Technische Universität München
TUM School of Engineering and Design

Context-sensitive linking of heterogeneous information models from the building and the urban domain

Stefan Fritz Beck

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

Doktors der Ingenieurwissenschaften

genehmigten Dissertation.

Vorsitz: Prof. Dr.-Ing. Frank Petzold

Prüfer*innen der Dissertation:
1. Prof. Dr.-Ing. André Borrmann
2. Prof. Dr. rer. nat. Thomas H. Kolbe

Die Dissertation wurde am 28.11.2022 bei der Technische Universität München eingereicht und durch die TUM School of Engineering and Design am 21.03.2023 angenommen.

Abstract

In recent years, there has been an increasing awareness of the potential of synergies between domains dealing with building information like Building Information Modelling (BIM) and domains dealing with urban information like Urban Information Modelling (UIM). One way to exploit the potential of synergies between the domains BIM and UIM is linking respective information models at instance-level. When linking information models, corresponding objects are identified and linked subsequently. However, information models are generally developed from different perspectives and for different purposes so that corresponding objects often do not match exactly. This dissertation addresses the question whether these kinds of mismatches cause problems for linking information models from the domains BIM and UIM and how those problems can be overcome. To answer these questions, this dissertation introduces a new perspective on information integration systems which is based on the discourse on computer-based information systems from a system theoretical perspective. Furthermore, the semantics of links are discussed based on the concept of model-theoretic semantics with the result that links are often limited to a specific validity scope, also known as contextual link. In addition, an extensive literature review emphasizes that information integration systems in the field of BIM-UIM integration are often idealized, so that different contextual variables are not considered in respective research works. Following the discourse on information integration systems and semantics of links, two types of context-sensitive linking were identified, namely context-dependent linking and contextual linking. Context-dependent linking means that variable input to information integration system, such as different source models, require different linkage. Contextual linking means that the implicit validity scope of the links is opaque to the user so that the links are pone for misinterpretations. These two types of context-sensitive linking refer to the problems caused by the mismatch of corresponding objects and occur in so called context-sensitive information integration systems. Furthermore, design principles on the implementation of information integration systems dealing with context-dependent and contextual linking were deduced and subsequently demonstrated. As a result, information integration systems dealing with context-dependent linking require an adaptive matching mechanism, while those dealing with contextual linking require making the implicit validity scope explicit. The dissertation contributes to the research knowledge through the introduction of a new perspective on information integration systems, semantics of links and context-sensitive linking. Furthermore, the dissertation results in design principles defining how to overcome problems within context-sensitive information integration systems caused by the mismatch of corresponding objects of information models from the domains BIM and UIM.

Zusammenfassung

In den vergangenen Jahren gab es ein wachsendes Interesse am Synergiepotenzial zwischen den Fachbereichen der Gebäudeinformation, wie Building Information Modeling (BIM), und der Stadtinformation, wie Urban Information Modelling (UIM). Eine Möglichkeit dieses Synergiepotenzial zu nutzen, ist die Verlinkung der jeweiligen Informationsmodelle auf Instanzebene. Bei der Verlinkung von Informationsmodellen werden zugehörige Objekte identifiziert und anschließend verknüpft. Informationsmodelle werden jedoch in der Regel aus unterschiedlichen Perspektiven und für unterschiedliche Zwecke erstellt, so dass diese Objekte oft nicht exakt übereinstimmen. Diese Dissertation befasst sich mit der Frage, ob diese Art von Nichtübereinstimmung Probleme bei der Verlinkung von Informationsmodellen aus den Fachbereichen BIM und UIM verursacht und wie diese überwunden werden können. Zur Beantwortung dieser Fragen wird in der Dissertation eine neue Perspektive auf Informationsintegrationssysteme eingeführt, die auf dem Diskurs über computergestützte Informationssysteme aus systemtheoretischer Sicht beruht. Darüber hinaus wird die Semantik von Links auf der Grundlage modelltheoretischer Semantik diskutiert, mit dem Ergebnis, dass Verlinkungen häufig auf einen bestimmten Gültigkeitsbereich beschränkt sind, was auch als kontextuelle Verlinkung bezeichnet wird. Zudem hat eine umfangreiche Literaturrecherche ergeben, dass Informationsintegrationssysteme im Bereich der BIM-UIM Integration oft idealisiert sind, so dass Kontextvariablen in den entsprechenden Forschungsarbeiten nicht berücksichtigt werden. In Anlehnung an den Diskurs über Informationsintegrationssysteme und die Semantik von Links wurden zwei Arten der kontextsensitiven Verlinkung identifiziert, nämlich die kontextabhängige Verlinkung und die kontextuelle Verlinkung. Kontextabhängige Verlinkung bedeutet, dass Änderungen im Informationsintegrationssystem, wie zum Beispiel unterschiedliche Funktionen, unterschiedliche Verlinkungen erfordern. Kontextabhängige Verlinkung bedeutet, dass der implizite Gültigkeitsbereich der Link für den Nutzer nicht einsehbar ist, sodass ein Risko besteht den Link fehlzuinterpretieren. Diese beiden Arten der kontextsensitiven Verlinkung beziehen sich auf die Probleme, die durch die Nichtübereinstimmung zugehöriger Objekte verursacht werden und in so genannten kontextsensitiven Informationsintegrationssystemen auftreten. Darüber hinaus wurden Gestaltungsprinzipien für die Implementierung von Informationsintegrationssystemen, in denen kontextabhängige und kontextuelle Verknüpfungen zu berücksichtigen sind, abgeleitet und anschließend demonstriert. Informationsintegrationssysteme mit kontextabhängiger Verlinkung benötigen einen adaptiven Verlinkungsmechanismus, und diejenigen mit kontextabhängiger Verlinkung benötigen eine explizite Festlegung des Gültigkeitsbereich des Links. Die Dissertation trägt zum Forschungswissen durch die Einführung einer neuen Perspektive auf Informationsintegrationssysteme, die Semantik von Links und die kontextsensitive Verlinkung bei. Darüber hinaus resultiert die Dissertation in Entwurfsprinzipien, die definieren, wie Probleme innerhalb nicht-idealisierter Informationsintegrationssysteme überwunden werden können, die durch die Nichtübereinstimmung der entsprechenden Objekte von Informationsmodellen aus den Fachbereichen BIM und UIM verursacht werden.

Danksagung

Zuallererst möchte ich mich bei **Getrud Obermeyer** bedanken, die dieses Promotionsstudium und die damit verbundene Forschung durch das Gertrud-Obermeyer Stipendium ermöglicht hat. Ich hoffe, dass meine Forschung Ihren Vorstellungen zur Verwendung der Stipendiengelder gerecht werden konnte.

Zudem möchte ich mich bei allen **Mitarbeitern** des Lehrstuhls für Computergestützte Modellierung and Simulation (CMS), des Lehrstuhls für Geoinformatik und des Leonhard Obermeyer Centers (LOC) bedanken, die mich einerseits fachlich, aber auch freundschaftlich unterstützt haben. Hier möchte ich mich insbesondere bei den Professoren André Borrmann und Thomas Kolbe, bei meinem Kollegen Ihab Hijazi und den Gruppenleitern des Lehrstuhls CMS Jimmy Abualdenien, Alexander Braun und Simon Vilgertshofer bedanken, die ich stets um Rat fragen durfte und mir mit Ihrer fachlichen Expertise zur Seite standen.

Zum Schluss möchte ich mich bei meiner **Familie** bedanken, die mich sowohl auf dem Weg zum Promotionsstudium als auch währenddessen in vielfältiger Weise unterstützt hat. Hier möchte ich mich insbesondere bei meiner Frau Annika Beck bedanken, die mich stets bekräftigt hat, und meiner Mutter Ulrike Beck, mit der ich mich stets über anstehende Herausforderungen austauschen konnte. Auch wenn mein Vater Michael Beck das Promotionsstudium leider nicht mehr mitbegleiten konnte, bin ich mir sicher, dass er mich hierbei gerne unterstützt hätte.

List of content

List of abbreviation **9**

1 Introduction **10**

1.1 Motivation 10
1.2 Research method 12
1.3 Thesis structure 16
1.4 Scope & assumptions 18
1.5 Reference example 19

2 Fundamentals **21**

2.1 Ontologies and information models 21
2.2 Formal languages 26
2.2.1 UML 26
2.2.2 XML/ XSD 27
2.2.3 SPF/ EXPRESS 27
2.2.4 RDF(S)/ OWL 28
2.3 Information models 30
2.3.1 IFC 30
2.3.2 BOT 31
2.3.3 CityGML 32
2.4 Modelling the built environment 34
2.4.1 Building Information Modelling 34
2.4.2 Urban Information Modelling 34
2.4.3 Comparison 35
2.5 The keyword „BIM-GIS Integration" 37
2.6 Semantic Web and Linked Data 38

3 Related Literature **45**

3.1 Research structure 45
3.2 BIM-GIS Integration 45
3.3 Semantic Web: Contextual linking 59
3.4 Summary 65

4 Research problem & objectives **67**

4.1 Research gap 67
4.2 Task problem 68

4.3 Objectives ... 73

5 Use Cases 74

5.1 Categorization approaches ... 74
5.2 Physical subject ... 74
5.3 Activity of use case ... 75

6 Instance-level heterogeneity 82

7 Information integration systems 86

7.1 Computer-based information system ... 86
7.2 System structure ... 91
7.2.1 Overview ... 91
7.2.2 Integration system ... 92
7.2.3 Application system ... 98
7.2.4 Communication system ... 105
7.3 Implementation ... 107

