 Aim and scope of research

This thesis aims to provide a holistic methodology for preserving the consistency of building models starting from the early design phases (incorporating the associated uncertainties) and facilitate formally capturing and reusing detailing decisions. The previous sections emphasized the main challenges of managing design information and knowledge across the design phases. This section presents the identified and tackled gaps within the scope of this thesis:

1. **Inconsistency of guidelines for defining information requirements**: Various guidelines have been published to deliver a standard for practitioners for defining design requirements. For example, the LOD, is a popular concept for defining the content and reliability of a model at a certain point during the design process. It clarifies the necessary efforts and milestone deliverables, addressing the expectations of the involved domain experts. However, due to the inconsistency among the different guidelines published worldwide, the content of the expected deliverable is debatable and prone to personal interpretations, although such information is part of the Project execution Plan (BxP) and is contractually binding. Additionally, the collaboration among the different disciplines heavily relies on the quality and completeness of the exchanged information to perform the different tasks. A detailed investigation of the existing guidelines, including an analysis of their deviations, is presented in Chapter 2.

2. **Lack of means for incorporating and communicating information vagueness**: As the knowledge about the design solution increases through the design phases, the unknown and uncertain information decreases. Currently, designers do not include indicators of information uncertainty in their designs. Additionally, simulation experts substitute missing and uncertain information by default values from literature or their domain knowledge. As 3D building models appear precise and certain during those phases, exchanging designs and performing simulations could cause false assumptions and evaluations since any associated uncertainties are not explicitly communicated. Existing concepts for the specification of project requirements and deliverables, such as the established design phases and LODs, do not provide the flexibility required for incorporating design uncertainties through the gradual development and exchange of building information (more investigations are provided in Chapter 3). Furthermore, as visualizations are an essential tool during design, analysis, and decision making, depicting the corresponding uncertainties over the 3D representations could improve the quality of collaboration (see chapter 4).

3. **Preserving the consistency of building models across design phases**: As the design process accommodates various perspectives and boundary conditions, decisions taken in one phase should be maintained and build upon those made in the previous phases. Each design phase addresses new challenges, bringing new knowledge and information. Hence, addressing those challenges should ensure the refinement consistency of models, maintaining the applicability of previously made decisions or performed calculations and simulations. Otherwise, project participants should reconsider the reliability of the design previously made design concepts and calculations.

4. **Validation of geometric detailing**: The degree of the modeled and delivered semantic and geometric BIM information is contractually binding and crucial when exchanging the different discipline models. This far, numerous commercial and open-source BIM

tools can automatically validate the semantic information. However, the automatic validation of the modeled geometry for fulfilling the expected detailing requirements is a complex and unsolved task. In current practice, geometric detailing is described with illustrations and textual descriptions. Hence, it is checked manually, which is time-consuming and open to different interpretations by domain experts.

5. **Documentation and reuse of design decisions**: Design decisions highly influence the cost and performance of the final design. Typically, architects and engineers employ their domain knowledge to reuse and detail building elements in a way that fulfills the current needs and boundary conditions. Detailing patterns are described through building information and the rationale behind them. So far, buildings are represented in complex data models, and the rationale behind the decisions is not documented or captured. Hence, there is a need for a methodology to capture detailing decisions and facilitate their reuse in future designs.

## 1.4 Research questions, hypotheses, and concept

Based on the identified research gaps and the involved challenges, the following research questions arise:

1. Which standards for specifying design maturity and detailing requirements in building models do exist and what are their distinctive features? Would they be able to represent the information vagueness?
2. Information vagueness has numerous types and representations. How can the vagueness in the building design information be represented?
3. Which visualization techniques are effective for depicting information vagueness for the use cases of the AEC industry?
4. How could the refinement consistency of building models from one design phase to another be formally described and preserved?
5. Which geometric features are capable of representing the detailing complexity of building elements? Which techniques can classify BIM objects based on their geometric complexity?
6. How could the rationale behind design decisions be captured?
7. How could detailing patterns be formally represented? Which techniques are capable of capturing detailing patterns and transferring them to new projects?

To approach these research questions, the following hypotheses and objectives were derived:

1. **Hypothesis**: Practitioners use common guidelines as a communication language for design requirements and deliverables.
   **Objective**: Identify the most popular and common guidelines used for the specification of design requirements and milestones' deliverables, and evaluate their capabilities.

2. **Hypothesis**: A formal specification of requirements, including the associated vagueness, can assist collaboration and decision-making during design phases.
   **Objective**: Development of a methodology for representing the different kinds of information vagueness and integrating them with design requirements in a formal way.

3. **Hypothesis**: Visualization techniques are capable of conveying the vagueness associated with building models at different rates of effectiveness assisting decisions in the different use cases.
   **Objective**: Development and evaluation of visual approaches for conveying the vagueness incorporated in BIM models taking into consideration the needs of the different use cases, such as scale and information details.

4. **Hypothesis**: Developing designs from one phase to another narrows down the variety of possible solutions while preserving previously taken decisions. Consistency of a model's refinement comprises the gradual determination of semantic properties, geometric dimensions and positioning, and maintaining the topological network.
   **Objective**: Development of a framework for representing and preserving the refinement of BIM information across design phases.

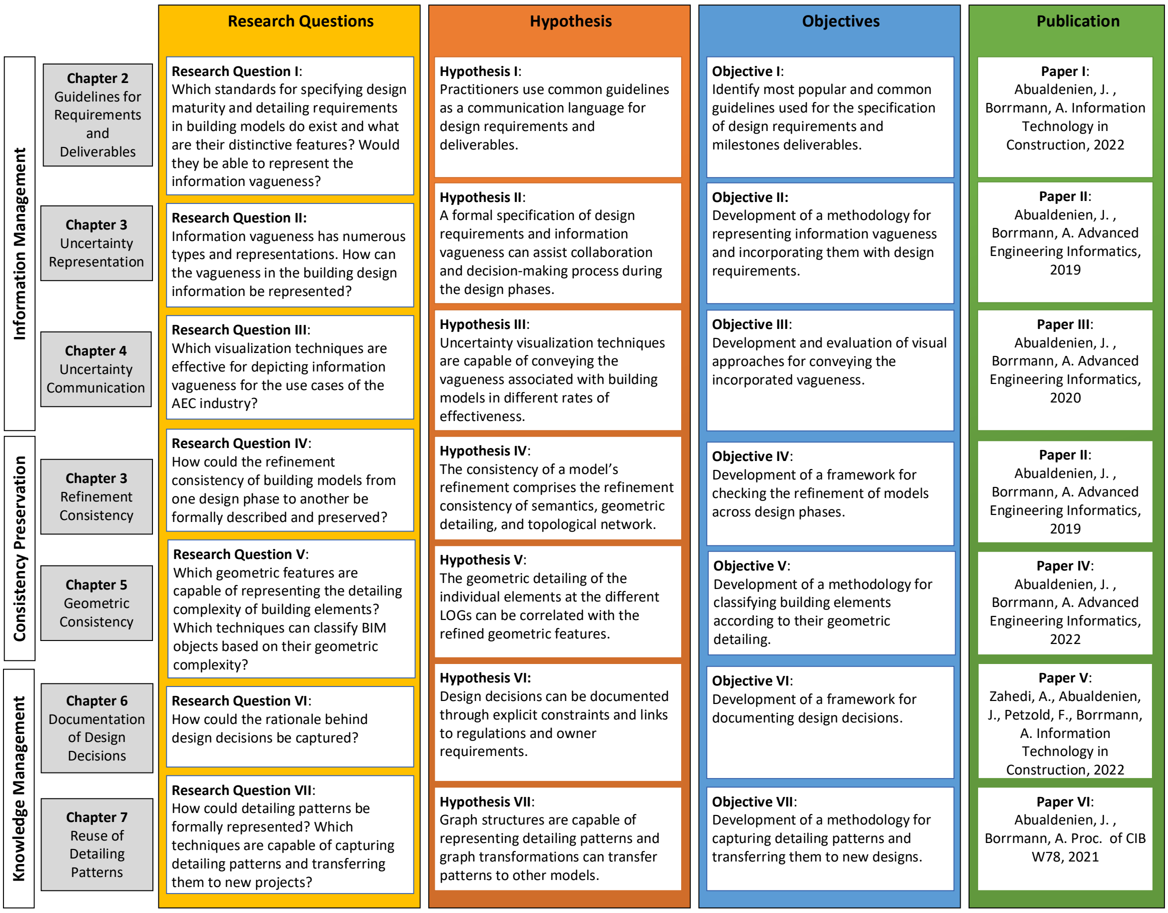
5. **Hypothesis**: As the geometric detailing of the individual elements at the different milestones is described in LOD guidelines, the geometric complexity correlates with the refined geometric features at every level.
   **Objective**: Identification of a representative set of geometric features that are capable of describing the elements' geometric complexity. Development of a framework for classifying elements according to their geometric complexity.

6. **Hypothesis**: Design decisions can be documented through explicit design constraints and links to specific regulations and owner requirements.
   **Objective**: Development of a methodology for representing design intentions through constraints. Additionally, assist in documenting design rationale through querying and linking regulations and requirements (in natural language) to design information.

7. **Hypothesis**: Graph structures are flexible enough to represent detailing patterns and their rationale. Additionally, graph rewriting systems can transfer detailing patterns from one design to another in a parametric way.
   **Objective**: Develop a graph representation that is flexible enough for capturing design

decisions. Developing a graph rewriting system that is coupled with a BIM-authoring tool to facilitate its evaluation.

The derived hypothesis and objectives are addressed in dedicated chapters as shown in Figure 1.4.



| | | **Research Questions** | **Hypothesis** | **Objectives** | **Publication** |
|---|---|---|---|---|---|
| **Information Management** | **Chapter 2** Guidelines for Requirements and Deliverables | **Research Question I**: Which standards for specifying design maturity and detailing requirements in building models do exist and what are their distinctive features? Would they be able to represent the information vagueness? | **Hypothesis I**: Practitioners use common guidelines as a communication language for design requirements and deliverables. | **Objective I**: Identify most popular and common guidelines used for the specification of design requirements and milestones deliverables. | **Paper I**: Abualdenien, J. , Borrmann, A. Information Technology in Construction, 2022 |
| | **Chapter 3** Uncertainty Representation | **Research Question II**: Information vagueness has numerous types and representations. How can the vagueness in the building design information be represented? | **Hypothesis II**: A formal specification of design requirements and information vagueness can assist collaboration and decision-making process during the design phases. | **Objective II**: Development of a methodology for representing information vagueness and incorporating them with design requirements. | **Paper II**: Abualdenien, J. , Borrmann, A. Advanced Engineering Informatics, 2019 |
| | **Chapter 4** Uncertainty Communication | **Research Question III**: Which visualization techniques are effective for depicting information vagueness for the use cases of the AEC industry? | **Hypothesis III**: Uncertainty visualization techniques are capable of conveying the vagueness associated with building models in different rates of effectiveness. | **Objective III**: Development and evaluation of visual approaches for conveying the incorporated vagueness. | **Paper III**: Abualdenien, J. , Borrmann, A. Advanced Engineering Informatics, 2020 |
| **Consistency Preservation** | **Chapter 3** Refinement Consistency | **Research Question IV**: How could the refinement consistency of building models from one design phase to another be formally described and preserved? | **Hypothesis IV**: The consistency of a model's refinement comprises the refinement consistency of semantics, geometric detailing, and topological network. | **Objective IV**: Development of a framework for checking the refinement of models across design phases. | **Paper II**: Abualdenien, J. , Borrmann, A. Advanced Engineering Informatics, 2019 |
| | **Chapter 5** Geometric Consistency | **Research Question V**: Which geometric features are capable of representing the detailing complexity of building elements? Which techniques can classify BIM objects based on their geometric complexity? | **Hypothesis V**: The geometric detailing of the individual elements at the different LOGs can be correlated with the refined geometric features. | **Objective V**: Development of a methodology for classifying building elements according to their geometric detailing. | **Paper IV**: Abualdenien, J. , Borrmann, A. Advanced Engineering Informatics, 2022 |
| **Knowledge Management** | **Chapter 6** Documentation of Design Decisions | **Research Question VI**: How could the rationale behind design decisions be captured? | **Hypothesis VI**: Design decisions can be documented through explicit constraints and links to regulations and owner requirements. | **Objective VI**: Development of a framework for documenting design decisions. | **Paper V**: Zahedi, A., Abualdenien, J., Petzold, F., Borrmann, A. Information Technology in Construction, 2022 |
| | **Chapter 7** Reuse of Detailing Patterns | **Research Question VII**: How could detailing patterns be formally represented? Which techniques are capable of capturing detailing patterns and transferring them to new projects? | **Hypothesis VII**: Graph structures are capable of representing detailing patterns and graph transformations can transfer patterns to other models. | **Objective VII**: Development of a methodology for capturing detailing patterns and transferring them to new designs. | **Paper VI**: Abualdenien, J. , Borrmann, A. Proc. of CIB W78, 2021 |

Figure 1.4: Scope of the publications presented in the following chapters.

## 1.5 Overall concepts

Based on the formulated objectives, the methodology presented in Figure 1.5 is deemed adequate for formally managing the design process and its deliverables. The proposed concept is based on three main processes, information management, consistency preservation, and reuse of detailing patterns.

The Information management part is concerned with formally representing design requirements and mitigating uncertainty through its integration and visualization on building information. The aim is to make domain experts aware of any unknown and imprecise information in

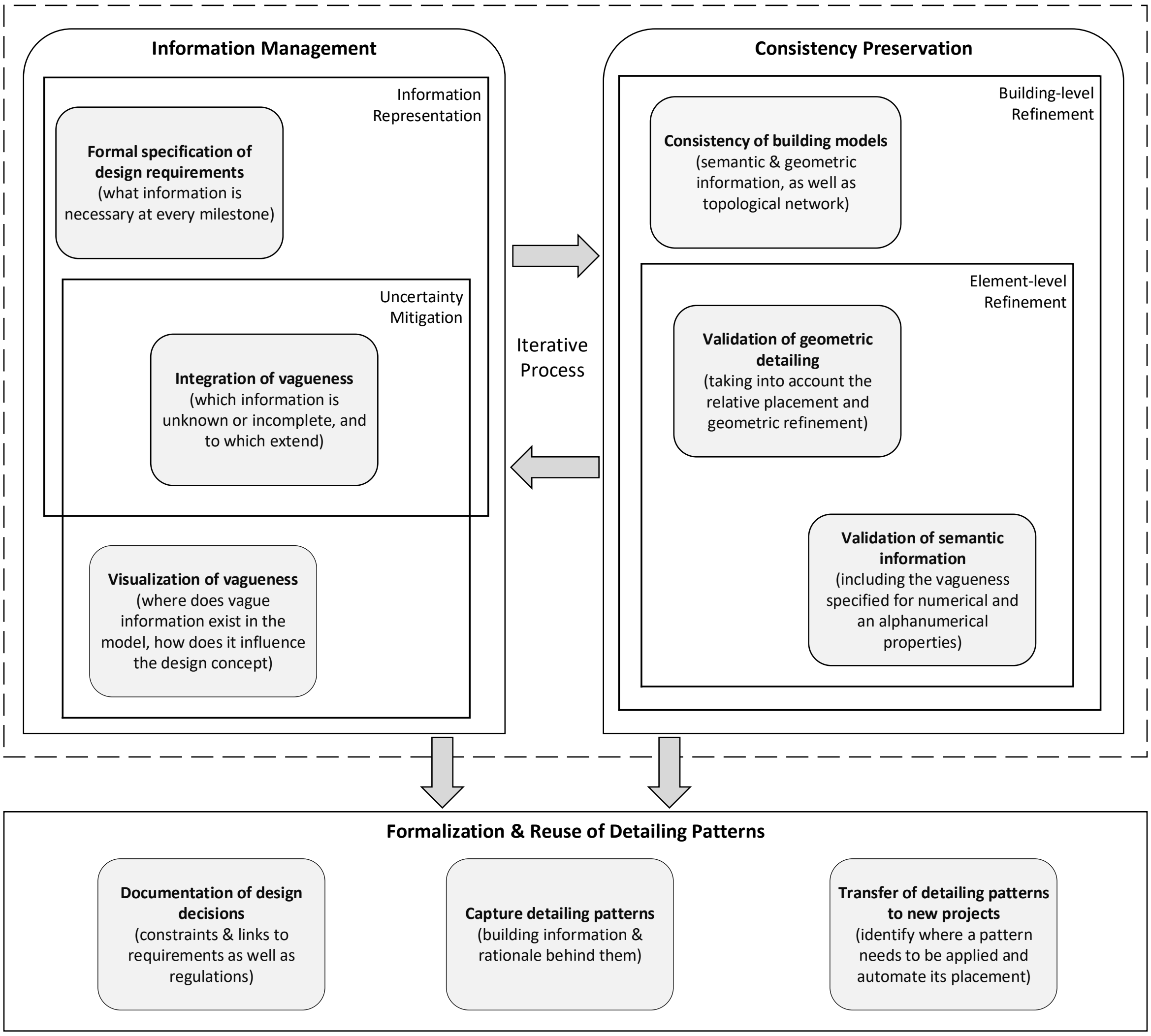


Figure 1.5: Thesis overall concept, covering three main areas: Managing information, preserving consistency of the refined designs, and finally capturing design knowledge and reusing detailing patterns in new projects.

BIM-authoring tools as well as simulation and analysis tools as the potential uncertainties are incorporated within the BIM information. Subsequently, requirements and building information are used as input for consistency preservation. Consistency preservation starts with validating the refinement of the semantic and geometric information of the individual elements and then propagates for validating the consistency of the overall design, taking into account its state from the previous design phases. This process is iterative, enhancing the model content across the design phases. Once a detailing pattern is developed, the involved information can be captured for documentation or reuse in new projects.

The proposed methodology addresses communicating design requirements, mitigating uncertainties, assisting the decision-making process, and ensuring the quality of the developed designs.

### 1.5.1 Information management

Building information is currently managed in two main approaches: (1) closed-BIM, where the data is represented using a vendor-specific format, limiting the interoperability with external tools from different vendors. (2) open-BIM, following a standardized vendor-neutral format, providing a ground for establishing connected workflows through seamless interoperability across diverse tools.

Explicitly specifying design requirements along their LOD and vagueness requires flexibility in associating building information with additional information. Besides the involved efforts in establishing agreements of standardization committees, extending either closed- or open-BIM models for including this information is complex as they are project-specific and demand a flexible framework.

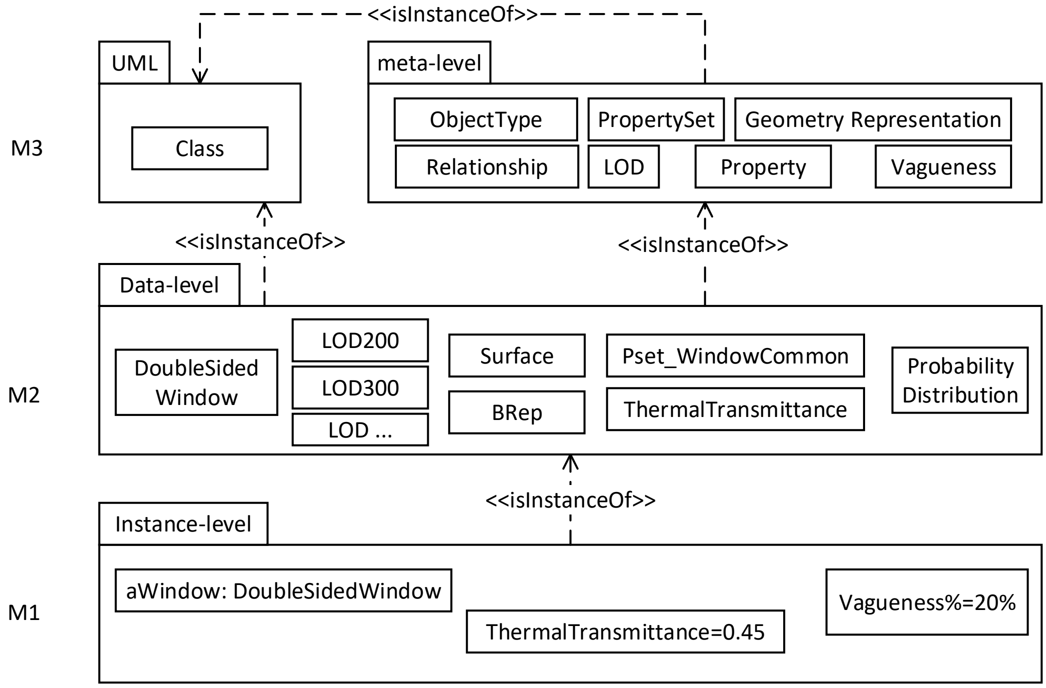


Figure 1.6: Illustrating the application of meta-modeling for specifying design requirements and incorporating vagueness to the different object types.

Hence, this research proposes employing meta-modeling techniques, which are capable of flexibly extending building information by providing multiple abstraction layers[1]. Figure 1.6 illustrates the application of meta-modeling in the context of BIM models on three layers. First, *M3 (meta-level)* defines the main specifications of the object types, including the class definition of properties, geometry representation, LOD, and vagueness. At *M2 (data-level)*, the LOD requirements, including geometric representation and semantics, are specified in

---

[1] Object Management Group (2019). Meta Object Facility (MOF) Core Specification 2.5.1. https://www.omg.org/spec/MOF/

addition to the associated vagueness and relationships to other object types. This layer acts as the requirements level of each object type. Afterwards, *M1 (instance-level)* represents an instance of the object type defined at *M2*, specifying the concrete values.

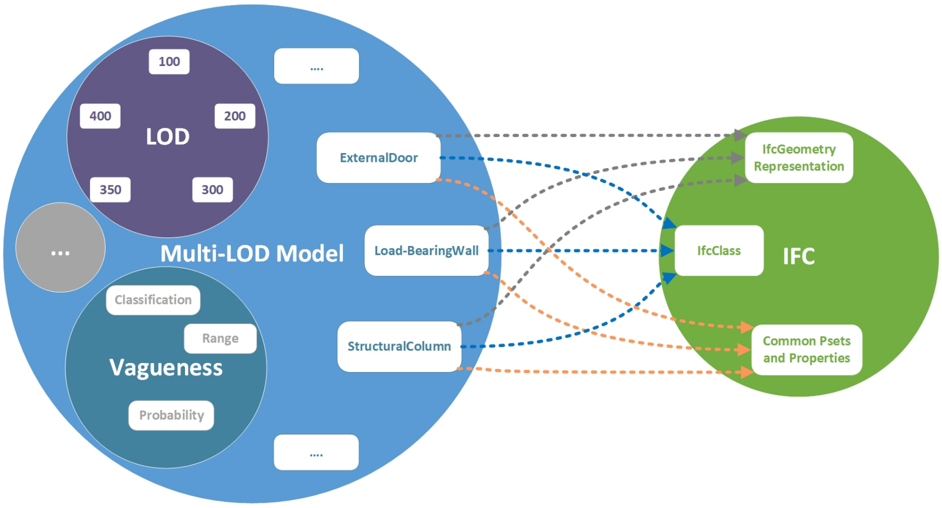


Figure 1.7: Highlighting the links between the objects defined in the multi-LOD meta-model and IFC.

Figure 1.7 elaborates further on the connection between the multi-LOD model and IFC. Each object is linked to an IFC class; here, the meta-model makes it possible to create more specialized objects, such as *External Door* or *Load-bearing Wall*. In IFC, such object types are only distinguishable through the properties attached to them, where they share the same IFC class. The meta-model also provides links to common Ifc geometry representations, property sets and properties, and IFC relationships. As a result, the individual objects of the IFC model are linked to objects from the meta-model, which are associated with LOD requirements and potential vagueness.

The LOD component within the multi-LOD model formalizes the structure of the widely-established LOD concepts (outcome of Research Question 1). Additionally, the multi-LOD model lays the necessary foundation for formally representing the vagueness associated with building information (answering Research Question 2).

### 1.5.2 Vagueness visualization

In the AEC industry, visualization is essential to the established workflows and exchange scenarios. The interactive visualization of 3D building models provides great support for many tasks related to building design and engineering. At the same time, understanding what is precise and complete and accounting for design uncertainty is critical to reason about the visualized building information effectively. However, as depicted in Figure 1.8, the vagueness associated with BIM information is not visualized in the current BIM-authoring tools.

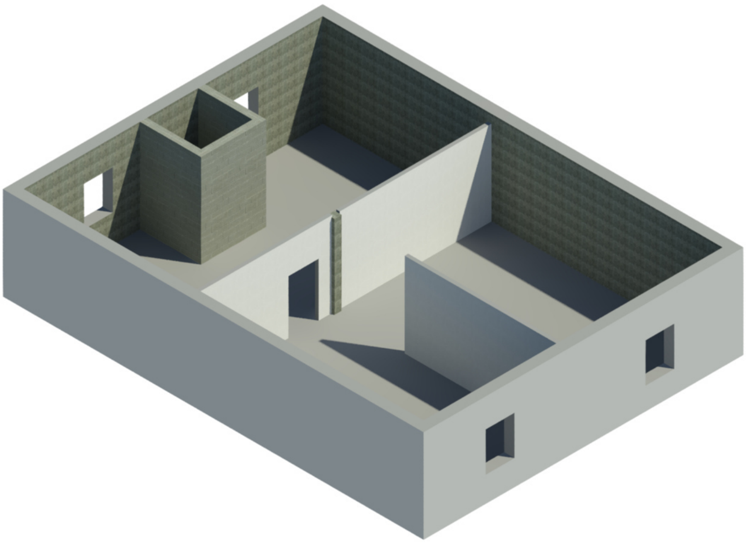


Figure 1.8: An example of how building information is visualized in the current BIM-authoring tools, even at the early design phases.

Vagueness visualization provides high communicative efficiency, offering an easier-to-search representation that simplifies recognition and inference. This research explored numerous visualization approaches (Objective 3), leveraging the specified vagueness on the previously proposed meta-model (Research Question 2) as an input to those visualizations. Figure 1.9 demonstrates one of the approaches on a simple storey. In comparison to Figure 1.8, the color coding, transparency, and border line styles convey the incorporated geometric and semantic vagueness (More details and additional approaches are described in Chapter 4). The developed visualization techniques were assessed for effectiveness in assisting multiple different use cases on multiple scales (answering Research Question 3).

### 1.5.3 Consistency preservation of models across design phases

Developing building models from one phase to another involves the refinement of the building's topology as well as the semantic and geometric information of the individual elements. The information of each building element impacts its function, which is interconnected with the design's validity and performance, including its structural stability, fire-safety concept, and multiple requirements. Considering the collaborative nature of building projects, modifications at an advanced phase should be adequately controlled in order to avoid any unexpected implications to the overall design. Additionally, the quality of the exchanged deliverables must be checked, taking into account the expected information maturity.

Therefore, this research proposes a methodology for checking the refinement consistency of building models (answering Research Question 4), including their geometric and semantic

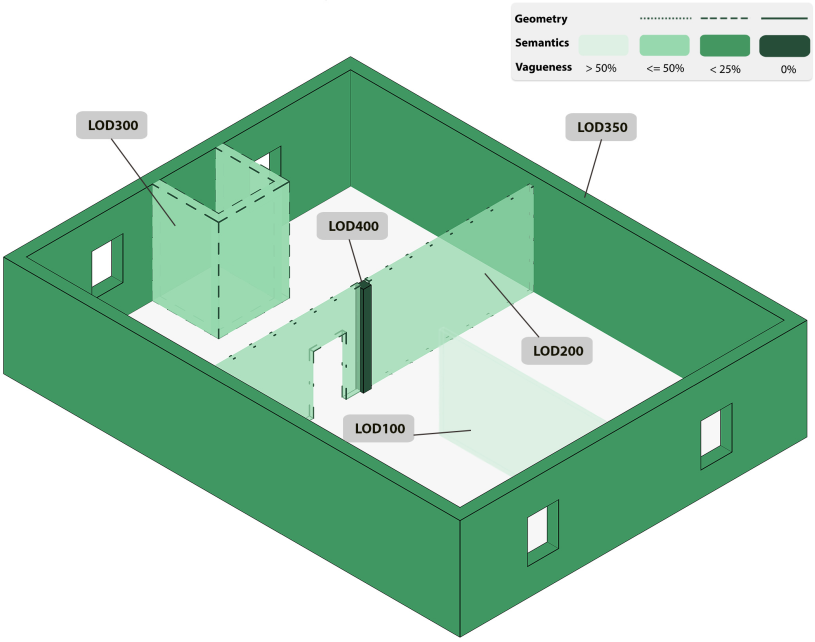


Figure 1.9: Illustrating one approach of the proposed visualizations for communicating the associated information vagueness.

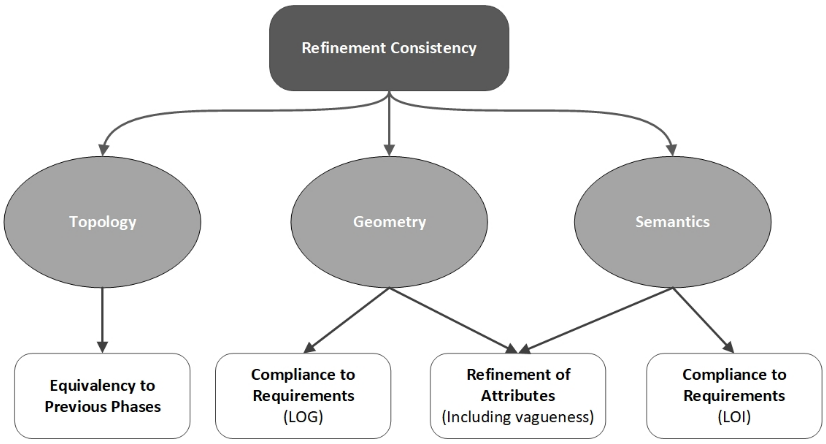


Figure 1.10: High-level overview of the proposed concept for preserving refinement consistency.

information as well as topological relationships across the design phases (as illustrated in Figure 1.10).

### 1.5.4 Classification of Level of Geometry (LOG)

Compliance with the geometric requirements is one part of the consistency preservation methodology presented in the previous section. Thus far, classifying BIM objects according to their geometric detailing (i.e., LOG) is an unresolved task. Hence, this research investigates which features are capable of describing the objects' geometric complexity. The hypothesis is that the geometric detailing at the different LOGs can be correlated with the refined geometric features.

As depicted in Figure 1.11, this research investigates a feature-engineering approach, where a LOG dataset is modeled according to the most common LOD specifications (described in detail in Chapter 5). Then, multiple geometric features are extracted, representing the detailing of each building element (Objective 5). Afterward, to prove that the identified features are capable of describing the degree of detailing, the training of multiple Machine Learning (ML) classification algorithms is evaluated for correctly classifying newly provided building elements (providing an answer to Research Question 5). The complete framework is discussed in detail in Chapter 5.

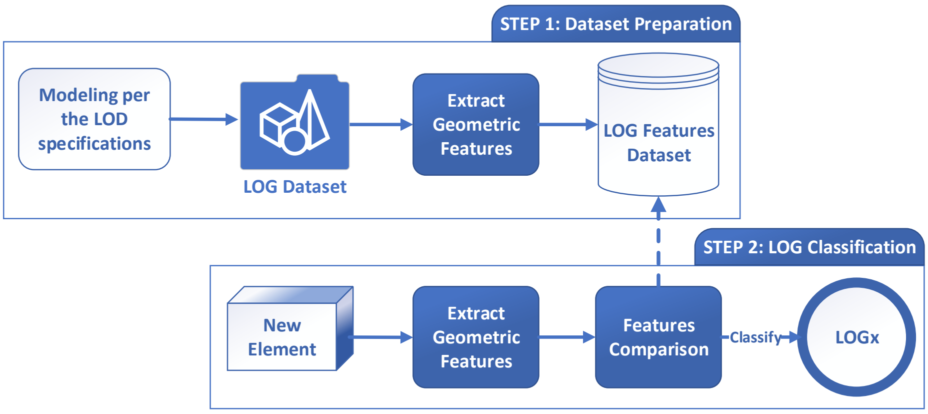


Figure 1.11: The proposed framework for validating the geometric detailing of building elements for compliance with the well-established LOG specifications.

### 1.5.5 Documentation & reuse of design decisions

Building information is the outcome of various decisions made during the design process. Therefore, additional contextual information is necessary to document and capture design intentions and the functional dependencies between building elements. In this research, we propose two methods for documenting and capturing design knowledge (Objective 6, see Figure 1.12). First, enriching BIM information with semantic and spatial constraints. Such constraints would capture the connection points between the different elements, including the relationships between their semantics. On the other hand, to include an additional context and reasoning behind the decisions made, we propose providing the ability to link building information to specific parts of owner requirements and regulation documents (e.g., building codes). As those documents are written in natural language, we propose using Natural Language Processing (NLP) techniques to facilitate querying and linking geometric or semantic properties to relevant phrases from those documents (Research Question 6).

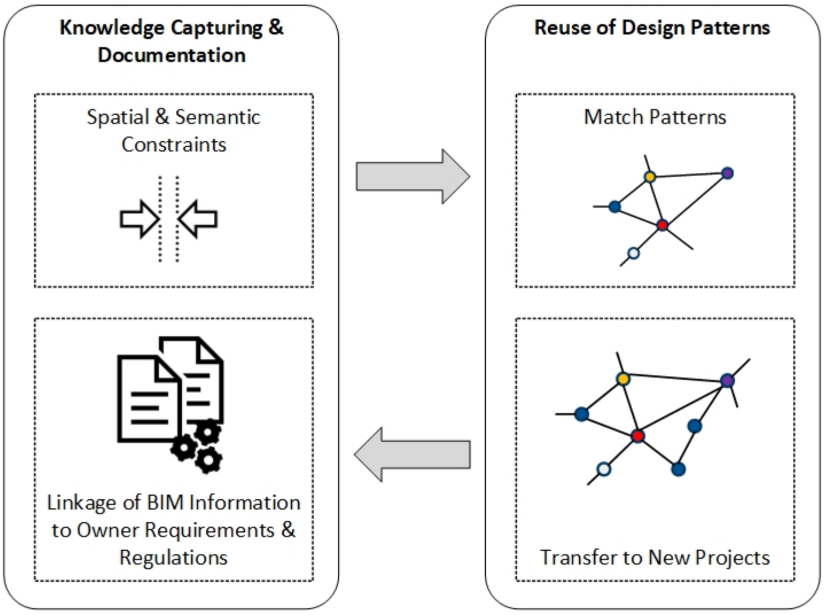


Figure 1.12: An overview of followed approaches for managing and reusing design knowledge.

Once design decisions can be documented, a framework is necessary for capturing and transferring detailing patterns to new projects (Objective 7). The proposed approach utilizes graph structures and Graph Rewriting Systems (GRS) for providing the necessary flexibility. In more detail, designers formulate a detailing pattern by selecting building elements, spaces, their relationships, and context information. When formulating a pattern, designers specify which information belongs to detailing and which belongs to reasoning when to apply it (matching pattern). A detailing pattern could include information of one or multiple elements. When designers detail a new design option or a new project, they can browse and select one of the stored patterns to be matched within the new model. Once matched, its corresponding nodes, edges, and their properties will be rewritten with information from the rewriting pattern, producing a detailed BIM model graph. Finally, the detailed graph is transformed back into a BIM model inside the BIM-authoring tool (providing an answer to Research Question 7).

## 1.6 Structure of the thesis

This chapter provided a general introduction to the current state-of-practice, identified challenges, and the derived hypothesis and objectives. Additionally, an overview of the developed concept was presented, describing the techniques employed behind each of the followed approaches. This cumulative thesis comprises several following chapters. Each chapter thoroughly emphasizes the existing research gap, then presents and evaluates the developed approaches.

As the overall aim of this research is the consistent management and evaluation of building information across the design phases, Chapter 2 presents the outcome of a critical literature review of the well-established standards worldwide for managing information requirements and describing the maturity of BIM models. This chapter highlights the advantages, limitations, as well as common misinterpretations in using those standards in the AEC industry.

Chapter 3 focuses on the collaboration among the different disciplines and the confronted challenges starting from the early phases. It emphasizes the need for more flexible and maturity-oriented milestones than the existing design phases. Additionally, it proposes a methodology for integrating the associated vagueness to requirements and building information and preserving the refinement consistency of models from one phase to another.

Afterwards, Chapter 4 evaluates and develops uncertainty visualization approaches for conveying the incorporated vagueness information in Chapter 3. The performance of multiple approaches (static and dynamic) was investigated in assisting different use cases on different scales (from a property and element level to the overall building).

To complement the consistency preservation framework presented in Chapter 3, Chapter 5 develops a framework for classifying the geometric complexity of building elements according to the well-established LOG guidelines. First, a set of geometric features, representative of their complexity, is extracted, and then the robustness of multiple ML algorithms is investigated and compared.

After managing the building information and its refinement consistency, Chapter 6 presents a framework for documenting design decisions using constraints and NLP techniques. The overall goal is to enrich building models with additional context information that is capable of capturing the reasoning behind the decisions made.

Finally, Chapter 7 presents a methodology for transferring detailing patterns from one design to another. First, it represents detailing patterns as a flexible graph representation and then leverages GRS for dynamically matching and replacing detailing patterns in new projects. All the objectives defined in this thesis are then evaluated in Chapter 8, highlighting the remaining gaps and an outlook for future research questions.

# Chapter 2

# Levels of detail, development, definition, and information need: A critical literature review

Previously published as: Abualdenien, J.; Borrmann, A.: *Levels of detail, development, definition, and information need: a critical literature review*, Journal of Information Technology in Construction 27, pp. 363-392, 2022, DOI: 10.36680/j.itcon.2022.018

## abstract

The construction industry relies on precise building information for evaluating designs performance, collaboration, and delivery. For more than a decade, the Level of Development (LOD) is the most popular concept for describing the progression of geometric and semantic information across the design phases. The LOD is a domain language that aims to establish a common understanding of what each level means to facilitate communication and defining deliverables in contracts among the project participants. However, multiple similar standards are published worldwide for a similar purpose, such as Level of Detail, Level of Definition, and Level of Information Need. However, although they are similar at first glance, in many cases, they have numerous deviations in their fundamentals. This paper investigates the differences of the LOD standards and their interpretation by the scientific community through a thorough analysis. For this purpose, 58 LOD guidelines were reviewed, and a systematic literature review of 299 peer-reviewed publications was conducted. As a result, existing trends in using the LOD in research and the most wide-spread LOD naming conventions and specifications were identified. Additionally, the results highlight 16 common use cases for applying the LOD.

## 2.1 Introduction

The processes conducted in the design and engineering of built facilities are typically multidisciplinary and comprise various activities. Such activities include defining requirements, modeling design concepts, and evaluating their compliance and performance. These activities are connected and interdependent, representing the workflow necessary for delivering a functional asset with an adequate quality throughout its life cycle.

Building Information Modeling (BIM) is method that uses digital information across the entire lifecycle of the facility under consideration. It is based on a comprehensive digital representation of the individual geometric and spatial elements, capturing their functional characteristics and dependencies (Borrmann et al., 2018b). The utmost use of BIM is to provide a holistic and reliable basis for decision-making throughout the life cycle of a construction project. BIM facilitates collaboration across the different domain experts, which assists in reducing project costs and delivery time (Cheung et al., 2012; Kolltveit & Grønhaug, 2004; Zanni et al., 2019). Typically, each domain expert has its own unique considerations, processes, and BIM tools. Hence, realizing the full potential of BIM requires a clear agreement of the modeled and exchanged information throughout the projects' life cycle.

The design process involves a set of interrelated activities that result in increasing the design solution knowledge (or reducing the uncertainty). A design solution is gradually elaborated, refined and detailed as the design evolves. Accordingly, the quantity and quality of the available information increases as the design becomes more mature.

It is crucial for the overall collaboration and coordination among the project participants to unambiguously define *what* information should be available at what milestone (*when*), which actor is supposed to deliver it (*who*), and to which end it is required *(purpose)*. The exchange of BIM data within the Architecture, Engineering and Construction (AEC) industry is prescribed in legal contracts where the information for each specific model is specified, meaning that a common legal framework for organizing BIM data is required (Sacks et al., 2018).

Conventional construction planning heavily relies on the use of different drawing scales for defining geometric information needs regarding a suitable level of detail as well as a certain degree of maturity and preciseness (Farrelly, 2008). The produced drawings evolve from sketches depicting the rough shape of the building and the floor plans to detailed workshop drawings presenting the precise design of individual elements, connection points, etc. Accordingly, a drawing's scale directly implies the degree of abstraction, vagueness and maturity of the design information conveyed, and typically, specific scales are requested in specific design phases. The concept of maturity is an essential requirement for supporting evolving design processes. As scale (as an indicator for maturity) cannot be applied for digital building models, an analogue concept is necessary.

In the scope of BIM, since more than a decade, multiple initiatives have been established with the focus of creating a consensus about what information should exist during the development of building elements during the life-cycle of a project (AIA, 2013b; BIMForum, 2019; Department of Veterans Affairs (VA) Office of Construction & Facilities Management (CFM), 2010; Maier, 2015; NATSPEC, 2013b; PROGETTIAMOBIM, 2018; VBI, 2016; VicoSoftware, 2005). The first initiative was with introducing the *Level of Detail (LoD)* for BIM objects (VicoSoftware, 2005). Although at that time it was new in this AEC industry, the *Level of Detail* concept is a topic that has been discussed in computer graphics (Luebke, Reddy, Cohen, et al., 2003) for a long time before. It is used to bridge complexity and rendering performance by regulating the amount of detail used to represent the virtual world. The LoD has been adopted and refined by the American Institute of Architects (AIA) to become the *Level of Development (LOD)*, referring to the completeness reliability of the building elements information (AIA, 2013b).

In computer graphics, the LoD concept is mainly concerned with rendering the geometrical detailing (from the visualization point of view, i.e. it does not provide information about the degree of elaboration and reliability of the model information). In the AEC industry, the LOD represents the availability and reliability of the geometric and semantic information, which takes into account the incremental availability of information during the design process. In addition to the specification of the geometric elaboration, it also includes requirements for the attribution, i.e. for the alphanumeric information to be specified. In contrast to the LoD, the LOD also determines the reliability of the geometric and alphanumeric information stored in the model element. This meta-information is an important basis for collaborating with other planning disciplines and for the assessment of the planning progress by the client and the construction companies.

The LOD is an essential part of any BIM project execution plan (BxP) (BIMForum, 2020a) and is contractually binding in most cases (Abualdenien & Borrmann, 2022). It clarifies the necessary efforts and milestone deliverables, addressing the expectations of the involved domain experts. For more than a decade, practitioners rely on the LOD terminology to specify which information they need to deliver (Hooper, 2015; Leite et al., 2011; van Berlo & Bomhof, 2014). However, as the different LOD specifications are inconsistently defined (Bolpagni & Ciribini, 2016; Gigante-Barrera et al., 2018; van Berlo & Bomhof, 2014), each practitioner has a different interpretation of what a specific LOD means and which information should be present in the model (Bolpagni & Ciribini, 2016; Leite et al., 2011; van Berlo & Bomhof, 2014). Bolpagni, 2016 was the first to highlight the differences among the LOD guidelines through a literature review, providing an analysis for their timeline and comparing their applicability. The results lay down the necessary basis for conducting this study.Such inconsistencies cause severe miscommunication and additional expenditure, which increases project risks (Abualdenien & Borrmann, 2022; Hooper, 2015; Leite et al., 2011; van Berlo & Bomhof, 2014).

This is the exact motivation behind the research presented in this paper. The aim is to highlight any deviations, misconceptions, and misinterpretations practitioners are confronted with when following the definitions provided by the variety of LOD specifications. For this purpose, a systematic literature review is conducted, where the most widespread LOD standards are reviewed and then their usage by practitioners is evaluated from multiple aspects. The contributions of this paper are threefold: (1) highlighting the differences between the different

LOD standards and their applicability during a projects' life cycle, (2) providing insights on how practitioners interpret and apply the LOD concept in research, and (3) identifying common misinterpretations and use cases, and highlight common needs. Those contributions assist in unifying the usage of the different LOD specifications and emphasize the necessity of carefully considering the use of LODs in a way that complies with the intended purpose.

The paper is organized as follows: Section 2.2 presents the methodology followed in this literature review and the derived research questions. Section 2.3 provides an overview of the existing LOD guidelines and standards, describing their followed concepts and highlighting their deviations. Section 2.4 presents the setup and findings of the conducted bibliography analysis, and Section 2.5 discusses the outcomes of this study. Finally, Section 2.6 summarizes the findings and gives an outlook for future research.

## 2.2 Methodology & research questions

As stated above, this study aims to highlight any deviations, misconceptions, and misinterpretations practitioners are confronted with when following the definitions provided by the LOD specifications. To achieve this, we present our result in two main parts. First, 58 international LOD guidelines and standards are analyzed, focusing on their differences in terms of concept definition and references to the design process. Concept definition includes the use of concept name, abbreviation, and its applicability to the overall building model and the individual building elements. References to the design process include associations of specific LODs with specific design phases as well as linkage or analogy of the LOD levels to the scales of 2D drawings (such as 1:200 – 1:100 for LOD 200).

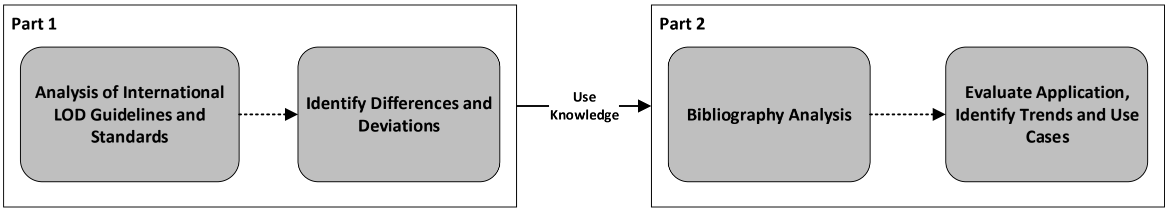


Figure 2.1: High-level overview of the methodology followed in this literature review.

The gained knowledge from the first part is subsequently used for the literature review presented in the second part. The second part focuses on the analysis of literature produced by scholars through the most prominent and influential journals in the field of *Building Information Modeling* during the period 2000 – 2020 using a set of keywords (discussed in detail in Section 2.4). Then each paper is evaluated against an LOD relevance criteria; papers not fulfilling the criteria are discarded, while others are selected to be thoroughly reviewed (more details about the relevance and exclusion criteria is provided in Section 2.4).

The conducted literature review follows the guidelines provided by Kitchenham, 2004 (Kitchenham, 2004), which comprise three main phases: planning, conducting, and reporting. Based on Kitchenham, 2004's guidelines, the following three research questions were derived, representing the focus during the assessment of the individual literature papers:

- RQ1: Is there a trend of increasing the application of the LOD concept? Is the LOD terminology required and preferable by practitioners in their different use cases? The focus here is to identify the use of LODs through time by the different domains, which emphasizes the increasing need for a standardized LOD guideline. This paper investigates this RQ through the investigation of scientific research; further investigation that is focused on industry application can complement this analysis and is planned for the future.
- RQ2: Which LOD specification is the most popular among researchers? The question identifies which of the common LOD specifications (see Section 5.2.3) is preferred by scholars in their research.
- RQ3: What are the primary use cases that require the LOD concept? This question aims to evaluate the relevant purposes for applying the LOD in the AEC industry, such as quantity take-off, visualization, simulations etc.

The content of each relevant paper is analyzed from the perspective of answering those research questions. The analysis results are then collected and used in statistical analysis, including counts of referenced terms, abbreviations, and guidelines through the years. This helps in identifying existing trends, the most common LOD standards, and the applicable use cases. Figure 2.2 illustrates the process in detail.

## 2.3 Analysis of LOD standards

The BIM industry is wealthy with different concepts and terms. Therefore, it is important to differentiate between their applicability and usage. In this section, the LOD concept, along with other related concepts, are described and discussed.

### 2.3.1 Level of Development (LOD)

The *Level of Detail* (LoD) concept is an old topic that existed in computer graphics for bridging the graphical complexity and performance by regulating the amount of detail used to visualize the virtual world (Luebke, Reddy, Cohen, et al., 2003). In 2005 VicoSoftware (Trimble, 2013; VicoSoftware, 2005) published the first LOD specification for the AEC

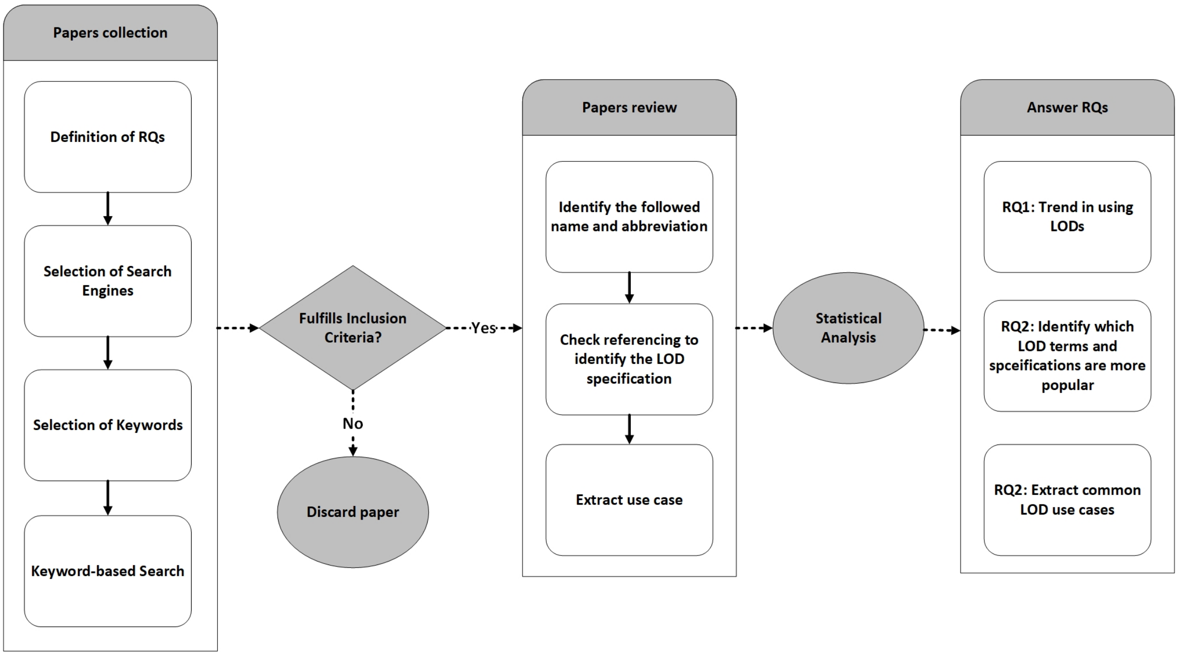


Figure 2.2: Search method: first scientific databases were searched via a set of keywords. The collected publications were then filtered by reviewing the title, abstract, and conclusions. The selected publications were then reviewed in detail with the aim of answering the previously defined research questions. Finally, a statistical analysis was performed to identify existing trends.

Industry when they introduced the *Level of Detail (LoD)*, describing the necessary semantic and geometric information with a set of five levels.

In 2008, the American Institute of Architects (AIA) built upon the LoD definitions and introduced the *Level of Development (LOD)*, which also comprises five levels starting from LOD 100 and reaching LOD 500. From 2013, the BIMForum working group has investigated the AIA definitions in detail and introduced LOD 350 (BIMForum, 2019). The BIMForum subsequently has published updated versions of the *Level of Development Specification* in a yearly cycle with the aim of providing a common understanding of the expected information at every LOD. The first level, LOD 100 (conceptual model), is limited to a generic representation of the building elements, meaning no shape information or geometric representation. The second level, LOD 200 (approximate geometry), consists of generic elements as placeholders with approximate geometric and semantic information. At LOD 300 (precise geometry), all the elements are modeled with their quantity, size, shape location, and orientation. Next, to enable the detailed coordination between the different disciplines, such as clash detection and avoidance, LOD 350 (construction documentation) is introduced, including the interfaces between all the building systems. Reaching LOD 400, the model incorporates additional information about detailing, fabrication, assembly, and installation. Lastly, at LOD 500 (as built), the model elements are a field verified representation in terms of size, shape, location, quantity, and orientation.

Figure 2.3 clarifies the progression of the design across the LODs with an elevator example. This example was modeled according to the BIMForum's specification (Abualdenien & Borrmann, 2022; BIMForum, 2019). At LOD 100, the elevator's material, dimensions, and even location are still flexible. LOD 200 provides a generic envelope representation and travel paths. At LOD 300, any associated equipment and structural support are modeled. Sizing, tracks, rails, and access zones are modeled and fixed at LOD 350. Finally, all connections, supports, framing, and other supplementary components are modeled at LOD 400 (BIMForum, 2019).

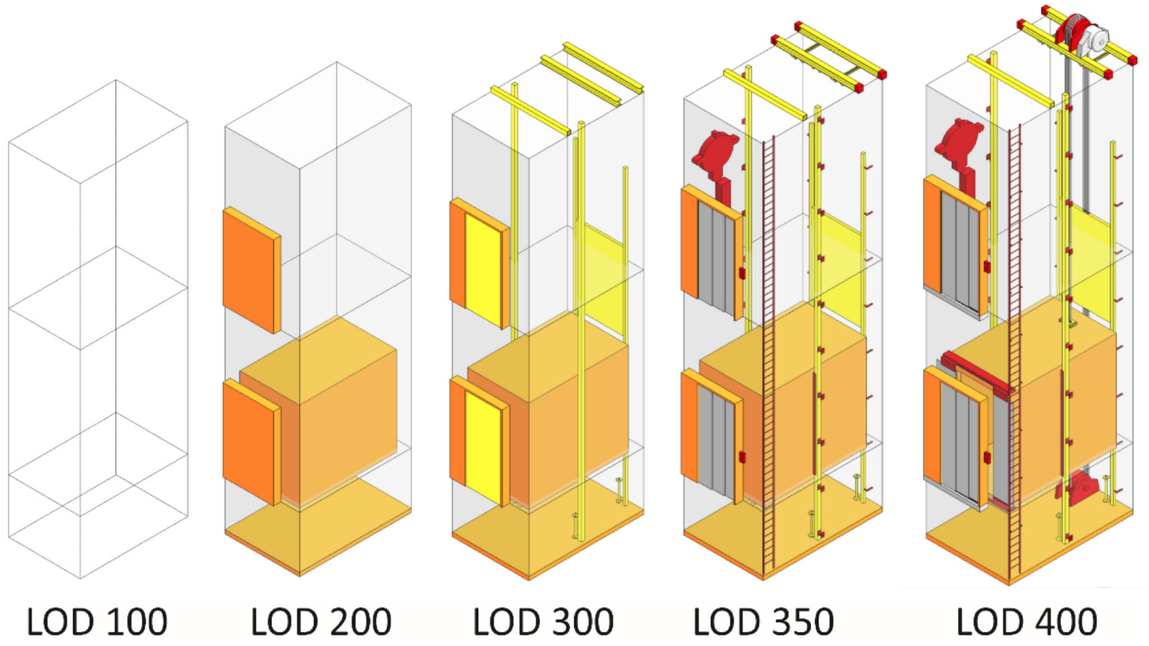


Figure 2.3: Illustrating the progression of design across the LODs with an example of Elevator. This example was modeled according the BIMForum's specification (Abualdenien & Borrmann, 2022; BIMForum, 2019)

Since then, many efforts have been invested in the U.S., Europe, Asia, and Australia to develop a suitable and practical standard assisting the delivery of the industry projects. Besides the BIMForum's definitions, the U.S. Department of Veterans Affairs (V.A.) has published a comprehensive spreadsheet, the *Object Element Matrix*, that provides a list of the expected LOD attributes for the building components throughout the building life-cycle (Department of Veterans Affairs (VA) Office of Construction & Facilities Management (CFM), 2010), which encourages the concept's applicability in the industry. Additionally, the U.S. General Services Administration (GSA) published an LOD guideline (U.S General Services Administration (GSA), 2018), where they used the Level of

Design, Development, and Detail as synonyms and represented as LOD. In the case of GSA, the guideline provided a mapping of the different elements to the corresponding discipline (like Architectecture, Structural Engineering, HVAC etc.) and to the design phases.

Most of the countries, especially in Europe, have proposed different terms for their regions. In the UK, the *Level of Definition* (BSI, 2017) has been introduced. It consists of seven levels and introduces two components: Levels of Model Detail, representing the graphical content of the models, and Levels of Model Information, representing the semantic information. The Danish definition includes seven *Information Levels* that correspond roughly to the traditional project life-cycle phases (van Berlo & Bomhof, 2014).

In Germany, the concept of *Modelldetaillierungsgrad (MDG)* (VBI, 2016) introduced by the German Engineering Consultancy Association VBI is a common alternative to the BIMForum's LOD. The MDG comprises ten levels (010, 100, 200, 210, 300, 310, 320, 400, 510, 600) that correspond to the HOAI project life-cycle phases (HOAI, 2013). In this regard, the MDG specification is defined according to the maturity demanded by each phase. This is a key difference to the LOD concept since the LOD has by purpose no connection to the design phases (BIMForum, 2019). However, the MDG specification provides a mapping to the BIMForum's LOD, where the MDG levels 100, 200, 300, 400, and 500 correspond to the LODs with the same level. The rest of the MDGs have no correspondence to the LOD.

The Italian LOD definition adopts the BIMForum's specification while adjusting it to seven levels with letters in an ascending order from *LOD A – LOD G* (PROGETTIAMOBIM, 2018). In Switzerland, the LOD concept is based on the BIMForum's definitions, but at the same time, it is assigned to projects life-cycle phases (Maier, 2015). Similarly, in Norway, the concept of *Model Maturity Index* (MMI) is commonly used as an equivalent to the LOD. The MMI has adopted the same scale as the BIMForum's LOD, 100 — 500, and is applicable for both the elements and building model level (H. W. and Skeie, G. and Uppstad, B. and Markussen, B. and Sunesen, S., 2018). However, MMI focuses more on developing and controlling the design process than on geometric specification. It facilitates, among others, the determination of milestones in the project.

In Australia, the American LOD concept was utilized in its current state, describing the maturity of the individual elements without any mapping to the life-cycle phases (AIA, 2014). Additionally, the NATSPEC National BIM Guide (NATSPEC, 2013b) adopted the VA's *Object Element Matrix* spreadsheet as basis. In China, the Construction Industry Council (CIC) published an LOD specification based on ISO 19650 (which uses a similar basis to the LOIN, published in ISO 17412) (Construction Industry Council, 2020; ISO, 2018, 2020). CIC defined an LOD-G and LOD-I for the geometric and semantic information, respectively. The refinement of building elements provided in the Chinese standard is relatively similar to the BIMForum's definitions.

In this literature review, we collected 58 international LOD guidelines for the period 2005 – 2020 (shown in Tables 2.1 and 2.5). Here, it is important to note that there could be other LOD guidelines that were not identified through our search. However, we believe that analyzing these many specifications is sufficient for drawing out a common ground of the widely-followed best practices. To outline the similarities and deviations of the collected specifications, they were evaluated from multiple aspects:

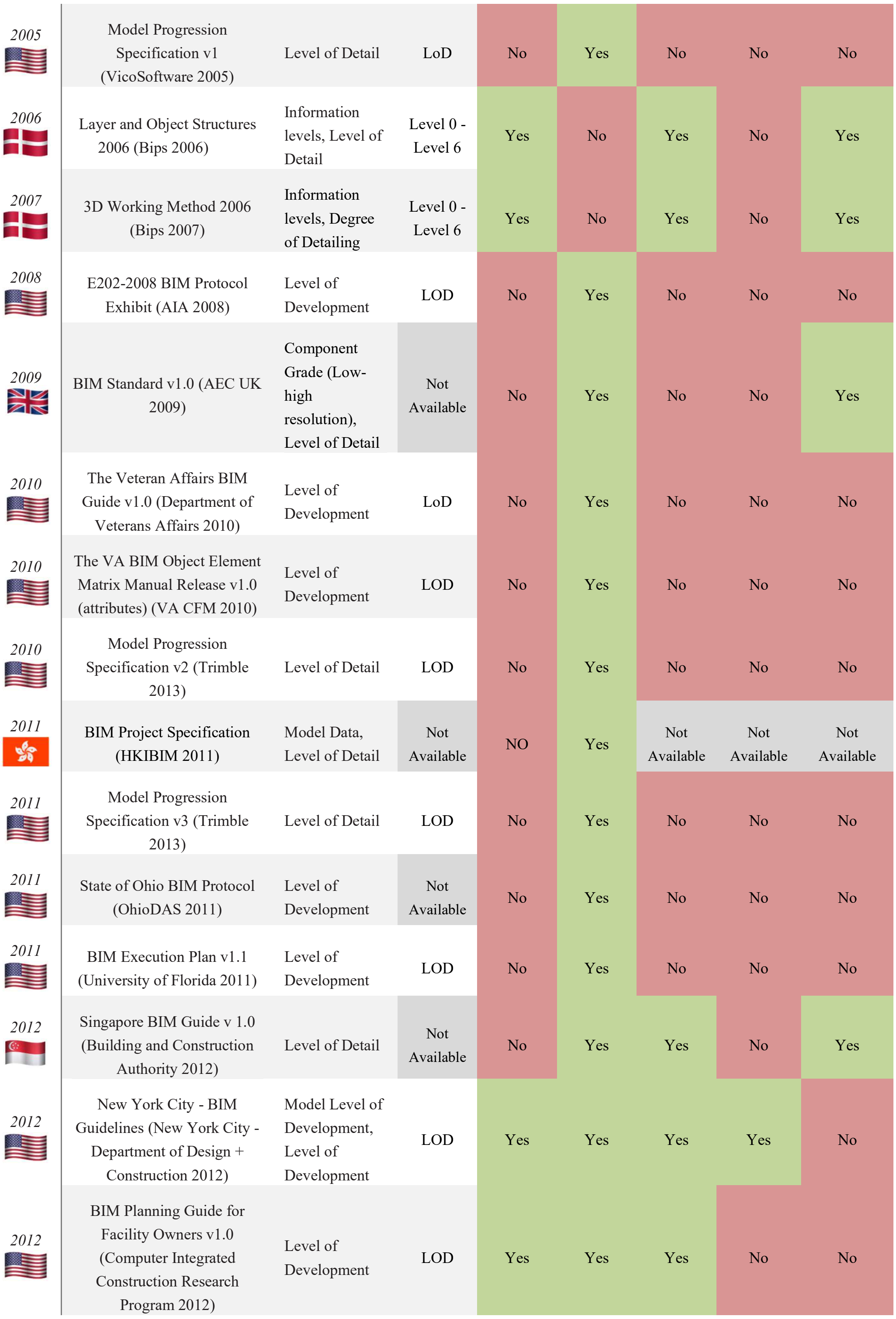
- The used concept and abbreviation
- The LOD application on building and element levels
- Overlap with design phases
- Linkage to use cases
- Linkage to 2D drawings scale

As shown in Tables 2.1 - 2.5, a color-coding is provided to highlight correlations between the specifications. *Green* indicates a positive relation, *Orange* a negative relation (i.e., inapplicability was identified from text or illustrations), and *Grey* when a relation was not found. Overall the use of the term *Level of Development* and abbreviation *LOD* is outstanding, 53.4% and 60% respectively, among the others through the years. Additionally, even though 38.9% of the specifications apply the LOD concept on the building level, we observe that the majority of them agree upon applying the LOD concept on the individual elements as well. Furthermore, 37% of them have shown an overlap between the assigned LODs and design phases, while only 6.7% developed the specification for the purpose of defining requirements of use cases. Finally, 23.7% have assigned one or multiple drawing scales to the individual LODs. The differences between the various concepts included in the LOD specifications will be discussed in detail in the following sections.

### 2.3.2 Level of Development vs. Level of Detail

The term Level of Development (LOD) is interchangeably used with the Level of Detail (LoD). However, there is an essential difference between both terms. Both terms follow the same scale of detailing from 100 – 500. The LoD describes the amount of detailing included in the model element regardless of its reliability. However, the LOD represents the amount of reliable information (i.e., fixed and thought through by the project participants). Accordingly, an element might be at a fabrication level of detailing (e.g., LoD 400) and at the same time at a low LOD (e.g., LOD 200), which means that a substantial part of this information is still uncertain and would probably change when progressing with the design. Practically, this is helpful during the design process, where designers explore the possible design options by detailing multiple variations of the same element to evaluate its suitability and performance.

Table 2.1: List of the collected and investigated LOD guidelines during the period of 2005 – 2020. Extends upon (Bolpagni, 2016) and(Gigante-Barrera et al., 2018).



| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 2005 🇺🇸 | Model Progression Specification v1 (VicoSoftware 2005) | Level of Detail | LoD | No | Yes | No | No | No |
| 2006 🇩🇰 | Layer and Object Structures 2006 (Bips 2006) | Information levels, Level of Detail | Level 0 - Level 6 | Yes | No | Yes | No | Yes |
| 2007 🇩🇰 | 3D Working Method 2006 (Bips 2007) | Information levels, Degree of Detailing | Level 0 - Level 6 | Yes | No | Yes | No | Yes |
| 2008 🇺🇸 | E202-2008 BIM Protocol Exhibit (AIA 2008) | Level of Development | LOD | No | Yes | No | No | No |
| 2009 🇬🇧 | BIM Standard v1.0 (AEC UK 2009) | Component Grade (Low-high resolution), Level of Detail | Not Available | No | Yes | No | No | Yes |
| 2010 🇺🇸 | The Veteran Affairs BIM Guide v1.0 (Department of Veterans Affairs 2010) | Level of Development | LoD | No | Yes | No | No | No |
| 2010 🇺🇸 | The VA BIM Object Element Matrix Manual Release v1.0 (attributes) (VA CFM 2010) | Level of Development | LOD | No | Yes | No | No | No |
| 2010 🇺🇸 | Model Progression Specification v2 (Trimble 2013) | Level of Detail | LOD | No | Yes | No | No | No |
| 2011 🇭🇰 | BIM Project Specification (HKIBIM 2011) | Model Data, Level of Detail | Not Available | NO | Yes | Not Available | Not Available | Not Available |
| 2011 🇺🇸 | Model Progression Specification v3 (Trimble 2013) | Level of Detail | LOD | No | Yes | No | No | No |
| 2011 🇺🇸 | State of Ohio BIM Protocol (OhioDAS 2011) | Level of Development | Not Available | No | Yes | No | No | No |
| 2011 🇺🇸 | BIM Execution Plan v1.1 (University of Florida 2011) | Level of Development | LOD | No | Yes | No | No | No |
| 2012 🇸🇬 | Singapore BIM Guide v 1.0 (Building and Construction Authority 2012) | Level of Detail | Not Available | No | Yes | Yes | No | Yes |
| 2012 🇺🇸 | New York City - BIM Guidelines (New York City - Department of Design + Construction 2012) | Model Level of Development, Level of Development | LOD | Yes | Yes | Yes | Yes | No |
| 2012 🇺🇸 | BIM Planning Guide for Facility Owners v1.0 (Computer Integrated Construction Research Program 2012) | Level of Development | LOD | Yes | Yes | Yes | No | No |

Figure 2.4 illustrates the difference between both terms on an example of an inverted T-Beam. The first illustration from the right looks detailed, where sloping surfaces and MEP

Table 2.2: Cont. List of the collected and investigated LOD guidelines during the period of 2005 – 2020. Extends upon (Bolpagni, 2016) and (Gigante-Barrera et al., 2018).

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *2012* 🇨🇦 | AEC (CAN) BIM Protocol v1.0 (CAN 2012) | level of Development, Level of Detail, Grade | LODev, LODet, G0 – G3 | Yes | Yes | No | No | Yes |
| *2012* 🇺🇸 | E, A design division BIM standard manual (The Port Authority of NY & NJ Engineering Department 2012) | Level of Development | LOD | Yes | Yes | Yes | No | No |
| *2012* 🇺🇸 | Building Information Modeling (BIM) Guidelines v1.6 (USC 2012) | Level of Detail | LOD | Yes | No | Yes | No | No |
| *2012* 🇳🇱 | Rijksgebouwendienst - BIM_Standard v1.0.1 EN 1.0 (Rijksgebouwendienst 2012) | Level of Detail | LOD | Yes | No | Yes | No | No |
| *2012* 🇬🇧 | BIM Standard v2.0 (AEC UK 2012) | Grade, Level of Detail (Scale) | G0 - G3 | Yes | Yes | No | No | Yes |
| *2012* 🇬🇧 | BS 8541-3-2012: Shape and measurement - code of practice (BSI 2012a) | Level of Detail, Level of Measurement | Not Available | No | Yes | No | Yes | No |
| *2012* 🇬🇧 | BS 8541-4-2012: Attributes for specification and assessment - code of practice (BSI 2012b) | Level of Attributing | Not Available | No | Yes | No | No | No |
| *2012* 🇫🇮 | Common BIM Requirements 2012 Series 3 Architectural Design (Gravicon 2012) | BIM Content Levels | Level 1 - Level 3 | No | Yes | Yes | No | No |
| *2013* 🇺🇸 | Document G202™–2013, Project BIM Protocol Form (AIA 2013a) | Level of Development | LOD | No | Yes | No | No | No |
| *2013* 🇺🇸 | BIMForum Level of Development (LOD) Specification (BIMForum 2020c) | Level of Development | LOD | No | Yes | No | No | No |
| *2013* 🇸🇬 | Singapore BIM Guide v 2.0 (Building and Construction Authority 2013) | Level of Detail | Not Available | No | Yes | Yes | No | Yes |
| *2013* 🇺🇸 | E203-2013 BIM and Digital Data Exhibit (AIA 2013b) | Level of Development | LOD | No | Yes | No | No | No |
| *2013* 🇺🇸 | Guide, Instructions and Commentary to the 2013 AIA Digital Practice Documents (AIA 2013c) | Level of Development | LOD | No | Yes | No | No | No |
| *2013* 🇺🇸 | National BIM Standards US v3_2.7 (NIBS 2013) | Level of Development | LOD | No | Yes | No | No | No |
| *2013* | BIM Planning Guide for Facility Owners v2.0 | Level of Development | LOD | Yes | Yes | Yes | No | No |

Table 2.3: Cont. List of the collected and investigated LOD guidelines during the period of 2005 – 2020. Extends upon (Bolpagni, 2016) and (Gigante-Barrera et al., 2018).

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 🇺🇸 | (Computer Integrated Construction Research Program 2013) | | | | | | | |
| 2013 🇺🇸 | The uses of BIM Classifying and selecting BIM uses v0.9 (Kreider and Messner 2013) | Level of Development | LOD | No | Yes | No | Yes | No |
| 2013 🇬🇧 | PAS 1192-2-2013: Specification for information management for the capital/delivery phase of construction projects using building information modelling (BSI 2013) | Level of Definition (level of model detail + level of information detail) | LOD / LOI | Yes | Yes | Yes | No | No |
| 2013 🇬🇧 | Best Practice Guide for Professional Indemnity Insurance When Using BIMs v1 (CIC UK 2013a) | Level of Detail | N/A | Yes | Yes | Yes | No | No |
| 2013 🇬🇧 | Building Information Model (BIM) Protocol v1 (CIC UK 2013b) | Level of Detail | LOD | Yes | Yes | Yes | No | No |
| 2013 🇩🇪 | BIM-Leitfaden für Deutschland - Information und Ratgeber (BIM-Leitfaden für Deutschland) (BMVBS 2013) | Fertigstellungsgrad | Not Available | Yes | Yes | Yes | No | Yes |
| 2013 🇬🇧 | Project Progression Planning with MPS 3.0 (Trimble 2013) | Level of Detail | LOD | No | Yes | No | No | No |
| 2013 🇦🇺 | BIM and LOD - Building Information Modeling and Level of Development (NATSPEC 2013) | Level of Development | LOD | No | Yes | No | No | No |
| 2014 🇫🇷 | BIM/MAQUETTE NUMÉRIQUE CONTENU ET NIVEAUX DE DÉVELOPPEMENT (Le Moniteur 2014) | Niveau de Développement | ND | Yes | No | Yes | No | Yes |
| 2014 🇦🇺 | NATSPEC National BIM Guide (NATSPEC 2014) | Level of Development | LOD | No | Yes | No | No | No |
| 2014 🇺🇸 | BIMForum Level of Development (LOD) Specification (BIMForum 2020c) | Level of Development | LOD | No | Yes | No | No | No |
| 2014 🇺🇸 | Minimum Model Element Matrix M3 v1.3 (attributes) (USACE 2014) | Level of Development (accuracy), Grade | LOD | No | Yes | No | No | No |
| 2014 🇪🇸 | Guía de Usuarios BIM (Building SMART Spanish Chapter 2014) | de los niveles de desarrollo | LOD | Yes | Yes | No | No | No |

Table 2.4: Cont. List of the collected and investigated LOD guidelines during the period of 2005 – 2020. Extends upon (Bolpagni, 2016) and (Gigante-Barrera et al., 2018).

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *2015*<br>🇺🇸 | BIMForum Level of Development (LOD) Specification (BIMForum 2020c) | Level of Development | LOD | No | Yes | No | No | No |
| *2015*<br>🇧🇪 | Building Information Modelling – Belgian Guide for the construction Industry (ADEB-VB 2015) | Level of Development | LOD | Yes | Yes | Yes | No | No |
| *2015*<br>🇨🇭 | Grundzüge einer open BIM Methodik für die Schweiz - Version 1.0 (Claus Maier 2015) | Level of Development | LoD | No | Yes | No | No | No |
| *2016*<br>🇩🇪 | BIM-Leitfaden für die Planerpraxis (VBI 2016) | Modelldetaillier ungsgrad | MDG (010 - 600) | Yes | Yes | Yes | No | No |
| *2016*<br>🇺🇸 | BIMForum Level of Development (LOD) Specification (BIMForum 2020c) | Level of Development | LOD | No | Yes | No | No | Yes |
| *2017*<br>🇨🇦 | Canadian Practice Manual for BIM (Dickinson and Iordanova 2017) | level of Development | LOD | No | Yes | No | No | No |
| *2017*<br>🇺🇸 | National BIM Guide for Owners (NIBS 2017) | Level of Development | LOD | No | Yes | No | No | No |
| *2017*<br>🇮🇹 | UNI 11337-4: Evoluzione e sviluppo informativo di modelli, elaborati e oggetti (PROGETTIAMOBIM 2018) | Level of Development (LOG/LOI) | LOD A - LOD G | No | Yes | No | No | No |
| *2017*<br>🇺🇸 | BIMForum Level of Development (LOD) Specification (BIMForum 2020c) | Level of Development | LOD | No | Yes | No | No | Yes |
| *2018*<br>🇺🇸 | GSA - Level of Detail (U.S General Services Administration 2018) | Level of Design/ Detail/ Development | LOD | Yes | Yes | No | Yes | No |
| *2018*<br>🇬🇧 | Building Information Model (BIM) Protocol v2 (Construction Industry Council 2018) | Level of Definition (Level of information + Level of Model Detail) | Not Available | Yes | Yes | Yes | No | No |
| *2018*<br>🇺🇸 | BIMForum Level of Development (LOD) Specification (BIMForum 2020c) | Level of Development | LOD | No | Yes | No | No | Yes |
| *2018*<br>🇳🇴 | MMI-Modell Modenhets Indeks (H. W., Skeie, G., Uppstad, B., Markussen, B. and Sunesen 2018) | Model Maturity Index | MMI | Yes | Yes | Yes | No | No |

Table 2.5: Cont. List of the collected and investigated LOD guidelines during the period of 2005 – 2020. Extends upon Bolpagni (2016a) and (Gigante-Barrera et al., 2018).

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *2019* 🇺🇸 | BIMForum Level of Development (LOD) Specification (BIMForum 2020c) | Level of Development | LOD | No | Yes | No | No | Yes |
| *2020* 🇨🇳 | Architecture and structural engineering - LOD specification (Construction Industry Council 2020) | Level of Graphics, Level of Information and Level of Documentation | LOD, LOD-G, LOD-I, DOC | Yes | Yes | Yes | No | No |
| *2020* 🇺🇸 | BIMForum Level of Development (LOD) Specification (BIMForum 2020c) | Level of Development | LOD | No | Yes | No | No | Yes |

penetrations are modeled. Accordingly, it is detailed up to LoD 350. However, if this detailing is not thought through and fixed, then the element is at a low LOD, in this case, LOD 100. The rest of the illustrations represent how the fixed information is increasing with the LOD.

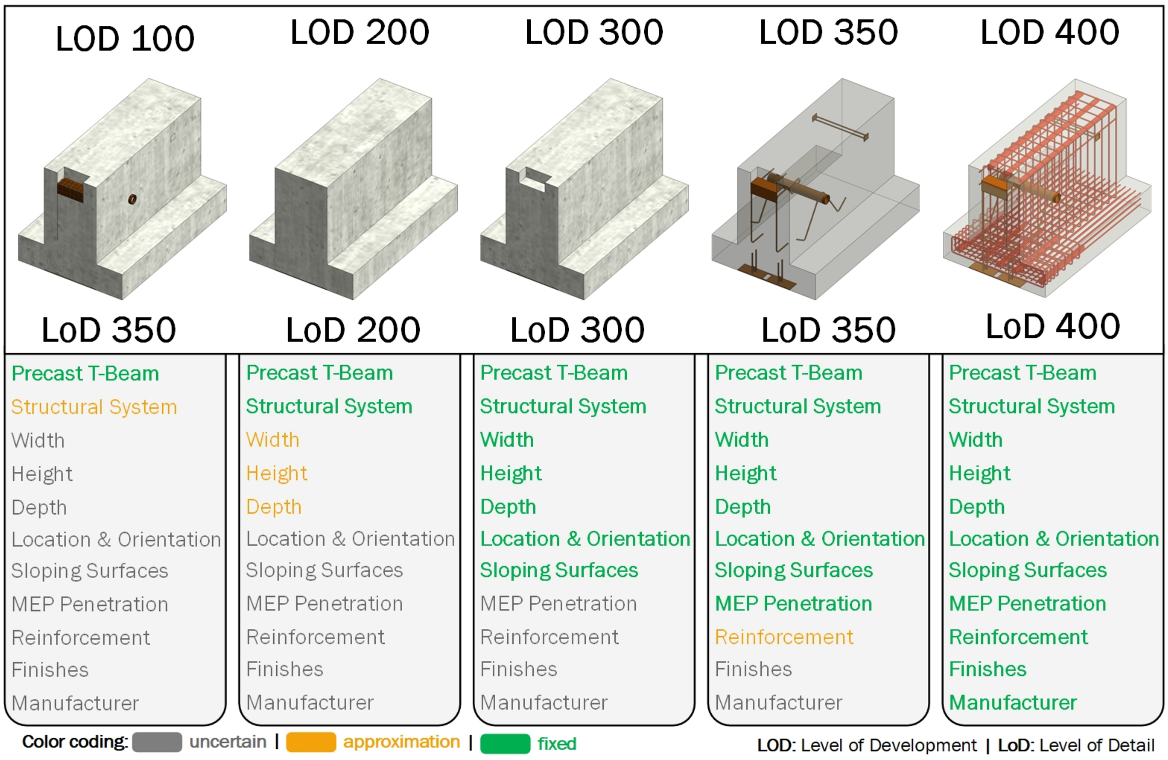


Figure 2.4: Illustrating the difference between both terms, LOD and LoD, by means of an example of an inverted T-Beam.

By contrast, the LoD concept is well established in the context of city models, where it is part of the exchange standard CityGML since 2005 (Kutzner et al., 2020). In this regard, as CityGML aims to represent the status of existing districts, including buildings and infrastructure assets, the LoD reflects the degree of detailing of the existing assets and provides the ability to reduce the geometrical complexity by providing coarser representations. Using the LoD concept for CityGML models is more appropriate than the LOD since they are typically used for archiving, visualization, navigation, and performing different kinds of analysis rather than representing the developing information maturity during the design process.

### 2.3.3 Level of Geometry (LOG) & Level of Information (LOI)

When specifying the LOD, a fundamental distinction is made between the specification of the geometric detailing (Level of Geometry, LOG) and the specification of the alphanumeric information to be provided (Level of Information, LOI). An LOD is usually understood as a combination of both specifications.

Geometric levels of development are usually described textually, but are also underpinned with visualizations of the various levels of development for individual element types. Often, an extensive catalog of illustrations is created, which provides a good reference point for model creators (see Figure 2.5). Recent studies highlighted the advantage of these visual descriptions during the modeling process (Abualdenien & Borrmann, 2022). The specification of the semantic information is usually done via a tabular representation in which it is specified for individual element types which attributes are required. For example, the BIMForum's specification describes the most essential semantic information along with the geometric descriptions and then provides an extended tabular representation for specifying the individual properties and their data types (BIMForum, 2019). Similarly, the NATSPEC's Element Matrix Manual (Department of Veterans Affairs (VA) Office of Construction & Facilities Management (CFM), 2010) provides a list of the required properties from every discipline at each LOD.

Best practice has shown that geometric specifications are generally applicable, while semantic specifications are largely defined on a customer- or project-specific basis. For this reason, it has recently been increasingly questioned whether the concept of *levels* is adequate for capturing the semantic information.

### 2.3.4 Level of Information Need (LOIN)

Recently, a European standard was introduced, called the Level of Information Need (LOIN), which specifies the information required (type of elements including their geometric details and information) at a particular design phase to perform a specific task by a specific actor (ISO, 2020). The standard was introduced with the goal to overcome the limitations of existing LOD definitions. A LOIN is defined for specific exchange scenarios – accordingly it needs to have a purpose, actors, and project milestone assigned as metadata. In its core

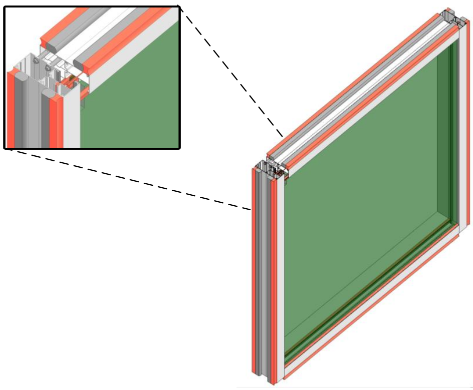
| | | |
|---|---|---|
| **LOD 350** | .... | .... |
| **LOD 400** | **LOI + LOG**<br>- Complete modeling of mullion profiles' shape and extrusion.<br>- Complete detailing of the connection points between walls and windows profile.<br>- Anchor layouts defined and modelled.<br>**Specific properties are additionally listed in a tabular format.** X | **LOG**<br> |

Figure 2.5: An example of the LOD specification of an exterior window on LOD 400 (inspired from the BIMForum's specification (BIMForum, 2019), the NATSPEC's Element Matrix Manual (Department of Veterans Affairs (VA) Office of Construction & Facilities Management (CFM), 2010), and Trimble's Project Progression Planning (Trimble, 2013)). The information available on an LOD also comprises the information from the previous LODs.

it specifies a set of semantic and geometric information requirements, and extends by the possibility to define requirements for additional documents. At this point, the authors of the LOIN standard refrain from using the term *level* as they believe that the geometric and semantic requirements are too diverse to be captured by a limited set of levels. At the same time, for the geometry specification, a set of more fine-grained sub-elements is introduced, including Detail, Dimensionality, Location, Appearance and Parametric Behavior. However, the standard remains vague when it comes to the exact usage of these elements in terms of choosable values etc.

Compared to the LOD concept, the LOIN neither describes the reliability of the provided information nor defines the maturity of the design. LOIN is focused on communicating which information is required to perform a specific task (use-case centered, such as visualization and quantity take-off). In contrast, the LOD is concerned with the refinement/development of building elements information during the design process (elements-maturity centered, independent from any use case), which can then be evaluated and used for fulfilling the needs of one or multiple use cases. For example, visualization typically requires highly detailed elements (LOD 400), whereas calculating heating and cooling demand could be performed with elements at LOD 300 (Abualdenien & Borrmann, 2019).

Figure 2.6 highlights the main differences between the structure of both concepts. The LOD focuses on the definition of the geometric and semantic information of one object type (e.g., a door). Additionally, the defined information at one level builds over the definitions specified at

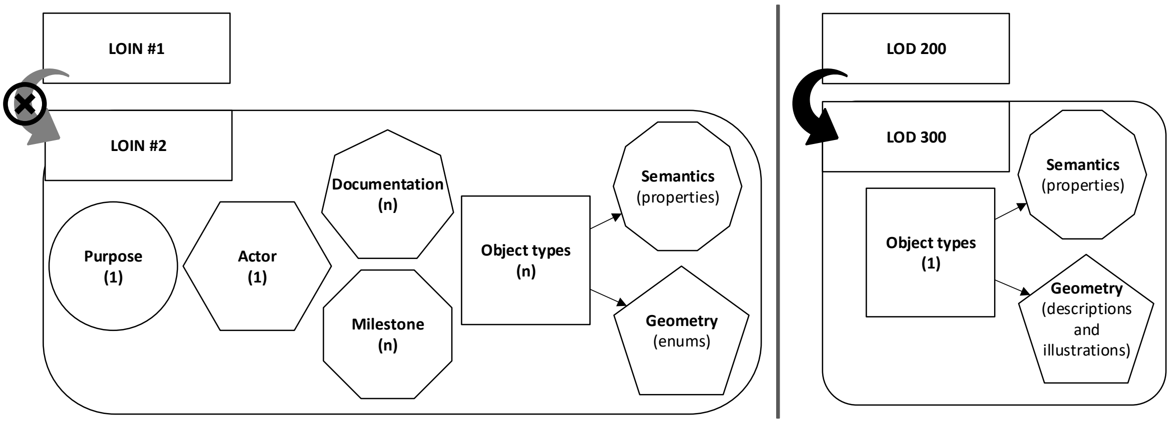


Figure 2.6: Comparison between the structure of the LOIN and LOD concepts. Besides the different set of required information, the LOIN comprises requirements of multiple object types (for achieving a specific use case), while the LOD defines the required information only for one object type. Additionally, the required information at every LOD build over the requirements from the previous levels.

the previous level (incremental). Whereas creating a LOIN instance requires more information, such as the purpose and actor. Furthermore, a LOIN comprises multiple building elements, including their geometric and semantic requirements, and does not have a connection to any previously created LOIN instances.

### 2.3.5 Deviations among the LOD standards

Most of the published standards in the U.S. are based on the AIA and BIMForum's specifications. While multiple standards do not have a known basis in Europe, the U.K.'s standards are based on the PAS 1192-2 Specification (BSI, 2017), and the available standards in the Netherlands, Italy, Switzerland, and Belgium are using the AIA and BIMForum's specification as a basis.

As discussed in the previous section, there are numerous LOD specifications published worldwide. In a recent study, Gigante-Barrera et al., 2018 identified 24 standards in the U.S and another 16 in Europe (Gigante-Barrera et al., 2018). A common ground among the developed standards is the concept of the maturity and refinement of a digital model from one level to another. Additionally, all specifications agree that each level has a description of both semantic and geometric information, where the information becomes more reliable when the level increases. A common convention among the specifications is the separation of geometric and semantic information, where the majority of the specifications make use of the terms, level of information (LOI), and level of geometry (LOG) when describing refinement at each level (Hausknecht & Liebich, 2017).

On the other hand, there are four crucial differences between the investigated specifications:

- The used term and abbreviation: the terms Level of Detail, Level of Development, Level of Definition, Level of Design are interchangeably used for the same purpose. Similarly, the abbreviation LOD vs. LoD. However, there are multiple fundamental differences between their official specifications (as discussed previously).

- Assigning the LOD to the entire model (for example, LOD 400 building model) vs. the individual building elements: this is practically misleading because when following the typical design process, foundation, exterior walls, or structural elements will be on a relatively high LOD compared to interior walls, HVAC system, or plumping. Figure 2.7 illustrates the difference between both conceptions.

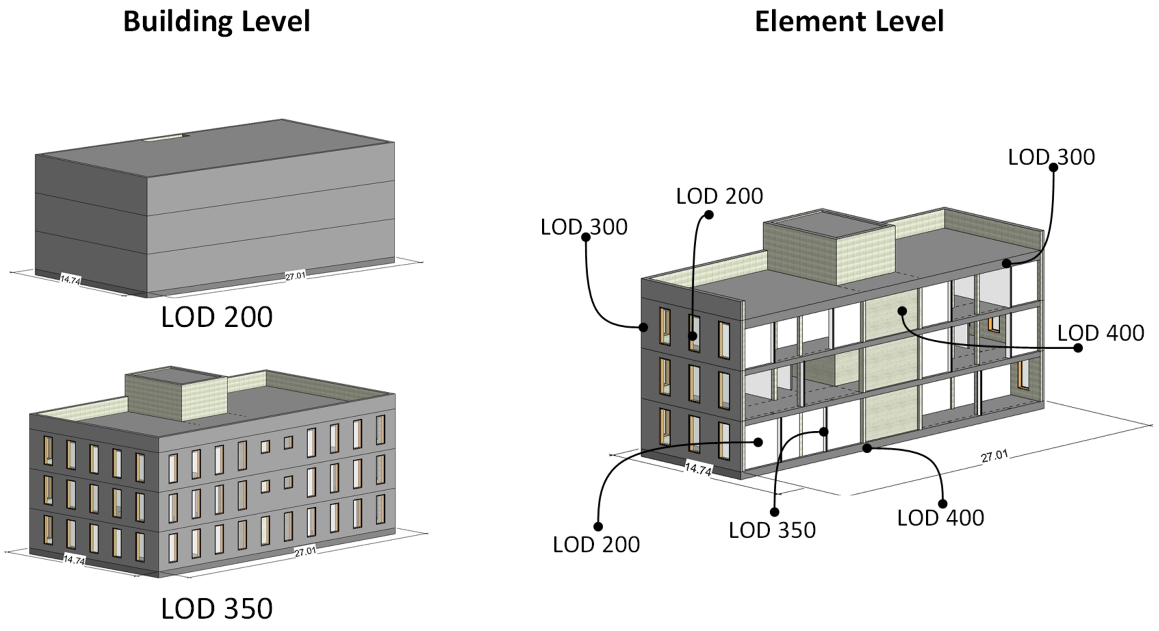


Figure 2.7: Illustration of the difference of assigning the LOD to the entire model (left side) or to the individual building elements (right side).