8 The rationale behind linking 110

8.1 Modell-theoretic semantics ... 110
8.2 Semantics of links ... 112
8.3 Information integration systems ... 119

9 Context-sensitive linking 120

9.1 Idealized information integration system ... 120
9.2 Context-sensitive information integration systems ... 121
9.3 Context-dependent linking ... 123
9.4 Contextual linking ... 130

10 Design Principles 132

10.1 Implementation ... 132
10.2 Design Principle 1: Context-dependent linking ... 139
10.2.1 Description ... 139
10.2.2 Implementation ... 139
10.2.3 Demonstration ... 140
10.2.4 Evaluation ... 146
10.2.5 Conclusion ... 147
10.3 Design Principle 2: Contextual linking ... 148
10.3.1 Description ... 148
10.3.2 Implementation ... 148
10.3.3 Demonstration ... 150
10.3.4 Evaluation ... 156

10.3.5 Conclusion 157

11 Discussion & future research 158

11.1 Summary..... 158
11.2 Discussion 166
11.3 Future research 169

References 170

List of abbreviations

9IM 9-intersection model

ADE Application Domain Extension

AEC Architecture Engineering and Construction

BIM Building Information Modelling

BOT Building Topology Ontology

CAD Computer-Aided Design, Computer-Aided Design

DSR Design Science Research

EDOAL Expressive and Declarative Ontology Alignment Language

ETL Extract Transform Load

GIS Geospatial Information System

GML Geographic Markup Language

ICT Information and communication technologies

IFC Industry Foundation Classes

MOF Meta-Object Facility

OMG Object Management Group

OWL Web Ontology Language

RDF Resource Description Framework

SKOS Simple Knowledge Organization System

SO Similarity Ontology

STEP Standard for Exchange of Product data

UIM Urban Information Modelling

UML Unified Modelling Language

VoID Vocabulary of Interlinked Datasets

XML Extensible Markup Language

1 Introduction

1.1 Motivation

In industries related to the built environment, the division of labor [1] has led to domains that have evolved to a large degree independent of each other, like Building Information Modeling (BIM) and Urban Information Modeling (UIM). Both BIM and UIM belong to respective information scopes such as building and urban information, and respective task specifications requiring this kind of information [2–4]. Apart from that, some task specifications require both building and urban information which can be expressed, for example, through cross-domain queries: Which dimensions and materials do windows next to high-traffic roads have? Is there a pedestrian walkway next to the gate which is the main entrance? What is the height of the fence of all kindergartens being placed adjacent to heavy traffic roads? These kinds of task specifications demand synergies of the domains BIM and UIM through the creation of interfaces relating to their underlying computer-based information systems [2–4].

In recent decades, there has been an increased awareness of the potential for synergies between the domains BIM and UIM [4–6], which can be attributed to **two major changes**:

- First, the *requirements for information* are increasing in these domains due to increasingly complex environmental demands. The increasing complexity of environmental demand is mainly due to demographic, environmental, and technological changes, as well as the changing needs and expectations of passengers and stakeholders. Thus, information sufficient for meeting the task requirements a few years ago is likely to be insufficient today. A particular approach aiming to meet the information requirements is sharing information across domains which is also known as lateral communication [7].
- Second, the preceding *digital transformation* is accompanied by new technological opportunities for sharing information across domains. On one hand, this is due to the use of new information and communication technologies (ICT) that enable the implementation of interfaces between computer-based information systems belonging to these domains. On the other hand, this is due to the increasing amount of data that is digitally captured and subsequently exchanged between these computer-based information systems.

In a nutshell, the need for synergies between the domain BIM and UIM comes from both increasingly complex environmental demands and new technological opportunities.

One way to exploit the potential of synergies between the domains BIM and UIM is by sharing information across respective computer-based information systems with the help of so-called ***information models*** *[4,8,9]*. Information models related to the domains BIM and UIM are generally developed from different perspectives and for different purposes which results in differences between these information models, also called heterogeneity [4,10]. Information models related to the domain BIM represent real-world objects like buildings which are primarily described from a prescriptive view for design and maintenance purposes. On the other hand, information models related to the domain UIM represent real-world objects like cities which are primarily described from a descriptive view for spatial analysis purposes [4,11–14]. The consequent heterogeneity of the information models causes information silos. These kinds of information silos need to be bridged to efficiently share information across the domains BIM and UIM so that the full potential of the synergies between these domains can be exploited.

These kinds of information silos can be bridged by *integrating* the respective information models. An increasingly relevant integration method in the scope of the domains BIM and UIM is ***instance-level linking*** which aims to relate corresponding information of the information models at instance level. Information integration through *linking* is a research topic in computer science since the late 1950s [15], also called *record linkage*. The applicability of the integration method *linking* for integrating information models from the domains BIM and UIM was originally demonstrated more than a decade ago [16,17] and afterward, several times confirmed for different integration scenarios [18–23].

In integration scenarios based on the integration method *linking*, corresponding objects of different information models, such as building and urban models, are identified and linked subsequently. However, the corresponding objects often do not **match** exactly due to the heterogeneity of the information models [13,19,24,25]. For example, in the domain BIM, a wall is generally represented as a solid structure and represents the actual physical object as designed. In contrast to that, in the domain UIM, the wall of a building may be represented only by its exterior wall surface and refers to the visually perceptible wall object that spans multiple floors. Consequently, the wall objects are related to some degree but do not match exactly.

Although the heterogeneity between information models of the domains BIM and UIM is analyzed in detail, its **consequences** for the linking process have not been investigated yet. Investigating these kinds of consequences goes hand in hand with answering the following questions:

> “*Does* the mismatch of corresponding objects cause problems for linking heterogeneous information models from the domains BIM and UIM? If yes, *which* problems does it cause, *when* does it cause these problems and *how* can these problems be overcome?”

These questions have led to this dissertation about the development of design principles to consider context-sensitivity for instance-level linking of heterogeneous information models of the domains BIM and UIM using the example of equivalence links.

1.2 Research method

This dissertation follows the research method **Design Science Research (DSR).** In contrast to natural science, research works following DSR do not aim to describe or explain some phenomena in the world but to acquire and communicate new research knowledge through the creation of an artifact [26,27]. Artifacts are artificially created products like constructs, models, design principles, or methods. This chapter describes the DSR process from a general perspective, while the next chapter on the structure of this dissertation describes how this dissertation is related to the described DSR process. This chapter does not aim to provide an extensive literature review about DSR but describes DSR as it is understood in this dissertation taking relevant research publications about DSR into account.

In the research literature on DSR, there are several procedural models representing the operational steps of the **DSR process** [26–28]. The following description is based on a composition of the procedural model from Peffers et al. [27] and Vaishnavi and Kuechler [28] (Figure 1.1). Here, the operational steps are called *Problem identification, Objective Definition, Design & Development, Demonstration & Evaluation*, and *Conclusion* as described in Table 1.1.

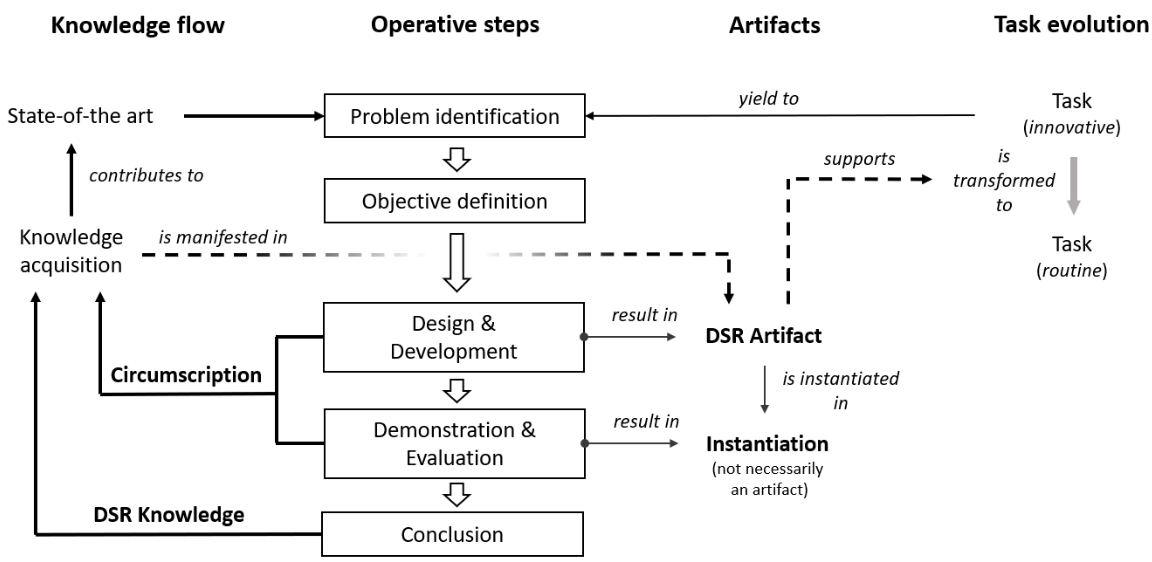


Figure 1.1: Procedural model of the DSR process following Peffers et al. [27] and Vaishnavi and Kuechler [28].

The operative steps in the described DSR process are consecutive, but the entire process is **iterative**. For example, new insights about the problem during the step *Design & Development* may lead to a reformulation of the research problem which initiates a new iteration process starting with the step *Problem identification*. To get to the point, this iterative character of the DSR process means that "the problem [of a DSR process] is not understood until after the formulation of a solution" [29] what emphasize the difficulties in the DSR process. Notably, this citation from Conklin et al. is a partial specification of wicked problems [29,30]. Here, the research problems in DSR are *not* called *wicked* since some common partial specifications of wicked problems do not fit to research problems in DSR. For example, the solution finding to a DSR problem is not a “one shot operation”.

Table 1.1: Description of operative steps in the DSR process.