  In this regard, the authors of the BIMForum specification have confined their LOD definitions to describe the maturity of the elements inside the building model; in their words: “There is no such thing as an ‘LOD ### model.’ As previously noted, project models at any stage of delivery will invariably contain elements and assemblies at various levels of development.” (BIMForum, 2019)

  This is crucial for the collaboration among the domain experts involved in the project as well as for the contractual agreements. When a designer agrees to deliver a LOD 400 model, it means that all the elements contained within the delivered model must be at LOD 400. Otherwise, the designer would breach a signed contract.

- Correlating and mapping the LODs to the design phases: some practitioners conclude that all elements reach a particular LOD (e.g., LOD 300) when a project reaches a specific design phase (e.g, the design development phase). There is indeed an overlap between the development of some building elements (such as the building foundation) and the design phase since their refinement is progressing from the beginning of the project. However, it is not the case for other elements (like windows, doors..etc.). The argument here is similar to the LOD of the entire model; the LOD of elements varies within each design phase. Therefore, it is more practical to define the requirements of completing each design phase using the LOD of elements (e.g., external walls at LOD 250, interior walls at LOD 150, and structural columns at LOD 300).

- Comparing LODs to the requirements necessary to perform a particular use-case (like structural analysis or cost estimation (Kreider & Messner, 2013)) rather than defining the refinement of building elements along the design phases. Use cases could require less information than what the model currently includes, which means they can be already performed. Sometimes, use cases need more information, which means performing them should be postponed until the design is more elaborate. Accordingly, use cases rely on the LOD but they are not analogous.

Those deviations form the basis of the research questions formalized in this study (described in Section 2.2), where a systematic literature review is conducted to assess the researchers' interpretation and application of the LOD concept. More details are presented in Section 2.4.

### 2.3.6 Industrial examples on the application of LODs

The adoption of BIM worldwide is rapidly increasing (Dodge Data & Analytics, 2017). The majority of users see a positive value of using BIM, where it improves their processes and project outcomes mostly by reducing errors and providing cost predictability (Dodge Data & Analytics, 2017).

In Germany, BIM adoption has increased especially after the announcement of the Ministry of Transport for making the use of BIM mandatory for all federal infrastructure projects (BMVI, 2015). Accordingly, multiple leading clients, including Deutsche Bahn (DB), Deutsche Einheit Fernstraßenplanungs- und -bau (DEGES), and many others, have developed their own detailed LOD specification which the different architectural and engineering planners are required to fulfill (Bahn, 2020; DEGES, 2020). In this regard, DB describes the geometric detailing using the LoD term and the maturity of the semantic information using the LOI term. Similar to the concept of MDG, the DB has mapped their specification to the national design phases definitions (HOAI, 2013). On the other hand, DEGES used the LOD term that comprises both LOG and LOI scaling from 100 — 500 (DEGES, 2020). Additionally, since digital drawings are still a required deliverable in practice, building models should be capable of producing drawings with different scales. In this regard, DEGES maps the LODs to the different drawing scales. For example, LOD 100 is mapped to M 1:5000 and M

1:1000 (conceptual design and pre-planning, respectively). Finally, DEGES also recommends specifying the LODs according to the design phases definitions.

To provide a foundation for the mandatory use of BIM in infrastructure, the German Ministry of Transport funded the project BIM4INFRA2020, which established a set of guidelines and recommendations (BIM4INFRA, 2019). An essential part of these guidelines was an LOD specification describing infrastructure elements. BIM4INFRA2020 has adopted the term LOD which comprises both LOG and LOI for describing the maturity of the geometrical detailing and semantics. Following the other LOD specifications in Germany, BIM4INFRA2020 mapped the LODs to the national design phases definitions.

Other companies across Europe, for example Modelical (modelical, 2016), REBIM (REBIM, 2020), Interscale (Interscale, 2020), Ergodomus (Ergodomus, 2020), Integrated BIM(Integrated BIM, 2020), and many others, follow diverse LOD specifications, like the UK's *Level of Definition* (BSI, 2017), LOIN (ISO, 2020), or sometimes a mix between them and the BIMForum's LOD specification. The U.K.'s National Building Specification (NBS) has developed a popular guideline (Kell & Mordue, 2015), where the *Level of Definition* is described by the *Level of Detail* for the geometric representations and *Level of Information* for the semantics. In the U.S., numerous companies published guidelines describing how they perceive the LOD, such as (Autodesk, 2019; lodplanner, 2018; Services, 2014; United BIM, 2020). The majority of them are compliant with the AIA and BIMForum's specifications. Similar to multiple international companies, the BIMForum's definitions are prevalent (A2Kstore, 2021; Invicara, 2019; Tekla, 2021).

The need for the LOD concept is evident in the different projects and countries. The different companies invest effort in managing their workflows based on the LOD as a communication language among the domain experts and as a contractually binding agreement. However, as discussed, several LOD specifications are published, and practitioners adopt the specification that best fits their understanding and established workflows, ideally as simple as possible and also flexible enough to precisely capture their information needs. For example, since the national design phase definitions are essential for cost and effort estimation, some practitioners favor mapping the LODs to the design phases. Typically, the design handover to the client happens at the end of the respective design phase. Hence, the content to be delivered is defined per phase, both in conventional and also in BIM-based projects. When multiple disciplines are involved, the current best-practice is to define both discipline- and phase-specific LOD specifications instead of a generic one, especially that the development velocities vary across the different disciplines.

### 2.3.7 Application of LODs in the design process

As the LODs provide means for specifying and communicating which information is expected to be present at a specific milestone, they were used by numerous practitioners and researchers for defining the required information throughout the design phases (Abualdenien & Borrmann, 2019; Gigante-Barrera et al., 2018; Schneider-Marin & Abualdenien, 2019; Vilgertshofer & Borrmann, 2017). Abualdenien and Borrmann, 2019 developed a meta-model approach for specifying the design requirements of individual families using the LODs, incorporating the information uncertainty (Abualdenien & Borrmann, 2019). In the same context, Gigante-Barrera et al., 2018 included the LODs as an indicator for the necessary information within Information Delivery Manuals (IDMs). Abou-Ibrahim and Hamzeh, 2016 developed a framework for applying lean design principles based on LODs (Abou-Ibrahim & Hamzeh, 2016). Additionally, Grytting et al., 2017 introduced a conceptual model of a LOD decision plan, based on a set of interviews and use-cases, to support design decisions (Grytting et al., 2017). Furthermore, Karlapudi et al., 2021 introduced representing LOD-related BIM data using ontologies, which facilitates their linkage and retrieval during the projects' life cycle (Karlapudi et al., 2021).

To support the decision-making process from the early design phases, Abualdenien et al., 2020 used the LODs to integrate the design process with energy simulations and structural analysis (Abualdenien et al., 2020). Additionally, Exner et al., 2019 proposed an LOD-based framework for comparing the different design variants and their detailing (Exner et al., 2019). To exchange design requests and issues between projects participants, Zahedi et al., 2019 proposed a communication protocol that leverages the LODs to describe design requirements (Zahedi et al., 2019), and Huang et al., 2022 introduced a workflow for enhancing the interoperability of multi-LOD BIM models (Huang et al., 2022). Abualdenien and Borrmann, 2020b developed multiple visualization techniques to depict the information uncertainty associated with the LODs throughout the design phases (Abualdenien & Borrmann, 2020b). Finally Abualdenien and Borrmann, 2022 explored machine learning approaches for checking the LOG of building elements Abualdenien and Borrmann, 2022.

Overall, the majority of existing literature highlights the importance of the LOD concept for managing design requirements, assisting the collaboration among the different disciplines, and supporting design decisions, starting from the early phases. Further analysis of the different use cases is presented in Section 2.4.2.

## 2.4 Bibliography analysis

To get an overview of the prominent sources in publishing relevant literature, Scopus[1] was used for searching peer-reviewed journal papers using both keywords *Building Information Modeling* and *LOD*, yielding into 1,580 publications. The results were then analyzed in terms of number of publications and citations between the different journals using VOSViewer [2], a well-known bibliography analysis and visualization tool. As Figure 2.8 shows, there are four main clusters conducting research in this area, providing an insight into the researched topics (from architecture to civil-engineering, geoinformatics, environmental-engineering, and more). The nodes' size is proportional to the total number of publications and the edges represent the citations.

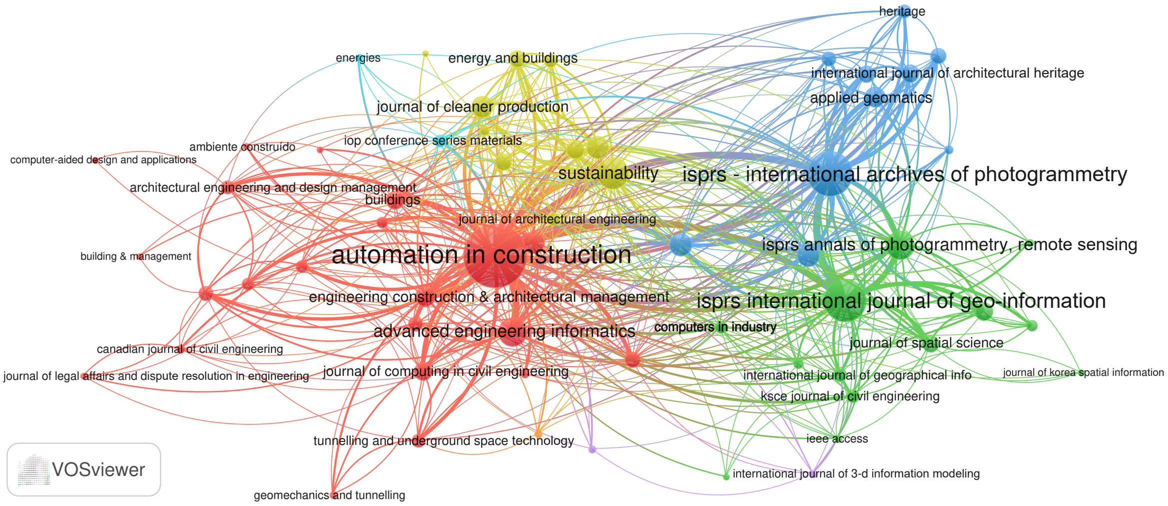


Figure 2.8: Bibliography analysis: citations network of publication sources analyzed and generated using VOSViewer. The relevant publications were collected through Scopus using both keywords *Building Information Modeling* and *LOD*, yielding into 1,580 publications.

### 2.4.1 Literature review inclusion and exclusion criteria

The results presented in Figure 2.8 in addition to the top ten journals in BIM research, identified in the literature review conducted by Z. Liu et al., 2019 for the period between 2004 – 2019 (Z. Liu et al., 2019), were used as the basis for searching relevant literature. As a result, papers published by the following journals were selected for this literature review:

- All journals available at Elsevier's *ScienceDirect®*[3]

---

[1] https://scopus.com/

[2] https://www.vosviewer.com/

[3] https://www.sciencedirect.com/

- All journals available at the American Society of Civil Engineers, *ASCE Library*[4]
- Journal of *Information Technology in Construction*[5]
- Journal of *Buildings*[6]
- Journal of *Civil Engineering and Management*[7]
- Journal of *Architectural Engineering and Design Management*[8]
- Journal of *Engineering construction and architectural management*[9]

The selected journals were searched for relevant publications during the period 2000 – 2020, where the search scope was refined using the combination of the keywords listed in Table 2.6.

Table 2.6: A list of keywords used to search the scientific databases. All of the keywords combinations were used during the search process.

| | |
|---|---|
| Building Information modeling | Level of Model Detail |
| Level of Detail | Level of Information Need |
| Level of Development | BIM |
| Level of Information | LOD |
| Level of Geometry | LOI |
| Level of Design | LOG |
| Level of Definition | LOIN |

The initial search resulted in a total of 741 potential publications. Each of the publications was then analyzed for applicability to our study by reviewing the individual papers' title, abstract, keywords, and in some cases, introduction and conclusion. The exclusion criteria followed for filtering publication includes:

- Studies investigating a different domain. Although the keywords were specific, multiple publications belong to computer graphics, biology, sociology etc.
- Studies investigating city and urban representations. Those publications were excluded since the LoD corresponds to the cityGML's Level of Detail, which addresses a different purpose than the scope of this study.

As a result, 299 publications out of 741 were selected as applicable for our study. Then each of the selected papers was analyzed with respect to answering the research questions defined in Section 2.2. Accordingly, first, the LOD standard was identified, where its definition and

---

[4] https://ascelibrary.org/
[5] https://www.itcon.org/
[6] https://www.mdpi.com/journal/buildings
[7] https://journals.vgtu.lt/index.php/JCEM
[8] https://www.tandfonline.com/toc/taem20/current
[9] https://www.emerald.com/insight/publication/issn/0969-9988

references were evaluated. Then, the parts that apply and use the LOD within each study were carefully evaluated to understand the authors' interpretation. Finally, multiple statistical calculations were performed to develop an overview of the current state of the art and identify trends over time.

## 2.4.2 Literature review analysis results

### Publications per year and journal

This section presents the analysis results of the selected 299 publications. The first results provide an overview of the publications over the period 2000 – 2020. Figure 2.9 shows the number of publications per year and per the LOD standard name (Level of Development, Detail, Definition, and not available). The category named not available represents papers that specify an abbreviation (e.g., LOD 200 or LoD 200) but do not provide any name or citation reference.

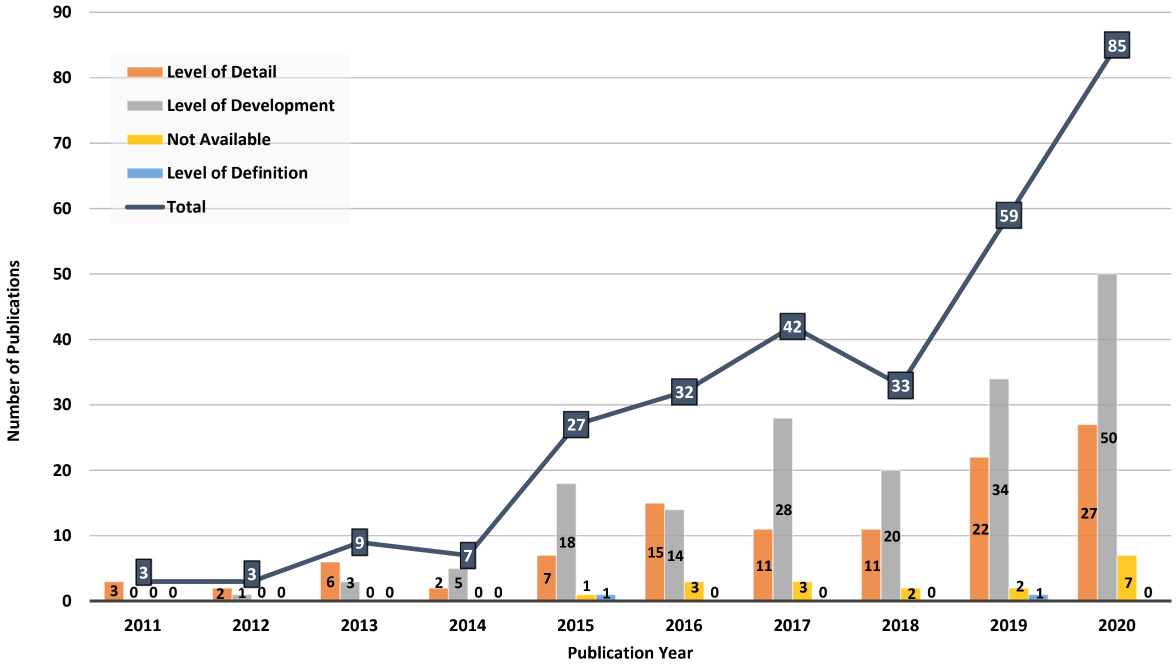


Figure 2.9: An overview of publications over the study period, 2000 – 2020. No relevant publications were found before the year 2011. The blue line shows the total publications per year and the bars at each year represent the total count of the selected LOD term.

Although the specifications of VicoSoftware (VicoSoftware, 2005) and AIA (AIA, 2013b) have been published since 2005 and 2008, respectively, only a few publications incorporating the LOD concept were published by 2011 in the investigated scholar databases. However, afterward, the LOD concept started gaining a continuous increase in popularity, reflecting the increasing demand for standardizing the progression of building information across the design phases. Figure 2.9 highlights the continuous co-existence of both naming conventions, the Level of Development and Detail. Some of the publications' background sections acknowledged the Level of Definition and Level of Information Need. The Level of Definition was used in two publications, and the Level of Information Need in none. The main reason for not using the LOIN in publications yet could be that it is still relatively recent in comparison to others.

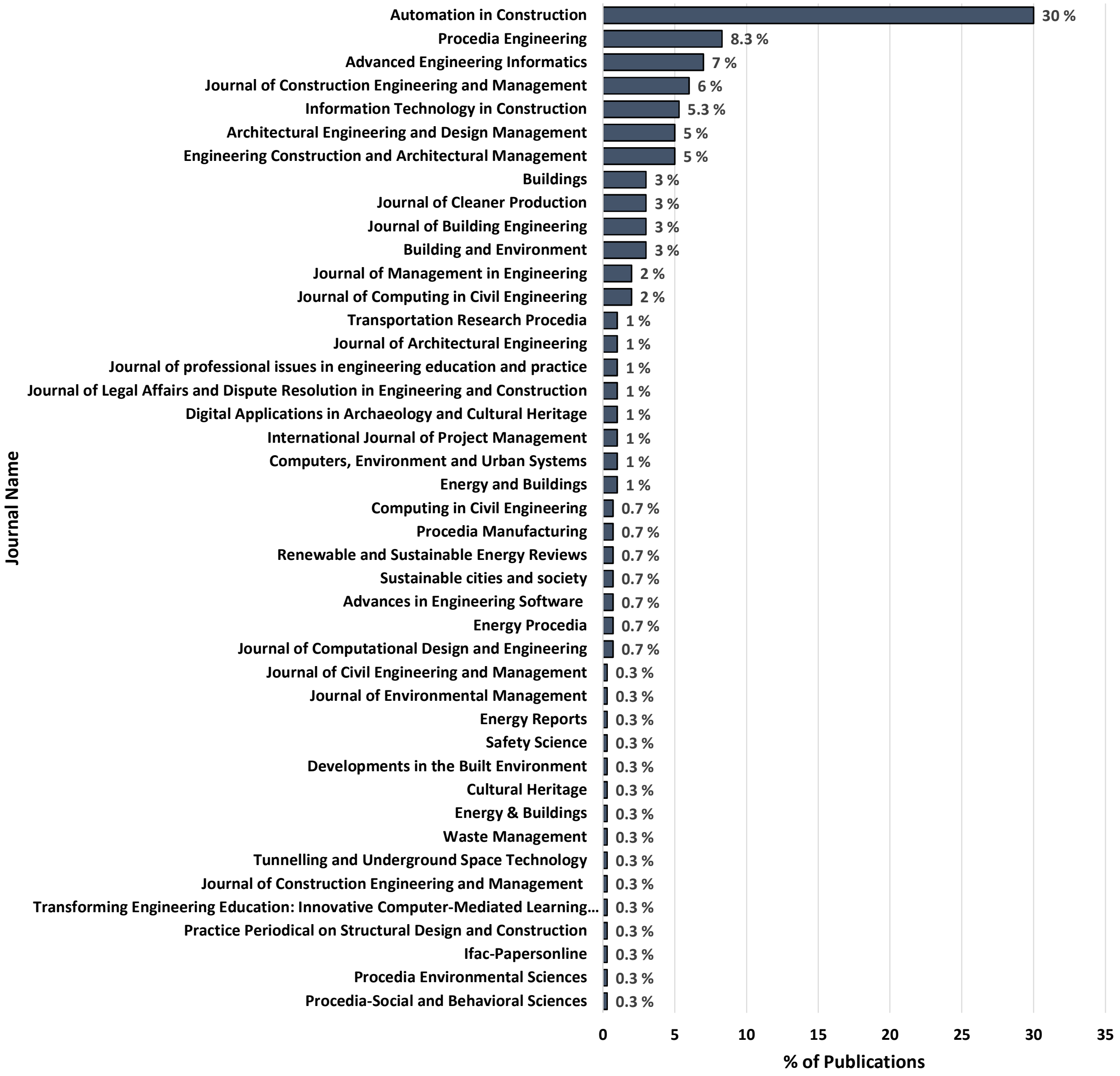


Figure 2.10: An overview of publications per journal over the study period 2000 – 2020. Journals are sorted in descending order. In total, 43 journals were identified.

To get an insight into the domains that are interested in investigating the LOD concept, the publications were grouped according to their source. Figure 2.10 depicts a sorted list of the identified journals with their corresponding percentage of publications. In total, 43 different journals were identified, which conveys the applicability of the LOD concept on the different domains and scales. In comparison to other journals, the contributions of the *Journal of*

*Automation in Construction (AutCon)* are outstanding (represents 30% of all publications), which is approximately 3.6-fold the publications of the subsequent journal.

The results presented in Figure 2.9 answer part of RQ1 as they show the current trend of increasing the adoption of the LOD concept and highlight which standards are more popular than others over time. Additionally, Figure 2.10 shows an overview of the LODs' relevant research domains.

### Analysis of the references to LODs

Afterwards, the selected publications were reviewed in detail. The first investigation was to identify which LOD concept is more popular. To identify this information, three main aspects were evaluated: (1) the concept's name, (2) its abbreviation, and (3) its scientific reference (citation).

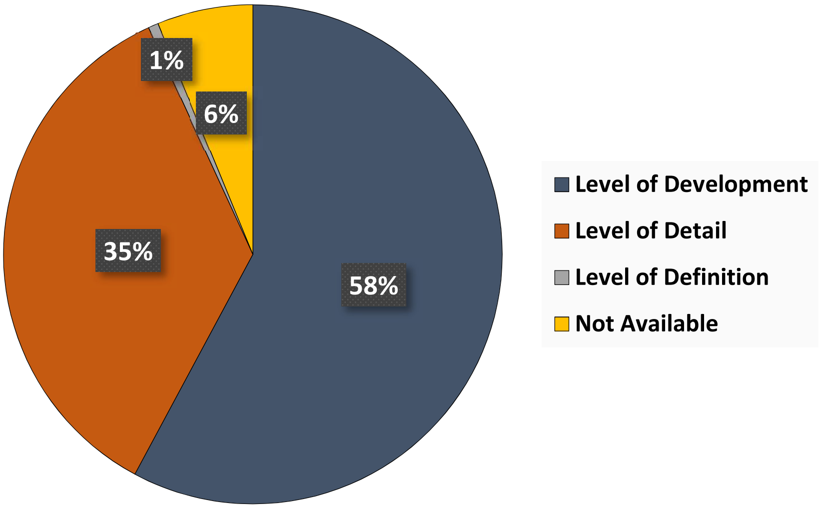


Figure 2.11: An overview of the used LOD standard names, *Level of Development*, *Level of Detail*, *Level of Definition*, and *Not Available*.

Figure 2.11 shows the LOD names and their corresponding percentages that were found during the literature review. The *Level of Development* was the most widespread naming convention, where it was used in 58% of the publications, followed by the *Level of Detail* with 35%. Additionally, 1% of the publications used the *Level of Definition* in their work. From those who used *Level of Development*, the decomposition of information to both *Level of Geometry* (LOG) and *Level of Information* (LOI) was frequently observed. Finally, 6% of publications did not mention the full concept name or a specific reference, represented as *Not Available*. Those publications considered mentioning the abbreviations *LOD* and *LoD* is clear enough for describing the geometric detailing (such as mentioning the extraction of exterior walls surfaces at LoD 200), presence of material layers (for example, the need for at least

LOD 300 for energy calculations), and the required information from the different disciplines during the design process.

It is worth mentioning that the authors' hesitation in using a specific term was clear in multiple publications; for example, when writing *Levels of Development (or detail)*, *Levels of Development/detail*, or vice versa. In the same context, numerous synonyms were also used across the different publications when describing the meaning of what an LOD is meant to represent, such as *Level of Design*, *Level of Representation*, *Level of Knowledge*, *Level of Granularity*, and *Level of Abstraction*.

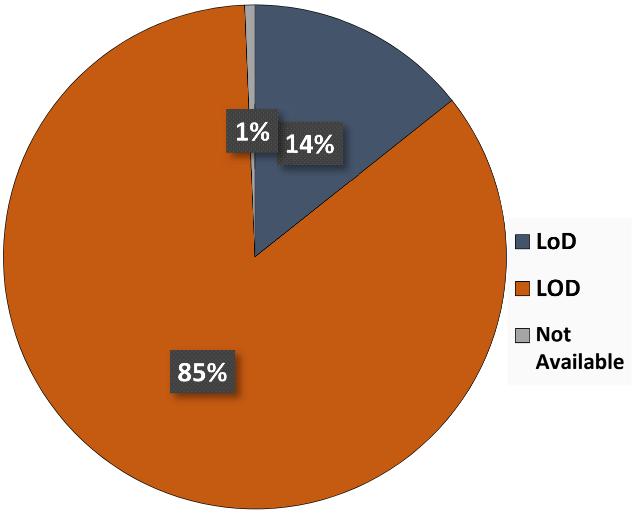


Figure 2.12: An overview of the used LOD abbreviations, *LOD*, *LoD*, and *Not Available*.

Figure 2.11 showed multiple LOD naming conventions, where 6% were *Not Available*. When checked the used abbreviations, 99% of publications included an abbreviation, and the most prevalent abbreviation was the *LOD* with 85% (see Figure 2.12). For example, the AIA and BIMForum's specifications use the abbreviation LOD while some others use LoD (Bahn, 2020; BSI, 2017; Informed, 2020). Additionally, a noticeable assignment of the *LoD* to the *Level of Detail* was observed, where 69% of publications that used *LoD* were referring to the *Level of Detail*.

Next, the referenced specifications or guidelines were checked in the individual publications. The results, presented in Figure 2.13, emphasize the lack of scientific publications referencing the guidelines, where 54% of the publications were satisfied with stating the LOD abbreviation or name and presumed that it is adequate to provide a clear and understandable meaning for readers. The explanation of this result can be related to the current state of practice, where domain experts use the LOD language to define their information requirements (e.g., requirements of LOD 300) without explicitly referring to a particular specification (van Berlo & Bomhof, 2014). However, based on surveys, every practitioner has an own idea of what requirements a specific LOD should include (van Berlo & Bomhof, 2014).

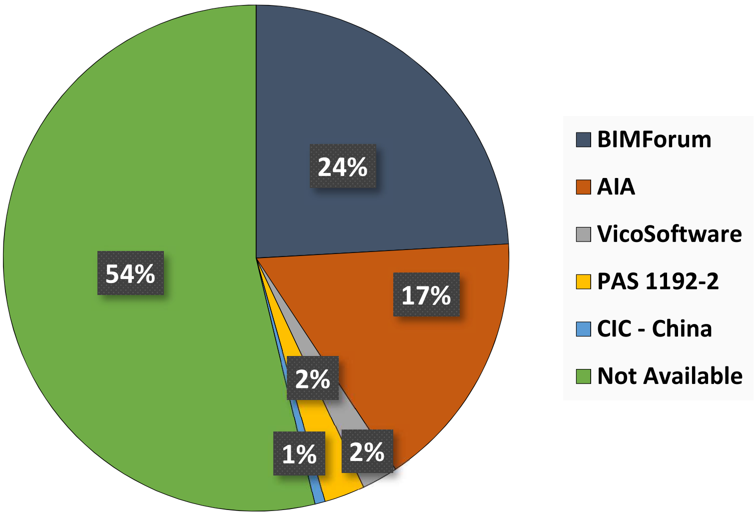


Figure 2.13: An overview of the referenced LOD specifications, *BIMForum* (BIMForum, 2019), *AIA* (AIA, 2013b), *VicoSoftware* (VicoSoftware, 2005), *PAS 1192-2* (BSI, 2017), *CIC - China* (Construction Industry Council, 2020), and *Not Available*.

From the publications that included a reference to a specific guideline, the *BIMForum* and *AIA* were the most referenced with 24% and 17%, respectively. Those were followed with few references to the UK's *PAS 1192-2* and China's LOD *CIC - China*.

Overall, from the results presented in this section, we observe the authors' preference of using the *Level of Development* term with *LOD* as an abbreviation. Additionally, the BIMForum and AIA specifications were referenced in 41% of the publications (these observations answer RQ2 for reporting about the popularity of the different standards and naming conventions).

### Analysis of the application of LODs

The previous section focused on evaluating the referenced LOD standards in the different publications. This section presents the results of carrying a detailed investigation on the use of the LOD standards in those publications. Accordingly, the areas where the application of the LOD concept was described were analyzed in detail.

The first aspect investigates whether the LOD was applied to the individual elements or the overall models, such as the a multilayered wall or an overall building model. Figure 2.14 shows the percentages of applying the LOD, where 48% of the publications referred to applying it on the overall building model, in comparison to 37% on the element level, and 15% did not provide sufficient information on how the LOD was applied. This result has multiple perspectives; as described in Section 2.3.5, the AIA and BIMForum have explicitly confined the use of their guidelines on the element level and stated that there is no correlation between the LOD and the progression during the design phases. However, 27.3% of those who

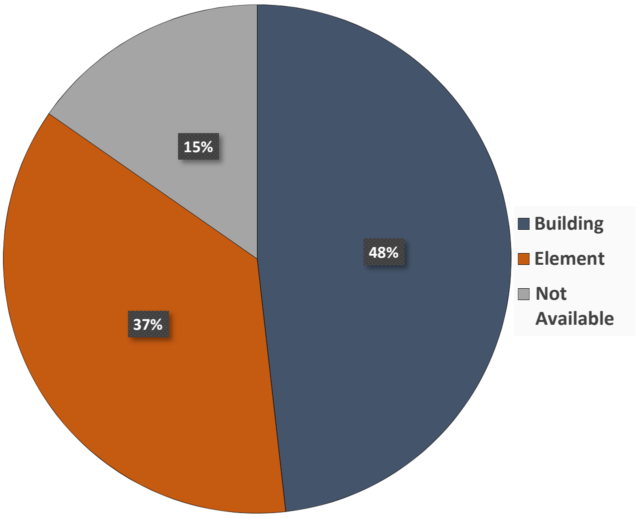


Figure 2.14: An overview of the applying the LOD concept in the investigated publications. 60% applied the LOD on the overall building and infrastructure models while 37% applied specifically on elements, while 11% did not include sufficient information to identify how the LOD was applied.

referenced the AIA or BIMForum have applied it on the overall building model. A reason for this kind of confusion could be as the influence of other well-established specifications, such as the LoD in cityGML (it describes the geometric representation of the overall building and city models), as some of them are established for more than a decade (KUTZNER et al., 2020). Additionally, there is a clear need to have an LOD standard that is capable of representing the overall building model across the design phases (ABUALDENIEN & BORRMANN, 2019). For example, ABUALDENIEN and BORRMANN, 2019 proposed a concept, *Building Development Level (BDL)*, which acts as a container describing the overall building model's requirements at a particular milestone (using the LOD language for specifying the requirements of the individual elements) (ABUALDENIEN & BORRMANN, 2019).

The next step was to categorize the individual publications according to their purpose. The aim is to identify which use cases the LOD was mainly involved in. The results are depicted in Figure 2.15, where 16 use cases were identified, including visualization, quantity take-off, model checking etc. Figure 2.15 shows the corresponding percentage of each use case. Here the use of LODs for supporting decisions, such as enhancing the collaboration and integration of the different disciplines starting from the early design phases, was used in more publications than others. The second highest use case was life cycle assessment and sustainability (represented as LCA), followed by requirements management and model checking.

The fifth highest use case was *Reality Capturing* (with 7.35%). This use case is essential in multiple phases of a projects' life cycle, including:

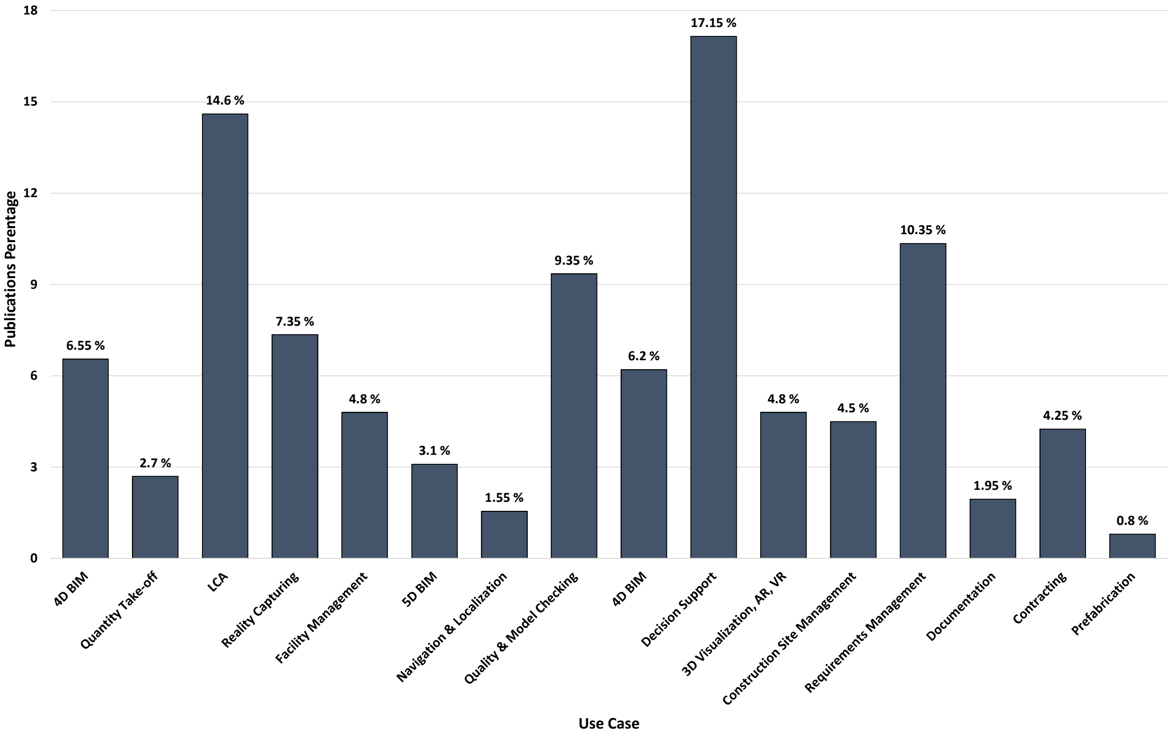


Figure 2.15: A list of the identified use cases for applying the LOD.

- design: supporting architects and owners who are considering renovating an existing asset, or capturing the current site's conditions to construct a new facility.
- construction: capturing the progress of construction to support carrying next tasks, such as capturing the anchor rod placement to confirm the interfaces of geometry and alignment of steel base plates with the anchor rods cast into the concrete (BIMForum, 2020b).
- documentation: documenting the current state of assets, including any existing health issues, such as cracks.

During the investigation of the publications it was observed that several authors (43% of the reality capturing relevant publications) used the *Level of Development* term to describe the geometric detailing of the existing assets' as-is conditions. However, based on the definitions provided by the different specifications (see Sections 5.2.3 and 2.3.2), the term *Level of Detail* is more suitable for this purpose as it describes the geometric detailing of elements rather than the state of their development (during the design process). Capturing reality is an essential use case for many applications in the AEC industry. Hence, especially with the emerging topic of Digital Twinning of the built environment, it is crucial to carefully apply standards according to their intended use. For example, when describing the accuracy of scanning and reconstructing the captured assets, it would be more suitable to combine the *Level of Detail* with specialized standards for representing the accuracy, such as the *Level of Accuracy (LOA)* (USIBD, 2019) and *Level of Acceptance (LoA)* (BIMForum, 2020b).

The different use cases highlight the applicability of the LOD concept for the entire life cycle of projects, from contracting to 4D/5D BIM, and finally, documentation and facility management (which provides sufficient information to answer RQ3).

## 2.5 Discussion

Different domain experts base their work on models provided by experts from other disciplines during the design, construction, and operation of an asset. At this point, the exchanged geometric and semantic information must fulfill specific criteria to develop the design further, evaluate its performance or actually build the asset. Hence, there is a need for a common language that the different disciplines can follow to define and communicate their requirements and specify the expected deliverables. This is the primary motivation behind creating and publishing all of these LOD specifications internationally. As illustrated in Figure 5.18, the requirements of the delivered BIM models are typically specified in contracts and BIM execution plans, where the geometric and semantic information required for the individual element types from each domain expert is specified for every design phase.

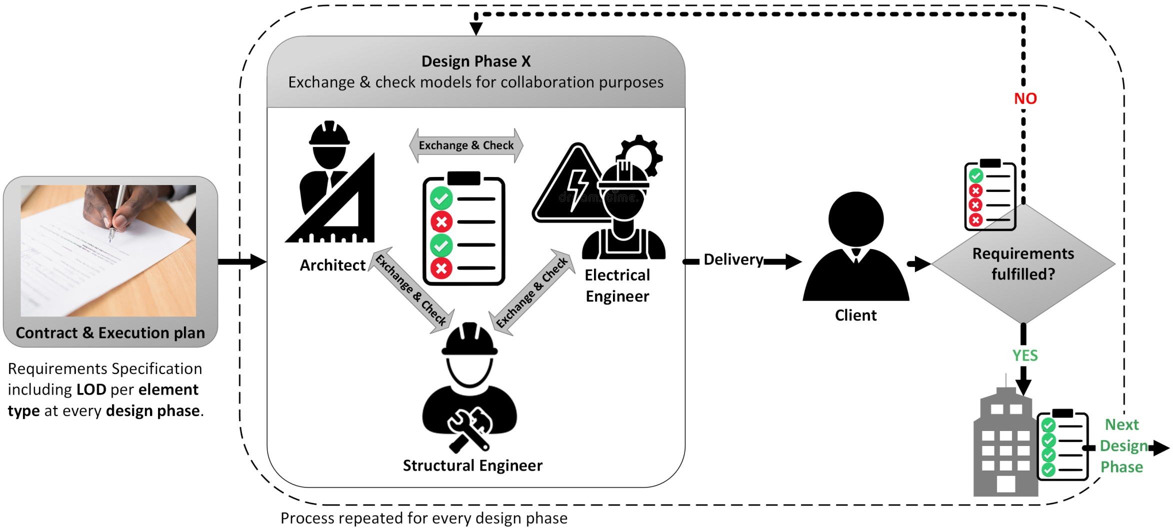


Figure 2.16: Illustration of the multidisciplinary design process, highlighting the specification of a project's LOD requirements in contracts and BIM execution plans, and then validating the specified requirements during the collaboration with different disciplines as well as delivery to the client.

By reviewing the evolvement of the LOD specifications between 2005 and 2020 we witness the different countries' attempts to reflect their needs by standardizing their best practices. In multiple cases, countries have adopted different methodologies and terminologies in their subsequently published specifications. This revolutionary period of 15 years has explored the advantages and limitations of various alternatives for standardizing design requirements (Section 2.3.5 emphasized the differences between the different approaches). Hence, future LOD specifications will be influenced by the currently dominant specifications, as their methodology and terminology are being adopted internationally, assisting in reaching a global consensus.

On the other hand, the literature review of scholarly publications has revealed a trend towards increasingly relying on the LOD concept over time as a fundamental aspect for carrying out the different tasks across diverse domains (which answers RQ1). Similar to the LOD specifications, the term *Level of Development* and abbreviation *LOD* were the most common among other alternatives. In our analysis, we identified that the term *Level of Detail* was repeatedly connected to the geometric detailing (answering RQ2). However, although the worldwide specifications differ from various aspects, we identified that almost half of the publications did not provide a reference (citation) to the specification they are following. This highlights the ready mentality for internationally standardizing the LOD definitions among practitioners, as it is seen as a common communication language.

However, so far, the AEC industry is still lacking this kind of common language. Hence, various European and international activities are trying to fill this gap with new standards, such as the *Level of Information Need* (ISO, 2020), and a simplified and computer-readable framework, like the *Information Delivery Specification* (buildingSMART INTERNATIONAL, 2021). Currently, numerous industry practitioners communicate their LOD requirements using a tabular matrix (an example is shown in Table 5.6). In this representation, each element type and its corresponding required LOG and properties are specified. The presence of properties is identified using an $X$ character, while the geometric detailing is described using the LOG levels between 100 – 500. This reduces the uncertainty of which semantic information must be present at each design phase. However, an agreement of what LOG 100 – 500 means is still necessary. Hence, explicitly referring to a particular LOD specification is crucial for clearly defining projects' scope and estimating efforts. After all, 16 common use cases were identified for the application of the LOD concept, where the top five are decision support, LCA, requirements management, quality & model checking, and Reality capturing (answering RQ3).

Other standards than the LOD, such as IDM and LOIN, are certainly more suitable for the specification of many aspects of the information exchange, including the process description, the semantic requirements and the documents. However, where LOD is unrivalled so far is the specification of design maturity with a clear focus on the geometry of building elements. It lays out a common understanding of the progression of BIM elements in a grouped set of additions in terms of modelling. The industry's rapid and wide adoption of the LOD clearly indicates that this is an essential part of specifying BIM deliverables.

Table 2.7: Example of an LOD specification, showing the required types of building elements and their corresponding LOG and LOI specifications.

| Identification | | Level of Geometry | Level of Information (LOI) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Element Type | IFC Class | LOG | Name | Core Material | Load-bearing Function | Surface Covering (texture) | Fire Protection Characteristics | Part of Escape Route? | Sound Insulation Characterstics | Is External? | Thermal Transmitance |
| Windows | IfcWindow | 300 | x | x | | | x | x | x | x | x |
| Walls | IfcWall | 350 | x | x | x | x | x | x | x | x | x |
| Curtain Walls | IfcCurtainWall | 350 | x | x | x | x | x | x | x | x | x |
| Stairs | IfcStair | 200 | x | x | | | | x | | x | |
| Ramps | IfcRamp | 200 | x | x | | | | x | | x | |
| Doors | IfcDoor | 200 | x | x | | | | x | | x | |
| Ceiling | IfcCeiling | 300 | x | x | x | | x | | | | |
| Sanitary | IfcSanitary | 100 | x | | | | | | | | |
| Rooms | IfcSpace | - | x | | | | | x | | x | |
| Slabs | IfcSlab | 300 | x | x | x | | x | | x | x | |
| Roofs | IfcRoof | 300 | x | x | x | | x | | x | | x |
| Beams | IfcBeam | 200 | x | x | x | | | | | x | |
| Columns | IfcColumn | 200 | x | x | x | | x | x | | x | |
| Structral Truss | IfcAssembly | 200 | x | x | x | | | | | | |
| Foundation | IfcFooting | 350 | x | x | x | | x | | | | x |
| Framing | IfcBuildingElementProxy | 300 | x | x | x | | x | | x | | |

Table 2.8 summarizes the applicability of the three concepts, LOD, LoD, and LOIN. All of them can be used as part of exchange requirements. However, As emphasized before, the LOIN is more comprehensive and suitable for defining exchange requirements as it comprises most of the necessary information (including the specification of milestones, actor, and all the necessary element types and their requirements). On the other hand, although the LoD and LOD define the exchange requirements for the individual element types on each level, they require additional details to be aligned with the project delivery or to fulfil a particular use case, such as which design phase and the responsible actor.

Table 2.8: Summarized comparison between the features provided by the three concepts, LOD, LoD, and LOIN.

| Feature / Concept | Maturity Indicator | Detailing Indicator | Incremental Levels | Applicable for One vs. Multiple Element Types | Use case Oriented | Part of Exchange Requirements |
|---|---|---|---|---|---|---|
| **Level of Development (LOD)** | x | x | x | One | | x |
| **Level of Detail (LoD)** | | x | x | One | | x |
| **Level of Information Need (LOIN)** | | | | Multiple | x | x |

This highlights an essential difference between LOIN and others, where LOIN focuses on specifying the requirements of multiple object types for a particular use case. At the same time, the LOD and LoD define the requirements of developing one object type further from one level to another. Furthermore, although the LOD and LoD follow a similar approach for describing the detailing of a single element through the different levels, the LOD is the only concept that provides an indicator of thought through information (i.e., mature and fixed).

## 2.6 Conclusions & future work

The LOD concept is contractually binding and essential for collaboration among the different domain experts. Practitioners typically define their design requirements and the detailing of their deliverables using the LOD language. However, numerous LOD specifications were published worldwide by public organizations and commercial companies. These specifications share a common basis of information progression and refinement from one level to another. However, they have multiple fundamental deviations, such as confining the applicability of their guidelines on the element-level rather than the building-level and describing the geometric detailing vs. the reliability of the information.

Hence, this paper investigated the interpretation and application of the LOD concept in the different domains. This paper presented a systematic literature review, where 299 peer-reviewed publications were analyzed in detail. The review results show an evident trend in increasing the adoption of the LOD through the years. A further investigation highlighted that practitioners favor the use of the term *Level of Development* in comparison to others, like *Level of Detail*, *Definition*, or *Information Need*. Additionally, this investigation identified a set of common domains and use cases in which the LOD concept was applied.

At the same time, this literature review revealed multiple misconceptions and application issues, including the use of *Level of Development* for describing the geometric detailing of as-is assets. Furthermore, more than 50% of the publications who used the LOD in their work did not provide a citation reference to which specification they are referring to, which emphasizes the authors' assumption that the meaning of *LOD 200* or *LOD 300* is a common knowledge and understandable by the community.

This study stressed the need for unifying the different LOD concepts internationally as their deviations cause multiple misinterpretations. Such issues hinder the value of the LOD as it is meant to provide a common ground and language for defining requirements and deliverables. Additionally, scholars should carefully apply the LOD concept in the research by revisiting the official LOD specifications and providing a proper citation reference. As a next step, it would be beneficial to internationally standardize the different LOD specifications (publishing international LOD guidelines). The authors are convinced that the different specifications share the same basis and can be unified.

## Acknowledgements

We gratefully acknowledge the support of the German Research Foundation (DFG) for funding the project under grant FOR 2363.

# Chapter 3

# A meta-model approach for formal specification and consistent management of multi-LOD building models

Previously published as: Abualdenien, J.; Borrmann, A.: *A meta-model approach for formal specification and consistent management of multi-LOD building models*, Advanced Engineering Informatics 40 (1474-0346), pp. 135-153, 2019, DOI: 10.1016/j.aei.2019.04.003

## abstract

The design of a building is a collaborative process among experts from multiple disciplines. Using Building Information Modeling (BIM), a model is developed through multiple refinement stages to satisfy various design and engineering requirements. Such refinements of geometric and semantic information are described as levels of development (LOD). Thus far, there is no method to explicitly define an LOD's requirements nor to precisely specify the uncertainties involved. Furthermore, despite the insufficient information in the early design stages, a BIM model appears precise and certain, which can lead to false assumptions and model evaluations, for example, in the case of energy efficiency calculations or structural analyses. Hence, this paper presents a multi-LOD meta-model to explicitly describe an LOD's requirements, incorporating the potential vagueness of both, geometric and semantic information of individual elements. The explicitly defined vagueness can be taken into account when applying simulations or analyses for assessing the performance of different building design variants. To support the continuous elaboration of a building from the conceptual to the detailed design stages, the multi-LOD model makes it possible to ensure the consistency of the geometric and semantic information as well as the topological coherence across the different LODs. The feasibility of the approach is demonstrated by its prototypical implementation as a web-server and user-interface, providing a means for managing and checking the exchange requirements both at the meta-level and for concrete building model instances. The paper is concluded with a case study of a real-world construction project that demonstrates the use of the meta-model to support model analysis and the decision-making process.

## 3.1 Introduction

The AEC industry is a collaborative environment that requires an iterative and cooperative exchange of information models (Chiu, 2002). For example, developing a structural design requires architectural design information as input. At the same time, the design of the HVAC system has to be coordinated with the structural system to take into account the required voids in slabs and structural members. In this collaboration, the information quality, such as compliance with regulations and analysis requirements, is essential for exchanging, coordinating and integrating the partial designs at various stages. A building design evolves through multiple stages, each of which is characterized by a set of consecutive and calibrated actions to satisfy the different design and engineering requirements.

BIM is a well-established methodology for cross-disciplinary building design based on the creation, management, and exchange of semantically rich 3D-models (C. M. Eastman et al., 2011). Recently, BIM has been increasingly adopted by the AEC industry (Young et al., 2009) because it improves the process' efficiency and quality by promoting the early exchange of 3D building models. Through the stages of a construction project, the building model is gradually refined from a rough conceptual design to highly detailed individual components. The LOD describes the sequential refinement of the geometric and semantic information by providing definitions and illustrations of BIM elements at the different stages of their development (BIMForum, 2019; Hooper, 2015).

The decisions made throughout the design stages, especially the early ones, steer a project's success and results (Howell, 2016). The impact of the decisions made in the early design stages (conceptual and preliminary stages) is significant, as they form the basis of the following stages (Kraft & Nagl, 2007; Steinmann, 1997). In these stages, the uncertainty on how the design may evolve is high, as many decisions have not yet been made (Knotten et al., 2015). Hence, several researchers have emphasized the advantages of integrating performance simulations early by incorporating the information uncertainty (Hopfe & Hensen, 2011; Struck et al., 2007).

However, a well-reported gap exists between the predicted and actual building performance (De Wilde, 2014; Menezes et al., 2012). One reason for this gap is the lack of information, where the practitioners quantify uncertainties in the model's inputs, such as geometric and material attributes, using information from literature, experience, or default values (De Wilde, 2014; Van Dronkelaar et al., 2016). Therefore, at every stage, the required information along with its uncertainties must be defined and communicated to the project participants to alleviate the uncertainties' impact on the simulation results and improve the quality of the decision-making (Sanguinetti et al., 2009).

The focus in the early stages is on the building's overall structural system, outer form, and interior organization (Joedicke, 1993; Steinmann, 1997; Struck et al., 2007). Presently, a wide range of model-based planning techniques is available. However, these tools require extensive input data and produce too detailed designs, even in the early stages (Cheung et al., 2012). A BIM model appears precise and certain, which can lead to false assumptions and model evaluations, as in the case of energy efficiency calculations or structural analysis, which affects the design decisions made throughout all design stages (Abrishami, 2016; Bueno & Fabricio, 2018; Cavieres et al., 2011). Hence, these tools are not adequate to support the early stages or to preserve the building model's consistency from the conceptual design to the detailed design (Gu & London, 2010; Kraft & Nagl, 2007; Penttilä, 2007). Additionally, the current LOD definitions are informal, textual definitions and graphical illustrations that do not incorporate potential uncertainties.

Within the scope of the research unit MultiSIM (FOR2363)[1], which is funded by the "Deutsche Forschungsgemeinschaft" (DFG), we aim to develop methods for evaluating building design variants in the early design stages. The variants may have different LODs as well as incomplete and uncertain information. The main approach focuses on providing:

- Consistent management of multiple LODs during the design stages
- Description of the information uncertainty
- Consistent management of design variants
- Support for model analysis at the early design stages
- Evaluation of design variants based on simulation results
- Improved communication between the domain experts

To provide a foundation for managing multiple LODs for BIM models, we propose developing a multi-LOD meta-model that explicitly describes the LOD requirements of each building component type, taking into consideration the potential uncertainties.

The multi-LOD meta-model introduces two levels, the *data-model level* and *instance level*, which offers high flexibility in defining per-project LOD requirements and facilitates the formal checking of their validity, such as defining and checking the required information to support the Life Cycle Assessment (LCA) at different design stages.

This paper discusses the advantages of representing the uncertainties during early design stages and highlights the benefits of systematically managing and checking exchange requirements between disciplines. In order to ensure the model's flexibility in handling different component types and applicability in supporting real-world data produced by different BIM authoring

---

[1] https://for2363.blogs.ruhr-uni-bochum.de

tools, its realization is based on the widely adopted data model IFC. The IFC model specification is an ISO standard that is integrated into a variety of software products (Liebich et al., 2013).

The paper is organized as follows: Section 3.2 describes the methodology used in this research and Section 3.3 discusses the background and related work. Section 3.4 provides an overview of the multi-LOD requirements and describes the design concepts, and Section 3.5 presents the meta-model design. A methodology for checking the refinement consistency across LODs is proposed in Section 3.6. In order to evaluate the multi-LOD model and the methodology proposed here, a prototypical implementation is discussed in Section 3.7 in terms of usability and potential integration in the design process. Finally, Section 3.8 presents a case study for applying the proposed approach on a real-world construction project, and Section 3.9 summarizes our progress hitherto and presents an outlook for future research.

## 3.2 Research method

This is an exploratory research study that seeks to find a solution to the current lack of methods for formally describing the design information vagueness allowed (or provided) at a given LOD. The outcome is a novel building information representation concept based on the meta-model paradigm. This concept facilitates the formal checking of the refinement consistency of the building components across multiple LODs, overcoming the error-prone manual processes prevalent in the design practice today.

The research was based on a comprehensive literature review of the information management in the early design stages and the decision-making processes. The review covered different aspects, including common practices in the design process, the available information at the early stages, and the current support provided by existing standards and tools.

Based on the knowledge gained from this literature review and the identified gaps, the contribution of this paper is as follow:

- A multi-LOD meta-model for defining the component types' LOD requirements, incorporating the potential uncertainties, in a formal manner. The multi-LOD meta-model provides the means for defining project-specific requirements and facilitates the modeling of information uncertainty
- An Extension of the BIMForum's LOD specification to support the nature of the early design stages by facilitating the estimation of information at an earlier stage
- A new concept, *Building Development Level (BDL)*, is introduced to describe the maturity of the overall building model. A BDL can be conceived as a milestone where

specific decisions need to be made. At the same time, each BDL can be used by engineers to specify the required building elements and their maturity to carry out a model analysis

- A methodology is proposed to check the refinement consistency of the geometric, semantic, and topological information across the BDLs

The aim of the proposed approach is to improve the communication and collaboration between the different disciplines as well as to ensure compliance with the design decisions made at previous stages. The approach's feasibility was evaluated by means of a real-world construction project. The information analysis and the evaluation results of the building model throughout the early stages are presented as a case study in Section 3.8.

## 3.3 Background & related work

### 3.3.1 Information uncertainty

Information uncertainty is complex, multidimensional, and has many interpretations. The terms uncertainty, fuzziness, and vagueness are used in various domains and application contexts (Raskin & Taylor, 2014); most commonly, uncertainty is an umbrella-term that describes a lack of knowledge or information, causing the occurrence of an uncertain future state (Hawer et al., 2018). On the other hand, fuzziness, as a synonym for vagueness, is related to a specific state of a specific object, and it refers to having imprecise or inaccurate information (Hawer et al., 2018; Klir, 1987). In the context of CAD modeling, Steinmann, 1997 described fuzziness as the distance from the complete and exact description.

In this paper, uncertainty represents the unknown variables affecting design variants and their fulfillment of the project's requirements and objectives. Accordingly, defining these variables can lead to fundamental changes to the proposed design, like changing the overall building's shape, increasing its height to add a new storey, or changing the internal spatial structure. vagueness is related to the reliability of the building elements' attributes and their refinement through the LODs, for example, the load-bearing components' exact position and the external walls' openings percentage.

### 3.3.2 Level of Development (LOD)

The LOD concept is employed to describe the development of a digital building model through the different stages of the building life-cycle. It formalizes the progressive nature of the design process, which enhances the quality of the decisions made (Hooper, 2015).

In most approaches, the individual levels of development are described by means of (informal) textual definitions and graphic illustrations for various building elements. Together these definitions represent the required information quality, i.e. reliability, preciseness, and completeness. A good example are the definitions provided by the American BIMForum (BIMForum, 2019), which are updated in a yearly cycle to provide a common understanding of the expected information at every LOD. In the course of a construction project, the LOD scale increases iteratively from a coarse level of development to a finer one, where additional object attributes are provided or specified more accurately.

Different information is required by the project participants at every stage to design and perform their analysis (Singaravel et al., 2018). The LOD concept facilitates defining BIM-based exchange requirements throughout the design process. The American Institute of Architects (AIA) introduced a definition of the term LOD that comprises five levels, starting from LOD 100 and reaching LOD 500. The BIMForum working group developed LOD 350 and published the *Level of Development Specification* based on the AIA definitions (BIMForum, 2019).

The first level, LOD 100 (conceptual model), is limited to a generic representation of the building, meaning, no shape information or geometric representation. The second level, LOD 200 (approximate geometry), consists of generic elements as placeholders with approximate geometric and semantic information. At LOD 300 (precise geometry), all the elements are modelled with their quantity, size, shape location and orientation. Next, to enable the detailed coordination between the different disciplines, such as clash detection and avoidance, LOD 350 (construction documentation) is introduced, where it includes the interfaces between all the building systems. Reaching LOD 400, the model incorporates additional information about detailing, fabrication, assembly, and installation. Lastly, at LOD 500 (as built), the model elements are a field verified representation in terms of size, shape, location, quantity, and orientation.

The authors of the BIMForum specification have confined their LOD definitions to describe the maturity of the elements inside the building model. This means that it is not applicable to describing the overall building maturity, which is what the BDL concept proposed here addresses; in their words:

> “There is no such thing as an ‘LOD ### model.’ As previously noted, project models at any stage of delivery will invariably contain elements and assemblies at various levels of development” (BIMForum, 2019)

Besides the BIMForum’s definitions, several guidelines have been proposed in an attempt to define the available graphical and non-graphical information at each LOD. The US Department of Veterans Affairs (VA) has published a comprehensive spreadsheet, the *Object Element Matrix*, that provides a list of the expected LOD attributes for the building components throughout the building life-cycle (Department of Veterans Affairs (VA) Office of Construction & Facilities Management (CFM), 2010), which encourages the concept applicability in the industry. This spreadsheet was adopted by the Australian's NATSPEC National BIM Guide (NATSPEC, 2013b).

In the UK, the *Level of Definition* (BSI, 2017) has been introduced. It consists of seven levels and introduces two components: Levels of model detail, representing the graphical content of the models, and Levels of model information, representing the semantic information. The Danish definition includes seven *Information Levels* that correspond roughly to the traditional construction stages (van Berlo & Bomhof, 2014).

In practice, knowing when a building model is at a specific LOD is crucial since it is depicted as a milestone for performing new tasks using newly defined information. However, the current LOD definitions are informal and imprecise, bring only textual and graphical, which leads to multiple interpretations and opinions regarding the expected information at each level. Furthermore, even at early design stages, BIM authoring tools produce too detailed designs. Hence, precisely defining the LOD requirements incorporating their uncertainty improves the quality of the collaborative process among the disciplines.

Recent approaches propagate the terms Level of Information (LOI) and Level of Geometry (LOG) to clearly distinguish semantic from geometric detailing grades (Hausknecht & Liebich, 2017). In this paper, the abbreviation LOD stands for the Level of Development comprising both the Level of Geometry (graphic-oriented) and Level of Information (semantics, non-graphic-oriented).

### 3.3.3 Refinement of LODs

Multiple efforts have been conducted for describing the LODs' refinement throughout the project's life-cycle. The main idea is the attempt to represent and formalize the model maturity, either by explicitly defining relationships or by controlling the amount of added details within an LOD, which makes it possible to check the model's consistency.

Biljecki et al., 2016 argue that five LODs are not enough to capture the building model's development, as the information ambiguity is high. Thus, they restrict the LODs refinement by allowing less specification and modeling freedom using a set of 16 stages. Similarly, van Berlo and Bomhof, 2014 looked into producing a more suitably refined set of LODs for the Dutch's AEC industry, they developed seven LODs after performing multiple geometric tests and analyzing the industrial practices.

From another perspective, Borrmann et al., 2014; Borrmann et al., 2015 presented a methodology for creating and storing multi-scale geometric models for shield tunnels by explicitly defining the dependencies between the individual levels of detail. For this purpose, a multi-scale product model is developed, including a geometric-semantic description of five levels; the levels 1-3 describe the outer shell in terms of the boundary representation of the tunnel volume, boundary surface as well as openings, and the fourth level includes the modeling of the tunnel's interior structure. It is shown how the LOD concept can be integrated into the IFC data model. In order to model the relationship between the different levels and maintain their aggregation, a new relationship class *IsRefinedBy*, a subclass of Aggregates, is introduced. The proposed multi-scale model makes use of the parametric modeling techniques to preserve the consistency among the different levels of detail by interpreting and processing the procedural geometry representations. Consequently, the change of a geometric object is propagated by updating all the dependent representations.

### 3.3.4 Interoperability

The design and construction of a building is a collaborative process that incorporates multiple disciplines. Each expert, such as the architect and structural engineer, uses different authoring tools and requires specific information to be present in the model to support a particular type of simulations and analysis. With the increasing specialization of the stakeholders, the building industry requires a high level of interoperability, which is deficient. The US national institute of standards and technology (GCR, 2004) as well as many researchers and case-studies (Cemesova et al., 2015; Hernández et al., 2018; Lai & Deng, 2018) have confirmed the difficulties and high annual costs resulting from the lack of interoperability between the AEC industry software systems.

The Industry Foundation Classes (IFC) schema (Liebich, 2013) is an open data exchange format promoted by buildingSMART for interoperability within the AEC industry. It aims to define a common interface for lossless geometric as well as semantic data exchange. IFC is a free vendor-neutral standard and includes a large set of building information representations, including a variety of different geometry representations and a large set of semantic objects modeled in a strictly object-oriented manner. To allow for dynamic (schema-invariant) extensions and adaptation to local or national requirements, the IFC data model provides the PropertySet (PSet) mechanism, which relies on dynamically definable name-value pairs.

Besides exchanging data using IFC, dealing with different kinds of building information, e.g. property sets and definitions, requires a standardized terminology. Thus, the buildingSmart Data Dictionary (bsDD) (buildingSMART, 2016) was developed as a central repository that stores multilingual definitions of the IFC entities and common schema extensions, for instance, an *IfcWall* entity description and *Pset_WallCommon*. Additionally, bsDD integrates multiple classification systems, including *OmniClass* (OmniClass, 2012) and *UniClass* (Chapman, 2013), which are widely adopted for structuring the building information. Each object in the dictionary is identified by a Globally Unique ID (GUID) which makes it computer-readable and independent from the object name and language (Bjorkhaug & Bell, 2007).

As the IFC data model is too large for software vendors to be fully implemented (Bazjanac, 2008), buildingSMART developed the Model View Definition (MVD) mechanism as a standard approach for IFC implementation. An MVD represents a subset of the IFC schema that specifies the requirements and specifications of the exchanged data between the involved software tools (Hietanen & Final, 2006). In order to ensure the exchanged data completeness, the required information for each discipline scenario needs to be documented and defined as computer-executable rules (Yang & Eastman, 2007). Hence, MVD and the associated open standard mvdXML (Chipman et al., 2012) can be used to structure the exchange requirements with specific IFC types, entities, and attributes (See et al., 2012).

In order to facilitate the collaboration between multiple disciplines, multiple vendor-specific (Autodesk, 2018; GRAPHISOFT, 2018) and IFC-based (Beetz et al., 2010) BIM server technologies as centralized platforms have been introduced. As for IFC-based servers, the open-source BIMserver, developed by TNO and the University of Eindhoven (Beetz et al., 2010), is becoming a popular solution among researchers, as it is open-source, free of cost and provides a high degree of flexibility (Shafiq et al., 2013). It simplifies the storage, sharing, and management of IFC models through a set of extendable features, including versioning, visualization, and filtering. BIMserver parses IFC data and stores it in a relational database for later manipulation of model information, such as merging and querying. Furthermore, it is capable of generating up-to-date IFC files.

So far, the IFC data model supports neither the notion of LOD nor a description of its uncertainty. However, as it is a very widespread and well-established format, we will show how an external meta-model can be used to enrich IFC data by these aspects.

## 3.4 Multi-LOD meta-model

The early design stages involve the selection among variant designs and the determination of costs, forming the basis of the following stages (Kraft & Nagl, 2007; Steinmann, 1997). In these stages, the efforts and costs required to make changes in a building model are lower than in the subsequent stages (Kolltveit & Grønhaug, 2004). However, the lack of adequate information impedes informed decision-making. Hence, it is crucial to maintaining the individual component's LOD requirements. Especially in the process of designing a building, the components are associated with diverse levels of development within the same stage. For example, load-bearing components can be described with a higher LOD than the interior fittings in the early design stages.

Currently, there is no approach for formally defining and maintaining multiple levels of development of a building information model as well as incorporating its information uncertainty. The developed building models throughout the design stages are decoupled and appear detailed as well as certain, even in the early stages. This can lead to false assumptions and model evaluations that affect the design decisions made throughout all design stages. To fill this gap, the authors developed a multi-LOD meta-model that allows for and supports the following activities:

- Define the building model's requirements at multiple design stages
- Define component types' LOD requirements
- Model the information vagueness
- Represent a building model of multiple stages
- Describe the relationships between LODs
- Check the consistency across the design stages

To manage the requirements of the individual building component types for a specific LOD, a component type is associated with multiple LOD definitions. An LOD definition consists of two separate groups: one for defining the geometric representation and alphanumerical attributes, and another for specifying the semantic alphanumerical attributes. This separation helps to achieve and maintain the semantic-geometric coherence of the overall model (Clementini, 2010; Stadler & Kolbe, 2007). Finally, the building model is presented by multiple instances of the defined component types.

### 3.4.1 Design process in the early design stages

At the beginning of a building project, designers capture the main intent by producing spatial models as variants, providing an overview of different solutions (a.k.a early design exploration (Rivard & Fenves, 2000)). The early design stages are characterized by a large number of abstract design concepts. Each of the developed concepts consists of three main aspects: the structural system (construction-oriented), the outer form and the building's facade (shape-oriented), and the organization inside the building (functionality-oriented), including the required rooms, their dimensions, and relationships (Joedicke, 1993; Steinmann, 1997). Accordingly, these aspects within the developed variants are evaluated in terms of fulfilling the owners' requirements, building performance, and cost. Once a variant is selected, its geometry and semantics are gradually detailed. To check the consistency of the assumptions and decisions made in the conceptual design, the building information, as well as the potential vagueness, must be captured.

The meta-model approach itself provides maximum flexibility and supports any kind of country- or project-specific LOD definition. In this paper, we use the BIMForum's definitions (LOD 100 – 500) as a basis, while diverging by introducing intermediate LODs, LOD 150 and 250, to better support the early stages of design. This way, the model's refinement is captured in minimal steps, which assists in developing consistent models.

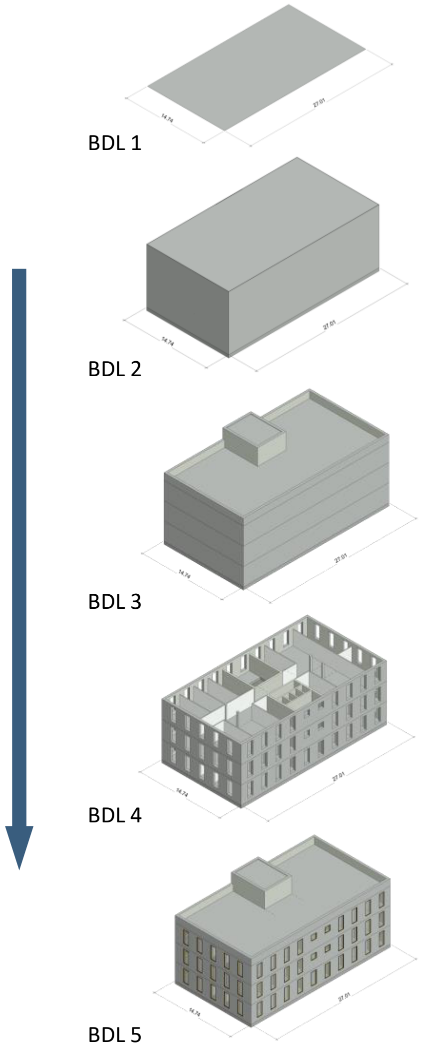


Figure 3.1: Refinement of building model at early design stages using the proposed Building Development Level (BDL) scale

Additionally, as the focus in the early stages is on the organization of the building as a whole, considering various functional and interrelated entities, it is essential to follow clear guidelines in describing the expected elements to be present in the building model as well as their maturity, i.e. LOD, at a particular stage. As the BIMForum's specification is not applicable for this purpose, we introduce a new concept, *Building Development Level (BDL)*, to describe the overall building refinement in five levels (BDL 1 – 5), as illustrated in Figure 3.1 and described below:

- *BDL 1*: The building is represented as a 2D site plan bounded by outlines of the external walls, without any geometric representation. In this level, information about the building usage, in addition to an estimated orientation and position is available. Additionally, the boundary conditions, such as side-way limitations, are considered.
- *BDL 2*: The building's height can be estimated, therefore, we can model the building's 3D volume. Here, information about the building foundation and external components' midsurfaces becomes available. Accordingly, the building's overall space is estimated.
- *BDL 3*: Information about the structural system, construction type, and the material is available. The building mass is divided into individual storeys, providing information about the number of storeys as well as the height and usage of each storey. As a result, the space of each storey is identified. Here, load-bearing components can be defined, usually represented by axis and grids.
- *BDL 4*: A more precise definition of the interior structure is modeled, which leads to a definition of the internal spaces. In this level, the percentage of the openings and an estimated load can be specified.
- *BDL 5*: A more precise material, construction type, load, and layer structure of building components is provided. The components can be represented by solids that provide a detailed geometry description.

The BDL concept describes the information quantity and quality with regard to the design process of an entire building model. A building model at a certain BDL comprises components with diverse LODs; for example, BDL 4 requires external walls in LOD 250, interior walls in LOD 150, and structural columns in LOD 300. This approach directly reflects the current BIM-based design practice (Sacks et al., 2018).

In the context of the presented research, the primary goal of specifying the development of building design is to explicitly describe the maturity (or inversely, the uncertainty) of the information (both geometric and semantic) provided. This allows for the use of analysis and simulation tools to already assess a building's performance in the early design stages while preventing the false impression of high accuracy through the consideration of the vagueness.

To illustrate the design process during the early stages, Figure 3.2 depicts the process of finding good building design solutions. The architect introduces different concepts based on the information available at every building development level by producing multiple variants.

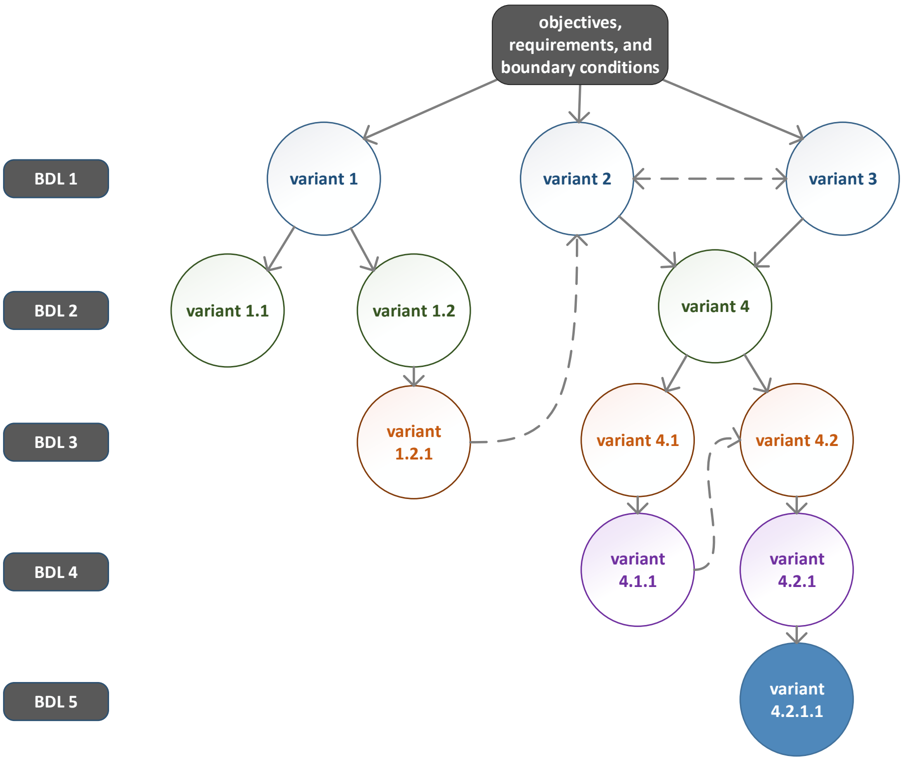


Figure 3.2: Development and selection of design variants during the early design stages (the BDL levels represent the detailing of the selected building model). This process is derived from the experience our research group has gained from the case study presented in Section 3.8

Subsequently, the project participants evaluate the proposed variants in terms of fulfilling the project's requirements. As a result, a design is selected or a new variant is proposed as a foundation for the next stage. The developed variants are evaluated iteratively until a consensus about the best solution is reached. In case not all requirements are satisfied after detailing a design, the process is repeated for a different variant. In Figure 3.2, *variant 1* was developed until BDL 3, but as it did not satisfy all the requirements, the project participants evaluated the other variants and proposed *variant 4* for the next stage. The process continued until they agreed that *variant 4.2.1.1* is a suitable solution for this project.

### 3.4.2 Geometric - semantic properties and vagueness

The multi-LOD meta-model aims to maintain a clear separation between the building components' semantic and geometric requirements. In terms of the geometric representation of a building component, it is refined along with increasing the level of development. For example, as demonstrated in Figure 3.3, in LOD 100 an external wall's position can be estimated, therefore, it is presented as a *centerline*. Since in the next LODs additional information is available, such as a thickness and material, it is possible to render the wall's *solid* model in its 3D shape and dimensions. This kind of hierarchical development of a *centerline* towards a *solid* model defines the dependencies between the geometric representations at the different levels of development. Accordingly, the relationships between the semantic requirements are determined, which supports the checking of the consistency between the LODs.

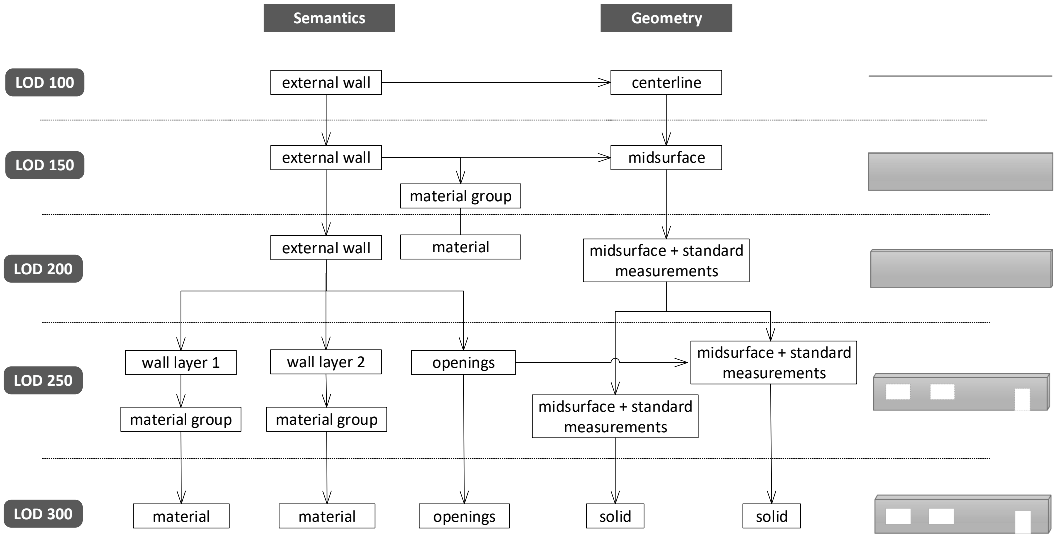


Figure 3.3: Separation of geometry and semantics at different LODs of an external wall, emphasizing the component types' refinement across the LODs

By incrementing the LOD, additional information becomes available; for example, the construction type and material can be determined from LOD 200. In some cases, it is uncertain whether a specific property is available or can be estimated at a specific LOD. Thus, the multi-LOD meta-model provides the ability to specify whether a property is mandatory or optional and offers a level of accuracy in specifying the property's assigned value in case of uncertainty. The level of accuracy in assigning the attribute's value is related to its type; it might be achieved by specifying an abstract value, such as a classification category, or a vagueness range. With that said, it is possible to model and analyze the known uncertainties of the building model at the early design stages where uncertainty is at its highest.

Figure 3.4 provides an example of the available attributes for an *External Wall* from LOD 100 to 300. The available BIMForum's definitions for each LOD are listed, which explains our interpretation with respect to the early stages. The BIMForum LOD specification provides a fundamental definition of each LOD that applies to all component types, and then it lists more specific definitions for each component type.

As Figure 3.4 exhibits, at LOD 100, the BIMForum's fundamental definition states that the components have no geometric representation and their existence can be represented as symbols with no shape or precise location. Whereas, the exterior walls' detailed definition assumes that a wall and its dimensions can be represented by a solid mass with flexible thickness and location.

| Attributes | LOD 100 | | LOD 150 | | LOD 200 | | LOD 250 | | LOD 300 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | existing | vagueness | existing | vagueness | existing | vagueness | existing | vagueness | existing | vagueness |
| Position | ✓ | ±20 % | ✓ | ±10 % | ✓ | ±5 % | ✓ | - | ✓ | - |
| Dimensions | | | ✓ | ±20 % | ✓ | ±10 % | ✓ | ±5 % | ✓ | - |
| Opening position | | | | | | | ✓ | ±10 % | ✓ | ±5 % |
| Opening percentage | | | | | | | ✓ | ±10 % | ✓ | ±5 % |
| Material | | | ✓ | material group (wood, concrete,..) | ✓ | material | ✓ | - | ✓ | - |
| Layers / material | | | | | | | ✓ | material group (wood, concrete,..) | ✓ | material |
| BIMForum Definitions | **Fundamental:** No geometric representation, symbols showing the existence of a component but not its shape, size or precise location.<br>**Exterior Walls:** Solid mass representing overall volume, or schematic elements. Depth/thickness and locations still flexible. | | N/A | | **Fundamental:** Generic placeholders, volumes for space reservation.<br>**Exterior Walls:** Generic wall objects separated by type of material. Approximate wall thickness represented by a single assembly. Locations still flexible. | | N/A | | **Fundamental:** Accurate size, shape, location, and orientation. Penetrations are modeled to nominal dimensions.<br>**Exterior Walls:** Single model element with specific overall thickness. | |

Figure 3.4: Example of assigning geometric-semantic attributes and vagueness to an external wall for LODs 100 – 300. The information is estimated earlier using intermediate LODs with a vagueness percentage or classification (the vagueness percentages are estimated based on an interpretation of the BIMForum's definitions and domain knowledge)

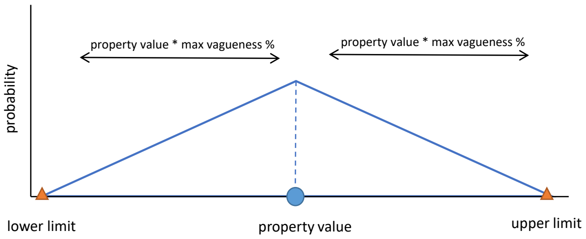
Considering the early design stages, when modeling an external wall in LOD 100, the building model is at BDL 1, i.e. the main focus is on defining the building's boundaries, orientation, and side-way limitations. Hence, it is beneficial to estimate the wall's position, as it is important to provide a solution at this level. However, modeling additional information, such as the wall's overall volume and dimensions, would wrongly suggest that the design information is more elaborate than it actually is. At this level, we have no information about the wall's main material or layers, thus, including them would produce very detailed and inaccurate compositions as design variants.

The other BIMForum's definitions, LOD 200 and 300, fit the design process at the early stages. To increase the LOD concept's support for the early stages, we propose intermediate LODs to estimate the information one step earlier with some vagueness.

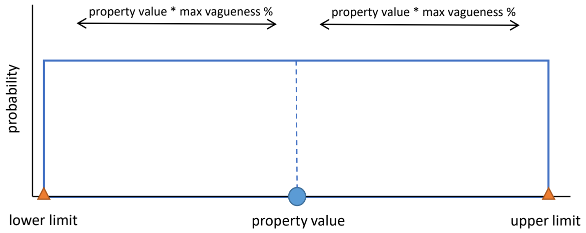
In the example presented by Figure 3.4, the position can be estimated from LOD 100 with ±20% and it becomes more certain by incrementing the LOD. From LOD 150, the dimensions can be estimated with ±20% and become certain at LOD 300. Per the BIMForum's definitions, the doors and windows' openings (penetrations) are modeled starting from LOD 300 with nominal dimensions. Therefore, the openings position and percentage are estimated at LOD 250. Considering a different type of vagueness, the information about material can be available at LOD 150, where in this level; it is defined by specifying the material group, such as *Ceramic*, whereas afterwards the exact material value, like *Brick*, should be assigned. Cross-validating the assigned values through the LODs ensures information consistency, as the model becomes more certain and mature.

### 3.4.3 Representing vagueness through distribution functions

Modeling the vagueness through a range of values means that all of them have a constant probability. This kind of probability, a.k.a *Uniform Distribution*, makes it easy to estimate the uncertainty, especially when the information is incomplete. In case the designer has a central tendency for some values than the others, the *Triangular, Quadratic,* and *Cosine Distributions'* characteristics fit into representing the values' probability (I. ISO & OIML, 1995). To apply these types of probability functions, it is enough to know the upper and lower limits and the expected value, which the designer assigns to the attribute from their knowledge, as shown in Figure 3.5a and 3.5b.



(a) *Triangular* distribution



(b) *Uniform* distribution

Figure 3.5: Modeling vagueness range with distribution functions

Additionally, as the *Normal Distribution* is the most frequently seen in representing the physical universe (Castrup, 2001), it is possible to apply it to the vagueness range. However, the *Normal Distribution* represents the uncertainty of observations, which means besides relying on the *mean*, i.e. the expected value, the Standard Deviation (STDV) needs to be provided. This, however, is rather counter-intuitive and thus uncommon in architectural design practice.

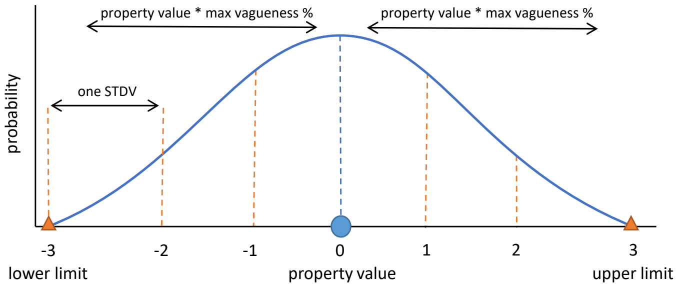


Figure 3.6: Modeling vagueness range with *Normal* distribution

A popular method that applies to normally distributed data is the *Empirical Rule* (Grafarend, 2006). This rule states that 99.7% of the possible values lie within three STDVs of the mean. Moreover, extensive studies using hundreds of probability models have verified that at least 97.5% of the possible values lie within three STDVs (Wheeler, Chambers, et al., 1992). With that said, the ± vagueness range provided from the designers' experience covers the possible values, and the STDV is concluded by dividing the vagueness range into six regions, three deviations to the left and another three to the right of the mean as illustrated in Figure 3.6.

## 3.5 Meta-model design

The multi-LOD meta-model design provides a means for defining project-specific requirements. It defines the required components, including their LOD, at a specific building development level and incorporates formal LOD definitions for individual component types.

The multi-LOD meta-model introduces two levels: (1) the *data-model level* defines the component types as well as their geometric and semantic requirements for each LOD. Subsequently, the components' LODs are assigned to a BDL. (2) the *instance level* represents the actual building elements and their relationships at multiple LODs.

The meta-model design complies with the object-oriented modeling principles, which offers high flexibility and extensibility. It allows for a dynamic definition of any component type as well as its properties for the different LODs. This provides the flexibility required when dealing with different construction types, different domains, and different analysis tools. At the same time, the meta-model provides a consistent way to query information about LOD definitions at both the data-model level and instance level.

As illustrated in Figure 3.7, the data-model level consists of multiple Building Development Levels (BDLs) and component types. A component type definition is represented as a separate class, where it is linked to an IFC entity, *IfcWall* as an example, and associated with a list of LOD definitions. The component types are mapped to instances of the IFC data model. This

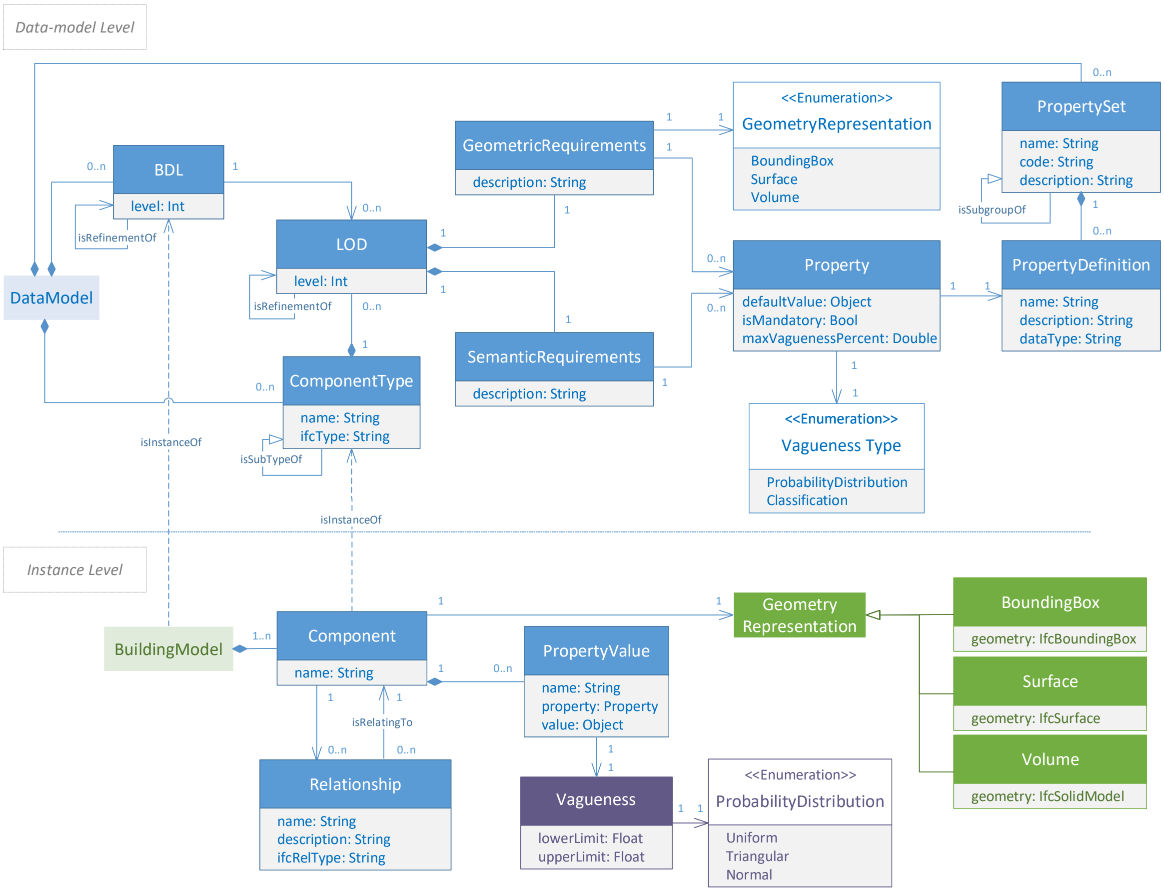


Figure 3.7: Multi-LOD meta-model (UML diagram)

allows on the one hand, to make use of the rich geometry representations provided by IFC and on the other hand, to experiment with real-world data produced by IFC-capable BIM authoring tools.

A component type LOD definition is produced out of two objects, geometric and semantic requirements. Both requirements are explicitly described in the form of properties, and at the same time, the geometric requirements allow for the specification of the required geometry representation.

The properties are managed separately by means of grouping, the *PropertySet* class. A *PropertySet* includes multiple *PropertyDefinition* instances that define property details but exclude its vagueness. The vagueness type and maximum percentage as well as whether the property is mandatory are specified when assigning a *PropertyDefinition* to an LOD property. This has multiple advantages, including the decoupling of the property definition from the LOD requirements, and flexibility in using the same property definition in multiple LODs along with different vagueness.

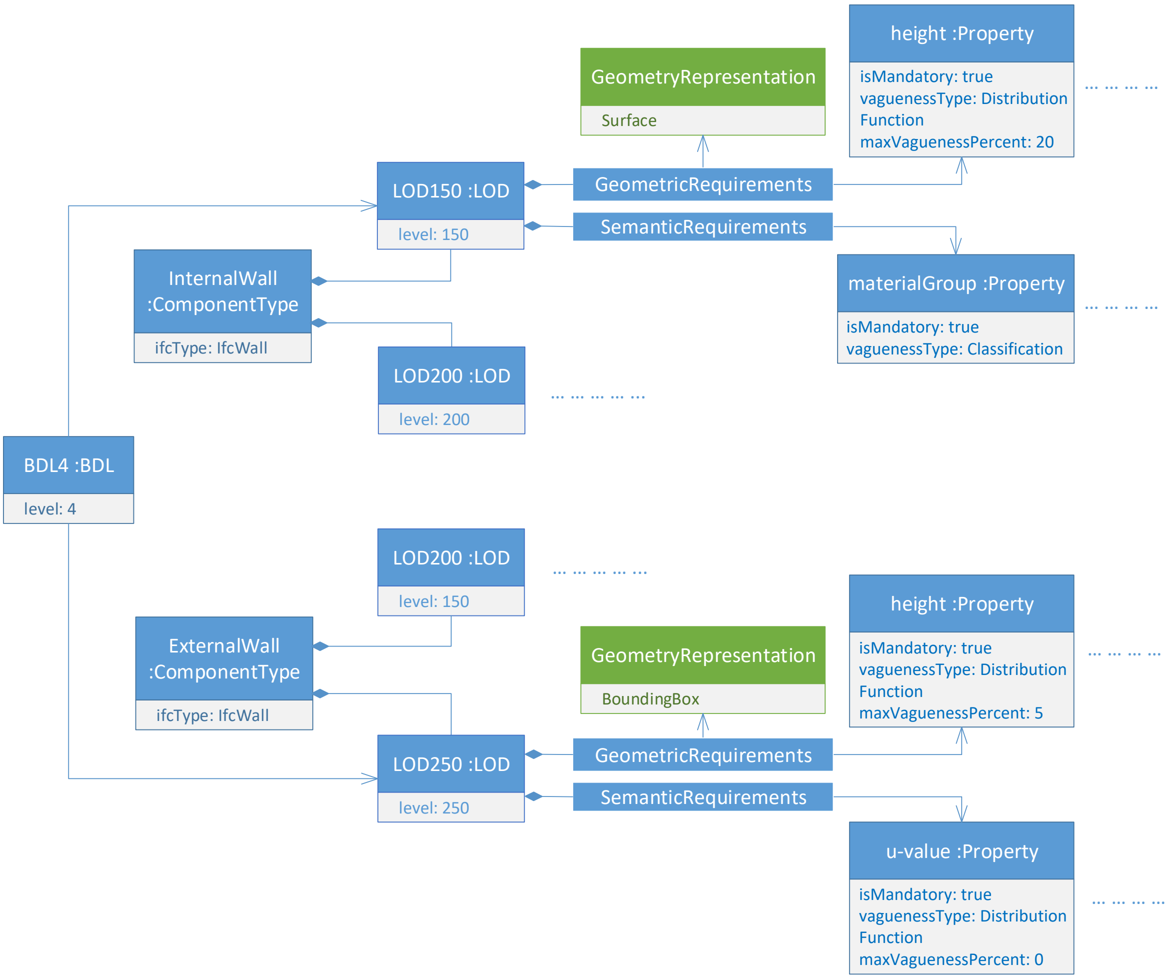


Figure 3.8: Instance of the multi-LOD meta-model for defining multiple component types' LODs and assigning the building model's requirements at BDL 4

In some cases, multiple components fall under the same category, such as *Heating*, *Ventilation*, *and Air Conditioning* (HVAC) systems, and share several properties. Hence, the *ComponentType* class supports the definition of the sub-types of a specific component type through inheritance. This means a sub-type inherits the parent component type's requirements in addition to specifying new requirements.

Thereafter, a BDL is comprised of a set of component types' LOD definitions to form the requirements of the overall building model. Figure 3.8 demonstrates the assignment of component types' LOD requirements for BDL 4. Each of the components is associated with two LODs, including geometric and semantic properties. BDL 4 here requires internal walls at LOD 150 and external walls at LOD 250.

After defining the component types' requirements, the building model is represented by multiple instances of the available types. Based on the defined requirements, each instance is assigned to a geometry representation, which complies with IFC, such as *IfcSurface*, and its properties are assigned to values. In terms of vagueness, a probability distribution function is specified and its range is automatically generated from the maximum vagueness percentage defined at the component type level. For example, 4% and an attribute value of 250 cm are translated into a range of ±10 cm. Moreover, at the instance level, it is possible to change the distribution function or increase the limitation of the range values, such as to between -5 cm and +7 cm.

Finally, the connections between the individual components within the same BDL, including aggregation and association, are presented through the *Relationship* class. The meta-model allows checking if the instance of a given component type at a particular LOD complies with the requirements defined in terms of semantics and geometric representation.

## 3.6 Consistency of BDLs

The design of the building model is developed through multiple BDLs. As a subsequent BDL brings additional information, new challenges arise. In some cases, overcoming these challenges requires the modification of previously made design decisions, like changing the structure of a load-bearing wall or moving a component into a different position, which is crucial for the model's structural integrity. Taking into consideration the collaborative nature of building projects, such modifications at an advanced BDL should be controlled properly in order to avoid any unexpected side-effects impacting the whole building model. Therefore, this paper proposes a methodology for checking the refinement consistency of the building components across the BDLs.

The building component's position, orientation, and dimensions define its existence in the overall model. This information is essential for many types of analyses, such as clash detection, where detecting whether a specific region touches or is included within another region is important.

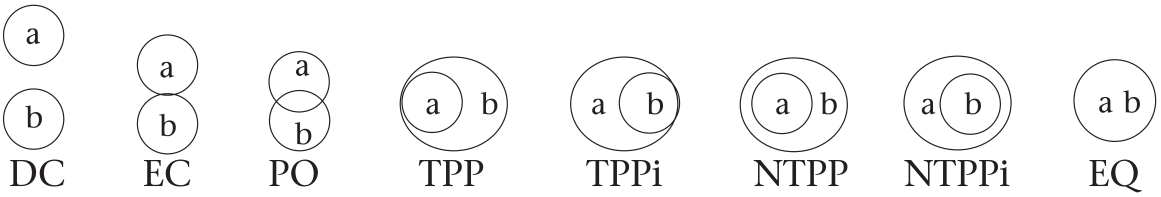


Figure 3.9: The Region-Connection Calculus (RCC) Representing Pairwise Relationships between Regions of Space (Li & Ying, 2003)

Qualitative Spatial Reasoning (QSR) provides representational primitives, a spatial vocabulary, and mechanisms for reasoning about the spatial data. The Region Connection Calculus (RCC) theory is a well-established formal system for qualitative spatial reasoning. It is based on a primitive connectedness relation, *C*, which is a binary symmetric relation (Bailey-Kellogg & Zhao, 2003). Using this relation, a set of binary relations are defined (Li & Ying, 2003) (some formal definitions are listed in Table 3.1). Most importantly, the eight relations illustrated in Figure 3.9, *{DC, EC, PO, TPP, TPPi, NTPP, NTPPi, EQ}*, form a Jointly Exhaustive and Pairwise Disjoint (JEPD) set, which means that any two regions stand to each other in exactly one of these relations. These eight topological relations are known as RCC8.

Table 3.1: Some definitions of the RCC relations (Li & Ying, 2003)

| **Relation** | **Interpretation** | **Definition of R(x, y)** |
|---|---|---|
| DC(x, y) | x is disconnected from y | $\neg C(x, y)$ |
| P(x, y) | x is a part of y | $\forall z[C(z, x) \rightarrow C(z, y)]$ |
| PP(x, y) | x is a proper part of y | $P(x, y) \wedge \neg P(x, y)$ |
| EQ(x, y) | x is identical with y | $P(y, x) \wedge P(y, x)$ |
| O(x, y) | x overlaps y | $\exists z[P(z, x) \wedge P(z, y)]$ |
| PO(x, y) | x partially overlaps y | $O(x, y) \wedge \neg P(x, y) \wedge \neg P(y, x)$ |
| EC(x, y) | x is externally connected to y | $C(x, y) \wedge \neg O(x, y)$ |
| DR(x, y) | x is discrete from y | $\neg O(x, y)$ |
| TPP(x, y) | x is a tangential proper part of y | $PP(x, y) \wedge \exists z[EC(z, x) \wedge EC(z, y)]$ |
| NTPP(x, y) | x is a non-tangential proper part of y | $PP(x, y) \wedge \neg \exists z[EC(z, x) \wedge EC(z, y)]$ |

### 3.6.1 Formal definition

The proposed methodology introduces a new relationship, *IsRefinedBy*, that represents the dependencies between the different BDLs. It comprises the geometric and semantic information as well as the topological relationships. Additionally, the permissible vagueness, i.e. the vagueness type and maximum percentage defined at the data-model level, at each LOD is taken into consideration. In order to consider a BDL refinement as consistent, it needs to at least conform to the information defined at the previous BDL. Consequently, each building component is represented by a set of components at the subsequent BDL, including their properties and relationships, which makes the BDLs interconnected and serves as the building model's refinement history.

Figure 3.10 represents the information validated for checking the consistency of a building model at two different levels. A building's topology is described as a network of adjacency relationships between all components (physical elements and spaces), see Figure 3.13. We define two BDLs as being consistent iff:

- The topological network of the objects and spaces at $BDL_x$ is topologically equivalent to the network at $BDL_y$ (explanation follows).

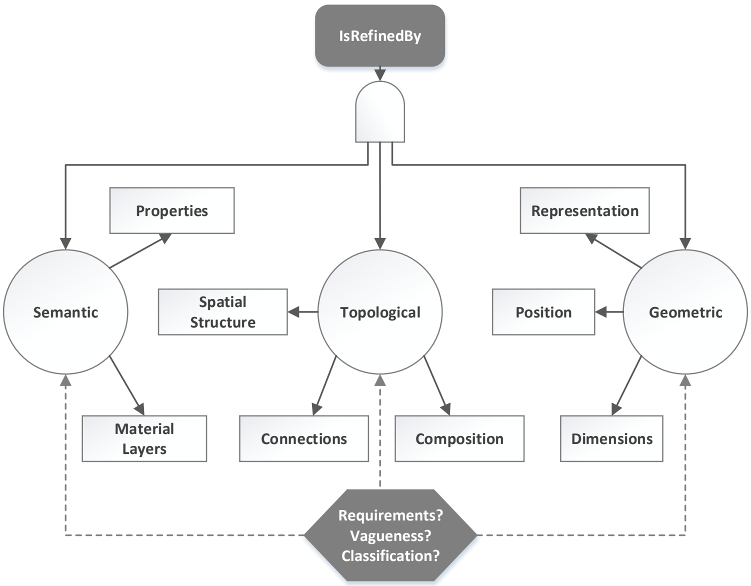


Figure 3.10: *IsRefinedBy* relationship composition

- If there is a refinement relationship between components $a \in BDL_x$ and $b \in BDL_y$, for all components $b$ holds: their position and size is *contained* (in the sense of $TPP$, $NTPP$, or $EQ$ of RCC8) in the geometric representation of $a$.
- If there is a refinement relationship between components $a \in BDL_x$ and $b \in BDL_y$, for all components $b$ holds: their semantic information (type and attributes) is a concretization of the semantic information of $a$.

### 3.6.2 Approach

To validate the consistency of two BDLs, multiple checks are conducted. To perform these checks, fundamental knowledge about the spatial relationships of the individual components at both BDLs is required. Thus, a pre-processing step mapping each component of $BDL_x$ to a set of components that occupy part or all of the same area at $BDL_y$ is performed. In this regard, qualitative spatial reasoning is applied to all the components by creating an *Axis Aligned Bounding Box* (AABB) around each component and finding the overlapping elements at the other BDL as depicted in Figure 3.11. Once there is a bounding box overlap, a *Ray / Triangle Intersection* (Möller & Trumbore, 2005) calculation is performed to accurately identify the mapped elements that are actually overlapping.

#### Topological consistency

First, the overall model's topological relationships are investigated. As changing the position and dimensions is allowed within a ± vagueness value, it is possible that a change results in a

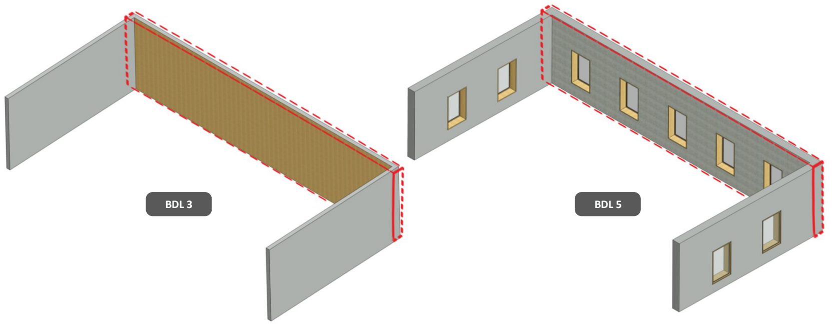


Figure 3.11: Pre-processing: matching components based on their position and dimensions, in this example, the wall at BDL 3 is matched with a more refined wall that has additional layers, openings, and multiple windows at BDL 5

critical modification of the building's topology as illustrated in Figure 3.12. Reducing *Wall05*'s dimensions within the allowed vagueness disconnects it from *Wall01*, which is critical, as it changes the function of *Wall05* from room dividing into non-room dividing. Such a change modifies the storey's spatial structure from two spaces into one space, which has a critical effect on various aspects, including the designed compartments for fire-safety regulations, life-cycle analysis, and load distribution in case the wall is load-bearing.

Consequently, the refined model is not considered consistent since it does not comply with the decisions made at the previous BDL. For that reason, it is necessary to maintain the building's topological relationships in a way that preserves the spatial structure's consistency. A more refined BDL can include additional / more detailed components or a more complex spatial structure, but it should at least comply with the spatial structure provided by the previous BDL.

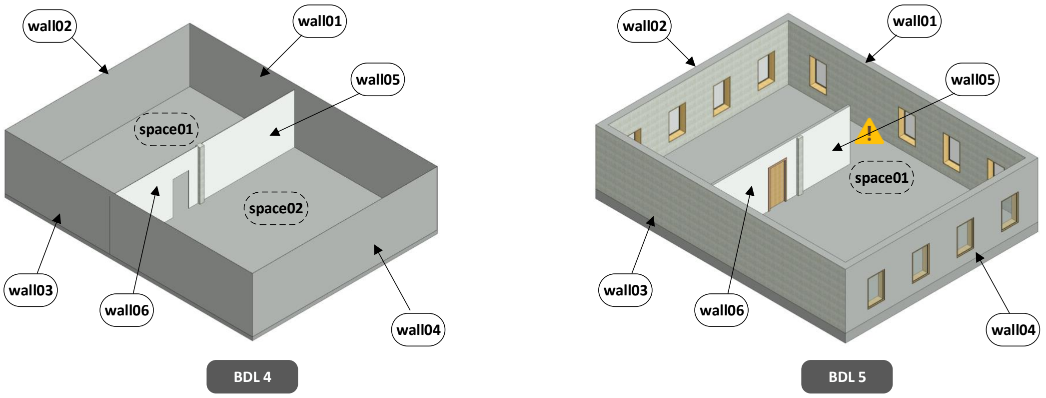


Figure 3.12: Demonstrating the motivation for maintaining the spatial structure's consistency across the BDLs by showing the effect of changing *Wall05*'s dimensions with the permissible vagueness at a subsequent BDL. Consequently, the function of *Wall05* has changed from room dividing into non-room dividing at BDL 5, modifying the spatial structure from two spaces into one space

Thereby, the proposed methodology aims to construct a labeled-graph representation of the building's spatial structure by including the available spaces, their boundaries, and the relationships between them. In this way, the topological complexity is simplified into graphs, which facilitates the comparison of two BDLs.

However, although information about the available spaces and their boundaries are supported by the IFC schema, using *IfcSpace* components and *IfcRelSpaceBoundary* relationships, they are not automatically exported by the BIM authoring tools (Lilis et al., 2017). Instead, they need to be either manually modeled or computationally determined. Similarly, the connections between walls and other boundaries, such as columns, are not automatically exported. Therefore, the RCC8 relations ($PO$, $EC$, and $DC$) are applied to extract the connections between the geometric components. As a result, a graph is constructed of the connected components, such as walls and columns, as vertices. Next, the bounded spaces are extracted by finding all the graph cycle spaces, a graph theory technique.

"A *Cycle Space* of a graph G, denoted $\text{W}_C(\text{G})$, is the subset of the edge space $\text{W}_E(\text{G})$ consisting of the null set (graph)$\phi$, all cycles in G, and all unions of edge-disjoint cycles of G." (Gross & Yellen, 2005)

For instance, Figure 3.13 exhibits two BDLs, BDL 4 at the top and BDL 5 at the bottom, including their graph representation. At BDL 4, the graph results in three closed cycles:

- Storey space (Space01 + Space02): *wall01, wall02, wall03, wall04*
- Space01: *wall01, wall02, wall03, wall06, column01, wall05*
- Space02: *wall01, wall04, wall03, wall06, column01, wall05*

As at BDL 5, more precise information about the storey's interior structure and load distribution is available, the model is refined by splitting each of *wall01* and *wall03* into two smaller walls and adding a structural load-bearing column in between. Additionally, a new internal wall, *wall07*, is added. Consequently, the constructed graph has different patterns and vertices than BDL 4. When processing the graph, five closed cycles are found:

- Storey space (Space01 + Space02 + Space03): *wall01.2, column01.3, wall01.1, wall02, wall03.1, column03.3, wall03.2, wall04*
- Space02 + Space 03: *wall01.2, wall04, wall03.2, column03.3, wall06, column01, wall05, column01.3*
- Space01: *column01.3, wall01.1, wall02, wall03.1, column03.3, wall06, column01, wall05*

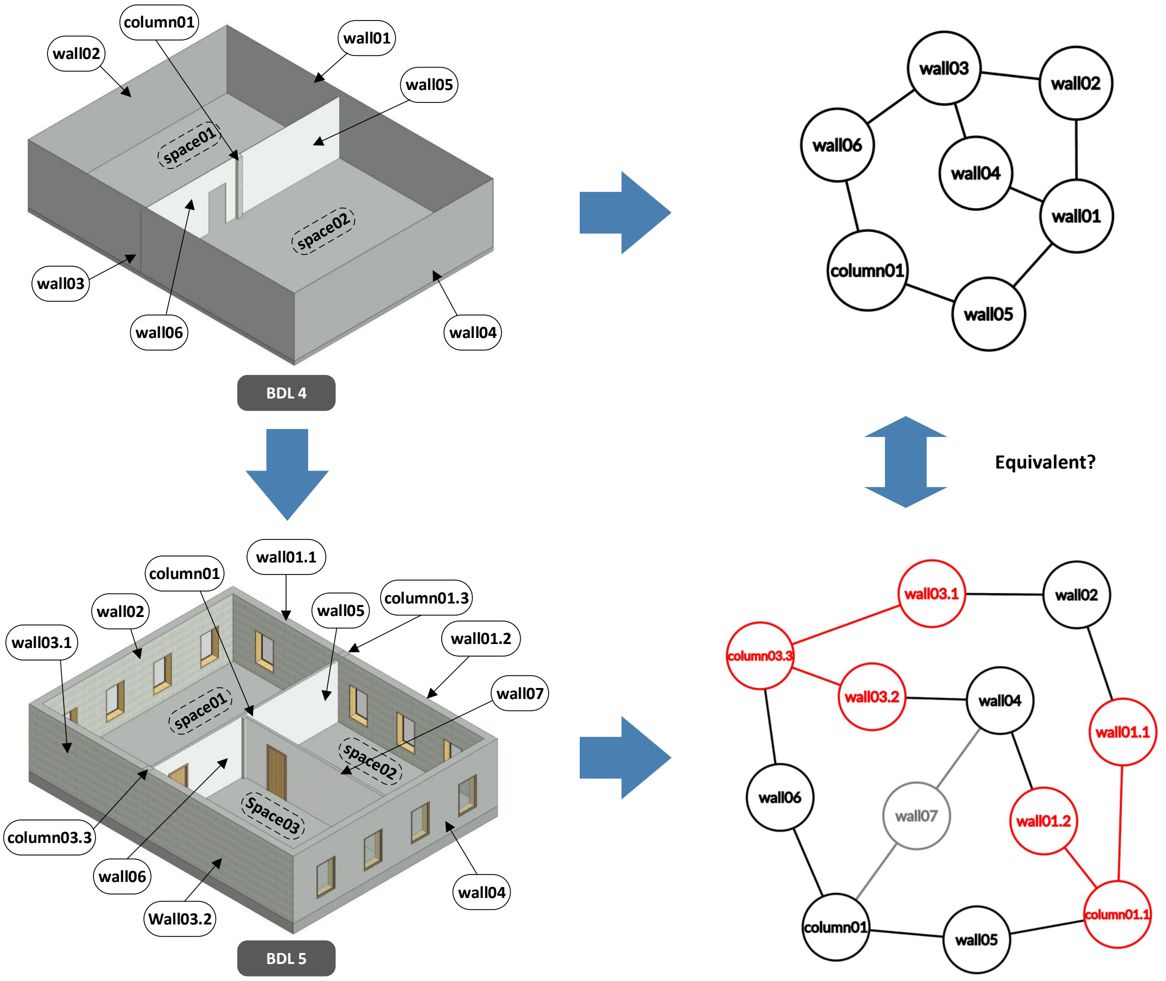


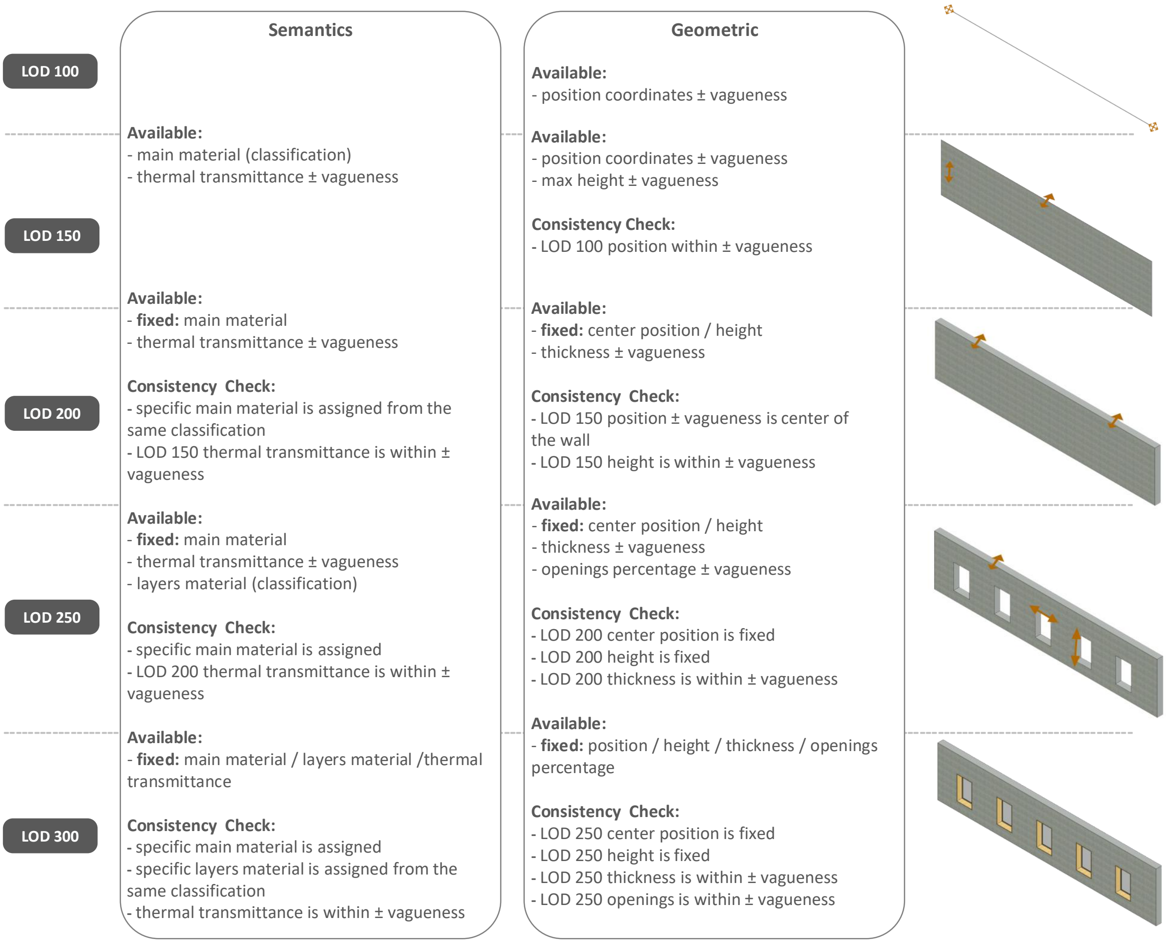
Figure 3.13: Labeled-graph representation of the building's spatial structure of two BDLs. The vertices represent the geometric components and the edges mean that there is a physical connection between two vertices

- Space02: *wall01.2, wall04, wall07, column01, wall05, column01.3*
- Space03: *wall04, wall03.2, column03.3, wall06, column01, wall07*

Next, the extracted cycles from both BDLs are compared for equivalency. In this context, the mapped components from the pre-processing step are replaced by the original component. In this example, *wall01.1*, *column01.1*, and *wall01.2* are replaced by *wall01*, and this is also the case for *wall03*. As a result, finding the exact cycles of BDL 4 as part of the BDL 5 cycles is guaranteed in case their topology is consistently refined. Finally, the relationships' correctness of the mapped components is investigated; if one wall is refined into two walls with openings, then the connections and voids relationships need to be assigned accordingly.

### Geometric and semantic consistency

The second check verifies whether the geometric information, including dimensions and position, and the semantics, like material, of two LODs comply with each other considering the permissible vagueness defined in the multi-LOD data model. The aim is to assure that each component refinement conforms to the decisions made at the previous LOD.



| | Semantics | Geometric |
|---|---|---|
| LOD 100 | | **Available:**<br>- position coordinates ± vagueness |
| LOD 150 | **Available:**<br>- main material (classification)<br>- thermal transmittance ± vagueness | **Available:**<br>- position coordinates ± vagueness<br>- max height ± vagueness<br>**Consistency Check:**<br>- LOD 100 position within ± vagueness |
| LOD 200 | **Available:**<br>- **fixed:** main material<br>- thermal transmittance ± vagueness<br>**Consistency Check:**<br>- specific main material is assigned from the same classification<br>- LOD 150 thermal transmittance is within ± vagueness | **Available:**<br>- **fixed:** center position / height<br>- thickness ± vagueness<br>**Consistency Check:**<br>- LOD 150 position ± vagueness is center of the wall<br>- LOD 150 height is within ± vagueness |
| LOD 250 | **Available:**<br>- **fixed:** main material<br>- thermal transmittance ± vagueness<br>- layers material (classification)<br>**Consistency Check:**<br>- specific main material is assigned<br>- LOD 200 thermal transmittance is within ± vagueness | **Available:**<br>- **fixed:** center position / height<br>- thickness ± vagueness<br>- openings percentage ± vagueness<br>**Consistency Check:**<br>- LOD 200 center position is fixed<br>- LOD 200 height is fixed<br>- LOD 200 thickness is within ± vagueness |
| LOD 300 | **Available:**<br>- **fixed:** main material / layers material /thermal transmittance<br>**Consistency Check:**<br>- specific main material is assigned<br>- specific layers material is assigned from the same classification<br>- thermal transmittance is within ± vagueness | **Available:**<br>- **fixed:** position / height / thickness / openings percentage<br>**Consistency Check:**<br>- LOD 250 center position is fixed<br>- LOD 250 height is fixed<br>- LOD 250 thickness is within ± vagueness<br>- LOD 250 openings is within ± vagueness |

Figure 3.14: An example of an external wall refinement, listing the available information and the geometric - semantic consistency checks

In more detail, Figure 3.14 demonstrates an external wall refinement, listing the available information and the consistency checks. In the beginning, information about the component's position, accompanied by vagueness, is available, which allows for a representation of the wall by a centerline. At LOD 150, the height of the wall can be estimated, which makes it possible to represent the wall as an extruded surface. The consistency check here focuses on maintaining the centerline position defined previously ±vagueness.

Afterwards, additional information about the wall material layers and insulation is available. Thus, the wall thickness can be estimated. In this case, checking the consistency involves verifying the wall's height and that the surface position ±vagueness represents the center of the wall.

In terms of semantic information, the consistency is checked based on its type. Semantics can have diverse types and meanings, including material layers, openings percentage, fire rating, thermal transmittance, and much more. Therefore, making sense of this information is a prerequisite for checking its consistency. Here, the defined requirements of the multi-LOD data-model provide additional context for mapping the same property between different LODs.

The data-model explicitly defines the property type in addition to the vagueness type and percentage, which yields a formal specification of the expected values at the refined LOD. Furthermore, mapping the defined properties to the classification systems, like *Uniclass* and OmniClass, as well as to the commonly known property sets, like *Pset_SlabCommon*, assists in validating the refinement consistency. For instance, when a *Ceramic* material group is specified at LOD 150, at LOD 200 an exact material that belongs to this group, such as *Brick*, *Earthenware*, and *Terracotta*, should be assigned.

## 3.7 Prototype

To evaluate the proposed multi-LOD model for practical use, it is implemented as a webserver and a client-side User Interface (UI), providing a user-friendly way to define disciplines, levels of development, property sets, component types, and building development levels.

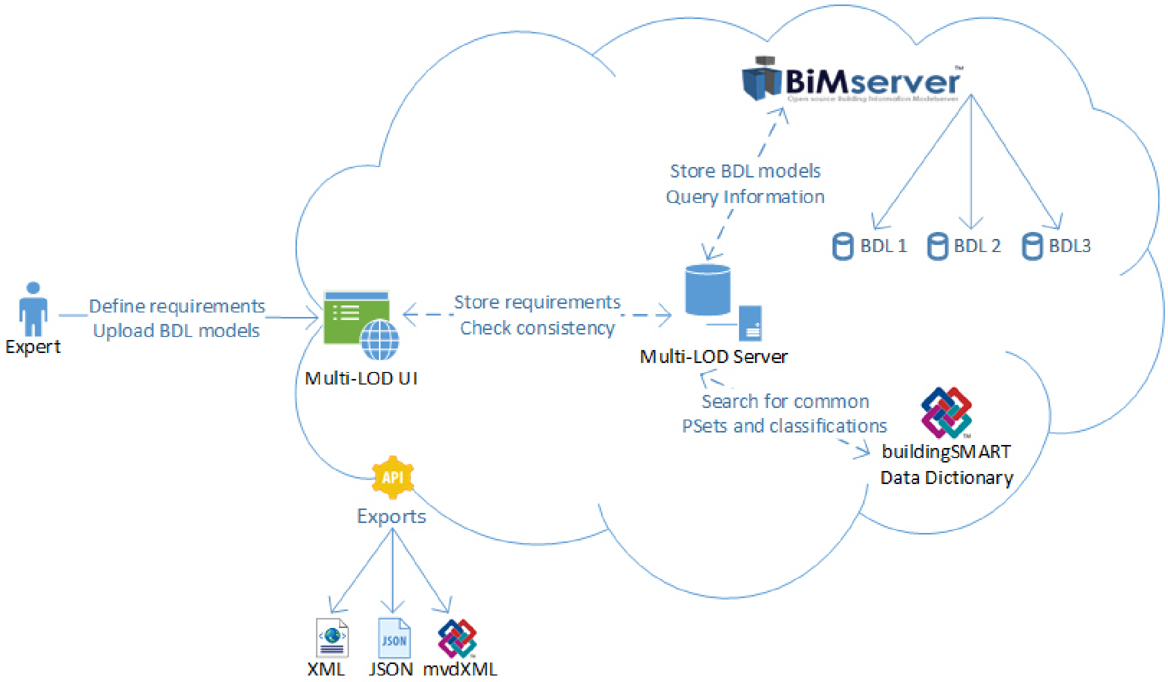
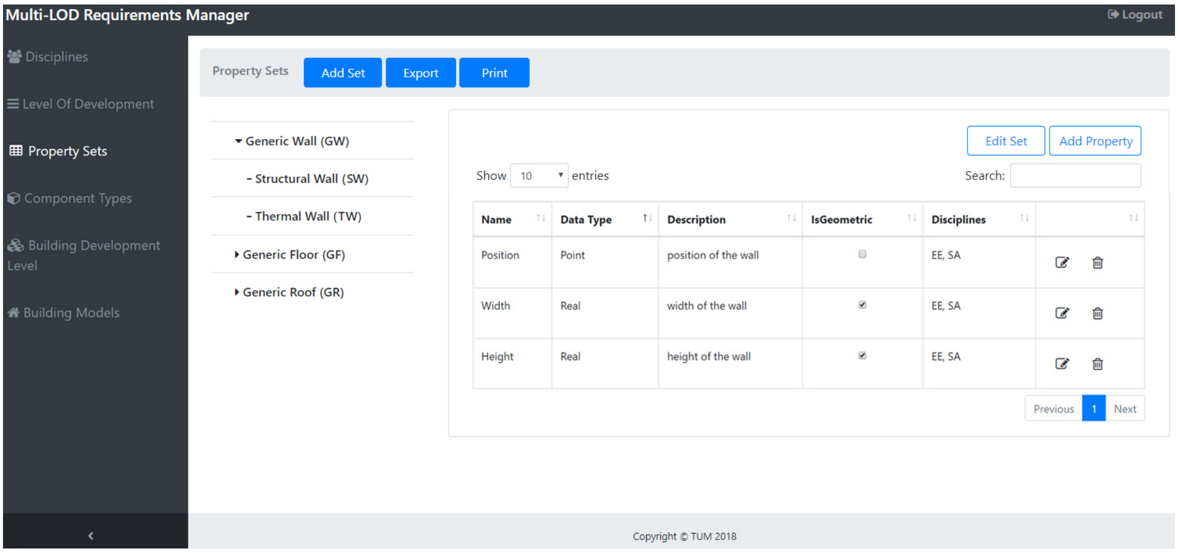


Figure 3.15: Overview of system design

The webserver alleviates the disciplines' collaboration by centralizing the storage of exchange requirements and building models' information and providing web-service access for all modeling, simulation, and analysis tools. Maintaining and managing the actual building models (at different BDLs) is realized by employing an instance of the BIMServer (Beetz et al., 2010), thus functioning as a back-end. Figure 3.15 provides an overview of the system design.



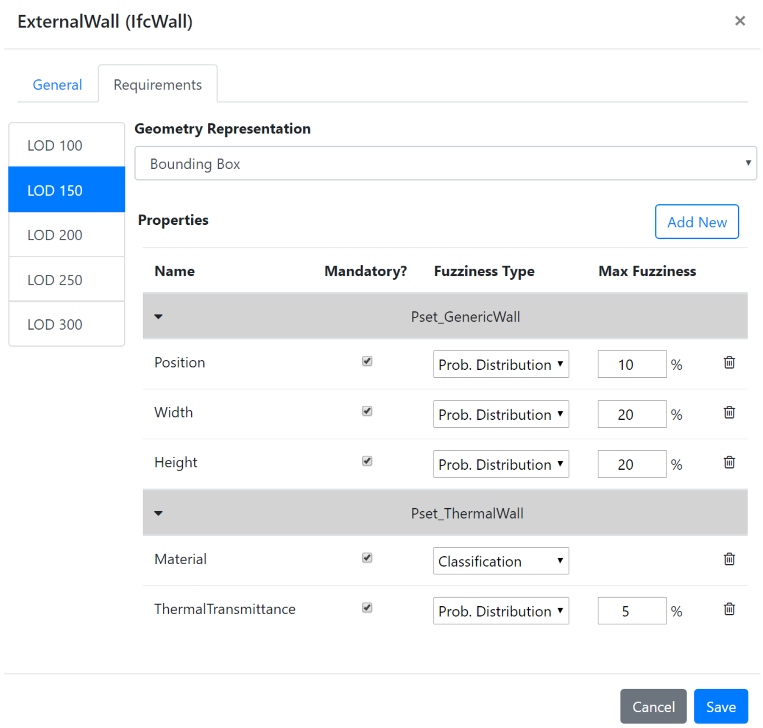
| Name | Data Type | Description | IsGeometric | Disciplines | |
|---|---|---|---|---|---|
| Position | Point | position of the wall | ☐ | EE, SA | |
| Width | Real | width of the wall | ☑ | EE, SA | |
| Height | Real | height of the wall | ☑ | EE, SA | |

Figure 3.16: Property Sets management screen (UI prototype)

The main concept is that every discipline is capable of defining its own property sets and then assigning particular properties to a specific component type's LOD. The property sets' management screen is demonstrated in Figure 3.16. A property set can have sub-sets in order to minimize the properties' redundancy. Additionally, a property is assignable to multiple disciplines.

Afterwards, the properties are assigned to an LOD at the component types' screen. Figure 3.17 shows the component details screen for an *ExternalWall*. The *General* tab is for defining the component name, IfcType, description, and whether the component is external and load-bearing. The second tab, *Requirements*, facilitates the association of every LOD with properties including a specification of their vagueness. The properties are grouped based on their *Property Set* name, following the naming scheme *Pset_**, for instance *Pset_ThermalWall*.

To improve the usability and increase the data integrity, the buildingSmart Data Dictionary's (bsDD) Application Programming Interface (API) (buildingSMART, 2016) is employed. It assists the process by listing the commonly known IFC elements, properties, and classifications to the user. Consequently, this mapping to the bsDD's GUID provides additional context and meaning to each value, which improves interoperability between different disciplines and assists in the model's analysis.



**ExternalWall (IfcWall)**

**Geometry Representation**: Bounding Box

**Properties**

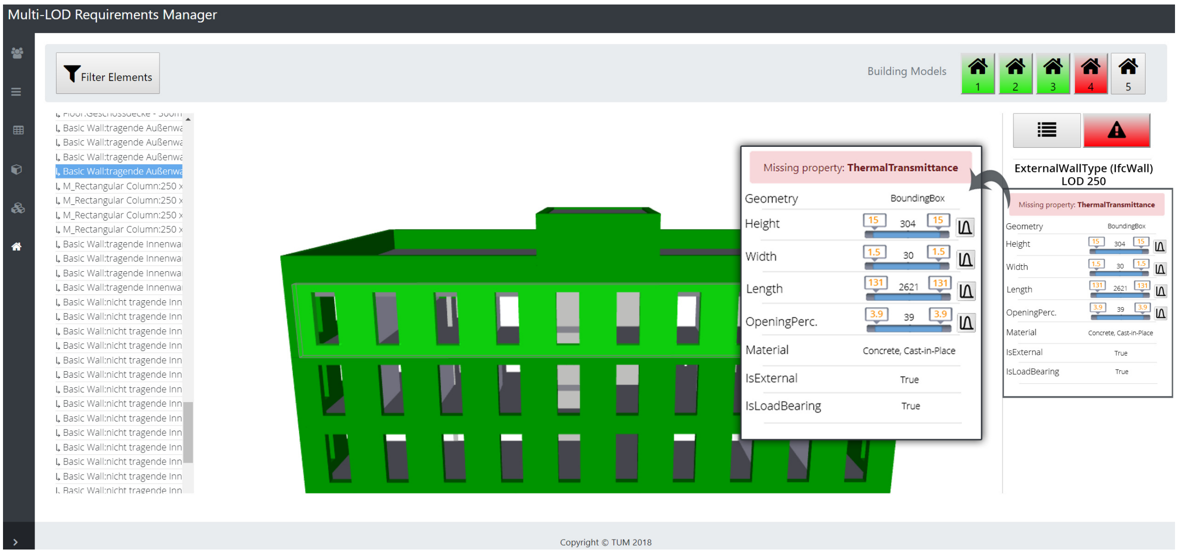
| Name | Mandatory? | Fuzziness Type | Max Fuzziness |
|---|---|---|---|
| Pset_GenericWall | | | |
| Position | ☑ | Prob. Distribution | 10 % |
| Width | ☑ | Prob. Distribution | 20 % |
| Height | ☑ | Prob. Distribution | 20 % |
| Pset_ThermalWall | | | |
| Material | ☑ | Classification | |
| ThermalTransmittance | ☑ | Prob. Distribution | 5 % |

Figure 3.17: Component details screen of an ExternalWall; the vagueness percentages are estimated based on an interpretation of the BIMForum's definitions and domain knowledge (UI prototype)

The multi-LOD webserver stores the component types' requirements into a relational database and exports them as XML and JSON formats using the REpresentational State Transfer (REST) API. To facilitate the usage of these exchange requirements and validate their existence, the webserver exports them into the common formats supported by BIM authoring tools, such as a PropertySets file provided by Autodesk Revit, and automatically generated mvdXML rules. Hereby, it is possible to use the requirements for external services, such as a Revit plugin, to automatically generate and ensure the exchanged building models' attributes completeness.

After defining the LOD requirements, the experts are able to share and validate their developed building models. As shown in Figure 3.18, an expert selects a particular building's BDL (from the buttons on the top) and uploads its corresponding IFC file to the system. When BDL 1 is uploaded, the multi-LOD service checks its compliance with the defined requirements, i.e. if all the mandatory properties exist and the geometry representation is as specified; in case it is valid, then it is stored at the BIMServer, otherwise, the expert is notified.

When the next BDL is uploaded, the same check regarding the defined requirements is performed, and then the information refinement consistency with the previous stage is verified using the approach described in Section 3.6. To retrieve the building model's information, the BIMServer provides a convenient implementation of BIMQL (Mazairac & Beetz, 2013). Additionally, to check the BDLs' topological consistency, the QL4BIM (Daum et al., 2017)



Missing property: **ThermalTransmittance**

Geometry BoundingBox
Height 15 304 15
Width 1.5 30 1.5
Length 131 2621 131
OpeningPerc. 3.9 39 3.9
Material Concrete, Cast-in-Place
IsExternal True
IsLoadBearing True

Figure 3.18: Building models management screen; an expert selects a BDL and uploads an IFC file. The defined requirements, as well as the model's consistency, are verified. On the right side, the component's properties are mapped to the defined LOD requirements and the expert is notified when a property is missing

is integrated into the process to query the connected components and generate a graph representation.

As demonstrated in Figure 3.18, the building model expects the external walls to be at LOD 250, which requires the *ThermalTransmittance* property to exist. Besides listing the component's properties and their defined vagueness, the user interface indicates that there is a required property missing for the highlighted external wall. The multi-LOD service serves as a gate for maintaining the model's consistency when updating or adding a new BDL.

To assist in checking the building models' completeness and consistency beforehand, the generated mvdXML rules can check the IFC models locally before uploading them to the system. For example, Listing 3.1 shows two mvdXML rules; the first rule checks the consistency of the *ThermalTransmittance* property value between two different LODs. The range limitation is generated by retrieving the value of the same property from the available LOD and multiply it by the allowed vagueness percentage, while the second mvdXML rule is formed from the list of the available materials assigned to the *Ceramic* material group in the *OmniClass* classification system.

Algorithm 3.1: mvdXML rules checking the consistency of *ThermalTransmittance* and *Material* between two different LODs

```
<TemplateRule Parameters="PSet[Value]='Pset_ThermalWall'␣AND␣PropertyName
    [Value]='ThermalTransmittance'␣AND␣PropertyValue[Exists]=TRUE␣AND␣
    PropertyValue[Value]␣>=␣0.15␣AND␣PropertyValue[Value]␣<=␣0.50"/>
```

```
<TemplateRule Parameters="PSet[Value]='Pset_StructuralWall'␣AND␣
    PropertyName[Value]='Material'␣AND␣PropertyValue[Exists]=TRUE␣AND␣
    PropertyValue[Value]␣=␣'Brick'␣OR␣PropertyValue[Value]␣=␣'Earthenware'
    ␣OR␣PropertyValue[Value]␣=␣'Terracotta'␣OR␣PropertyValue[Value]␣=␣'
    Fired␣Shale'␣OR␣PropertyValue[Value]␣=␣'Porcelain'␣OR␣PropertyValue[
    Value]␣=␣'Vitreous␣China'"/>
```

## 3.8 Case study: Design of the Tausendpfund building

Figure 3.19: *Ferdinand Tausendpfund GmbH & Co. KG* office building, in Regensburg, Germany built in 2017. It has three storeys and is 27m long, 14.7m wide, and 9.8m tall. The gross volume is approx. 3950 $m^3$, with a gross area of 1290.5 $m^2$ and a window-to-wall ratio of 25%

In this case study, the proposed approach was applied to the definition of the exchange requirements and to check the consistency across the BDLs of the real-world construction project depicted in Figure 3.19. The benefits of specifying the information vagueness to reduce the uncertainty and support the decisions are presented below. The targeted type of analysis is the Life Cycle Assessment (LCA) calculation and its corresponding Embedded GreenHouse Gases (EGHG) in the early design stages.

LCA is one of the most established and well-developed methods for assessing the potential environmental impacts and resource consumption throughout a product's life-cycle (Ness et al., 2007). As one of its applications, LCA is used to calculate the embedded energy, which is represented as the sum of non-renewable energy consumption during a building's life cycle (Merkblatt, 2010). The GreenHouse Gases (GHG) emissions resulting from the embedded energy are defined as EGHG. Performing the LCA calculation involves a variety of geometric and semantic information, including the building location, dimensions, number of storeys, material, and window-to-wall ratio. Additionally, custom energy-related attributes, such as the *Thermal Transmittance* (U-value), are required for each component and need to be transferred when exchanging the model.

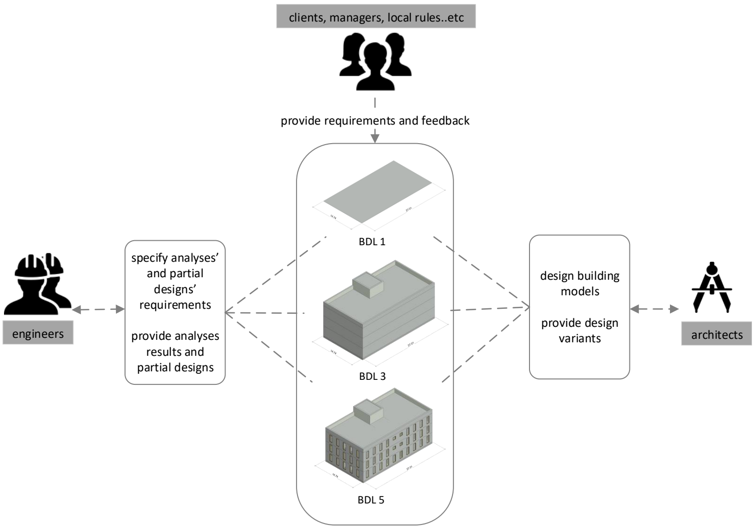


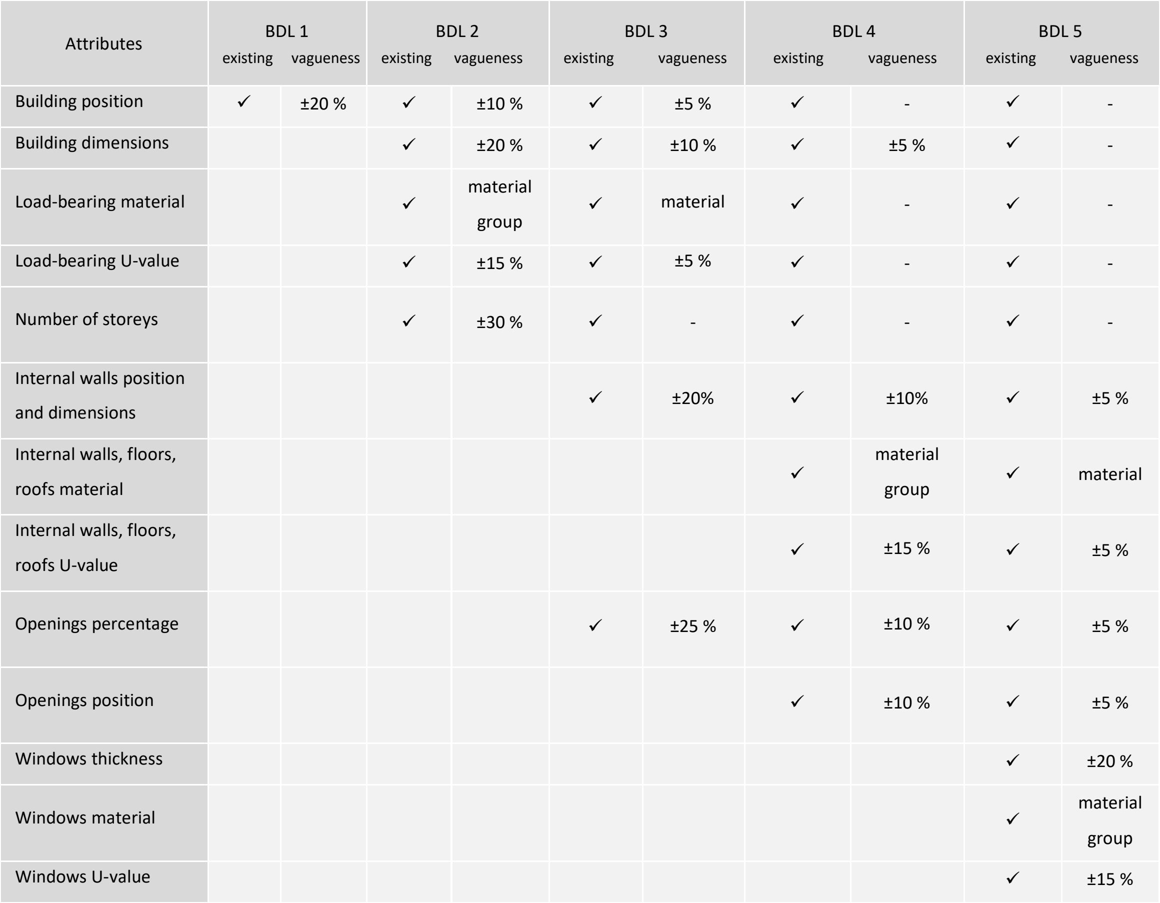
Figure 3.20: Collaboration between several disciplines to define a building project's requirements and objectives

Our research group includes architects and several engineers specialized in embedded and operational energy as well as structural analysis. At each design stage, engineers and architects need a detailed list of requirements to exchange building information models.

Figure 3.20 illustrates the collaborative process between several actors when developing a building. At every building development level, each discipline requires specific information to be present in the model to perform a model analysis. Similarly, architects incorporate clients' feedback and engineers' analyses results in the building models and produce design variants. Supporting the different kinds of evaluations for the same model is a very challenging task, as the information needs to represent the attributes and types of vagueness in a way that allows the various simulation tools to integrate them in the correct way. Here, the multi-LOD data-model comes into play, as it enables the requirements of the individual component types to be defined at every LOD.

While developing the conceptual design, the owner decided to build a sustainable building and explore multiple design variants, such as different numbers of storeys, a window-to-wall ratio for each side of the building, and different building dimensions.

Figure 3.21 lists the required attributes for LCA calculation in BDLs 1 – 5. The set of attributes and their associated vagueness are estimated by the research group's engineers based on domain knowledge, interpretation of the BIMForum's definitions, and numerous studies on the required information for energy performance simulation (Bambardekar & Poerschke, 2009; de Souza, 2012; Ellis & Mathews, 2001; Weytjens & Verbeeck, 2010).



| Attributes | BDL 1 | | BDL 2 | | BDL 3 | | BDL 4 | | BDL 5 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | existing | vagueness | existing | vagueness | existing | vagueness | existing | vagueness | existing | vagueness |
| Building position | ✓ | ±20 % | ✓ | ±10 % | ✓ | ±5 % | ✓ | - | ✓ | - |
| Building dimensions | | | ✓ | ±20 % | ✓ | ±10 % | ✓ | ±5 % | ✓ | - |
| Load-bearing material | | | ✓ | material group | ✓ | material | ✓ | - | ✓ | - |
| Load-bearing U-value | | | ✓ | ±15 % | ✓ | ±5 % | ✓ | - | ✓ | - |
| Number of storeys | | | ✓ | ±30 % | ✓ | - | ✓ | - | ✓ | - |
| Internal walls position and dimensions | | | | | ✓ | ±20% | ✓ | ±10% | ✓ | ±5 % |
| Internal walls, floors, roofs material | | | | | | | ✓ | material group | ✓ | material |
| Internal walls, floors, roofs U-value | | | | | | | ✓ | ±15 % | ✓ | ±5 % |
| Openings percentage | | | | | ✓ | ±25 % | ✓ | ±10 % | ✓ | ±5 % |
| Openings position | | | | | | | ✓ | ±10 % | ✓ | ±5 % |
| Windows thickness | | | | | | | | | ✓ | ±20 % |
| Windows material | | | | | | | | | ✓ | material group |
| Windows U-value | | | | | | | | | ✓ | ±15 % |

Figure 3.21: Required building attributes for LCA calculation in the early design stages (vagueness percentages are estimated based on domain knowledge and interpretation of the BIMForum's definitions)

Using the Multi-LOD user interface, the LCA requirements are defined and assigned to component types. For each BDL, a set of components and their LOD definitions, including vagueness type and percentage, are specified. For instance, in BDL 2, the building is associated with fuzzy dimensions, position, and a number of storeys. Load-bearing components, such as *Columns*, *External Walls*, and *Foundation*, are associated with thickness, material and U-value.

Estimating the attributes with a vagueness percentage makes performing the LCA calculation on an earlier BDL viable. In this way, the impact of each attribute on the calculation results can be assessed. This makes it possible to make better decisions that improve the building's performance during the building's life cycle and fit into the design intentions (Hopfe & Hensen, 2011).

As part of our research group, Harter et al., 2018 used the methodology proposed in this paper to calculate the EGHG for the proposed variants. Figure 3.22 illustrates how the information vagueness across the BDLs influences the uncertainty in the EGHG calculation.

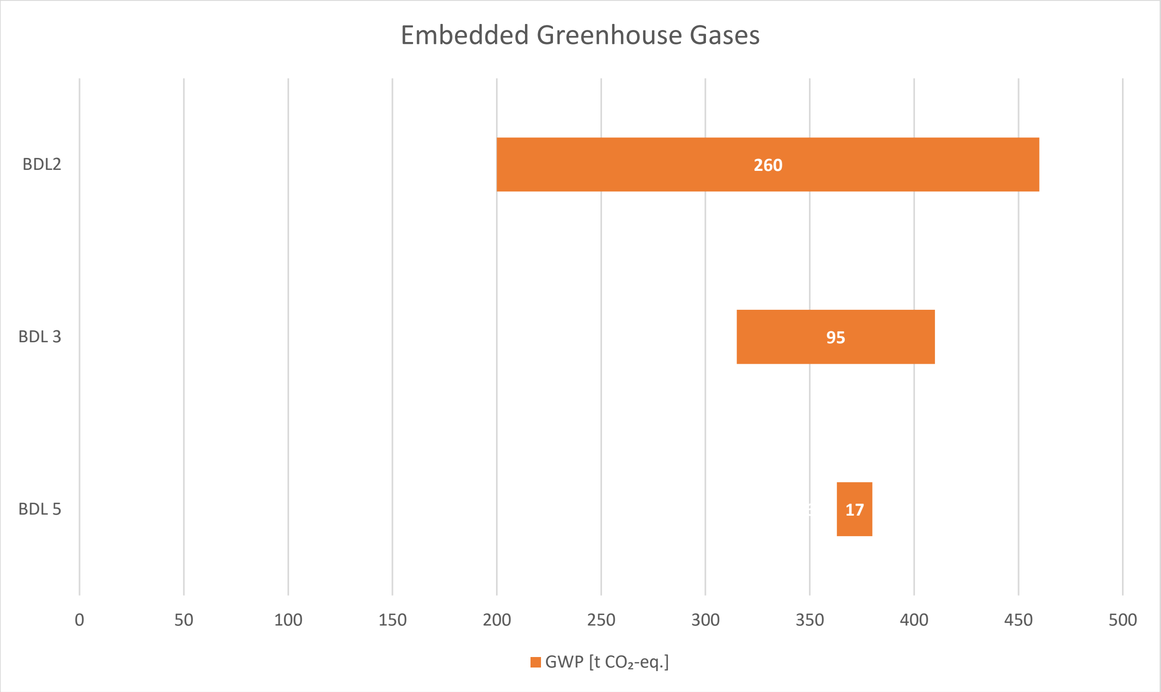


Figure 3.22: Comparison of the uncertainty range in EGHG results for BDLs 2, 3, and 5 (Harter et al., 2018), regenerated with permission

The uncertainty of the results decreases in inverse proportion to the increase in BDL, from a difference of 260 GWP [t $CO_2$-eq.] in BDL 2 to 17 GWP [t $CO_2$-eq.] in BDL 5. Hereby, the previously performed analyses' results are still considered valid and become more accurate by including the more precise information.

With the building model in BDL 2, multiple concepts were proposed. Figure 3.23 compares the EGHG results of the building model in BDL 2 with the impact of varying the building's dimensions by ±10%, window-to-wall ratio to 25% and 50%, and dividing the building into two and three storeys. The simulation results act as a weighting approach for the potential vagueness, i.e. they shed a light on which attributes have the greatest influence on the evaluation results compared to the others, which improves the designer's awareness and the quality of the decisions made.

At BDL 4, the interior structure, including the rooms' division and usage, was selected. Figure 3.24 depicts the floor plan layout of Level 0. At this BDL, the different kinds of analysis were performed and the results evaluated in terms of how they fulfill the project's requirements. The building model was then uploaded to the multi-LOD system. The system compared the uploaded model with BDL 3 and since the changes involved adding openings and interior walls, BDL 4 was successfully stored as a consistent refinement of BDL 3.

At BDL 5, the owner requested two design changes: (1) replacing one of the walls surrounding the staircase by one curtain wall and adding two structural columns, (2) reducing the height

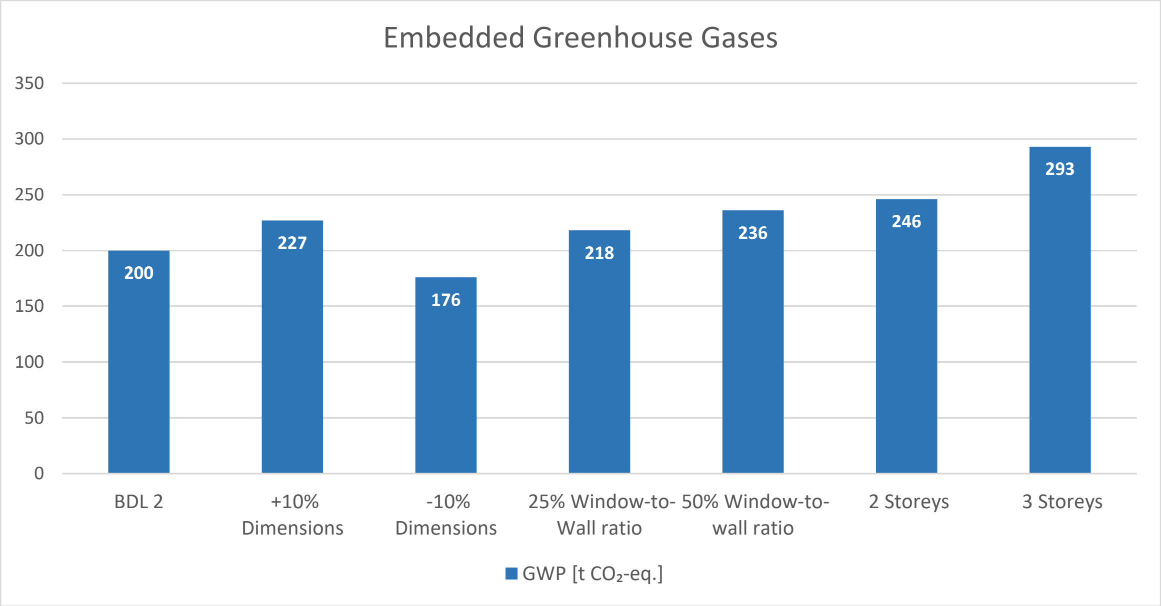
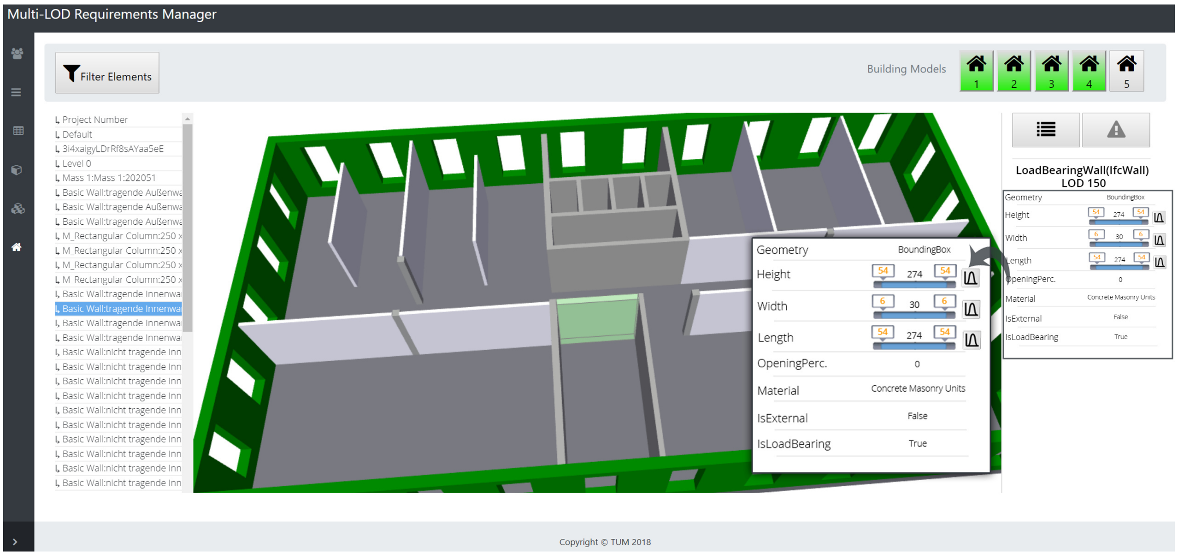


Figure 3.23: Impact of the information vagueness on EGHG results in comparison to BDL 2 (Harter et al., 2018), regenerated with permission



↳ Project Number
↳ Default
↳ 3i4xaigyLDrRf8sAYaa5eE
↳ Level 0
↳ Mass 1:Mass 1:202051
↳ Basic Wall:tragende Außenwa
↳ Basic Wall:tragende Außenwa
↳ Basic Wall:tragende Außenwa
↳ M_Rectangular Column:250 x
↳ M_Rectangular Column:250 x
↳ M_Rectangular Column:250 x
↳ M_Rectangular Column:250 x
↳ Basic Wall:tragende Innenwa
↳ Basic Wall:tragende Innenwa
↳ Basic Wall:tragende Innenwa
↳ Basic Wall:tragende Innenwa
↳ Basic Wall:nicht tragende Inn
↳ Basic Wall:nicht tragende Inn
↳ Basic Wall:nicht tragende Inn
↳ Basic Wall:nicht tragende Inn
↳ Basic Wall:nicht tragende Inn
↳ Basic Wall:nicht tragende Inn
↳ Basic Wall:nicht tragende Inn
↳ Basic Wall:nicht tragende Inn
↳ Basic Wall:nicht tragende Inn
↳ Basic Wall:nicht tragende Inn

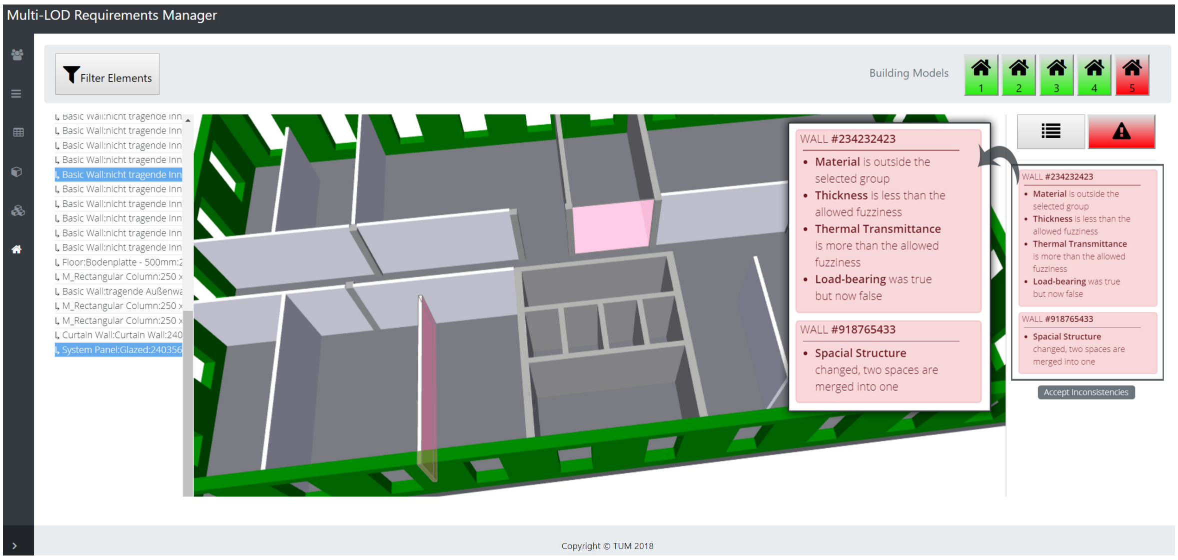
LoadBearingWall(IfcWall)
LOD 150

| | |
|---|---|
| Geometry | BoundingBox |
| Height | 54 274 54 |
| Width | 6 30 6 |
| Length | 54 274 54 |
| OpeningPerc. | 0 |
| Material | Concrete Masonry Units |
| IsExternal | False |
| IsLoadBearing | True |

Figure 3.24: Floor plan of level 0 at BDL 4 (UI prototype)

of one of the interior walls to allow for smooth communication between two of the offices, as their usage is similar. At BDL 4, the staircase walls are designed as load-bearing with 240mm concrete masonry units, and the offices were completely separated.

Although these changes satisfied the owner's request, they did not follow the decisions made in the earlier stages. Changing the wall's material and merging two spaces into one are major decisions that affect the different kinds of analysis and evaluations, such as EGHG, heat-flow, the structural system, and satisfying the fire-safety regulations. To guarantee that these changes did not affect the analyses performed previously and are at least equivalent to the previous design, the analyses need to be repeated.



WALL **#234232423**

- **Material** is outside the selected group
- **Thickness** is less than the allowed fuzziness
- **Thermal Transmittance** is more than the allowed fuzziness
- **Load-bearing** was true but now false

WALL **#918765433**

- **Spacial Structure** changed, two spaces are merged into one

WALL **#234232423**

- **Material** is outside the selected group
- **Thickness** is less than the allowed fuzziness
- **Thermal Transmittance** is more than the allowed fuzziness
- **Load-bearing** was true but now false

WALL **#918765433**

- **Spacial Structure** changed, two spaces are merged into one

Figure 3.25: Detection of inconsistent refinement at BDL 5 (UI prototype)

Consequently, the system considered the building model at BDL 5 as inconsistent and flagged a warning to the designer, as shown in Figure 3.25. At this point, the designer can make the decision to re-evaluate the model before approving this change or accept the changes and upload the current building model to the system as BDL 5.

Using the BDL concept to describe the building model development offered a spatial overview of the project and encouraged consideration of the different use-cases. Additionally, explicitly modeling the information vagueness facilitated an evaluation of the impact of the different attributes on the building performance. The presented approach assisted in making informed decisions and reduced the likelihood of having to perform major changes to the model at later stages, which in turn prevented a substantial amount of rework and added expenditure.

## 3.9 Conclusion & future work

This paper has contributed a new approach for the formal specification of maturity levels of building information models, in particular for the early stages of building design. To facilitate the early integration of analyses and simulations, this paper has proposed extending the BIMForum's LOD specification by adding intermediate levels to specify the maturity levels in a more fine-grained granularity. Additionally, the *Building Development Level (BDL)* has been introduced as a means to describe the required maturity of an entire digital building at a particular stage, through the composition of component-wise LOD specifications.

To enable the precise specification of a BDL/LOD content, a multi-LOD meta-model has been introduced. It offers an interface on the meta-level for specifying and querying the BDL definitions of buildings and the LOD definitions of individual component types. The meta-model provides two levels, the *data-model level* and the *instance level*. This offers a high degree of flexibility in defining per-project BDL/LOD requirements. Most importantly, it supports the formal checking of a building model's conformance with the defined semantic and geometric requirements at a specific stage or for a specific application, such as building performance simulations or structural analyses.

In particular, the proposed multi-LOD meta-model allows to explicitly define the vagueness of geometric and semantic information, both for defining the requirements of an LOD and for specifying information of a concrete building model. This allows to check a building model for formal conformance with the specification of an LOD, not only with respect to the existence of properties and the provision of values within a given range, but also with respect to the maximum allowed vagueness on a given LOD. The definition of vagueness on the instance level, on the other hand, delivers significant advantages in assessing the building's performance at the early design stages, as simulations and analyses can make direct use of the modeled uncertainties.

Finally, the explicitly defined vagueness allows verifying the building model's consistency across different BDLs. This enables tracking whether earlier assumptions still hold after the design process has progressed and the building model has been correspondingly refined. This, in turn, gives a strong indication whether the results of simulation performed on coarser BLDs still hold.

As a proof of concept, the meta-model has been prototypically implemented in a client-server software system based on web technologies. The system provides a means for managing the component types' LOD definitions and BDLs' requirements. On top of this, the building models are maintained throughout the BDLs, where they are checked for consistency and compliance with the defined requirements. The system exports the LOD definitions into JSON, XML, and automatically generated mvdXML rules to encourage their integration in the modeling process. To check the consistency across multiple BDLs, the building's topology is evaluated for equivalency and the individual components' geometric, semantic and topological information refinement is validated.

As demonstrated in the case study, the feasibility of the proposed approach was validated on a real-world construction project. The project participants emphasized the advantage of specifying the required information along with its potential vagueness in communicating the uncertainties in the input as well as the simulation results. Moreover, checking the building model's refinement consistency prevented a disregarding the previously made decisions and flagged up the necessity to repeat the performed analysis.

Despite its expressive power and flexibility in defining LOD requirements and checking the refinement consistency, the presented approach also has limitations. On the one hand, the refinement and detailing process remains a manual activity, i.e. the presented approach does not provide a consistency preservation mechanism, but only an inconsistency detection mechanism. On the other hand, as of now, there is no defined response in the detection of inconsistencies between different BDLs. Whether the coarser model would need to be updated or the finer one would be discarded heavily depends on the detailing work-flow and the goals associated with it.

As a next step, further research is necessary to support the specification of relative requirements for a group of components, where a condition can be defined to link a property value to another property that belongs to the same or a different component. Additionally, the quantification and communication of the information vagueness using multiple visualization techniques can support making informed decision. In various scenarios, the properties of specific components are dependent on other components' properties, such as the position and distribution of columns. Additionally, visualization is essential for representing and simplifying the meaning of information.

## Acknowledgements

We gratefully acknowledge the support of the German Research Foundation (DFG) for funding the project under grant FOR 2363. We thank Ferdinand Tausendpfund GmbH for providing their office building as a sample project.

# Chapter 4

# Vagueness visualization in building models across different design stages

Previously published as: Abualdenien, J.; Borrmann, A.: *Vagueness visualization in building models across different design stages*, Advanced Engineering Informatics 45, pp. 101-1018, 2020, DOI: 10.1016/j.aei.2020.101107

## abstract

The iterative and developing nature of designing a building involves the specification and handling of vague, imprecise, and incomplete information. A crucial factor for mitigating the impact of these uncertainties on the decision-making process is to effectively quantify and communicate them among the project stakeholders. The interactive visualization of 3D building models provides great support for evaluating building designs. However, the currently available visualization methods of the available authoring tools do not incorporate the potential uncertainties associated with the geometric and semantic information of building elements. Currently, building models appear precise and certain, even in the early design stages, which can lead to false assumptions and model evaluations, affecting the decisions made throughout the design stages. Hence, this paper presents a set of visualization approaches, including intrinsic, extrinsic, animation, and walkthroughs, that have been developed to present the uncertainties associated with the building elements' information. The efficiency of the approaches developed in this study was evaluated through an online survey and interviews. More specifically, the approaches were compared in terms of intuitiveness, applicability, and acceptance. The evaluation results positively indicated the participants' ability to understand the amount and impact of the uncertainties on the design by using the developed approaches.

## 4.1 Introduction

The comprehensive exchange of information is a key factor in supporting the design decisions involved in a construction project. The process of designing and constructing a building involves multidisciplinary domain experts, including architects as well as structural, mechanical, and fire safety engineers, collaborating to develop a holistic solution. Each of the experts contributes with specialized domain knowledge to fulfill various and sometimes contradictory requirements and objectives while fulfilling the boundary conditions, including budget, environmental impact, and structure. This process involves a set of interrelated activities that result in increasing the design solution knowledge (or reducing the uncertainty).

The design process has an iterative nature, in which the attention of domain experts oscillates between understanding the problem and developing a solution. This iterative nature is essential and beneficial to the developed design. However, as the design process is multidisciplinary, coordination and communication throughout the design stages are crucial to avoid a substantial amount of unnecessary rework (resulting from false assumptions, misunderstandings, and incomplete information) (Knotten et al., 2015; Love & Edwards, 2004; Tribelsky & Sacks, 2010; Wesz et al., 2018). This kind of rework is a significant cause of problems with time and schedule overruns as well as quality deviations and added expenditure (Knotten et al., 2015; Love & Edwards, 2004; Tribelsky & Sacks, 2010). In this regard, multiple researchers monitored and analyzed the information flow in the design of numerous projects and found a direct relationship between the quality and completeness of exchanged information and the effectiveness of design documents (Joseph Garcia & Mollaoglu, 2020; Knotten et al., 2015; Tribelsky & Sacks, 2010).

Building Information Modeling (BIM) provides a suitable foundation for storing and sharing various kinds of information during the course of a building life cycle (Borrmann et al., 2018a). A well-managed BIM process relies on communicating which information needs to be included in the building model as well as assigning different responsibilities for each project participant in each design stage. This kind of coordination facilitates a seamless integration of the different partial models.

Although the early design stages (conceptual and preliminary stages) are characterized by high uncertainty due to the lack of information and knowledge (Steinmann, 1997), the decisions made during those stages significantly influence the costs and success of the project (Howell, 2016; Leite et al., 2011). In the early stages, the efforts and costs required to make changes in a building model are lower than in the subsequent stages (Kolltveit & Grønhaug, 2004). However, the lack of information affects the decision-making process and outcomes; the uncertainty of how the design may evolve is high, as many decisions have not yet been made (Knotten et al., 2015). In this paper, the term *uncertainty* is used as an umbrella term to encompass many different descriptions, including lack of definition, lack of knowledge, and lack of trust in the knowledge. On the other hand, the term *vagueness* is related to a specific state of a specific object, and it refers to having imprecise information (Hawer et al., 2018).

As discussed above, a design solution is gradually refined and detailed as the design emerges. Accordingly, the quantity and quality of the available information increases as the design becomes more mature. The Level of Development (LOD) concept describes the progressive refinement of the geometric and semantic information by providing definitions and illustrations of BIM elements at different stages of their development (BIMForum, 2019; Hooper, 2015).

To provide a foundation for managing multiple LODs of BIM models, the authors have developed a multi-LOD meta-model (Abualdenien & Borrmann, 2019), which facilitates a formal specification of the LOD definitions, including the explicit specification of the vagueness associated with the building information. Accordingly, the individual properties are assigned to different kinds of vagueness, including a range of values and a distribution function or an abstract classification rather than a fixed value.

In the Architecture, Engineering, and Construction (AEC) industry, visualization is an essential component of the established workflows and exchange scenarios, including communicating design intent, checking the integrity of partial designs, and evaluating design variants (Bouchlaghem et al., 2005). The interactive visualization of 3D building models provides great support for many tasks related to building design and engineering. At the same time, understanding what is precise and complete and accounting for design uncertainty is critical to effectively reason about the visualized building information. However, the existing BIM authoring tools lack of methods for depicting vagueness simultaneously with building models and interacting with those depictions in an understandable way. The current visualization would wrongly suggest that the design is more elaborate than it actually is, which can lead to false assumptions and model evaluations, affecting the decisions made throughout the design stages (Bouchlaghem et al., 2005; Kraft & Nagl, 2007). Various researchers emphasized that people rely on cognitive biases when making decisions under uncertainty (Hullman et al., 2018; A. M. MacEachren et al., 2012). Uncertainty visualization provides high communicative efficiency by means of graphical representation, offering an easier-to-search representation that simplifies recognition and inference (Smith Mason et al., 2017).

This paper addresses the problem of effectively visualizing the vagueness (or inversely, the reliability) associated with the geometric and semantic building information. By conveying the possibility that a position, a geometric dimension or a property value is not fixed and may vary, showing the impact on surrounding elements and spaces, uncertainty visualization enables domain experts to make informed decisions.

To the best of the authors' knowledge, there has been little research that attempts to visually communicate the vagueness of the building information in the early design stages. The contributions of this study are twofold. First, the development of multiple visualization approaches that are suitable for expressing the information vagueness in building models, including intrinsic, extrinsic, animation, and walkthroughs. Second, the evaluation of the approaches' effectiveness from three main aspects: (1) intuitiveness in expressing the information vagueness, (2) applicability on different scales (from model overview to zone/room view), and (3) users' acceptance in terms of using the visualization approaches in their practical work.

This paper is organized as follows: Section 2 describes the methodology applied for this research. Section 3 discusses the background and related work, and Section 4 provides an overview of the previously published multi-LOD meta-model. Section 5 forms the core of the paper as it discusses and demonstrates the developed visualization approaches, and Section 6 presents the results of evaluating the visualization approaches in terms of intuitiveness, applicability, and acceptance by conducting an online survey and interviews. Finally, Section 7 summarizes our progress hitherto and presents an outlook for future research.

## 4.2 Research method

This research aims to explore approaches that seek to improve the communication and collaboration among the different disciplines participating in a construction project, especially at the early design stages where architects and engineers deal with partial and uncertain information.

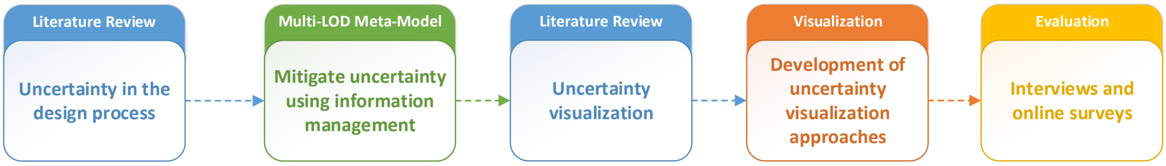


Figure 4.1: The research method used during this research.

As illustrated in Figure 5.13, the first step to achieve this goal is to identify the possible sources of uncertainty through a comprehensive literature review that is focused on understanding the uncertainty during the design process. The literature review took into account the impact of the owner's requirements, the reliability of the design decisions, the conventional approaches in developing design variants, as well as the required interaction among the project participants to make informed decisions based on model analyses and evaluations.

Through the literature review, we found that the current LOD definitions are informal (graphical illustrations and textual definitions), which leads to diverse interpretations (BIMForum, 2019; Gigante-Barrera et al., 2018; Hooper, 2015). Additionally, a major reason why buildings rarely perform as predicted is that practitioners quantify uncertainties in the building model using information from literature, experience, or default values (a well-reported gap exists between the predicted and actual building performance) (De Wilde, 2014; Menezes et al., 2012; Zanni et al., 2019).

To fill this gap and provide a framework for formally managing the information requirements and incorporating the potential vagueness throughout the design stages, the authors have previously developed a multi-LOD meta-model (Abualdenien & Borrmann, 2019), which makes it possible to assign a vagueness definition to each of the individual properties. While evaluating the meta-model, we identified the users' need to navigate through the 3D building model and view the properties of the individual elements, because the current visualization depicts building models as precise and reliable even in the early design stages (Bouchlaghem et al., 2005; Kraft & Nagl, 2007). Additionally, as each building element can have numerous properties, in some cases even with the inclusion of the information vagueness for each property, domain experts can struggle to understand the impact of vagueness on the overall design (Hullman, 2019; Hullman et al., 2018; A. M. MacEachren et al., 2012; Webster, 2003).

Therefore, aggregating and visually conveying the overall vagueness can assist in effectively communicating the potential uncertainties and efficiently managing the design interdependencies. This paper presents a set of visualization approaches that were developed based on reviewing state-of-the-art visualization approaches from various domains. Finally, a survey and interviews were conducted to evaluate the intuitiveness, applicability, and acceptance of the developed approaches.

## 4.3 Background & related work

### 4.3.1 Drawing conventions and scales

Common graphic conventions are incorporated to describe a drawing's layout without the need to include additional explanatory text. In this context, the design reliability is represented by varying the thickness of the lines; a thicker line suggests more permanence while the thinner line suggests a more temporary quality (Farrelly, 2008).

Additionally, conventional construction planning relies heavily on the use of different drawing scales for representing geometric information on a suitable level of detail and degree of preciseness (Farrelly, 2008). The produced drawings evolve from sketches depicting the rough shape of the building and the floor plans to detailed workshop drawings presenting the precise design of individual components, connection points, etc. Accordingly, a drawing's scale directly implies the degree of abstraction, vagueness, and maturity of the design information conveyed, and typically, specific scales are requested in specific design stages. As the concept of scale cannot be applied for digital building models, an analogue concept must be found.

### 4.3.2 Level of Development (LOD)

Several countries worldwide are promoting the research and development of BIM-based methodologies to increase the efficiency of the design, construction and operation of built facilities. As construction projects involve a large number of different parties, a fundamental pillar for integrating BIM is specifying the building elements' maturity at a particular stage.

This is crucial for the overall collaboration among the project participants, because this specification acts as an agreement on *what* information should be available at what time (*when*). Based on that information, it can be decided what the model can be used for *(purpose)*, which makes it possible to decide on what model deliverables are expected from the actors involved (*who*) (Beetz et al., 2018). The exchange of BIM data within the AEC industry must be prescribed through legal agreements where the information for each specific model is specified, meaning that a common legal framework for organizing BIM data is required (Sacks et al., 2018).

As a response to the need of having a consensus about what information should exist during the development of building elements, various guidelines were published to deliver a standard which practitioners can use as a basis for a common language in their projects. The first initiative involved introducing the *Level of Detail (LoD)* (VicoSoftware, 2005). The LoD concept has been adopted and refined by the American Institute of Architects (AIA) to become the *Level of Development (LOD)*, referring to the completeness reliability of the building elements information (AIA, 2013b). Although at that time, it was new in the AEC industry, the *Level of Detail* concept is an old topic that existed in computer graphics. It is used to bridge complexity and performance by regulating the amount of detail used to represent the virtual world (Luebke, Reddy, Cohen, et al., 2003). In computer graphics, the LoD concept is mainly concerned with the geometrical detailing, whereas in the AEC industry, the LOD represents the availability and reliability of the geometric and semantic information.

The AIA introduced a definition of the *Level of Development (LOD)* that comprises five levels, starting from LOD 100 and reaching LOD 500. The BIMForum working group developed LOD 350 and published the *Level of Development Specification* based on the AIA definitions (BIMForum, 2019).

The BIMForum specification is updated annually to provide a common understanding of the expected information at every LOD. The first level, LOD 100 (conceptual model), is limited to a generic representation of the building element, meaning no shape information or geometric representation. The second level, LOD 200 (approximate geometry), consists of generic elements as placeholders with approximate geometric and semantic information. At LOD 300 (precise geometry), all the elements are modeled with their quantity, size, shape, location, and orientation. Next, to enable the detailed coordination between the different disciplines, such as clash detection and avoidance, LOD 350 (construction documentation) is introduced, which includes the interfaces between all the building systems. Reaching LOD 400, the model incorporates additional information about detailing, fabrication, assembly, and installation. Lastly, at LOD 500 (as built), the model elements are a field-verified representation in terms of size, shape, location, quantity, and orientation.

In this paper, the abbreviation *LOD* represents the *Level of Development* that comprises both the *Level of Geometry (LOG)* and *Level of Information (LOI)*. The *LOG* and *LOI* abbreviations are used in the next sections for describing the total vagueness associated with the geometric and semantic information.

### 4.3.3 Information uncertainty

The assumptions made due to the lack of information or knowledge throughout the design stages is a primary cause of information uncertainty (Nilsen & Aven, 2003). The presence of uncertainty influences the produced designs and their performance, impacting the decisions made (Raskin & Taylor, 2014). Typically, exchanging building information between the project participants involves communicating the model's content (BIM model) and additional information describing its reliability (e.g. LOD of building elements). The LOD concept is capable of specifying which information is defined at a particular stage. However, it does not provide the ability to specify additional information that is not certain yet (imprecise or vague) to support the decision-making processes, preparing for the next stage.

Information uncertainty is complex, multidimensional, and has many interpretations. The terms uncertainty, fuzziness, and vagueness are used in various domains and application contexts (Raskin & Taylor, 2014); most commonly, uncertainty is an umbrella term that describes a lack of knowledge or information, causing the occurrence of an uncertain future state (Hawer et al., 2018). A fundamental definition of the term *uncertainty* encompasses multiple concepts, liability to chance or accident, lack of knowledge, lack of information, or lack of trust in knowledge (Murray et al., 1961; Wynn et al., 2011). On the other hand, vagueness is related to a specific state of a specific object, and it refers to having imprecise or inaccurate information (Hawer et al., 2018; Klir, 1987). Fisher, 1999 described uncertainty at a conceptual level as a vague or ambiguous object definition, which refers to the correct use of information (Fisher, 1999). In the context of Computer-Aided Design (CAD) modeling, Steinmann, 1997 described fuzziness (a synonym of vagueness) as the distance from the complete and exact description (Steinmann, 1997). In this paper, we follow the uncertainty definition provided by Longley et al., 2005:

“a measure of the user's understanding of the difference between the contents of a dataset and the real phenomena that the dataset are believed to represent” (Longley et al., 2005).

Based on the authors' experience and the knowledge gained from the literature review, the design process uncertainty is categorized as follows:

- *Requirements uncertainty*: The main intentions of the building design, including its usage, environmental impact, and cost, guide the decisions being made. Understanding the client's requirements and decisions is important for an efficient design process

(Sujan et al., 2019). Kometa et al., 1996 explain the client's influence on the successful execution of construction projects (Kometa et al., 1996). Additionally, the source of requirements uncertainty can be regulations and other boundary conditions.

- *Design uncertainty*: Significant decisions in construction projects are reliant on heuristic processes where assumptions are developed from past experience (Kerzner & Kerzner, 2017). Typically, the process involves choosing among design alternatives and variants while fulfilling the project goals and requirements. This kind of uncertainty has a wide range of combinations at the early design stages and becomes narrower as more decisions are made in the subsequent stages. Design uncertainty and decisions have an impact on the information flow and latency (Sujan et al., 2019).

- *Interaction uncertainty*: Design decisions are built on the continuous feedback of information among the participating domain experts. The architect, as a leading discipline for the designing process, evaluates the various requirements, including functional, operational, and architectural requirements, to make design decisions. Architects are usually concerned with *what the building is*, rather than *how the building performs* (Rezaee et al., 2015). Therefore, with the presence of requirements and design uncertainties, the interaction among the project participants is necessary to agree on the model content and incorporate the building performance in making subjective estimations and decisions.

- *Performance uncertainty*: Performing model analysis utilizes the design information as an input. Accordingly, the design and requirements uncertainties, such as material properties, a scenario of use or other boundary conditions, propagate into the analysis results, producing a range or a set of outcomes. This kind of uncertain results inform the architect decisions with regard to developing an optimized solution, fulfilling the project's requirements and boundary conditions.

Figure 4.2 illustrates the different kinds of uncertainty and their dependencies. The project's requirements and regulations constrain the resultant design and its performance. Developing the design further requires the involvement of the project participants, since changes to the design impact the building performance. Accordingly, the design is collaboratively developed and evaluated based on the analysis results.

Making design decisions under uncertainty is driven by increasing the confidence that choosing $variant_x$ or $value_a$ will result in a better building solution. In this paper, uncertainty represents the unknown variables affecting design variants and their fulfillment of the project's requirements and objectives. Accordingly, defining these variables can lead to fundamental changes in the proposed design, such as changing the overall building shape, increasing its height to add a new storey, or changing the internal spatial structure. Vagueness is related to the reliability of the building elements' attributes and their refinement throughout the LODs,

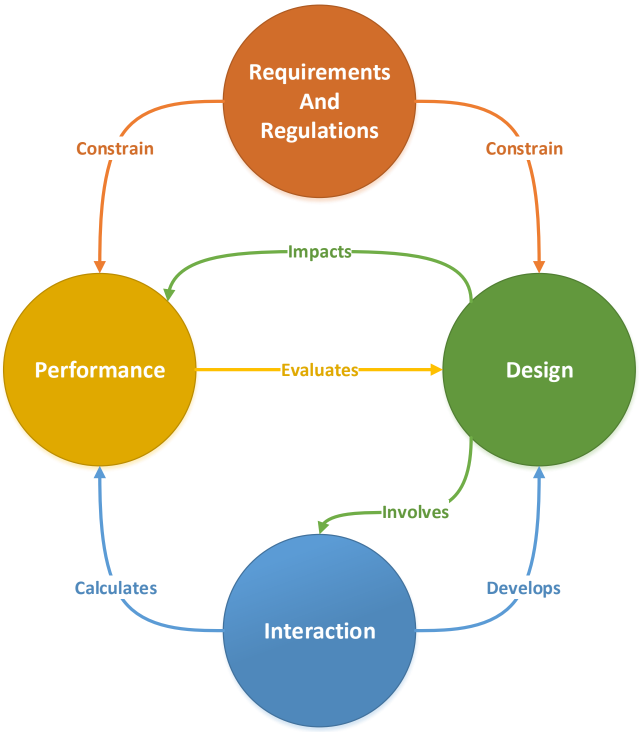


Figure 4.2: Classification and dependencies of information uncertainty during the design process

for example, exact position of the load-bearing components and the percentage of the external walls' openings.