DSR Research Step	Description
Problem identification	The research problem is identified and defined. Additionally, the value of the research is ensured.
Objective definition	The objectives of a solution are described. The objectives can be qualitative or quantitative measurements.
Design & Development	The artifact is created based on newly acquired knowledge
Demonstration & Evaluation	The artifact is instantiated, and its use is demonstrated. The result of the demonstration is evaluated against the previously defined objectives.
Conclusion	The question how the artifact support respective tasks and its consequences are discussed

The DSR process has two major **outputs** and one secondary output:

- First, the *knowledge* which is acquired throughout the DSR process and contributes to the state of the art. Here, one might further differentiate between *Circumscription* and *DSR knowledge* [28]. C*ircumscription knowledge* is acquired through the act of creating the DSR artifact and influences the understanding of the research problem. *DSR knowledge* is created through analyzing the utilization of the artifact.
- Second, the created *DSR artifact* in which the acquired knowledge is manifested. The DSR artifact manifests the knowledge acquired through the DSR process and is intended to support the transformation of an innovative to a routine task [26,28]. Here, *innovative* means that the successful accomplishment of the task requires the acquisition of new knowledge while the opposite is called *routine*.
- Third (secondary output), the *instantiation* which is used for the demonstration and evaluation of the DSR artifact regarding the defined objects. An instantiation is the realization of the DSR artifact in an environment [31]. Or using the words from Vaishnavi et al. [28], an instantiation is the situated implementation of the artifact. This kind of implementation may be an artifact itself and is essential for knowledge acquisition.

The DSR process is initiated through the identification of the ***research problem***. The research problem is composed of the *task problem* and the respective *research gap*. The *task problem* refers to the lack of knowledge required for the successful conduction of a task. The *research gap* refers to the comparison of this lack of knowledge to the existing research knowledge. There are two ways to derive the research problem and initiate the DSR process, namely *task-initiated* and *knowledge-initiated DSR* (Figure 1.2).

- *Task-initiated DSR* is the most common approach and is triggered by a task problem encountered while performing an innovative task. The lack of knowledge required for overcoming the task problem is compared to the existing research knowledge to identify the respective research gap.
- *Knowledge-initiated DSR* starts with the identification of the research gap through the rational deduction from existing research knowledge. Based on the identified research gap, the task problem and the corresponding innovative task specification are then derived.

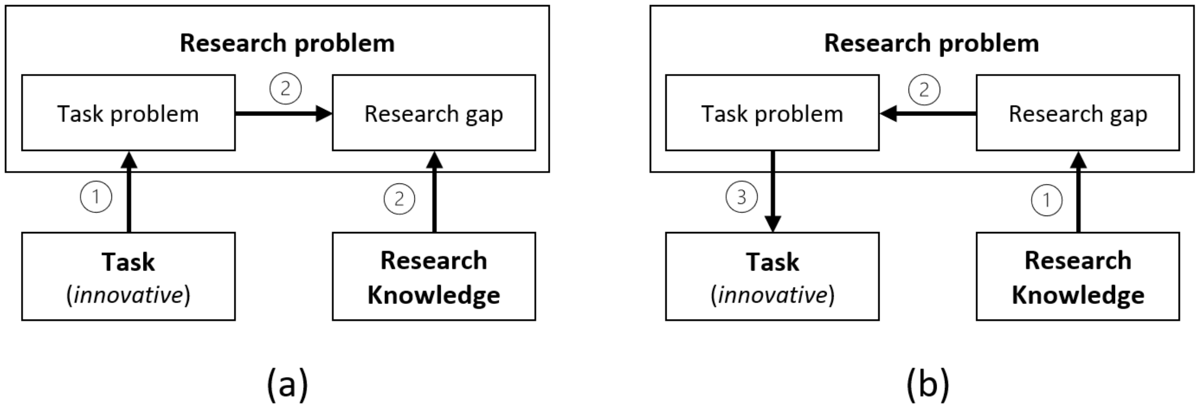


Figure 1.2: Structural components of the DSR step Problem Identification and their sequence in the scope of (a) task-initiated DSR and (b) knowledge-initiated DSR.

In ***knowledge-initiated DSR***, the task problem is not empirically identified through some real-world task specification, but rationally deduced based on the research gap. Thus, the nature of the task problem in knowledge-initiated DSR is rather hypothetical unless it is empirically confirmed that the rationally deduced task problem is a matter of a real-world task. Among others, this kind of confirmation can be achieved through an additional operative step or within the step *Demonstration & Evaluation*. Following the latter option, the artifact is created based on the assumption that the task problem is a matter of some real-world task specification (Figure 1.3).

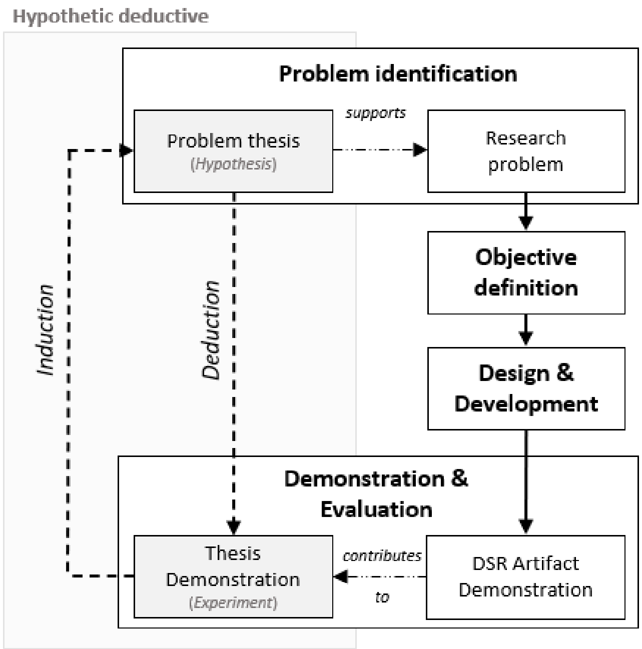


Figure 1.3: Procedural model representing the knowledge-initiated DSR process and illustrating the hypothetical character of the task problem.

1.3 Thesis structure

The **structure** of this dissertation follows the knowledge-initiated DSR process (Figure 1.4). *Knowledge-initiated* means that the research problem is derived through a literature review, which is why the chapter related to the step *Problem identification* follows the chapters about the state of the art. The state of the art is described through the chapters *Fundamentals* and *Related Literature* and also covers *circumscription knowledge* acquired through the artifact creation. The DSR steps *Problem identification* and *Object definition* are addressed by the chapter *Research problem & objectives*. The DSR artifacts are derived in the chapter five to nine (Design & Development) and subsequently in chapter *Design Principle* described in more detail and demonstrated (Demonstration & Evaluation). In the following, the subjects of the main chapters are briefly described and their contribution to the dissertation is explained.

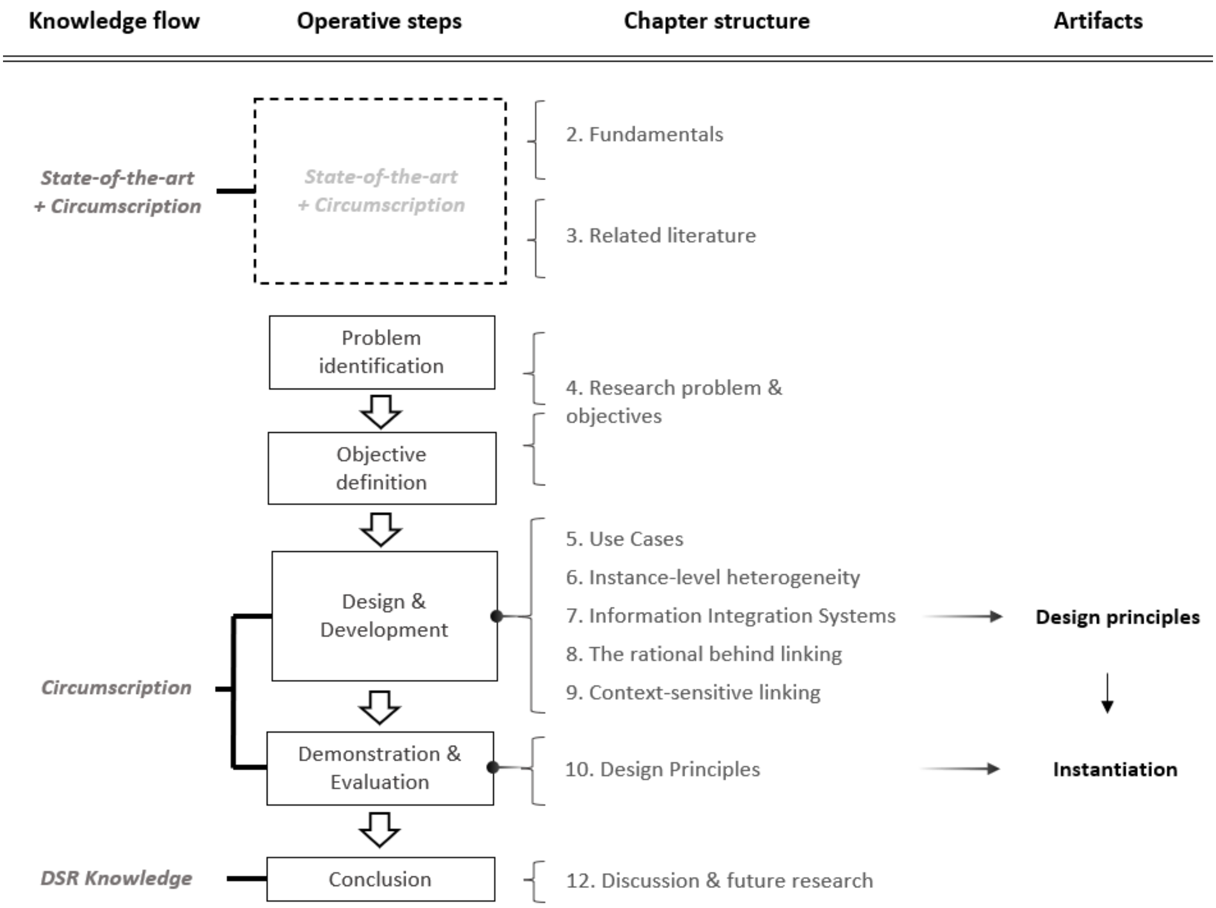


Figure 1.4: Structure of the dissertation and its relation to the knowledge-initiated DSR process.