### 4.3.4 Uncertainty visualization

In the process of designing a building, each of the disciplines involved understands and evaluates the proposed design from a different perspective; for example, the architect is concerned with the building's footprint, facade, and interior layout, energy engineers look at the building from the perspective of heat loss and gain, and structural engineers are interested in the performance of the structural system. Hence, visually representing information uncertainty encourages using the domain experts' knowledge, which assists with carrying out tasks more effectively (Card, 1999; Munzner, 2014).

Conveying the quantity of uncertainty in the information is crucial for making rational conclusions (Deitrick, 2007; Griethe & Schumann, 2005). This particularly applies to the architectural design and engineering of buildings. Visual communication of information has advantages over verbal description of it, as humans process visual information with high-efficiency (Smith Mason et al., 2017), which can improve the estimates made (Greis et al., 2018). Multiple researchers from different domains, including geospatial information (A. M. MacEachren et al., 2005), navigation systems (Andre & Cutler, 1998), and architecture (Griethe & Schumann, 2005; Houde et al., 2015), have suggested and applied a variety of techniques for visually representing uncertainty. The most common attempts at categorizing uncertainty visualization are:

- Static vs. dynamic approaches (Davis & Keller, 1997): This categorization distinguishes animation approaches from the others. In the same context, numerous researchers have investigated interactive approaches in uncertainty visualization, including animation type, duration, and rate (DiBiase et al., 1992; Fisher, 1993; Hullman et al., 2015).
- N. Gershon, 1998 proposed two general categories: (1) intrinsic, changes the graphical variables of an object, such as color, transparency, texture, or shape, and (2) extrinsic, involves including additional graphical objects, like text, glyphs, or overlay, to describe the status of an object while leaving the original component unchanged (N. Gershon, 1998).

Several researchers have emphasized the effectiveness of visually depicting uncertainty using visual variables, including intensity, value, lightness, saturation, and opacity (Drecki, 2002; Hengl, 2003; Lines, 2018). Visual variables were first introduced by Bertain (Roth, 2009) as seven *Retinal Variables*, which were subsequently extended by Morrison and MacEachren (A. MacEachren, 1992; Morrison, 1974), rendering a total of 12 variables: (1) location, (2) size, (3) color hue, (4) color value, (5) grain, (6) orientation, and (7) shape, (8) color saturation, (9) arrangement, (10) clarity (fuzziness), (11) resolution, and (12) transparency.

These visual variables received wide acceptance in the community; for example, Hengl, 2003 manipulated saturation and color value to display uncertain data in a more white or pale representation (Hengl, 2003). A. MacEachren, 1992 proposed that data with less certain information should use a correspondingly less saturated color, thereby making their color hue uncertain (A. MacEachren, 1992). Drecki, 2002 proposed representing an uncertain object with transparency, as it is not real, while certain objects are visualized in a relatively opaque representation (Drecki, 2002). R. Brown, 2004 argued that the perception uncertainty using color variables alone is not high enough. Therefore, he suggested including blurring effects for depicting uncertainty (R. Brown, 2004).

In the same context, A. MacEachren, 1992 considered that texture grain is the most appropriate approach to depict whether information is *certain enough* or *not certain enough* (A. MacEachren, 1992). Davis and Keller, 1997 suggested that color hue, color value, and texture are potentially the best choices for representing uncertain information using static approaches (Davis & Keller, 1997). Additionally, Schulz et al., 2018 used transparency, waveform, and frequency to provide a qualitative analysis of uncertainty (Schulz et al., 2018).

From a different point of view, Pang, 2001 suggested adding different types of glyphs to describe uncertainty (Pang, 2001). To include additional information, Finger and Bisantz, 2002 examined the use of degraded icons combined with a numerical probability estimate (Finger & Bisantz, 2002). To support the design for flood management, Ribicic et al., 2012 used error bars and range symbols over city maps for communicating the uncertainty (Ribicic et al., 2012). Although the extrinsic approaches simplify quantifying the amount of uncertainty, Cliburn et al., 2002 cautioned that extrinsic visualization could be confusing or overwhelming (Cliburn et al., 2002).

The presented literature discusses visualizing uncertainty in diverse domains. Developing an uncertainty visualization approach for the AEC industry is a challenging task that requires understanding how the individual domain experts perceive building information. This is crucial for understanding how the knowledge of the uncertainties would influence the decisions taken.

## 4.4 Multi-LOD meta-model

In practice, it is necessary to explicitly specify which information is reliable and estimate the accuracy of the unreliable information at a specific LOD; an LOD is depicted as a milestone for making design decisions. Consequently, precisely defining the LOD requirements while incorporating their uncertainty improves the quality of the collaborative process among the disciplines.

The management of information on multiple LODs requires both representing the building elements on different LODs as well as providing the ability to specify the required information on each LOD in a formal way. The multi-LOD meta-model fulfills these requirements by supporting the following activities (Abualdenien & Borrmann, 2019):

- Formal specification of the overall information requirements at a particular design stage.
- Formal specification of the individual elements' LOD definitions.
- Formal incorporation of the potential vagueness.
- Representation of the building models' instances at different design stages.
- Verification of building models consistency across the design stages, i.e. ensuring that the decisions made in one stage are respected in the subsequent stage.

The meta-model introduces two levels: *data-model level*, which defines the component types' requirements for each LOD, and *instance level*, which represents the actual building components and their relationships. In order to ensure the model's flexibility and applicability, its realization is based on the widely adopted data model Industry Foundation Classes (IFC). The IFC model specification is an ISO standard, which is integrated into a variety of software products (Liebich et al., 2013).

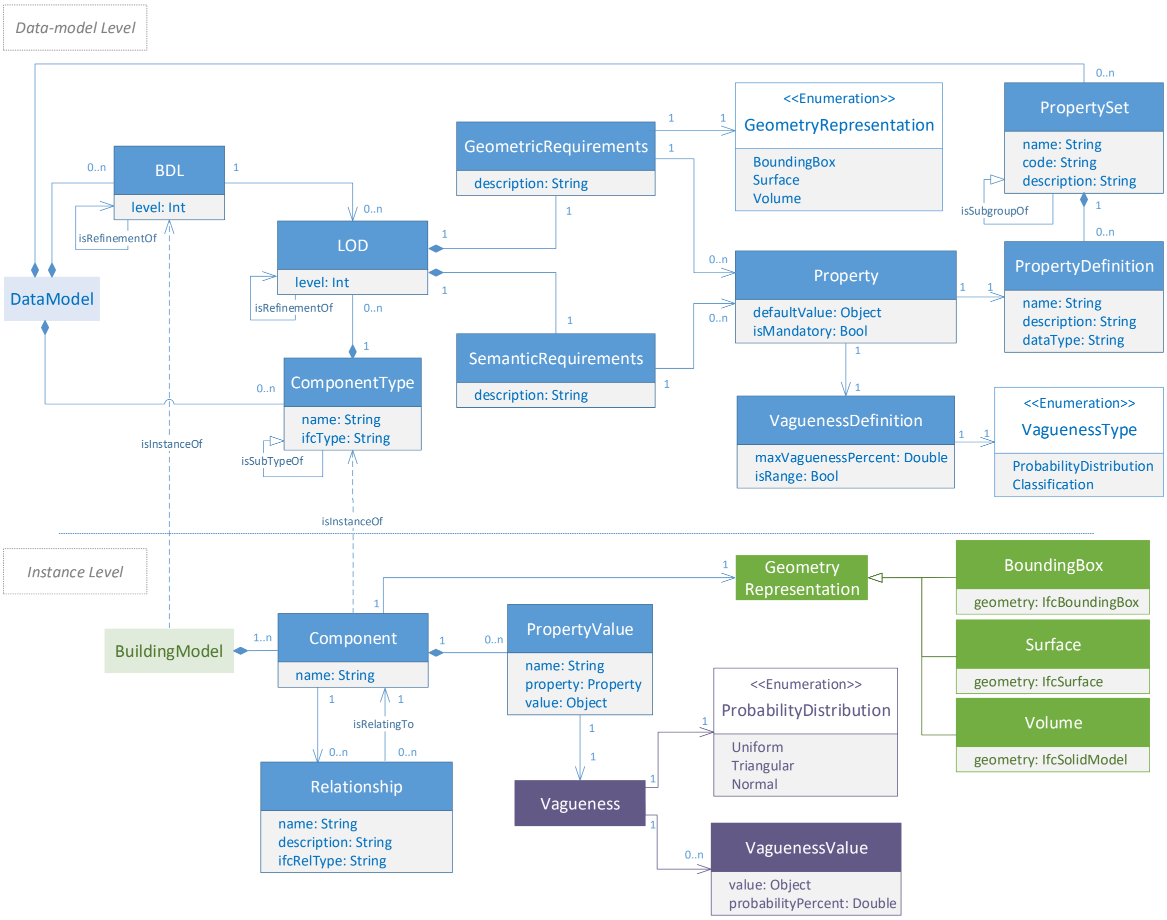


Figure 4.3: Multi-LOD meta-model (UML model)

In more detail, each component type is linked to an IFC type, IfcColumn as an example, and associated with multiple LOD definitions. An LOD definition consists of geometric and semantic requirements, specifying the required geometry representation and properties. The details of each property are determined in addition to the permissible vagueness. In terms of vagueness, a property can be assigned to a vagueness type (classification or probability distribution), a maximum vagueness percentage, and to whether the vagueness values are expected to be as a range.

The vagueness values at the instance level are automatically generated from the vagueness definition specified at the data-model level. For example, in case the vagueness type is a probability distribution, the vagueness percentage is 4%, and the attribute value is 250 cm, the vagueness values are generated to form a range of ±20 cm. Moreover, at the instance level, it is possible to increase the limitation of the range values, such as to be between -5 and +7 cm.

A comprehensive explanation and evaluation of the multi-LOD meta-model approach are available in Abualdenien and Borrmann, 2019 (Abualdenien & Borrmann, 2019). In this paper, the meta-model is extended to support the visualization approaches presented in Section 4.5. More specifically, the property class at the Data-model level is now assigned to a vagueness definition which specifies the nature of the assigned vagueness, i.e. a range or a set of options. Additionally, at the instance level, the vagueness associated with each property value is now represented by a vagueness value and a probability percentage; this way, it is possible to assign a probability percentage to the individual values in case the vagueness is a set of options instead of a range.

Formally specifying a component's LOD definitions, incorporating the potential vagueness, assists in evaluating the performance of different design options before making a design decision. In the same context, engineers and architects work together to determine the realistic design options that fit into the project's requirements. Therefore, expressing the specified vagueness using visualizing would communicate and quantify its effect on the overall building model, and thus facilitate the awareness and inclusion of various use cases.

## 4.5 Proposed visualization approaches

The vagueness specified in the multi-LOD meta-model represents the reliability of attributes at each LOD. In the meta-model, there are two kinds of attributes: geometric ones, including position, shape, and dimensions, as well as semantic ones, such as construction type, material, and cost information. Visualizing the components' vagueness enhances the engineers' awareness of both the reliability of the visualized information, and how a component might evolve in the subsequent LODs. Additionally, vagueness visualization facilitates evaluating the surrounding components' relationships, which improves the quality of the decisions made.

Figure 4.4 illustrates the framework used for vagueness visualization in this paper. It consists of three main steps:

1. *Preparation*, in this step, the actual building model is represented by the multi-LOD meta-model. Thereby, the individual components are mapped to component types, and their properties are assigned to the specified vagueness.

2. *Visualization*, in order to decide which visualization approach is more suitable, the intention and use-case for visualizing the vagueness need to be identified and considered. Analyzing which visualization is more suitable can be done by answering questions such as:

    - Are we interested in acquiring a rough idea about the information reliability?

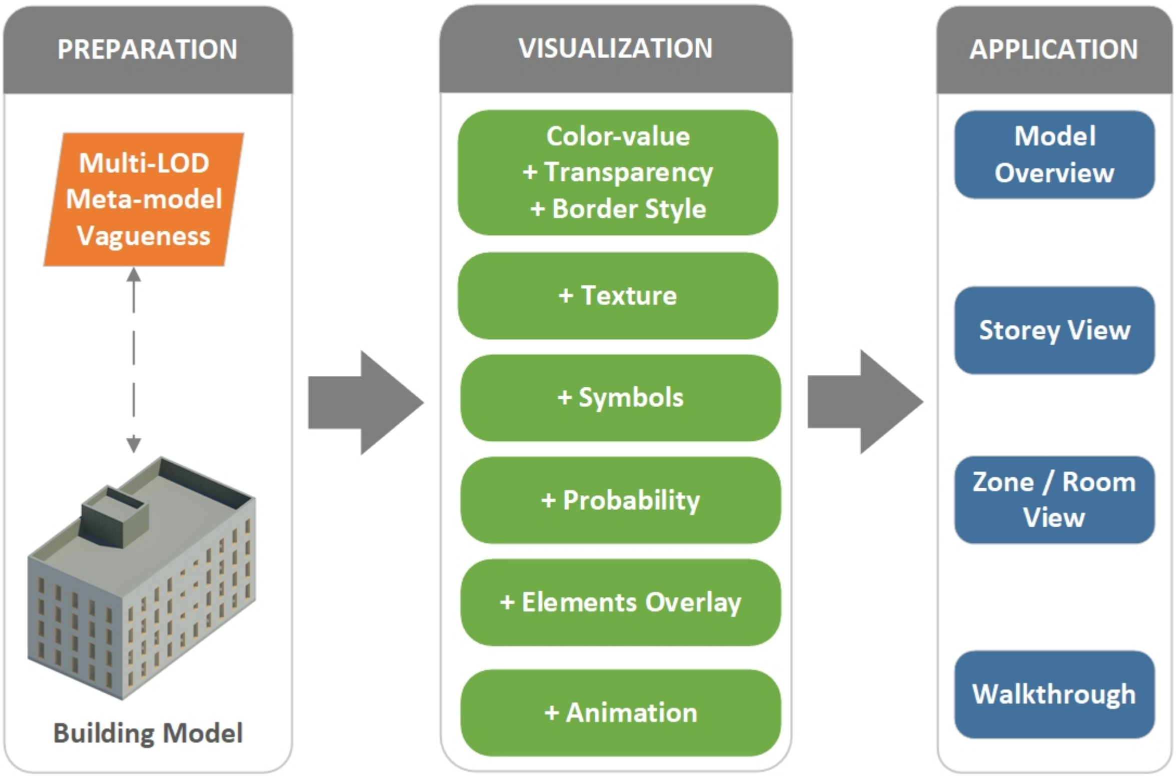


Figure 4.4: Vagueness visualization framework that consists of three main steps: (1) *Preparation*, which combines the actual building information with the vagueness defined in the multi-LOD meta-model, (2) *Visualization*, which focuses on selecting a suitable visualization approach for the intended use case, and (3) *Application*, in which the information vagueness is depicted on different scales. The visualization approaches presented here are discussed in detail in this section and evaluated in Section 4.6 in terms of their user intuitiveness, acceptance, and application view suitability.

  - Are we trying to make spatial or topological design decisions, i.e. designing the space program of the building?
  - Are the components' material layers, structural usage, and thermal properties crucial to the task we want to perform?

3. *Application*, the chosen visualization approach influences which view is more applicable and beneficial for understanding the impact of the information vagueness on the design. In this paper, the developed visualization approaches are applied on different scales, starting from a 3D model overview (the entire building) to the storey view, zone/room view, and finally, the walkthrough. The concept is to use the developed visualizations to highlight the potential vagueness for supporting the possible use cases.

To quantify the vagueness of a particular component (total vagueness), the average of the vagueness assigned to each property is calculated. Equation 4.1 illustrates how the total vagueness is calculated for the geometric properties ($TV_{LOG}$) and the semantics ($TV_{LOI}$) at a particular LOD.

$$TV_{LOG_x/LOI_x}(component) = \frac{1}{n}\sum_{i=1}^{n} PV_{i,LOD_x} \tag{4.1}$$

where:

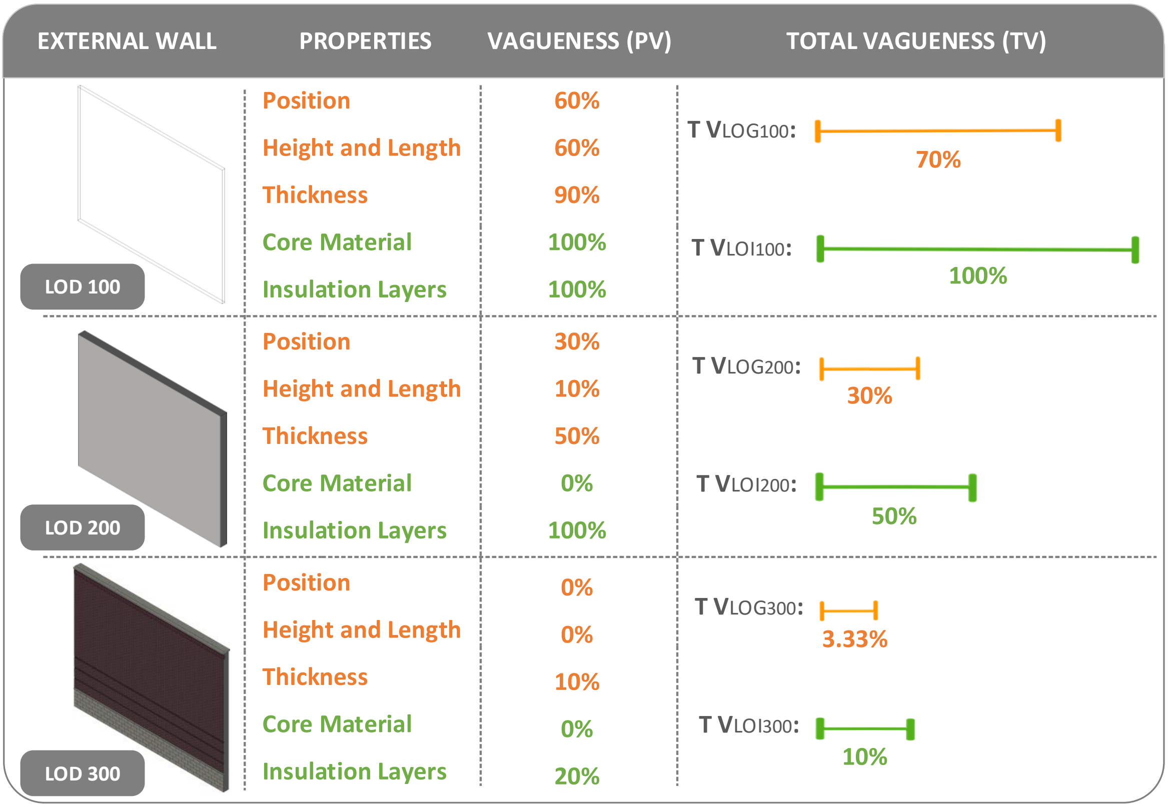
$TV_{LOG_x}$ = total vagueness % of geometric properties at a particular LOD
$TV_{LOI_x}$ = total vagueness % of semantic properties at a particular LOD
$n$ = total number of geometric or semantic properties in all LODs
$PV_{i,LOD_x}$ = vagueness % of a property (at index $i$) at a particular LOD

When calculating the total vagueness of a particular component based on the multi-LOD meta-model definitions, the known properties with a classification vagueness (e.g. when using Omniclass (OmniClass, 2012) and Uniclass (Chapman, 2013) classification systems) are substituted with a percentage that corresponds to the hierarchical depth of the classification system (50% in case of two levels and 33.3% when the classification has three levels). On the other hand, the properties associated with a vagueness of *distribution function* type use the vagueness percentage. Finally, the unknown properties are represented by 100% vagueness.

For example, Figure 4.5 illustrates the process of developing a wall throughout the LODs 100 − 300 with a selected set of properties. Per the BIMForum's specification, at LOD 100, there is no information regarding a wall's material layers, and the position, dimensions, as well as the thickness are still flexible. In this case, based on the authors' estimations, the $TV_{LOG_{100}}$ equals to 70%, as it is calculated by averaging the vagueness of all the geometric properties, and the $TV_{LOI_{100}}$ equals to 100%, since information about the main material and insulation layers is not known at this level (completely unreliable). Next, at LOD 200, the main material is defined and the vagueness of the geometric information is reduced. Similarly, at LOD 300, the position and dimensions become fixed, while the material of the insulation layers and their corresponding thickness are still uncertain.

### 4.5.1 Static intrinsic approaches

The LOD requirements for the component types can vary from one project to another (BIMForum, 2019). Accordingly, in many cases, a component's geometry can be more developed than its semantics. Hence, we propose visualizing the information vagueness for each type separately, using two intrinsic approaches. The first approach aims to express the geometry's vagueness by varying the components' border style in four styles. When the vagueness is high (>50%), it is visualized without a border. Subsequently, when the



| EXTERNAL WALL | PROPERTIES | VAGUENESS (PV) | TOTAL VAGUENESS (TV) |
|---|---|---|---|
| LOD 100 | Position | 60% | $TV_{LOG100}$: 70% |
| | Height and Length | 60% | |
| | Thickness | 90% | |
| | Core Material | 100% | $TV_{LOI100}$: 100% |
| | Insulation Layers | 100% | |
| LOD 200 | Position | 30% | $TV_{LOG200}$: 30% |
| | Height and Length | 10% | |
| | Thickness | 50% | |
| | Core Material | 0% | $TV_{LOI200}$: 50% |
| | Insulation Layers | 100% | |
| LOD 300 | Position | 0% | $TV_{LOG300}$: 3.33% |
| | Height and Length | 0% | |
| | Thickness | 10% | |
| | Core Material | 0% | $TV_{LOI300}$: 10% |
| | Insulation Layers | 20% | |

Figure 4.5: Total vagueness calculation: external wall example with a selected set of properties throughout the LODs 100 – 300. The main idea is that the total geometric and semantic vagueness decreases with incrementing the LOD. The percentages provided are based on the authors' interpretation of the BIMForum's specification and practical experience. These estimated percentages describe the potential change in property values in the subsequent stage.

vagueness is reduced, the border style appears as dotted, dashed, and solid at the end when the vagueness is equal to zero, i.e., the geometry is precise and certain. Similarly, the second approach conveys the semantics vagueness by changing the color value and its transparency in four levels, from light-transparent to dark-opaque.

Figure 4.6 illustrates applying the proposed approaches on a simple storey. At LOD 100, walls represent the overall volume, but information regarding the material or construction type is missing. Additionally, thickness and position are still flexible. Therefore, the wall's geometric and semantics vagueness is more than 50%, i.e., represented by no border and light-transparent fill color. At LOD 200 and 300, walls are depicted with dotted and dashed borders because their position and dimensions become more certain.

Additionally, as the walls' main material is known at LODs 200 and 300, their fill color is darker and less transparent than for LOD 100. As described in the BIMForum's LOD specification (BIMForum, 2019), the walls at LOD 350 have fixed and reliable geometry,

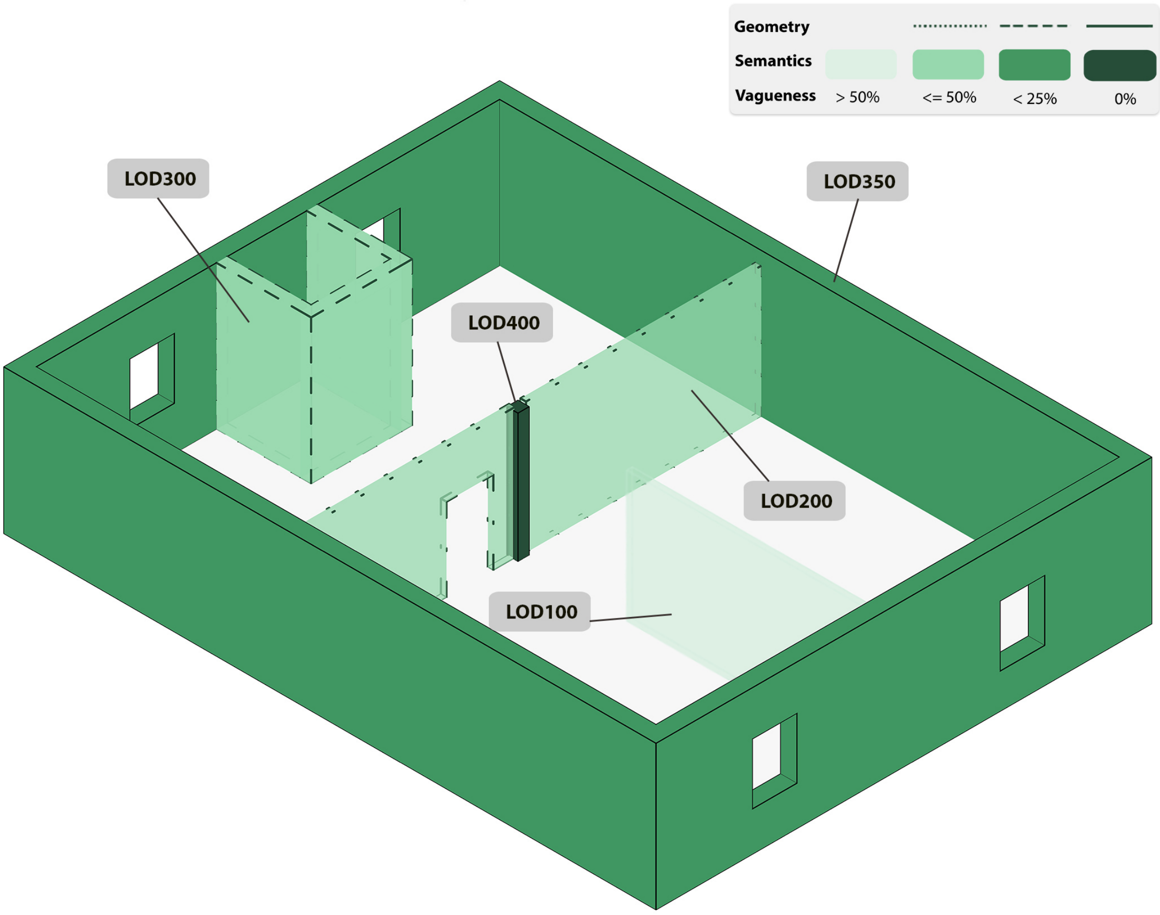


Figure 4.6: Intrinsic approach - Border style, color value, and transparency: visualizing vagueness in four levels, >50%, <=50%, <25%, and 0%. Geometric vagueness is represented by a border style ranging from no border to solid style, and semantic vagueness is represented by varying the color and transparency values in four levels, from light-transparent to dark-opaque. Additionally, to make the concept understandable, the LODs are assigned to the different walls based on the definitions available in the BIMForum's specification (BIMForum, 2019) and in Abualdenien and Borrmann, 2019 (Abualdenien & Borrmann, 2019).

$TV_{LOG_{350}}$ equals to zero. Therefore, a solid border style is used. Finally, at LOD 400, the semantic information also becomes certain, as the case for the column in the middle, where the building element is visualized in a solid border style and a dark fill color.

Expressing the vagueness associated with the components' geometry and semantics using two separate approaches is helpful with regard to a variety of decisions, especially for the geometric information, as it is specifically describing the component's shape and position. However, in some cases, when evaluating the structural system or compliance with fire safety regulations, the vagueness associated with the components' structure, including material layers as well as thermal and structural properties, is more important than the other semantic information.

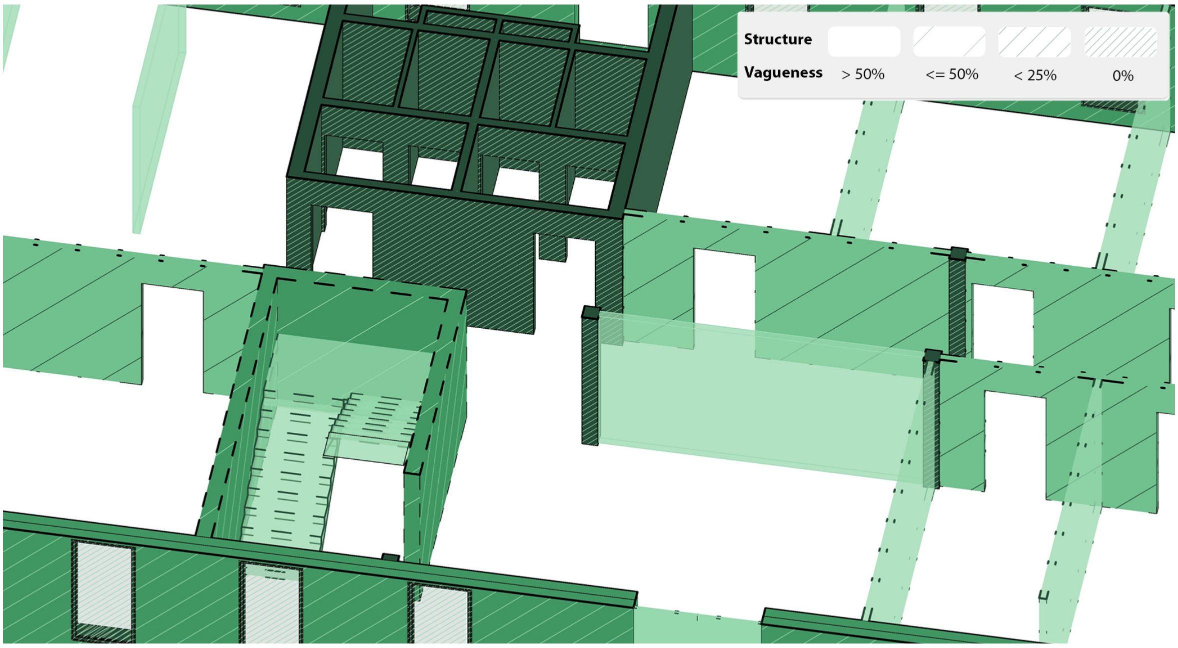


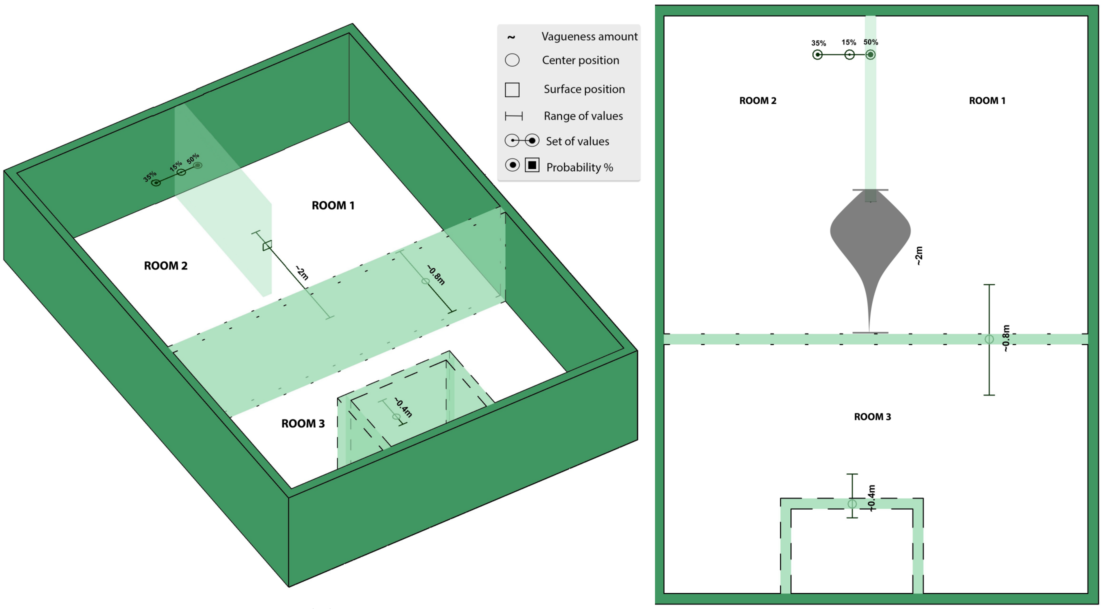
Figure 4.7: Intrinsic approach - Texture grain: visualizing vagueness in four levels, >50%, <=50%, <25%, and equals to 0%. This is an extension to the approach illustrated in Figure 4.6 where it represents the vagueness associated with the elements' structure, including material layers as well as thermal and structural properties. The approach varies the texture grain in four levels, where the texture becomes more condensed when the vagueness is reduced.

In such situations, employing one indicator for the semantic vagueness might not be sufficient to assist the decision-making process, especially because semantics can include additional diverse information, including vendor, brand, cost, etc. Therefore, a more specialized visualization approach that can depict the vagueness of the components' structure would be beneficial when making design decisions or carrying out different simulations. Accordingly, Figure 4.7 shows, an additional indicator representing the elements' structure using four levels of texture grain, starting from no texture when the vagueness is high and then becoming more condensed when the vagueness is reduced.

### 4.5.2 Static extrinsic approaches

The proposed intrinsic approaches in the previous section provide an overview of the vagueness corresponding to the entire building model, showing the amount of vagueness associated with all elements. Usually, when designers detail the building model, they evaluate the individual component's positions and dimensions while considering all the possible cases. Therefore, in this section, we propose applying two extrinsic approaches to represent the impact of vagueness on the possible positions and dimensions.

The first approach includes adding the combination of *property symbols*, *bars*, and *text* with a tilde (∼) symbol (showing the possible values as an approximation). The property symbols convey two position types, one for the element's *center position* (a circle) and another for the *surface position* (a rectangle). As discussed in Section 4.4, the vagueness assigned to a property at the multi-LOD meta-model can be in the form of a continuous range assigned to a probability distribution function bounded by an upper and lower limit, or a set of options, in which each is assigned to a probability percentage. If the specified vagueness is *range*, then it is represented by a bounded bar, where the distribution function is depicted over it. Whereas in case of vagueness *options*, each option is shown as a circle or rectangle that is filled according to the specified probability percentage; the more it is filled, the higher the probability. The selection of the symbols is based on an extensive evaluation (Beck, 2019).

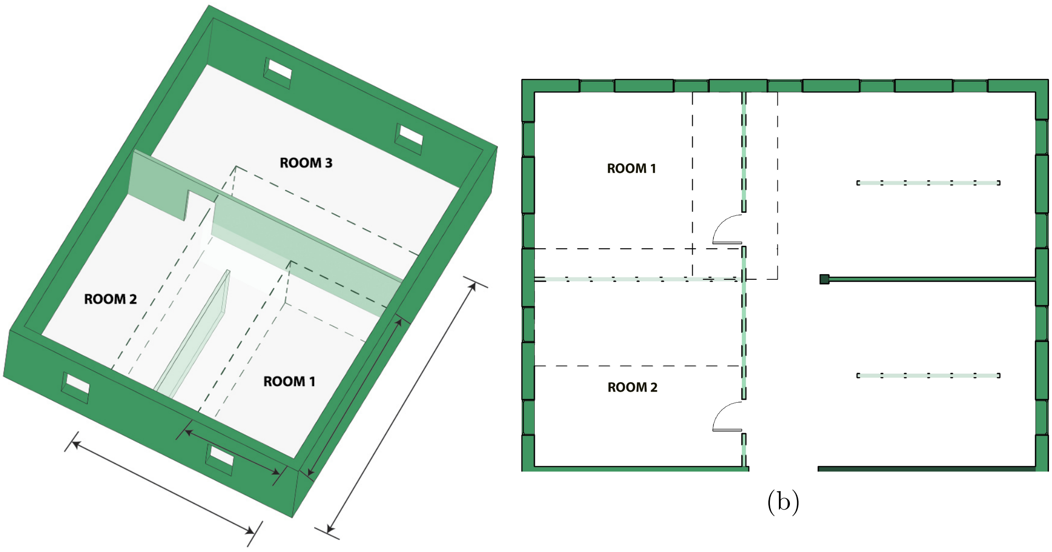


(a)

(b)

Figure 4.8: Static extrinsic approach - Symbols: 3D and 2D views of expressing the vagueness associated with the surface and center positions using symbols (rectangle and circle, respectively). In the 3D view, the bars are assigned to a rectangular probability distribution function, and the text with a tilde (∼) symbol shows the possible values as an approximation. Whereas in the 2D view, an example of depicting a different distribution function is presented. In both views, the possible position options of the wall that separates Room 1 and 2 are shown as circles filled according to the specified probability percentage, the more the circle is filled, the higher the probability.

Figure 4.8 demonstrates the approach through 2D and 3D views. The 3D view shows possible position options for the wall separating Rooms 1 and 2 with circles filled according to the specified probability percentage at the multi-LOD meta-model. Additionally, since the possible values of the length and position can be a continuous range, the vagueness of the other elements is depicted using bars, where the vagueness amount is shown as a descriptive text. Here, we can notice the difference between the symbol used for the center position and the surface position (a circle and rectangle, respectively). Additionally, the bars shown in the 3D view are assigned to a rectangular probability distribution function, while the 2D view demonstrates adding a different probability distribution function. Both approaches were evaluated on two reference projects in Section 4.6.



(a)

(b)

Figure 4.9: Static extrinsic approach - Overlay: depicting the possible changes in the interior layout (room sizes and separation) due to the currently defined vagueness. The 3D view, Figure 4.9a, expresses that the size of Room 1 can still be reduced or expanded, this can be due to unspecified room usage (e.g., a kitchen vs. a living room). Additionally, Room 1 can be separated from Room 2. In Figure 4.9b, the 2D view depicts the possible change in areas assigned to Room 1; its size can be expanded further into Room 2, reducing the size of Room 2. Furthermore, the position of the wall containing the room door is still flexible and can move in both directions.

The second extrinsic approach signifies the vagueness by generating an overlay over the original element. In this approach, the main focus is to depict the possible changes in the interior layout (room dimensions), which impact the rooms' usage and their available space. Figure 4.9 illustrates two examples of the proposed approach to communicate the possible room dimensions due to the vagueness of the interior walls. The 3D view illustrated in Figure 4.9a depicts the possible changes in the dimensions of Room 1. Additionally, the vagueness in the inner walls' length influences their function, in this case, from being a room divider to non-room divider, causing Room 1 and Room 2 to be separate. Such a change modifies the spatial structure of the storey, which affects the designed compartments for fire safety regulations, life cycle analysis, and load distribution, where the wall is load-bearing. Figure 4.9b shows a different example. The focus here is on indicating that the area of Room 1 can be increased from two directions, towards Room 2 and the corridor; Room 1 can expand to almost half of Room 2, and the position of the wall containing the room door is still flexible in both directions.

### 4.5.3 Animation as vagueness indicator and 3D walkthroughs perspective

The effectiveness of vagueness visualization approaches is evaluated by measuring the participants' ability to seamlessly perceive and interpret the amount of vagueness in a presented context. In this regard, animation can highlight the differences in the visualization parameters (N. D. Gershon, 1992; Lundström et al., 2007; Masuch & Strothotte, 1998). For instance, Lundström et al., 2007 introduced probabilistic transfer functions that assign probabilities to different materials. The probabilities are visualized through an animation, where each material is shown for a duration that is proportional to its probability (Lundström et al., 2007).

In this paper, animation is utilized to signify and communicate the impact of the possible positions and lengths of elements. For example, the vagueness associated with the interior walls strongly affects the story's layout and the designed functions. Additionally, when visualizing vagueness using animation, it is crucial to take into account the different topological constraints and relationships, for example, respecting the position of the external walls and door openings.

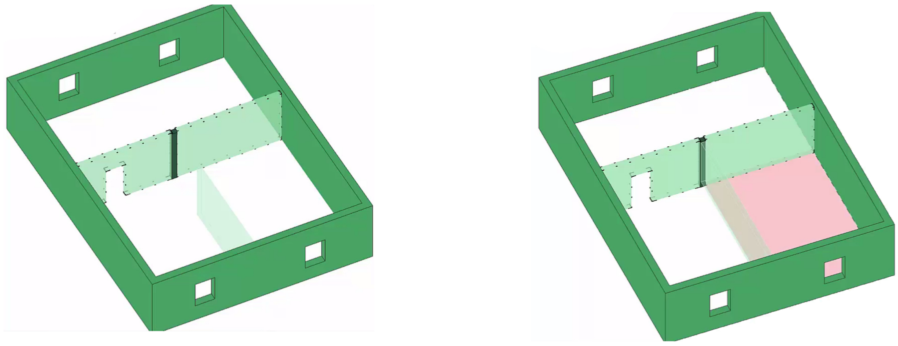

(a) (b)

Figure 4.10: Animation as vagueness indicator: two techniques for quantifying vagueness by animating the building elements. Figure 4.10a utilizes the animation speed to communicate the amount of vagueness; higher speed implies higher vagueness, whereas, Figure 4.10b depicts the impact of the elements' vagueness on the interior layout by highlighting the floor of the changing room. The animation is available online[1].

Figure 4.10 illustrates the proposed animations as a video. Here, the interior walls are animated with a speed corresponding to their defined vagueness. In Figure 4.10a, the position

---

[1] (a) https://youtu.be/sCJEsRlSECo | (b) https://youtu.be/NlK6FailauM

of the wall functioning as a separator changes more quickly than the other walls because it has higher vagueness, whereas Figure 4.10b highlights the impact of changing the storey's topology due to the vagueness assigned to the wall's length, causing the room to be separated and disconnected. Figure 4.12 shows an example of applying the proposed animations on a reference project during the conducted surveys and interviews. First, the interior walls separating the rooms are animated relatively faster and with longer distance than the stairs, since walls are associated with higher vagueness. Then, the possible separations of the offices on the other side of the model are depicted by highlighting the change in the interior layout (more details are provided in Section 4.6).

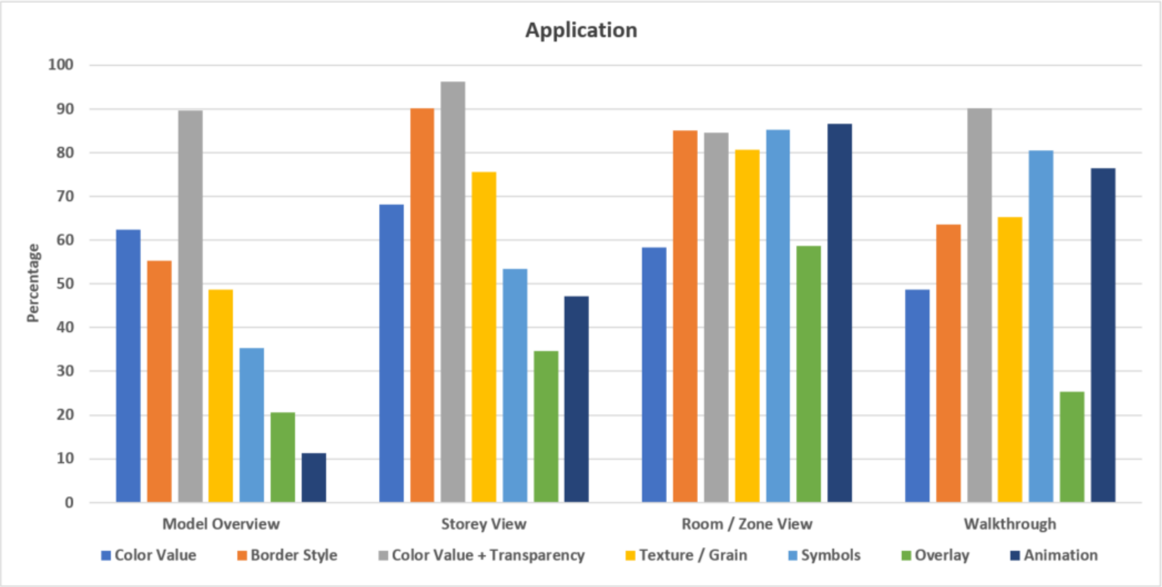


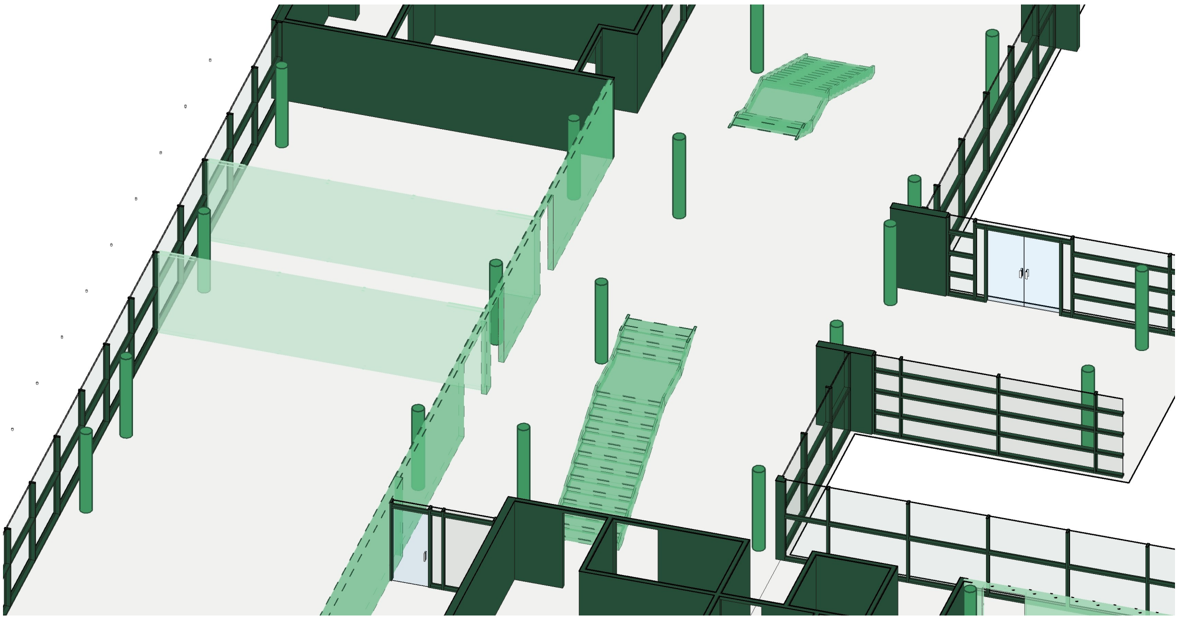
Figure 4.11: Survey results - Application View: participants were asked to choose which visualization approaches are suitable for each application view, including *Model Overview*, *Storey View*, *Room/Zone View*, and *Walkthrough*.

Additionally, Figure 4.13 applies animation using the vagueness bars in a way that uses the animation speed to communicate the probability percentage of each value. The assigned distribution function in Figure 4.13a is rectangular, and thus the animation speed is the same for all values (the wall stays in each position for the same duration). However, as shown in Figure 4.13b, the animation speed increases when the probability gets lower and decreases when the probability is higher, giving the impression that the wall is more likely to be in those positions because it stays in those positions for a longer duration. Based on our evaluations when developing the concepts, using animation can be overwhelming to users, as many aspects might change simultaneously. Therefore, we propose carefully applying animation by confining its application to an individual element and one attribute at a time.

---

[2] https://youtu.be/TyytLIMzHqE

[3] (a) https://youtu.be/PXgc1qO7xas | (b) https://youtu.be/WotYEXyn_Hw



(a)

Figure 4.12: Animation as vagueness indicator - Example: an example of applying the proposed animations on a reference project during the conducted surveys and interviews. The animation was used to indicate the amount of vagueness associated with the interior walls and stairs. Additionally, the change in the interior layout (possible separations of offices) was highlighted. More details are provided in Section 4.6. The animation is available online[2].

Representing the building in 3D facilitates understanding the relationships between objects. Numerous approaches were investigated and evaluated in the AEC industry to improve the project participants' experience (Leicht, 2009). Walkthrough is one of the most common extensions of 3D visualization; it offers a more realistic depiction of the relationships between elements and fosters a better spatial understanding of the proposed design (Leicht, 2009). The experience resulting from this kind of visualization can highlight essential aspects and provoke detailed discussions, which can lead to the discovery of unexpected conflicts and safety issues when collaboratively working with the different domain experts (in a design review meeting, for example) (Y. Liu et al., 2014). Hence, as Figure 4.14 demonstrates, the developed visualization approaches were implemented and evaluated from a walkthrough perspective.

## 4.6 Evaluation

The main focus when evaluating the approaches developed for vagueness visualization is to compare the accuracy of the user's subjective judgment against a ground truth (A. M. MacEachren et al., 2005). The approach used in expressing vagueness has a significant influ-

---

[4] https://youtu.be/x6GsGSbzFSs

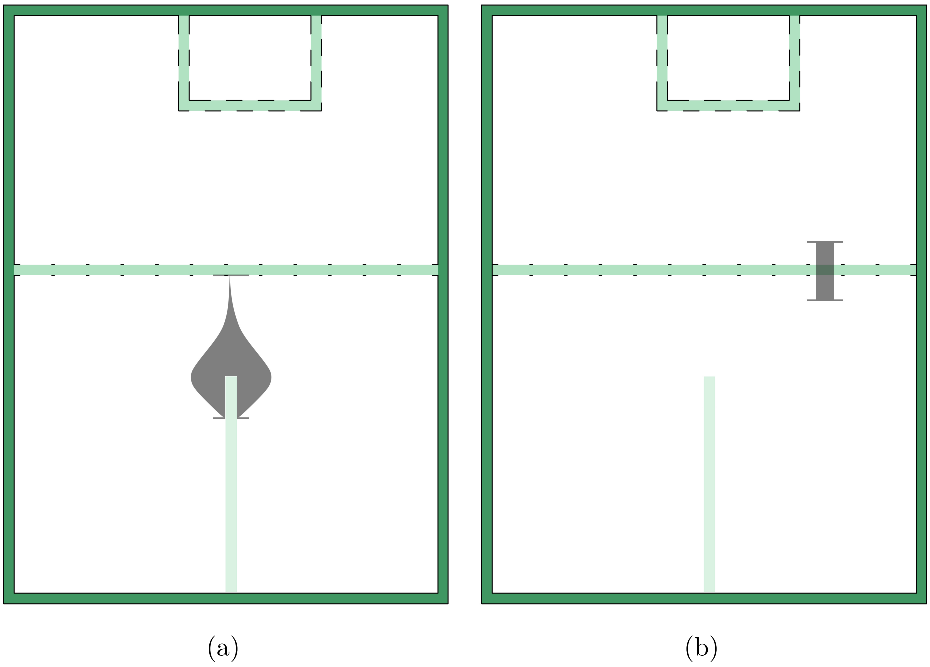


Figure 4.13: Animation as vagueness indicator - Probability: including probability distribution and using animation speed to emphasize on the most probable position, the longer the wall stays in a particular position, the higher the probability. The animation is available online[3].

ence on visualization effectiveness and usefulness (BOUKHELIFA & DUKE, 2009). Performing a user evaluation requires the consideration of multiple aspects, including the user knowledge, visualization type (2D, 3D, or walkthrough), method of depiction (intrinsic, extrinsic, or animation), and the target use case. The evaluation of the approaches presented in this paper took into account accuracy and response time. We conducted an online survey with 60 participants from the industry as well as from research/education and performed interviews with domain experts from three different subcontractors (architecture and engineering offices).

The evaluation utilized the information available from a real project, an office building in Germany (depicted in Figure 4.15), and an Autodesk sample project[5] (illustrated in Figure 4.16).

### 4.6.1 Proof of concept

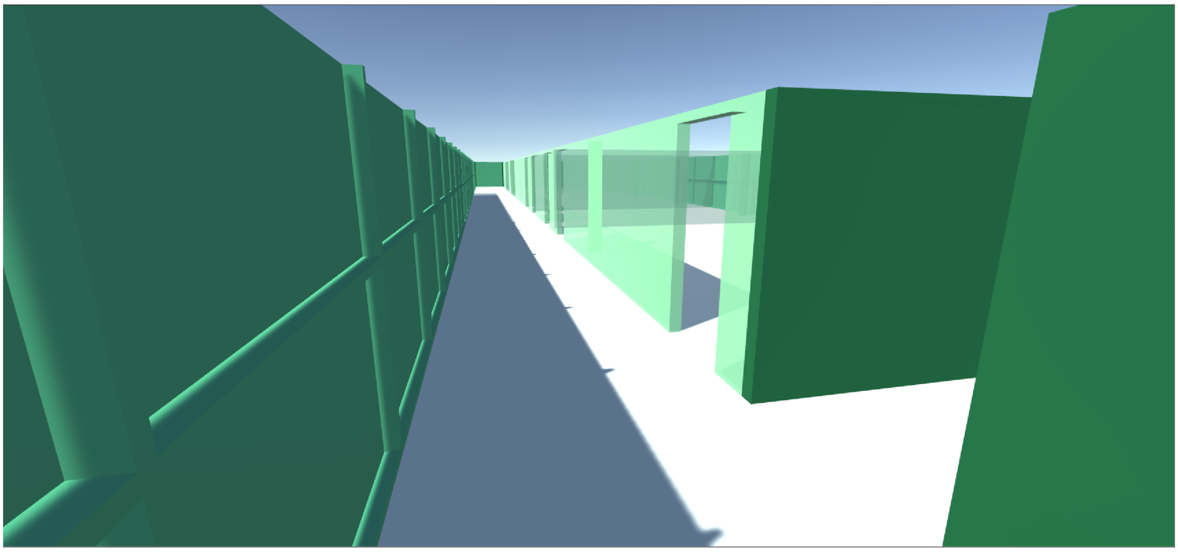
To evaluate the proposed visualization approaches, a proof of concept was implemented as an Autodesk Revit[6] plugin and Unity[7] 3D walkthroughs. While the capabilities of Unity are well known in visualization and animation, it was also feasible to apply different coloring, textures, border styles, as well as symbols, change element dimensions, and change element positions to

---

[5] https://autode.sk/2qLXiVV

[6] https://www.autodesk.com/products/revit/overview

[7] https://unity.com/



(a)

Figure 4.14: Walkthrough perspective: an overview of the implemented use cases. The user can walk through the building model and review the different aspects. The walkthrough is available online[4].

realize the proposed animations using the Revit Application Programming Interface (API). Both prototypes provide interactive interfaces for users to navigate and review the different aspects of the building design.

### 4.6.2 Online survey design

The proposed visualizations and the prototype were evaluated by conducting an online survey. The approaches were presented to the participants gradually to assess the influence of each. First, varying the *Color Value* to represent the geometric and semantic information was evaluated. Next, the other approaches, *Border Style*, *Transparency*, *Texture Grain*, etc. were included step by step.

The survey was designed using a framework called *LimeSurvey*[8], which makes it possible to capture the time participants took to answer each of the questions. The survey aimed at identifying extent to which participants understood each of the proposed visualization approaches and measuring the intuitiveness of each approach. A set of 22 required questions examined the participants' understanding using single and multiple-choice options. The expected answer (100% correct) for each question consists of one or multiple options, where the 100% distributed equally over the number of correct options. Additionally, at the end of the survey, participants ranked the acceptance of each visualization approach on a scale of one to five, with one being strongly disagree and five being strongly agree. Additionally,

---

[8] https://www.limesurvey.org/

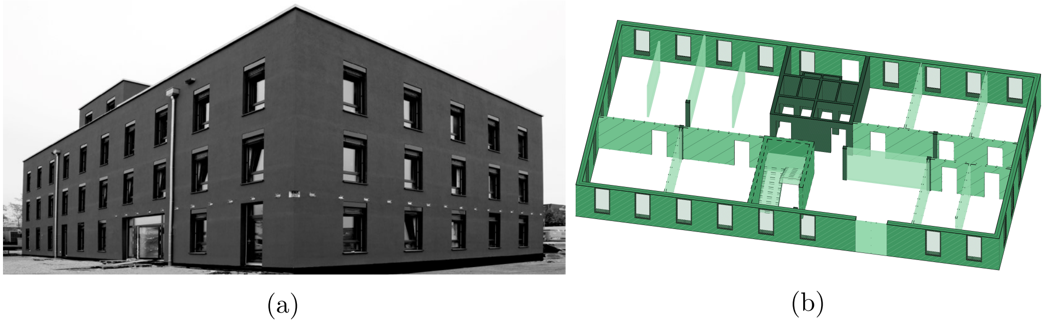

(a) (b)

Figure 4.15: Evaluation of reference project #1: Ferdinand Tausendpfund GmbH & Co. KG office building, in Regensburg, Germany, built in 2017. (a) is a picture of the actual building, and (b) is a snapshot depicting the first storey of the BIM model, including an application of the proposed visualization approaches.

they were asked to choose which visualization approach is more applicable for each of the application views (model overview, storey view, zone/room view, and walkthrough).

The survey began with a descriptive overview of the purpose of the visualizations, and then specific explanations were provided for each question. The answers and response times were automatically collected in a database through a functionality provided by LimeSurvey. An invitation to participate in the survey was sent to multiple subcontractor offices as well as to graduate students (masters and doctorate levels) from diverse but relevant domains of the Technical University of Munich (TUM)[9]. A majority of the students attended lectures in which the motivation for the visualizations was explained. Figure 4.17 presents the list of participants grouped by domain. In total, 60 participants took part in the survey.

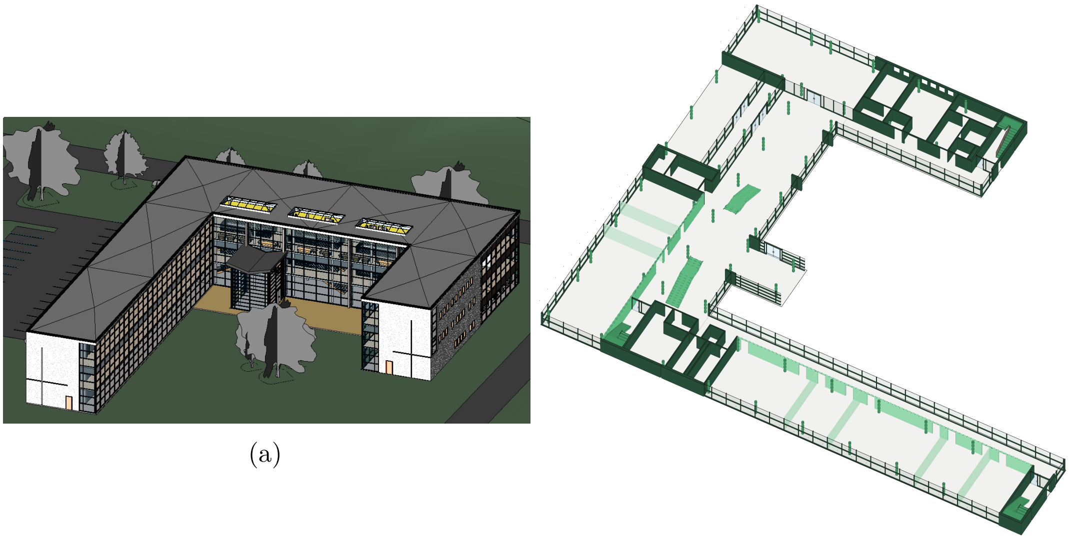
### Online survey: Results

Survey responses were evaluated and ranked in terms of accuracy by taking into account the expected answers and the corresponding response times.

Figure 4.18 presents a comparison of the intuitiveness and response time of the developed approaches. The values shown represent the average and standard deviation for each approach. First, *Color Value* (varies the fill color value from light to dark green) attained an acceptable level of intuitiveness and response time. Then, adding the *Border Style* improved the intuitiveness and reduced the differences among the participants' response times. Including *Transparency* to the fill color value as well as adding *Animation* made a noticeable improvement in the intuitiveness and the response time.

---

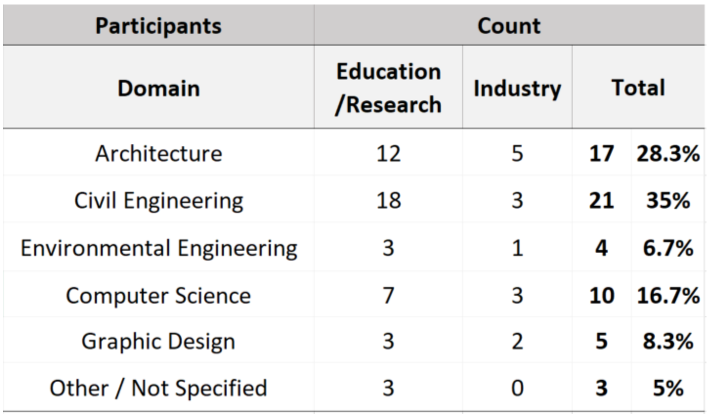
[9] https://www.tum.de/



(a)

(b)

Figure 4.16: Evaluation of reference project #2: Autodesk sample project. (a) is a picture of the actual 3D model, and (b) is a snapshot of the first storey, including an application of the proposed visualization approaches.



| **Participants** | **Count** | | | |
|---|---|---|---|---|
| **Domain** | **Education /Research** | **Industry** | **Total** | |
| Architecture | 12 | 5 | **17** | **28.3%** |
| Civil Engineering | 18 | 3 | **21** | **35%** |
| Environmental Engineering | 3 | 1 | **4** | **6.7%** |
| Computer Science | 7 | 3 | **10** | **16.7%** |
| Graphic Design | 3 | 2 | **5** | **8.3%** |
| Other / Not Specified | 3 | 0 | **3** | **5%** |

Figure 4.17: Online survey: list of participants grouped by domain. Each domain is split into two categories: *Education/Research*, for masters and doctorate students, and *Industry* for the employees working in subcontractor offices.

Although the results of using *Texture Grain* to represent the building elements' structure and *Symbols* to communicate the possible positional values were relatively lower than the others, the results were acceptable. However, the *Overlay* approach as well as adding *Probability* were not ranked as intuitive; intuitiveness was drastically lower in this case than the other approaches and the participants' response time was longer.

As the order of the survey questions started with evaluating the *Color Value* first, followed by adding *Border Style*, *Transparency*, etc., an improvement in the participants' performance is reflected in the results; response times became shorter and more consistent and intuitiveness

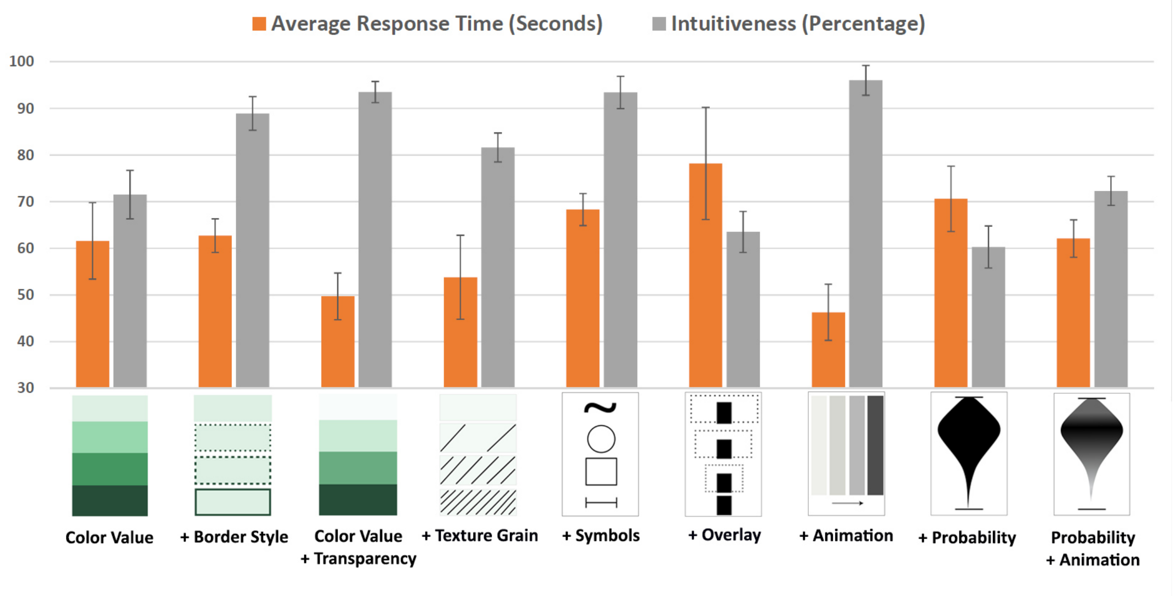


Figure 4.18: Survey results - intuitiveness: the developed approaches were evaluated in terms of intuitiveness, taking into account the expected answers and the corresponding response time. The values shown here represent the average and standard deviation for each approach.

increased. This indicates that the developed approaches entail a learning step for the participants, making the developed approaches easier to understand with time and practice.

In a different set of survey questions, the participants were asked to select which visualization approaches are applicable for each application view, including *Model Overview*, *Storey View*, *Room/Zone View*, and *Walkthrough*. As shown in Figure 4.11, using *Color Value + Transparency* was ranked the highest among the other approaches for communicating the vagueness of the overall building model, storey view, and walkthrough. For the room view, five out of seven approaches yielded equivalent results and received over 80% of the votes. Although the *Color Value* approach is highly similar to *Color Value + Transparency*, it was not ranked as highly acceptable for any of the evaluated views (received a maximum ranking of 68% for storey view), which means that adding transparency and border style assisted in making the approach more understandable and suitable.

Considering a different visualization approach, varying the *Border Style* also received relatively high votes with both views, storey view and room view, in comparison to others (with a rank of 90% and 82%, respectively). The *Texture Grain* approach was ranked more applicable for the storey and room views than the other views (with 76% and 81%, respectively). The *Symbols* and *Animation* approaches attained a similar acceptance pattern; they received high applicability rankings for the small-scale views (room view and walkthrough) and low applicability for the large-scale views (model overview and storey view). Finally, the *Overlay* approach did not perform well in any of the views. The reason can be deduced from the results presented in Figure 4.18, low intuitiveness and long response time.

Finally, the participants were asked to compare the visualization approaches by specifying the degree to which they would accept using the approaches in their practical work. The questions allowed participants to rank each approach on a scale from 1 – 5 (strongly refrain, rather not, neutral, accept, and strongly accept).

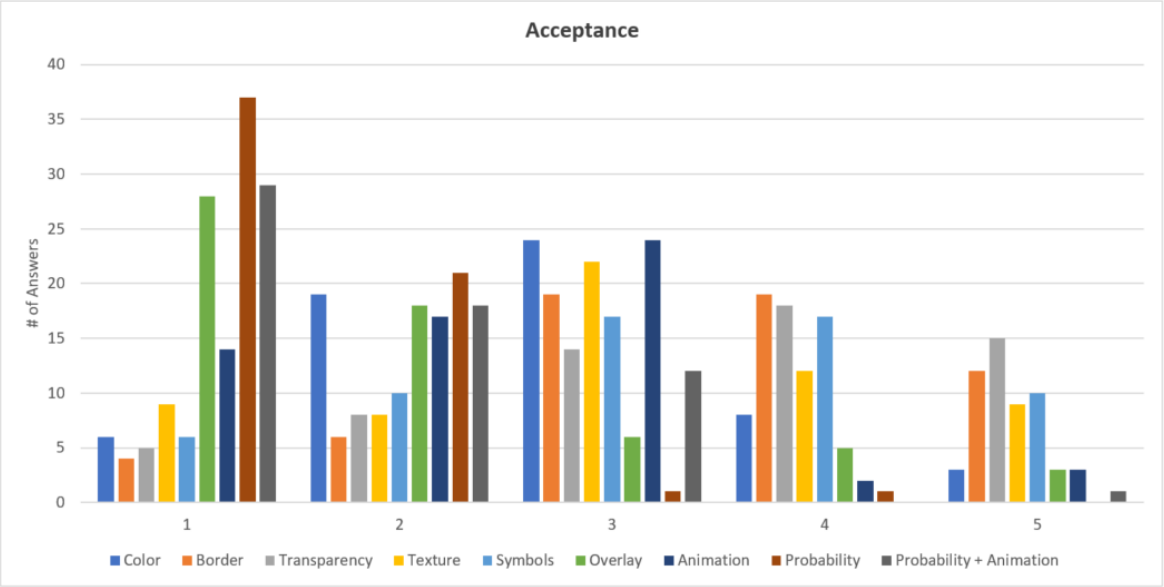


Figure 4.19: Survey results - Acceptance: participants were asked to compare the visualization approaches by specifying the degree to which they would accept to use them in their practical work. The question allows participants to rank each approach in terms of acceptance, using a scale from 1 – 5 (strongly refrain from using, rather not, neutral, accept, and strongly accept).

As Figure 4.19 illustrates, the majority of the participants decided not to use *Probability*, *Probability + Animation*, or *Overlay*, where the 80%, 78.3% and 76.6% values, respectively, represent the percentage of the votes for *strongly refrain* from using and *rather not* use these approaches. On the other hand, varying the *Color Value + Transparency* and *Border Style* performed the best with 55% and 52%, respectively, representing the percentage of votes for *accept* and *strongly accept* to use the approaches. In the end, if all the votes for the *neutral* option are also included in the percentage of votes, the *Color Value + Transparency* and *Border Style* approaches received 78.3% and 83.3%, respectively, as the ranking of voters who did not choose to refrain from using them.

The other approaches, *Symbols*, *Texture Grain*, and *Animation*, received lower acceptance rankings (45%, 35%, and 9%, respectively) and higher neutral rankings (28.3%, 36.6%, and 40%, respectively). According to the intuitiveness results presented in Figure 4.18, animation is well suited to represent positional uncertainty, as more participants interpreted the animation correctly, compared to the static visualizations. However, contradictory to those results, the acceptance results make it evident that participants showed a clear preference for the static visualizations over animation.

After compiling the survey results, we tried to deduce a relationship between the participants' results (intuitiveness, applicability, and acceptance) and their domain knowledge or familiarity with 3D models. The hypothesis assumed that the visualizations would be more intuitive and acceptable with more familiarity or relevant experience. However, the results did not reveal any pattern that would positively support this hypothesis.

### 4.6.3 Interviews

First, the interviews were conducted with subcontractors experienced in either architectural designs, fire safety simulations, or pedestrian flow simulations. The interviews were conducted in two iterations, where the feedback obtained from the participants in the first iteration, regarding possible use cases, was considered in the second iteration. Each iteration consists of a series of questions, including identifying elements with a particular geometric or semantic vagueness, as well as carrying out tasks from the subcontractor perspective, for example, accounting for the impact of vagueness while performing analyses or making a change in the design. The questions and tasks included in each iteration were designed to evaluate the intuitiveness of the approaches. After each iteration, the responses were reviewed and assessed.

**Interviews: Analysis of responses**

Figure 4.20 presents the results of both interview iterations. The y-axis represents the number of questions asked for each approach, and the x-axis depicts the intuitiveness results of both iterations. Except for the *Overlay* approach, the intuitiveness of the approaches noticeably improved in the second iteration; the participants correctly interpreted the information vagueness in most of the approaches. However, similar to the online survey results presented in Figure 4.18, the results of the *Overlay* approach showed relatively low intuitiveness in both iterations.

At the end of the interviews, participants were asked to propose new approaches or extensions to the developed approaches. Two subcontractors proposed extending the BIM authoring tools by including additional indicators over the elements' properties, as illustrated in Figure 4.21. In this case, when the orange color is darker, it implies that the vagueness is higher, and when there is a check mark beside the property, it implies that it is fixed and certain.

## 4.7 Conclusions & future research

Information vagueness is a fundamental issue affecting the process and outcome of designing a building. Careful management and visualization of the information vagueness at the early

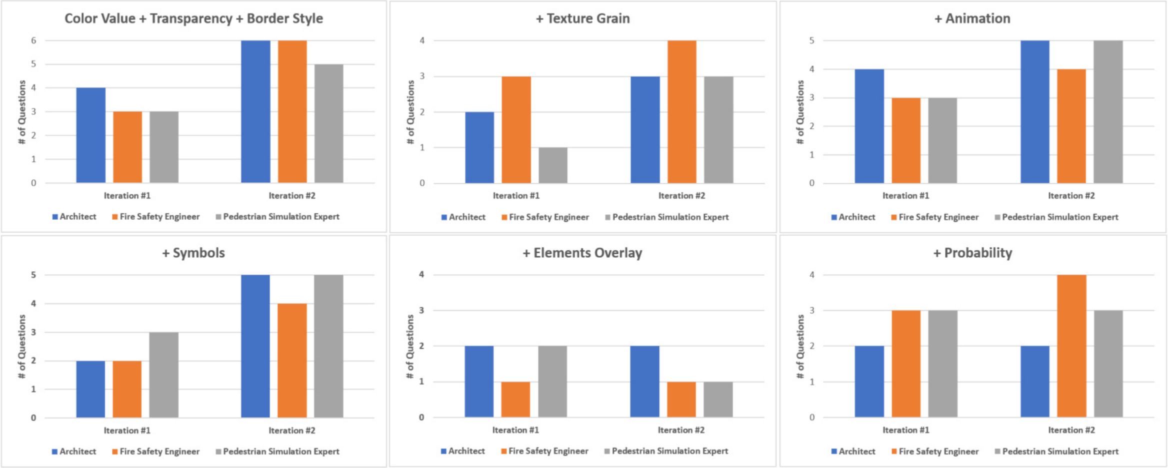
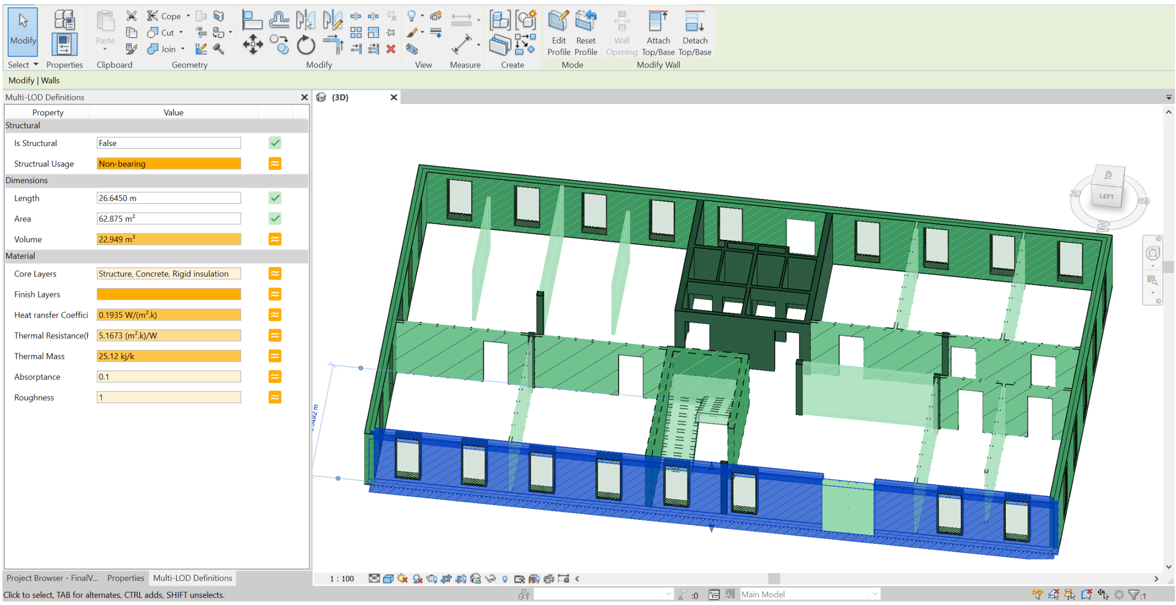


Figure 4.20: Survey results - Interviews: the developed approaches were evaluated through two iterations in terms of intuitiveness. The y-axis represents the number of questions asked for each approach, and the x-axis depicts the intuitiveness results of both iterations.



Multi-LOD Definitions

| Property | Value | |
|---|---|---|
| Structural | | |
| Is Structural | False | ✓ |
| Structrual Usage | Non-bearing | ≈ |
| Dimensions | | |
| Length | 26.6450 m | ✓ |
| Area | 62.875 m² | ✓ |
| Volume | 22.949 m³ | ≈ |
| Material | | |
| Core Layers | Structure, Concrete, Rigid insulation | ≈ |
| Finish Layers | | ≈ |
| Heat ransfer Coeffici | 0.1935 W/(m².k) | ≈ |
| Thermal Resistance(I | 5.1673 (m².k)/W | ≈ |
| Thermal Mass | 25.12 kj/k | ≈ |
| Absorptance | 0.1 | ≈ |
| Roughness | 1 | ≈ |

Click to select, TAB for alternates, CTRL adds, SHIFT unselects.

Figure 4.21: Interviews - Proposed extension: extending BIM authoring tools by the inclusion of additional indicators over the elements' properties. When the orange color is darker, it implies that the vagueness is higher, and when there is a check mark beside the property, it implies that it is certain.

design stages can improve planning quality and reduce project risks. The multi-LOD meta-model facilitates managing the building information throughout the different stages. It makes it possible to formally specify the required information, including a description of the potential vagueness. Additionally, it represents the individual components of the actual building model and verifies information consistency across the design stages.

Expressing the amount of vagueness using visualization techniques assists in evaluating how the model can evolve in the subsequent stages. This paper contributed multiple visualization approaches for depicting vagueness associated with building information models. The approaches developed here aim to address the problem of communicating the information vagueness among the project participants, especially at the early design stages, to support the decision-making process.

The developed approaches were evaluated through an online survey and interviews. The evaluation results positively indicated the participants' ability to use the developed approaches to understand the amount and impact of the vagueness associated with the geometric and semantic information. More specifically, varying the building elements' border style for representing vagueness of the geometric information, and using the combination of color value and transparency for quantifying the reliability of the semantics resulted in relatively high intuitiveness and acceptance by the participants. Hence, using those approaches as a basis for the other approaches assisted in expressing the vagueness associated with more specific use cases, such as including texture for describing the structure reliability as well as animation and symbols for depicting the potential lengths and positions. Additionally, although the participants took relatively less time to solve the survey tasks correctly when animation was included, they preferred the static approaches more.

Based on the experience gained from this research, attempting to communicate the vagueness of multiple building elements or properties simultaneously can be overwhelming to users. In the same context, some domain experts preferred managing the information vagueness solely through attaching it to the individual properties rather than relying on the visualization approaches. In this regard, the visualization approaches presented in this paper can express the information vagueness on various scales, from the overall building model (where the properties' vagueness are aggregated) to the individual elements (where the properties' vagueness are presented as they are), like position and length. Furthermore, the extension proposed by the conducted interviews, shown in Figure 4.21, depicts the associated vagueness information on both the 3D representation as well as the individual properties. Typically, reluctance in using new visualization methods can be reduced through the users' practical evaluation in real-world projects. Certainly, more research is required to advance uncertainty visualization methods further, refine our findings, and provide more evidence.

The developed visualization approaches were evaluated on building models. As a next step, further research is necessary to collect and support infrastructure use cases, such as bridges and tunnels. Accordingly, these approaches can be refined and extended to convey specific and relevant indicators for each particular case. Finally, the exploration and evaluation of the benefits that additional visualization approaches, such as virtual and augmented reality, could support more advanced use cases, such as accounting for the condition and context of construction site by establishing early feedback on the constructability of the developed design.

## Acknowledgements

We gratefully acknowledge the support of the Deutsche Forschungsgemeinschaft (DFG) for funding the project under grant FOR 2363. We thank Ferdinand Tausendpfund GmbH for providing its office building as a sample project. Thank you also to our colleague Fritz Beck for his valuable input and evaluation.

## Appendix: Survey questions

0%

# Vagueness Visualization in Building Models

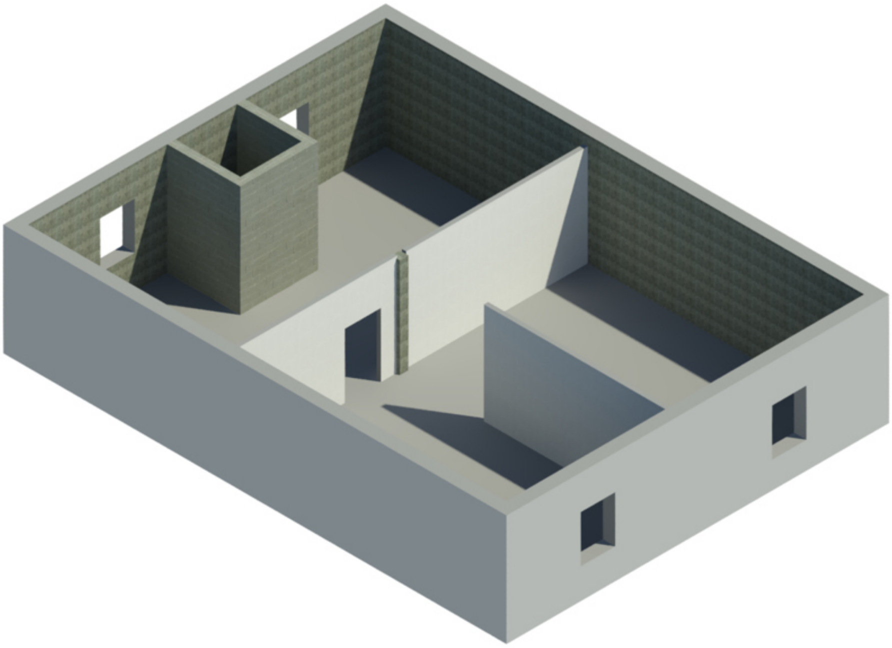
In each design stage, architects and engineers need to make design decisions to develop the design further. Those decisions need to take into account different kinds of boundary conditions and requirements.

In the early design stages, architects explore and evaluate multiple design variants. At those stages, the information is not completely fixed or certain.

The current visualization would wrongly suggest that the design is more elaborate than it actually is, which can lead to false assumptions and model evaluations.

This survey aims at evaluating new visualization approaches that are developed to communicate and convey the amount of vagueness associated with the different building models.

**Current practice for building information visualization:**



There are 38 questions in this survey.

## This survey is anonymous.

The record of your survey responses does not contain any identifying information about you, unless a specific survey question explicitly asked for it.

If you used an identifying token to access this survey, please rest assured that this token will not be stored together with your responses. It is managed in a separate database and will only be updated to indicate whether you did (or did not) complete this survey. There is no way of matching identification tokens with survey responses.

Next

0%

# Introduction

The purpose of the evaluation is to find an adequate way for visualizing and communicating uncertainties concerning the current design of a building.

As always, some questions to find out more about your previous knowledge.
And no worries, you don't need previous knowledge for the evaluation.

✱Please select your domain:

❗ Choose one of the following answers

○ Architecture
○ Civil Engineering
○ Environmental Engineering
○ Computer Science
○ Graphic Design
○ Other / Not Specified
○ Other:

✱Please select what fits best:

❗ Choose one of the following answers

○ Industry
○ Student (Master)
○ Research Associate (Doctorate)
○ Other:

✱Are you familiar with 3D / Geometry modeling? (1 = not familiar / 5 = very familiar)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Next

9%

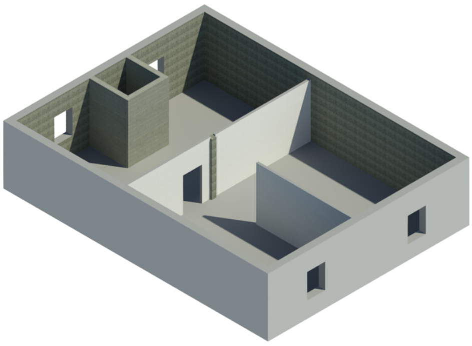
# Approach 1: Color Value

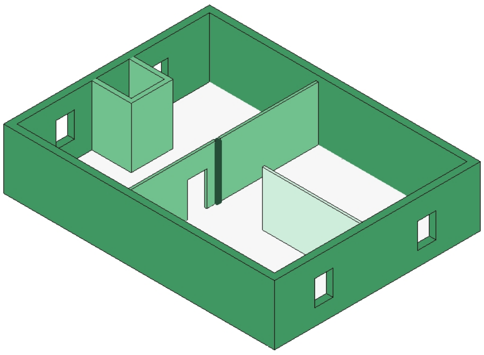
Building models consist of **Geometric** and **Semantic** information.

The **geometric** attributes represent an element's shape and its dimensions, and the **semantics** describe various non-geometric aspects of building elements, including material, cost information, and fire resistance properties.etc.

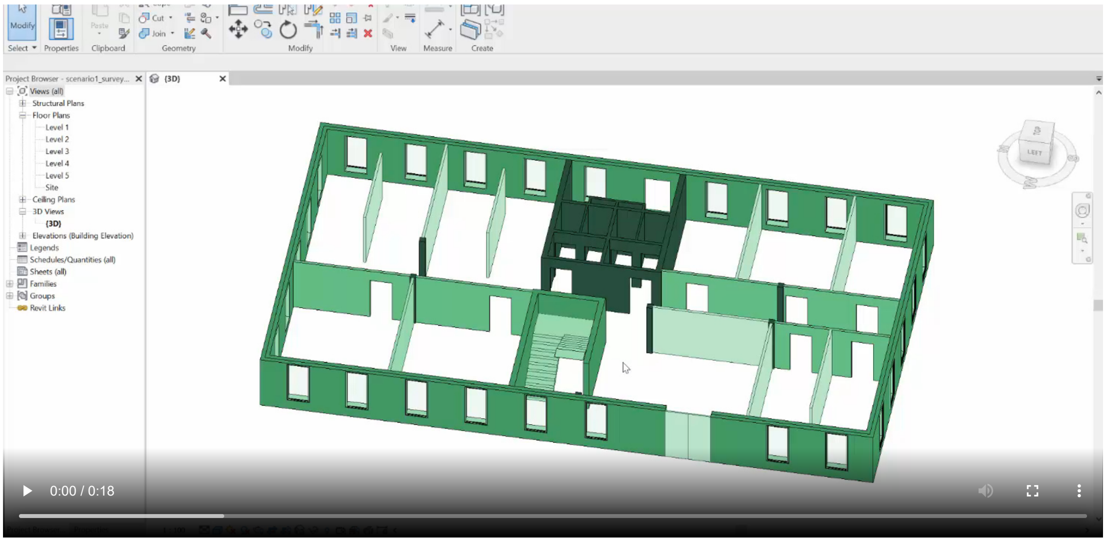
Visualizing building elements' potential vagueness improves the engineers' awareness of the possible states in the subsequent stages. Additionally, such visualization facilitates evaluating the surrounding components' relationships, which improves the quality of the decisions taken.

**Current practice VS. the first developed vagueness visualization approach:**





**Supplementary Video:**



How do you interpret this visualization?

✱Which building elements could have **vague material**:

❗ Check all that apply

☐ DARK green elements

☐ Very LIGHT green elements

☐ LIGHTER green means MORE vagueness

☐ DARKER green means MORE vagueness

✱Please select **all the correct** options:

❗ Check all that apply

☐ The SEMANTIC information of all INTERIOR WALLS is vague

☐ The SEMANTIC information of the DARK green column is vague

☐ The very LIGHT green wall has very vague GEOMETRIC information (it is possible to change the design to split the room)

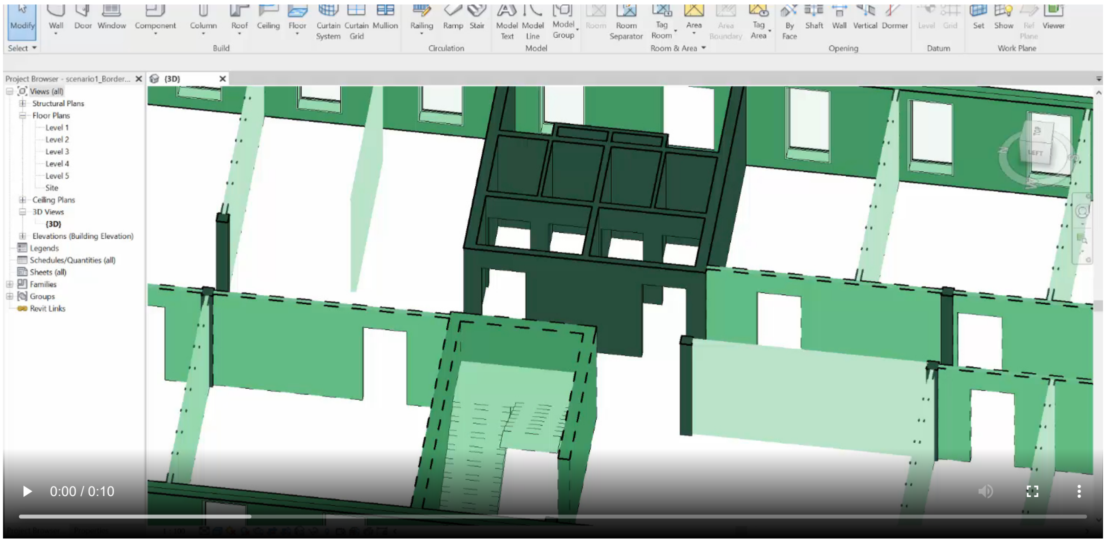
☐ The EXTERIOR WALLS are more vague than the INTERIOR WALLS

Next

18%

# Approach 2: Border Style

This approach extends the previous one by **varying** the **border style** to convey the **vagueness** associated with the **geometric** information.



How do you **interpret** this **visualization**?