In the chapter ***Fundamentals***, the basic understandings relevant for the integration of heterogeneous information models from the domains BIM and UIM are described. First, the terms ontology, information model, heterogeneity, and alignment are conceptualized. Afterward, formal languages and information models relevant for this dissertation are described, such as Industry Foundation Classes (IFC), Building Topology Ontology (BOT), and CityGML. Then, the domains Building Information Modeling, Geographic Information Systems (GIS), and Urban Information Modeling are discussed and compared to each other. Subsequently, the keyword “BIM-GIS Integration” is discussed through the comparison of the acronyms BIM, GIS and UIM. Last, Semantic Web and Linked Data are described. Here, the role of Semantic Web technologies within the domains BIM and GIS, and for the integration of these domains are described from the state-of-the-art view.

In the chapter ***Related Literature***, research literature related to the topics BIM-GIS Integration[1] and contextual linking in the field of Semantic Web is reviewed. The literature review is relevant for the deduction of the research problem which is why the chapter is a fundamental part of this dissertation. The reviewed literature about BIM-GIS Integration was categorized regarding several aspects like integration methods, use cases, and instance-level alignments. Furthermore, relevant approaches to contextual linking in the field of Semantic Web are analyzed and categorized.

The chapter ***Research problem & objectives*** refers to the DSR steps *Problem Identification* and *Objective definition*. In this chapter, the research problem is described which is composed of the research gap and related task problems. The research gap is deduced from the conducted literature review about both BIM-GIS Integration and contextual linking in the field of Semantic Web. The task problems related to this research gap are briefly described using the reference example. Furthermore, the relevance of the research problem is emphasized and the objectives against which the DSR artifacts will be evaluated are defined.

In the chapter ***Use Cases*** a new perspective on use cases demanding the integration of information models related to the domains BIM and GIS is introduced. Here, a use cases is further subdivided into the components *subject* and *activity*. This kind of new perspective on use cases is supported through an analysis of the reviewed literature on BIM-GIS Integration.

The chapter ***Instance-level heterogeneity*** introduces a new perspective on describing the differences between information models at instance-level for the scope of this dissertation. Here, common categorization approaches for heterogeneity are analyzed and discussed with respect to instance-level heterogeneities. The new perspective on instance-level heterogeneities is based on the concept of variants and versions, and the categorization of the differences between information models at instance-level with respect to their use in the information integration process.

In the chapter ***Information integration systems***, a system theoretical perspective on information integration is introduced what is further referenced by the term information integration systems. First, computer-based information integration systems are described from the perspective of system theory and defined with respect to this dissertation. Afterward, information integration is de-

[1] In the related literature, the keyword BIM-GIS Integration is rather common than the keyword BIM-UIM Integration which is why the literature research focus on BIM-GIS Integration. In chapter 2.1, both keywords are discussed in more detail.

scribed in terms of a computer-based information system composed of integration system, application system, and communication system. Furthermore, the implementation of information integration systems are discussed.

In the chapter ***The rational behind linking*** a new perspective on the semantics of links in the scope of BIM-GIS Integration is introduced. Here, the concept *correspondence* is defined as foundation of a link and categorized with respect to the related real-world objects. Afterward, the semantics of links in integrated information models are discussed based on model-theoretic semantics. In this dissertation, model-theoretic semantics are relevant for the evaluation of whether a link is misleading or not.

The subsequent chapter ***Context-sensitive linking*** refers to the DSR process step *Design & Development.* In this chapter, the concepts based on which the artifact is developed, and the consequent artifacts are described. The concepts are based on the system theoretical perspective on information integration and the semantics of links as described in the previous chapters. In more detail, context sensitivity in information integration systems is discussed and consequent understanding of contextual linking and context-dependent linking is described. The resulting artifacts are two design principles for the design of artifacts dealing with contextual linking and context-dependent linking respectively.

The chapter ***Design Principles*** refers to the DSR process steps *Demonstration & Evaluation* and *Conclusion* in relation to the respective design principle. In the beginning of the chapter, the implementation of the demonstration environment, namely the information integration system, is described. Afterward, the two design principles on contextual linking and context-dependent linking are implemented and demonstrated based on following steps: First, the design principles are described in more detail. Second, the instantiation of the design principles is described. Third, the instantiation is applied on respective examples. Fourth, the results are evaluated with respect to the objectives as defined in the chapter *Research problem & Objectives*. Last, the meaning of the evaluation results with respect to the related tasks is discussed.

In the chapter ***Discussion & Future Research****,* the findings of the thesis are summarized and subsequently discussed. Here, the findings are discussed with respect to the research question as described in the chapter *Motivation* (chapter 1.1). Furthermore, relevant directions for future research related to the investigated topic are emphasized.

1.4 Scope & assumptions

The dissertation is based on several scope limitations and assumptions to reduce the complexity of the topic under study and enable coherent work. The developed design principles dissertation are **limited** to ...

- ... information models from the domains BIM and UIM.
- ... linking of heterogeneous instance models (instance-level heterogeneity)

- ... equivalence correspondences between objects of these instance models.
- ... physical objects (such as walls and beams) as corresponding objects
- ... physical properties of these building elements as query subjects (such as material, length, fire resistance).

Furthermore, the investigated requirements on the alignment creation do not cover syntactical and structural requirements like requirements on the alignment language or its syntax. Apart from these limitations, the dissertation is based on the following **assumptions**:

- A computer-based function of an application has expectations on the information necessary for its purposeful execution.
- Information meeting these expectations are considered as ‘true’ in the scope of a function of an application and in the sense of truth-conditional semantics.

Noteworthy, information is considered as ‘true’ in the scope of the function but not as ‘true’ in an absolute sense. Here, the true value expresses whether the expectations on the information are met.

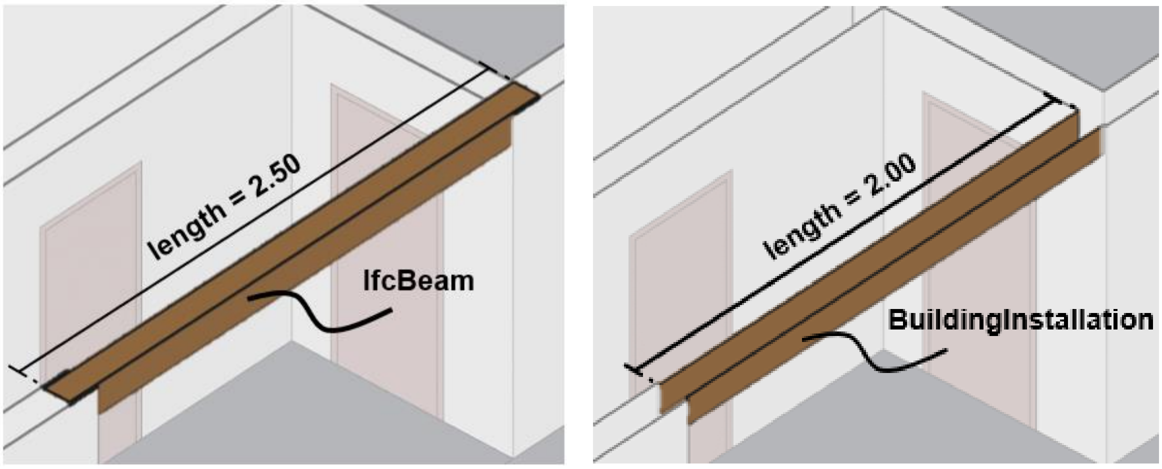
1.5 Reference example

Throughout this dissertation, the developed concepts are illustrated using the following **reference example**. The reference example is kept simple for three reasons: First, for explanatory reasons. Second, to facilitate the transfer to other related scenarios. Third, to show that the developed concepts are already relevant for non-complex scenarios.

In the reference example, **two information models** (A and B) shall be linked.

- Information model A was modeled by an architect for design purposes and represents a building from a prescriptive view. Thus, the information model A also represents geometrical data that is not visually perceivable from the outside, like the joint of walls.
- Information model B was generated for documentation purposes using photogrammetric methods and represents the same building as information model A but from a descriptive view. Thus, the information model B only represents geometrical data which is visually perceivable.

Both information models A and B represent the same real-world object: a constructive **beam** relating two walls (Figure 1.5). Information model A represents the beam including the support length (further called *true length*), while information model B represents the beam without support length (further called *visible length*). Thus, both information models represent beam objects describing the same real-world object but from different perspectives. The beam object of information model A is further referenced by the concept *IfcBeam* while the beam of information model B is further referenced by the concept *BuildingInstallation*. The concepts *IfcBeam* and *BuildingInstallation* are related to the information models IFC and CityGML respectively.



(a) (b)

Figure 1.5: Information models of the reference example representing (a) IfcBeam representing the length including support length and (b) Building Installation representing the length without support length.

2 Fundamentals

2.1 Ontologies and information models

In the context of computer-based information systems, information models are generally considered as a specific type of ***ontology*** or *knowledge representation*. The terms *ontology* and *knowledge representation* are often used synonymously but are not uniformly defined. In the following the concepts are briefly discussed and defined for the scope of this dissertation.

- *Knowledge representation:* Literally taken, knowledge representations refer to the representation of *knowledge*. From the perspective of epistemology, a requirement of knowledge is the *truthfulness* of the made statements. Thus, the representation of knowledge requires the evaluation of the truthfulness of the represented statements. The evaluation of the truthfulness of a statement requires respective formal structures such as description logics. [2]
- *Ontology*: The term *ontology* originally refers to the subject of general metaphysics, which is a subarea of philosophy. General metaphysics is concerned with the understanding of reality and being through understanding things, their properties and their relation to each other. Similarly, the term *ontology* means literally translated from Greek *doctrine* (Greek: *logos*) *of being* (Greek: *on*).
- *Information model:* The term information model refers to the representation of reality limited to the use of specific formal languages, which is further described below.

Conclusively, both information models and knowledge representations are types of reality representations which is, in wider sense, the subject of ontology (also called metaphysics). In computer science, the term ontology is often used synonymously to formal representations of reality. In this dissertation, the interpretation about ontology as in computer science is picked up due to practical reasons and is further described below.

Furthermore, there are different interpretation of an ontology in terms of a machine-readable reality representation depending on the **purpose of the ontology**. This circumstance is illustrated through independent trends in research areas referring to ontologies in terms of computer accessible reality representations. These trends have originated in subjects like *Philosophy*, *Electronic Data Processing* (EDP), *Natural languages,* and *Mathematics*, while the trends of *Philosophy* and *EDP* are exemplarily described below.

[2] Notably, other interpretations on knowledge representations may be less strict to the actual meaning of knowledge in epistemology. For example, Ackoff [32] writes that knowledge refers to *how-to* questions which is why knowledge differs from information.

- In philosophy, formal languages were introduced to describe the reality since the use of natural language is accompanied by several drawbacks such as the ambiguity of symbols [33,34]. The increasing processing power of computers has facilitated the use of these formal languages and allows the exploitation of the potential of these ontologies such as the reasoning of implicit information.
- EDP has emerged with the advent of computers. EDP focuses on the storage and processing of data and has continuously developed further methods for representing data adequate for respective tasks.

In a nutshell, the purpose of an ontology from the perspective of philosophy is to represent reality as it is, while the purpose of an ontology from the perspective of EDP is the representation of information adequate for a specific task.

Consequently, a common **definition of the term ontology** is challenging due its ambiguous use. An exemplary definition of the term ontology is provided by Euzenat and Shvaiko who state that an “ontology typically provides a vocabulary describing a domain of interest and a specification of the meaning of terms in that vocabulary” [35]. The most often cited definition of the term ontology in the context of computer science is from Gruber, who states that an ontology is “an explicit specification of a conceptualization” [36]. Notably, this definition is often cited without further explaining the meaning of *specification* or *conceptualization*. An approach explaining the components of the definition from Gruber is provided by Uschold and Gruninger [10]:

- A *conceptualization* is an abstract model of an original object (i.e., real-world object) described by concepts and relationships.
- An *explicit specification* of a conceptualization means that the concepts and relationships in the abstract model are given explicit names and definitions.

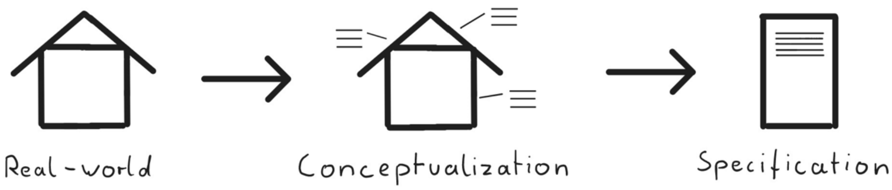


Figure 2.1: Simplified illustration of the concept ontology adopting the definition from Gruber [36].

Thus, an ontology is based on an abstract ***model*** that is, following Stachowiak [37], characterized by three features:

- First, it is based on an original (*Mapping feature*).
- Second, it reflects only a relevant selection of an original’s properties *(Reduction feature)*.
- Third, it needs to be usable in place of an original with respect to some purpose *(Pragmatic feature)*.

In an ontology, the abstract model is described through concepts and relationships. These concepts and relationships are expressed through explicit names (in terms of symbols) and their definitions. The relation between concepts and symbols can be illustrated through the semiotic triangle which is further discussed in chapter 7.2.4. The described understanding of an ontology is simplified illustrated in Figure 2.1. In addition to the definition from Gruber, some definitions state that an ontology is a *shared* conceptualization to emphasize its use by different persons [10,38]. Furthermore, some definitions include the term *formal* and state that an ontology is a *formal* explicit specification of a conceptualization [10,38].

Similar to the terms *conceptualization* and *explicit specification*, the term *formal* is generally not further explained when defining the term ontology. The term ***formal*** is not well-defined with respect to ontologies in the scope of computer-based information systems. Here, ontologies are formal in the sense that the meaning of terms used to specify the information is constrained [10,35]. There are different approaches aiming to constrain the meaning of terms like modeling languages like Unified Modeling Language (UML) [39] or logical language like description logics. The term *formal* with respect to ontologies is often used with different interpretations which are discussed below.

- *Machine-readable*: In the context of ontologies, the *formal* is often used synonymously to machine readable. In this dissertation, making information formal does not mean making information machine-readable, since information can be represented in formal structures without being machine-readable (such as representation based on visual modeling languages). Instead, making information machine-readable generally utilize formal structures for representing information.
- *Formal ontologies*: Ontologies based on formal semantics are often called *formal ontologies*. In this dissertation, ontologies based on formal semantics are covered the term knowledge based representation.

Among others, ontologies can be distinguished by their **degree of formality** (Figure 2.2). The degree of formality of an ontology refers to the expressivity of that ontology to constrain the meaning of the terms specifying the information. The illustration of Figure 2.2 is often cited with some variations regarding the naming of the ontologies, or the term *expressivity* is used instead of *degree of formality* [10,35,40]. Examples of ontologies based on a low degree of formality are taxonomies or dictionaries. Some authors further distinguish between lightweight and heavyweight ontologies [40–42]. However, these terms are not commonly defined, and their discussion is not considered as relevant for further discourse. In this dissertation, the major focus is on *information models* that refer to a higher degree of formality than thesauri or database schema, but do not belong to knowledge representations.

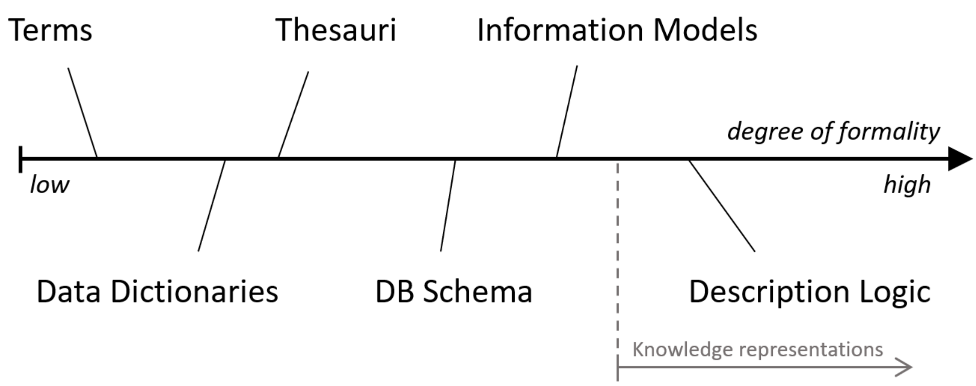


Figure 2.2: Exemplary types of ontologies distinguished by their degree of formality. The figure is adapted from Uschold and Gruninger [10].

This dissertation focuses on *information models* following the **Meta-Object Facility** (MOF) [43]. The MOF was introduced by the Object Management Group (OMG) [44] and distinguishes between four layers of an information model, namely M0, M1, M2, and M3 (Figure 2.3). The M0 layer represents concrete instances of real-world objects and is also called *instance level*. The M1 layer represents instantiated objects following a formal language like Unified Modeling Language (UML) [39] and is also called *schema level*. The M2 layer represents the structure of formal languages like UML and the M3 layer is one abstraction level above the M2 layer. The M2 and M3 layers are not further addressed in this dissertation as these kinds of abstraction levels are of minor importance for linking information models at instance level. Information Models instantiated at instance level or schema level are called instance or schema model respectively.

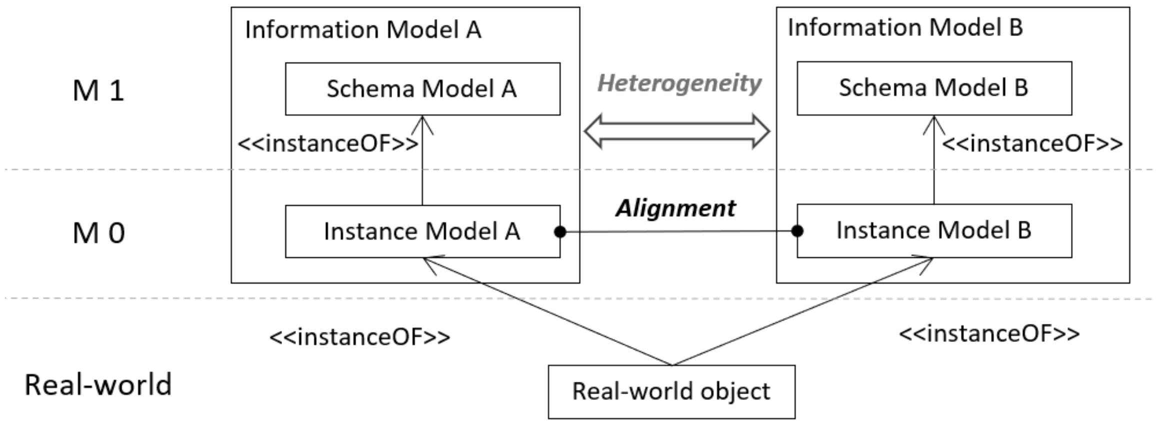


Figure 2.3: Structure of information models following the Meta-Object Facility (MOF) and their relation through instance-level alignments [3].