✱Which building elements can still move (have **a vague position**):

❶ Check all that apply

☐ ALL elements with DOTTED border style

☐ ALL elements with SOLID border style

☐ ALL elements with DASHED border style

☐ ALL elements with NO border

☐ ALL elements with DARK green color

✱Please select **all the correct** options:

❶ Check all that apply

☐ ALL columns have HIGH POSITION vagueness

☐ EXTERIOR WALLS have LESS DIMENSIONAL vagueness than the stairs

☐ LIGHT green walls can be made of concrete or glass in the next stage

☐ ALL walls are fixed

☐ Walls with NO border has high THICKNESS vagueness

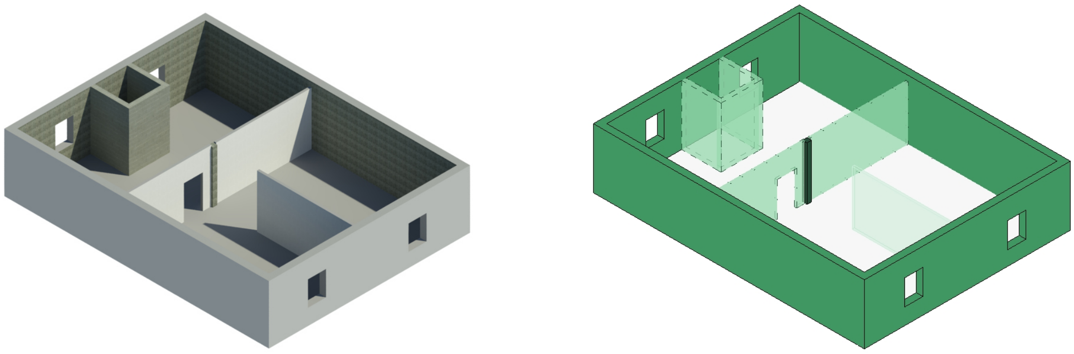
Next

27%

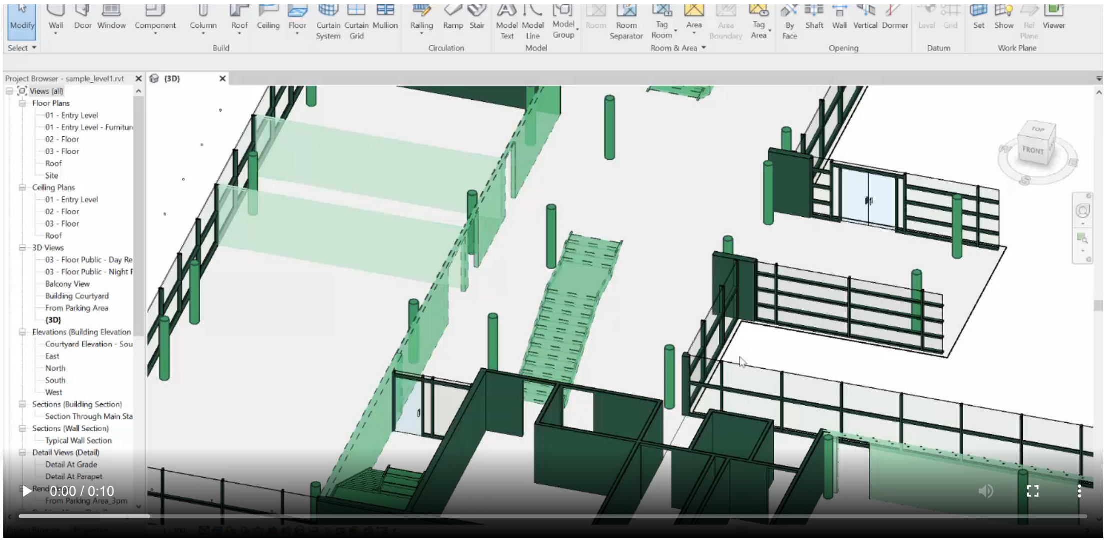
# Approach 3: Transparency

This approach extends the previous approach by including **transparency**.

**Current practice VS. a developed vagueness visualization approach:**



**Supplementary Video:**



How do you interpret this visualization?

✱Which building elements could have **vague dimensions**:

❶ Check all that apply

☐ Very DARK green column with SOLID border style

☐ ALL walls

☐ TRANSPARENT walls with DOTTED border style

☐ DARK green walls with SOLID border style

☐ TRANSPARENT walls with DASHED border style

☐ Walls with NO border

✱Please select **all the correct** options:

❶ Check all that apply

☐ Walls with VERY DARK green color have MORE vague position

☐ Walls with VERY LIGHT and TRANSPARENT green color have HIGHLY vague material and semantics

☐ DARKER green color represents LESS vagueness

☐ Fill color value (dark green, lighter green, and transparent) conveys the GEOMETRIC vagueness

☐ BORDER style (dotted, dashed, and solid) conveys the GEOMETRIC vagueness

✱Which building elements could have a **vague material**:

❗ Check all that apply

☐ Very DARK green column with SOLID border style

☐ TRANSPARENT walls with DOTTED border style

☐ DARK green walls with SOLID border style

☐ TRANSPARENT walls with DASHED border style

☐ Walls with NO border

☐ All walls

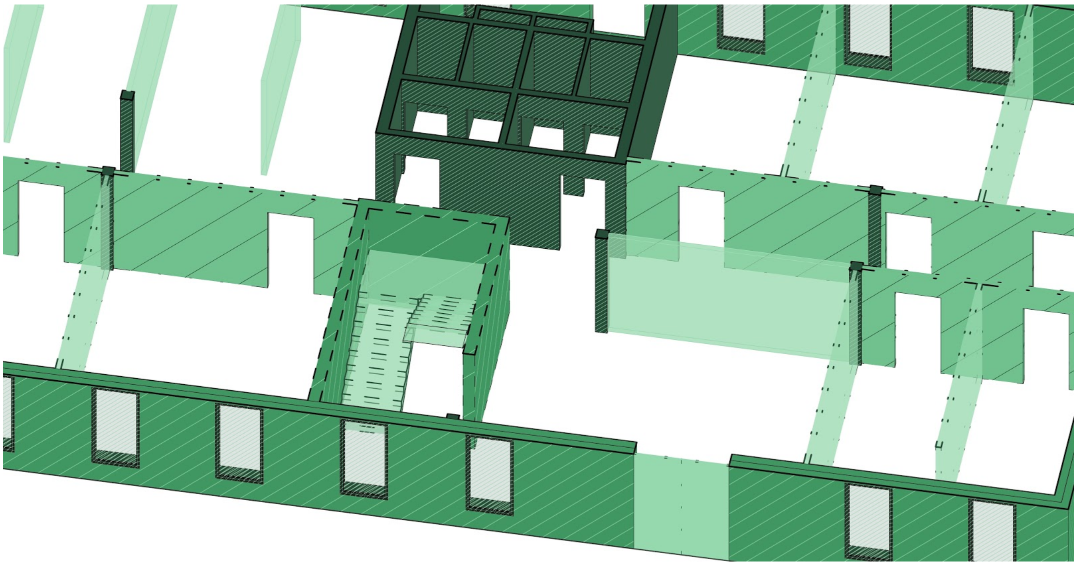
Next

36%

# Approach 3: Texture Grain

In some cases, when evaluating the structural system or compliance with fire-safety regulations, the vagueness associated with the **components' structure**, including **material layers** as well as **thermal** and **structural properties**, is more important than the other semantic information.

**A newly developed vagueness visualization approach to address this use-case based on the previous approaches:**



How do you interpret this visualization?

✱How many **different textures** are available in the picture:

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

✱Which building elements have the highest structure **vagueness** (vagueness related to material layers, insulations, thermal, and structural properties):

❗ Choose one of the following answers

○ EXTERIOR walls

○ INTERIOR walls NO texture overlay

○ INTERIOR walls WITH TEXTURE overlay and DOTTED border style

○ Staircase walls (TEXTURE overlay and DASHED border style)

○ Columns

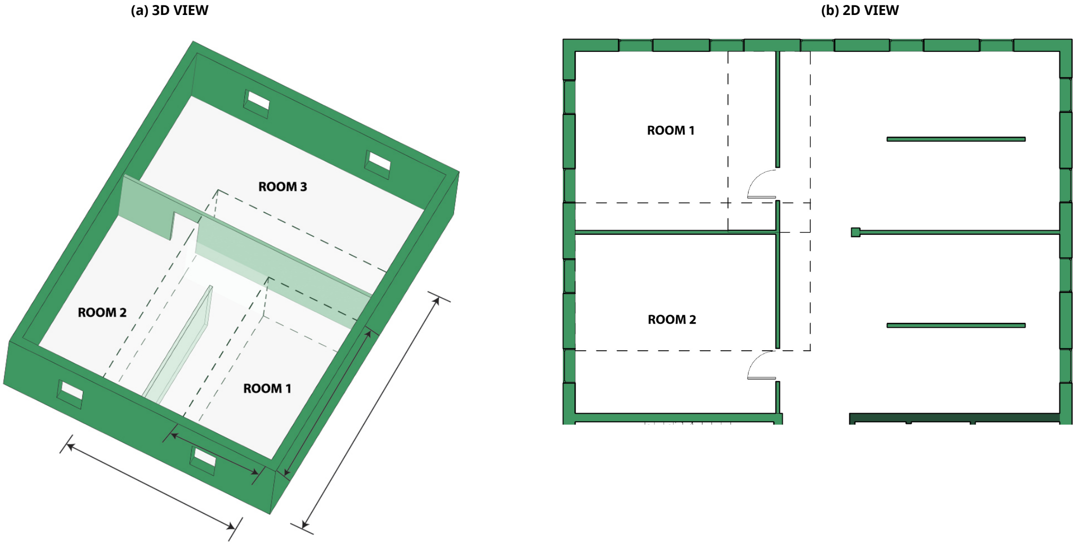
Next

45%

# Approach 4: Overlay

Usually, when designers detail the building model, they evaluate the **individual components' position** and **dimensions** while considering all the possible use-cases.

Therefore, in this section, we propose extending the previous approaches to represent the impact of vagueness on the possible **positions** and **dimensions**.



✱For **(a) 3D view:** which of the following statements is **correct.**

**<u>The walls inside the DASHED box:</u>**

❶ Check all that apply

☐ Can move LEFT and RIGHT

☐ Have vague LENGTH

☐ Have fixed THICKNESS

☐ Have vague THICKNESS

☐ Can be CONNECTED

✱For **(a) 3D view:** which of the following statements is **correct.**

❶ Check all that apply

☐ ROOM 1 and ROOM 2 can be completely SEPARATED

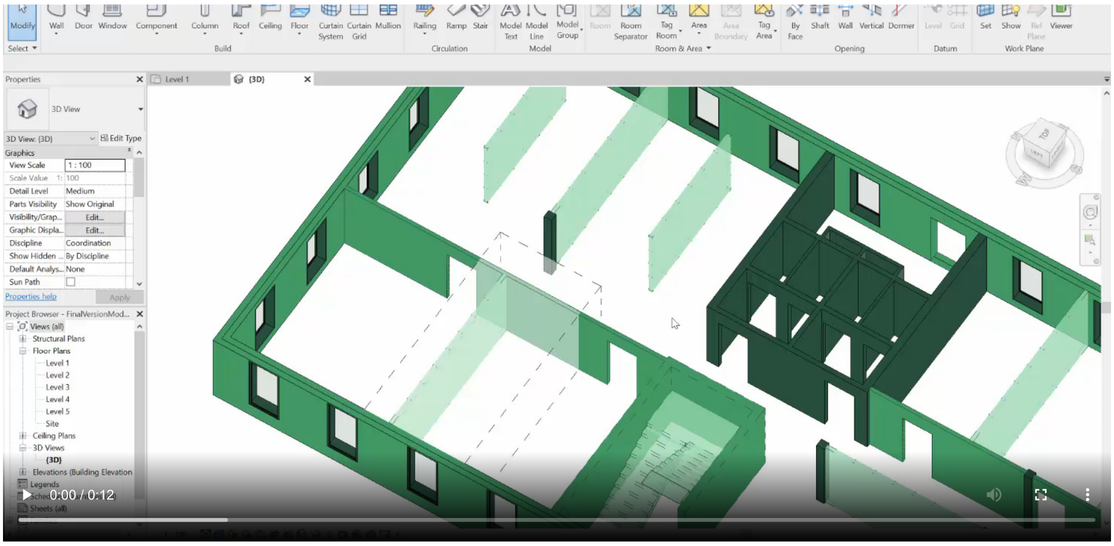
☐ ROOM 1 area is NOT fixed

☐ ROOM 1 and ROOM 2 can be MERGED into one room

☐ ROOM 3 and ROOM 1 can be MERGED into one room

✱For **(b) 2D view:** which of the following statements is **correct.**

**Supplementary Video:**



❗ Check all that apply

☐ ROOM 1 area can increase A LOT into the direction of ROOM 2

☐ ROOM 2 area can increase A LOT into the direction of ROOM 1

☐ POSITION of the entrance / door of ROOM 2 is fixed

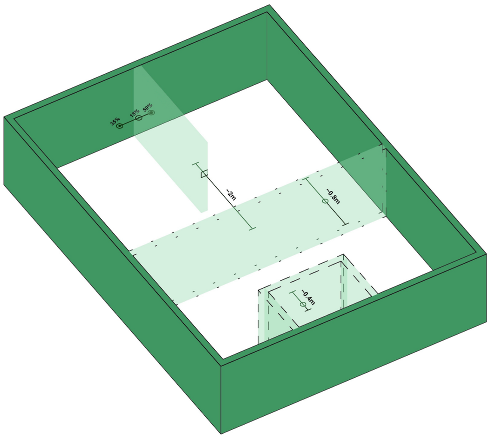
☐ POSITION of the entrance / door of ROOM 1 is fixed

Next

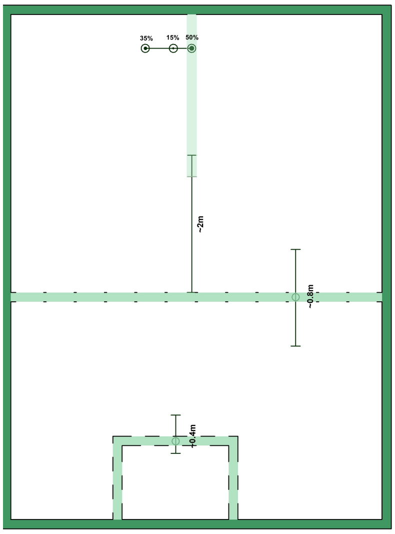
# Approach 5: Symbols

In this approach, **additional symbols** are added to the model to communicate the **vagueness** associated with the building elements **length** and **position**.

**(a) 3D VIEW**

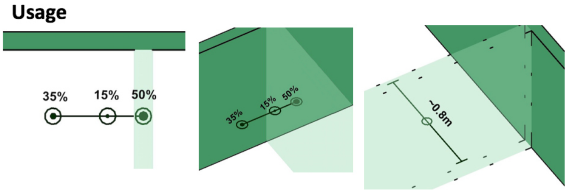


**(b) 2D VIEW**



*** Symbol** | **Usage**

○



Which of the following statements is **correctly** describing this **symbol:**

❶ Check all that apply

☐ CIRCLE represents THICKNESS

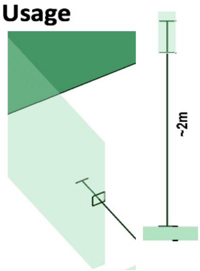
☐ CIRCLE represents CENTER position

☐ CIRCLE represents SURFACE position

☐ None of the above

*** Symbol** | **Usage**

□



Which of the following statements is **correctly** describing this **symbol:**

❶ Check all that apply

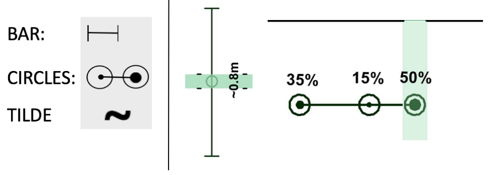
☐ RECTANGLE represents LENGTH

☐ RECTANGLE represents SURFACE position

☐ RECTANGLE represents CENTER position

☐ RECTANGLE represents ORIENTATION

*** Symbols** **Usage**



Which of the following statements is **correctly** describing these **symbol:**

❶ Check all that apply

☐ BAR represents exactly TWO values

☐ BAR represents a RANGE of values

☐ CIRCLES represent a RANGE of values

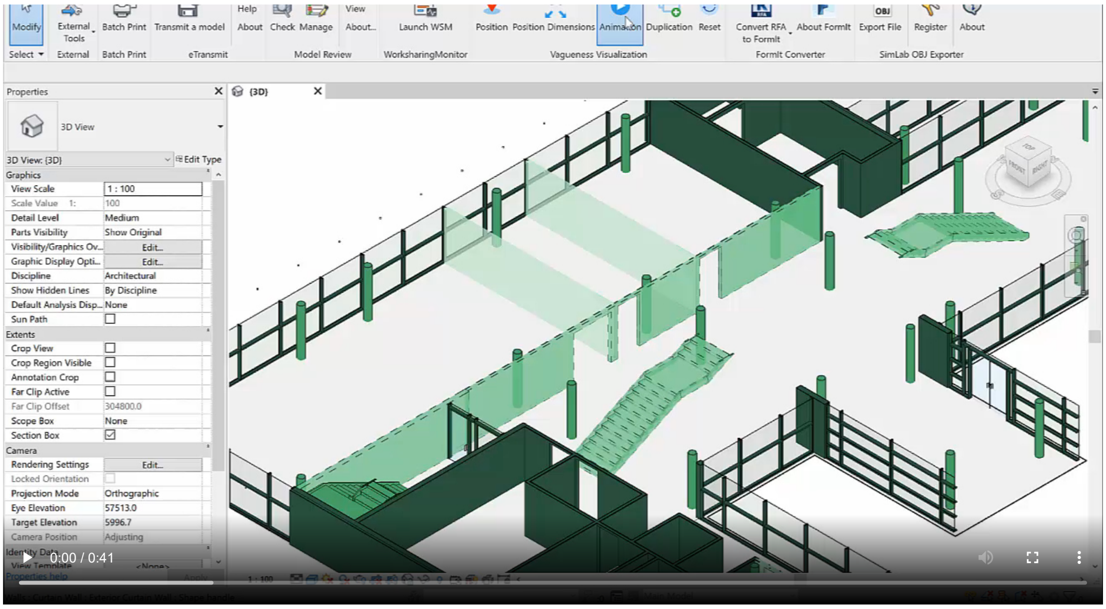
☐ CIRCLES represent a SET of values

☐ TILDE symbol (~) describes the AMOUNT of vagueness

Next

## Approach 6: Animation

The following **animation** approach aims to **signify** the impact of the elements' possible **positions** and **lengths**, for example, the vagueness associated with the interior walls strongly affects the story's layout and the designed functions.



| | |
|---|---|
| Graphics | |
| View Scale | 1 : 100 |
| Scale Value 1: | 100 |
| Detail Level | Medium |
| Parts Visibility | Show Original |
| Visibility/Graphics Ov... | Edit... |
| Graphic Display Opti... | Edit... |
| Discipline | Architectural |
| Show Hidden Lines | By Discipline |
| Default Analysis Disp... | None |
| Sun Path | ☐ |
| Extents | |
| Crop View | ☐ |
| Crop Region Visible | ☐ |
| Annotation Crop | ☐ |
| Far Clip Active | ☐ |
| Far Clip Offset | 304800.0 |
| Scope Box | None |
| Section Box | ☑ |
| Camera | |
| Rendering Settings | Edit... |
| Locked Orientation | ☐ |
| Projection Mode | Orthographic |
| Eye Elevation | 57513.0 |
| Target Elevation | 5996.7 |
| Camera Position | Adjusting |
| Identity Data | |
| View Template | <None> |

How do you interpret this visualization?

✱Taking into account the video above, which building elements have the **highest positional vagueness**?

❶ Check all that apply

☐ The DARK green column

☐ The INTERIOR walls with DOTTED border style

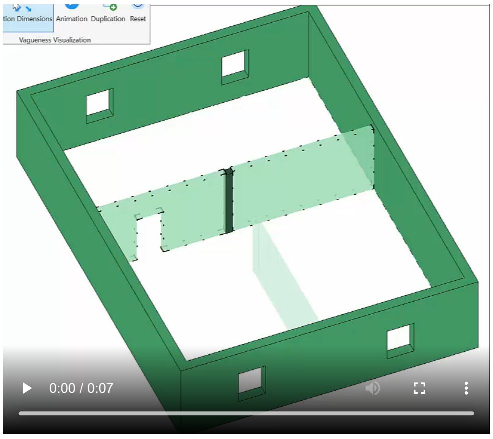
☐ Walls moving for BIGGER distance

☐ The INTERIOR wall with NO border

☐ The EXTERIOR walls

☐ Walls moving FAST

✱



Taking into account the video above, please select **all the correct** options:

❶ Check all that apply

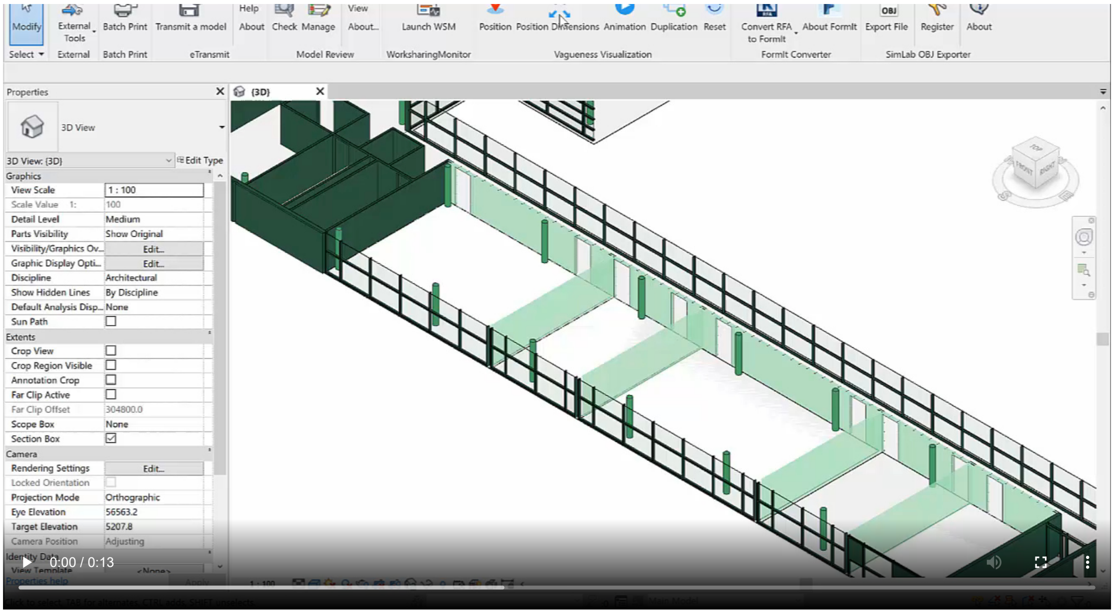
☐ The LENGTH of the DOTTED walls is fixed

☐ The LENGTH of the wall with NO border is still flexible

☐ The STOREY/FLOOR might become as ONE room (no separations, all walls are open)

☐ The STOREY/FLOOR might be separated to THREE rooms

*



3D View: {3D}

| | |
|---|---|
| **Graphics** | |
| View Scale | 1 : 100 |
| Scale Value 1: | 100 |
| Detail Level | Medium |
| Parts Visibility | Show Original |
| Visibility/Graphics Ov... | Edit... |
| Graphic Display Opti... | Edit... |
| Discipline | Architectural |
| Show Hidden Lines | By Discipline |
| Default Analysis Disp... | None |
| Sun Path | ☐ |
| **Extents** | |
| Crop View | ☐ |
| Crop Region Visible | ☐ |
| Annotation Crop | ☐ |
| Far Clip Active | ☐ |
| Far Clip Offset | 304800.0 |
| Scope Box | None |
| Section Box | ☑ |
| **Camera** | |
| Rendering Settings | Edit... |
| Locked Orientation | ☐ |
| Projection Mode | Orthographic |
| Eye Elevation | 56563.2 |
| Target Elevation | 5207.8 |
| Camera Position | Adjusting |
| **Identity Data** | |
| View Template | <None> |

Taking into account the video above, how do you interpret this visualization?

❶ Check all that apply

☐ Rooms' AREAs might change in the next stage

☐ SEPARATED rooms might be MERGED in the next stage

☐ Stairs' POSITIONs are flexible

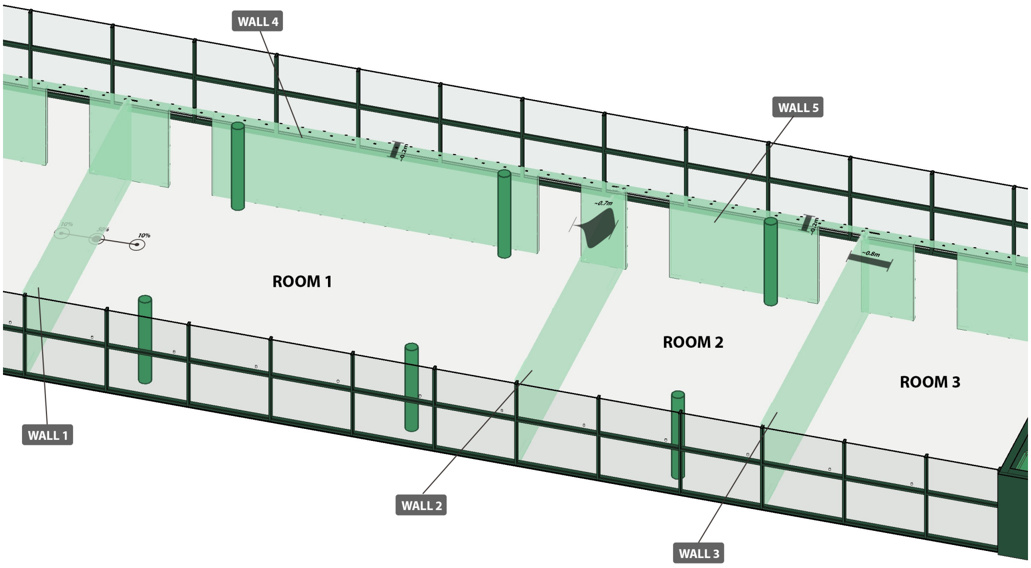
☐ Columns' POSITIONs are fixed

Next

72%

# Approach 7: Probability

In most of the cases, although the **information is vague** (there can be multiple possible values), architects and engineers might have **particular value with higher probability** than the others.



✱Please select **all the correct options**:

❶ Check all that apply

☐ WALL 4 has a FIXED position

☐ WALL 3 position has a RANGE of possible values

☐ WALL 1 is now in the MOST probable position

☐ WALL 2 is now in the MOST probable position

☐ WALL 1 has exactly THREE possible positions

✱Please select **all the correct options**:

❶ Check all that apply

☐ ROOM 3 area can be BIGGER in the next stage

☐ ROOM 1 area will MOST probably have the SAME area in the next stage

☐ ROOM 2 has a HIGH probability of expanding towards ROOM 1

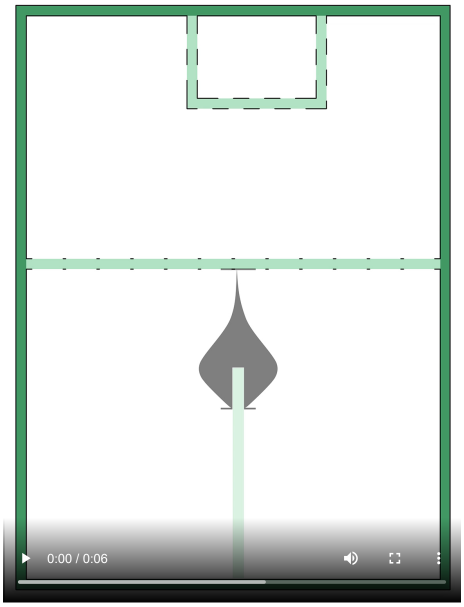
☐ ROOM 2 has a HIGH probability of expanding towards ROOM 3

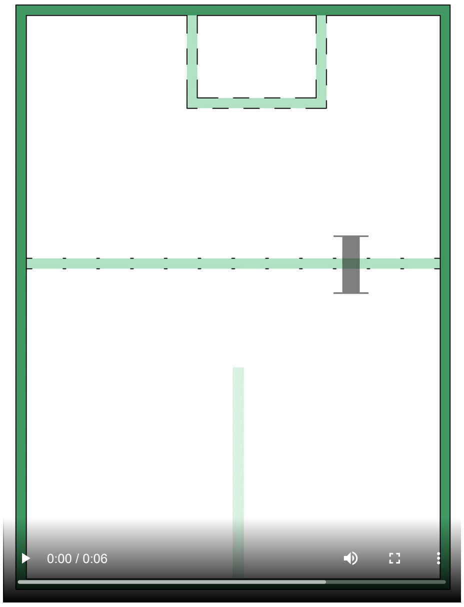
☐ None is correct

Next

81%

## Approach 8: Probability Animation





How do you interpret this visualization?

✱Please select **all the correct options**:

❶ Check all that apply

☐ The DISTRIBUTION FUNCTION overlay represents the probability of ALL possible values

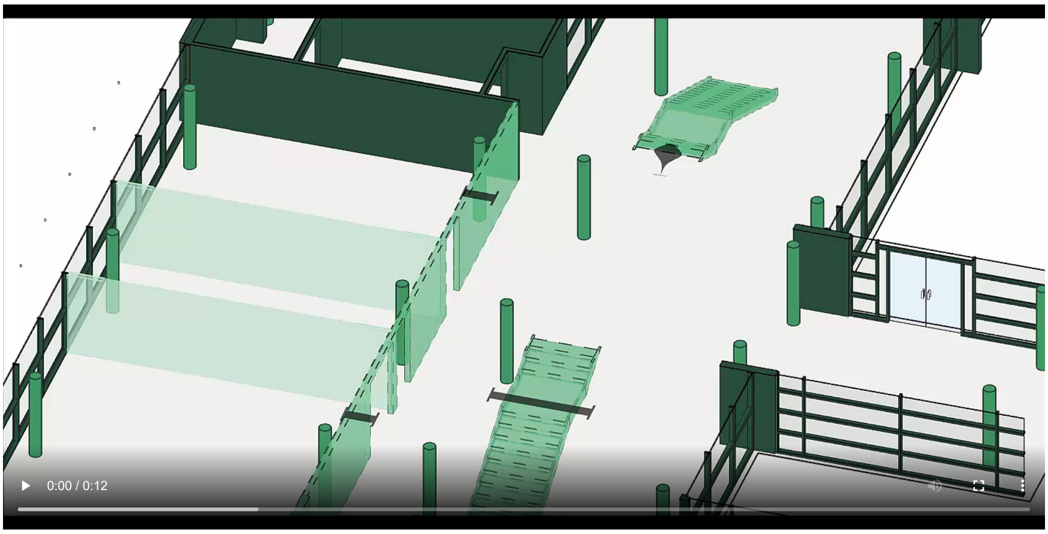
☐ The animation SPEED represents the AMOUNT of vagueness (HIGHER vagueness means FASTER animation)

☐ The animation SPEED represents the PROBABILITY of a particular value (SLOWER animation means HIGHER probability)

☐ The animation SPEED is HIGHER when the wall has NO border style

☐ None of the above

**✱Supplementary Video:**



Taking into account the video above, please select **all the correct options**:

Check all that apply

☐ ALL stairs have high probability of changing their POSITION in the next stage

☐ SOME stairs have low probability of changing their POSITION in the next stage

☐ SOME stairs have high probability of changing their LENGTH in the next stage

☐ ALL stairs have high probability of changing their LENGTH in the next stage

☐ None of the above

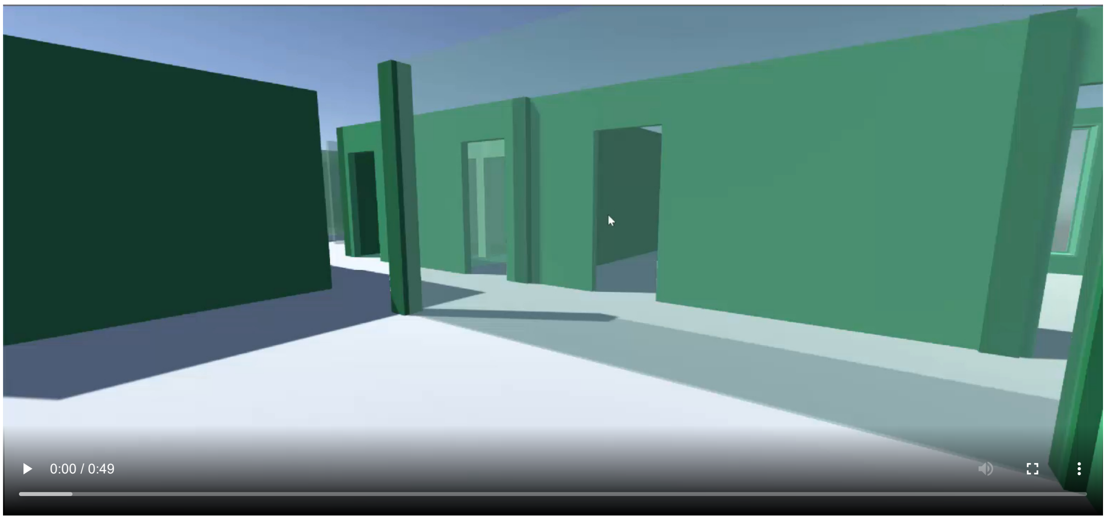
Next

# Application View and Acceptance

**Walkthroughs** are one of the most common extensions of 3D visualization. They offer a more realistic depiction of the relationships between elements and foster a better spatial understanding of the proposed design.

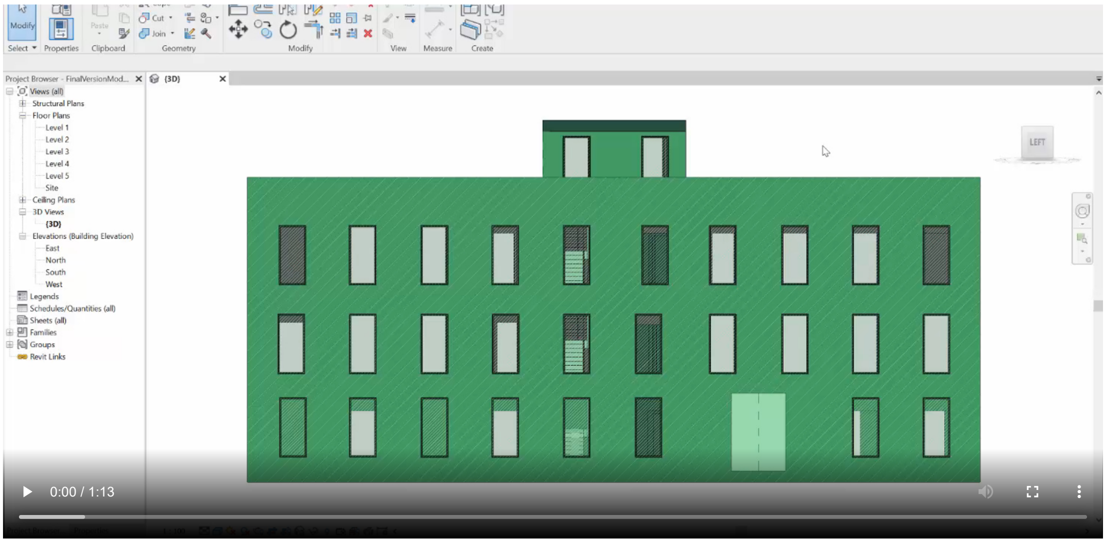
The experience resulting from this kind of visualization can highlight essential aspects and provoke detailed discussions, such as the discovery of unexpected conflicts and safety issues when collaboratively working with the different domain experts (in a design review meeting).

**Vagueness visualization using walkthroughs:**



Additionally, the following video illustrates applying the proposed approaches to conveying the vagueness of the overall building model:

**Model overview for ALL approaches:**



✱Which **visualization approaches** are **suitable** for visualizing the vagueness associated with the **overall building model** (the entire 3D building at once)? (**1 = strongly disagree | 5 = strongly agree**)

❶ Check all that apply

☐ Color Value

☐ Border Style

☐ Color Value + Transparency

☐ Texture Grain

☐ Symbols

☐ Overlay

☐ Animation

☐ None

✱Which **visualization approaches** are **suitable** for visualizing the vagueness associated with the **Storey / Floor view**? (**1 = strongly disagree | 5 = strongly agree**)

❶ Check all that apply

☐ Color Value

☐ Border Style

☐ Color Value + Transparency + Border Style

☐ Color Value + Border Style + Texture Grain

☐ Symbols

☐ Overlay

☐ Animation

☐ None

✱Which **visualization approaches** are **suitable** for visualizing the vagueness associated with the **Zone / Room view**? (**1 = strongly disagree | 5 = strongly agree**)

❶ Check all that apply

☐ Color Value

☐ Border Style

☐ Color Value + Transparency + Border Style

☐ Color Value + Border Style + Texture Grain

☐ Symbols

☐ Overlay

☐ Animation

☐ None

✱Which **visualization approaches** are **suitable** for visualizing the vagueness associated with the **Walkthrough**? (**1 = strongly disagree | 5 = strongly agree**)

❶ Check all that apply

☐ Color Value

☐ Border Style

☐ Color Value + Transparency + Border Style

☐ Color Value + Border Style + Texture Grain

☐ Symbols

☐ Overlay

☐ Animation

☐ None

✱Would you prefer to use the vagueness visualization approach **Color Value** in your work? (**1 = strongly refrain** | **2 = rather not** | **3 = neutral** | **4 = accept** | **5 = strongly accept**)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

✱Would you prefer to use the vagueness visualization approach **Color Value + Border Style** in your work? (**1 = strongly refrain** | **2 = rather not** | **3 = neutral** | **4 = accept** | **5 = strongly accept**)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

✱Would you prefer to use the vagueness visualization approach **Color Value + Border Style + Transparency** in your work? (**1 = strongly refrain** | **2 = rather not** | **3 = neutral** | **4 = accept** | **5 = strongly accept**)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

✱Would you prefer to use the vagueness visualization approach **Texture Grain** in your work? (**1 = strongly refrain** | **2 = rather not** | **3 = neutral** | **4 = accept** | **5 = strongly accept**)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

✱Would you prefer to use the vagueness visualization approach **Symbols** in your work? (**1 = strongly refrain** | **2 = rather not** | **3 = neutral** | **4 = accept** | **5 = strongly accept**)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

✱Would you prefer to use the vagueness visualization approach **Overlay** in your work? (**1 = strongly refrain** | **2 = rather not** | **3 = neutral** | **4 = accept** | **5 = strongly accept**)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

✱Would you prefer to use the vagueness visualization approach **Animation** in your work? (**1 = strongly refrain** | **2 = rather not** | **3 = neutral** | **4 = accept** | **5 = strongly accept**)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

✱Would you prefer to use the vagueness visualization approach **Probability** in your work? (**1 = strongly refrain** | **2 = rather not** | **3 = neutral** | **4 = accept** | **5 = strongly accept**)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

✱Would you prefer to use the vagueness visualization approach **Probability + Animation** in your work? (**1 = strongly refrain** | **2 = rather not** | **3 = neutral** | **4 = accept** | **5 = strongly accept**)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Submit

# Chapter 5

# Ensemble-learning approach for the classification of Levels Of Geometry (LOG) of building elements

Previously published as: Abualdenien, J.; Borrmann, A.: *Ensemble-learning approach for the classification of Levels Of Geometry (LOG) of building elements*, Advanced Engineering Informatics 51 (1474-0346), pp. 10149, 2022, DOI: 10.1016/j.aei.2021.101497

## abstract

The provision of geometric and semantic information is among the most fundamental tasks in BIM-based building design. As the design is constantly developing along with the design phases, there is a need for a formalism to define its maturity and detailing. In practice, the concept of Level of Development (LOD) is used to specify what information must be available at which time. Such information is contractually binding and crucial for different kinds of evaluations. Numerous commercial and open-source BIM tools currently support the automatic validation of semantic information. However, the automatic validation of the modeled geometry for fulfilling the expected detailing requirements is a complex and still unsolved task. In current practice, domain experts evaluate the models manually based on their experience. Hence, this paper presents a framework for formally analyzing and automatically checking the Level of Geometry (LOG) of building information models. The proposed framework first focuses on generating a LOG dataset according to the popular LOD specifications. Afterwards, multiple geometric features representing the elements' complexity are extracted. Finally, two tree-based ensemble models are trained on the extracted features and compared according to their accuracy in classifying building elements with the correct LOG. Measuring the modeling time showed a 1.88 – 2.80-fold increase between subsequent LOGs, with an 8 – 15-fold increase for LOG 400 compared to LOG 200. The results of classifying the LOG indicated that the combination of 16 features can represent the LOG complexity. They also indicated that the trained ensemble models are capable of classifying building elements with an accuracy between 83% and 85%.

## 5.1 Introduction

The design and detailing decisions made throughout the building design phases significantly influence a project's time, effort, and cost (Hooper, 2015; Howell, 2016). Starting from the schematic design, the project size, building shape, and materiality are defined broadly in order to explore the different possible options. The decisions made in the early phases form the design intent, representing the basis for further detailing (Howell, 2016; Steinmann, 1997). Such detailing includes refining the elements' geometry and evaluating the different combinations of material layers.

As construction projects are multi-disciplinary, a fundamental pillar for integrating building information models is describing the required elements' maturity at every milestone and for every deliverable through the design phases. This is crucial for the overall collaboration among the project participants because it acts as an agreement on (what) information should be available at what time (when). Based on the available information, it can be decided what the model can be used for (purpose), which makes it possible to determine what model deliverables are expected from the actors involved (who) (Beetz et al., 2018). The exchange of complete and compliant Building Information Modelling (BIM) data within the Architecture, Engineering, and Construction (AEC) industry is crucial, as it is prescribed in legal agreements, where the content of the individual elements is specified. Accordingly, a common legal framework for organizing this data is required.

Data quality is described by compliance with its requirements' characteristics (ISO, 2015). More specifically, the quality of building information is expressed by the correctness and completeness of the topological relationships, geometric detailing, and semantics. Various guidelines have been published to deliver a standard that practitioners can use as a basis for a common language in their projects. When describing the detailing decisions, the Level of Development (LOD) (BIMForum, 2019), is a popular concept for defining the content of a model at a certain point during the design process. The LOD refers to the completeness and reliability of the building elements' information.

For more than a decade, practitioners have relied on the LOD terminology to specify which information they need to carry out and deliver their tasks (Hooper, 2015; Leite et al., 2011; van Berlo & Bomhof, 2014). However, as the different LOD definitions are loosely defined (Bolpagni & Ciribini, 2016; van Berlo & Bomhof, 2014), each practitioner has a different interpretation of what a specific LOD means and which information should be present in the model (Bolpagni & Ciribini, 2016; Leite et al., 2011; van Berlo & Bomhof, 2014). Such inconsistencies cause severe miscommunication and additional expenditure, which increase project risks (Hooper, 2015; Leite et al., 2011). Therefore, multiple efforts have been dedicated to developing comprehensive specifications worldwide to provide a consensus on the required information at the different LODs (more details are provided in Section 5.2.3). The most popular among these are the BIMForum's LOD specification (BIMForum, 2019) and Trimble's Project Progression Planning (Trimble, 2013), which are used as the basis for the research conducted in this paper.

The geometric detailing of building elements is essential for carrying out different kinds of analyses and evaluations (e.g., energy analysis, evaluation of design options, and cost estimation) that support decisions during design and construction (Abualdenien & Borrmann, 2020b; Leite et al., 2011; Schneider-Marin et al., 2020; Zahedi et al., 2019). For example, according to the BIMForum's specification, performing clash detection or analyzing the constructability of the outer building shell requires modeling the precise geometry (available from LOD 300) and connections between the elements (available from LOD 350). The requirements of each LOD comprise both semantic information (a.k.a. Level of Information (LOI)) and geometric detailing (a.k.a. Level of Geometry (LOG)). The LOI is represented by a set of properties, whereas the LOG is described by the geometric parts that need to be modeled, like modeling the overall shape precisely or the necessary reinforcement parts.

As the LODs are referenced in contracts and BIM execution plans, their requirements must be fulfilled when delivering and exchanging building models. In this regard, automatically checking the completeness of the semantic information is straightforward (Abualdenien & Borrmann, 2019) and supported by numerous commercial and open BIM tools. However, automatically checking that the detailing of the modeled geometry fulfills the expected LOG requirements is a complex and still unsolved task; currently, domain experts evaluate the models manually based on their experience. Therefore, the primary focus of this paper is to formally define the LOGs to identify a given BIM element's LOG in accordance with these specifications.

Machine Learning (ML) algorithms, including Support Vector Machines (SVM), Random Forests (RF), and Artificial Neural Networks (ANN), have demonstrated high performance in addressing non-linear multi-class classification and regression problems in different domains (Géron, 2019). Generally, these approaches utilize statistics in order to extract generalizable and predictable patterns from a training dataset. Accordingly, the basic concept lies in implicitly deducing correlations between the provided data (input) and the expected result (output). In the AEC industry, the application of ML algorithms has become popular for multiple use-cases. For example, J. Zhang et al., 2019 developed a RF model for predicting the uniaxial compressive strength of lightweight self-compacting concrete (J. Zhang et al., 2019). Dong et al., 2020 trained an eXtreme Gradient Boosting (XGBoost) model (T. Chen & Guestrin, 2016) to predict concrete electrical resistivity for structural health monitoring (Dong et al., 2020). Finally, Braun et al., 2020 developed a deep-learning model for supporting progress monitoring through detecting elements in point cloud data and comparing them to the BIM model (Braun et al., 2020).

This paper addresses the currently existing gap of determining the LOG of building elements by investigating the major characteristics representing the degree of detailing based on a formal metric. To this end, a set of different BIM element types (a.k.a. families) are modeled at multiple LOGs, and the geometric information of each level is investigated. In more detail, the elements are modeled using Autodesk Revit[1] and exported into triangulated meshes. Then, for each LOG, the geometric features are extracted, and their complexity is measured using a combination of multiple advanced geometry processing algorithms. Finally, RF and XGBoost models are trained on the extracted geometric features to classify the LOG of any given building element automatically. The contributions of this study are threefold: First, evaluating and measuring the necessary time and effort in modeling according to the common LOD specifications. Second, identifying the geometric features that are capable of representing the building elements' complexity across the LOGs. Third, evaluating the performance of state-of-the-art ensemble-learning models for classifying the LOG of elements. In this paper, *Shape Complexity* is a high-level term used to describe the overall shape composition, including the modeled parts on the different LOGs. Additionally, the term *Geometric Complexity* describes the geometric features necessary for representing the different shape parts, including vertices, edges, etc.

The paper is organized as follows : Section 2 discusses the background and related work, including shape complexity, LODs, and ensemble-learning. Section 3 provides an overview of the framework developed in this paper, explaining the process followed in modeling the different families to generate the LOG dataset. Additionally, Section 3 presents the geometric features selected to represent the building elements' complexity at the different LOGs. For classifying the LOG of building elements, RF and XGBoost models are developed and evaluated in Section 4. Moreover, Section 4 assesses the trained models' robustness by evaluating their performance on a re-meshed test dataset. Section 5 emphasizes the applicability of using the developed approach in practice via a real-world case study. Finally, Section 6 summarizes our results and presents an outlook for future research.

## 5.2 Background & related work

### 5.2.1 3D shapes

The 3D representation of objects is a fundamental perspective for numerous domains, from computer graphics to BIM. Especially in BIM, the 3D representation of building elements is the primary way of defining the shape of a building and its components. It is also a fundamental aspect for performing a variety of tasks, including clash detection, quantity take-off, or even exploring the reliability of the building information across the design phases

---

[1] https://www.autodesk.com/products/revit/overview

(Abualdenien & Borrmann, 2020b; Borrmann et al., 2018a). In BIM models, the 3D geometry is typically represented through two main approaches (Borrmann & Berkhahn, 2018), (1) explicit modeling (a.k.a. boundary representation), which describes the geometrical surface characteristics, volume, and topology through a graph of faces, edges, and vertices, and (2) implicit modeling, which describes the geometric features through a sequence of operations that form the final representation when performed in the defined order.

A popular approach of explicit modeling is the polygon mesh representation (Botsch et al., 2010; Garland, 1999; Shikhare, 2001). Polygonal meshes require only a small number of polygons to represent simple shapes (regardless of their size). Additionally, a polygonal mesh has the necessary capability to comprehensively represent complex shapes with high resolution, capturing the salient surface features. Accordingly, simple shapes are represented by a few large polygons, while detailed and complex shapes are represented by many small polygons. Polygons comprise a set of vertices, which are interpolated through a connectivity graph to approximate the desired surface. On the other hand, Constructive Solid Geometry (CSG), extrusions, and sweeps are common operations of procedural modeling. In comparison to explicit modeling, procedural approaches have the advantage that the modeling history can be transported, which provides the potential for modifying the geometry in the receiving application. As however, misinterpretations are more likely when processing procedural descriptions, boundary representations are often favored over procedural representation in many BIM exchange scenarios (Borrmann & Berkhahn, 2018).

Extracting the geometric features from building models is a fundamental part of the methodology presented in this paper. Hence, it is crucial to choose and follow a unified approach during the study. Since the implicit modeling approaches can also be represented using boundary representations, all the 3D shapes investigated in this paper were represented as polygon meshes.

### 5.2.2 Shape complexity

The meaning and measurement of shape complexity varies according to different aspects. Processing geometric models can be as simple as iterating over a mesh's vertices, faces, and edges, or as complex as performing different calculations to extract information about curvature or shape topology. Numerous researchers have developed algorithms to retrieve the most dominant features of the different shapes (Botsch et al., 2010), including detecting sharp edges, deducing surface patches, and decomposing the shape into smaller and meaningful shapes, a.k.a. segmentation (Shapira et al., 2008). Dominant features provide an essential description of the geometrical objects' resolution and detail. In the same context, Hanocka et al., 2019 and Nikhila et al., 2020 have developed MeshCNN (Hanocka et al., 2019) and

PyTorch3D (Nikhila et al., 2020) by employing deep-learning approaches to analyze, process, and extract features from 3D shapes.

A popular classification for shape complexity was first introduced by Forrest, who defines three main types (Forrest, 1974): (1) geometric, which describes the shapes' basic features, such as lines, curves, faces, etc., (2) combinatorial, which refers to the topology of the shape, i.e., the number of components that it comprises, and (3) dimensional, which classifies the shape as 2D, 2.5D, or 3D. Other researchers have interpreted 3D shapes and their complexity through shape grammars (Heisserman, 1994). Shape grammars describe the shape decomposition as a set of rules and a series of transformations, including addition, subtraction, rotation, etc.

Accordingly, defining what shape complexity means in the AEC industry requires the specification of which geometric features are essential for capturing the degree of maturity of building elements at the different LOGs (Abualdenien & Borrmann, 2020a).

### 5.2.3 Level of Development (LOD)

As a response to the need to have a consensus about what information should exist during the design process of building elements, various guidelines were published to deliver a standard that practitioners can use as a basis for a common language in their projects. Prior to the LOD concept, a relatively similar concept, a.k.a. Level of Detail (LoD), was already common in computer graphics. The LoD is used to bridge the graphical complexity and rendering performance of a computer program by regulating the amount of detail used to represent the virtual world. In computer graphics, the LoD concept is mainly concerned with geometrical detailing (Luebke, Reddy, Cohen, et al., 2003). In the context of the data exchange standard CityGML, the LoD represents different levels of geometric and semantic complexity of a city model (Kolbe et al., 2005). The software vendor VicoSoftware (Trimble, 2013; VicoSoftware, 2005) was the first to apply the concept in a similar fashion to BIM models.

In the AEC industry, the term Level of Development (LOD) was favoured over Level of Detail (LoD) as it represents the maturity, completeness, and reliability of the geometrical and semantical information provided by building elements (BIMForum, 2019). The LoD concept has then been adopted and refined by the American Institute of Architects (AIA) to become LOD (AIA, 2008). The AIA introduced a LOD definition that comprises five levels, starting from LOD 100 and reaching LOD 500. The BIMForum working group developed a new level, LOD 350, and published the Level of Development Specification based on the AIA definitions (BIMForum, 2019). At the same time, Trimble's Project Progression Planning (Trimble, 2013) was published and is widely used in practice.

Numerous countries, especially in Europe, have proposed different terms for their regions. In the UK, the Level of Definition (BSI, 2017) has been introduced. It consists of seven levels and introduces two components: Levels of model detail, which represents the graphical content of the models, and Levels of model information, which represents the semantic information. The Danish definition includes seven Information Levels that correspond roughly to the traditional project life-cycle stages (van Berlo & Bomhof, 2014). Similarly, in Germany, the MDG comprises 10 levels (010, 100, 200, 210, 300, 310, 320, 400, 510, 600) that also correspond to the project life-cycle stages (VBI, 2016). The Italian LOD definition adopts the BIMForm's specification while adjusting it to seven levels with letters in ascending order from LOD A – LOD G (PROGETTIAMOBIM, 2018). In Switzerland, the LOD concept is based on the BIMForum's definitions, but at the same time, its usage is assigned to project life-cycle stages (Maier, 2015).

Recently, a similar concept was introduced by the European Standardization Organization (CEN) (DIN, 2019), which defines the term Level of Information Needs comprising specifications for LOG and LOI for supporting a particular use-case.

### 5.2.4 Supporting the design process using LODs

As the LODs provide means for specifying and communicating which information is expected to be present at a specific time, they were used by numerous practitioners and researchers for defining the required information throughout the design phases (Abualdenien & Borrmann, 2019; Gigante-Barrera et al., 2018; Schneider-Marin & Abualdenien, 2019; Vilgertshofer & Borrmann, 2017). Abualdenien and Borrmann, 2019 developed a meta-model approach for specifying the design requirements of individual families using the LODs, incorporating the information uncertainty (Abualdenien & Borrmann, 2019). In the same context Gigante-Barrera et al., 2018 included the LODs as an indicator for the necessary information within Information Delivery Manuals (IDMs). Abou-Ibrahim and Hamzeh, 2016 developed a framework for applying lean design principles based on LODs (Abou-Ibrahim & Hamzeh, 2016). Additionally, Grytting et al., 2017 introduced a conceptual model of a LOD decision plan, based on a set of interviews and use-cases, to support design decisions (Grytting et al., 2017).

To support the decision-making process from the early design phases, Abualdenien et al., 2020 used the LODs to integrate the design process with energy simulations and structural analysis (Abualdenien et al., 2020). Additionally, Exner et al., 2019 proposed a LOD-based framework for comparing the different design variants and their detailing (Exner et al., 2019). To exchange design requests and issues between projects participants, Zahedi et al., 2019 proposed a communication protocol that leverages the LODs to describe design requirements (Zahedi et al., 2019). Finally, Abualdenien and Borrmann, 2020b developed multiple visualization techniques to depict the information uncertainty associated with the LODs throughout the design phases (Abualdenien & Borrmann, 2020b).

### 5.2.5 Analysis and validation of LOGs

The process of adopting a LOD specification in a particular country (or even internally in individual firms) requires a comprehensive analysis and understanding of which geometric and semantic information should be present at each LOD. However, practitioners have an inconsistent understanding of the information necessary at each LOD (Abualdenien & Borrmann, 2019; van Berlo & Bomhof, 2014). The main reason is that although the specification of semantics is usually simplified to a list of properties, systematically checking the geometric detailing is an unresolved task.

In this regard, Leite et al., 2011 evaluated the modeling effort associated with generating BIM models at different LoDs. The authors have shown the need for an increased modeling time, ranging from doubling the modeling effort to eleven folding it, to detail models further to reach a higher LoD (Leite et al., 2011). In comparison to our research, Leite et al., 2011 were referring to the overall building model or the combination of building elements while experimenting with the LoDs, whereas in this paper, the detailing and experiments are conducted per the individual families.

In the same context, van Berlo and Bomhof, 2014 has analyzed 35 building models (where each comprises multiple building elements), taking into account different ratios between volume, triangles, space areas, and the number of properties, in an attempt to find a relationship between the different LODs (van Berlo & Bomhof, 2014). However, the authors did not find any pattern for the increase of detailing across the LODs. The main reason for that is the inconsistencies and the different interpretations of the LOD specifications (Abualdenien & Borrmann, 2019; Bolpagni & Ciribini, 2016; Gigante-Barrera et al., 2018). While some approaches use the LOD concept for describing the maturity of the overall building model, others do so only for the individual element types. van Berlo and Bomhof, 2014 performed their experiments on the overall building models rather than the individual elements. By contrast, the LOD specifications provided by the AIA (AIA, 2008), BIMForum (BIMForum, 2019), and Trimble (Trimble, 2013) describe the geometric and semantic information of the individual elements rather than the overall building model. On a wider scale, Wong and Ellul, 2016 analyzed the geometry of 3D city models for fit-for-purpose by looking into the ratios between the number of buildings, geographic area, geometrical details, and disk size (Wong & Ellul, 2016).

In essence, there is a significant research gap resulting in a lack of computational methods that formally specify levels of geometry on the basis of a corresponding metric, and subsequently, apply this metric on concrete building models to assess the LOG they provide.

### 5.2.6 Ensemble-learners

The process of inferring generalized patterns from a training dataset (consisting of a set of instances where their classes are known) is described as inductive inference (Quinlan, 1996a). The simplest way to analyze a training dataset is to develop a classification system that consecutively splits the data (based on feature values) in a way that groups similar classes together as much as possible. It is important to note that the metric of similarity is not pre-defined but part of the solution-finding process. Given a set of $N$ instances where each belongs to one of $K$ classes, a classification system can construct a set of rules through training on the set of instances. This is precisely what a decision tree performs while following a particular route, yielding a specific result (Breiman, 2001).

A decision tree comprises a series of nodes (Boolean questions or tests), branches (results of the tests), and leaves (classification classes). Each node questions the data and splits it into two branches, eventually leading to a predicted class. In order to measure the quality of the split, two criteria are commonly used: *Information Gain*, which uses the entropy measure to split the data in a way that returns the most homogeneous branches, and the *Gini Index*, which represents the likelihood of classifying a new instance incorrectly (Raileanu & Stoffel, 2004). For a given training dataset $T$, the Gini Index can be expressed as Eq.5.1 (Raileanu & Stoffel, 2004):

$$\sum \sum_{j \neq i} (f(C_i, T)/|T|)(f(C_j, T)/|T|) \tag{5.1}$$

Where $f(C_i, T)/|T|$ is the probability that a specific element belongs to class $C_i$.

Real-world data is imperfect and includes noise arising from misclassifications or inaccurate measurements. Modeling such data using one decision tree results in the generation of a long tree, which is, in this case, overfitted to the selected dataset. This is mainly because a decision tree is based on a greedy model, meaning it tries to find the most optimal decision at each step and does not consider the global optimum. Therefore, smaller trees are preferable, as they are less prone to overfitting (Breiman, 2001), which imposes a trade-off between developing a generalized model versus its accuracy. To overcome this limitation, researchers have invented the concept of ensemble learners, which will be discussed in the next subsections.

#### Random Forest (RF)

Random forests fall into a broader category called ensemble learners (Breiman, 2001), which generate multiple weak models and then aggregates their classifications to produce better results. As illustrated in Figure 5.1, a random forest model constructs a set of decision trees and aggregates the unweighted average of their classifications (a.k.a. votes) to determine the final prediction (Breiman, 2001).

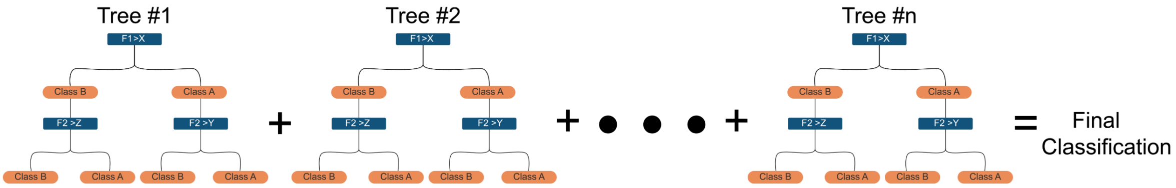


Figure 5.1: Random Forest (RF) Schematic Representation: it consists of multiple decision trees, where the unweighted average of their classifications is calculated to decide on the final classification.

Using methods such as bootstrap aggregating or bagging (Breiman, 1996; Quinlan, 1996b), each of the decision trees within a forest is built using a randomly selected set of features and instances. Such methods manipulate the training data to generate diverse classifiers (which makes it hard to overfit). Additionally, these methods support parallelization, making it possible to construct and train the trees within a random forest model independently from each other, which is relatively faster than other models, such as boosting, which will be discussed in more detail in the next section.

### eXtreme Gradient Boosting (XGBoost)

From the same category as random forests, gradient boosting is an ensemble learner (combining the result of multiple weak models) (Freund & Schapire, 1995). The main difference between a random forest and boosting is that the former constructs decision trees independently, simultaneously, and uses an unweighted average of votes, while the latter iteratively builds and evaluates individual trees (which are usually short, a.k.a. decision stumps) and tries to learn from wrongly classified observations by adding a higher weight on them in the subsequently built trees (Freund & Schapire, 1995; Quinlan, 1996b) (the concept is illustrated in Figure 5.2). An increased weight represents an increased contribution of a class or an instance to the loss function. Then, as boosting cannot be parallelized (the weights used for each tree are dependent on the results of the previously constructed tree), it takes much longer to train than a random forest.

Numerous popular boosting-based algorithms have recently been developed, including Adaptive Boosting (AdaBoost) (Freund & Schapire, 1995) and eXtreme Gradient Boosting (XGBoost) (T. Chen & Guestrin, 2016). AdaBoost is follows the weighting approach discussed previously. However, XGBoost (the currently dominant algorithm (Dong et al., 2020; Ke et al., 2017)) defines a loss function, and while iteratively constructing new trees, it focuses on minimizing that loss function. XGBoost can be expressed as Eq.5.2 (T. Chen & Guestrin, 2016):

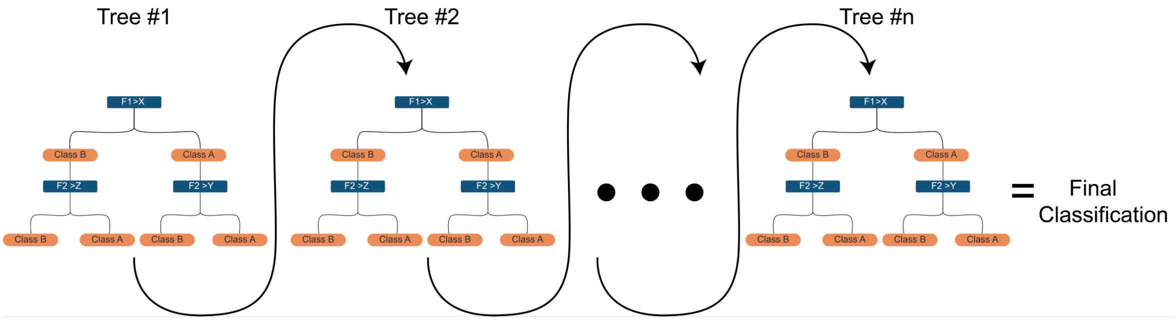


Figure 5.2: eXtreme Gradient Boosting (XGBoost) Schematic Representation: it builds decision trees consecutively and tries to learn from wrongly classified observations by adding a higher weight on them in the subsequently built trees.

$$\hat{y}_i = \phi(x_i) = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{5.2}$$

Where $f_k$ represents an independent decision tree, $F$ is the space of trees, $x_i$ represents the independent variables, and $K$ are the additive functions. The goal is to minimize Eq.5.3 (T. Chen & Guestrin, 2016):

$$\mathcal{L}(\phi) = \sum_{i} l(\hat{y}_i, y_i) + \sum_{k} \Omega(f_k) \tag{5.3}$$

Where $\mathcal{L}$ is a loss function and $\Omega$ is a penalty value representing the complexity of the model by taking into account the number and score of leaves.

## 5.3 Methodology

The hypothesis of this paper is that the detailing of the individual elements at the different LOGs can be correlated with multiple geometric features. These features form the basis for formally assessing the geometric complexity of a given model. Thereby, the LOG of building elements can be identified through analyzing the detailing patterns of the extracted features across the LOGs.

As depicted in Figure 5.3, the proposed approach consists of two main steps. First, a LOG dataset is modeled according to the most common LOD specifications (described in detail in Section 5.3.1). The dataset generation took into account modeling different kinds of building elements as well as additional cases for including openings and reinforcement. Afterwards, multiple geometry processing algorithms are performed to extract the most prominent features representing the detailing of each building element. The result is a dataset of geometric features for diverse building elements at the LOGs 200 – 400.

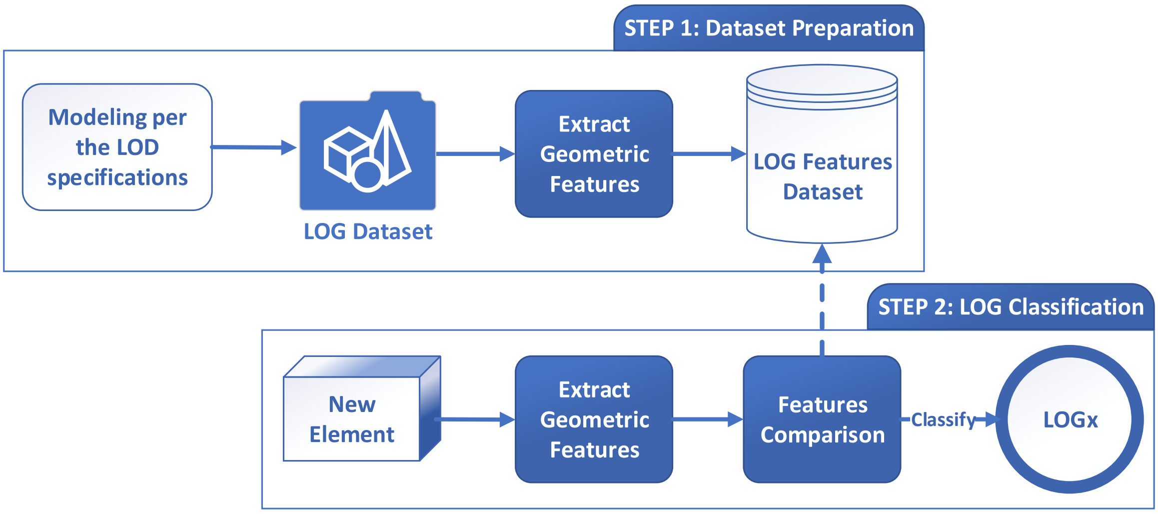


Figure 5.3: The developed approach presented in this paper consists of two main steps: (1) generation of a dataset containing the geometric features representing the complexity of the individual building elements across the LOGs, and (2) classification of new building elements in which their LOG is unknown.

The second step describes the process of classifying the LOG of a given element that was not part of the training set (a new element). The geometric features of the new element are extracted in a similar way to the dataset generation. The individual features are then compared to the features available in the dataset to classify the LOG of the new element. This step represents the actual application of the developed approach for classifying the elements of a BIM model provided by the end-user. The complete framework is discussed in detail in the next subsections.

### 5.3.1 Modeling according to the LOD specifications

In this study, the BIMForum's LOD specification (BIMForum, 2019) and Trimble's Project Progression Planning (Trimble, 2013) were comprehensively reviewed and followed during the modeling of different families on multiple LODs. In addition to the authors' practical experience, the combination of the mentioned specifications was followed. Although the BIMForum's definitions are descriptive for many building elements, they are, in many cases, vague in describing the progression of the geometric detailing. Despite the fact that the specification is prepared in a way that visualizes the newly added parts in every LOD, the graphical illustrations for many elements are missing or inconsistent and ambiguous. For example, when modeling a staircase, information regarding the riser count and height should be available starting from LOD 300 (per the text description). However, the graphical illustration at LOD 200 already includes these information. Whereas, in Trimble's specification, for this particular case, the graphical illustration reflects the available information clearly. Therefore, it was necessary to use both specifications.

According to the LOD specifications, LOG 100 (conceptual model) is limited to a generic representation of the building elements, meaning no shape information or geometric representation is provided. At LOG 200 (approximate geometry), elements are represented by generic placeholders depicting the overall area reserved by their volume. At LOG 300 (precise geometry), the elements' main shape is refined, showing the fundamental detailing required for describing the element type. Next, at LOG 350 (construction documentation), any necessary parts for depicting the connections with other elements that are attached or connected are additionally modeled. Modeling these parts, like supports and connections, is crucial for the coordination with different domain experts. Finally, at LOG 400, elements and their connections are fully detailed, providing the accuracy required for fabrication, assembly, and installation. LOG 500 represents the field verified model state, but in terms of design and detailing, it is the same as LOG 400.

The modeling process followed to generate the dataset has focused on the LOGs 200 – 400. To have confidence in how to model the families, those which are associated with both textual description and visual illustration were modeled first. Afterwards, we expanded the dataset size by making use of the available BIM objects libraries [2]. In this regard, the families were downloaded and adjusted to fit the requirements of the different LOGs. In total, the modeled dataset includes 408 objects (102 families at four LOGs). A complete list of the modeled family names is provided in Figure 5.5. Figure 5.4 shows a selected set of families on multiple LOGs from the modeled dataset. Additionally, the part of the dataset for which the authors possess full ownership is provided as open data to the public [3].

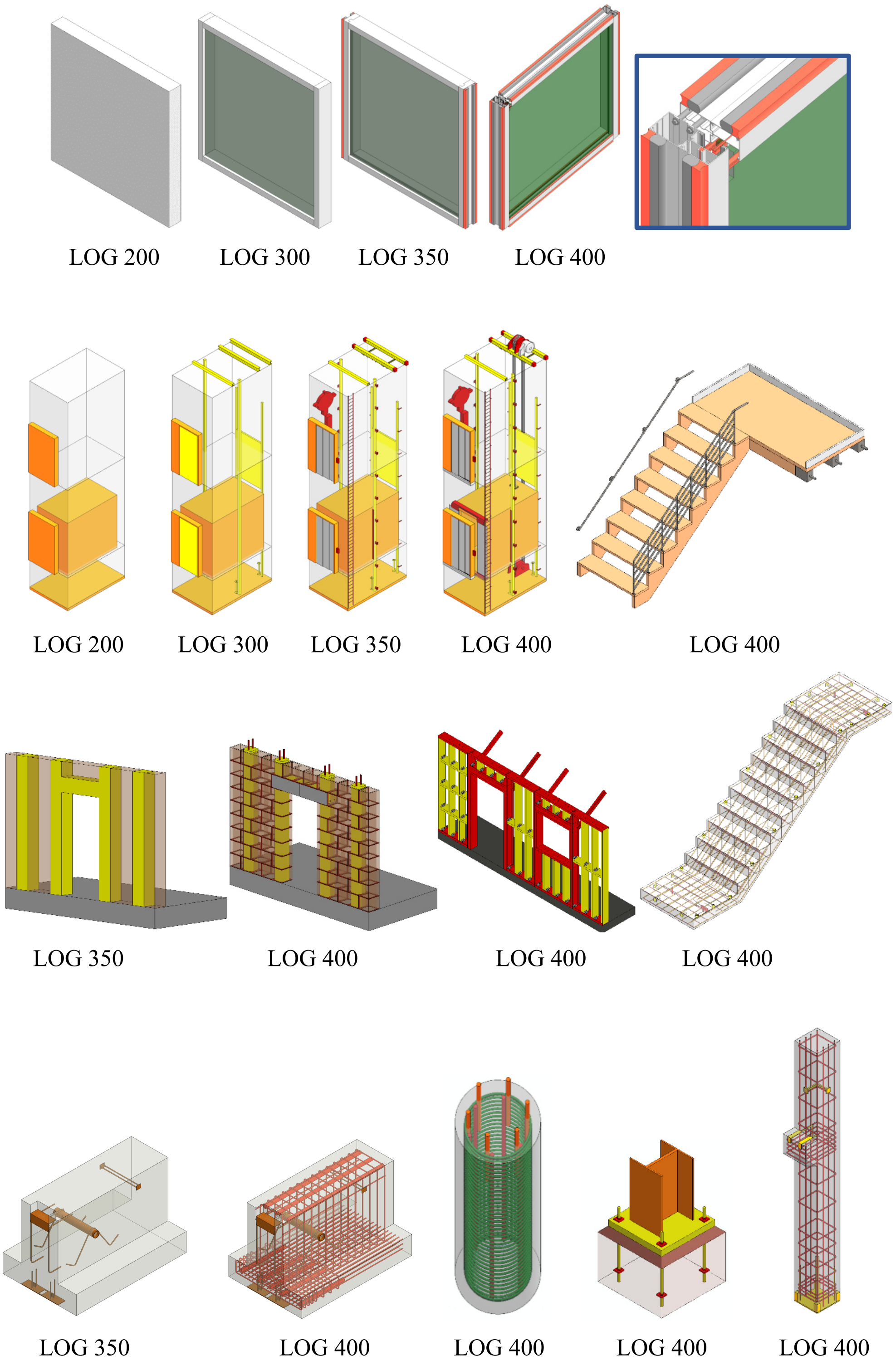
While modeling the different families, the necessary time for modeling each LOG of each family was measured. The modeling process was conducted by two domain experts, who were responsible for the measurements. The experts are trained designers who work in an architectural office. They have a clear understanding of the LOD concept and sufficient experience in modeling families. The time starts after discussing and deciding which geometric features should be included in each family to fulfill the descriptions provided by the LOD specifications and ends after modeling all features. The aim is to investigate the necessary modeling effort associated with detailing the families from one LOG to the subsequent one.

Figure 5.6 presents the resultant time measurements from modeling the entire dataset. The figure shows the minimum, maximum, and average necessary time (in minutes) for modeling the building elements at each LOG. Modeling elements at LOG 200 required between two and 40 minutes. Detailing the elements further to LOG 300 utilized two to threefold of the time

---

[2] www.bimobject.com, www.nationalbimlibrary.com, market.bimsmith.com, www.revitcity.com, www.familit.com, www.arcat.com

[3] http://u.pc.cd/6fXctalK



LOG 200 LOG 300 LOG 350 LOG 400

LOG 200 LOG 300 LOG 350 LOG 400 LOG 400

LOG 350 LOG 400 LOG 400 LOG 400

LOG 350 LOG 400 LOG 400 LOG 400 LOG 400

Figure 5.4: LOG Dataset: A small selection of the building elements at different LOGs.

at LOG 200. That is mainly because modeling the outer shape at LOG 300 needs to describe the overall shape's dimensions precisely. When modeling connections with the surrounding

1. Ramp
2. Portable Water Storage Tank
3. Wood Stair (variant 1)
4. Escalator (variant 1)
5. Domestic Water Equipment
6. Fireplace (variant 1)
7. Escalator (variant 2)
8. Domestic Water piping
9. Fireplace (variant 2)
10. Rectangular Pier (Reinforced Concrete)
11. Plumbing Fixtures
12. Sink
13. Cylindrical Pier (Reinforced Concrete)
14. Stormwater Drainage Piping
15. Balcony Railing
16. Helical Pile
17. Stormwater Drains
18. Heating Pipe Fittings
19. Exterior Wall (Brick)
20. Fuel Storage Tanks
21. Concrete Batch Plant
22. Exterior Wall (Wood)
23. Heat Generation
24. Steel Base Plate
25. Masonry Framing (variant 1)
26. Supply Air
27. Fire mains
28. Masonry Framing (variant 2)
29. Water-Based Fire Suppression
30. Bathtub
31. Cold-Form Metal Framing
32. Packaged Generator Assembly
33. Chimney (Brick)
34. Precast Structural Inverted T Beam (Concrete)
35. Electrical Service Entrance
36. Tube Light
37. Roof (Clay)
38. Power Distribution
39. Cables System
40. Roof (Wood Shingles)
41. Lighting Fixtures
42. Toilet
43. Floor Structural Frame
44. Metal Building Systems - Primary Framing
45. Office Desk (variant 1)
46. Multilayers Slab (Reinforced Concrete)
47. Metal Building Systems - Secondary Framing
48. Office Desk (variant 2)
49. Precast Structural Inverted T Beam (Concrete)
50. Electric Distribution System
51. Laundry Sink
52. Precast Structural Column (Concrete)
53. Concrete Column Formwork
54. Office Chair (variant 1)
55. Steel Framing Column
56. Concrete Slab Formwork
57. Office Chair (variant 2)
58. Steel Framing Beam
59. Highway Bridges Precast Structural Girder (Concrete)
60. Double Electrical Door
61. Steel Framing Bracing Rods
62. Slide Window
63. Bed Side Drawer
64. Steel Joists
65. Architectural Column
66. Water Boiler
67. Wood Floor Trusses
68. Open Balcony
69. Roof Ladder
70. Precast Structural Double Tee (Concrete)
71. Covered Balcony
72. Sofa
73. Precast Structural Stairs (Concrete, variant 1)
74. Curtain Wall
75. Sliding door
76. Precast Structural Stairs (Concrete, variant 2)
77. Garage Door
78. HVAC System
79. Metal Walkways
80. Lamp (variant 1)
81. Trefoil Round Arch Window
82. Precast Wall Construction (Concrete)
83. Lamp (variant 2)
84. Trefoil Round Arch Door
85. Exterior Window (variant 1)
86. Window Shading
87. Ventilation System
88. Exterior Window (variant 2)
89. Sliding door
90. Wardrobe
91. Interior Door (variant 1)
92. Rotate Door (variant 1)
93. Inline Pump
94. Exterior Door (variant 1)
95. Rotate Door (variant 2)
96. Roof Hatch
97. Elevator (variant 1)
98. Spiral Metal Stair
99. Tube System
100. Elevator (variant 2)
101. Reinforced Wall
102. Wood Stair (variant 2)

Figure 5.5: Families Dataset: list of the modeled family names.

elements at LOG 350, the necessary time increases to be between four and seven-fold the time at LOG 200. Finally, as LOG 400 demands fabrication-level detailing, the necessary time doubles, reaching eight to 15-fold compared to LOG 200. When comparing the increase in the time between subsequent LOGs, we observe that it ranges between 1.88 – 2.80-fold.

### 5.3.2 Analysis & extraction of LOG features

Typically, shapes having more numerous or smaller features can be viewed as more detailed. The challenge in identifying the LOG through analyzing the geometric features lies in deducing a standard pattern (a metric) that describes the individual LOGs. The simplest geometric metrics can be based on the total number of vertices, faces, and edges. However, an increased number of these features does not necessarily mean an increased detailing or higher LOG. For example, a window at LOG 200 (rectangular shape) consists of 30 vertices, 16 faces, and 78 edges, while a cylindrical column or heating tank at LOG 200 could be formed by 2,358 vertices, 4,268 faces, and 13,244 edges.

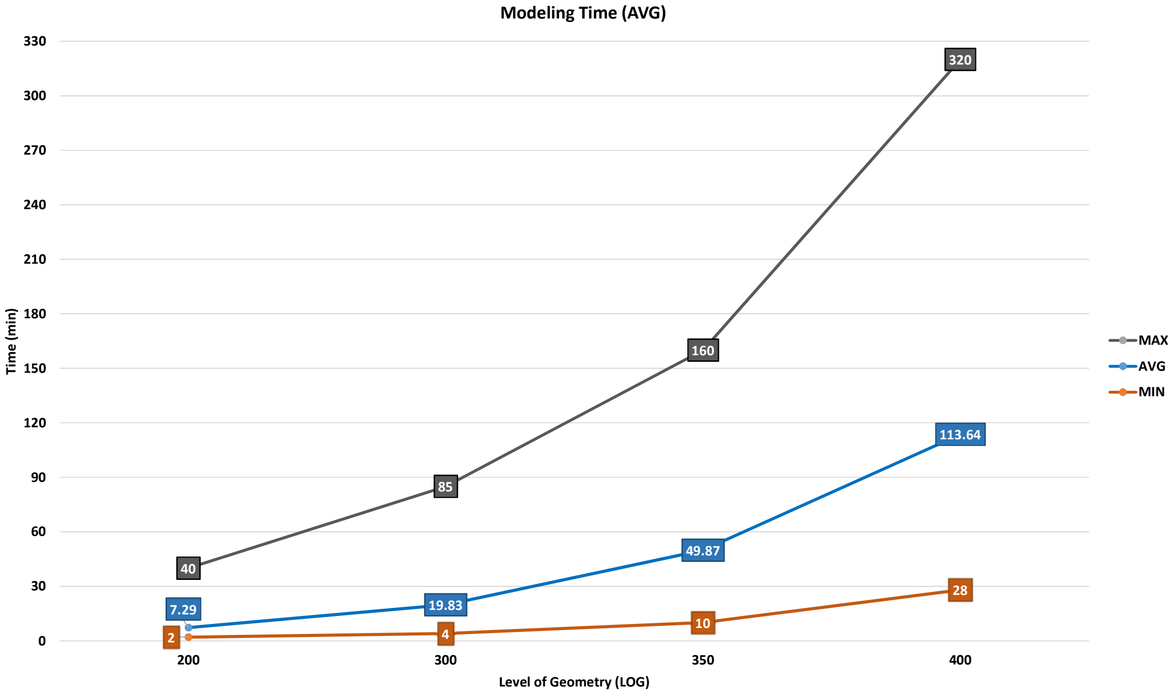


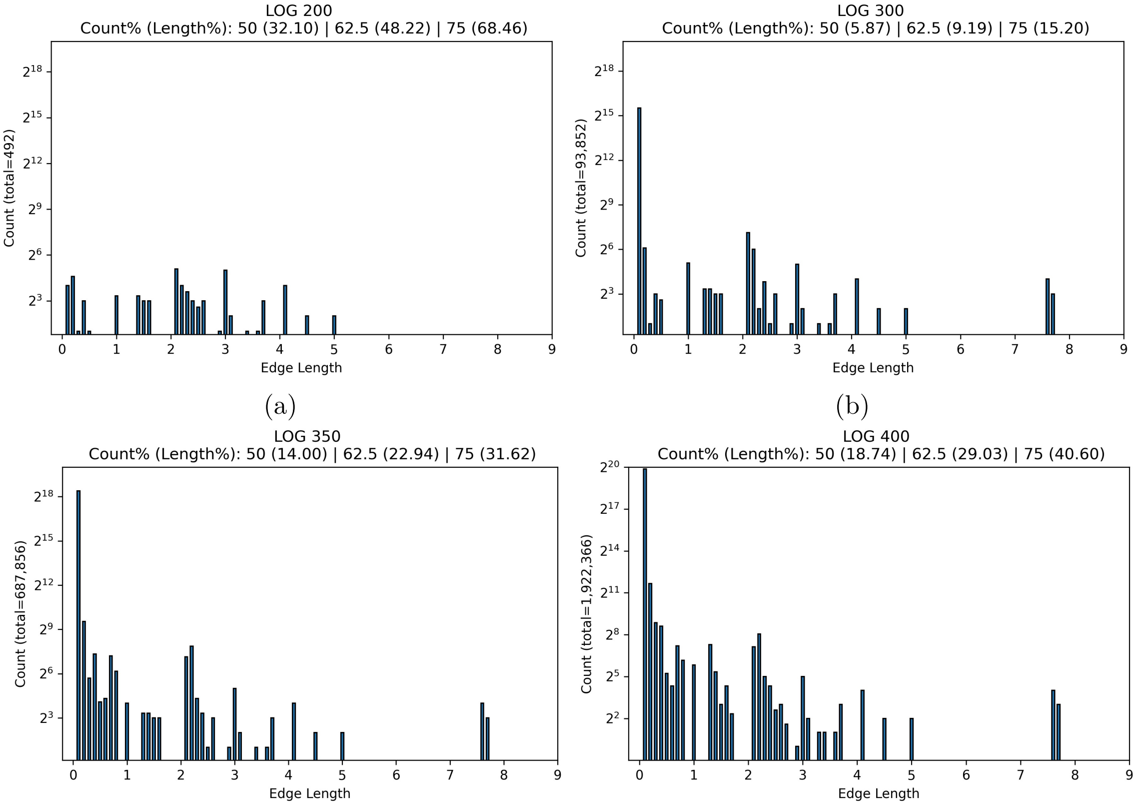
Figure 5.6: LOG Dataset: an investigation of the necessary modeling time when detailing building elements from LOG 200 – 400. The figure shows the minimum, maximum, and average time (in minutes).

Thus, the sole consideration of vertices, faces, and edges does not provide a suitable metric. To measure the geometric detailing (i.e., LOG) of elements, the set of selected features needs to be capable of representing the geometric detailing of elements taking into account the overall shape complexity. Hence, in this paper, we propose combining the extracted results of multiple geometric features to observe various aspects of the shape's detailing. In total, we investigated the effect of detailing across the LOGs through three main aspects, which are discussed in detail in the next subsections.

### Basic features: Vertices, faces, and edges

Vertices, faces, and edges represent the most fundamental ingredients for describing the detailing of any shape. In this regard, the ratio of vertices to faces is capable of providing an insight into the overall shape form. Based on our experiments, a shape with only rectangular parts always has a ratio of two. When adding more complex parts, like screws or reinforcement, the ratio is substantially reduced.

The count and length of edges can also provide a strong description of the shape resolution and complexity (D. Zhang et al., 1998). When a shape comprises a low number of edges with a similar length, then the overall shape is basic and does not include high details. On the other hand, when the majority of edges are relatively short, then the shape comprises numerous complex parts. In this regard, we measure the mean of the edges' length as well as the total length of 50%, 62.5%, and 75% of the edges, after ordering them in ascending order to prioritize the short edges, as they are an indicator for the increase in detailing. Then, the measured lengths' ratio to the total edges' length is calculated at the different LOGs. For example, for a particular element, the length of 75% of the edges could represent 60% of the length of the total edges at LOG 200, while the length of 75% of the edges represents 10% at LOG 400.



(a) (b)

(c) (d)

Figure 5.7: Basic Geometric Features: count and percentage of edges' lengths across the LOGs for the elevator element depicted in Figure 5.4. Here, the mean of the edges' length as well as the total length of 50%, 62.5%, and 75% of the edges after ordering them in an ascending order were measured. Prioritizing the short edges provides an indicator for the increase in detailing.

Figure 5.7 presents the different lengths of edges on the x-axis and the total count of each length on the y-axis for the elevator element depicted in Figure 5.4. The edges' lengths (on the x-axis) were rounded to the first decimal place and grouped. The edges' counts (on the y-axis) are shown on a logarithmic scale (to the base of two) to highlight the different lengths. At LOG 200, the total count of edges is 492. Here, we can notice that the edges' counts are relatively comparable across the lengths of zero to five and mostly dense in the middle. At LOG 300, the total count of edges became 93,852 (19-fold the count at LOG 200). Although numerous relatively long edges were added, the majority of the edges are short. The increase in the count and length of edges across the LOGs 350 and 400 follows a similar pattern.

Additionally, Figure 5.7 lists the statistical percentages of the edges' counts and lengths. Such statistics highlight the overall geometrical detailing. In more detail, at LOG 200, the shape is expected to be an approximation, represented by bounding boxes. Therefore, the length of 62.5% and the length of 75% of the edges equals 48.22% and 68.46% of the overall length, respectively, which are relatively high. Whereas, at LOG 300, the shape is refined further to represent a precise shape, resulting in numerous additional short edges. Accordingly, the length of the edges is much shorter than at LOG 200. Next, at LOGs 350 and 400, connections and additional geometric details (for example, for fabrication or even vendor-specific details) are modeled, gradually increasing the length of the statistical percentages.

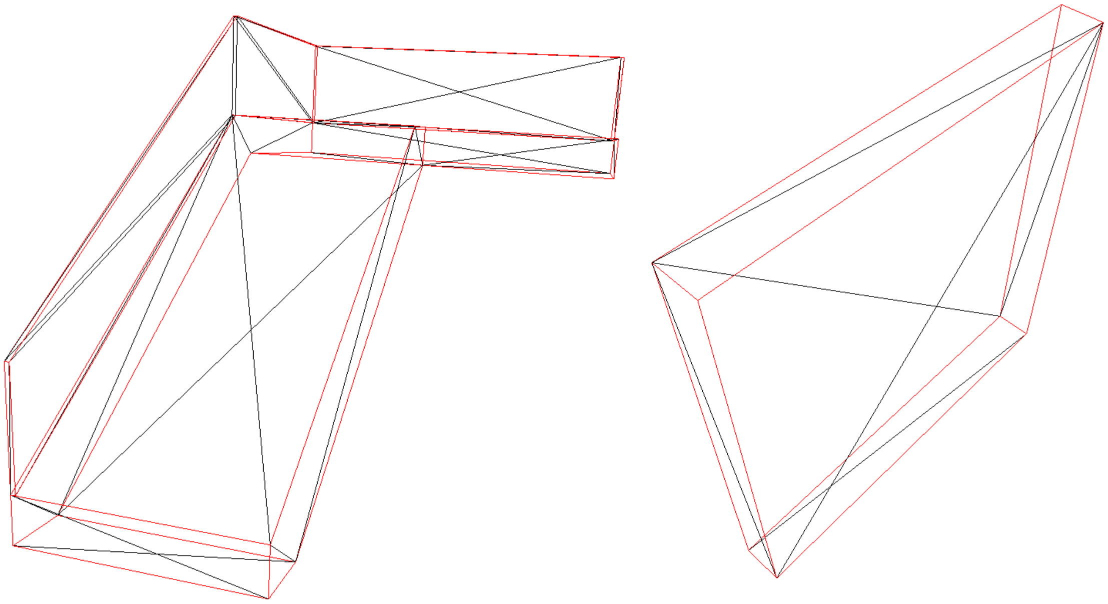
### Sharp edges & feature lines

Feature lines identify the most prominent surface characteristics of a geometric shape (HILDEBRANDT et al., 2005). The extraction of these lines has been intensively researched in various domains, including the analysis of medical data (MONGA et al., 1995) and point clouds (WEBER et al., 2010). The fundamental description of feature lines is the local extrema of principal curvatures along with corresponding principal directions (HILDEBRANDT et al., 2005). In other words, the angle between the two normal vectors of adjacent triangles is measured, and when the angle is sharp (the surface curvature is changing), then the edge is considered as a feature edge. Finally, the detected edges form the shape's feature lines.