Schema models are textually or graphically **represented** through a modeling language (Figure 2.4). Common modeling languages are Unified Modeling Language (UML) [39], Standard for Exchange of Product data Part 11 (STEP P11) which is also known as EXPRESS [45], Resource Description Framework (RDF) [46], and Web Ontology Language (OWL) [47]. Notably, OWL is developed to represent ontologies in terms of knowledge representations rather than information models. Differences and commonalities of OWL and UML are described by Zedlitz [48] Furthermore, instance models are made computer-accessible through a syntactical language for file-based serialization. Common syntactical languages are Extensible Markup Language (XML) [49], Geographic Markup Language (GML) [50], Standard for Exchange of Product data Part 21 (STEP P21) also called SPF [51], or Turtle [52].

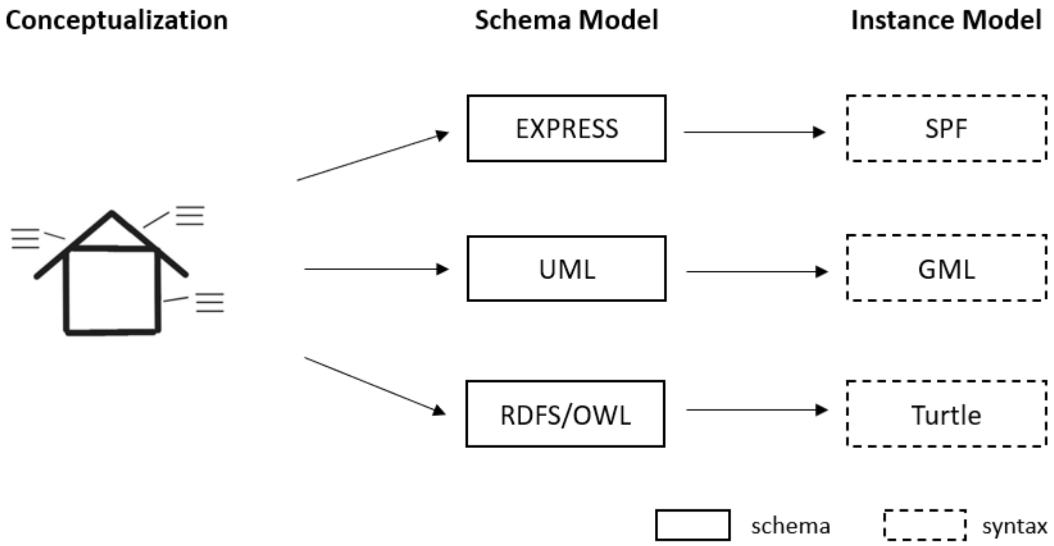


Figure 2.4: Representing a conceptualization through schema and instance models based on different modeling languages and syntaxes. The figure is adapted from Pauwels et al. [53].

Information models can be **further categorized** in *internal* vs. *external* models and *proprietary* vs. *vendor neutral* models (Figure 2.5). Internal models are directly embedded within a software product and are called proprietary when their usage is limited to software products of specific vendors. The issue of the dependency on the vendors in the Architecture Engineering and Construction (AEC) industry and its consequences for software interoperability is illustrated by a common metaphor by Hannus et al. [54] who describe the landscape of data and software products as islands of automation. The development and standardization of vendor-neutral external models aim to reduce the dependency on specific vendors.

A software product is composed of both application and information models. The close relation between application and information model, as described in chapter 7.1, is accompanied by the circumstance that the terms **integration and interoperability** are often used synonymously. For example, some authors in the field of BIM-GIS Integration refer to the interoperability of information models [11], even though information models cannot *operate* on information. In general, interoperability in the scope of information modeling means the ability of software products to exchange information and to process this exchanged information [55] and is more precisely referenced by the term *software product interoperability*. In more detail, software product interoperability is required when a function of a software product demands information from the internal information model of another software product. The transfer of the information from one internal to the other internal model is generally achieved through *information integration*.

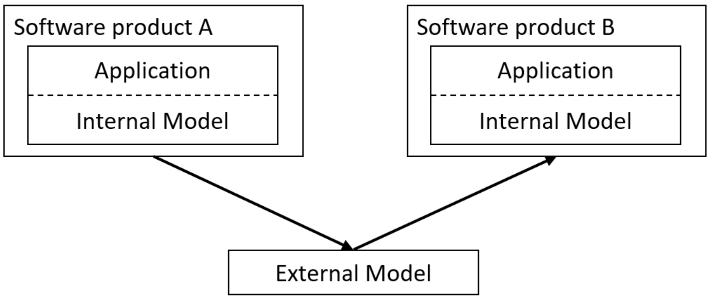


Figure 2.5: Information models as internal models of a software product and as external model used for software product interoperability.

Common vendor-neutral information models in the field of 'BIM-GIS Integration' are Industry Foundation Classes (IFC) [56] and CityGML [57,58]. IFC is generally associated with the domain BIM and aims to support the information exchange between proprietary software products in the whole life cycle of a building project in terms of an external information model. IFC is generally represented through EXPRESS and SPF. CityGML is generally associated with the domain GIS or UIM and is an application schema of the Geographic Markup Language (GML) [50] which aims to support the modeling, storage, and exchange of city models. This dissertation focuses on IFC and CityGML and alternative representations based on RDFS/OWL and Turtle. RDF-based alternative representations to IFC are ifcOWL [59,60] and Building Topology Ontology (BOT) [42]. Noteworthy, GML is technologically originated in RDF and therefore closely related. In this thesis, BOT is used to represent the IFC model through an RDF graph due to its lower complexity compared to ifcOWL and which is sufficient for this thesis.

2.2 Formal languages

2.2.1 UML

The Unified Modeling Language (UML) is a collection of formal notations to represent object-oriented models. The UML is standardized through ISO/IEC 19505 [61], and developed by the Object Management Group (OMG) [44]. In general, there are two major types of UML diagrams: first, diagrams which represent the structure of the model such as class diagrams. Second, diagrams which represent the process such as activity diagrams. In this dissertation, solely class diagrams are relevant for representing the built environment. In Figure 2.6, an abstract UML class diagram is represented illustrating the core components, class, attribute, inheritance relation and association relation.

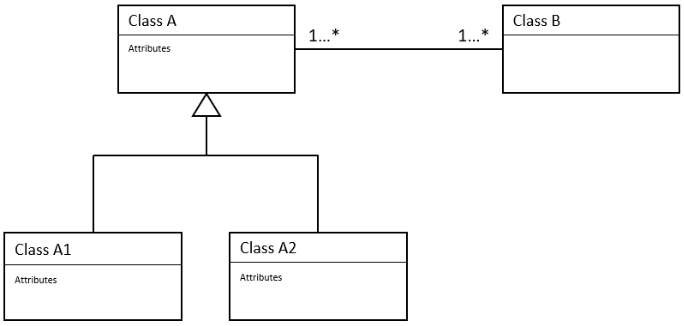


Figure 2.6: An exemplary abstract UML class diagram representing the components class, attributes, association, and specialization.

2.2.2 XML/ XSD

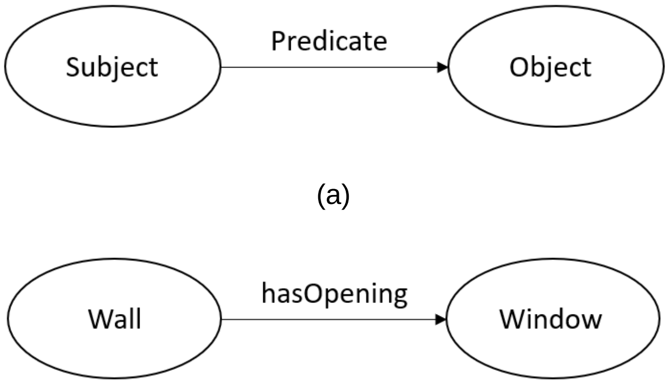
The Extensible Markup Language (XML) is a syntactical language and standardized through the World Wide Web Consortium (W3C) [62]. An XML document is hierarchically structured through elements and child elements. In general, an element is specified by a start tag and an end tag, enclosing its contents. Both start and end tag are marked by brackets, while the end tag is signed by a backslash, such as <wall> and </wall> respectively. Basically, the data describing an element can be expressed through its naming (like an element representing a wall, <wall>), added as attribute (like an id added as attribute to the element, <wall id=123>), or through its content (like the height of a wall <height>2.50</height>). The XML-Schema (XSD) is intended to define the schema of an information model. While in UML the class structure is represented visually, in XSD the class structure is represented textually through an XML document.

2.2.3 SPF/ EXPRESS

EXPRESS is a data modeling language to represent object-oriented data models and is standardized through STEP-Standard Part 11 (ISO 10303-11) [45]. EXPRESS is a textual language, and the corresponding visual notation is EXPRESS-G. Among others, EXPRESS allows to define entity types, their attributes, and relationships and inheritance to other entity types. An entity type can be considered as equivalent to classes in object-oriented modeling. In addition to other data modeling languages, EXPRESS allows to define inverse relationships. The syntactical language to create a STEP Physical File (SPF) is standardized in STEP-Standard Part 21 (ISO 10303-21) [63]. A SPF file covers a header and a data section. In the data section, an object covers a number which is unique in the file and the name of the instantiated entity, such as “#123=BUILDING(…)”. The relations to other objects within the file can be represented through referencing respective unique numbers.