We extract and count the sharp edges as well as the number of surface patches bound by these edges. Figure 5.8 shows a stair and a window at LOG 200. Here the sharp edges are marked with a red color.

### Diameter-based segmentation

In this approach, the shape is segmented into smaller meaningful pieces based on the change in its diameter (SHAPIRA et al., 2008). The segmentation is based on measuring the Shape-Diameter Function (SDF) at every point, where the change of an SDF value from a point to its neighbors determines whether there is a new segment. Let $M$ be a triangulated mesh surface of any building element. The SDF is defined as the scalar function on the surface $fv : M \to \mathbb{R}$, representing the diameter at every neighbor point $p \in M$. The SDF provides an effective link between the object's volume to its surface. The algorithm provided by SHAPIRA et al., 2008(SHAPIRA et al., 2008) applies clustering on the facets according to their corresponding SDF values. Afterwards, the dihedral-angle and concavity of the surfaces is taken into account to produce the final segments.



(a) (b)

Figure 5.8: Sharp Edges: an example of a stair and window at LOG 200. The edges marked with red represent the extracted sharp edges.

This kind of segmentation provides additional insights into the complexity of the parts that are forming the building element. Therefore, we count the segments, measure their area, and evaluate their shape (flat surfaces, cubic, or cylindrical). Segments with similar shapes are grouped, and the ratio of their count and area is used to characterize the form and complexity of the overall shape. For example, a window at LOG 200 comprises few surfaces and cubic segments. Whereas, in the case of a tube system at LOG 200, it comprises few surfaces and cylindrical segments. Additionally, at LOG 400, numerous smaller segments with diverse shapes are typically added. Figure 5.9 shows two examples, highlighting the individual segments.

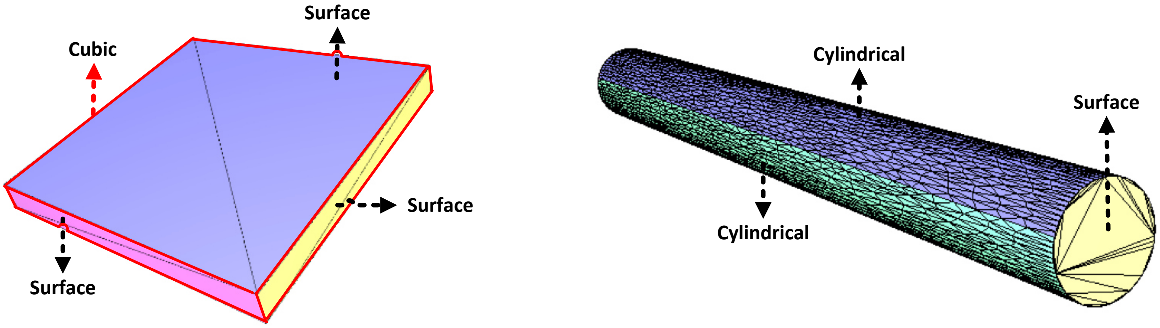
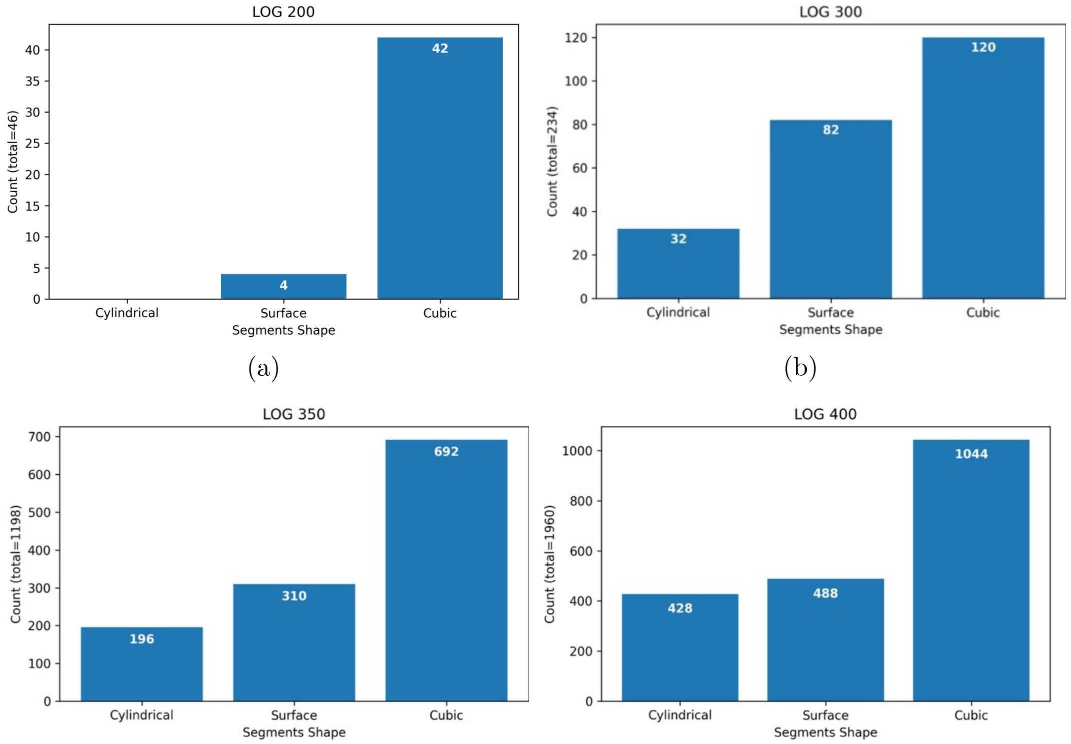


Figure 5.9: Diameter-based Segmentation: two examples highlighting the results of segmenting the building elements. The colors here represent the individual visible segments.



(a) (b)

(c) (d)

Figure 5.10: Diameter-based Segmentation: statistical analysis of the extracted segments based on their shape (surface, cubic, and cylindrical). These calculations were extracted from the elevator element depicted in Figure 5.4. The x-axis lists the segments' shape and the y-axis shows the total count of segments.

To highlight the benefit of counting and grouping the shape of the extracted segments, Figure 5.10 shows the segments' shapes on the x-axis and the total count of segments on the y-axis for the elevator element depicted in Figure 5.4. Besides increasing the total number of segments, these statistical calculations provide additional insights into what kind of detailing was added at each LOG. Additionally, this information facilitates identifying the shape characteristics. For example, based on our evaluations, when the count of the cylindrical segments is low and represents more than ∼50% of the overall area, the overall shape has a high probability of having a cylindrical overall shape (a pipe, for example). Moreover, rectangular and complex shapes (such as a window and a stair) at the LOGs 350 and 400 are composed of a high number of cylindrical segments representing less than ∼40% of the overall area, which indicates the presence of screws and additional detailing parts. When reinforcement is modeled, then the number of cylindrical segments is relatively high (∼50 – 80%), while their aggregated area is less than ∼40% of the overall area.

### Features dataset

The discussed geometric features above were extracted for the LOG dataset presented in Section 3.1. Additionally, multiple ratios were calculated to capture any positive or negative correlations among the features, including the average area per surface patch and per segment, as well as the average number of vertices per face, patch, and segment. Finally, the extracted features were normalized to make the features correspond to the elements' geometric complexity regardless of their total area or total length of edges.

To get an overview of the degree of association between the extracted features, a pair-wise Pearson correlation coefficient (PCC) (Pearson, 1896) was calculated. PCC measures the level of linear correlation between two variables. Accordingly, these coefficients are leveraged during the ensemble models' training to filter and optimize which features are selected for the training process. The features that prove a linear correlation (0.8 – 1 PCC) are considered the first candidates to be dropped to simplify the vector representation of each element. Additionally, the PCCs were combined with the features' importance (shown in Figure 5.12) to decide which features could be dropped. Such filtering is important when training tree-based models since unimportant features could construct weak trees that could then affect the model's accuracy. The features dataset included 22 features per element before filtering and 16 features after filtering. Table 5.1 depicts sample features of two building elements, a *Brick Wall* and a cylindrical *Reinforced Concrete Pier*, across the LOGs.

## 5.4 Classification of LOG

The analysis of the extracted geometric features indicated multiple patterns that are helpful in identifying the LOG. Identifying which class (i.e., LOG) an observation (i.e., the features representation of an element) belongs to is a classification problem. The manual classification of the LOG from the extracted features is an unfeasible task due to the large number of features and the heterogeneity of the different families. Hence, in this paper, we propose classifying the LOG of building elements using RF and XGboost, tree-based ensemble-learning models. Such models are popular nonlinear predictive models. In the following, we compare the approaches regarding their performance for the problem at hand.

### 5.4.1 Models training setup

The features dataset presented in Section 3.2.4 was split into training and testing sets with a ratio of 80% (326 elements) and 20% (82 elements), respectively. Splitting the dataset involved taking into account the different classes (LOGs 200, 300, 350, and 400) and the

Table 5.1: Features Dataset: an example of the selected features for training the ensemble models. The examples shown here belong to a *Brick Wall* and a *Reinforced Concrete Pier* across the LOGs.

| **LOG** | **Vertices / Faces** | **Vertices / Patches** | **Vertices / Segments** | **Faces / Patches** | **Area / Patches** | **Area / Segments** | **(%) SharpEdges** | **SharpEdges / Area** | **SharpEdges / Vertices** |
|---|---|---|---|---|---|---|---|---|---|
| **200** | 1.8 | 4.5 | 4.5 | 2.5 | 12.5 | 12.5 | 37.5 | 2.78 | 1 |
| **300** | 1.8 | 4.5 | 4.5 | 2.5 | 6.25 | 6.25 | 37.5 | 1.39 | 1 |
| **350** | 0.73 | 62.53 | 62.53 | 85.96 | 0.88 | 0.88 | 12.58 | 0.02 | 0.59 |
| **400** | 0.64 | 72.1 | 74.32 | 112.57 | 0.031 | 0.03 | 8.81 | 0.001 | 0.45 |
| **200** | 0.67 | 1.34 | 8 | 2 | 16.67 | 100 | 33.34 | 8.34 | 1.5 |
| **300** | 0.57 | 1.6 | 16 | 2.8 | 10 | 100 | 28.57 | 4.17 | 1.5 |
| **350** | 0.53 | 1.65 | 37.45 | 3.11 | 0.2 | 4.55 | 27.44 | 0.078 | 1.56 |
| **400** | 0.50 | 1.43 | 172.17 | 2.84 | 0.02 | 2.08 | 28.40 | 0.007 | 1.69 |

| **LOG** | **(%) Mean Edges Length** | **(%) 75% Edges Length** | **(%) Mean Segments Area** | **(%) 62.5% Segments Area** | **Cylindrical Segments Area** | **(%) Cylindrical Segments Count** | **(%) Cylindrical Segments Area** |
|---|---|---|---|---|---|---|---|
| **200** | 2.08 | 76.97 | 12.85 | 13.81 | 0 | 0 | 0 |
| **300** | 1.04 | 59.72 | 6.34 | 23.30 | 0 | 0 | 0 |
| **350** | 0.006 | 33.89 | 0.89 | 12.73 | 2.39 | 22.81 | 4.24 |
| **400** | 0.0001 | 35.92 | 0.03 | 12.024 | 25.74 | 19.41 | 25.65 |
| **200** | 5.56 | 55.23 | 100 | 100 | 3.78 | 100 | 100 |
| **300** | 2.38 | 38.48 | 100 | 100 | 3.93 | 100 | 100 |
| **350** | 0.04 | 18.77 | 4.56 | 3.74 | 4.41 | 100 | 100 |
| **400** | 0.004 | 3.25 | 2.09 | 12.46 | 5.75 | 100 | 100 |

type of families (to ensure a sufficient diversity). Some families at a specific LOG were only available in one of the sets.

Training an ensemble model involves tuning its hyperparameters to make its architecture more suitable for the used features. Such activity highly influences the model's accuracy and capability to generalize from individual observations. During the tuning of parameters, the best performing values are identified by searching through a range of values and evaluating all the possible combinations. In order to evaluate the performance of each set of parameters, we use a technique called k-fold cross-validation (K-foldCV) (OPENML.ORG, 2020). K-foldCV iteratively splits the training set into $k$ smaller sets. For each $k$ of the folds, the model is trained using $k-1$ while tested on the remaining part to evaluate the model's accuracy during training. Afterwards, the performance of the final model is measured by validating it against the test set. A too-large k-fold means that a low number of samples is validated in every iteration. Therefore, based on multiple experiments and given the diversity and size of the dataset used in this study, 5-fold cross-validations were performed to cover enough samples in every iteration.

The interpretation of the classifications resulting from the ensemble models is critical to understand the contribution of the individual features. Therefore, we use the Shapley Additive exPlanations (SHAP) (Lundberg & Lee, 2017) approach to assign each feature an importance value for each LOG class, providing detailed features' importance. SHAP is based on game theory (Štrumbelj & Kononenko, 2014) and local explanations (Ribeiro et al., 2016). Assume an ensemble model that is trained on all feature subsets $S \subseteq F$, where $F$ is the set of all features. The contribution of each feature $\phi_i$ on the model output is computed based on its marginal contribution compared to the rest of the features. The computation of SHAP values for a withheld feature can be represented as Eq.5.4 (Lundberg & Lee, 2017):

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} \left[ f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S) \right] \tag{5.4}$$

Where $x_S$ represents the values of the input features in the set $S$. Additionally, $f_{S \cup \{i\}}(x_{S \cup \{i\}})$ represents the model's classification when trained on all features without the withheld feature, and $f_S(x_S)$ is the model's classification when trained on the withheld feature (See (Lundberg & Lee, 2017) for more details).

### 5.4.2 RF model training

The hyperparameters that require tuning when training a RF model are shown in Table 5.2. The figure includes the range of values examined to find the best combination of parameters, which are also shown in a separate column (with *Selected* as a title). The selection of parameters was based on evaluating the model's accuracy while examining all the possible combinations. In addition to these parameters, the RF model is configured to bootstrap the training data while constructing the decision trees. Bootstrapping brings more variation to the training samples through shuffling and random filtering. Finally, the *Gini Index* is selected as the function to measure the quality of each split.

Table 5.2: RF model hyperparameters, including the search ranges and the selected values.

| **Parameter Name** | **Description** | **Search Range** | **Selected** |
|---|---|---|---|
| **n_estimators** | Number of trees in the forest | 20 - 1000 | 170 |
| **max_depth** | Maximum depth of the trees | 2 - 20 | 5 |
| **max_features** | Maximum number of features to consider when splitting the data at a particular node | 2 - 8 | 2 |
| **min_samples_leaf** | Minimum number of samples required to be at the leaf | 2 - 6 | 2 |
| **min_samples_split** | Minimum number of samples required to split a node | 2 - 6 | 2 |

To explore the structure of the decision trees comprising the RF model, Figure 5.11 shows a randomly selected tree out of the 170 trees in the forest. Each tree in the forest is built differently, where different order and feature types are used to split the dataset at every node. For each tree, the upper nodes split the data into smaller clusters, while the leaves classify the data into different LOG classes, specifying the confidence of every classification. The confidence of final classifications is represented using the *Gini Index* measure, where zero means 100% confidence and one means 100% uncertain. All nodes are colored according to their respective LOG class, and the color saturation reflects the classification confidence. For example, LOG 300 and 100% confidence (0.0 gini) is colored with dark green, while LOG 200 with 50% confidence (0.5 gini) is colored with light orange.

For the particular example shown in Figure 5.11, 211 elements out of the 326 were selected using boosting (other trees are constructed using a differently selected set of elements). In this example, the percentage of the cylindrical segments to all segments is the first feature splitting the training set to LOG 200 (75 elements) and LOG 350 (136 elements). Then, the total area ratio to all segments splits the data at the upper branches to LOG 300 (42 elements) and 200 (33 elements), whereas, the percentage of the cylindrical segments splits the elements at the lower branch to LOG 350 and 300. This process of splitting the dataset continues until reaching the leaves. At the leaves, a final classification is predicted for each group of elements. For instance, the first blue leaf at the top is 100% confident of classifying three of the samples as LOG 350, and the third green leaf under it is 81% confident of classifying ten samples as LOG 300. To provide additional insight on the RF model structure, multiple decision trees are provided online [4].

As shown in Figure 5.11, tree-based models base their classifications on the combination of different features. Typically, the features that contribute more to determining the final classification have higher importance. Such features are present multiple times within the constructed trees and split the data with high confidence. Figure 5.12 presents the importance of the features within the RF model (based on all 170 decision trees). A higher SHAP mean value implies higher importance of the corresponding features. Additionally, the bar of each feature is divided into four parts to quantify its influence on classifying each LOG. In this particular case, the top four important features involve the count and area of the extracted segments as well as the count and length of the detected sharp edges, including the resultant surface patches. On the other hand, the basic geometric features, like the count of vertices, faces, and edges, have considerably lower importance in contributing to the final classifications. Although the overall importance of the top five features is relatively higher than others, some of the other features are essential for differentiating a particular LOG from others, such as the cylindrical segments that are more present in LOG 400 than LOG 200, unless the overall shape is cylindrical.

---

[4] https://bit.ly/3jPJBeu

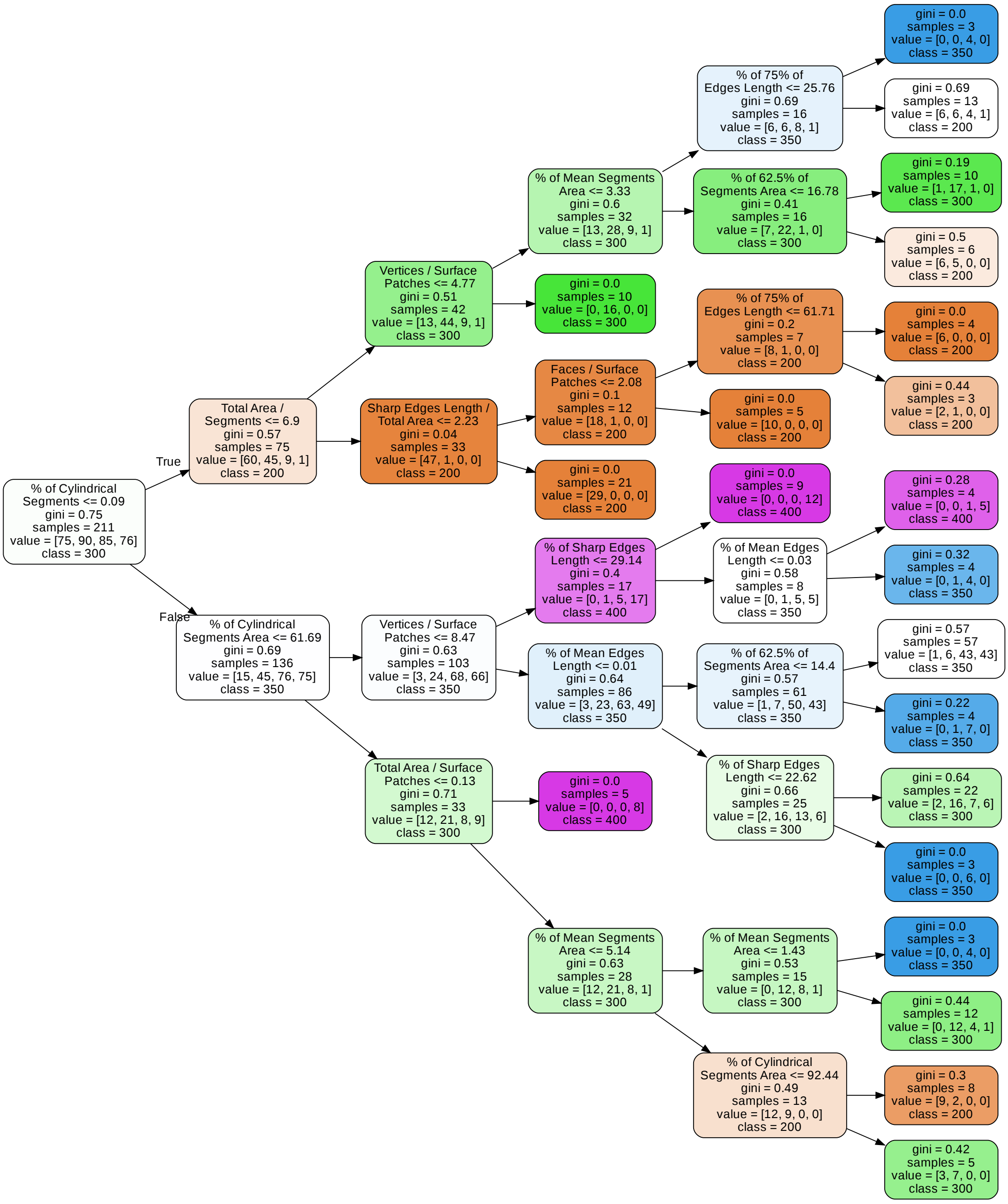


Figure 5.11: RF model: showing one decision tree out of 170. The nodes are colored according to their respective LOG class, and the color saturation reflects the classification confidence. For example, LOG 300 with 100% confidence is colored with dark green, while LOG 200 with 50% confidence is colored with light orange.

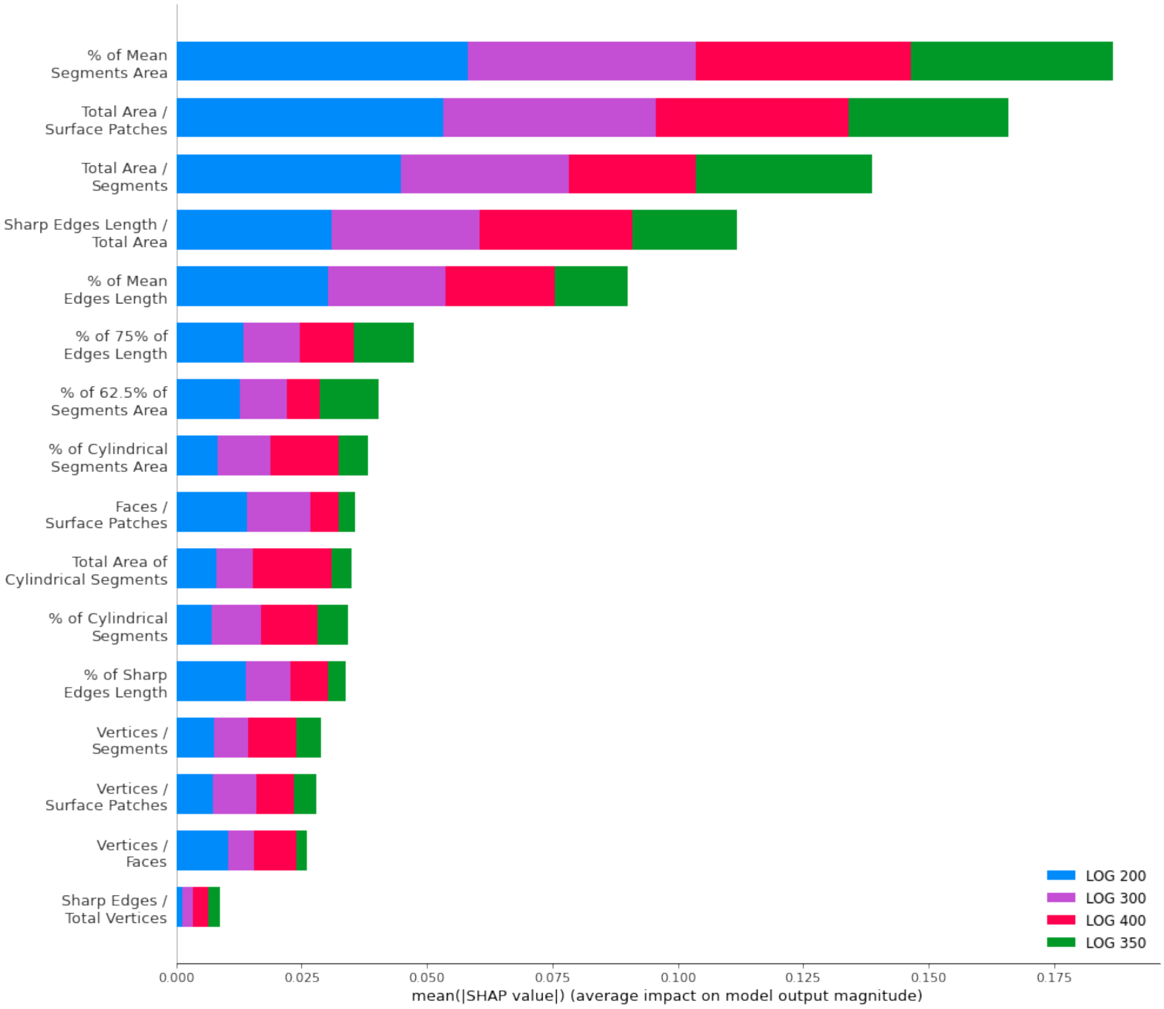


Figure 5.12: Features' importance for the RF model (all trees): using SHAP mean values as an indicator, a higher value implies higher importance of the corresponding feature.

### 5.4.3 XGBoost model training

For comparison, XGBoost was trained on the same dataset as it is one of the best performing algorithms for solving classification problems (Dong et al., 2020; Ke et al., 2017). The hyperparameters tuned during training are shown in Table 5.3, including 150 decision trees, a maximum depth of four, and multiple other parameters that influence the learning process. Similarly to the RF model, choosing the model parameters was based on evaluating a range of values.

Since the concept behind XGBoost is different from RF, the structure of the decision trees is also different. The trees are built subsequently and dependently rather than simultaneously and independently. Accordingly, the leaves of every branch within each tree produce a margin value (between -1–1), contributing to the overall classification probability of each class. This process is repeated for each class to represent the probability that a path through each tree

Table 5.3: XGBoost model hyperparameters, including the search ranges and the selected values.

| Parameter Name | Description | Search Range | Selected |
|---|---|---|---|
| **n_estimators** | Number of trees in the forest | 20 - 1000 | 150 |
| **max_depth** | Maximum depth of the trees | 2 - 20 | 4 |
| **learning_rate** | Step size shrinkage used in update to prevent overfitting | 0.001 - 1 | 0.001 |
| **gamma** | Minimum loss reduction required to make a further partition on a leaf node of the tree | 0.1 - 1 | 0.54 |
| **min_child_weight** | Minimum sum of instance weight (hessian) needed in a child. If the tree partition step results in a leaf node with the sum of instance weight less than min_child_weight, then the building process will give up further partitioning | 0.1 - 1 | 0.8 |
| **subsample** | Subsample ratio of the training instances | 0.1 - 1 | 0.6 |

classifies each class with a particular value. In the end, the sum of values from the subsequent trees provides the overall classes probabilities.

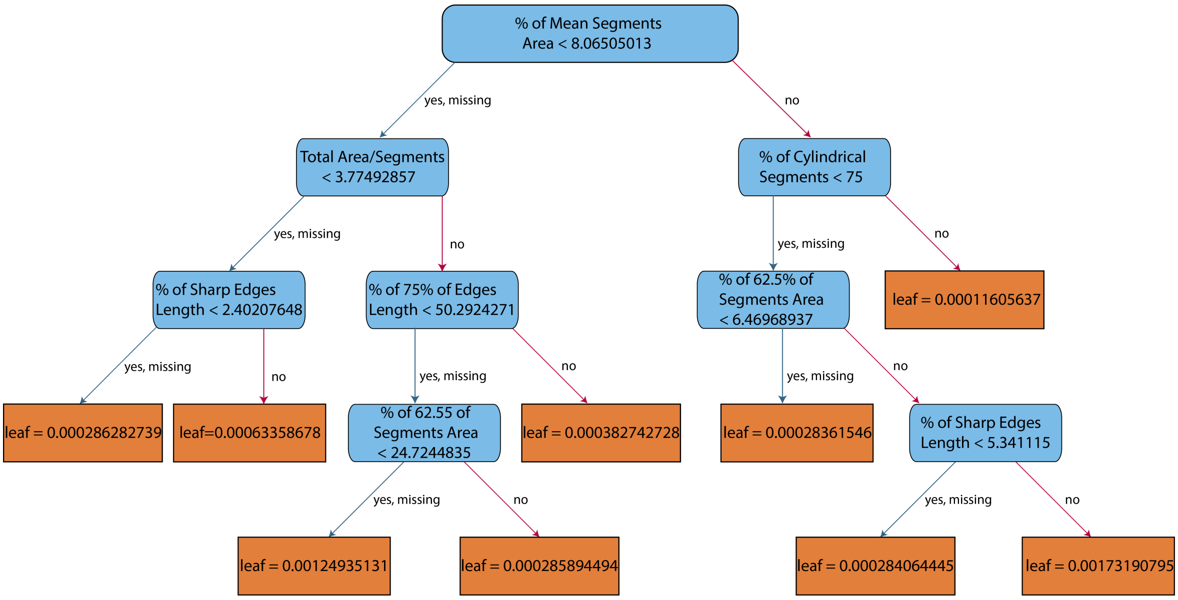


Figure 5.13: XGBoost model: showing one decision tree out of 150. The leaves of every branch within each tree produce a margin value (either a positive or negative number), which contributes to the overall classification of each element when combined with the previous and subsequent trees' results.

Figure 5.14 presents more insights on the overall features' importance for the XGBoost model. In this regard, the top four features are similar to the RF model, involving the count and area of the extracted segments as well as the count and length of the detected sharp edges, including the resultant surface patches. However, those features have different SHAP mean values as well as different proportions for the LOG classes. Additionally, the rest of the features are ordered differently from the RF model. In general, we observe that the XGBoost model relies on fewer features than the RF model to make the final classification.

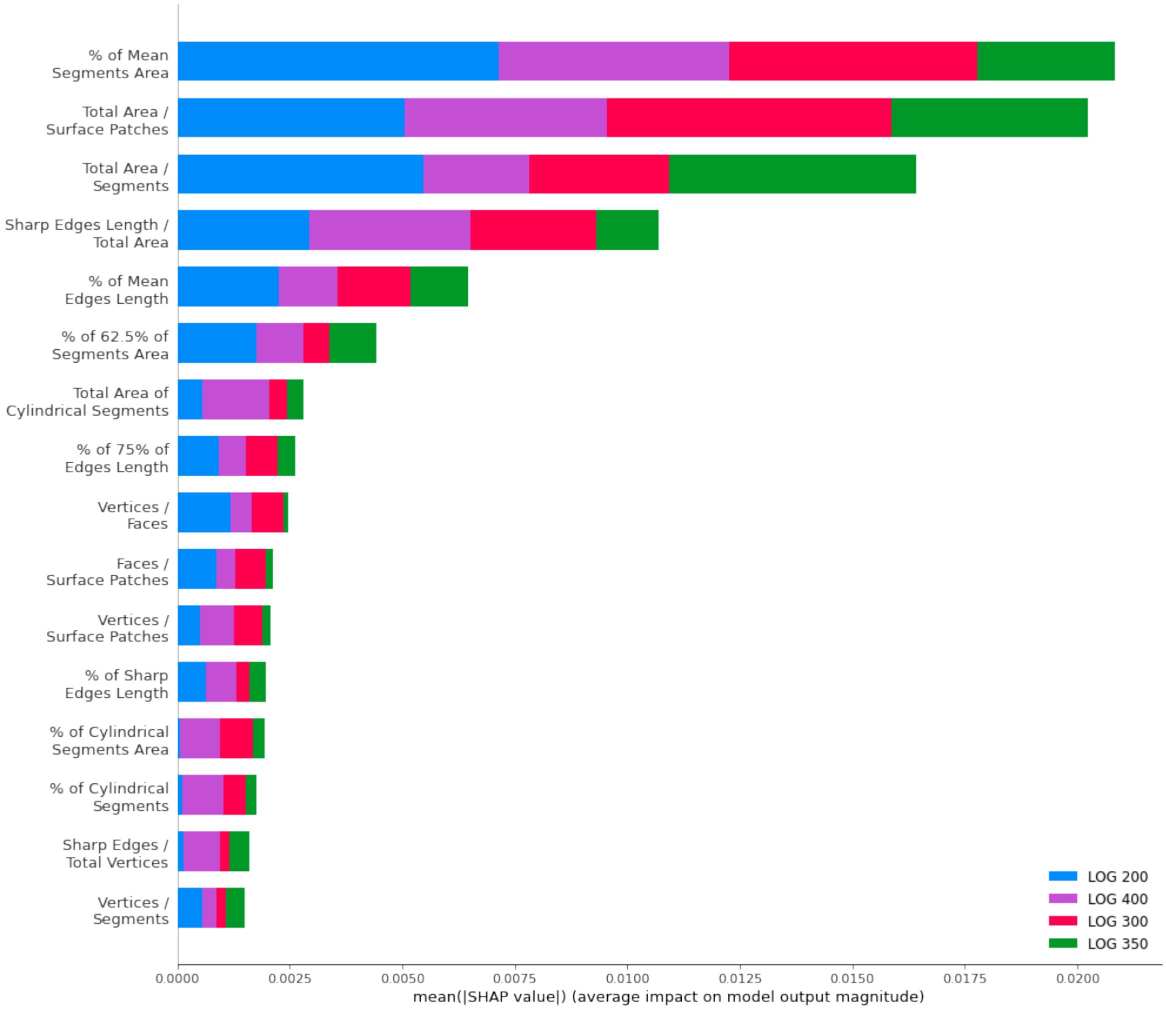


Figure 5.14: Features importance for the XGBoost model (all trees): using SHAP mean values as an indicator, a higher value implies higher importance of the corresponding feature.

### 5.4.4 Evaluation of RF and XGBoost models

The performance of the developed ensemble models was evaluated on a new set of elements (a test dataset that consists of 82 elements, entirely disjoint from the training set). The performance metrics are described as precision, recall, and F1-Score. Precision describes the model performance in positive predictions while considering false positives. Recall incorporates false negatives instead of false positives, and F1-score provides a balance between precision and recall. Table 5.4 presents the evaluation results of both models. Generally, the performance of both models in classifying the four LOGs is relatively close. The XGBoost outperforms RF in the precision, recall, and F1-score for all the LOGs. However, the F1-score's difference is not substantial for most LOGs (2 -– 3% for all except LOG 350, 5%).

Table 5.4: Evaluation results: performance results of the RF and XGBoost models on test data. The performance metrics are described as precision, recall, and F1-Score. Precision describes the model performance in positive predictions while considering false positives. Recall incorporates false negatives instead of false positives, and F1-score provides a balance between precision and recall.

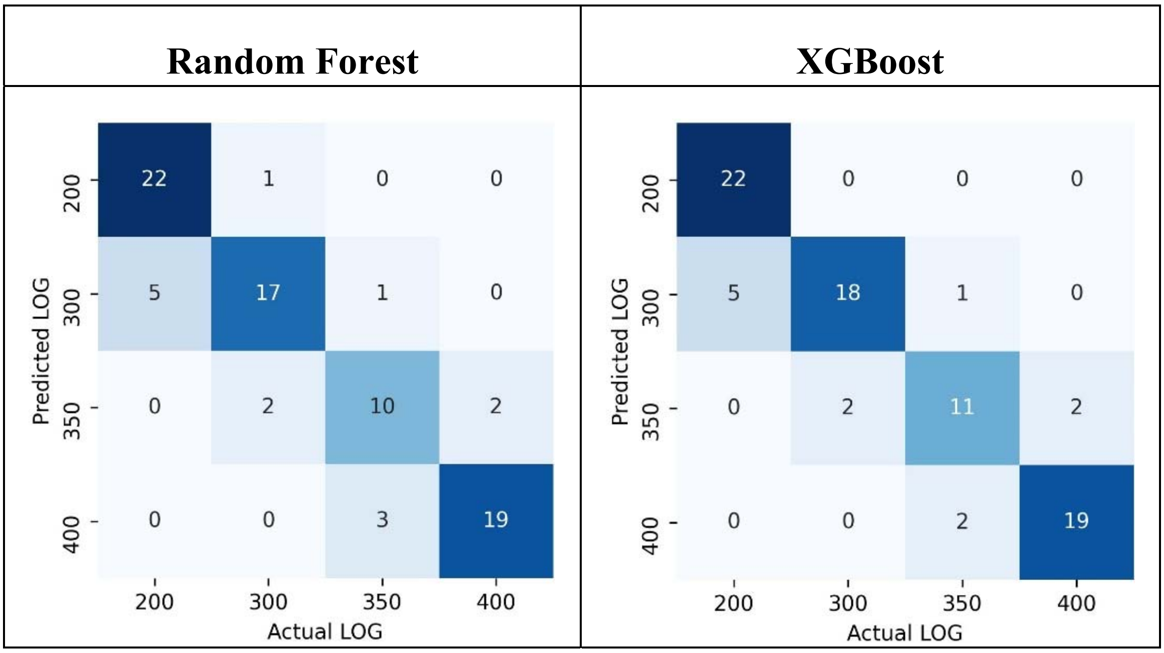
| **LOG** | **Precision** | | **Recall** | | **F1-score** | |
|---|---|---|---|---|---|---|
| | **RF** | **XGBoost** | **RF** | **XGBoost** | **RF** | **XGBoost** |
| **200** | 0.96 | 1.00 | 0.81 | 0.81 | 0.88 | 0.90 |
| **300** | 0.74 | 0.75 | 0.85 | 0.90 | 0.79 | 0.82 |
| **350** | 0.71 | 0.73 | 0.71 | 0.79 | 0.71 | 0.76 |
| **400** | 0.86 | 0.90 | 0.90 | 0.90 | 0.88 | 0.90 |
| **Accuracy** | | | | | **0.83** | **0.85** |
| **Macro Avg.** | 0.82 | 0.85 | 0.82 | 0.85 | 0.82 | 0.84 |
| **Weighted Avg.** | 0.84 | 0.87 | 0.83 | 0.85 | 0.83 | 0.86 |

To understand the evaluation results in more detail, Figure 5.15 shows the confusion matrix for both models depicting the difference between the actual and predicted LOGs. 68 and 70 out of 82 elements were classified correctly using RF and XGBoost, respectively. When investigating the incorrect classifications further, we notice that the LOGs were confused with their nearest neighbors. For instance, five elements[5] at LOG 200 were classified as LOG 300, and two elements at LOG 300 were classified as LOG 350[6]. This is mainly because the number of changes modeled to detail the elements further from LOG 200 to 300 does not necessarily increase the shape complexity enough to be differentiated. Moreover, this approach heavily relies on the dataset size (finding similar observations). Thus, increasing the dataset size even more has a potential for improving the accuracy of the classifications.

---

[5] Fire Mains, Heating Pipe Fittings, Wood Stair, Trefoil Round Arch Window, Ventilation System

[6] Multilayered Slab (Reinforced Concrete), Escalator (variant 2)


Random Forest

XGBoost

Figure 5.15: Evaluation results: confusion matrix of the RF and XGBoost models performance on test data. The x-axis represents the actual LOG, and the y-axis represents the predicted LOG. The diagonal boxes show the correctly predicted LOG classes.

### 5.4.5 Experiment: Performance robustness evaluation

As discussed previously, the ensemble models developed produce their classifications based on the extracted geometric features. The elements dataset presented in this paper was entirely modeled by using Autodesk Revit. Additionally, the Revit API was used to export the triangulated mesh representations that were used to extract the different geometric features. The Revit API provides specialized methods for retrieving the geometric representation of the individual elements [7] [8]. Our implementation was based on an available example code provided by Autodesk [9], where we used the maximum value for the *LevelofDetail* parameter when generating the triangulated mesh. Considering that every BIM-authoring tool might have its own geometry kernel, which could represent the geometry with more or fewer triangles, this experiment aims to evaluate whether the performance of the developed ensemble models would be affected by re-meshing the testing dataset with differently distributed triangles.

The re-meshing process was performed using the Isotropic Explicit re-meshing algorithm (Alliez et al., 2003), where the shapes became more condensed and uniform. The Isotropic Explicit re-meshing can deal with a variety of mesh shapes and is widely used and implemented

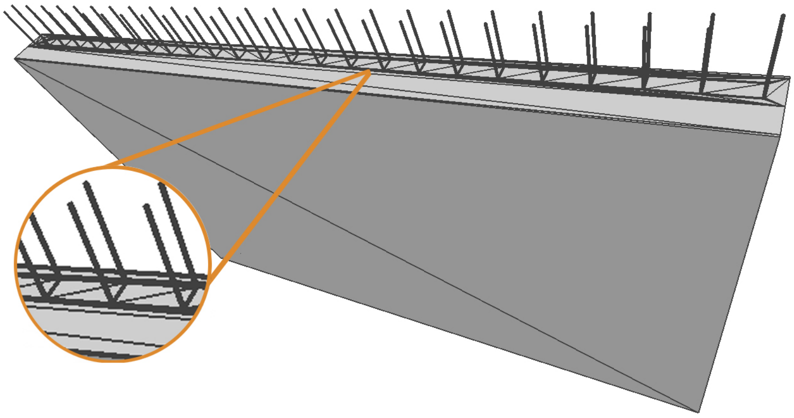
---

[7] https://www.revitapidocs.com/2015/d8a55a5b-2a69-d5ab-3e1f-6cf1ee43c8ec.htm

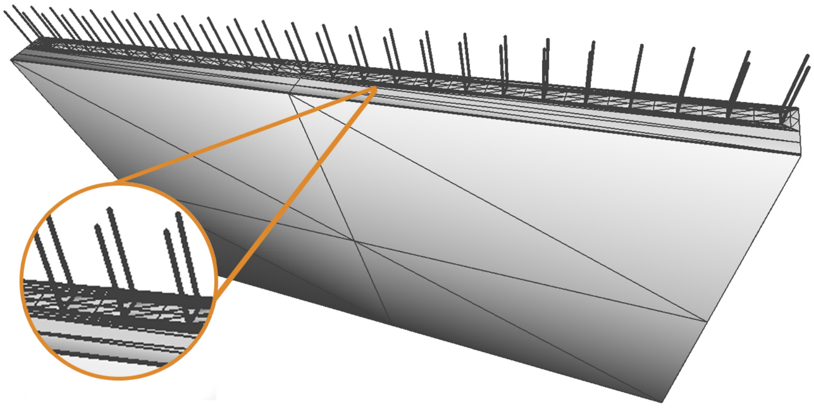
[8] https://thebuildingcoder.typepad.com/blog/2015/04/exporting-3d-element-geometry-to-a-webgl-viewer.html

[9] https://jeremytammik.github.io/tbc/a/0792_obj_export_v1.htm#6

in multiple geometry processing tools, such as Meshlab[10]. Figure 5.16 shows an example of a multilayered reinforced wall before and after re-meshing. The wall's vertices, edges, and faces after re-meshing are ∼5.1-fold their values before re-meshing. Any existing BIM-authoring tool will typically export less condensed meshes, which are more similar to the original dataset (before re-meshing). Thus, using such condensed re-meshing is adequate for evaluating the robustness of the trained models' performance.



(a) Before re-meshing: 627,702 vertices, 1,167,543 faces, and 3,589,782 edges.



(b) After re-meshing: 3,189,421 vertices, 6,049,491 faces, and 18,482,220 edges.

Figure 5.16: Re-meshing experiment: a sample reinforced wall before and after re-meshing.

After re-meshing, the geometric features are extracted again for the re-meshed test dataset. As illustrated by Figure 5.16, the basic geometric features of the re-meshed elements are approximately five-fold their values before re-meshing. However, the extracted sharp edges and segments were not affected by re-meshing, as neither the elements' diameter nor their outline has changed. The features that are affected by re-meshing are those which rely on the edges' lengths as well as the count of vertices, faces, and edges. As indicated in Figures 5.12 and 5.14, the affected features are not part of the top four important features. However,

---

[10] https://www.meshlab.net/

the *% of Mean Edges Length* is the fifth feature, and the rest of the features combined also contribute to the final classification of the different LOGs.

Table 5.5: Re-meshing experiment results: performance results of the RF and XGBoost models on test data. The performance metrics are described as precision, recall, and F1-Score. Precision describes the model performance in positive predictions while considering false positives. Recall incorporates false negatives instead of false positives, and F1-score provides a balance between precision and recall. The colors highlight the change in values compared to before re-meshing (Table 5.4); green when improved and red when degraded.

| **LOG** | **Precision** | | **Recall** | | **F1-score** | |
|---|---|---|---|---|---|---|
| | **RF** | **XGBoost** | **RF** | **XGBoost** | **RF** | **XGBoost** |
| **200** | 1.00 | 1.00 | 0.78 | 0.81 | 0.88 | 0.90 |
| **300** | 0.70 | 0.75 | 0.95 | 0.90 | 0.81 | 0.82 |
| **350** | 0.69 | 0.52 | 0.79 | 0.86 | 0.73 | 0.65 |
| **400** | 0.94 | 0.92 | 0.81 | 0.57 | 0.87 | 0.71 |
| **Accuracy** | | | | | **0.83** | **0.78** |
| **Macro Avg.** | 0.83 | 0.80 | 0.83 | 0.79 | 0.82 | 0.77 |
| **Weighted Avg.** | 0.86 | 0.84 | 0.83 | 0.78 | 0.83 | 0.79 |

After extracting the geometric features for the test dataset, they were used to evaluate the performance of the already trained models (in Sections 5.2 and 5.3). The evaluation results are presented in Table 5.5. To highlight the difference to the performance before re-meshing (Table 5.4), the metrics are colored according to the change in their values; green when improved and red when degraded.

When comparing the overall accuracy of both models before and after re-meshing, although the RF model metrics have increased and decreased across the different LOGs, it maintained the same accuracy of 83%. In contrast, the accuracy of the XGBoost model dropped from 85% to 78% after re-meshing. The XGBoost has maintained its performance for the LOGs 200 and 300, which is an advantage compared to RF. However, the performance degraded at the LOGs 350 and 400. In this regard, multiple elements at LOG 400 were classified as LOG 350 (see the confusion matrix shown in Figure 5.17). We observe that mesh density has only a slight impact on the models' performance. Hence, the trained ensemble models are capable of classifying the LOG of elements that are meshed differently than the training data.

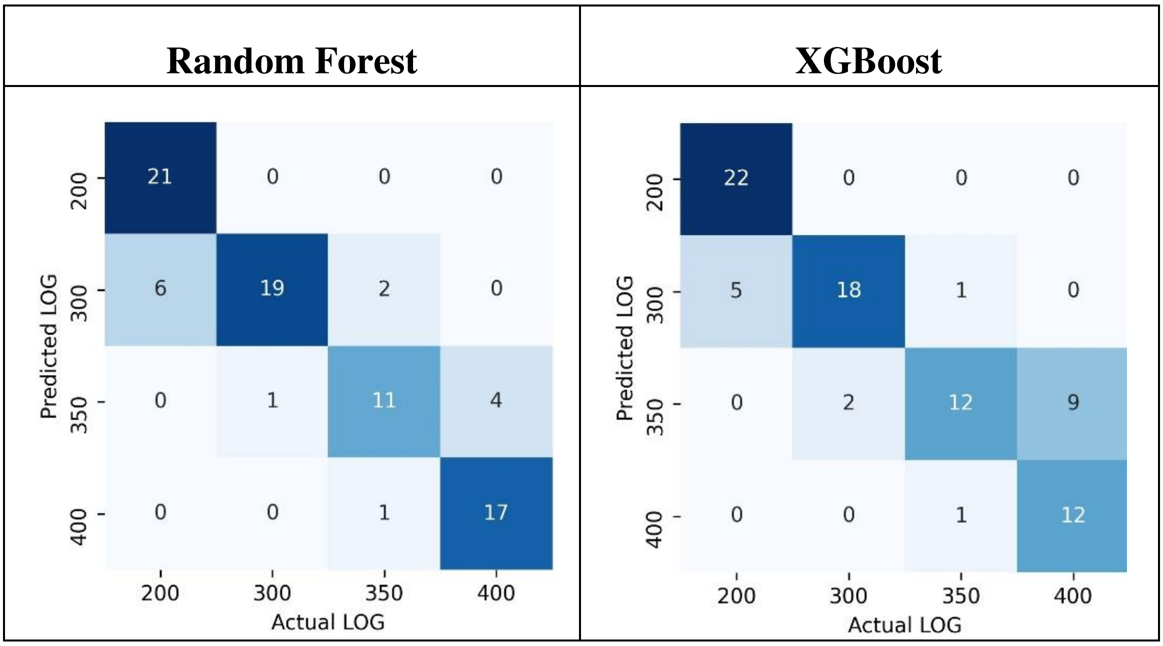


Figure 5.17: Re-meshing experiment results: confusion matrix of the RF and XGBoost models performance on test data. The x-axis represents the actual LOG, and the y-axis represents the predicted LOG. The diagonal boxes show the correctly predicted LOG classes.

## 5.5 Case study

This section highlights the applicability of the developed approach in checking the LOG within the established workflows in practice. As illustrated in Figure 5.18, the requirements of the delivered BIM models are typically specified in contracts and BIM execution plans. These specifications include the LOI and LOG of the individual element types required from each domain expert on every design phase.

During every design phase, the different domain experts base their work on models provided by experts from other disciplines. At this point, the exchanged models need to be checked for fulfilling the minimum requirements necessary by the recipient discipline for carrying out its tasks. Once the different models are integrated and handed over to the client, a final quality check for fulfilling the various requirements is performed. When issues are detected in the project delivery, feedback is sent back to the project participants requesting clarification and solving the issues identified. Otherwise, the design phase delivery is confirmed by the client, which will be used as a basis for developing the design further in the subsequent design phase.

The example shown in Table 5.6 is a subset of the requirements specified for a real-world project (*Ferdinand Tausendpfund GmbH & Co. KG* [11] office building, in Regensburg, Germany). While modeling the conceptual design, the owner decided to build a sustainable building

---

[11] https://www.tausendpfund.group/

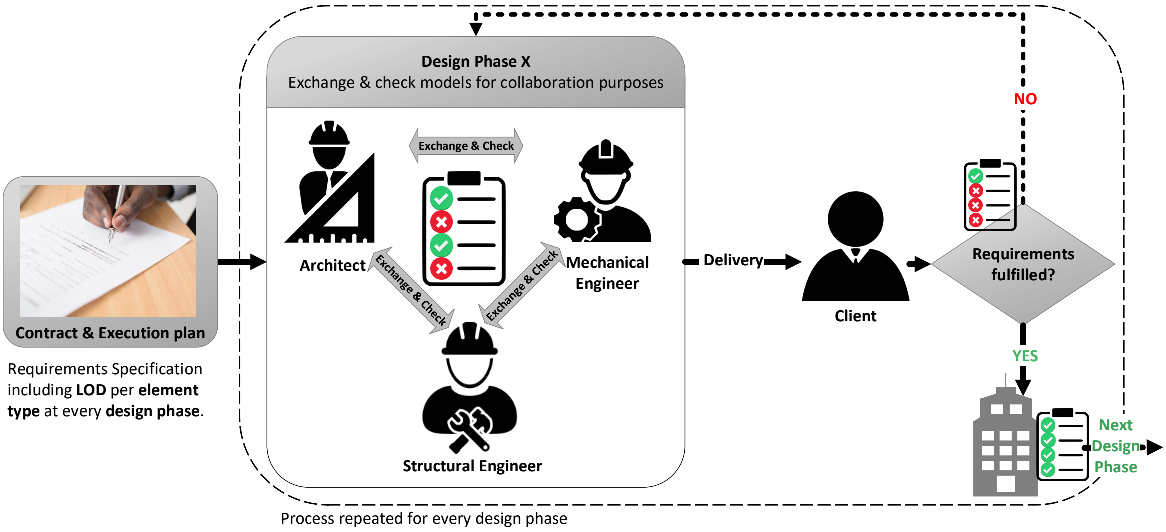


Figure 5.18: Illustration of the multidisciplinary design process, highlighting the specification of a project's LOD requirements in contracts and BIM execution plans, and then validating the specified requirements during the collaboration with different disciplines as well as delivery to the client.

and explore multiple design options through evaluating the performance of their structural system as well as embodied and operational energy consumption (Abualdenien et al., 2020; Abualdenien & Borrmann, 2019). The table lists a subset of the mappings between the building element types and their geometric and semantic requirements that must be delivered at the end of the conceptual design.

Table 5.6: Example of an LOD specification, showing the required types of building elements and their corresponding LOG and LOI specifications.

| Identification | | Level of geometry | Level of information (LOI) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Element type | IFC class | LOG | Name | Core material | Load-bearing function | Surface covering (texture) | Fire protection characteristics | Part of escape route? | Sound insulation characteristics | Is external? | Thermal transmittance |
| Windows | IfcWindow | 300 | x | x | | | x | x | x | x | x |
| Walls | IfcWall | 350 | x | x | x | x | x | x | x | x | x |
| Curtain Walls | IfcCurtainWall | 350 | x | x | x | x | x | x | x | x | x |
| Stairs | IfcStair | 200 | x | x | | | | x | | x | |
| Ramps | IfcRamp | 200 | x | x | | | | x | | x | |
| Doors | IfcDoor | 200 | x | x | | | | x | | x | |
| Ceiling | IfcCeiling | 300 | x | x | x | | x | | | | |
| Sanitary | IfcSanitary | 100 | x | | | | | | | | |
| Rooms | IfcSpace | - | x | | | | | x | | x | |
| Slabs | IfcSlab | 300 | x | x | x | | x | | x | x | |
| Roofs | IfcRoof | 300 | x | x | x | | x | | x | | x |
| Beams | IfcBeam | 200 | x | x | x | | | | | x | |
| Columns | IfcColumn | 200 | x | x | x | | x | x | | x | |
| Structural truss | IfcAssembly | 200 | x | x | x | | | | | | |
| Foundation | IfcFooting | 350 | x | x | x | | x | | | | x |
| Framing | IfcBuildingElementProxy | 300 | x | x | x | | x | | x | | |

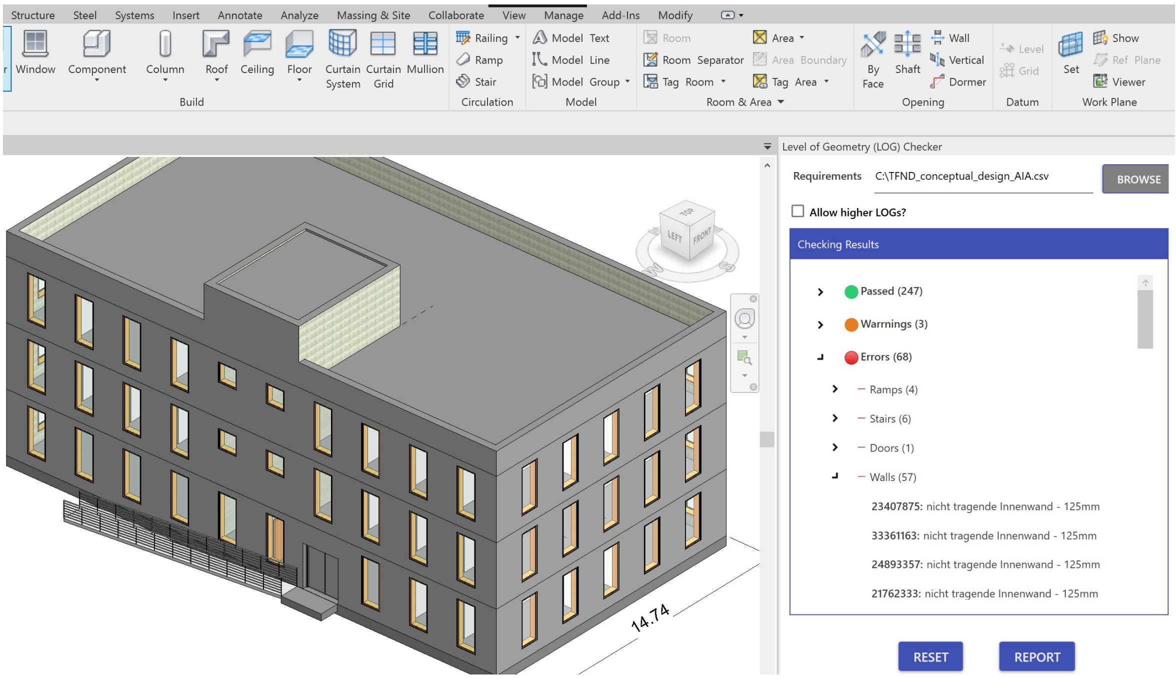
To emphasize on the integration of checking the LOG within the design process, a plugin that uses the developed approach was developed inside Revit, shown in Figure 5.19. The figure shows a design option of the Tausendpfund's office building on the left and the results of checking the LOG of the individual elements on the right. For this purpose, the trained XGBoost model from this study was hosted on a *Flask*[12] server, where the plugin inside Revit sends the geometrical features of the individual elements through a representational state transfer (REST) web-service and receives the predicted LOG as a response. Then each predicted LOG is compared to the project's requirements (which are selected as a CSV file at the top). Finally, the results are reported as lists grouping the elements as *Passed*, *Errors* (when they did not pass), and *Warnings* (for element types that are part of the specification and were not found in the model). Checking the LOG revealed the following deviations:

- Stairs and ramps were identified as LOG 350, whereas they are required to be at LOG 200. The elements used in the model were not as simple as generic representations as they have included detailed railings and connections. After a discussion with the modelers, their reasoning was that the used families are standard and were developed for other similar projects.
- Interior walls were identified as LOG 200 rather than 350. After inspecting the model, we found that the used walls are single-layered walls and do not model the exterior or interior details such as framing, insulation, or connections. Generally, interior walls are not much developed at this phase; however, the specification should have differentiated between interior and exterior walls.
- Entrance door was identified as LOG 300 rather than 200 as the automatic door opener is additionally modeled.

Reconsidering the XGBoost's evaluation matrix (Figure 5.15), the model's accuracy is 83% due to confusion with the adjacent LOGs. However, no LOG was confused with another LOG that is higher or lower than one level, like confusing LOG 200 with LOG 350. Hence, in this case study, the trained model could raise certain warning flags when the modeled LOG is not compliant with the specification. In some cases, when there is no considerable increase in detailing between the LOGs 200 and 300, the model's prediction might be less accurate. In those cases, it would be helpful to inform practitioners about the prediction probability (e.g., 65% to highlight any potential inaccuracies) as well as enhance the accuracy of the LOG prediction by checking the provided semantics. Additionally, custom industry cases can be handled with tailored behaviors, such as considering elements at a higher LOG than what is required as compliant and marking them as passed.

---

[12] https://flask.palletsprojects.com/en/2.0.x/



Level of Geometry (LOG) Checker

Requirements C:\TFND_conceptual_design_AIA.csv

☐ Allow higher LOGs?

Checking Results

- ● Passed (247)
- ● Warrnings (3)
- ● Errors (68)
  - Ramps (4)
  - Stairs (6)
  - Doors (1)
  - Walls (57)
    - **23407875:** nicht tragende Innenwand - 125mm
    - **33361163:** nicht tragende Innenwand - 125mm
    - **24893357:** nicht tragende Innenwand - 125mm
    - **21762333:** nicht tragende Innenwand - 125mm

Figure 5.19: A snapshot of the developed plugin inside Revit for checking the LOG of building elements.

## 5.6 Conclusions & Future Research

The automatic validation of building information for compliance with the design requirements is crucial for an efficient and successful project outcome. The LOD concept is used to specify the expected information of the individual elements. Currently, automatically checking the completeness of the semantic information against the LOD specification is supported by multiple tools. However, validating the conformance of the provided geometry to the required LOG is currently a manual, laborious task and solely based on domain experts' subjective assessment.

This paper contributes a framework for formally defining and automatically checking the LOG of building elements. The proposed approach is based on modeling and analyzing a dataset of 408 building elements (102 families at the LOGs 200, 300, 350, and 400). The families were modeled according to the most established and widespread LOD specifications, the BIMForum's LOD specification and Trimble's Project Progression Planning, and are provided to the scientific community as open data. The existing descriptions and especially the graphical illustrations from these specifications were used as baseline for the modeling process. Additionally, measuring the necessary modeling time (which reflects the required effort and cost) to detail the elements further from one LOG to the subsequent one showed a 1.88 – 2.80-fold increase.

From the experience gained in this study, we highlight that even when modelers might have different interpretations of the fine details expected at each LOD, the outcome would be comparable and sufficient as long as they follow a unified specification and understand which features must be modeled (such as the exact shape dimensions at LOG 300 or connections at LOG 350). Additionally, following and interpreting the specifications to model the geometry can be improved by providing graphical illustrations for all families. Currently, numerous families are only described with textual descriptions. Furthermore, providing open-access LOG examples would support achieving a common understanding of what needs to be modeled. Currently, no LOG examples are available online (other than the dataset published by this research).

On the basis of the created dataset, the detailing of the individual building elements at each LOG was formally analyzed and represented by a set of features. A main scientfic contribution of the paper is the identification of the relevant geometric features. The geometric features were extracted using multiple geometry analysis techniques, including the basic geometric features (such as vertices, faces, and edges), sharp edges, and diameter-based segmentation. The extracted features were then used to train RF and XGBoost models to classify the LOG of any building element.

The results show that the extracted geometric features can describe the elements' complexity in a way that represents the modeled features at every LOG. Both of the ensemble models were able to classify the LOG of the test dataset with an accuracy of 83% for the RF model and 85% for the XGBoost model. After detailed investigations (as shown in Figure 5.15), we found that the misclassified elements (18% for RF and 15% for XGBoost) were only confused with their nearest neighbors, e.g., LOG 200 with LOG 300. Hence, both of the trained models are capable of providing practitioners a reliable indicator of the geometric detailing of the individual elements. Additionally, to enhance the reliability of predictions, we propose informing practitioners about the predictions' probability to highlight potential inaccuracies that require a manual inspection.

In order to evaluate the robustness of the trained models, the test dataset was re-meshed in a much denser triangulation, where the RF model maintained its accuracy of 83%. In contrast, the accuracy of the XGBoost model dropped to 78%. These results demonstrate the capability of the trained models in correctly classifying the LOG of elements that are meshed differently than the training data. Accordingly, using the RF model in practice would provide more robust accuracy for classifying the LOG, when the evaluated families are provided from diverse BIM-authoring tools.

In general, the ensemble models have proven their capability of learning the geometric features. Increasing the dataset size further has a potential for improving the ensemble models' accuracy, especially given that the change from one LOD to the other does not substantially increase the shape's complexity for all the families. Based on the knowledge gained from this paper, the process of extracting geometric features is a sensitive and time-consuming task. As shown in this study, the elements' geometry must be pre-processed into a set of (human-made) representative features when using ensemble learners, which involves performing multiple computationally extensive tasks. Therefore, as a next step, mesh convolutional neural networks will be evaluated for directly extracting geometric features and classifying the LOG of triangulated meshes. Classifying the LOG of building elements directly from meshes would reduce the necessary processing effort and could improve accuracy, since the geometric features would be statistically inferred by the neural network rather than manually identified and extracted.

## Acknowledgements

We gratefully acknowledge the support of the German Research Foundation (DFG) for funding the project under grant FOR 2363. We thank Ferdinand Tausendpfund GmbH for providing their office building as a sample project. We also thank Patrick Nordmann for his contribution in modelling the LOG dataset.

# Chapter 6

# BIM-based design decisions documentation using design episodes, explanation tags, and constraints

Previously published as: Zahedi, A.; Abualdenien, J.; Petzold, F.; Borrmann, A.: *BIM-based design decisions documentation using design episodes, explanation tags, and constraints*, Journal of Information Technology in Construction 27, pp. 756-780, 2022, DOI: https://dx.doi.org/10.36680/j.itcon.2022.037

## abstract

The process of designing a building involves producing design concepts while fulfilling various requirements and regulations. Furthermore, during the project's life-cycle, multiple experts from multiple domains collaborate in developing the different partial models, including architectural, structural, and HVAC among others. Accordingly, clearly communicating the rationale behind design decisions is crucial for developing regulatory compliant designs that also fit the owner's needs. The developed designs are the main deliverables exchanged and handed over. However, these deliverables do not include any explanation of design intentions or documentation of design decisions. Communication among parties and reuse of knowledge are hindered by the absent explanation of existing design. To overcome this deficiency, this paper proposes a methodology for digitally documenting design decisions, incorporating their intention and rationale. Architectural concepts and evaluation criteria are represented in the form of explanation tags as well as spatial and semantic constraints, which are assigned to the individual model elements and properties. Additionally, to document how design decisions fulfill owner requirements and regulatory documents, natural language processing (NLP) is employed to facilitate querying those documents and then the individual requirements are linked to specific elements, properties, and constraints. To evaluate the proposed methodology, a prototype was implemented as a plugin inside a BIM-Authoring tool and multiple real-world use cases are discussed.

## 6.1 Introduction

Construction companies are frequently challenged to innovate and develop customized solutions in order to solve project-specific obstacles. This results in the creation of new knowledge, which should be adequately recorded and maintained (Zahedi et al., 2022). Designing a building is an iterative task that gradually progresses through multiple phases. Throughout the design process, the design task and its solutions co-evolve. Starting from the early design phases, the maturity of the design increases in the form of more precise and detailed information. Typically, for every project, designers are confronted with a set of requirements and boundary conditions that need to be fulfilled and accounted for. However, throughout the design process, far more knowledge about the client's requirements is gathered compared to the beginning. Furthermore, construction projects are multidisciplinary, involving diverse domain experts, where each has their own perspective and interest. In many cases, the interests of these experts contradict each other. For example, a structural designer might focus on massive construction due to a high load-bearing capacity. The architect, however, might prefer structures that appear lighter and more slender, whereas the energy consultant recommends using renewable construction materials.

Each building ought to fulfill a combination of requirements and goals that do not necessarily share the same nature. Some of these requirements and conditions are based on objective criteria that could be measured and compared rather quantitatively. Others are based on subjective criteria that could not be easily measured and compared due to their qualitative nature and description. The most essential kinds of requirements that must be fulfilled during the design process are Request for Proposal (RFP) and building codes. An RFP describes the owner's requirements, including the main form, building use, as well as privacy and sustainability standards. During the design phases, designers use RFP documents as the guideline for fulfilling the owner's requirements and needs (C. Eastman et al., 2009). However, when looking at real building briefs, one sees that they're often incomplete documents created without a thorough understanding of the design process and technological knowledge and that they require extensive interpretation and addition. Interpreting a project's RFP is typically based on the designers' knowledge and experience (Odusami, 2002). Furthermore, the content of RFPs depends on multiple aspects, such as culture, building usage, and even year of construction (Uhm et al., 2015). For example, a residential house has a much simpler RFP than a residential building or a hospital. Additionally, privacy and sustainability requirements may differ if we compare Middle Eastern countries to the US or today's buildings to those of 50 years ago. In the same context, before permitting building designs to be constructed, they must first fulfill numerous building codes and regulations. Building codes provide prescriptive- and performance-based requirements for different building types, including shopping centers, offices, and educational facilities. Connecting these RFP documents to building codes via NLP techniques to further help the architects in this matter is part of this paper's focus.

In everyday practice, it is not mandatory to document the intermediary architectural design choices and variants. Thus, design variants are hardly ever documented as an intermediate step, but mostly only the final result is recorded, e.g. graphically through drawings and views, as well as in the text, or sometimes in digital models (WIESBADEN, 2013). This can lead to reinventing the wheel or making the same mistakes over and over again. We believe that management of design Knowledge is becoming in some senses the core intangible asset of architectural firms competing in the global information-intensive construction industry with ever more complex technologies and demanding clients. The authors argue that comprehensive documentation of design knowledge, sharing it with various stakeholders and decision-makers, and reusing it for future design projects is lacking currently in the majority of the construction industry. However, it is becoming increasingly beneficial and important in the construction industry to manage, share and reuse the design knowledge to improve efficiency and productivity in an industry which is known for its lack of advancement compared to other modern industries (TANG et al., 2006). The value of capturing and documenting design knowledge for architectural firms can be discussed on different levels (BRACEWELL et al., 2009; HEYLIGHEN, NEUCKERMANS, CASAER, et al., 2007):

- first and foremost, to preserve the company's intellectual property and shared knowledge and make it available for reuse in other projects, which in turn leads to fewer redundancies of work and more effective teamwork and greater client satisfaction, and less reliance on the experience and knowledge of key individuals;
- to enable systematic self-criticism and self-improvement inside the firm by learning from mistakes and successes in other projects, resulting in fewer mistakes, fewer resources wasted and more effective decision making and innovative thinking;
- while considering the associated copyright issues, creating the opportunity to share and learn from each other in a profession known to be highly secretive and over-protective of their designs.

  This paper addresses the problem of systematically capturing the tacit design knowledge through documenting and explaining the design decisions. Therefore, a novel approach is introduced based on BIM (Building Information Modeling) methodology and Natural Language Processing (NLP) techniques to link client requirements and building codes to design concepts and to record and document design decisions and their explanation in a transparent manner for all stakeholders. Recording and exchanging explanations about the decisions made during the design process will improve mutual understanding and collaboration among all designer parties and domain experts throughout the design process and will enhance the inter-organizational exchange and reuse of the shared knowledge and designs for other future projects and design problems.

  The contributions of this paper are threefold: first, the development of multiple concepts and approaches for expressing and documenting design decisions, i.e. Explanation

Tag (ET) and Design Episode (DE) and Constraints; second, introducing a framework for parsing, querying, and linking natural text to BIM models: third, the proposed approaches are formally represented using the multi-LOD meta-model and evaluated for practical use through multiple use cases including two real-world building projects. Through the utilization of explanation tags and constraints, various documented design episodes can be created and stored, and later accessed and retrieved using case-based reasoning as well as NLP techniques.

This paper is organized as follows: after the introduction, in section 6.2, we review the background and related work, followed by section 6.3 in which we explain the applied methodology and introduced concepts in this paper. Subsequently, at the end of section 6.3, we discuss the implementation part as proof of concept, and then in section 6.4, three demonstrative use cases, including two real-world building projects, are discussed to illustrate our novel approach through various design examples. Finally, section 5 summarizes our progress and presents an outlook for future research.

## 6.2 Background & related work

This section starts by addressing the challenges involved in recording the architectural building design process due to its unique nature with special regard to the importance of early stages of design and then follows with reviewing related literature addressing this problem. During which, BIM and its benefits, as well as its shortcomings, together with the related literature regarding proposed solutions and enhancements for it, are discussed. The use of references in architecture and suggested solutions for knowledge extraction from semantic models will be discussed next. We will also review and discuss similar work in the field of case-based reasoning and design since the proposed concepts of explanation tags and design episodes (among the contributions of this paper) provide opportunities for using case-based reasoning techniques for future retrieval and reuse of design knowledge. Finally, related research in the field of Natural Language Processing (NLP) and design constraints will be covered.

### 6.2.1 Architectural knowledge

Ackoff (Ackoff, 1989) in his data–information–knowledge–wisdom hierarchy (DIKW) described data as symbols that denote the attributes of objects and events, whereas information is data that has been processed to improve its usefulness. Data and information differ in terms of function rather than structure. Moving up in the hierarchy is knowledge that Ackoff defines as know-how that can be obtained through training or instructions from someone who possesses it (Ackoff, 1989; Rowley, 2007). Literature also defines the term "knowledge" as a concept with multiple layered meanings (Habraken, 1997; Polanyi, 2009; Schön, 1987). Therefore, knowledge tends to be addressed via distinctions between its different types, whether it is between declarative and procedural knowledge (Ryle, 2009), or between explicit and tacit knowledge (Polanyi, 2009). Tacit knowledge is formed by the experience of individuals. This type of knowledge is expressed via evaluations, attitudes, points of view, commitments, motivations, and similar forms of human actions. However, on the practical level such as in architecture, many experts fail to articulate their knowledge, abilities, decision process, and conclusion deduction. In a professional context, there is a notable difference between the knowledge base, i.e. the formal and codified domain expertise claimed by a profession (Habraken, 1997), and the practitioner's 'knowing-in-practice', which as Schön indicated, is greatly implicit and learned by engagement.

Regarding architectural design, there is a discussion of whether architectural knowledge is specific and requires unique treatment in contrast to other fields of knowledge. According to Lawson (Lawson, 2018), design education is different compared to other major learning approaches. Lawson argues that schools of design tend to follow a very similar pattern grounded in the traditional master-apprentice model; students working in the studio on limited yet realistic design projects are tutored and supervised by designers with more experience. CB de Souza discusses that the knowledge associated with the architectural design of buildings is mainly constructivist, it is a knowledge that comes from experience (de Souza, 2012). This exceptional cultivation of knowledge-through-practice in architecture has led to the lack of formal codification of a common knowledge-base, as practiced in other professions, such as law or medicine (Habraken, 1997). It appears that the architectural knowledge-base is mainly implicit and embedded within the architects' reasoning and creativity, which in turn leads to challenges in incorporating knowledge management theories and methodologies that have gained widespread acceptance in other fields (Heylighen, Martin, et al., 2007). A key challenge here is that the professional language of architecture is not easy to define, as it can certainly be seen on the one hand as a technical language, the language of civil engineers, and on the other hand as the artists' specialized language (Kuznecova & Löschmann, 2008). Moreover, as Habraken convincingly argues, architecture lacks a common lexicon of general recognition and significance, for architects have an alarming tendency to coin personal vocabulary and rename elements on a regular basis.

### 6.2.2 Building design decision-making process

The building design process is challenging to be captured and comprehensively documented due to so many reasons, most of which relate to the nature of design problems. Design problems are identified as 'wicked' problems by Rittel and Webber (Rittel & Webber, 1973), which makes them basically ill-structured. Thus, according to Rittel and Webber dealing with wicked problems, one should see the concept of planning as an argumentative process in which a vision of the task and solution coevolve progressively among the participants as a result of continuous reasoning and critical debate. Furthermore, Gero introduced the Function-Behaviour-Structure (FBS) ontology as a design ontology to describe all designed artifacts and then based on that the FBS framework and later the situated FBS (sFBS) framework to describe all designing processes (GERO & KANNENGIESSER, 2014). Another interesting perspective on design decisions is the naturalistic decision-making theory, which views decision-making as a continuous flow of acts that work toward a set of goals rather than as discrete choices (KLEIN et al., 1993). Design problems are listed as one of the domains where naturalistic decision making may be found; where problems and goals are poorly structured and shifting; a dynamic and uncertain context in which the decision-maker must deal with incomplete and vague information; and situations in which a series of choices and events rather than a single decision must be made (KLEIN et al., 1993).

Using BIM (Building Information Modelling) methodology, complete digital representations of built facilities are created as building information models and utilized for storing, maintaining, and sharing information (BORRMANN et al., 2018a). The Level of Development (LOD) concept describes the progressive refinement of the geometric and semantic information by providing definitions and illustrations of BIM elements at different stages of their development (BIMFORUM, 2019; JANSON & TIGGES, 2014). Even though BIM is potentially altering the way architects, engineers and contractors conduct their work and daily jobs, it's still early in its implementation and the construction industry's fragmentation prevents BIM from becoming completely adopted and more widely used (BORRMANN et al., 2018a).

### 6.2.3 The importance of early design stages

Building design as a problem-solving process starts with the customer's demands, which are then converted into a design job. However, requirements or even an RFP are not the same as defining the design problem, and the designer must interpret the requirements in a meaningful way. Furthermore, it is not only the clients' wishes and demands that form a building design, but also numerous regulations, constraints, and technical aspects. The most important phases of the building design are the early phases (preliminary and conceptual phases), where fundamental and crucial design decisions are made (KOLLTVEIT & GRØNHAUG, 2004). The earlier the design stage, the easier it is to change or modify design aspects, whereas, in more advanced phases, it becomes more difficult to change or modify prior design decisions (STEINMANN, 1997). The main difficulty during these early phases is the sheer load of design decisions, and the lack of sufficient information and knowledge about the consequences of those decisions (ZEILER[1] et al., 2007).

The BIM methodology substantially enhances the coordination of design operations, simulation integration, and the transfer of building information (BORRMANN et al., 2018a), however, utilizing BIM during early design stages has its own difficulties. While the information contained in BIM models appears exact and certain, most design aspects and details are uncertain and ambiguous, during the early stages of building design. To address this challenge, Abualdenien and Borrmann developed a multi-LOD meta-model (Exner et al., 2019) for formal specification of maturity levels of building information models, while allowing the explicit expression of potential information vagueness during the early design phase. Abualdenien and Borrmann also presented different approaches and concepts for visualizing the vagueness and uncertainty in building models across different design stages (Abualdenien & Borrmann, 2020b) and for formally analyzing and classifying the geometric detailing of building elements (Abualdenien & Borrmann, 2022).

Furthermore, to ask for expert opinions about different design aspects (via simulations and analysis), more information and details are required, which are only available in later design phases (Zahedi et al., 2019). Similarly, collaborations and cooperation between multiple domain experts and stakeholders have proven to be essential for achieving a good and optimal design. To deal with this problem, Zahedi and Petzold developed a minimal machine-interpretable communication protocol based on BIM to facilitate the workflow and communicate the proposed detailings and their corresponding evaluation results for supporting the decision-making process (Zahedi et al., 2019). Matern and König introduced an approach for managing various design variants across multiple planning stages in a consistent digital building model (Mattern & König, 2018). Geyer and Singaravel showed that engineering surrogate models based on components and machine learning (ML) can predict energy demand with the required accuracy in the early stages of design (Geyer & Singaravel, 2018) and with a small prediction gap in comparison to the dynamic simulation approach (Singh et al., 2020).

### 6.2.4 References and knowledge extraction from semantic models

The use of references in architecture is considered a recognized method (Gänshirt, 2012) for supporting design, testing ideas, clarifying design parameters, or showing new ways and possibilities. It is a method that supports decision-making. The built and planned models serve as a knowledge base that includes spatial situations as well as solutions for specific architectural expressions. The use of analogies in references is an efficient method for documentation, both in design and in downstream activities. Due to the growing acceptance of the BIM methodology, BIM models are increasingly being stored in cloud repositories. A retrieval system is a prerequisite for effectively managing and using these models. Most commercial BIM retrieval approaches use text-based and keyword-based search strategies that rely on metadata (e.g. keywords, tags, descriptions). Gao et. al. (Gao et al., 2015) presented a concept for a text-based semantic search engine and its prototypical implementation "BIMSeek" to make online BIM resources accessible. Based on the IFC data model (Industry Foundation Classes), a domain ontology was built to encode BIM-specific knowledge in the search engine.

By combining both the ontology and local context analysis techniques, an automatic search-enhancement method was integrated to improve search performance. In addition to the textual search, a graphical search is viable; in Inanc (Inanc, 2000) with a 2D graphical search and in Funkhouser et al. (Funkhouser et al., 2003) with a 3D graphical search. Among others, Demian et al. (Demian et al., 2016) presented a combination of graphical and topological search. The use of graphs in the BIM context for analyzing and extracting information and knowledge has been the focus of various research projects. Langenhan et. al. introduced the concept of semantic fingerprint of buildings to formalize architectural spatial situations and the computer-aided determination of similarity (Langenhan et al., 2013). Furthermore, Ayzenshtadt et. al. designed an extension assistance system based on the distributed AI-based methodology FLEA (Find, Learn, Explain, Adapt) to inform architects and offer solution suggestions on how the current floor plan solution tends to evolve during the design process (Ayzenshtadt et al., 2018; Eisenstadt et al., 2019).

### 6.2.5 Case-Based Design (CBD)

A general approach in problem-solving, called case-based reasoning, is carried out by drawing on a previously solved similar problem case (Maher et al., 1995). Likewise, learning from previous design cases and using them as inspirations to solve at-hand problems or to use similar details and information from other building designs are the goal of many researchers in the field of capturing and documenting tacit architectural design knowledge.

Qualitative assessments have been discussed especially in research projects in the field of case-based reasoning (CBR) as mapping procedures defined for classifying and documenting design cases. Individual design situations that are represented by design episodes that correspond to specific design features are well known as episodic case-based designs (Maher et al., 1995). In a more graphical approach, as part of their case-based design (CBD) tool called DYNAMO (Dynamic Architectural Memory Online), Neuckermans et al. (Neuckermans et al., 2002) and Heylighen et al. (Heylighen, Neuckermans, Casaer, et al., 2007; Richter et al., 2007) designed and prototypically implemented "visual keys" for visually indexing design cases and as an access mechanism. These visual keys are used as labels, allowing the user to tag design situations and later search for and access similar cases. Visual keys convey architectural expressions and features. For instance, a visual key can refer to an open-ended grid for the building or to the plan-libre (as introduced by Le Corbusier as a free plan arrangement of non-structural partitions determined by functional convenience) for the spatial configuration. A visual key can also refer to the functionality of the building such as a hospital, or a formal qualification such as symmetry for the arrangement of spaces (Heylighen et al., 2003; Martin et al., 2003). Based on two review papers by Heylighen et al. (Heylighen & Neuckermans, 2001; Richter et al., 2007), some other case-based design (CBD) tools and projects include Archie-II (E. Domeshek & Kolodner, 1993; E. A. Domeshek &

Kolodner, 1992), CADRE (Hua et al., 1996), FABEL (Schmidt-Belz & Hovestadt, 1996; Voss, 1997), IDIOM (Smith et al., 1995; Smith et al., 1996), PRECEDENTS (Oxman, 1994), SEED-Layout (Flemming, 1994), SL-CB (J.-H. Lee et al., 2002), TRACE (Mubarak, 2004), CaseBook (Inanc, 2000), MONEO (Taha et al., 2007) and Case Base for Architecture-CBA (LIN & CHIU, 2003).

### 6.2.6 Natural Language Processing (NLP)

Automatically extracting knowledge from unstructured data, such as RFP or building codes, which is written in natural language, is necessary in order to make use of it during the project's life cycle. Natural language processing (NLP) provides the techniques that can provide a computer-readable representation of a natural language text. NLP was leveraged for supporting multiple use cases in the AEC industry. Jung and Lee developed a method that is based on NLP and unsupervised learning to automatically classify the different case studies of construction projects according to their BIM use (Jung & Lee, 2019). Additionally, for supporting in performing an automated compliance checking, Salama and El-Gohary, and Jung and Lee combined NLP with supervised learning algorithms (Jung & Lee, 2019; Salama & El-Gohary, 2016). Moreover, Wu et. al. proposed an NLP-based retrieval engine for BIM object databases, leveraging a domain ontology (Wu et al., 2019) and Lin et. al. introduced an approach for data retrieval from BIM models hosted on the cloud (Lin et al., 2016).

In order to explore and query requirements and regulation documents during the design phases, identifying the semantic text similarity between a natural language query and those documents is necessary. To perform multiple calculations on the natural language, words and sentences from these regulatory documents must be represented in a computer-readable way, typically achieved through a process known as Vectorization (Wilbur & Sirotkin, 1992). A vector is a list of numeric values, where the combination of them represents the overall meaning, which makes it possible to measure the semantic similarity represented by the text, where similar words have vector representations that are closer (Wilbur & Sirotkin, 1992). Measuring the similarity between the numeric vectors has performed remarkably well in different domains (P.-H. Chen, 2020). A key aspect of vectorization is the vocabulary taken into account (the vector space) to generate the vector representations of new sentences or words. The larger the unique words and the dimension of each vector, the better is the resultant vector representation. A typical workflow for performing NLP comprises:

- Tokenization: splitting the sentence into discrete units, i.e., singular words.
- Lemmatization: converting each word to its original form (i.e., dictionary form or lemma). For example, the lemma of the words best and better will be the same, good.

- Part of speech (POS): the generation of POS tags, for example, identifying if a word is a noun, adjective, etc.
- Stop Word Removal: removing stop words, which are tokens that appear with high frequency across the entire document. They typically introduce more noise than signal (benefit).
- Vectorization: converting the textual representation into a vector representation. The main advantage of vectorization is that we can measure the similarity between words to resolve confusion with words that have a similar meaning, e.g., external is very similar to outer and exterior. This paper employs the exact above-mentioned NLP workflow to filter and recommend an applicable set of requirements and regulations for designers during the design process. More details are provided in the following sections.

### 6.2.7 Design constraints

Current BIM-authoring and parametric design tools maintain the integrity of the design based on the imposed geometric constraints (G. Lee et al., 2006; C. Zhang et al., 2020). Domain knowledge includes numerous aspects. For example, if the design is meant to be used for fabrication, a specific set of properties, including material, must be specified. Additionally, good design practices, taking into account the acoustics, circularity, and privacy of the design, pose different kinds of requirements and constraints.

Multiple researchers have investigated incorporating domain knowledge using constraints (Bettig & Shah, 2001; Bhooshan, 2017; N. C. Brown & Mueller, 2019). However, these studies have primarily focused on optimizing the geometric design to fulfill specific building performance indicators rather than on capturing domain knowledge in the form of geometric and semantic constraints. The currently available BIM-authoring tools provide the ability to add dimensional and positional constraints. However, the currently available constraints only support the basic use cases, for example, it is not possible to freely assign constraints to property values (for restricting them) or to constrain the connection position and angle of two walls. Most popular BIM-authoring tools, such as Autodesk Revit, support aligning element position and dimension to each other using predefined constraints, such as equality constraints. Furthermore, the tools automatically apply other constraints implicitly, such as attaching a wall to a roof. The constraints in these systems are meant to support the design process and handle the most common use cases.

However, when considering constraints from the perspective of capturing design knowledge, designers implicitly apply many additional constraints while trying to fulfill owner requirements and regulations. Typically, constraints can be expressed geometrically on element dimensions, positions, and their topological connections, as well as semantically, demanding a specific value, a list of values, or a permissible range of values. To fill this gap, this paper proposes a meta-model approach for capturing domain knowledge in the form of semantic and geometric constraints. The individual constraints are then assigned to the individual elements and properties.

## 6.3 Methodology

The process of capturing and sharing architectural knowledge with its complexity and dynamism requires the consideration of various aspects. Some knowledge is stored within construction documents or the designed model, yet neither can reveal the constantly changing conditions that actually structure the process of designing. As illustrated abstractly in Figure 1, each construction project is bound to specific site information and boundary conditions, which influence the selected architectural concepts, and then the detailing of the individual elements. For example, the site of a residential building that is close to a highway (where traffic is heavy) or near a school facility, requires careful consideration of the designed facade, especially in terms of noise reduction techniques. On the other hand, a site facing a nature preserve or wooded area fosters using curtain walls or big windows.

Taking into account the project's site information, architects and engineers need to take into account fulfilling owner requirements and building codes (requirements level). All of these aspects are combined with the designers' style and domain knowledge to create multiple concepts covering the different aspects of the design's functionality (concept level). Finally, each of these concepts is implemented in the form of detailed components, their connections, and the constraints bounding them (design level).

Numerous aspects of the design knowledge, including organization of spaces, navigation between spaces, the choice of insulation and material layers, etc., are implicitly embedded into design artifacts. But the design processes, including the assessments of intermediate design variants and corresponding design decisions, are hardly comprehensibly documented today. This type of design knowledge is extremely valuable, as it opens new possibilities for improving productivity and efficiency in architectural building design. Decisions in the selection and further detailing of variants ought to be recorded and in particular, the reasons why they were made, to ensure later traceability and transferability of design knowledge to other projects. Such knowledge is highly valuable as it provides a solution that combines architectural tacit knowledge, fulfilling owners' demands, building codes, and the various regulations. To capture design knowledge, we propose the following concepts and approaches.

### 6.3.1 Design episodes and explanation tags

One of the solutions for transferring architectural knowledge discussed in the literature is through storytelling (HEYLIGHEN, MARTIN, et al., 2007). Stories that are engaging and easy

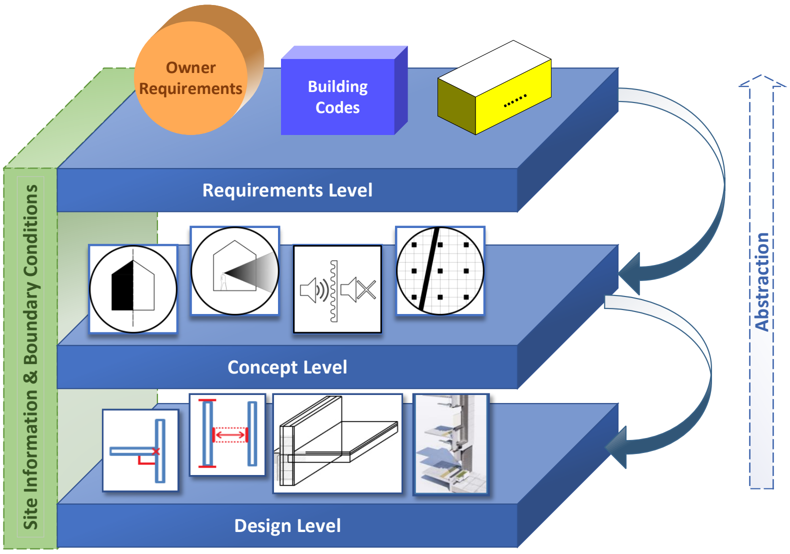


Figure 6.1: Construction projects' design abstraction levels based on the methodology and introduced concepts of this paper are envisioned. The design process of a construction project takes into account the surrounding boundary conditions, requirements, and regulations in order to apply specific design concepts, followed by design constraints (Abualdenien & Borrmann, 2021).

to understand, are especially useful for sharing individual tacit knowledge. Although it may not transfer huge amounts of information, it is a means of catalyzing understanding. In addition to the benefits of using narrative, storytelling is non-adversarial and non-hierarchical, providing an opportune breach in the defensive nature of the creative work that architecture is, where ideas and outcomes are essential in terms of ownership and recognition. Storytelling is not a replacement for rigorous analytical thinking, but it complements our understanding of a phenomenon by bringing alternative perspectives and worldviews into play. Storytelling also allows for multiple issues of importance to be addressed in terms of complexity in architectural design. In addition to stories being direct, easy to read, and entertaining, they respect the intricate relationship of things, making them quite memorable. Therefore, storytelling permits a dense and compact way of communicating complexity in a short time. The stories' outcomes cast ownership onto the reader by connecting the story to their personal experience. The outcome is irrelevant to the fact but relates more to the ideas, processes, decisions, and implications of the interactions demonstrated within the story. The potential of storytelling for capturing and storing the tacit design knowledge is proven effective in the "Building Stories" project, developed and run by Berkeley University in California, with support from some leading architectural companies in the San Francisco Bay Area. During this project, various teams of architectural students, interns, and professionals built and revised stories about some architectural projects that were being designed or had already been built (Martin et al., 2005). With this in mind, in this paper, we introduce the concept of design episodes (DEs) to divide and store various pieces and chapters of design. Each DE contains a name, ID, and textual description that explains the designer's intentions and clarification, together with the list of corresponding building elements and spaces that represent this situation. Within the framework of storytelling, we believe that DEs provide the ability to break down the overall design into essential components and features addressing unique project-specific challenges, and thus to effectively record and manage innovative and newly created design knowledge. More demonstrative examples for DEs are discussed in section 6.4.