2.2.4 RDF(S)/ OWL

Roughly spoken, the Resource Description Framework (RDF) [64] refers to two aspects, namely structure, and vocabulary. The RDF structure follows the concept of triples in terms of *subject – predicate – object,* which can be represented through a *node – link – node* structure (Figure 2.7). As an example, the statement *wall has a window* could be represented through the triple *Wall – hasOpening – Window*. Combining several triples result in a graph structure, the so-called *RDF graph*.



(a)

(b)

Figure 2.7: (a) The subject, predicate object and (b) the statement wall has a window visually represented as RDF graph structure.

The subject and predicate refer to *resources* while the object refers to either *resource* or *literal*. A literal is a value like a number or a date. A *resource* is a unique concept and referenced by a Unique Resource Identifier (URI). Notably, URIs are often based on URLs to emphasize the meaning of the resource but URIs do not have to be valid URLs or follow the structure of URLs. URIs can be abbreviated through so-called *namespaces*. RDF provides some vocabularies and refers to a namespace with resources such as *rdf:type* or *rdf:about*. In Figure 2.8, the statement that *a specific wall has a specific window* is represented using exemplary prefixes.

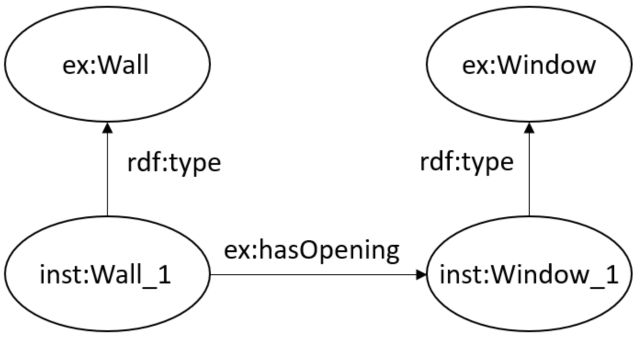


Figure 2.8: RDF graph representing the statement that a specific wall has a specific window. The inst:Wall_1 and inst:Window_1 are instances of the classes ex:Wall and ex:Window and related through ex:hasOpening. The prefixes inst and ex are abbreviations for exemplary URIs.

An RDF graph can be queried using the query language SPARQL [65]. An RDF graph can be considered as a data model for itself and be expressed through different **syntactical representations** like RDF/XML, or Turtle [52]. The syntactical components, the structure of an RDF document and a SPARQL query are described in more detail in *Example 1*.

RDF Schema [66] and Web Ontology Language are W3C specifications and allow to **extend the RDF vocabularies**. The RDF Schema (RDFS) extends the vocabulary provided by RDF through vocabulary defining the schema such as *rdfs:class* and *rdfs:subClassOf*. An instance of a class is related to its underlying class through the predicate *rdf:type*.

The **Web Ontology Language** (OWL) [47] extends the vocabulary of RDF(S) through concepts based on first predicate logics to support reasoning mechanisms. Similarly, there are different subsets of OWL referring to different expressivities based on predicate logics, such as OWL DL and OWL Lite. Among others, a key difference of OWL, compared to other data modeling languages such as UML or EXPRESS, is that OWL allows to define both necessary and sufficient requirements. For example, given that a model covers the class *wall* and its subclass *exterior wall*. OWL allows to define necessary requirements such as “a wall must have a structure” but also sufficient requirements such as “if an object has structure XY, then it is an exterior wall”. The differences between UML and OWL are analyzed in more detail by Zedlitz [48].

RDFS and OWL refer to **model-theoretic semantics** to constrain the meaning of the used triples. Model-theoretic semantics is a subject of extensional semantics and is relevant for reasoning. Roughly spoken, model-theoretic semantics allow to relate the used vocabulary to their actual interpretations and, therefore, evaluate whether an RDF statement is ‘true’ in a specific universe of discourse. In chapter 8.1, model-theoretic semantics are discussed in more detail.

Among others, the high expressivity of OWL allows to relate resources which is why OWL is often used to align different ontologies. OWL covers two major types of link predicates relating entities at instance level (also called objects or individuals), namely *owl:sameAs* and *owl:differentFrom.* The link predicate *owl:sameAs* states that the linked objects are identical which is why such kind of link is also called *identity link.* Respectively, the link predicate *owl:differentFrom* states that the linked objects are not identical. In *Example 2*, the RDF triples as defined in *Example 1* are related through the link predicate *owl:sameAs* and queried using SPARQL subsequently.

A set of links in terms of an alignment is made explicit and formal through some formal language [35], here called alignment language. An **alignment language** covers vocabulary through which a link is expressed, also called link predicate or relation type, and constrains the meaning of this vocabulary. An alignment language is compatible with specific ontology languages and made computer-accessible through a syntactical language. In the field of Semantic Web, examples for alignment languages are Web Ontology Language (OWL), Simple Knowledge Organization System (SKOS) [67] and Similarity Ontology [68]. Further alignment languages are discussed in the scope of the literature review about contextual linking in the field of Semantic Web (chapter 3.3).

2.3 Information models

2.3.1 IFC

The **Industry Foundation Classes** (IFC) is an information model intended to digitally represent the built environment with the aim to support the vendor-neutral information exchange of these digital models between software products. IFC is standardized through the international standard ISO 16739-1:2018 [56]. The structure of the IFC is based on four layers:

- The *core layer* covers the fundamental classes which can be used and concretized in other layers.
- The *shared layer* is located at the interface between core layer and domain specific schemas and defines classes like *IfcWall* or *IfcBeam*.
- The *domain layer* covers classes which refer to a specific domain.
- The *resource layer* provides the relevant data structures

IFC allows to represent both semantics and geometries of the elements. The geometry of the elements can be described by either explicit or implicit representation methods. The geometries are referenced to relative coordinates systems defined by the related objects.

In general, the **IFC schema** is represented through EXPRESS or EXPRESS-G. Alternatively, the IFC schema can be represented through other schema languages such as XSD or OWL. The IFC schema translated to OWL is called ifcOWL. In Figure 2.9, the IfcBeam class is represented through EXPRESS.

```
ENTITY IfcBeam
 SUPERTYPE OF (ONEOF
        (IfcBeamStandardCase))
 SUBTYPE OF (IfcBuildingElement);
        PredefinedType : OPTIONAL IfcBeamTypeEnum;
 WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcBeamTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcBeamTypeEnum.USERDEFINED) AND EXISTS
                (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
  ('IFC4.IFCBEAMTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;
```

Figure 2.9: EXPRESS schema of the class IfcBeam [69] which specify the sub and super classes through SUBTYPE and SUPERTYPE respectively, and necessary requirements through the WHERE clause.

The meaning of the classes is additionally specified through a textual description. In the following, the textual description of IfcWall, IfcBeam and IfcStair is represented.

- *IfcWall* – “The wall represents a vertical construction that bounds or subdivides spaces. Wall are usually vertical, or nearly vertical, planar elements, often designed to bear structural loads. A wall is however not required to be load bearing.”

- *IfcBeam* – “An IfcBeam is a horizontal, or nearly horizontal, structural member that is capable of withstanding load primarily by resisting bending. It represents such a member from an architectural point of view. It is not required to be load bearing“.
- *IfcStair* – “A stair is a vertical passageway allowing occupants to walk (step) from one floor level to another floor level at a different elevation. It may include a landing as an intermediate floor slab.”

The IFC instance model can be encoded through different syntactical languages such as SPF, XML or RDF/XML. In IFC, the objects are identified through a Globally Unique Identifier (GUID).

2.3.2 BOT

The Building Topology Ontology (BOT) aims to digitally represent buildings in a less complex manner than IFC and is intended to serve as basis for domain specific ontologies following the W3C principles [42,70]. The BOT is based on RDF and focuses on the representation of the topological structure of the Buildings with minimal complexity. The main three classes of BOT are *bot:Zone, bot:Element* and *bot:Interface.* A zone can contain, be adjacent to or intersects with other zones or can have elements, and elements can have sub elements. In Figure 2.10, the use of these classes is illustrated with example of simple building structures.