Meanwhile, what are the expectations of architects who are willing to document and justify their design decisions while designing? The thought process of designers is both graphical, as it works through, in, and with images, as well as textual, e.g. engineering numbers and linguistic words, creating a silent dialogue using elements similar to all other visual artists (Cross, 1982). Discarding, selecting, and further detailing architectural design decisions and variants depends not only on objective (quantitative) criteria but also on subjective (qualitative) criteria. In addition to building model and quantitative criteria, qualitative and descriptive (sometimes episodic) assessments and evaluations are necessary for documenting the selection of variants in order to make the decisions made and their justifications, e.g. the architectural quality, comprehensible and to support the interpretation of the architectural solution. The goal is to store and document design decisions and variants selection without significantly interrupting the design process. With this in mind, a collection of so-called Explanation Tags (ETs) is offered to the architects to choose from while designing inside a BIM authoring tool, enabling them to argue and justify their design decisions by assigning these ETs to building components and spaces or to their specific attributes. More clarifications on how to use the ETs will be discussed in sections 6.3.2 and 6.4. This open-ended collection aims to represent a graphical codification of architectural terms, inspired by major theoretical architectural publications and empirical guidelines. It is important to mention that this collection of ETs is not limited to what is presented in this paper as a set of examples and can be extended by new users and domain experts based on their needs. Furthermore, it should be noted that the collection and provision of these ETs are not the main focus of this paper, but rather the framework in which these tags could be expanded and offered to the designers is of importance and among the contributions of this paper.

Our first selection of ETs was based on SNAP (Systematik für Nachhaltigkeitsanforderungen in Planungswettbewerben) (Fuchs et al., 2013). SNAP was developed under the Federal Ministry of Transport and Digital Infrastructure in Germany. Likewise in Switzerland, the Swiss SNARC methodology "Systematik zur Beurteilung der Nachhaltigkeit von Architekturprojekten für den Bereich Umwelt" (Schweizerischer Ingenieur- und Architekten-Verein, 2004) was developed for use in competition procedures. We then expanded our collection of explanation tags using some other related work and architectural literature (Janson & Tigges, 2014; Neuckermans et al., 2002). Each Explanation Tag (ET) is represented with an icon and stored together with an ID, name and textual description, and sometimes graphical explanatory examples, such as photos, plans & sections, 3D models, and partial BIM models. Using NLP techniques and domain-expert-knowledge, the tags are also cross-connected via meta-data markers (in the back-end of the system) through a series of overlapping meanings such as synonyms, antonyms, complementary, related, or associated meanings, which will be used for suggestions and recommendations to help the architects upon using them. Since it is almost impossible for us to collect all the terms and criteria for the whole architectural domain, due to its complexity and variability for different projects and experts, only an exemplary collection of ETs is presented and used in this paper. However, our system design guarantees extensibility, and new ETs can be added to this collection. The open-ended aspect of our system allows some experienced and knowledgeable users to create their own ETs and enhance the vocabulary of architectural terms.

Our collection of ETs along with their definitions and in some cases, best practice suggestions, and examples for them are in Appendix A. Explanation Tags at the end of this paper. This collection contains both subjective (qualitative) and objective (quantitative) design criteria. To differentiate between these two categories, the icons for the subjective ones are enclosed inside a circle frame, whereas the icons for the objective ones are framed inside a box. Table 1 shows two of these ETs. Once again, it is crucial to note that this collection of ETs, in Appendix A. Explanation Tags, is by no means complete and is subject to improvements and enhancements. However, the introduced concept of Explanation Tags and the presented framework in which new ETs could be added guarantees the adjustability and expansibility of our system, and is of importance to this paper and among its contributions.

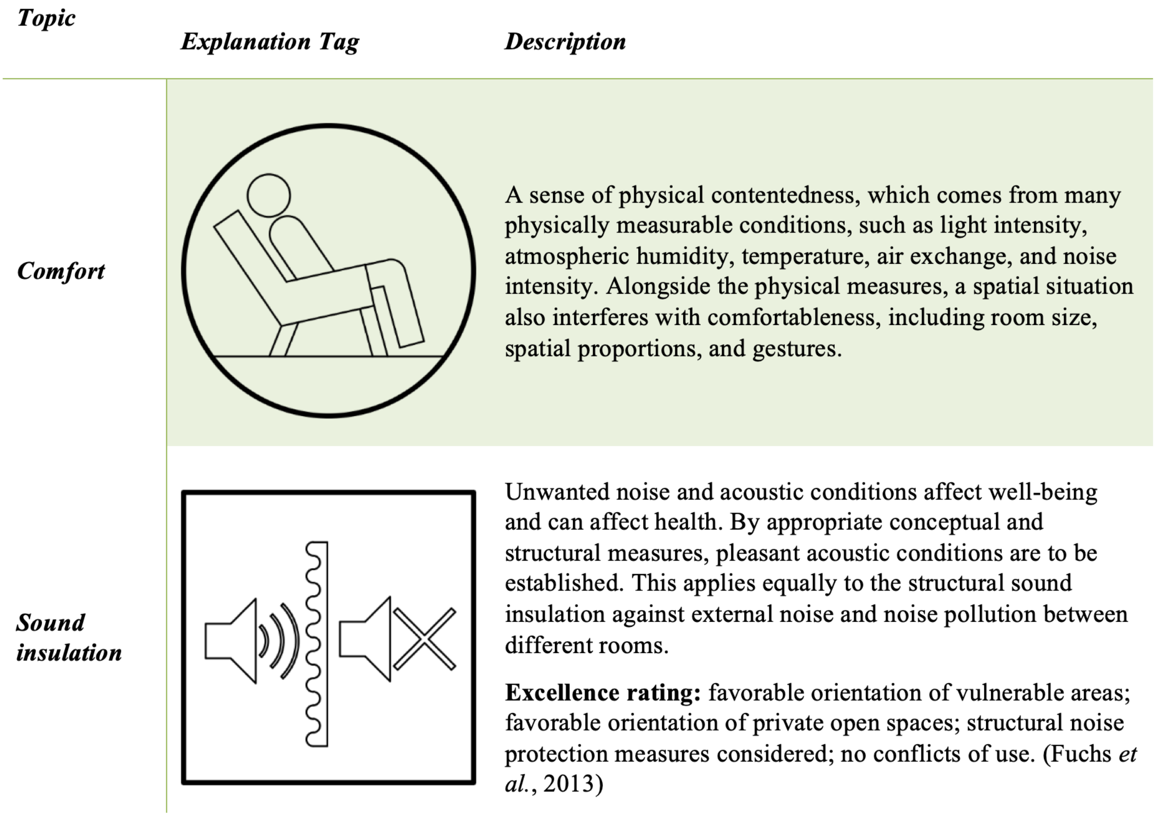
The way our concept for documenting design decisions works will be discussed in greater detail using some demonstrative examples and use cases. In a nutshell:

- The designers can split the overall design into multiple design episodes and explain their intentions and solutions for different design challenges using storytelling techniques.
- They can assign explanation tags to different building components and spaces or their specific attributes to mark and clarify the reasons and goals for different design decisions graphically and in more detail.
- They can set up constraints, which will be explained next in section 6.3.2, to make sure some design aspects and decisions will be kept intact and unchanged as the design process moves forward and the design model is further developed.

### 6.3.2 Design constraints: Multi-LOD meta-model

Explicitly specifying design requirements and constraints could support documenting design intentions and decisions, especially during early design stages. Additionally, such constraints could be checked to verify and confirm that design decisions are still being maintained. In this paper, we propose two kinds of constraints, geometric and semantic. Figure 2 illustrates the

Table 6.1: Two of the Explanation Tags related to subjective and objective criteria.



| ***Topic*** | ***Explanation Tag*** | ***Description*** |
|---|---|---|
| ***Comfort*** | | A sense of physical contentedness, which comes from many physically measurable conditions, such as light intensity, atmospheric humidity, temperature, air exchange, and noise intensity. Alongside the physical measures, a spatial situation also interferes with comfortableness, including room size, spatial proportions, and gestures. |
| ***Sound insulation*** | | Unwanted noise and acoustic conditions affect well-being and can affect health. By appropriate conceptual and structural measures, pleasant acoustic conditions are to be established. This applies equally to the structural sound insulation against external noise and noise pollution between different rooms.<br>**Excellence rating:** favorable orientation of vulnerable areas; favorable orientation of private open spaces; structural noise protection measures considered; no conflicts of use. (Fuchs *et al.*, 2013) |

concept behind the geometric constraints. Each face of the individual elements is represented by its center point, which is used to describe the connection constraint among multiple elements. The constraint can refer to the face center point in addition to a directional anchor (e.g., top, left, etc.) and a numerical padding to provide the necessary flexibility. To describe the spatial constraint between two elements, the distance and the degree are captured. On the other hand, semantic constraints are focused on specifying the permissible property values in multiple ways (explained in detail in this section).

In practice, it is necessary to explicitly specify which information is reliable and estimate the accuracy of the unreliable information at a specific LOD; an LOD is depicted as a milestone for making design decisions. Consequently, precisely defining the LOD requirements while incorporating their uncertainty improves the quality of the collaborative process among the disciplines. Managing information on multiple LODs requires both representing the building elements on different LODs as well as providing the ability to specify the required information on each LOD in a formal way. The multi-LOD meta-model fulfills these requirements by supporting the following activities (Exner et al., 2019):

- Formal specification of the overall information requirements at a particular design stage.
- Formal specification of the individual elements' LOD definitions.

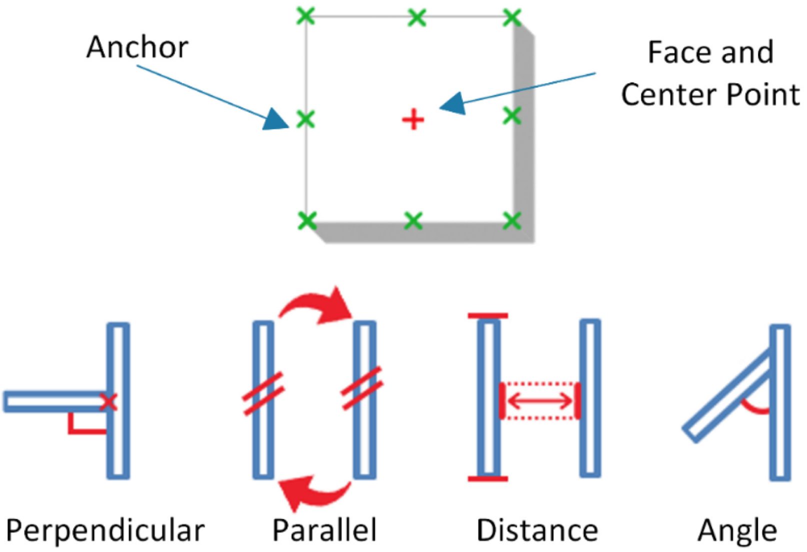


Figure 6.2: The proposed approach for capturing spatial constraints between building elements, using the distance and angle between the elements as well as vertical and horizontal anchors and padding.

- Formal incorporation of the potential vagueness.
- Representation of the building models' instances at different design stages.
- Verification of building models consistency across the design stages, i.e., ensuring that the decisions made in one stage are respected in the subsequent stage. The meta-model introduces two levels: data-model level, which defines the component types' requirements for each LOD, and instance level, which represents the actual building components and their relationships. To ensure the model's flexibility and applicability, its realization is based on the widely adopted data model Industry Foundation Classes (IFC). The IFC model specification is an ISO standard, which is integrated into a variety of software products (Liebich, 2013). More specifically, entities from the meta-model are linked to existing IFC entities and then provide extensions, including component types, properties, relationships, and geometry representation. This makes it possible to attach requirements, vagueness, constraints, and documentation.

In more detail, each component type is linked to an IFC type, IfcColumn as an example, and associated with multiple LOD definitions. An LOD definition consists of geometric and semantic requirements, specifying the required geometry representation and properties. The details of each property are determined in addition to the permissible vagueness. In terms of vagueness, a property can be assigned to a vagueness type (classification or probability distribution), a maximum vagueness percentage, and whether the vagueness values are expected to be a range. The vagueness values at the instance level are automatically generated from the vagueness definition specified at the data-model level. For example, in case the vagueness type

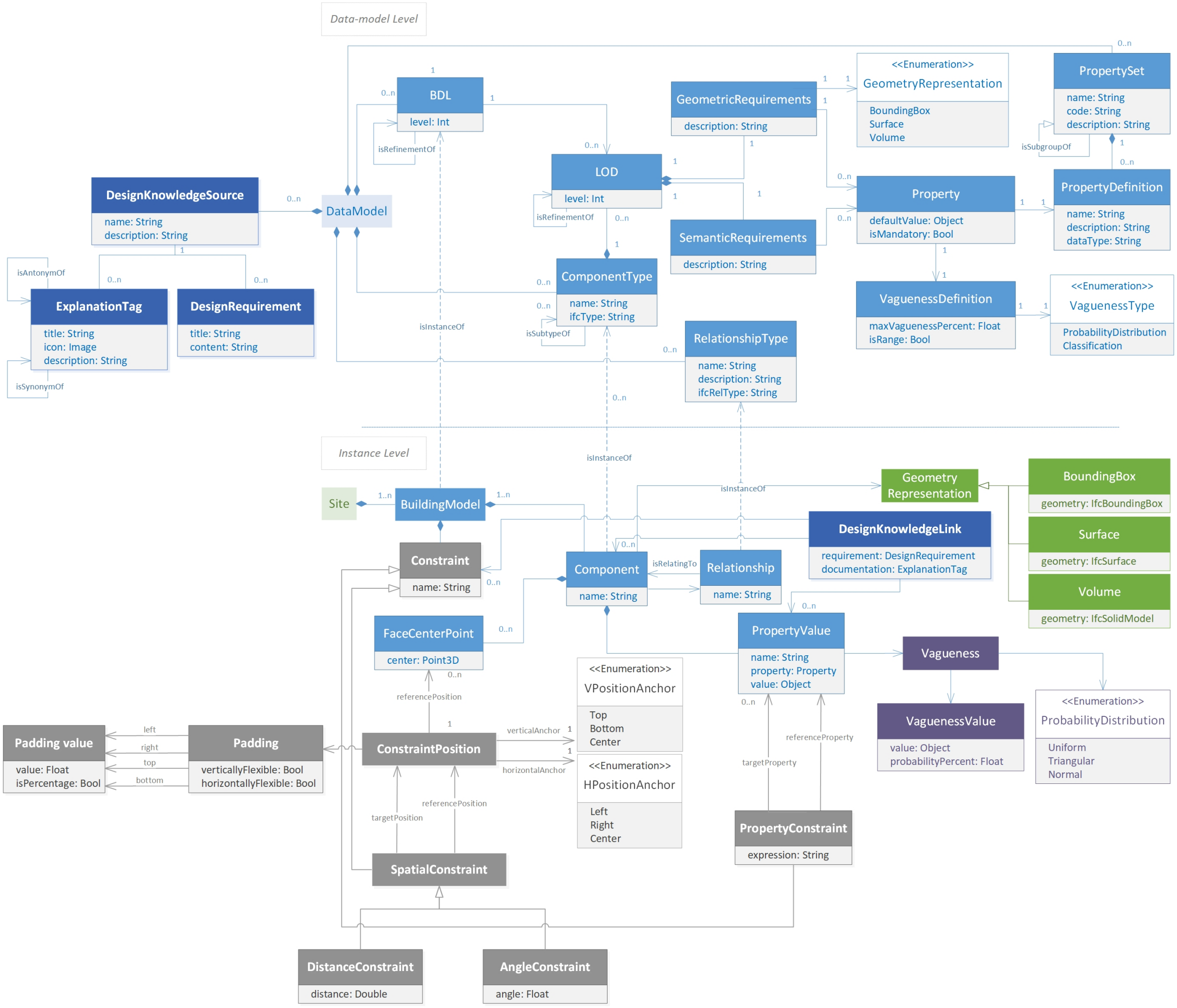


Figure 6.3: Multi-LOD meta-model (UML class diagram): describing design requirements, knowledge at the data-model level, and representing building elements, including properties uncertainties, constraints, and documentation at the instance level. The purpose of color-coding the boxes is to be able to distinguish between the entities that belong to knowledge, requirements, constraints, geometry, and vagueness.

is a probability distribution, the vagueness percentage is 4%, and the attribute value is 250 cm, the vagueness values are generated to form a range of ±20 cm. Moreover, at the instance level, it is possible to increase the limitation of the range values, such as to be between -5 and +7 cm. A comprehensive explanation and evaluation of the multi-LOD meta-model approach are available in (Exner et al., 2019).

The constraints concept is implemented in the meta-model as shown in Figure 3. Accordingly, in this paper, the meta-model design is extended to incorporate the documentation of design decisions and constraints. In more detail, the data-model level is extended to allow defining design knowledge in three forms, explanation tags (discussed in Section 6.3.1), design requirements (which can contain the RFP requirements or building code provisions), and design episodes (also discussed in Section 6.3.1). At the instance level, ETs, requirements, and DEs can be assigned to describe components, property values, and constraints. This way, the reason behind using a particular property value or constraint is documented. The meta-model supports two main constraint types: *SpatialConstraint* and *PropertyConstraint*. The *SpatialConstraint* comprises two children: *DistanceConstraint* and *AngleConstraint* for describing the spatial constraints between multiple elements. Each of these spatial constraints is assigned to a vertical and horizontal anchor as well as four padding values. In the same context, the *PropertyConstraint* allows limiting a reference property with a specific value (e.g., length <= 2m) or the value of one or more properties (e.g., wall1.length = wall2.length).

While constraints are mainly used to maintain design decisions throughout the design phases, explanation tags and design episodes are largely used for documenting and explaining the design decisions as comprehensively as possible. To use an analogy from software programming and design, constraints in our concept are frameworks and blueprints to keep the further detailing and maturation of design decisions in line with previously discussed and decided fundamental decisions, whereas, using the same analogy, ETs and DEs are like commenting the code while programming so that it would be understandable later on.

### 6.3.3 Linking owner requirements, building codes, and Design Episodes to design decisions using natural NLP

Owner requirements, building codes, and design episodes' descriptions are in plain natural text. However, typically, these documents are the reason behind many of the decisions that are made, such as parameters' values or even constraints. Therefore, in this section, we present an approach for extracting these requirements using NLP techniques and for storing a link between these textual definitions and the different elements, their properties, and design constraints. As shown in Figure 4, first, the natural text is preprocessed by organizing it in a tabular format, providing a clear definition in each row. In this research, each row includes a specific building code provision along with its section and chapter titles. Then each row of these requirements is processed using NLP techniques, including tokenization, lemmatization, part of speech, and vectorization. Figure 5 demonstrates an example of processing a rule from the international building code into tokens, then lemmatization, part of speech, and finally the vector representation (which represents every row in a vector space of 300 dimensions). In this paper, we use the open-source NLP neural network spaCy (Honnibal & Montani, 2017) (Honnibal and Montani, 2017), which offers state-of-the-art accuracy in multiple languages (Colic & Rinaldi, 2019). We use the pre-trained large model of spaCy, which includes over one million unique vectors (SpaCy, 2021).

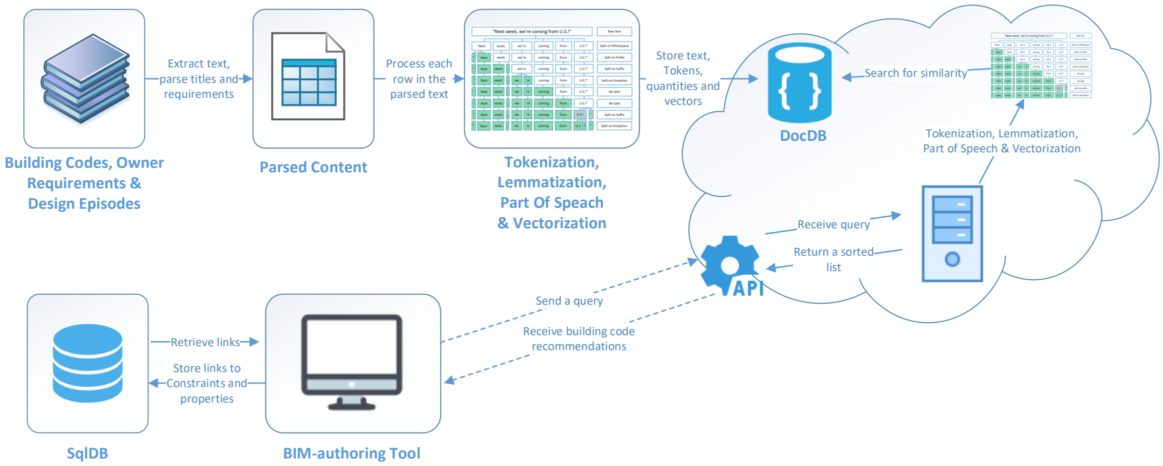
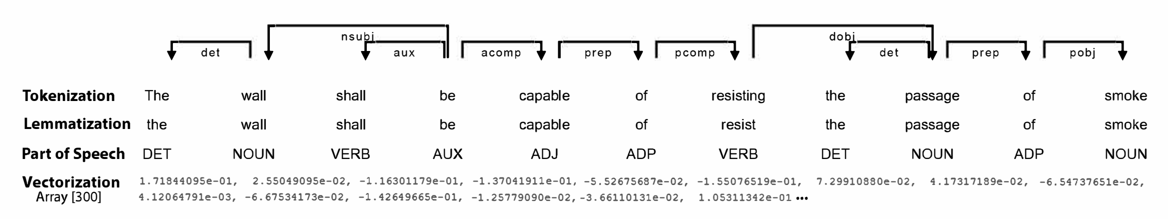


Figure 6.4: NLP integration approach: the incorporation of NLP during the design process to facilitate querying and linking the individual requirements to the different building elements and their corresponding properties.



| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Tokenization** | The | wall | shall | be | capable | of | resisting | the | passage | of | smoke |
| **Lemmatization** | the | wall | shall | be | capable | of | resist | the | passage | of | smoke |
| **Part of Speech** | DET | NOUN | VERB | AUX | ADJ | ADP | VERB | DET | NOUN | ADP | NOUN |

**Vectorization** Array [300]

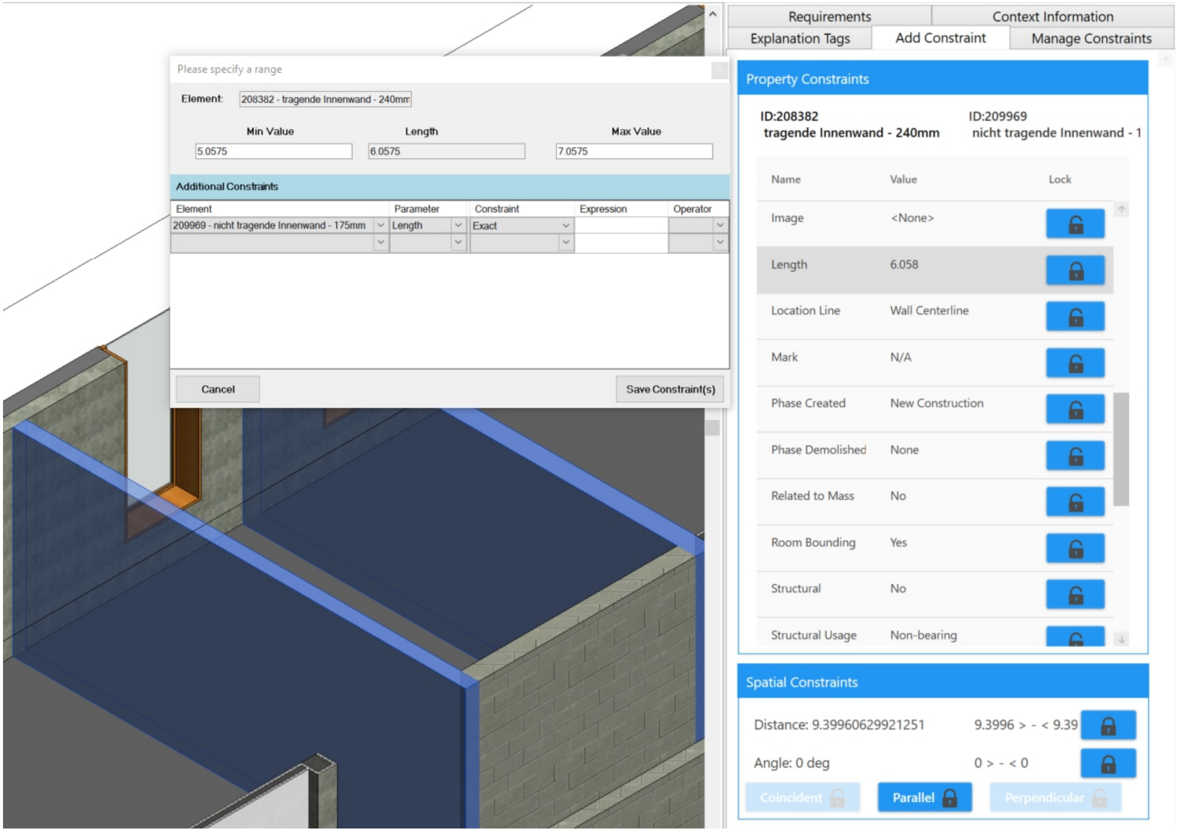
```
1.71844095e-01,  2.55049095e-02, -1.16301179e-01, -1.37041911e-01, -5.52675687e-02, -1.55076519e-01,  7.29910880e-02,  4.17317189e-02, -6.54737651e-02,
4.12064791e-03, -6.67534173e-02, -1.42649665e-01, -1.25779090e-02, -3.66110131e-02,  1.05311342e-01 •••
```

Figure 6.5: NLP processing example: a rule from the international building code is processed through multiple steps, tokenization, lemmatization, and then the extraction of POS tags. Finally, a vector representation of the complete rule is generated to support comparing rules for similarity to a search query.

The original text as well as the processed content is stored in a document database for future query and use. Afterward, from the BIM-authoring tool, users can query the stored requirements and link a specific requirement to one or multiple properties, or to existing semantic and geometric constraints. Going into greater detail, the BIM-authoring tool communicates with a server through a REST API and sends a query. Then, this query is also processed using the same NLP techniques and compared with similarity to the vector representation available in the document database. According to the state of the art in NLP (SpaCy, 2021), the cosine similarity is the most popular similarity measure when comparing vector representations. Next, the top 10 requirements, sorted by their similarity percentage, are displayed to the user in the BIM-authoring tool.

### 6.3.4 Proof of concept

This section discusses an introduction to the implementation part as proof of concept, followed by three demonstrative use cases in section 6.4 to illustrate exactly how our concepts and implementations work together. The presented approach is implemented as a plugin inside Revit. When the user selects one or multiple elements, the plugin will display their properties as well as possible spatial constraints, including their corresponding distance and angle. For each item of information shown, the user can add constraints according to the concept introduced by the meta-model (Section 6.3.2). Figure 6 shows an example of two staircase walls. The lock icons indicate whether a constraint is added or not. Spatial constraints are added here, where the elements must be always Parallel to each other, which is described by a distance and an angle of zero degrees. Additionally, the length property of this particular wall is constrained within a specific range and linked to be exactly the same length as the other wall.



Element: 208382 - tragende Innenwand - 240mm

| Min Value | Length | Max Value |
|---|---|---|
| 5.0575 | 6.0575 | 7.0575 |

| Element | Parameter | Constraint | Expression | Operator |
|---|---|---|---|---|
| 209969 - nicht tragende Innenwand - 175mm | Length | Exact | | |

ID:208382
tragende Innenwand - 240mm

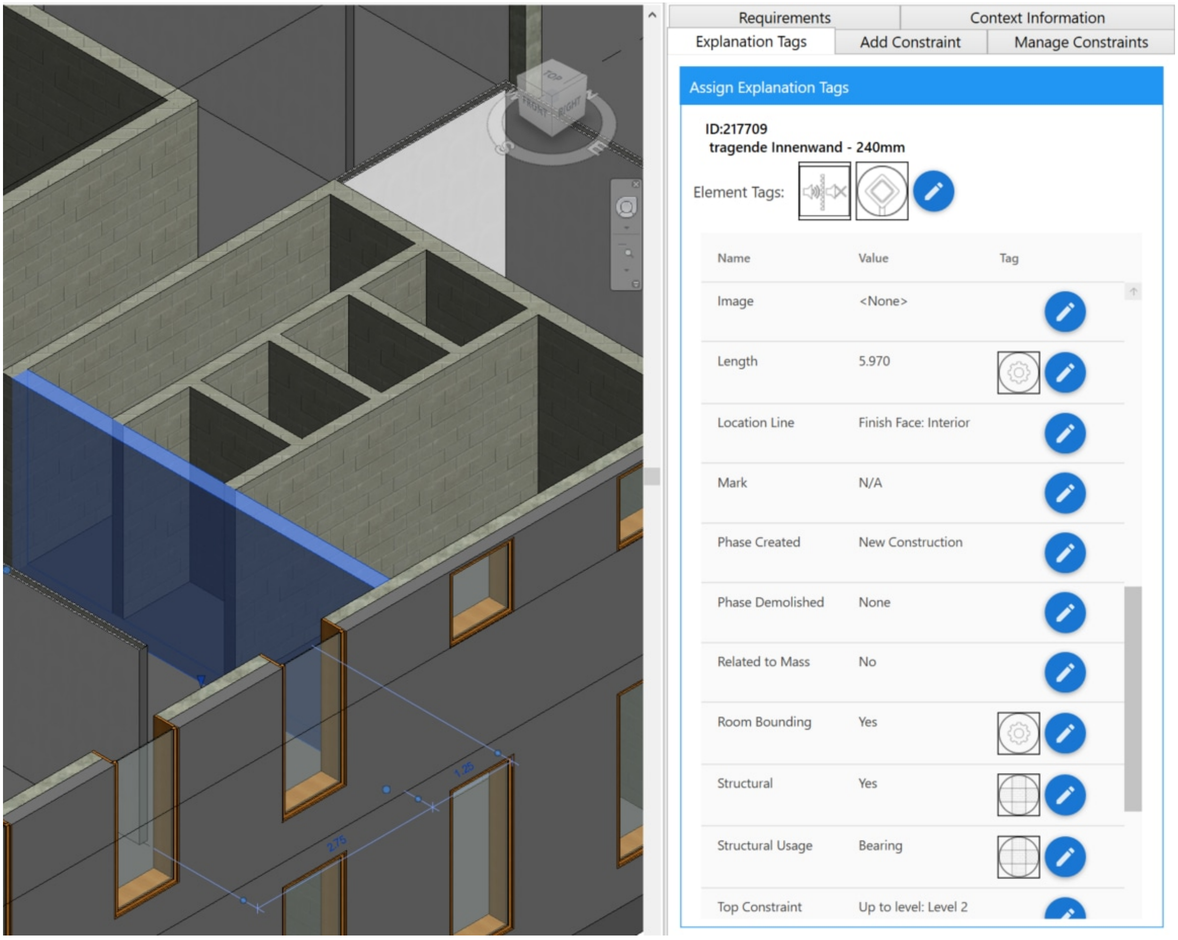
ID:209969
nicht tragende Innenwand - 1

| Name | Value | Lock |
|---|---|---|
| Image | <None> | 🔓 |
| Length | 6.058 | 🔒 |
| Location Line | Wall Centerline | 🔓 |
| Mark | N/A | 🔓 |
| Phase Created | New Construction | 🔓 |
| Phase Demolished | None | 🔓 |
| Related to Mass | No | 🔓 |
| Room Bounding | Yes | 🔓 |
| Structural | No | 🔓 |
| Structural Usage | Non-bearing | 🔓 |

Distance: 9.39960629921251 9.3996 > - < 9.39 🔒
Angle: 0 deg 0 > - < 0 🔒

Figure 6.6: Revit plugin prototype: an example of adding spatial and semantic constraints on two walls of a staircase.

Similarly, the user can assign one or more explanation tags to the selected elements or their individual properties. Figure 7 demonstrates the concept on a load-bearing wall that is bounding a server room. Two tags were assigned to the element: (1) Sound insulation, describing that this is an important characteristic for avoiding the workspace disturbance, and (2) Safety & Security, raising the consideration for fire-safety regulations or electrical hazards. On the other hand, the Functionality tag is assigned to the properties Length and Room Bounding, and the Material tag was assigned to the Structural Usage property, highlighting their importance for providing efficient management of the space as well as serving the intended functionality expected from this particular element (all tags are described in the appendix in detail). Furthermore, the process of assigning ETs to BIM elements or their individual properties is identical for all objects, spatial or physical the same. Finally, to document the design according to its fulfillment in terms of requirements and to store a particulate DE, the Requirements tab facilitates querying the document database with natural text with the help of NLP. When a query is entered, it is then sent to a server, where it is processed and compared with the requirements database in terms of similarity. The results ordered by similarity are then displayed to the user.



ID:217709
tragende Innenwand - 240mm

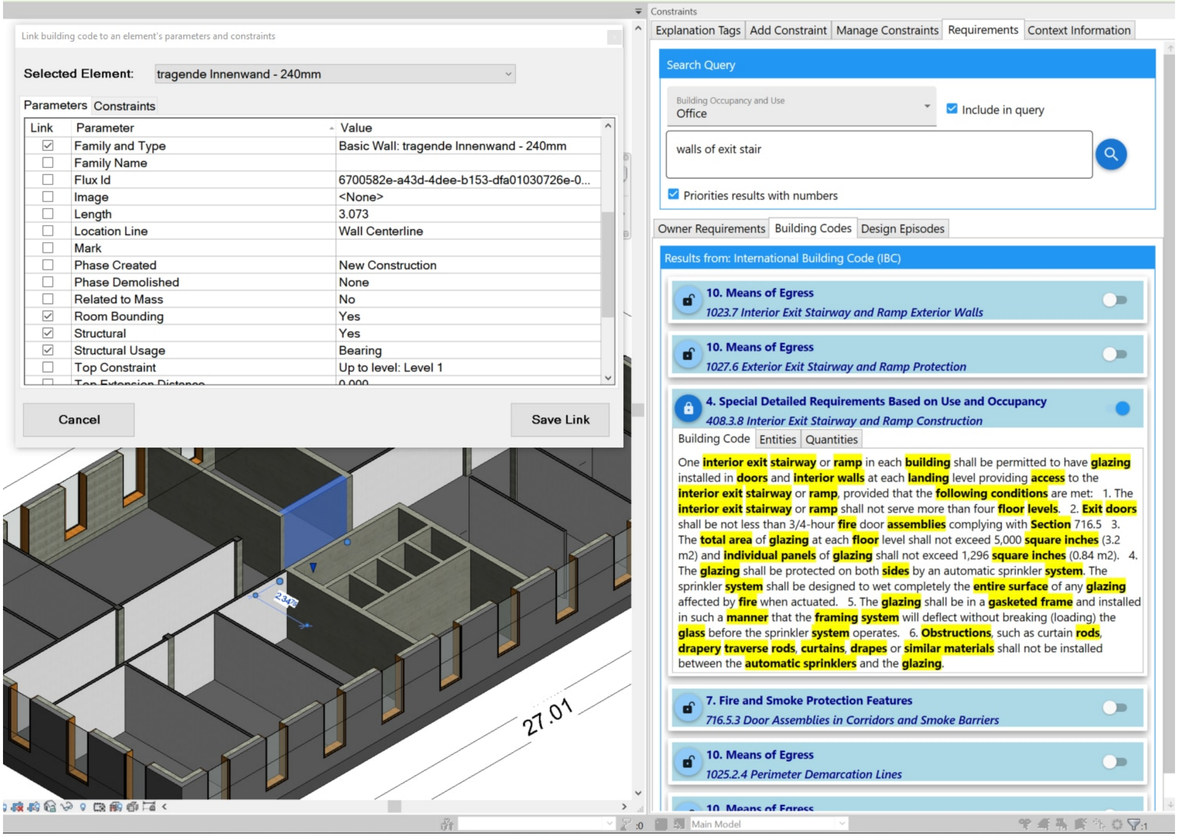
Element Tags:

| Name | Value | Tag |
|---|---|---|
| Image | <None> | |
| Length | 5.970 | |
| Location Line | Finish Face: Interior | |
| Mark | N/A | |
| Phase Created | New Construction | |
| Phase Demolished | None | |
| Related to Mass | No | |
| Room Bounding | Yes | |
| Structural | Yes | |
| Structural Usage | Bearing | |
| Top Constraint | Up to level: Level 2 | |

Figure 6.7: Revit plugin prototype: an example of adding explanation tags to a load-bearing wall that is separating a server from the working space.

Figure 8 shows an example, where a query with walls of exit stair is entered for searching the international building code. Additionally, the building occupancy and use was assigned to Office and also the building code provisions with numbers were given a higher priority. Accordingly, the returned results are mainly concerned with the Means of Egress, stairs role for occupancy, and fire safety aspects, which is compliant with the requested query. The results panel shows the stored requirements natural text with the nouns highlighted to help the user identify which entities and properties this code is describing. Additionally, there are two additional tabs, one lists the entities/nouns and the other quantities, for example, 60 feet (18,288mm).

Furthermore, as shown in Figure 8, the user can link a particular requirement to specific properties or constraints. Once the user clicks on the open lock icon, a dialog will pop up showing the selected elements and listing the corresponding properties and constraints. At this point, the user is capable of assigning the building code to one or multiple values. This kind of linkage provides additional reasoning for the corresponding values, which designers could refer to when they consider different values. To demonstrate the performance of the developed approach, a screencast was captured and published online.



Selected Element: tragende Innenwand - 240mm

| Link | Parameter | Value |
|---|---|---|
| ☑ | Family and Type | Basic Wall: tragende Innenwand - 240mm |
| ☐ | Family Name | |
| ☐ | Flux Id | 6700582e-a43d-4dee-b153-dfa01030726e-0... |
| ☐ | Image | <None> |
| ☐ | Length | 3.073 |
| ☐ | Location Line | Wall Centerline |
| ☐ | Mark | |
| ☐ | Phase Created | New Construction |
| ☐ | Phase Demolished | None |
| ☐ | Related to Mass | No |
| ☑ | Room Bounding | Yes |
| ☑ | Structural | Yes |
| ☑ | Structural Usage | Bearing |
| ☐ | Top Constraint | Up to level: Level 1 |

Building Occupancy and Use: Office

☑ Include in query

walls of exit stair

☑ Priorities results with numbers

Results from: International Building Code (IBC)

**10. Means of Egress**
*1023.7 Interior Exit Stairway and Ramp Exterior Walls*

**10. Means of Egress**
*1027.6 Exterior Exit Stairway and Ramp Protection*

**4. Special Detailed Requirements Based on Use and Occupancy**
*408.3.8 Interior Exit Stairway and Ramp Construction*

One **interior exit stairway** or **ramp** in each **building** shall be permitted to have **glazing** installed in **doors** and **interior walls** at each **landing** level providing **access** to the **interior exit stairway** or **ramp**, provided that the **following conditions** are met: 1. The **interior exit stairway** or **ramp** shall not serve more than four **floor levels**. 2. **Exit doors** shall be not less than 3/4-hour **fire** door **assemblies** complying with **Section** 716.5 3. The **total area** of **glazing** at each **floor** level shall not exceed 5,000 **square inches** (3.2 m2) and **individual panels** of **glazing** shall not exceed 1,296 **square inches** (0.84 m2). 4. The **glazing** shall be protected on both **sides** by an automatic sprinkler **system**. The sprinkler **system** shall be designed to wet completely the **entire surface** of any **glazing** affected by **fire** when actuated. 5. The **glazing** shall be in a **gasketed frame** and installed in such a **manner** that the **framing system** will deflect without breaking (loading) the **glass** before the sprinkler **system** operates. 6. **Obstructions**, such as curtain **rods**, **drapery traverse rods**, **curtains**, **drapes** or **similar materials** shall not be installed between the **automatic sprinklers** and the **glazing**.

**7. Fire and Smoke Protection Features**
*716.5.3 Door Assemblies in Corridors and Smoke Barriers*

**10. Means of Egress**
*1025.2.4 Perimeter Demarcation Lines*

Figure 6.8: Revit plugin prototype: an example of querying the international building code and linking a rule to a specific wall and its properties.

## 6.4 Demonstrative use cases

This section provides three different use cases, two of which are real-world building projects, to demonstrate the use and applicability of our proposed concepts and approaches. The first use case is based on our own hypothetical design to which we applied our approach and concepts during development. The other two use cases, however, are real building projects that are analyzed and used for our purposes, in terms of the intentions of the designers and the arguments for their design decisions, after the design is complete and they have already been built.

### 6.4.1 Use case no. 1 – open living and dining room

The first use case demonstrates an example for a design episode where the following paragraph could be viewed as the episode description where the designer has written to explain the design's intent. "In this big living room, the intention is to preserve openness and transparency, while separating the dining area from the living area (which could also be used as a TV room). Clear visual contact between the two sub-spaces is another goal. To achieve that in this floor plan, elevation is used as means of space division and transition, while conserving the continuity and transparency of the two inter-connected sub-spaces. This way, the dining area is separated from the living area inside the (big) living room. The aim is to create a virtual division of spaces while preserving the continuity of the one big living room, which provides a sense of openness and transparency.

On the other hand, the use of an exposed-brick wall in this floor plan presents a personal style in the design, which contributes to the aesthetics or pleasing qualities of design in visual terms. In this case, an exposed brick wall brings an appealing contrast to the other white walls and imposes a warm atmosphere and a tasteful transition into the living area. This will also enhance the acoustics in the living area."

### 6.4.2 Use case no. 2 – concrete house by Carl-Viggo Hølmebakk in Norway

This use case is developed according to a plan from the Concrete Lake House by Carl-Viggo Hølmebakk (Concrete House - Carl-Viggo HAS, Stange, Norway, completion in 2015; Carl-Viggo Hølmebakk AS, founded in 1990). The following design episode is a summary of Mr. Hølmebakk's opinion about this design taken with his permission from his website ("Concrete House - Carl-Viggo Hølmebakk AS", Stange, Norway, completion in 2015).

"Although the grand view played an important role in the design, the facade is not fully glazed, but rather "masked out" with varying openings that are positioned and sized with the treatment in mind of natural light, exterior views, and the intended use of interior spaces.

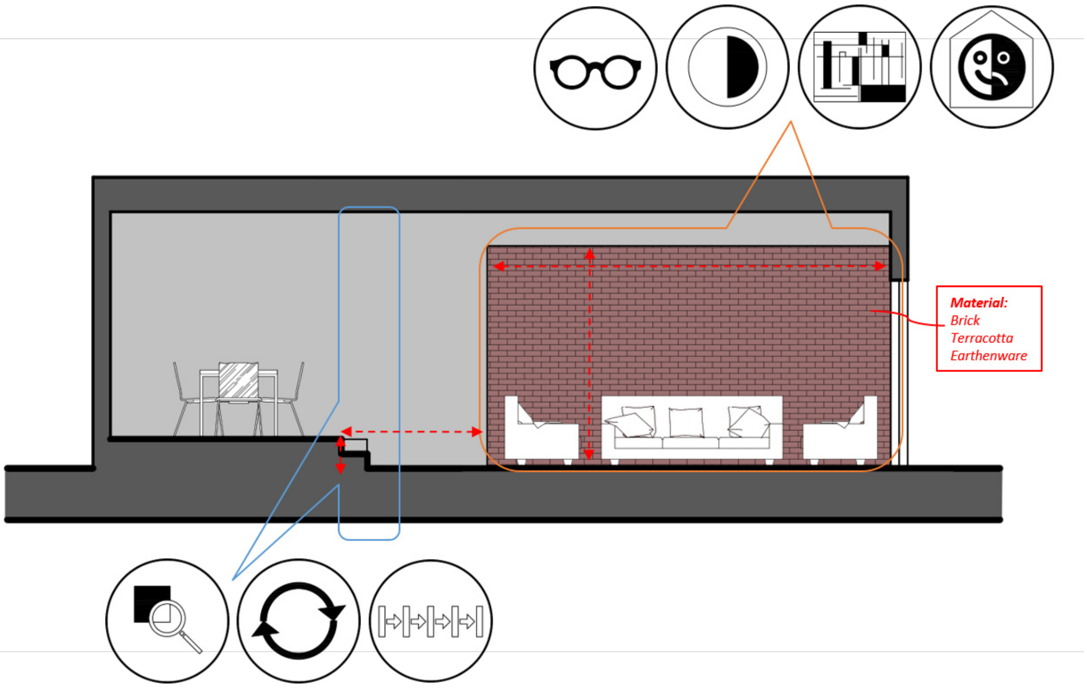


Figure 6.9: Use case no. 1 - showing the ETs of Transparency, Continuity, and Transition on the elevation steps and Personal Style, Contrast, Aesthetics and Atmosphere on the exposed brick, together with Material and Dimensional constraints for both To document and communicate the design intentions for other project participants in a more graphical way, multiple ETs can be assigned to different components, as displayed in Figure 9. Two room labels for the dining area and living area (or the TV area) can separate the two sub-spaces. Multiple ETs, e.g., transparency, transition, and continuity of space in the overall living room, describing the openness, can be attached to the elevation of stairs between the two areas. The ETs can be then interpreted into geometric and semantic constraints to document the design in more detail. Accordingly, the proportion of each area can be restricted with dimensional constraints. Moreover, the elevation between both areas can be represented by a minimum height and number of steps that could keep the separation between the spaces tangible and at the same time keep them open to each other. On the other hand, labeling the exposed brick wall with ETs such as aesthetics, contrast, and acoustics will document the design rationale for this specific wall. The finish material layer of this particular wall influences the aesthetics greatly. The architect in this case could add a constraint for the permissible material layers, e.g., Brick, Terracotta, Earthenware, as well as their thickness.

Spaces could span two floors, and openings for daylight and views could be tailored to different rooms and situations. The load-bearing in-situ cast concrete also allowed for compelling constructions both in the exterior and in the interior, enabling cantilevering of staircases, roofing, terraces, galleries, etc. Exterior and interior staircases connect the different floors and different areas of the house. This adds to a complex pattern of spatial sequences and movement within a rather rationally executed organization: All living areas and bedrooms face the view, and are distributed over three floors. Secondary functions, such as bathrooms, lavatories, laundry rooms, etc., are located in the rear end of the house, where the facade is relatively closed. Several west-facing terraces protrude from the building, furthering the living spaces' relationship with the water and the view."

In this case, as is demonstrated in Figure 10, using ETs and constraints would be valuable for justifying and explaining the architectural concept and design intention. A special aspect of this design is the seemingly random distributed openings and windows on its facade that create the feeling of complexity and dazzle, which is best explained by using the complexity ET along with position and dimension constraints. However, the interior has a simple three-story layout with a rational organization that puts all the secondary functions at the rear of the house, while all the living areas and bedrooms have a great view. Describing this arrangement of spaces with the simplicity ET will enhance the design documentation. This house is a perfect case study for making a series of openings that bring in different angles of natural light and provide comfort for its residents. The architect here leverages the windows to selectively frame composed views from different perspectives and angles, which can be labeled using the Comfort and View ETs. This use case also shows the use of skylights to bring in natural lighting and solar gain to heat the room and foster the feeling of coziness when the sun shines. These windows could be tagged with daylight or natural lighting. Some small windows are used as ventilation panels, whereas, in another plan, an eccentric window is used to bring a well-diffused light into the bedroom, allowing the tenants to see the sky while in bed. Such a window could be tagged with view, natural lighting, and comfort. Each window has a location and a design with a unique intention in mind, one that is hard to capture in the regular design method but it could be done in our approach by adding constraints for the exact position and dimensions of each window. As illustrated in this plan, by using blended spaces and down-drops in some areas, the architect creates high ceilings and large dimensions and proportions, which ultimately creates the sense of immensity, spaciousness, and vastness for its inhabitants, with quality similar to that of a cathedral. This could be labeled using Immensity as an ET.

### 6.4.3 Use case no. 3 - Tausendpfund building in Germany

The Ferdinand Tausendpfund building in Regensburg is an office building erected at the end of 2016, which consists of three different exterior wall constructions. The building has a first floor and two upper floors with a gross volume of 3950 m3, a gross area of 1290.5 m2, and a window-to-wall ratio of 25%. The building does not have a basement, which is why the floor slab, the exterior walls, and the roof form the thermal building envelope. In the building itself, all zones are considered to be heated to normal temperatures (Vollmer et al., 2019).

The application of ETs is shown in Figure 11, whereas in this design, structural elements are mostly put in the outer walls or the core with vertical circulation and services in the center and only a few columns are left elsewhere, which creates spatial efficiency. This was

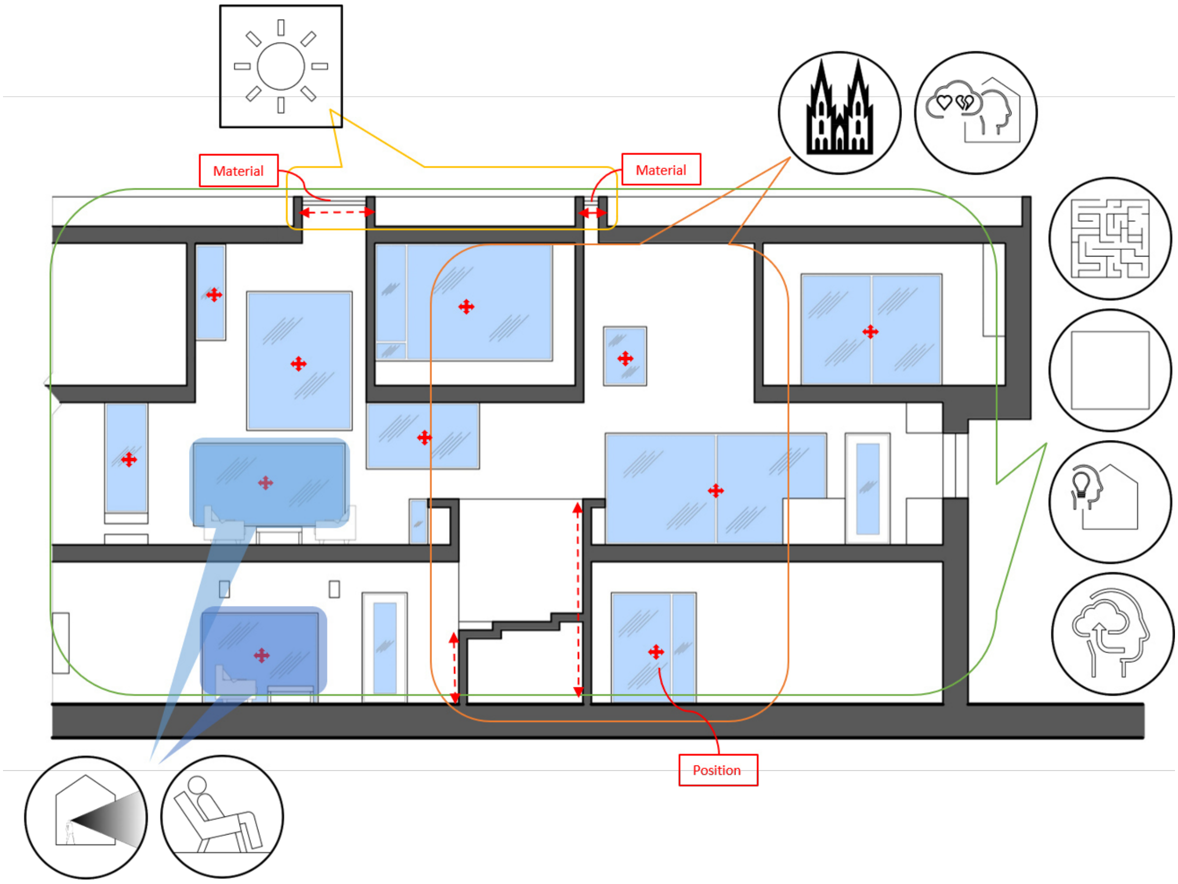


Figure 6.10: Use case no. 2 - recreated based on the facade section of the Concrete House by Carl-Viggo Hølmebakk (Concrete House - Carl-Viggo HAS, Stange, Norway, completion in 2015) illustrating the explanation tags (ET) for Daylight (solar gain) on the top two ceiling windows, the ETs of Immensity and Expression for the blended spaces, and high down-drops in the middle, the ETs of Complexity vs Simplicity, Concept, Experience, Comfort and View for the layout and organization of spaces and openings, together with the constraints for material and position of the windows

done according to the owner's requirements for making it possible to flexibly use the building design for both occupancy usages, as an office or residential building. This building is also thoughtfully designed considering criteria such as accessibility and barrier-free access, external space quality and spaces for social integration, etc.

The exterior walls of the building are built floor by floor in three different solid construction methods. The load-bearing material is reinforced concrete on the ground floor, thermal insulation bricks on the first floor, and sand-lime bricks on the second floor. In addition, a composite thermal insulation system is used as external insulation for the outer walls. The three exterior wall constructions each have approximately the same heat transfer coefficient (U-value) of 0.18 to meet the Effizienzhaus KfW55 standard (Vollmer et al., 2019). The floor slabs, the load-bearing interior walls, and the roof slab are constructed of reinforced concrete. Designing according to this kind of requirement demands careful consideration

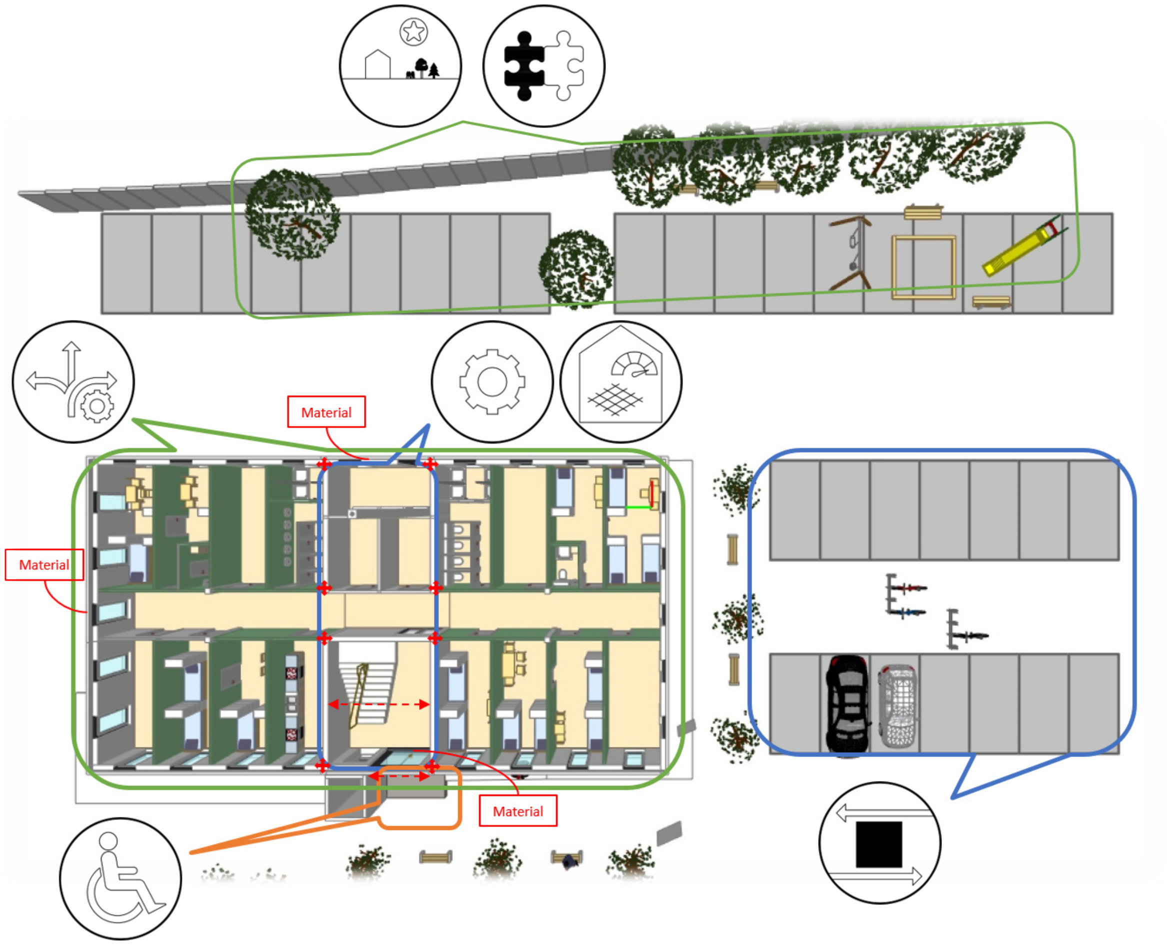


Figure 6.11: Use case no. 3 - Tausendpfund project, (Copyright Ferdinand Tausendpfund GmbH ("Ferdinand Tausendpfund GmbH & Co. KG", Stablished in 1892)) demonstrating the design ideas using ETs, namely the Use-flexibility tag for the arrangement of the structural elements, Functionality and Spatial efficiency tags for the wrapping and centralization of the core with vertical circulation and services, together with some constraints for material and position of these elements, some other tags include Accessibility for parking spots, the Barrier-free access for the ramp on the entrance door, External space quality and Spaces for social integration for the green space outside the main building.

of the various aspects of the design, which influence the design performance and embedded concepts. Accordingly, documenting which requirements were fulfilled using which design concepts is essential for communicating the design solution to the owner or the different domain experts involved in the project. As demonstrated in Figure 11, using explanations tags and constraints describe the designed concepts and helps the owners and domain experts understand the reasoning that went into the design.

## 6.5 Conclusion & future work

The construction industry is a knowledge-intensive sector that draws on a diverse set of skills from a variety of sources (Joe et al., 2013). For many years, the industry has amassed explicit information in the form of building codes, manuals, best practice guides, standards, processes, and so on. Furthermore, individuals with certain expertise and experience possess tacit knowledge. If a strategy to capture such knowledge is not established as people retire, many knowledge-intensive organizations will risk a constant loss of unrecoverable valuable knowledge (Calo, 2008). It's more difficult to formalize, maintain, and exchange this sort of knowledge. The master-apprentice relationship was and still is a common method of passing on tacit knowledge. There is also a broad gap between research and practice, which implies that vital knowledge is sometimes overlooked. This can lead to 'reinventing the wheel' or making the same mistakes over and over again. Architectural firms must adopt a systematic and consistent approach to design process documentation as construction gets more complicated and clients become more demanding. Documenting design knowledge, intentions, and decisions is a fundamental step for communicating with owners and domain experts. Additionally, it facilitates the future evaluation and re-use of completed projects, which can support decisions during the use and facility management of these projects as well as provide guidance when designing new projects. We believe that proper design documentation can lead to better reuse of design knowledge and experience, and optimize design decisions in current projects. From the authors' point of view, the design rationale contained in numerous projects is a precious and insightful source of knowledge that if captured and documented properly could be used and learned from to make better decisions.

BIM models have the potential to serve as procedural realizations of multidisciplinary knowledge, but currently, they store information rather than knowledge. Existing BIM models include raw geometries and semantics but lack any justification or explanation of design decisions. Existing methodology such as storytelling can help facilitate the transfer of design knowledge, however, a tool for documentation in this regard is missing for BIM authoring tools. In this paper, we tackled this problem and introduced novel solutions for it. We started by posing the question of how design decisions can be explained and digitally documented thoroughly based on existing conditions and assumptions. We introduced an innovative solution for the designers to express their motives and argumentation for numerous design decisions. The most remarkable result to emerge from this study is that a framework and meta-model is presented to encapsulate not just the details of design models but also the subjective justifications behind design decisions and choices (more details can be found in sections 6.3.1 & 6.3.2). Our study provides the blueprint for a new and holistic way to document the design process.

This paper presented a methodology that comprises multiple concepts to address this gap. First, explanation tags, as well as semantic and spatial constraints were introduced to capture the implemented design concepts and intentions. By applying explanation tags, the rationale and reasoning behind design decisions are captured and envisioned in a comprehensible and graphical way. It should be noted that while the proposed non-exclusiveness aspect of the explanation tag concept brings freedom to create and assign user-defined terms and descriptions, it should be advised to watch out for potential overuse of this feature that can increase the risk of semantic derivations, which in turn hinders the communication and reuse of design. Through the use of constraints, certain design details are laid down as frameworks that keep the integrity of design decisions as the design progresses. Furthermore, we introduced the concept of design episode to divide and store different parts of the overall design that each addresses a certain design challenge or task. By means of design episodes, different chapters of a design are described through storytelling that helps others understand the process and the reasons behind certain decisions. NLP techniques were then employed to query and link design requirements and episodes, which are in a natural text format, to one or multiple building elements, properties, or constraints. Such a link coupled with explanation tags enhances the design documentation with regard to both subjective (qualitative) and objective (quantitative) aspects of design.

The proposed methodology was evaluated for applicability via a prototype that was implemented as a plugin in Autodesk Revit. Additionally, the methodology was applied and discussed in the context of three use cases, which include two real-world projects. Accordingly, the use cases have shown the suitability of the proposed methodology for the current state of practice. For future research, the proposed methodology will be extended to support the search for and reuse of design knowledge across various reference projects and multiple design options. Further evaluations via user studies are intended to enhance the understandability and usability of the developed approach. Moreover, intensive and conclusive design documentation in sample projects, from start to end, is planned as future steps. In addition, our future research will focus on the reuse and utilization of the captured design knowledge for current and future design processes and projects. The captured design rationale will be queried and searched for, and for this purpose, the different BIM query languages will be evaluated for querying and filtering BIM models.

## Acknowledgments

We gratefully acknowledge the support of the German Research Foundation (DFG) for funding the project under grant FOR 2363 (DFG - FOR 2363: Evaluation of building design variants in early phases on the basis of adaptive detailing strategies). We also thank Ferdinand Tausendpfund GmbH (Ferdinand Tausendpfund GmbH & Co. KG, Established in 1892) for providing their office building as a sample project. Likewise, we appreciate the Carl-Viggo Hølmebakk (Carl-Viggo Hølmebakk AS, founded in 1990) for their Concrete House design.

## Chapter 7

# PBG: A parametric building graph capturing and transferring detailing patterns of building models

Previously published as: Abualdenien, J.; Borrmann, A.: *PBG: A parametric building graph capturing and transferring detailing patterns of building models*, In: Proc. of the CIB W78 Conference 2021, Luxembourg, 2021

## abstract

Design and detailing decisions result from numerous considerations and boundary conditions. Such decisions highly influence the cost and performance of the final design. Typically, architects and engineers tend to employ their domain knowledge and reuse successful Detailing Patterns (DPs) that fulfill the current needs and boundary conditions. DPs are described through building information and the rationale behind them. This paper presents a Parametric Building Graph (PBG) for capturing DPs. Additionally, it proposes a framework for automatically transferring DPs to new building projects. In more detail, DPs are stored as subgraph templates, and then when detailing a new building, a DP is matched and replaced across a graph representation of the building using GRS. Finally, the detailed building graph is brought back to the BIM-authoring tool. The paper is concluded with a feasibility study that demonstrates the realization of the proposed approach in a prototype and a use case.

## 7.1 Introduction

Building designs are wealthy with numerous implicit design decisions and domain knowledge. Every construction project must fulfill various owner requirements, regulations, as well as boundary conditions (Strug & Ślusarczyk, 2017). Accordingly, as depicted in Figure 7.1, satisfying these requirements is reflected within the selected architectural concepts (concept level).

The selected concepts are then realized through modeling and detailing the individual elements, including their geometric and semantic information as well as their topological relationships

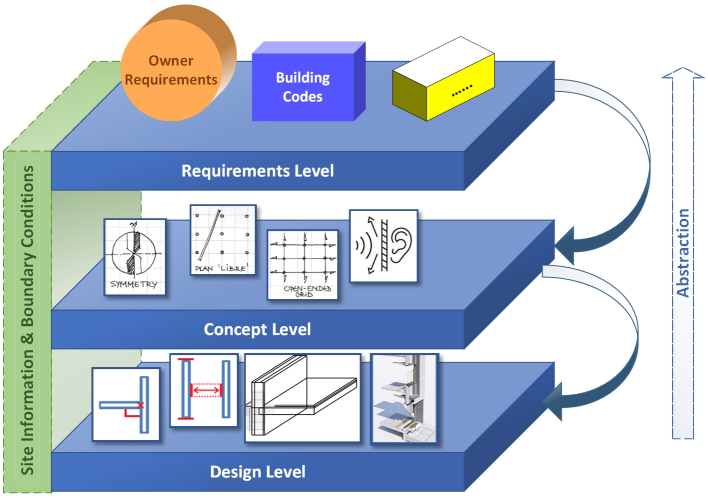


Figure 7.1: Illustration of the design process through abstraction levels

and functional dependencies (design level). For example, a site of a residential building close to a highway (where traffic is high) requires careful consideration of the designed facade, especially in terms of noise reduction techniques. On the other hand, a site facing a nature preserve fosters using curtain walls or big windows. Detailing decisions can be as simple as deciding on the position of a staircase or as complex as selecting the type of junction between walls and slabs, including choosing the combination of their material layers (Schneider-Marin & Abualdenien, 2019).

Detailing decisions significantly influence the performance of the resultant building design from various aspects, including energy efficiency, cost, and comfort. Hence, designers typically produce and detail multiple design options to explore and evaluate several possibilities at the different phases. Furthermore, although each construction project is unique in its context, designers tend to rely on their domain knowledge gained from previous successful projects, following a similar combination of building information and their dependencies for achieving similar function or performance. Examples of detailing patterns (DPs) can be the selected material layers of exterior walls from a specific side of the building, adding windows shading, or the type of joins between walls and slabs, which has a major impact on the transmission of thermal energy and sound (Châteauvieux-Hellwig et al., 2022). Detailing rationale includes the context information necessary to apply such DPs, such as the element's relative position to the storey's entrance and building's orientation (taking into account its sun path during the different seasons). Such domain knowledge is beneficial when detailing design options or designing new projects. However, currently, detailing decisions are embedded in building models, and detailing rationale is implicit in the designers' minds, hindering their proper management and reuse.

This paper introduces a parametric building graph (PBG) to capture DPs, including the geometric, semantic, topological relationships and the rationale behind them. Additionally, it proposes a framework for automatically transfer DPs from one design to another, using graph transformation systems. The development of the PBG was based on reviewing the currently existing graph representations in the Architecture Engineering and Construction (AEC) industry.

The paper is organized as follows: Section 7.2 discusses the background and related work. Section 7.3 presents the research methodology, categorizes existing graph representations in the AEC industry, and proposes a practical framework for transferring detailing patterns. A feasibility study is presented and discussed in Section 7.4. Finally, Section 7.5 summarizes our progress hitherto and gives an outlook for future research.

## 7.2 Background & related work

### 7.2.1 Graph representations in the AEC industry

For more than a decade, graph structures were used in the AEC industry for various use-cases, including path planning (Hamieh et al., 2020; Rüppel et al., 2010), retrieval of similar designs (Langenhan et al., 2013), integration of heterogeneous building models (A. H. Hor et al., 2016), and encoding or engineering knowledge (Vilgertshofer & Borrmann, 2017). Graphs structures are popular in the different domains due to their ability to represent complex relationships, which is the case in BIM (Isaac et al., 2013).

According to the graphs developed in the BIM domain, graphs include nodes representing building elements, in some cases their properties as well, and edges represent the relationships between them (Denis et al., 2017; Donato, 2017; Ismail et al., 2018; Khalili & Chua, 2015). Depending on the use-case, graphs could be as simple as raw nodes and edges or attributed, where nodes and edges hold attributes (key-value pairs). The existing graph representations will be discussed in more detail in Section 7.3, where a categorization of these efforts is provided.

### 7.2.2 Computational design synthesis and graph rewriting

The field of Computational Design Synthesis (CDS) aims to formally describe design knowledge. Graphs structures are computationally well supported and capable of describing modular product models. The concept of graph rewriting is described as a production system based on the combination of nodes and edges and their transformation rules (Helms et al., 2009).

Graph rewriting systems (GRS) are prevalent in capturing real-world and engineering design rules to synthesize design solutions (Chakrabarti et al., 2011). Rewriting systems are being investigated for more than a decade on formulizing design space of multiple domains, including mechatronic products (Helms et al., 2009), automotive powertrains (Helms & Shea, 2012), multi-scale shield tunnel products (Vilgertshofer & Borrmann, 2017), layout generation of architectural designs (Ruiz-Montiel et al., 2013) and evaluation of the connectivity of design solutions (Donato, 2017). Performing graph rewriting requires three main parts: an original graph, a transformation subgraph, and a set of logical rules that match a particular subgraph pattern and perform a set of operations, including altering, deleting, or replacing nodes, edges, and their attributes. The result is an updated graph, where each matched pattern from the original graph is modified according to the logical rules.

The proposed framework in this paper for transferring DPs from one building model to another makes use of GRS. The building model is represented as a graph, and the detailing decision is represented as a subgraph. Then, based on defined matching and rewriting patterns, the rewriting system produces a detailed graph of the building model.

## 7.3 Methodology

The hypothesis of this paper is divided into two main parts: (1) DPs, including building information and the rationale behind them, can be captured using a graph representation. (2) GRS are capable of automatically transferring DPs between models through automatically generated rewriting rules.

### 7.3.1 Proposed approach for capturing and transferring detailing decisions

The proposed methodology is illustrated in Figure 7.2. First, designers formulate a DP, through a BIM-authoring tool, by selecting building elements, spaces, their relationships, and context information. When formulating a DP, designers specify which information belongs to detailing and which belongs to reasoning when to apply it.

A DP could include information of one or multiple elements. Such information includes: (1) geometric representation, such as shape, material layers, and junction types, (2) semantic information, represented with properties that include fire rating, load bearing, etc. (3) context information, describing relations to the nearby elements and the corresponding storey, building, or site. Examples of this context information could be the bounding room types, adjacent and accessible room types, distance from the entrance, or side of the building. The formulated DP is transformed into a graph representation and stored as a template in a data store. In the end, a DP can be described as:

Detailing Pattern (DP) = Matching Pattern + Rewriting pattern ( 1 )

Where the matching pattern finds the corresponding elements and the rewriting pattern specifies which nodes, edges, and attributes should be added, updated, or deleted. When

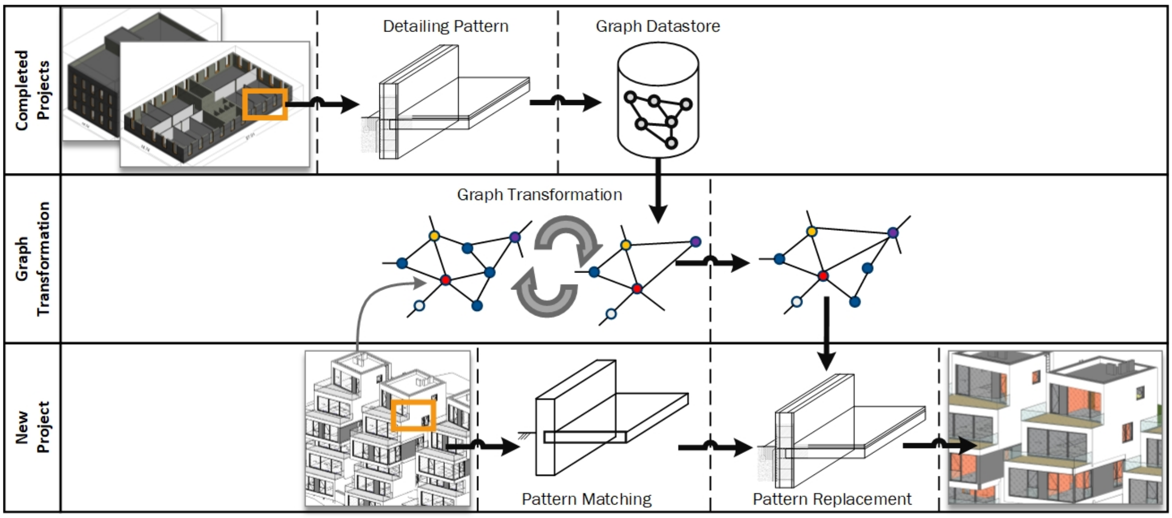


Figure 7.2: Proposed approach for capturing and transferring detailing decisions between models

designers detail a new design option or a new project, they can browse and select one of the stored DPs. As described before, transferring a DP to another model is based on GRS. Hence, the new model has to be transformed into a graph representation to apply the selected DP on it. Applying the DP involves finding all its matches within the model graph. Then its corresponding nodes, edges, and their properties will be transformed with information from the rewriting pattern, producing a detained BIM model graph. Finally, the detail graph is transformed back into a BIM model inside the BIM-authoring tool.

### 7.3.2 BIM-authoring tool and detailing decisions

An essential step for storing and applying a detailing pattern is the transformation of building information from the BIM-authoring tool to a graph, and then back to the BIM-authoring tool after the application of detailing. In this paper, we selected Autodesk Revit as a BIM-authoring tool since its API provides the ability to collect all the necessary information about building elements and their topological dependencies.

Currently, BIM-authoring tools provide various kinds of analysis and advanced detailing information[1] in a parametric way. Accordingly, practitioners are provided with a user-friendly interface for detailing their models and specifying geometric constraints. Such capabilities were leveraged by researchers for multiple purposes, including performing automatic code compliance checking through the definition of calculated parameters (Patlakas et al., 2018). As designers use the functionalities offered by the BIM-authoring tool to develop their models, we have evaluated all the possible actions a designer can perform to detail a building model. This helps in confining the scope of this research, providing a practically applicable approach.

---

[1] https://autode.sk/3rWjXc3 | https://autode.sk/3dEYcs1 | https://autode.sk/2RiNksB

Figure 7.3 shows a categorization of the possible detailing decisions. There are three main categories, geometric, semantics, and joins/connections. Geometrically, a designer can add a new geometric element (like placing a wall) as well as modify the representation via modelling or changing geometric parameters. Modifying the representation via modeling involves manipulating the element's family by refining or adding geometric shapes and parts.

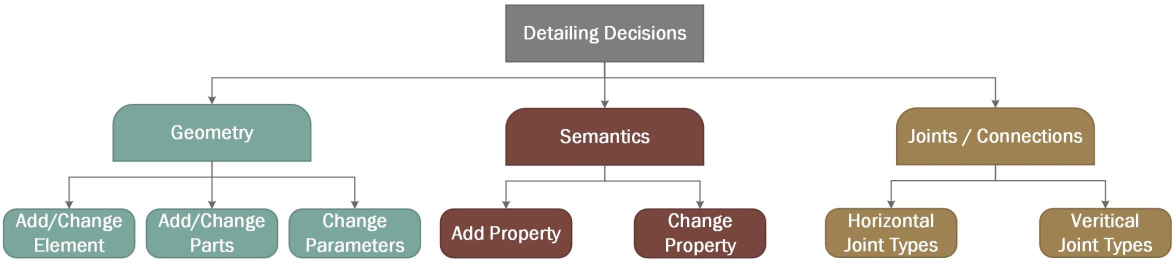


Figure 7.3: An overview of the available detailing decisions in BIM-authoring tools

Additionally, BIM-authoring tools provide functionalities assisting the modification of the geometry through manipulating a set of parameters using their user interface, such as adding a new material layer with a specific thickness to a wall. The category of semantics is straightforward; a designer can add, modify or delete a property. Finally, there are two main approaches for joining building elements, either horizontally (e.g., when joining two walls) or vertically (e.g., when joining a wall and a slab), where each has a set of possible options (enumeration).

### 7.3.3 Categorization of graph representations

Based on our literature review, existing graph representations in the AEC industry can be categorized into four groups:

1. Space connectivity graphs: spaces are represented as nodes, and edges represent either or both of the accessibility and adjacency between the different spaces. Connectivity graphs were used for evaluating similarity between designs, in a sense of a fingerprint (He et al., 2018; Langenhan et al., 2013), evaluating design quality (Donato, 2017), reasoning about disability mobility (Strug & Ślusarczyk, 2017), emergency path planning (Ismail et al., 2018; Rüppel et al., 2010), and security analysis (Porter et al., 2014).
2. Navigation graphs: for the purpose of simulating pedestrian's behavior or navigating robots and drones, a space graph is not sufficient. Therefore, a more fine graph representation is necessary, including additional special nodes representing visibility points (Kneidl et al., 2012) or navigation goals and interaction with the environment (Al Hattab & Hamzeh, 2018; Dubey et al., 2020).

3. IFC model graphs: multiple researchers have investigated transforming the IFC building models into graph representations (Exner et al., 2019; Ismail et al., 2018; Khalili & Chua, 2015). The resultant nodes do not only represent building elements, but also their geometric representations, material layers, and more since the IFC schema is substantially expanding with every new release to support additional use-cases[2]. In a similar sense, ontology approaches were investigated in providing building representations for the purpose of seamlessly exchanging BIM models through web services, such as the Building Topology Ontology (BOT) (Rasmussen et al., 2019).

4. Knowledge representation graphs: multiple researchers have leveraged graphs for formalizing knowledge (Solihin & Eastman, 2016; Vilgertshofer & Borrmann, 2017) and linking heterogeneous data models (A. E. Hor et al., 2018), where a customized graph representation or the combination of multiple graph structures is used. The same applies to parametric models, where a specific logic is embedded within the different graph nodes.

The closes category to our needs for capturing and transferring detailing decisions is the IFC model graphs. However, the IFC schema is a strict representation intended to be implemented by BIM software vendors to provide a neutral medium for exchanging BIM models. Accordingly, IFC is based on a relational model representation, where it includes objectified relationships and properties. Such representation is not flexible enough for capturing the custom detailing patterns and is not optimal for the usage as a graph "as-is"; various simplifications and manipulations are required. Additionally, transferring detailing patterns back to the BIM-authoring tools is an essential requirement for this research. Therefore, a simple graph structure that is capable of representing spaces and building elements, including their detailing, is necessary.

### 7.3.4 Graph representation for capturing detailing patterns

Based on investigated detailing decisions and reviewed graph representations in the AEC industry, the need for a new and simplified graph representation that is capable of capturing DPs was identified. The meta-model of the proposed graph representation is shown in Figure 7.4. A graph comprises at least one node and can include multiple edges. The class ElementNode is the parent node class that holds attributes describing a node's identity as well as its corresponding matching and rewriting patterns. In terms of geometric representation, the geometric parameters (including the geometric parts, their properties, and order), as well as the bounding box of each element, are captured.

---

[2] https://technical.buildingsmart.org/standards/ifc/ifc-schema-specifications/

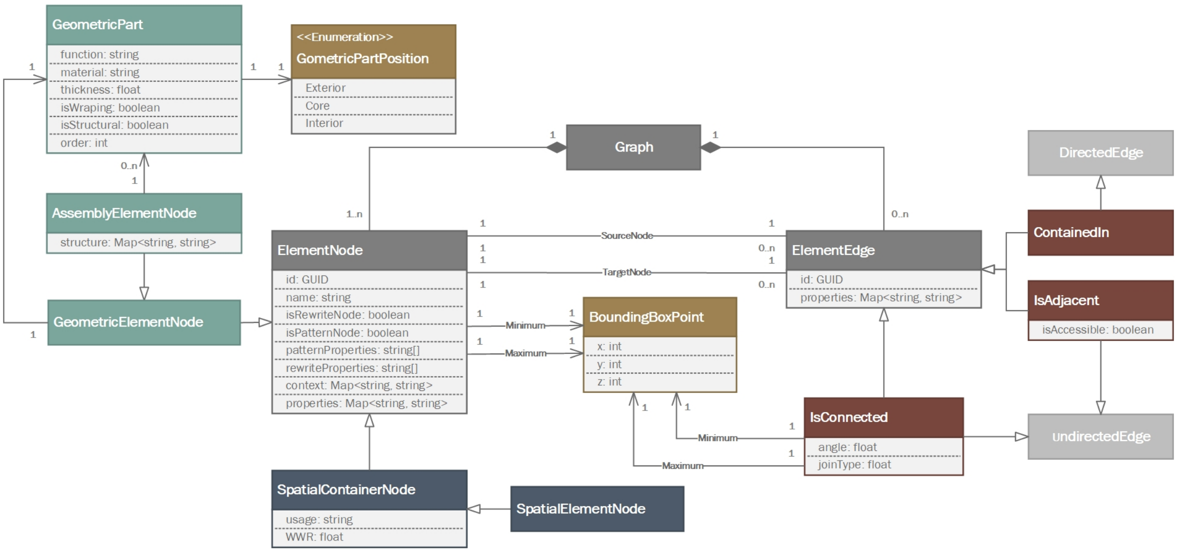


Figure 7.4: Parametric Building Graph (PBG): meta-model (UML diagram)

As inheritance nodes from the ElementNode, the GeometricElementNode and SpatialContainerNode capture an additional set of properties. Here we differentiate between geometric elements that are simple (e.g., a column or a one-layered wall) and assembly (e.g., multi-layered wall or multi-part components). The SpatialContainerNode represents any space that has an implicit representation, like a storey or a building, while the SpatialElementNode represents the actual spaces, (e.g., modelled rooms and zones).

There are three main types of edges for describing the relationships between the different kinds of nodes: (1) ContainedIn (directed edge), describes the relationship between geometric and spatial elements, where the direction specifies the containment's host, for example, a wall is ContainedIn a room, and an opening is ContainedIn a wall, (2) IsAdjacent (undirected edge), links adjacent spaces with each other, identifying their accessibility, and (3) IsConnected (undirected edge), describes the connections and joins between the geometric elements. The connection point between two elements is represented through the angle of their bounding boxes and a detailed connection position between their faces, using horizontal and vertical anchors and paddings.

Figure 7.5 highlights multiple concepts that were discussed so far for describing the captured geometry. The horizontal and vertical joins show two different options for each. Additionally, when describing the connections between the bounding boxes of the elements, the angle, anchor, and padding are measured to describe their relative position as either a raw value or percentage.

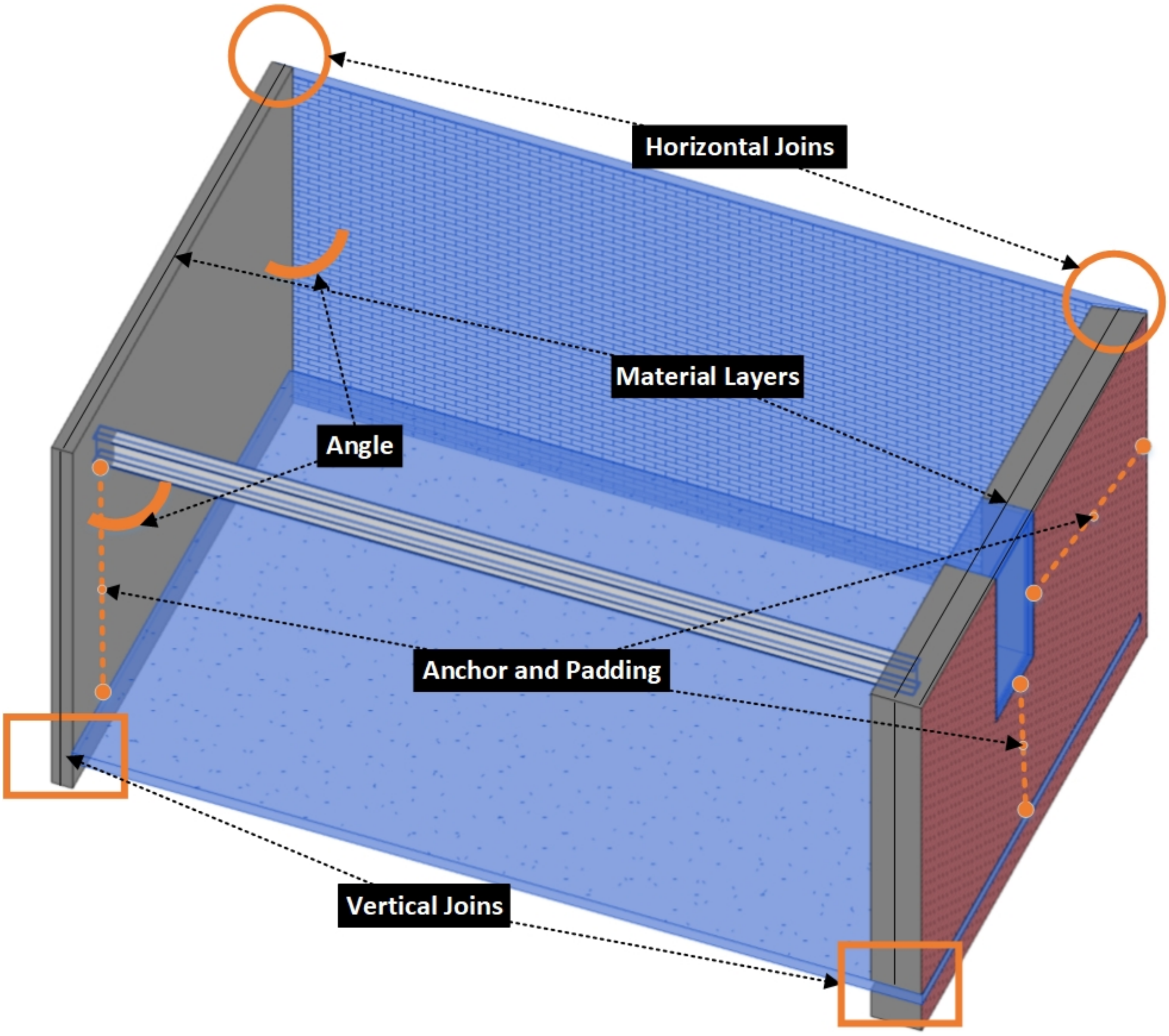


Figure 7.5: Illustration of the captured positions, connections, and joins between building elements

### 7.3.5 Graph rewriting systems for transferring detailing patterns

As described previously, we propose transferring detailing patterns using GRS. Graph transformations are based on declarative rules that specify a set of modifications of graphical structures. The essential process of performing graph transformations matches a pattern graph within a large graph (a.k.a., host graph) and then applying graph modifications. Subgraph matching is known as an NP-complete problem (Geiss et al., 2006). A popular algorithm for overcoming such a problem is Search Plan (Batz et al., 2007). Search Plan is a heuristic optimization algorithm where a sequence of primitive matching operations is performed. In this regard, a cost value is assigned to the different operations. Accordingly, such algorithms perform matching gradually during runtime on the corresponding host graph.

GRS make use of heuristic pattern matching algorithms to perform their graph transformations. The configuration of such transformation is represented through a set of Rewriting Rules. A rule consists of four main parts (see Figure 7.6). Two parts match a pattern according to a set of nodes, edges, as well as logical checks (including if-else conditions) that are performed on their properties. For example, a pattern of a wall separating two rooms, where the wall's material is wood and the area of one of the rooms is larger than or equal to 20sqm. To prevent encountering infinite updates and loops through the matched patterns, a Negative Pattern is defined through a graphlet that acts as a skipping criterion. Typically, a negative pattern includes a graphlet of the matching pattern after rewriting. When a pattern passes the negative pattern check, a Rewriting Pattern is applied on it. A rewriting pattern includes a description of what modifications will be performed, including adding, deleting, and modifying nodes, edges, and their properties.

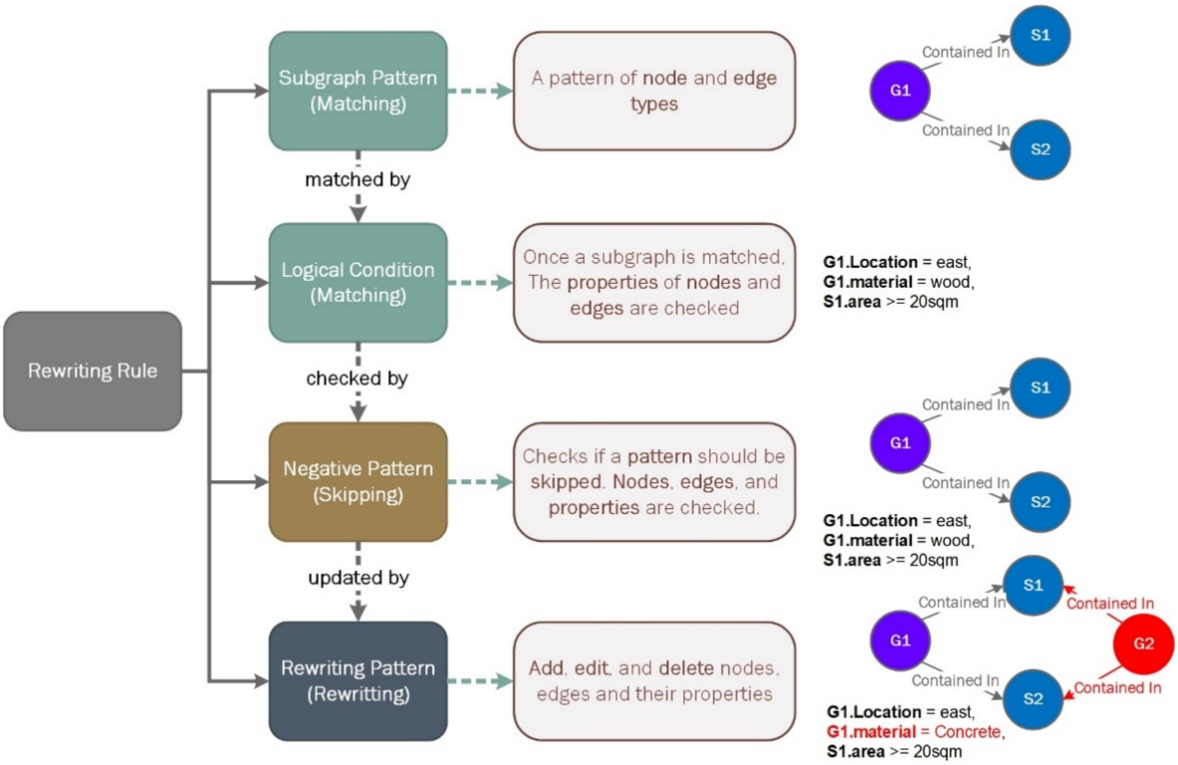


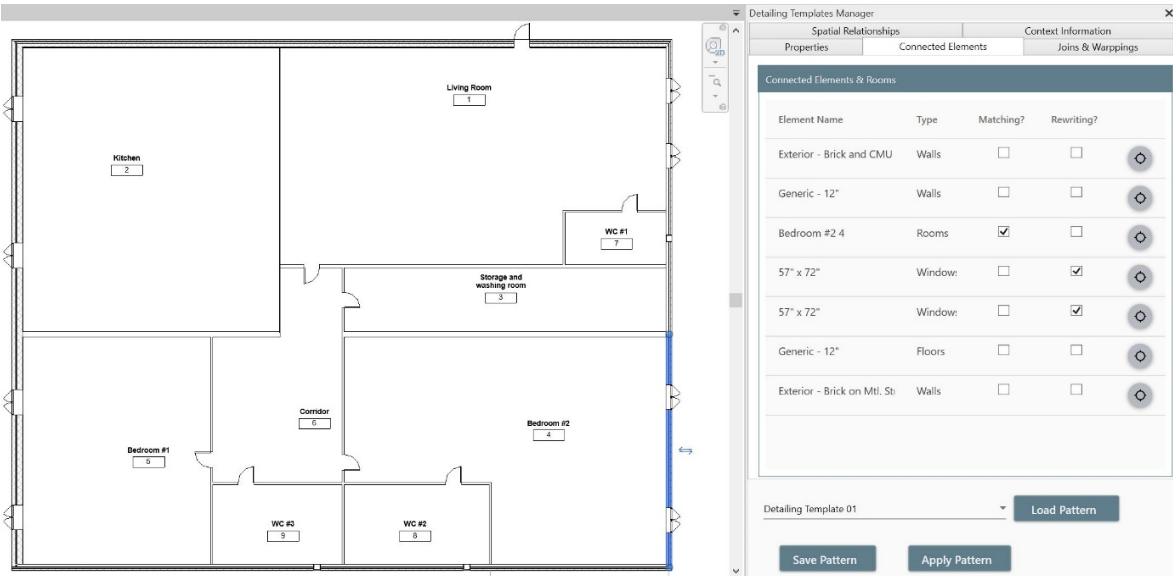
Figure 7.6: Graph rewriting system: structure of a rewriting rule

Our approach proposes an automatic generation of rewriting patterns from the captured detailing pattern, which is specified through the user interface of the BIM-authoring tool. Accordingly, the main advantage of this automation step is to reduce the burden for engineers and architects of having additional knowledge in formally defining rewriting rules as a prerequisite of using the GRS.

## 7.4 Feasibility study

To evaluate the applicability of the proposed approach, the API of Autodesk Revit[3] was used to export the building information into the developed graph representation. Additionally, we have selected GrGen.NET (Jakumeit et al., 2010) as a GRS since it provides an API for interacting with its algorithmic kernel and its libraries can be integrated within a Revit plugin. GrGen uses the Search Plan algorithm to perform subgraph matching. Accordingly, a DP match is achieved through a sequence of search operations for the individual matching nodes and edges, taking into account the structure of the building graph during runtime.

Inside Revit, when a modeler selects an element, like a wall, its corresponding properties, material layers, joins, and connections to adjacent elements as well as rooms are shown in a developed plugin. Using the user interface, it is possible to select which properties and nodes belong to the matching pattern or the rewriting pattern of the DP. Then, the formulated DP can be stored in order to be transferred later to another building model. Figure 7.7 shows a snapshot of the developed plugin inside Revit. Here, the DP is formalized for the selected exterior wall, where it is bounding a room with a bedroom as usage, this relationship is selected as part of the matching pattern, and two windows, contained in the wall, are selected as part of the rewriting pattern. The relative position anchor and padding can be specified through the position button beside each row. The formulation of the matching and rewriting patterns combines more information about the selected elements under the other tabs.



| Element Name | Type | Matching? | Rewriting? |
|---|---|---|---|
| Exterior - Brick and CMU | Walls | ☐ | ☐ |
| Generic - 12" | Walls | ☐ | ☐ |
| Bedroom #2 4 | Rooms | ☑ | ☐ |
| 57" x 72" | Window | ☐ | ☑ |
| 57" x 72" | Window | ☐ | ☑ |
| Generic - 12" | Floors | ☐ | ☐ |
| Exterior - Brick on Mtl. St | Walls | ☐ | ☐ |

Detailing Template 01

Figure 7.7: Prototype: Autodesk Revit plugin for capturing and transferring detailing patterns

In this study, we evaluated automatically transferring the detailing decisions discussed in Section 7.3 on multiple building designs. As a result, we were able to successfully generate a

---

[3] https://www.autodesk.com/products/revit/overview

detailed building graph for all the categorized detailing decisions, except automatically editing a building element family. Automatically detailing an element family requires extending the proposed graph structure for capturing the geometric operations in more detail.

Figure 7.8 presents an example of the graph representation and an automatically generated rewriting rule. The building graph represents a storey with three rooms that are surrounded by simple and assembly walls. The detailing pattern tries to find a pattern of two not accessible spatial elements, where the area of one of the rooms is bigger than 20sqm, then places a door in the wall separating them, according to their relative position, and changes the wall's material to Brick. Such detailing pattern searches for a matching subgraph of nodes and edges (same structure) and then checks whether the area is larger than 20sqm. Before transferring the detailing pattern, the kitchen and living room were separated by a Concrete wall, and there was no accessibility between them. Once the pattern was matched and rewritten, a door was added to the separating wall, and the wall's material was changed to Brick.

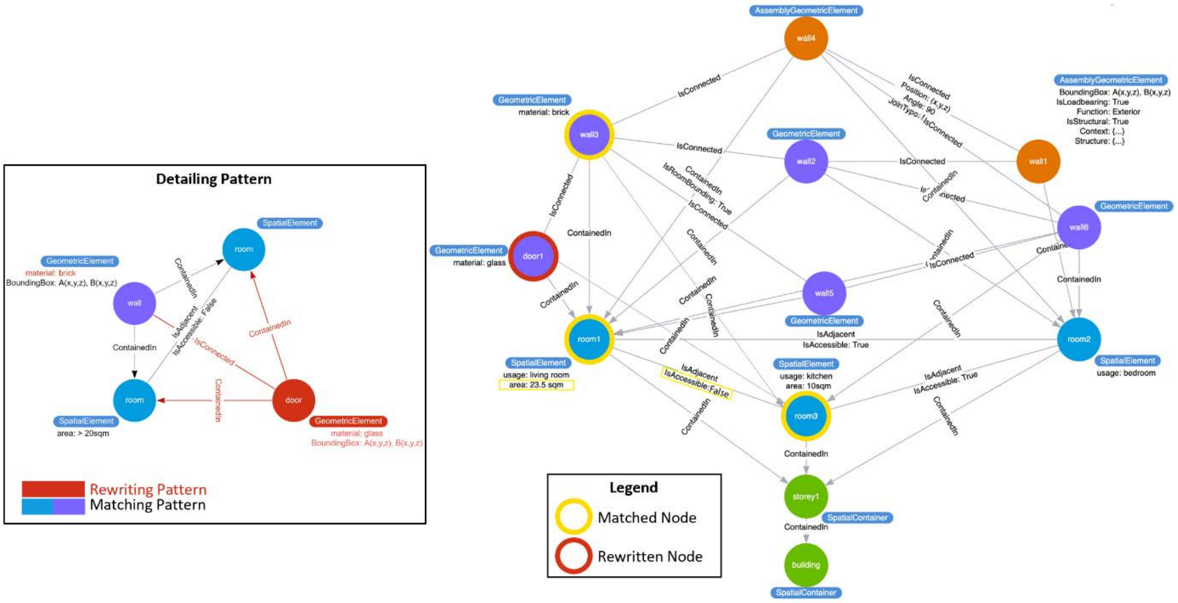


Figure 7.8: An example of a graph representation and an automatically generated rewriting rule

## 7.5 Conclusions & future research

Design and detailing decisions highly influence the resultant building's performance and compliance with regulations. Architects and engineers reuse their successful experiences to transfer those decisions to new projects and design variants. This paper introduced a simplified and parametric building graph that is capable of capturing detailing patterns, including building information and the rationale behind them. Additionally, a framework that is based on graph transformation systems was proposed for automatically transferring detailing decisions from one design to another. Through evaluation of the implemented prototype, the proposed approach was able to handle multiple detailing decisions, including adding elements, modifying elements' geometry through parameters, as well as manipulating their semantics.

As future work, modifying the element's type geometry will be investigated in detail. In this regard, the fundamental geometric operations will be captured and reproduced through graph theory techniques. Additionally, an extensive evaluation of the developed framework on different sizes and types of building projects will be investigated.

## Acknowledgements

We gratefully acknowledge the support of the German Research Foundation (DFG) for funding the project under grant FOR 2363.

## Chapter 8

# Conclusions & future research

The existing methods and tools for managing design information and requirements are inflexible for assisting the collaboration needs during the development and delivery of BIM artifacts starting from the early design phases. On the one hand, the involved vagueness is not incorporated into design and performance analysis; BIM models appear precise and certain, even during design exploration. Additionally, checking the consistency of models is still a manual and laborious task, open to multiple interpretations according to the subjective assessment of domain experts. On the other hand, the resultant detailing patterns (design artifacts), including their rational, are not captured to make the documentation and reuse of knowledge possible in other projects.

The previous Chapters introduced the concepts and the techniques investigated for addressing the overall thesis' objective: "*consistently managing and evaluating building models across the design stages*". This chapter discusses the main findings of the conducted investigations and the main characteristics of the introduced methods, emphasizing the tackled research questions and objectives, highlighting the limitations of the methods as well as potential future research.

## 8.1 Research question I: Evaluation of standards for specifying design maturity and detailing requirements

Chapter 2 has investigated the interpretation and application of the LOD concept through a systematic literature review. The investigation has focused on answering the following research question:

“Which standards for specifying design maturity and detailing requirements in building models do exist and what are their distinctive features? Would they be able to represent the information vagueness?”

As a result, it identified a trend of increasing the adoption of the LOD concept through the years to perform multiple use cases, including the specification of design requirements, detailing, and reliability of BIM devlierables. At the same time, this literature review revealed numerous misconceptions and misapplications. In more than 50% of the relevant publications, authors did not explicitly cite which LOD guideline they are referring to. This emphasizes either their assumption that the LOD concept is well known and understandable by the community or the complexity of choosing among the numerous guidelines available due to their deviations.

In a collaborative environment like the AEC industry, having a common understanding of requirements and the content of deliverables is crucial. The identified issues with using the LOD concept reflect the demanding needs of practitioners to perform their tasks. Hence, this study stressed the importance of internationally standardizing an LOD framework that abstracts from and handles the deviations of the existing guidelines. Examples of deviations include whether there is an LOD of a building model, whether it is only confined to the element level, and whether it can represent the needs of a particular use case.

This study has covered 58 LOD guidelines and 299 peer-reviewed publications in-depth. The results highlighted an increasing trend in using the LOD concept and identified the most widespread LOD guidelines and naming conventions. Finally, 16 common use cases for applying the LOD were observed. The findings reflect the current practice in published research in academic journals. The hypothesis of "practitioners use common guidelines as a communication language for design requirements and deliverables" holds true and was successfully proven.

As a future research, a further investigation of the understanding and application of the LOD within the industry could emphasize more the demanded use cases. This could be accomplished by conducting multiple interviews with different companies, focusing on incorporating the LOD standard in practical projects.

## 8.2 Research question II: Formal representation of building information's vagueness

The following research question was derived to evaluate means for incorporating information vagueness with requirements and design information to explicitly communicate any unknown information during exchanging models and performing the different simulations and analyses:

"Information vagueness has numerous types and representations. How can the vagueness in the building design information be represented?"

Chapter 3 presented the developed methodology, where a multi-LOD meta-model approach was developed to specify information requirements of the individual elements for different project milestones, including the involved vagueness. In this regard, multiple kinds of vagueness were handled, including numerical and alphanumerical information, providing a ground framework for managing and checking the design's refinement.

At its core, the overall concept behind managing the information vagueness is systematically narrowing down the range of potential options with the progression of design. This way, the impact of the decisions is gradually evaluated and maintained throughout the design phases. The proposed approach was evaluated for managing the information vagueness and its impact on the calculation of the LCA's EGHG, starting from the early phases. The results positively assure the hypothesis: "A formal specification of requirements, including the associated vagueness, can assist collaboration and decision-making during design phases". Where the potential range of EGHG emissions is reduced with the progression of design. This is providing domain experts the ability to control the tendency of design performance directly from the design information towards the project goals.

As a future research, the intuitiveness and potential adoption of the developed methodology by practitioners can be investigated further, as estimating vagueness could involve additional effort and debate by the involved domain experts. Hence, assessing the proposed approach on multiple real-world projects that vary in size and complexity would identify means for enhancing user experience of the developed system. Additionally, collecting vagueness estimations from previous projects could provide sufficient knowledge to assist domain experts in specifying the involved vagueness in new projects.

## 8.3 Research question III: Techniques for visualizing vagueness

Numerous sources highlighted the importance of the 3D interactive visualization in the AEC industry for coordination with the project participants and evaluating the design integrity. This research tackled the lack of vagueness visualization of building information through the investigation and development of specialized visualization techniques by answering the following research question: "Which visualization techniques are effective for depicting information vagueness for the use cases of the AEC industry?"

As described in Chapter 4, the same BIM model is used for numerous use cases, where each demands a different granularity of the presented information (from depicting the vagueness associated with overall building model to the positional and material details of the individual elements). Hence, this research presented a vagueenss visualization framework where specialized visualizations are evaluated for each application type.

The evaluation for the developed framework was conducted through an extensive survey, where the results positively indicated the participants' ability to use the developed approaches to understand the amount and impact of the vagueness associated with the building information. Accordingly, some of the explored approaches outperformed the others for a specific application type. Hence, proving that the following hypothesis holds true: "Visualization techniques are capable of conveying the vagueness associated with building models at different rates of effectiveness assisting decisions in the different use cases".

It was helpful to categorize the visualization approaches according to their applicability. In this context, further refinement and evaluations could be necessary to support infrastructure use cases, accounting for their specialization, such as handling vertical and horizontal alignments. Moreover, with the rise of the importance of digital twinning, visualization techniques, while incorporating vagueness information, are gaining more attention and additional use cases to support.

## 8.4 Research question IV: Preservation of building models' consistency

The refinement of building models through the design phases is multi-dimensional, involves decisions from various disciplines, and affects the functionality and performance of the organization within each storey as well as of the individual elements. Chapter 3 investigated this in detail and proposed a framework for controlling the permissible refinement scope. The resultant framework answers the following research question:

"How could the refinement consistency of building models from one design phase to another be formally described and preserved?"

The proposed approach assists in maintaining a consecutive development of the model's information while accounting for previously taken decisions. The approach was built upon making use of the types of vagueness presented previously, where the specified vagueness is evaluated. Additionally, it assessed the topological refinement of consecutive milestones by evaluating them for equivalency. This was achieved by expressing the elements' relationships as a labeled graph and then evaluating both graphs for equivalency, meaning that the building model of a subsequent milestone has maintained the topological decisions taken in the previous milestone (described with examples in Chapter 3). As a result of assessing the developed approach on multiple real-world use cases, it successfully proved Hypothesis 3, where refinement inconsistencies are detected.

The presented approach is capable of flagging warnings to experts when any inconsistencies occur. However, proposing design changes or solving the detected inconsistencies is a still manual task. Hence, formalizing the design space and proposing suitable solutions is still open for investigation as a future research.

## 8.5 Research question V: Representation and classification of Level of Geometry

Chapter 3 tackled managing the vagueness associated with the geometric and semantic attributes. However, as highlighted in Chapter 2, it is common in the AEC industry to describe the required geometric maturity or detailing according to the common LOD specifications. Chapter 3 checked the required attributes, including their vagueness. However, in the LOD specifications, it is also specified which geometric details must be modeled. When those geometric parts are not parametric (i.e., are not represented as attributes), their verification requires analyzing the geometry manually (by inspecting the 3D representation), which is a time consuming and an error-prone task. Hence, Chapter 5 presented the developed methodology for classifying building elements only from their geometric representation by answering the following research question: "Which geometric features are capable of representing the detailing complexity of building elements? Which techniques can classify BIM objects based on their geometric complexity?"

The proposed approach identified which geometric features are capable of representing the geometric complexity on each LOG, then multiple ML techniques are evaluated for their efficiency and robustness in classifying any given building element. The results have shown high accuracy (85% and robustness 83%). This proves that the geometric complexity of building elements correlates with the refined geometric features at each LOG (i.e., Hypothesis 4 holds true).

To improve the achieved accuracy, a future research could investigate the impact of both, increasing the included features by semantic properties, or increasing the dataset size (size of training samples) and evaluating the latest deep learning architectures.

## 8.6 Research question VI: Capturing the rationale behind design decisions

The produced building models are a result of various processes combined. Architects and engineers evaluate client requirements and regulations and then employ their creativity and experience to deliver a valid design solution fulfilling diverse needs. Accordingly, behind the building information of those produced artifacts, there is a lot of reasoning leading to numerous design decisions. Chapter 6 developed approaches for answering the following research question:

"How could the rationale behind design decisions be captured?"

Chapter 6 proposed documenting design decisions and their rationale through three main approaches. First, provide means for enriching building elements and their relationships by multiple constraints (semantical and topological). Then, facilitate the linkage of building information to specific rules and sentences in building codes and owner requirements. Here, NLP techniques were leveraged for querying the natural text. The third approach focused on capturing a subset of building information as a detailing pattern, which can be leveraged for documentation and design transfer to different projects (described in detail in the following research question). The developed approach has been successfully evaluated for documenting design decisions for multiple case studies. Accordingly, the following hypothesis holds true: “Design decisions can be documented through explicit design constraints and links to specific regulations and owner requirements”.

Next, as the interaction between practitioners and the developed framework is crucial for effectively integrating it within the established workflows, evaluating the framework’s usability and intuitiveness could be tackled as a future research.

## 8.7 Research question VII: Techniques for capturing and transferring detailing patterns

One of the objectives of this research is to capture detailing patterns (of one or multiple elements), including building information and the reasoning behind them. This objective was tackled in Chapter 7, where it answers the following research question:

“How could detailing patterns be formally represented? Which techniques are capable of capturing detailing patterns and transferring them to new projects?”

As investigated in Chapter 7, the reasoning behind detailing decisions could be based on constraints or fulfilled regulations. Additionally, it could be described by the elements’ position and connections to other elements, or their relative location to the building (e.g., distance from the entrance, contained spacial structure, and nearby elements), or to the surrounding environment (like orientation in relevance to sun exposure, nearby facilities, roads, and nature).

The followed approach has identified the need for a simple and flexible building graph to sufficiently manage the dynamic nature of detailing patterns, including their reasoning. Accordingly, a generic graph representation was proposed and used as a template for a graph rewriting system to deliver an adaptive framework for transferring detailing patterns when coupled with BIM-authoring tools.

Overall, graph structures have proven their flexibility for capturing detailing patterns. Additionally, it was possible to develop a parametric GRS that is capable of matching and rewriting the detailing pattern graph to the building graph. The developed framework was able to handle multiple detailing patterns, including adding elements, modifying elements' geometry through parameters, as well as manipulating their semantics and topological relationships. Finally, the applicability of the developed approach was demonstrated by coupling it with a BIM-authoring tool as a plugin to bring the detailed building graph back. The achieved results were sufficient to successfully prove the hypothesis: "Graph structures are flexible enough to represent detailing patterns and their rationale. Additionally, graph rewriting systems can can transfer detailing patterns from one design to another in a parametric way".

As a future research, handling the cases where the elements' geometry is not adjustable via parameters could be investigated. In this regard, extracting a graph representing the Boolean operations that are encoding the final geometry is a promising approach that can assist their modification and transfer.

## 8.8 Final remarks

This thesis presented the approaches investigated to realize the methodology presented in Figure 1.5 for formally managing and reusing design information. According to the performed evaluations, the proposed method has proven to be feasible. From the various investigations conducted during this research, we witnessed the advantages gained from systematically specifying, communicating, and mitigating vague information through the design phases. Defining and validating design maturity, as well as dealing with vague information, are key aspects of design development that have received insufficient attention and computational support so far. Implementing the developed concepts in practice will further advance the adoption of digital methods in the AEC industry and help it achieve a higher level of efficiency and performance. As described in the previous section, all of the derived research questions were successfully answered. Nevertheless, as highlighted in the previous section, there is room for further research, extending and evaluating the developed approaches.

# References

A2Kstore. (2021). Levels of development / levels of detail / lod explained. Retrieved July 31, 2021, from https://www.a2kstore.com/levels-of-development-levels-of-detail-lod-explained

Abou-Ibrahim, H., & Hamzeh, F. (2016). Enabling lean design management: An lod based framework. *Lean construction journal*, *2016*, 12–24.

Abrishami, S. (2016). Conceptual design automation: Consideration of building materials impact at early stages of aec design. *Advances in Civil, Architectural, Structural and Constructional Engineering: Proceedings of the International Conference on Civil, Architectural, Structural and Constructional Engineering, Dong-A University, Busan, South Korea, August 21-23, 2015*, 263.

Abualdenien, J., & Borrmann, A. (2020a). Formal analysis and validation of levels of geometry (log) in building information models. *Proc. of the 27th International Workshop on Intelligent Computing in Engineering.*

Abualdenien, J., & Borrmann, A. (2021). Pbg: A parametric building graph capturing and transferring detailing patterns of building models. *Proc. of the CIB W78 Conference 2021.*

Abualdenien, J., & Borrmann, A. (2022). Ensemble-learning approach for the classification of levels of geometry (log) of building elements. *Advanced Engineering Informatics*, *51*(1474-0346), 10149. https://doi.org/https://doi.org/10.1016/j.aei.2021.101497

Abualdenien, J., Schneider-Marin, P., Zahedi, A., Harter, H., Exner, H., Steiner, D., Mahan Singh, M., Borrmann, A., Lang, W., Petzold, F., König, M., Geyer, P., & Schnellenbach-Held, M. (2020). Consistent management and evaluation of building models in the early design stages. *Journal of Information Technology in Construction*, *25*, 212–232. https://doi.org/https://doi.org/10.36680/j.itcon.2020.013

Abualdenien, J., & Borrmann, A. (2019). A meta-model approach for formal specification and consistent management of multi-lod building models. *Advanced Engineering Informatics*, *40*, 135–153. https://doi.org/https://doi.org/10.1016/j.aei.2019.04.003

Abualdenien, J., & Borrmann, A. (2020b). Vagueness visualization in building models across different design stages. *Advanced Engineering Informatics*, *45*, 101107. https://doi.org/10.1016/j.aei.2020.101107

Ackoff, R. L. (1989). From data to wisdom. *Journal of applied systems analysis*, *16*(1), 3–9.

ADEB-VB. (2015). Building information modelling – belgian guide for the construction industry. Retrieved July 19, 2020, from https://docplayer.net/9550222-The-guide-to-building-information-modelling.html

AEC CAN. (2012). Bim protocol v1.0. Retrieved November 19, 2021, from https://docplayer.net/45856652-Aec-can-bim-protocol.html

AEC UK. (2009). Aec (uk) bim standard - a practical & pragmatic bim standard for the architectural, engineering and construction industry in the uk. Retrieved November 19, 2021, from https://aecuk.files.wordpress.com/2009/12/aecukbimstandard-v1-0.pdf

AEC UK. (2012). Bim standard v2.0. Retrieved November 19, 2021, from https://aecuk.wordpress.com/documents/

AIA. (2008). E202-2008 building information modeling protocol exhibit. Retrieved May 7, 2021, from http://www.durhamnc.gov/agendas%5C_new/2015/cm20150302/10290%5C_%5C%5CCONTRACT%5C_E202%5C_CONTRACT%5C_ATTACHMENT%5C_%5C_365760%5C_627078.PDF

AIA. (2013a). Document g202—2013, project bim protocol form. Retrieved November 19, 2021, from https://fdocuments.net/document/g202-2013-project-bim-protocol-form-crec-capitol-document-g202-tm-2013.html

AIA. (2013b). E203™–2013, building information modeling and digital data exhibit. Retrieved May 7, 2021, from https://help.aiacontracts.org/summary-e203-2013/

AIA. (2013c). Guide, instructions and commentary to the 2013 aia digital practice documents. Retrieved May 7, 2021, from https://www.aiacontracts.org/resources/69541-guide-instructions-and-commentary-to-the-2013-aia-digital-practice-documents

AIA. (2014). Bim in practice: Bim what is it? Retrieved August 12, 2021, from https://wp.architecture.com.au/bim/wp-content/uploads/sites/40/2014/07/BIM-What-is-it.pdf

Al Hattab, M., & Hamzeh, F. (2018). Simulating the dynamics of social agents and information flows in bim-based design. *Automation in Construction*, *92*, 1–22.

Alliez, P., De Verdire, E. C., Devillers, O., & Isenburg, M. (2003). Isotropic surface remeshing. *2003 Shape Modeling International.*, 49–58.

Andre, A. D., & Cutler, H. A. (1998). Displaying uncertainty in advanced navigation systems. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *42*, 31–35.

Autodesk. (2018). *Bim360: Collaborative construction management software*. Retrieved May 24, 2018, from https://bim360.autodesk.com/

Autodesk. (2019). Assemble products: Level of development (lod). Retrieved July 31, 2021, from https://knowledge.autodesk.com/support/assemble-products/getting-started/caas/simplecontent/content/level-development-lod.html

Ayzenshtadt, V., Espinoza-Stapelfeld, C., Langenhan, C., & Althoff, K.-D. (2018). Multi-agent-based generation of explanations for retrieval results within a case-based support framework for architectural design. *ICAART (1)*, 103–114.

Bahn, D. (2020). Vorgaben zur anwendung der bim-methodik. Retrieved August 12, 2021, from http://www1.deutschebahn.com/sus-infoplattform/start/Vorgaben%20zu%20Anwendung%20der%20BIMMethodik.html

Bailey-Kellogg, C., & Zhao, F. (2003). Qualitative spatial reasoning extracting and reasoning with spatial aggregates. *AI Magazine*, *24*(4), 47.

Bambardekar, S., & Poerschke, U. (2009). The architect as performer of energy simulation in the early design stage. *11th International IBPSA Conference-Building Simulation 2009, BS 2009*.

Batz, G. V., Kroll, M., & Geiss, R. (2007). A first experimental evaluation of search plan driven graph pattern matching. *International Symposium on Applications of Graph Transformations with Industrial Relevance*, 471–486.

Bazjanac, V. (2008). IFC bim-based methodology for semi-automated building energy performance simulation.

Beck, F. (2019). *Categorization and visualization of model-based informational distance during the bim-based design process* (Master's thesis). Technische Universität München.

Beetz, J., Borrmann, A., & Weise, M. (2018). Process-based definition of model content. In A. Borrmann, M. König, C. Koch, & J. Beetz (Eds.), *Building information modeling: Technology foundations and industry practice* (pp. 127–138). Springer International Publishing. https://doi.org/10.1007/978-3-319-92862-3_6

Beetz, J., van Berlo, L., de Laat, R., & van den Helm, P. (2010). BIMserver.org–an open source IFC model server. *Proceedings of the CIP W78 conference*.

Bettig, B., & Shah, J. (2001). Derivation of a standard set of geometric constraints for parametric modeling and data exchange. *Computer-Aided Design*, *33*(1), 17–33.

Bhooshan, S. (2017). Parametric design thinking: A case-study of practice-embedded architectural research. *Design Studies*, *52*, 115–143.

Biljecki, F., Ledoux, H., & Stoter, J. (2016). An improved lod specification for 3d building models. *Computers, Environment and Urban Systems*, *59*, 25–37.

BIM4INFRA. (2019). Handreichung bimfachmodelle und ausarbeitungsgrad. Retrieved August 12, 2021, from https://bim4infra.de/wp-content/uploads/2019/07/BIM4INFRA2020%5C_AP4%5C_Teil7.pdf

BIMForum. (2019). Level of development specification guide. Retrieved March 19, 2020, from http://bimforum.org/lod/

BIMForum. (2020a). 2020 bim project execution plan guide. Retrieved August 9, 2021, from https://bimforum.org/Bxp

BIMForum. (2020b). 2020 level of acceptance (loa) specification for bim reality capture and simulation. Retrieved August 9, 2021, from https://bimforum.org/Reality

Bips. (2006). Lag- og objektstruktur 2006. Retrieved November 19, 2021, from http://prodstoragehoeringspo.blob.core.windows.net/7d5a1aa5-e172-464e-90bd-e1c58171ef9f/lagobjektstruktur%5C_endelig.pdf

Bips. (2007). 3d working method 2006. Retrieved November 19, 2021, from https://changeagents.blogs.com/Linked%5C_Documents/BIPS%5C%203D%5C%20Working%5C%20Method.pdf

Bjorkhaug, L., & Bell, H. (2007). Ifd: Ifd in a nutshell. *SINTEF Building and Infrastructure*.

BMVBS. (2013). Bim-leitfaden für deutschland - information und ratgeber (bim- leitfaden für deutschland). Retrieved July 30, 2021, from https://www.bmvi.de/SharedDocs/DE/Anlage/DG/Digitales/bim-leitfaden-deu.pdf?%5C_%5C_blob=publicationFile

BMVI. (2015). Bundesministerium für verkehr und digitale infrastruktur - stufenplan digitales planen und bauen. Retrieved August 12, 2021, from https://www.bmvi.de/SharedDocs/DE/Publikationen/DG/stufenplan-digitales-bauen.html

Bolpagni, M. (2016). The many faces of 'lod'. Retrieved May 29, 2022, from https://www.bimthinkspace.com/2016/07/the-many-facesof-lod.html

Bolpagni, M., & Ciribini, A. L. C. (2016). The information modeling and the progression of data-driven projects. *CIB world building congress*, 296–307.

Borrmann, A., & Berkhahn, V. (2018). Principles of geometric modeling. In A. Borrmann, M. König, C. Koch, & J. Beetz (Eds.), *Building information modeling: Technology foundations and industry practice* (pp. 27–41). Springer International Publishing. https://doi.org/10.1007/978-3-319-92862-3_2

Borrmann, A., Flurl, M., Jubierre, J. R., Mundani, R.-P., & Rank, E. (2014). Synchronous collaborative tunnel design based on consistency-preserving multi-scale models. *Advanced Engineering Informatics*, *28*(4), 499–517.

Borrmann, A., Kolbe, T. H., Donaubauer, A., Steuer, H., Jubierre, J. R., & Flurl, M. (2015). Multi-scale geometric-semantic modeling of shield tunnels for gis and bim applications. *Computer-Aided Civil and Infrastructure Engineering*, *30*(4), 263–281.

Borrmann, A., König, M., Koch, C., & Beetz, J. (2018a). *Building information modeling technology foundations and industry practice: Technology foundations and industry practice*. Springer.

Borrmann, A., König, M., Koch, C., & Beetz, J. (2018b). *Building information modeling: Why? what? how?* Springer.

Botsch, M., Kobbelt, L., Pauly, M., Alliez, P., & Lévy, B. (2010). *Polygon mesh processing*. CRC press.

Bouchlaghem, D., Shang, H., Whyte, J., & Ganah, A. (2005). Visualisation in architecture, engineering and construction (aec). *Automation in construction*, *14*(3), 287–295.

Boukhelifa, N., & Duke, D. J. (2009). Uncertainty visualization: Why might it fail? *CHI'09 Extended Abstracts on Human Factors in Computing Systems*, 4051–4056.

Bracewell, R., Wallace, K., Moss, M., & Knott, D. (2009). Capturing design rationale. *Computer-Aided Design*, *41*(3), 173–186.

Braun, A., Tuttas, S., Borrmann, A., & Stilla, U. (2020). Improving progress monitoring by fusing point clouds, semantic data and computer vision. *Automation in Construction*, *116*, 103210. https://doi.org/10.1016/j.autcon.2020.103210

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5–32.

Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*(2), 123–140. https://doi.org/10.1007/BF00058655

Brown, N. C., & Mueller, C. T. (2019). Design variable analysis and generation for performance-based parametric modeling in architecture. *International Journal of Architectural Computing*, *17*(1), 36–52.

Brown, R. (2004). Animated visual vibrations as an uncertainty visualisation technique. *Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, 84–89.

BSI. (2012a). Bs 8541-3-2012: Library objects for architecture, engineering and construction - shape and measurement. code of practice. Retrieved November 19, 2021, from https://shop.bsigroup.com/products/library-objects-for-architecture-engineering-and-construction-shape-and-measurement-code-of-practice

BSI. (2012b). Bs 8541-4-2012: Library objects for architecture, engineering and construction - attributes for specification and assessment. code of practice. Retrieved November 19, 2021, from https://shop.bsigroup.com/products/library-objects-for-architecture-engineering-and-construction-attributes-for-specification-and-assessment-code-of-practice

BSI. (2017). *Pas 1192-2:2013 specification for information management for the capital/delivery phase of construction projects using building information modelling*. Retrieved January 11, 2018, from http://shop.bsigroup.com/Navigate-by/PAS/PAS-1192-22013/

Bueno, C., & Fabricio, M. M. (2018). Comparative analysis between a complete lca study and results from a bim-lca plug-in. *Automation in Construction*, *90*, 188–200.

Building, & Authority, C. (2012). Singapore bim guide v 1.0. Retrieved November 19, 2021, from https://www.corenet.gov.sg/media/586135/Singapore%5C_BIM%5C_Guide%5C%5C%5C_Version%5C_1.pdf

Building, & Authority, C. (2013). Singapore bim guide v 2.0. Retrieved May 7, 2021, from https://www.corenet.gov.sg/media/586132/Singapore-BIM-Guide%5C_V2.pdf

buildingSMART. (2016). Data dictionary. Retrieved January 5, 2018, from https://www.buildingsmart.org/standards/standards-tools-services/data-dictionary/

buildingSMART International. (2021). Machine readable information delivery specification. Retrieved November 19, 2021, from https://technical.buildingsmart.org/projects/information-delivery-specification-ids/

buildingSMART Spanish Chapter. (2014). Guía de usuarios bim. Retrieved July 19, 2021, from https://www.scribd.com/document/321450171/Guia-de-Usuarios-BIM-BuildingSmart-Spanish-Chapter

Calo, T. J. (2008). Talent management in the era of the aging workforce: The critical role of knowledge transfer. *Public Personnel Management*, *37*(4), 403–416.

Card, M. (1999). *Readings in information visualization: Using vision to think*. Morgan Kaufmann.

Castrup, H. (2001). Distributions for uncertainty analysis. *Proc. International Dimensional Workshop, Knoxville, TN*.

Cavieres, A., Gentry, R., & Al-Haddad, T. (2011). Knowledge-based parametric tools for concrete masonry walls: Conceptual design and preliminary structural analysis. *Automation in Construction*, *20*(6), 716–728.

Cemesova, A., Hopfe, C. J., & Mcleod, R. S. (2015). Passivbim: Enhancing interoperability between bim and low energy design software. *Automation in Construction*, *57*, 17–32.

Chakrabarti, A., Shea, K., Stone, R., Cagan, J., Campbell, M., Hernandez, N. V., & Wood, K. L. (2011). Computer-based design synthesis research: An overview. *Journal of Computing and Information Science in Engineering*, *11*(2).

Chapman, I. (2013). An introduction to uniclass2. Retrieved March 22, 2018, from https://www.thenbs.com/knowledge/an-introduction-to-uniclass-2

Châteauvieux-Hellwig, C., Abualdenien, J., & Borrmann, A. (2022). Analysis of early-design timber models for sound insulation. *Advanced Engineering Informatics*, *53*, 101675. https://doi.org/10.1016/j.aei.2022.101675

Chen, P.-H. (2020). Essential elements of natural language processing: What the radiologist should know. *Academic radiology*, *27*(1), 6–12.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Cheung, F. K., Rihan, J., Tah, J., Duce, D., & Kurul, E. (2012). Early stage multi-level cost estimation for schematic bim models. *Automation in Construction*, *27*, 67–77.

Chipman, T., Liebich, T., & Weise, M. (2012). Mvdxml: Specification of a standardized format to define and exchange model view definitions with exchange requirements and validation rules. *V*, *1*, 34.

Chiu, M.-L. (2002). An organizational view of design communication in design collaboration. *Design studies*, *23*(2), 187–210.

Clementini, E. (2010). Ontological impedance in 3d semantic data modeling. *5th 3d Geoinfo Conference (Ed. by TH Kolbe, G. König & C. Nagel), ISPRS Conference*, *38*, 4.

Cliburn, D. C., Feddema, J. J., Miller, J. R., & Slocum, T. A. (2002). Design and evaluation of a decision support system in a water balance application. *Computers & Graphics*, *26*(6), 931–949.

Colic, N., & Rinaldi, F. (2019). Improving spacy dependency annotation and pos tagging web service using independent ner services. *Genomics & informatics*, *17*(2).

Concrete house - Carl-Viggo Hølmebakk AS. (Stange, Norway, completion in 2015). Retrieved February 22, 2021, from https://www.holmebakk.no/project/concrete_house

Construction Industry Council. (2018). Building information model (bim) protocol v2. Retrieved July 30, 2021, from https://cic.org.uk/admin/resources/bim-protocol-2nd-edition-2.pdf

Construction Industry Council. (2020). Cic bim standards for architecture and structural engineering. Retrieved July 30, 2021, from https://www.bim.cic.hk/en/resources/publications%5C_detail/86

Construction Industry Council, UK. (2013a). Best practice guide for professional indemnity insurance when using bims v1. Retrieved July 30, 2021, from https://issuu.com/constructionitalliance/docs/cic%5C_best%5C_practice%5C_guide%5C%5C%5C_for%5C_profess

Construction Industry Council, UK. (2013b). Building information model (bim) protocol v1. Retrieved July 30, 2021, from https://www.em.admin.cam.ac.uk/files/4%5C_uoc%5C_cicprotocol%5C_v1.1.1.pdf

Cross, N. (1982). Designerly ways of knowing. *Design studies*, *3*(4), 221–227.

Daum, S., Borrmann, A., & Kolbe, T. H. (2017). A spatio-semantic query language for the integrated analysis of city models and building information models. *Advances in 3d geoinformation* (pp. 79–93). Springer.

Davis, T. J., & Keller, C. P. (1997). Modelling and visualizing multiple spatial uncertainties. *Computers & Geosciences*, *23*(4), 397–408.

De Wilde, P. (2014). The gap between predicted and measured energy performance of buildings: A framework for investigation. *Automation in Construction*, *41*, 40–49.

DEGES. (2020). Bundesministerium für verkehr und digitale infrastruktur - stufenplan digitales planen und bauen. Retrieved August 12, 2021, from https://www.deges.de/wp-content/uploads/2020/02/BIM-LOD-Konzept%5C_V16.pdf

Deitrick, S. A. (2007). Uncertainty visualization and decision making: Does visualizing uncertain information change decisions. *Proceedings of the XXIII International Cartographic Conference*, 4–10.

Demian, P., Ruikar, K., Sahu, T., & Morris, A. (2016). Three-dimensional information retrieval (3dir): Exploiting 3d geometry and model topology in information retrieval from bim environments. *International Journal of 3-D Information Modeling (IJ3DIM)*, *5*(1), 67–78.

Denis, F., de Temmerman, N., & Rammer, Y. (2017). The potential of graph theories to assess buildings' disassembly and components' reuse: How building information modelling (bim) and social network analysis (sna) metrics might help design for disassembly (dfd). *Proceedings of the HISER International Conference.*

Department of Veterans Affairs. (2010). The va bim guide. Retrieved November 19, 2021, from https://www.cfm.va.gov/til/bim/VA-BIM-Guide.pdf

Department of Veterans Affairs (VA) Office of Construction & Facilities Management (CFM). (2010). *The object element matrix.* Retrieved July 19, 2018, from https://www.cfm.va.gov/til/bim/BIMGuide/downloads/oemf.xls

de Souza, C. B. (2012). Contrasting paradigms of design thinking: The building thermal simulation tool user vs. the building designer. *Automation in Construction*, *22*, 112–122.

DiBiase, D., MacEachren, A. M., Krygier, J. B., & Reeves, C. (1992). Animation and the role of map design in scientific visualization. *Cartography and Geographic Information Systems*, *19*(4), 201–214.

Dickinson, J., & Iordanova, I. (2017). Canadian practice manual for bim. Retrieved May 8, 2021, from https://espace2.etsmtl.ca/id/eprint/19754/

DIN. (2019). Din en 17412:2019-07 building information modelling - level of information need - concepts and principles. Retrieved April 1, 2020, from https://www.beuth.de/en/draft-standard/din-en-17412/306353980

Dodge Data & Analytics. (2017). The business value of bim for infrastructure 2017. Retrieved July 30, 2021, from https://www2.deloitte.com/content/dam/Deloitte/us/Documents/finance/us-fas-bim-infrastructure.pdf

Domeshek, E., & Kolodner, J. L. (1993). Using the points of large cases. *Ai Edam*, *7*(2), 87–96.

Domeshek, E. A., & Kolodner, J. L. (1992). A case-based design aid for architecture. *Artificial intelligence in design'92* (pp. 497–516). Springer.

Donato, V. (2017). Towards design process validation integrating graph theory into bim. *Architectural Engineering and Design Management*, *13*(1), 22–38.

Dong, W., Huang, Y., Lehane, B., & Ma, G. (2020). Xgboost algorithm-based prediction of concrete electrical resistivity for structural health monitoring. *Automation in Construction*, *114*, 103155. https://doi.org/10.1016/j.autcon.2020.103155

Drecki, I. (2002). Visualisation of uncertainty in geographical data. *Spatial data quality*, 140–159.

Dubey, R. K., Khoo, W. P., Morad, M. G., Hölscher, C., & Kapadia, M. (2020). Autosign: A multi-criteria optimization approach to computer aided design of signage layouts in complex buildings. *Computers & Graphics*, *88*, 13–23.

Eastman, C. M., Eastman, C., Teicholz, P., & Sacks, R. (2011). *Bim handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors*. John Wiley & Sons.

Eastman, C., Lee, J.-m., Jeong, Y.-s., & Lee, J.-k. (2009). Automatic rule-based checking of building designs. *Automation in construction*, *18*(8), 1011–1033.

Eisenstadt, V., Lanhgenhan, C., & Althoff, K.-D. (2019). Supporting architectural design process with flea. *International Conference on Computer-Aided Architectural Design Futures*, 58–73.

Ellis, M., & Mathews, E. (2001). A new simplified thermal design tool for architects. *Building and environment*, *36*(9), 1009–1021.

Ergodomus. (2020). Lod (level of development). Retrieved July 31, 2021, from https://www.ergodomus.it/tech%5C_article/lod-level-of-development/

Exner, H., Abualdenien, J., König, M., & Borrmann, A. (2019). Managing building design variants at multiple development levels. *Proc. of the 36th International Council for Research and Innovation in Building and Construction (CIB W78).*

Farrelly, L. (2008). *Basics architecture 01: Representational techniques* (Vol. 1). AVA Publishing.

Ferdinand Tausendpfund GmbH & Co. KG. (Stablished in 1892). Retrieved January 27, 2021, from https://www.tausendpfund.group/

Finger, R., & Bisantz, A. M. (2002). Utilizing graphical formats to convey uncertainty in a decision-making task. *Theoretical Issues in Ergonomics Science*, *3*(1), 1–25.

Fisher, P. F. (1993). Visualizing uncertainty in soil maps by animation. *Cartographica: The International Journal for Geographic Information and Geovisualization*, *30*(2-3), 20–27.

Fisher, P. F. (1999). Models of uncertainty in spatial data. *Geographical Information Systems*, *1*, 191–205.

Flager, F., Welle, B., Bansal, P., Soremekun, G., & Haymaker, J. (2009). Multidisciplinary process integration and design optimization of a classroom building. *Journal of Information Technology in Construction (ITcon)*, *14*(38), 595–612.

Flemming, U. (1994). Case-based design in the seed system. *Automation in construction*, *3*(2-3), 123–133.

Forrest, A. R. (1974). Computational geometry-achievements and problems. *Computer aided geometric design* (pp. 17–44). Elsevier.

Freund, Y., & Schapire, R. E. (1995). A desicion-theoretic generalization of on-line learning and an application to boosting. In P. Vitányi (Ed.), *Computational learning theory* (pp. 23–37). Springer Berlin Heidelberg.

Fuchs, M., Hartmann, F., Henrich, J., Wagner, C., & Zeumer, M. (2013). Snap systematik für nachhaltigkeitsanforderungen in planungswettbewerben-endbericht: Forschungsprogramm zukunft bau / aktenzeichen: 10.08.17.7-11.38: Bundesinstitut für bau-, stadt- und raumforschung (bbsr) im bundesamt für bauwesen und raumordnung (bbr) referat ii 5 nachhaltiges bauen. https://www.nachhaltigesbauen.de/fileadmin/publikationen/SNAP_1_Empfehlungen-korr.pdf

Funkhouser, T., Min, P., Kazhdan, M., Chen, J., Halderman, A., Dobkin, D., & Jacobs, D. (2003). A search engine for 3d models. *ACM Transactions on Graphics (TOG)*, *22*(1), 83–105.

Gänshirt, C. (2012). *Werkzeuge für ideen: Einführung ins architektonische entwerfen*. Walter de Gruyter.

Gao, G., Liu, Y.-S., Wang, M., Gu, M., & Yong, J.-H. (2015). A query expansion method for retrieving online bim resources based on industry foundation classes. *Automation in construction*, *56*, 14–25.

Garland, M. (1999). Quadric-based polygonal surface simplification, carnegie mellon university.

GCR, N. (2004). Cost analysis of inadequate interoperability in the us capital facilities industry. *National Institute of Standards and Technology (NIST).*

Geiss, R., Batz, G. V., Grund, D., Hack, S., & Szalkowski, A. (2006). Grgen: A fast spo-based graph rewriting tool. *International Conference on Graph Transformation*, 383–397.

Gero, J. S., & Kannengiesser, U. (2014). The function-behaviour-structure ontology of design. *An anthology of theories and models of design* (pp. 263–283). Springer.

Géron, A. (2019). *Hands-on machine learning with scikit-learn, keras, and tensorflow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media.

Gershon, N. (1998). Visualization of an imperfect world. *IEEE Computer Graphics and Applications*, *18*(4), 43–45.

Gershon, N. D. (1992). Visualization of fuzzy data using generalized animation. *Proceedings Visualization'92*, 268–273.

Geyer, P., & Singaravel, S. (2018). Component-based machine learning for performance prediction in building design. *Applied energy*, *228*, 1439–1453.

Gigante-Barrera, A., Ruikar, D., Sharifi, S., & Ruikar, K. (2018). A grounded theory based framework for level of development implementation within the information delivery manual. *International Journal of 3-D Information Modeling (IJ3DIM)*, *7*(1), 30–48.

Gilbertson, A. L. (2006). Briefing: Measuring the value of design. *Proceedings of the Institution of Civil Engineers-Municipal Engineer*, *159*(3), 125–128.

Grafarend, E. W. (2006). *Linear and nonlinear models: Fixed effects, random effects, and mixed models*. de Gruyter.

GRAPHISOFT. (2018). *Bimcloud — bim without constraints*. Retrieved May 24, 2018, from http://www.graphisoft.com/bimcloud/overview/

Gravicon. (2012). Common bim requirements 2012 series 3 architectural design. Retrieved November 19, 2021, from https://docplayer.net/20698379-Cobim-series-3-architectural-design-common-bim-requirements-2012-v-1-0.html

Greis, M., Joshi, A., Singer, K., Schmidt, A., & Machulla, T. (2018). Uncertainty visualization influences how humans aggregate discrepant information. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 505.

Griethe, H., & Schumann, H. (2005). Visualizing uncertainty for improved decision making. *Proceedings of the 4th International Conference on Business Informatics Research BIR 2005*, 1–11.

Gross, J. L., & Yellen, J. (2005). *Graph theory and its applications*. Chapman; Hall/CRC.

Grytting, I., Svalestuen, F., Lohne, J., Sommerseth, H., Augdal, S., & Lædre, O. (2017). Use of lod decision plan in bim-projects. *Procedia engineering*, *196*, 407–414.

Gu, N., & London, K. (2010). Understanding and facilitating bim adoption in the aec industry. *Automation in construction*, *19*(8), 988–999.

H. W. and Skeie, G. and Uppstad, B. and Markussen, B. and Sunesen, S. (2018). Mmi-modell modenhets indeks. Retrieved August 12, 2021, from https://www.rif.no/wp-content/uploads/2018/11/mmi-modell-modenhets-indeks.pdf

Habraken, N. J. (1997). Forms of understanding: Thematic knowledge and the modernist legacy. *The Education of the Architect, MIT Press, Cambridge, MA*, 267–293.

Hamieh, A., Makhlouf, A. B., Louhichi, B., & Deneux, D. (2020). A bim-based method to plan indoor paths. *Automation in Construction*, *113*, 103120.

Hanocka, R., Hertz, A., Fish, N., Giryes, R., Fleishman, S., & Cohen-Or, D. (2019). Meshcnn: A network with an edge. *ACM Transactions on Graphics (TOG)*, *38*(4), 1–12.

Harter, H., Schneider-Marin, P., & Lang, W. (2018). The energy grey zone - uncertainty in embedded energy and greenhouse gas emissions assessment of buildings in early design phases. *Proceedings of the Sixth International Symposium on Life-Cycle Civil Engineering (IALCCE 2018).*

Hausknecht, K., & Liebich, T. (2017). *Bim-kompendium: Building information modeling als neue planungsmethode*. Fraunhofer IRB Verlag.

Hawer, S., Schönmann, A., & Reinhart, G. (2018). Guideline for the classification and modelling of uncertainty and fuzziness. *Procedia CIRP*, *67*, 52–57.

He, T., Zhang, J., Lin, J., & Li, Y. (2018). Multiaspect similarity evaluation of bim-based standard dwelling units for residential design. *Journal of Computing in Civil Engineering*, *32*(5), 04018032.

Heisserman, L. (1994). Generative geometric design. *IEEE Computer Graphics and Applications*, *14*(2), 37–45.

Helms, B., & Shea, K. (2012). Computational synthesis of product architectures based on object-oriented graph grammars. *Journal of Mechanical Design*, *134*(2).

Helms, B., Shea, K., & Hoisl, F. (2009). A framework for computational design synthesis based on graph-grammars and function-behavior-structure. *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, *49057*, 841–851.

Hengl, T. (2003). Visualisation of uncertainty using the hsi colour model: Computations with colours. *7th International Conference on GeoComputation.*

Hernández, J. L., Lerones, P. M., Bonsma, P., Delft, A. v., Deighton, R., & Braun, J.-D. (2018). An ifc interoperability framework for self-inspection process in buildings. *Buildings*, *8*(2), 32.

Heylighen, A., Martin, W. M., & Cavallin, H. (2007). Building stories revisited: Unlocking the knowledge capital of architectural practice. *Architectural Engineering and Design Management*, *3*(1), 65–74.

Heylighen, A., & Neuckermans, H. (2001). A case base of case-based design tools for architecture. *Computer-Aided Design*, *33*(14), 1111–1122.

Heylighen, A., Neuckermans, H., Casaer, M., & Dewulf, G. P. (2007). Building memories. *Building Research & Information*, *35*(1), 90–100.

Heylighen, A., Neuckermans, H., & Morisse, P. (2003). What you see is what you get. *IEEE multimedia*, *10*(3), 48–56.

Hietanen, J., & Final, S. (2006). IFC model view definition format. *International Alliance for Interoperability*, 1–29.

Hildebrandt, K., Polthier, K., & Wardetzky, M. (2005). Smooth feature lines on surface meshes. *Symposium on geometry processing*, 85–90.

HKIBIM. (2011). Bim project specification rev 3.0. Retrieved March 19, 2020, from https://silo.tips/download/hong-kong-institute-of-building-information-modelling-bim-project-specification

HOAI. (2013). Honorare für architekten- und ingenieurleistungen (honorarordnung für architekten und ingenieure – hoai). Retrieved August 12, 2021, from https://www.hoai.de/online/HOAI%5C_2013/HOAI%5C_2013.php

Honnibal, M., & Montani, I. (2017). Spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, *7*(1), 411–420.

Hooper, M. (2015). Automated model progression scheduling using level of development. *Construction Innovation*, *15*(4), 428–448.

Hopfe, C. J., & Hensen, J. L. (2011). Uncertainty analysis in building performance simulation for design support. *Energy and Buildings*, *43*(10), 2798–2805.

Hor, A. E., Gunho, S., Claudio, P., Jadidi, M., & Afnan, A. (2018). A semantic graph database for bim-gis integrated information model for an intelligent urban mobility web application. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, *4*(4).

Hor, A. H., Jadidi, A., & Sohn, G. (2016). Bim-gis integrated geospatial information model using semantic web and rdf graphs. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci*, *3*(4), 73–79.

Houde, S., Bonde, S., & Laidlaw, D. H. (2015). An evaluation of three methods for visualizing uncertainty in architecture and archaeology. *Scientific Visualization Conference (SciVis), 2015 IEEE*, 149–150.

Howell, I. (2016). The value information has on decision-making. *New Hampshire Business Review*, *38*, 19.

Hua, K., Fairings, B., & Smith, I. (1996). Cadre: Case-based geometric design. *Artificial Intelligence in Engineering*, *10*(2), 171–183.

Huang, M., Zhu, H., Ninić, J., & Zhang, Q. (2022). Multi-lod bim for underground metro station: Interoperability and design-to-design enhancement. *Tunnelling and Underground Space Technology*, *119*, 104232. https://doi.org/https://doi.org/10.1016/j.tust.2021.104232

Hullman, J. (2019). Why Authors Don't Visualize Uncertainty. *arXiv e-prints*, Article arXiv:1908.01697, arXiv:1908.01697.

Hullman, J., Qiao, X., Correll, M., Kale, A., & Kay, M. (2018). In pursuit of error: A survey of uncertainty visualization evaluation. *IEEE Transactions on Visualization and Computer Graphics*, *25*(1), 903–913.

HULLMAN, J., RESNICK, P., & ADAR, E. (2015). Hypothetical outcome plots outperform error bars and violin plots for inferences about reliability of variable ordering. *PloS One*, *10*(11), e0142444.

INANC, B. S. (2000). Casebook. an information retrieval system for housing floor plans.

INFORMED, B. (2020). Log, loi, lod – was sind die unterschiede? Retrieved July 31, 2021, from https://www.buildinformed.com/log-lod-loi/

INTEGRATED BIM. (2020). What is a lod (level of development). Retrieved July 31, 2021, from https://www.integratedbim.com/post/what-is-a-lod-level-of-development

INTERSCALE. (2020). Bim 101 – bim level of development (lod) explained. Retrieved July 31, 2021, from https://www.interscale.com.au/bim-101-bim-level-of-development-lod-explained/

INVICARA. (2019). Reduce owner costs by leveraging lod with bim. Retrieved July 31, 2021, from https://invicara.com/reduce-owner-costs-leveraging-lod-bim/

ISAAC, S., SADEGHPOUR, F., & NAVON, R. (2013). Analyzing building information using graph theory. *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, *30*, 1.

ISMAIL, A., STRUG, B., & ŚLUSARCZYK, G. (2018). Building knowledge extraction from bim/ifc data for analysis in graph databases. *International Conference on Artificial Intelligence and Soft Computing*, 652–664.

ISO. (2015). Iso 9000:2015(en) quality management systems — fundamentals and vocabulary. Retrieved April 1, 2020, from https://www.iso.org/obp/ui/es//%5C#iso:std:iso:9000:ed-4:v1:en

ISO. (2018). Iso 19650-1: Organization and digitization of information about buildings and civil engineering works, including building information modelling (bim) – information management using building information modelling – part 1: Concepts and principles. Retrieved May 12, 2019, from https://www.iso.org/standard/68078.html

ISO. (2020). Bs en 17412-1:2020 - building information modelling. level of information need. concepts and principles. Retrieved May 12, 2021, from https://shop.bsigroup.com/ProductDetail?pid=000000000030382760

ISO, I., & OIML, B. (1995). Guide to the expression of uncertainty in measurement. *Geneva, Switzerland.*

JAKUMEIT, E., BUCHWALD, S., & KROLL, M. (2010). Grgen. net. *International Journal on Software Tools for Technology Transfer*, *12*(3), 263–271.

JANSON, A., & TIGGES, F. (2014). *Fundamental concepts of architecture: The vocabulary of spatial situations.* Walter de Gruyter.

JOE, C., YOONG, P., & PATEL, K. (2013). Knowledge loss when older experts leave knowledge-intensive organisations. *Journal of knowledge management.*

JOEDICKE, J. (1993). *Entwerfen und gestalten.* Krämer.

Joseph Garcia, A., & Mollaoglu, S. (2020). Individuals' capacities to apply transferred knowledge in aec project teams. *Journal of Construction Engineering and Management*, *146*(4), 04020016.

Jung, N., & Lee, G. (2019). Automated classification of building information modeling (bim) case studies by bim use based on natural language processing (nlp) and unsupervised learning. *Advanced Engineering Informatics*, *41*, 100917.

Karlapudi, J., Valluru, P., & Menzel, K. (2021). Ontological approach for lod-sensitive bim-data management. *Proc. of 9th Linked Data in Architecture and Construction Workshop (LDAC2021).*

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems 30* (pp. 3146–3154). Curran Associates, Inc. http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf

Kell, A., & Mordue, S. (2015). Levels of definitions. Retrieved July 22, 2021, from https://toolkit.thenbs.com/articles/levels-of-definition

Kerzner, H., & Kerzner, H. R. (2017). *Project management: A systems approach to planning, scheduling, and controlling*. John Wiley & Sons.

Khalili, A., & Chua, D. H. K. (2015). Ifc-based graph data model for topological queries on building elements. *Journal of Computing in Civil Engineering*, *29*(3), 04014046.

Kitchenham, B. (2004). Procedures for performing systematic reviews. *Keele, UK, Keele University*, *33*(2004), 1–26.

Klein, G. A., Orasanu, J., Calderwood, R., Zsambok, C. E., et al. (1993). Decision making in action: Models and methods.

Klir, G. J. (1987). Where do we stand on measures of uncertainty, ambiguity, fuzziness, and the like? *Fuzzy sets and systems*, *24*(2), 141–160.

Kneidl, A., Borrmann, A., & Hartmann, D. (2012). Generation and use of sparse navigation graphs for microscopic pedestrian simulation models. *Advanced Engineering Informatics*, *26*(4), 669–680.

Knotten, V., Svalestuen, F., Hansen, G. K., & Lædre, O. (2015). Design management in the building process-a review of current literature. *Procedia Economics and Finance*, *21*, 120–127.

Kolbe, T., Gröger, G., & Plümer, L. (2005). Citygml: Interoperable access to 3d city models. *Geo-information for disaster management* (pp. 883–899). Springer.

Kolltveit, B. J., & Grønhaug, K. (2004). The importance of the early phase: The case of construction and building projects. *International Journal of Project Management*, *22*(7), 545–551.

Kometa, S. T., Olomolaiye, P. O., & Harris, F. C. (1996). A review of client-generated risks to project consultants. *International Journal of Project Management*, *14*(5), 273–279.

Kraft, B., & Nagl, M. (2007). Visual knowledge specification for conceptual design: Definition and tool support. *Advanced Engineering Informatics*, *21*(1), 67–83.

Kreider, R. G., & Messner, J. I. (2013). The uses of bim: Classifying and selecting bim uses. *State College-Pennsylvania*, 0-22.

Kutzner, T., Chaturvedi, K., & Kolbe, T. H. (2020). Citygml 3.0: New functions open up new applications. *PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, *88*(1), 43–61.

Kuznecova, N. G., & Löschmann, M. (2008). Deutsch für architekten: Arbeit am fachwortschatz. *Das Wort: Germanistisches Jahrbuch GUS*, *2008*, 47.

Lai, H., & Deng, X. (2018). Interoperability analysis of ifc-based data exchange between heterogeneous bim software. *Journal of Civil Engineering and Management*, *24*(7), 537–555.

Langenhan, C., Weber, M., Liwicki, M., Petzold, F., & Dengel, A. (2013). Graph-based retrieval of building information models for supporting the early design stages. *Advanced Engineering Informatics*, *27*(4), 413–426.

Lawson, B. (2018). *The design student's journey: Understanding how designers think*. Routledge.

Lee, G., Sacks, R., & Eastman, C. M. (2006). Specifying parametric building object behavior (bob) for a building information modeling system. *Automation in construction*, *15*(6), 758–776.

Lee, J.-H., James, P., Garrett, H., Stephen, P., & Lee, R. (2002). Integrating housing design and case-based reasoning. *School of Architecture and Institute of Complex Engineered Systems (ICES)*.

Leicht, R. M. (2009). A framework for planning effective collaboration using interactive workspaces.

Leite, F., Akcamete, A., Akinci, B., Atasoy, G., & Kiziltas, S. (2011). Analysis of modeling effort and impact of different levels of detail in building information models. *Automation in Construction*, *20*(5), 601–609. https://doi.org/10.1016/j.autcon.2010.11.027

Li, S., & Ying, M. (2003). Region connection calculus: Its models and composition table. *Artificial Intelligence*, *145*(1-2), 121–146.

Liebich, T., Adachi, Y., Forester, J., Hyvarinen, J., Richter, S., Chipman, T., Weise, M., & Wix, J. (2013). Industry foundation classes IFC4 official release. *Online: http://www.buildingsmart-tech.org/ifc/IFC4/final/html*.

Liebich, T. (2013). IFC4—the new buildingSMART standard. *IC Meeting*.

Lilis, G. N., Giannakis, G. I., & Rovas, D. V. (2017). Automatic generation of second-level space boundary topology from IFC geometry inputs. *Automation in Construction*, *76*, 108–124.

LIN, C.-J., & CHIU, M.-L. (2003). Smart semantic query of design information in a case library.

Lin, J.-R., Hu, Z.-Z., Zhang, J.-P., & Yu, F.-Q. (2016). A natural-language-based approach to intelligent data retrieval and representation for cloud bim. *Computer-Aided Civil and Infrastructure Engineering*, *31*(1), 18–33.

Lines, P. (2018). Modeling color difference for visualization design. *IEEE Transactions on Visualization and Computer Graphics*, *24*(1).

Liu, Y., Lather, J., & Messner, J. (2014). Virtual reality to support the integrated design process: A retrofit case study. *Computing in civil and building engineering (2014)* (pp. 801–808).

Liu, Z., Lu, Y., & Peh, L. C. (2019). A review and scientometric analysis of global building information modeling (bim) research in the architecture, engineering and construction (aec) industry. *Buildings*, *9*(10), 210.

lodplanner. (2018). What is lod 500? level of development explained. Retrieved July 31, 2021, from https://www.lodplanner.com/lod-500-explained/

Longley, P. A., Goodchild, M. F., Maguire, D. J., & Rhind, D. W. (2005). *Geographic information systems and science*. John Wiley & Sons.

Love, P. E., & Edwards, D. J. (2004). Determinants of rework in building construction projects. *Engineering, Construction and Architectural Management*, *11*(4), 259–274.

Luebke, D., Reddy, M., Cohen, J., Varshney, A., Watson, B., & Huebner, R. (2003). *Level of detail for 3d graphics*. Morgan Kaufmann.

Luebke, D., Reddy, M., Cohen, J. D., Varshney, A., Watson, B., & Huebner, R. (2003). *Level of detail for 3d graphics*. Morgan Kaufmann.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 4765–4774.

Lundström, C., Ljung, P., Persson, A., & Ynnerman, A. (2007). Uncertainty visualization in medical volume rendering using probabilistic animation. *IEEE Transactions on Visualization and Computer Graphics*, *13*(6), 1648–1655.

MacEachren, A. (1992). Visualizing uncertain information, cartographic perspective, 13.

MacEachren, A. M., Robinson, A., Hopper, S., Gardner, S., Murray, R., Gahegan, M., & Hetzler, E. (2005). Visualizing geospatial information uncertainty: What we know and what we need to know. *Cartography and Geographic Information Science*, *32*(3), 139–160.

MacEachren, A. M., Roth, R. E., O'Brien, J., Li, B., Swingley, D., & Gahegan, M. (2012). Visual semiotics & uncertainty visualization: An empirical study. *IEEE Transactions on Visualization and Computer Graphics*, *18*(12), 2496–2505.

Maher, M. L., Balachandran, M., & Zhang, D. M. (1995). *Case-based reasoning in design*. Psychology Press.

Maier, C. (2015). Grundzüge einer open bim methodik für die schweiz - version 1.0. Retrieved May 29, 2019, from https://www.ebp.ch/sites/default/files/2016-11/ki-leitfaden-open-bim.pdf

Martin, W. M., Heylighen, A., & Cavallin, H. (2003). Building$^2$ stories. a hermeneutic approach to studying design practice. *Proceedings of the 5thEuropean Academy of Design Conference*.

Martin, W. M., Heylighen, A., & Cavallin, H. (2005). The right story at the right time. *AI & SOCIETY*, *19*(1), 34–47.

Masuch, M., & Strothotte, T. (1998). Visualising ancient architecture using animated line drawings. *Proceedings. 1998 IEEE Conference on Information Visualization. An International Conference on Computer Visualization and Graphics (Cat. No. 98TB100246)*, 261–266.

Mattern, H., & König, M. (2018). Bim-based modeling and management of design options at early planning phases. *Advanced Engineering Informatics*, *38*, 316–329.

Mazairac, W., & Beetz, J. (2013). Bimql–an open query language for building information models. *Advanced Engineering Informatics*, *27*(4), 444–456.

Menezes, A. C., Cripps, A., Bouchlaghem, D., & Buswell, R. (2012). Predicted vs. actual energy performance of non-domestic buildings: Using post-occupancy evaluation data to reduce the performance gap. *Applied energy*, *97*, 355–364.

Merkblatt, S. (2010). Graue energie von gebäuden. *Zürich, Schweizerischer Ingenieur-und Architektenverein*.

modelical. (2016). Level of development – bim concepts. Retrieved July 31, 2021, from https://www.modelical.com/en/gdocs/level-of-development/

Möller, T., & Trumbore, B. (2005). Fast, minimum storage ray/triangle intersection. *ACM SIGGRAPH 2005 Courses*, 7.

Monga, O., Armande, N., & Montesinos, P. (1995). Thin nets and crest lines: Application to satellite data and medical images. *Proceedings., International Conference on Image Processing*, *2*, 468–471.

Moniteur, L. (2014). Bim/maquette numérique contenu et niveaux de développement. Retrieved July 19, 2021, from https://www.syntec-ingenierie.fr/wp-content/uploads/2019/06/2014-05-09-cahier-moniteur-bim-maquette-contenu-niveau-developpement.pdf

Morrison, J. L. (1974). A theoretical framework for cartographic generalization with the emphasis on the process of symbolization. *International Yearbook of Cartography*, *14*(1974), 115–27.

Mubarak, K. (2004). Case based reasoning for design composition in architecture. *Pittsburgh, Carnegie Melon University*.

Mullins, J., Ling, Y., Mahadevan, S., Sun, L., & Strachan, A. (2016). Separation of aleatory and epistemic uncertainty in probabilistic model validation. *Reliability Engineering & System Safety*, *147*, 49–59.

Munzner, T. (2014). *Visualization analysis and design*. AK Peters/CRC Press.

Murray, J. A., Bradley, H., Craigie, W. A., & Onions, C. T. (1961). *The oxford english dictionary* (Vol. 11). Clarendon Press Oxford.

NATSPEC. (2013a). Bim and lod - building information modeling and level of development. Retrieved July 19, 2021, from https://bim.natspec.org/images/NATSPEC%5C_Documents/%5C%5CNATSPEC%5C_BIM%5C_LOD%5C_Paper%5C_131115.pdf

NATSPEC. (2013b). *Natspec national bim guide*. Retrieved July 19, 2018, from https://bim.natspec.org/documents/natspec-national-bim-guide

Ness, B., Urbel-Piirsalu, E., Anderberg, S., & Olsson, L. (2007). Categorising tools for sustainability assessment. *Ecological economics*, *60*(3), 498–508.

Neuckermans, H., Heylighen, A., & Morisse, P. (2002). Visual keys to architectural design.

New York City, D. o. D. +. C. (2012). New york city - bim guidelines. Retrieved November 19, 2021, from http://www.nyc.gov/html/ddc/downloads/pdf/DDC%5C_BIM%5C_Guidelines.pdf

NIBS. (2013). National bim standards us v3_2.7. Retrieved May 7, 2021, from https://classes.engr.oregonstate.edu/cce/winter2018/cce203/NBIMS-US%5C_V3/NBIMS-US%5C_V3%5C_2.7%5C_LOD%5C_Specification%5C_2013.pdf

NIBS. (2017). National bim guide for owners. Retrieved May 8, 2021, from https://www.nibs.org/files/pdfs/NIBS%5C_BIMC%5C_NationalBIMGuide.pdf

Nikhila, R., Jeremy, R., David, N., Taylor, G., Lo, W., Justin, J., & Georgia, G. (2020). Accelerating 3d deep learning with pytorch3d.

Nilsen, T., & Aven, T. (2003). Models and model uncertainty in the context of risk analysis. *Reliability Engineering & System Safety*, *79*(3), 309–317.

Odusami, K. T. (2002). Perceptions of construction professionals concerning important skills of effective project leaders. *Journal of Management in Engineering*, *18*(2), 61–67.

OhioDAS. (2011). State of ohio bim protocol. Retrieved March 19, 2020, from https://slidelegend.com/state-of-ohio-bim-protocol-ohio-department-of-administrative-services%5C_59dc55a51723ddfffb83725f.html

OmniClass, A. (2012). A strategy for classifying the built environment. Retrieved January 11, 2018, from http://www.omniclass.org/

openml.org. (2020). 10-fold crossvalidation. Retrieved November 1, 2020, from https://www.openml.org/a/estimation-procedures/1

Oxman, R. E. (1994). Precedents in design: A computational model for the organization of precedent knowledge. *Design studies*, *15*(2), 141–157.

Pang, A. (2001). Visualizing uncertainty in geo-spatial data. *Proceedings of the Workshop on the Intersections between Geospatial Information and Information Technology*, 1–14.

Patlakas, P., Livingstone, A., Hairstans, R., & Neighbour, G. (2018). Automatic code compliance with multi-dimensional data fitting in a bim context. *Advanced Engineering Informatics*, *38*, 216–231.

Pearson, K. (1896). Vii. mathematical contributions to the theory of evolution.—iii. regression, heredity, and panmixia. *Philosophical Transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character*, (187), 253–318.

Penttilä, H. (2007). Early architectural design and bim. *Computer-aided architectural design futures (caadfutures) 2007* (pp. 291–302). Springer.

Polanyi, M. (2009). *The tacit dimension*. University of Chicago press.

Porter, S., Tan, T., Tan, T., & West, G. (2014). Breaking into bim: Performing static and dynamic security analysis with the aid of bim. *Automation in Construction*, *40*, 84–95.

PROGETTIAMOBIM. (2018). I gradi dei lod. le uni 11337-4 confrontate con la scala americana. Retrieved November 29, 2019, from https://www.progettiamobim.com/blog/approfondimenti/i-gradi-dei-lod/

Program, C. I. C. R. (2012). Bim planning guide for facility owners v1.0. Retrieved November 19, 2021, from https://oliebana.files.wordpress.com/2012/10/bim%5C_planning%5C_guide%5C_for%5C_facility%5C_owner-version%5C_1-021.pdf

Program, C. I. C. R. (2013). Bim planning guide for facility owners v2.0. Retrieved May 7, 2021, from https://bim.psu.edu/owners%5C_guide/

Quinlan, J. R. (1996a). Bagging, boosting, and c4. 5. *AAAI/IAAI, Vol. 1*, 725–730.

Quinlan, J. R. (1996b). Learning decision tree classifiers. *ACM Computing Surveys (CSUR)*, *28*(1), 71–72.

Raileanu, L., & Stoffel, K. (2004). Theoretical comparison between the gini index and information gain criteria. *Annals of Mathematics and Artificial Intelligence*, *41*(1), 77–93.

Raskin, V., & Taylor, J. (2014). Fuzziness, uncertainty, vagueness, possibility, and probability in natural language. *Norbert Wiener in the 21st Century (21CW), 2014 IEEE Conference on*, 1–6.

Rasmussen, M. H., Lefrançois, M., Schneider, G. F., & Pauwels, P. (2019). Bot: The building topology ontology of the w3c linked building data group. *Semantic Web*, (Preprint), 1–19.

REBIM. (2020). Level of detail or development: Lod in bim. Retrieved July 31, 2021, from https://rebim.io/level-of-detail-or-development-lod-in-bim/

Rezaee, R., Brown, J., Augenbroe, G., & Kim, J. (2015). Assessment of uncertainty and confidence in building design exploration. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, *29*(4), 429–441. https://doi.org/10.1017/S0890060415000426

RIBEIRO, M. T., SINGH, S., & GUESTRIN, C. (2016). "why should i trust you?" explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135–1144.

RIBICIC, H., WASER, J., GURBAT, R., SADRANSKY, B., & GRÖLLER, M. E. (2012). Sketching uncertainty into simulations. *IEEE Transactions on Visualization and Computer Graphics*, *18*(12), 2255–2264.

RICHTER, K., HEYLIGHEN, A., & DONATH, D. (2007). Looking back to the future.

RIJKSGEBOUWENDIENST. (2012). Building information modeling (bim) guidelines v1.6. Retrieved November 19, 2021, from https://english.rijksvastgoedbedrijf.nl/binaries/central-government-real-estate-agency/documents/publication/2014/07/08/rgd-bim-standard-v1.0.1-en-v1.0%5C_2/Rgd%5C_BIM%5C_Standard%5C_v1%5C_0%5C_1%5C_EN%5C_v1%5C_0%5C_%5C_2%5C_.pdf

RITTEL, H. W., & WEBBER, M. M. (1973). Dilemmas in a general theory of planning. *Policy sciences*, *4*(2), 155–169.

RIVARD, H., & FENVES, S. J. (2000). A representation for conceptual design of buildings. *Journal of Computing in Civil Engineering*, *14*(3), 151–159.

ROTH, R. E. (2009). A qualitative approach to understanding the role of geographic information uncertainty during decision making. *Cartography and Geographic Information Science*, *36*(4), 315–330.

ROWLEY, J. (2007). The wisdom hierarchy: Representations of the dikw hierarchy. *Journal of information science*, *33*(2), 163–180.

RUIZ-MONTIEL, M., BONED, J., GAVILANES, J., JIMÉNEZ, E., MANDOW, L., & PÉREZ-DE-LA-CRUZ, J.-L. (2013). Design with shape grammars and reinforcement learning. *Advanced Engineering Informatics*, *27*(2), 230–245.

RÜPPEL, U., ABOLGHASEMZADEH, P., & STÜBBE, K. (2010). Bim-based immersive indoor graph networks for emergency situations in buildings. *Proceedings of the International Conference on Computing in Civil and Building Engineering*, *65*.

RYLE, G. (2009). *The concept of mind*. Routledge.

SACKS, R., EASTMAN, C., LEE, G., & TEICHOLZ, P. (2018). *Bim handbook: A guide to building information modeling for owners, designers, engineers, contractors, and facility managers*. John Wiley & Sons.

SALAMA, D. M., & EL-GOHARY, N. M. (2016). Semantic text classification for supporting automated compliance checking in construction. *Journal of Computing in Civil Engineering*, *30*(1), 04014106.

SANGUINETTI, P., EASTMAN, C., & AUGENBROE, G. (2009). Courthouse energy evaluation: Bim and simulation model interoperability in concept design. *Eleventh International IBPSA Conference*, 1922–1929.

SCHMIDT-BELZ, B., & HOVESTADT, L. (1996). Scenario of an integrated design support for architects. *Design Studies*, *17*(4), 489–509.

Schneider-Marin, P., & Abualdenien, J. (2019). A framework to facilitate an interdisciplinary design process using bim. *Proc. of the 31th Forum Bauinformatik.*

Schneider-Marin, P., Harter, H., Tkachuk, K., & Lang, W. (2020). Uncertainty analysis of embedded energy and greenhouse gas emissions using bim in early design stages. *Sustainability*, *12*(7), 2633.

Schön, D. A. (1987). Educating the reflective practitioner.

Schulz, C., Schatz, K., Krone, M., Braun, M., Ertl, T., & Weiskopf, D. (2018). Uncertainty visualization for secondary structures of proteins. *2018 IEEE Pacific Visualization Symposium (PacificVis)*, 96–105.

Schweizerischer Ingenieur- und Architekten-Verein. (2004). *Snarc - systematik zur beurteilung der nachhaltigkeit von architekturprojekten für den bereich umwelt (dokumentation)*. SIA Schweizerischer Ingenieur- u. Architektenverein.

See, R., Karlshoej, J., & Daves, D. (2012). An integrated process for delivering IFC based data exchange.

Services, L. (2014). Lod – development or detail & why it matters. Retrieved July 31, 2021, from http://lanmarservices.com/2014/05/14/lod-in-scan-to-bim/

Shafiq, M. T., Matthews, J., & Lockley, S. (2013). A study of bim collaboration requirements and available features in existing model collaboration systems. *Journal of Information Technology in Construction (ITcon)*, *18*, 148–161.

Shapira, L., Shamir, A., & Cohen-Or, D. (2008). Consistent mesh partitioning and skeletonisation using the shape diameter function. *The visual computer*, *24*(4), 249.

Shikhare, D. (2001). Complexity of geometric models. *National Centre for Software Technology, Mumbai.*

Singaravel, S., Suykens, J., & Geyer, P. (2018). Deep-learning neural-network architectures and methods: Using component-based models in building-design energy prediction. *Advanced Engineering Informatics*, *38*, 81–90.

Singh, M. M., Singaravel, S., Klein, R., & Geyer, P. (2020). Quick energy prediction and comparison of options at the early design stage. *Advanced Engineering Informatics*, *46*, 101185.

Smith, I., Lottaz, C., & Faltings, B. (1995). Spatial composition using cases: Idiom. *International Conference on Case-Based Reasoning*, 88–97.

Smith, I., Stalker, R., & Lottaz, C. (1996). Creating design objects from cases for interactive spatial composition. *Artificial intelligence in design'96* (pp. 97–116). Springer.

Smith Mason, J., Retchless, D., & Klippel, A. (2017). Domains of uncertainty visualization research: A visual summary approach. *Cartography and Geographic Information Science*, *44*(4), 296–309.

Solihin, W., & Eastman, C. M. (2016). A knowledge representation approach in bim rule requirement analysis using the conceptual graph. *ITcon*, *21*, 370–401.

SpaCy. (2021). Word vectors and semantic similarity. Retrieved January 29, 2021, from https://spacy.io/usage/vectors-similarity

Stadler, A., & Kolbe, T. H. (2007). Spatio-semantic coherence in the integration of 3d city models. *Proceedings of the 5th International Symposium on Spatial Data Quality, Enschede.*

Steinmann, F. (1997). Modellbildung und computergestütztes modellieren in frühen phasen des architektonischen entwurfs.

Struck, C., Kotek, P., & Hensen, J. (2007). On incorporating uncertainty analysis in abstract building performance simulation tools. *Proceedings of the 12th Symp for Building Physics.*

Strug, B., & Ślusarczyk, G. (2017). Reasoning about accessibility for disabled using building graph models based on bim/ifc. *Visualization in Engineering*, *5*(1), 1–12.

Štrumbelj, E., & Kononenko, I. (2014). Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, *41*(3), 647–665.

Sujan, S. F., Kiviniemi, A., Jones, S. W., Wheathcroft, J. M., & Hjelseth, E. (2019). Common biases in client involved decision-making in the aec industry. *Frontiers of Engineering Management*, 1–18.

Taha, D., Hosni, S., Sueyllam, H., & Streich, B. (2007). The role of cases in architectural practice and education moneo: An architectural assistant system.

Tang, A., Babar, M. A., Gorton, I., & Han, J. (2006). A survey of architecture design rationale. *Journal of systems and software*, *79*(12), 1792–1804.

Tekla. (2021). Lod: Ausführungsreife modelle. Retrieved July 31, 2021, from https://www.tekla.com/de/bimwissen/bim/lod-bim-ausfuehrungsreife-modelle

The Port Authority of NY & NJ Engineering Department. (2012). E, a design division bim standard. Retrieved November 19, 2021, from https://www.panynj.gov/content/dam/port-authority/pdfs/-available-engineering-documents/EAD-BIM-Standard-Manual.pdf

Tian, W., Heo, Y., De Wilde, P., Li, Z., Yan, D., Park, C. S., Feng, X., & Augenbroe, G. (2018). A review of uncertainty analysis in building energy assessment. *Renewable and Sustainable Energy Reviews*, *93*, 285–301.

Tribelsky, E., & Sacks, R. (2010). The relationship between information flow and project success in multi-disciplinary civil engineering design. *Proceedings of the IGLC, Haifa, Israel*, 14–16.

Trimble. (2013). Project progression planning with mps 3.0. Retrieved March 19, 2020, from http://support.vicosoftware.com/FlareFiles/Content/KB/Trimble%5C%20-%5C%20Progression%5C%20Planning%5C%20V15.pdf

Uhm, M., Lee, G., Park, Y., Kim, S., Jung, J., & Lee, J.-k. (2015). Requirements for computational rule checking of requests for proposals (rfps) for building designs in south korea. *Advanced Engineering Informatics*, *29*(3), 602–615.

United BIM. (2020). Bim level of development | lod 100, 200, 300, 350, 400, 500. Retrieved July 31, 2021, from https://www.united-bim.com/bim-level-of-development-lod-100-200-300-350-400-500/

University of Florida (UF). (2011). Bim execution plan v1.1. Retrieved November 19, 2021, from https://www.scribd.com/document/349210619/Bim-Execution-Plan

U.S General Services Administration (GSA). (2018). Level of detail. Retrieved July 30, 2021, from https://www.gsa.gov/real-estate/design-construction/3d4d-building-information-modeling/guidelines-for-bim-software/document-guides/level-of-detail

USACE. (2014). Minimum model element matrix m3 v1.3 (attributes). Retrieved July 19, 2021, from https://www.lrl.usace.army.mil/Portals/64/docs/Engineering/LDMDG/%5C%5CChapter15%5C_CADDBIM%5C_2020098.pdf?ver=2020-09-01-103201-337

USC. (2012). Building information modeling (bim) guidelines v1.6. Retrieved November 19, 2021, from https://vdocument.in/negotiated-contracts-addendum-to-usc-bim-including-model-assembly-in-navisworks.html

USIBD. (2019). Level of accuracy (loa) specification version 3.0. Retrieved August 9, 2021, from https://usibd.org/product/level-of-accuracy-loa-specification-version-3-0/

van Berlo, L., & Bomhof, F. (2014). Creating the dutch national bim levels of development. *Computing in civil and building engineering* (pp. 129–136).

Van Dronkelaar, C., Dowson, M., Burman, E., Spataru, C., & Mumovic, D. (2016). A review of the energy performance gap and its underlying causes in non-domestic buildings. *Frontiers in Mechanical Engineering*, *1*, 17.

Van Gelder, L., Janssen, H., & Roels, S. (2014). Probabilistic design and analysis of building performances: Methodology and application example. *Energy and Buildings*, *79*, 202–211.

VBI. (2016). Bim-leitfaden für die planerpraxis. Retrieved May 8, 2019, from https://www.vbi.de/fileadmin/redaktion/Dokumente/Infopool/Downloads/VBI_BIM-Leitfaden_0916-final.pdf

VicoSoftware. (2005). Bim level of detail. Retrieved May 13, 2019, from http://www.vicosoftware.com

Vilgertshofer, S., & Borrmann, A. (2017). Using graph rewriting methods for the semi-automatic generation of parametric infrastructure models. *Advanced Engineering Informatics*, *33*, 502–515. https://doi.org/10.1016/j.aei.2017.07.003

Vollmer, M., Schneider-Marin, P., Harter, H., & Lang, W. (2019). *Ferd. tausendpfund - lebenszyklusanalyse und gebäudemonitoring* [isbn: 978-3-948278-03-8, url: https://bit.ly/37ChlbS (visited on 2021-02-22)]. Bayerischer Bauindustrieverband e.V. (BBIV). https://www.ppe.tum.de/fileadmin/w00bqx/www/content_uploads/20191206_Endbericht_Tausendpfund.pdf

Voss, A. (1997). Case design specialists in fabel. *Issues and applications of case-based reasoning in design*, 301–335.

Weber, C., Hahmann, S., & Hagen, H. (2010). Sharp feature detection in point clouds. *2010 Shape Modeling International Conference*, 175–186.

Webster, M. (2003). Communicating climate change uncertainty to policy-makers and the public. *Climatic Change*, *61*(1-2), 1.

Wesz, J. G. B., Formoso, C. T., & Tzortzopoulos, P. (2018). Planning and controlling design in engineered-to-order prefabricated building systems. *Engineering, Construction and Architectural Management*, *25*(2), 134–152.

Weytjens, L., & Verbeeck, G. (2010). Towards' architect-friendly'energy evaluation tools. *Proceedings of the 2010 Spring Simulation Multiconference*, 179.

Wheeler, D. J., Chambers, D. S. et al. (1992). *Understanding statistical process control*. SPC press.

Wiesbaden, S. F. (2013). *Hoai 2013-textausgabe/hoai 2013-text edition: Honorarordnung für architekten und ingenieure vom 10. juli 2013/official scale of fees for services by architects and engineers dated july 10, 2013*. Springer-Verlag.

Wilbur, W. J., & Sirotkin, K. (1992). The automatic identification of stop words. *Journal of information science*, *18*(1), 45–55.

Wong, K., & Ellul, C. (2016). Using geometry-based metrics as part of fitness-for-purpose evaluations of 3d city models. *ISPRS Annals*, *4*, 129–136.

Wu, S., Shen, Q., Deng, Y., & Cheng, J. (2019). Natural-language-based intelligent retrieval engine for bim object database. *Computers in Industry*, *108*, 73–88.

Wynn, D. C., Grebici, K., & Clarkson, P. J. (2011). Modelling the evolution of uncertainty levels during design. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, *5*(3), 187.

Yang, D., & Eastman, C. M. (2007). A rule-based subset generation method for product data models. *Computer-Aided Civil and Infrastructure Engineering*, *22*(2), 133–148.

Young, N., Jones, S. A., Bernstein, H. M., & Gudgel, J. (2009). The business value of bim-getting building information modeling to the bottom line. *Bedford, MA: McGraw-Hill Construction*, *51*.

Zahedi, A., Abualdenien, J., Petzold, F., & Borrmann, A. (2019). Minimized communication protocol based on a multi-lod meta-model for adaptive detailing of bim models. *Proc. of the 26th International Workshop on Intelligent Computing in Engineering*.

Zahedi, A., Abualdenien, J., Petzold, F., & Borrmann, A. (2022). Bim-based design decisions documentation using design episodes, explanation tags, and constraints. *Journal of Information Technology in Construction*, *27*, 756–780. https://doi.org/https://dx.doi.org/10.36680/j.itcon.2022.037

Zanni, M., Sharpe, T., Lammers, P., Arnold, L., & Pickard, J. (2019). Standardization of whole life cost estimation for early design decision-making utilizing bim. *Advances in informatics and computing in civil and construction engineering* (pp. 773–779). Springer.

Zeiler[1], W., Savanovic[1], P., & Quanjel[1], E. (2007). Design decision support for the conceptual phase of the design process.

Zhang, C., Zhu, A.-X., Zhou, L., Che, M., & Qiu, T. (2020). Constraints for improving information integrity in information conversion from cad building drawings to bim model. *IEEE Access*, *8*, 81190–81208.

Zhang, D., Johnson, A., Hebert, M., & Liu, Y. (1998). On generating multi-resolution representations of polygonal meshes.

Zhang, J., Ma, G., Huang, Y., sun, J., Aslani, F., & Nener, B. (2019). Modelling uniaxial compressive strength of lightweight self-compacting concrete using random forest regression. *Construction and Building Materials*, *210*, 713–719. https://doi.org/10.1016/j.conbuildmat.2019.03.189