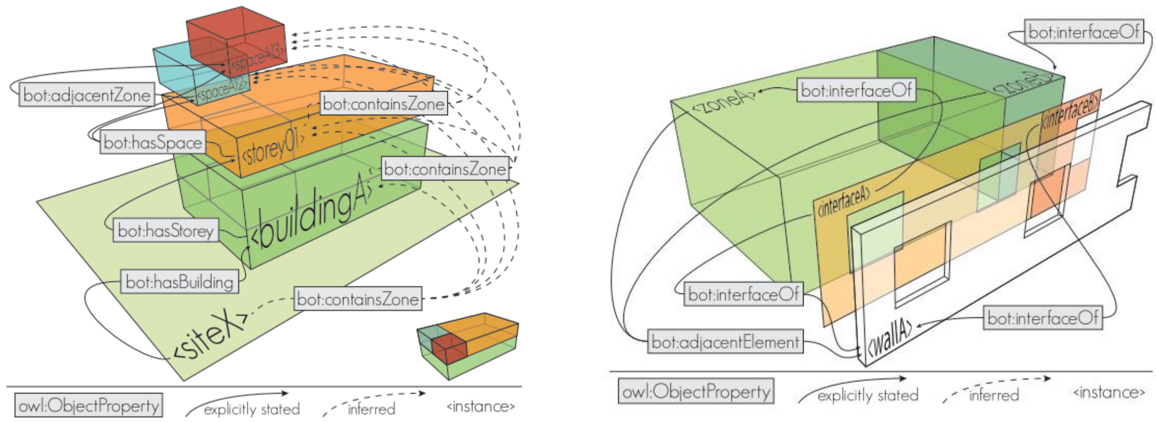
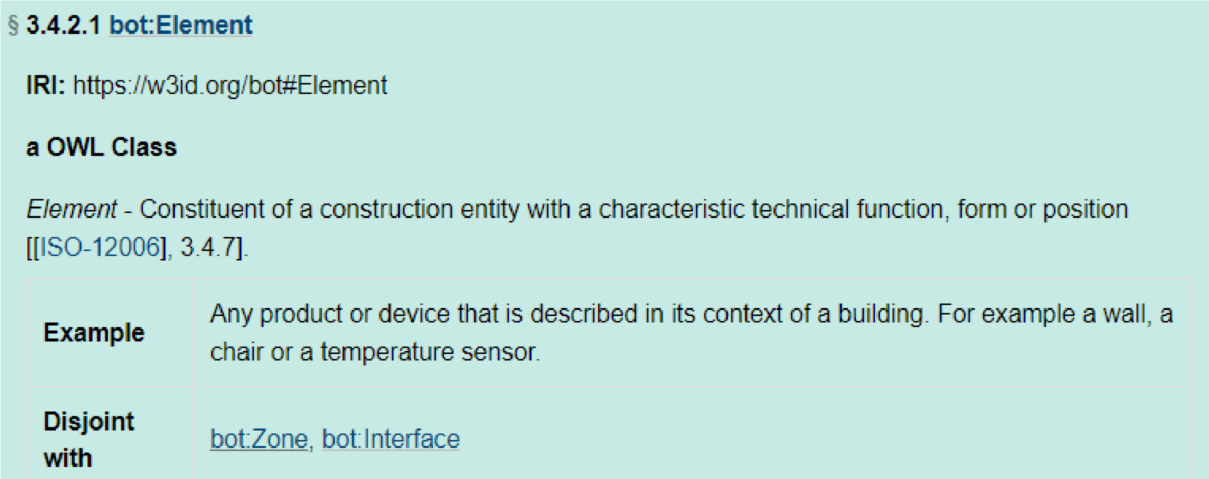


Figure 2.10: Representation of the structure of BOT with the example of (a) the relation of spaces to the site and (b) the relation of a wall through an interface to a zone [42].

The class *bot:elements* refers to physical elements of buildings such as walls, chair or sensors (Figure 2.11). BOT does not provide classes to specify the elements in more detail but explicitly suggest relating to other models with respective classes. For example, the Building Element Ontology (BEO) [71] provide classes based on the *IfcBuildingElement* subtree of IFC specification.



§ 3.4.2.1 bot:Element

IRI: https://w3id.org/bot#Element

a OWL Class

Element - Constituent of a construction entity with a characteristic technical function, form or position [[ISO-12006], 3.4.7].

Example	Any product or device that is described in its context of a building. For example a wall, a chair or a temperature sensor.
Disjoint with	bot:Zone, bot:Interface

Figure 2.11: The definition of the bot:element which includes formal and textual description specifying the meaning of bot:element [70].

2.3.3 CityGML

CityGML 3.0 is intended to digitally represent 3D city models for several use cases like a variety of analysis, visualization or the improvement of software product interoperability [57,58]. CityGML is an application schema of the Geography Markup Language (GML) [50] which is based on the ISO 191xx series. The GML is a vendor-neutral language for the representation of geographical data and standardized through ISO 19136:2007. Among others, GML supports several geometrical representations like points, lines, or polygons. GML is based on both XSD and XML and can be extended through application schemata such as CityGML.

CityGML has a core schema which is enriched through domain specific UML packages. The domain-specific UML packages are Transportation, LandUse, WaterBody, Relief, CityObjectGroup, CityFurniture, Vegetation and Construction. The UML package Construction covers the domains Building, Bridge, and Tunnel. CityGML supports the semantical and geometrical representation of objects related to these modules. In this dissertation, the building module is of major relevance which is why it is describe here in more detail. In Figure 2.12, the UML class diagram describing the building module from the most recent version of CityGML (version 3.0) is represented. The building module covers classes representing physical building parts, classes representing logical elements like story and building units, and class representing constructive elements such as walls or beams.

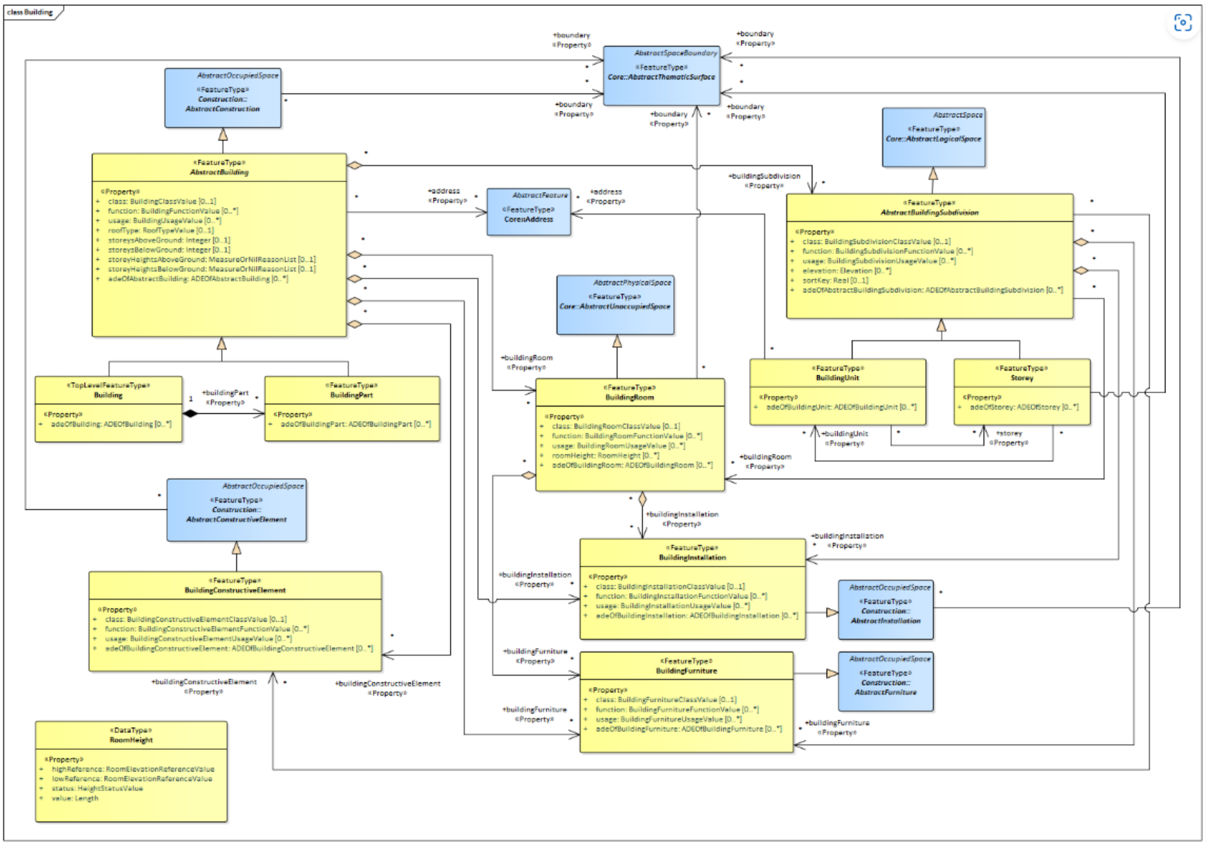


Figure 2.12: The UML class diagram describing the building module of CityGML [57].

In CityGML, four different Levels of Detail (LOD) are defined to support data sources and data requirements with different granularity. In Figure 2.13, the LODs are illustrated with the example of building. In LOD0, the footprint of the building is represented as surface. In LOD1, building is represented through a box based on the dimensions of the footprint and the height of the building. In LOD2, the roof is modelled in addition to the box representation. In LOD3, the openings and interior structures such as rooms and furniture are modelled in addition to the representation of LOD2.

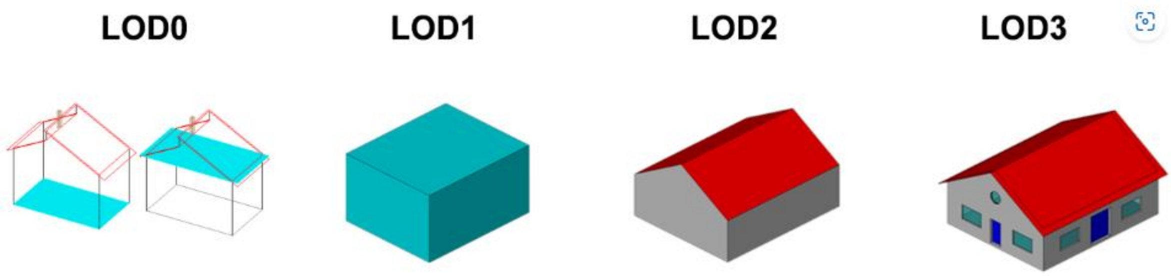


Figure 2.13: Illustration of the different Level of Details specified in CityGML with the example of a building [57].

2.4 Modeling the built environment

2.4.1 Building Information Modeling

The acronym **BIM** refers to the field of Computer-Aided Design (CAD) and denotes either a subject in terms of a *Building Information Model* or a method in terms of *Building Information Modeling*. It is true for both the subject and the method that there is no general definition. A *Building Information Model* is generally associated with a digital representation of the built environment in terms of a three-dimensional geometric model enriched with non-geometric information [72]. On the other hand, *Building Information Modeling* refers to a method for which there are two different courses of interpretation: First, a method in terms of the creation, modification, and maintenance of a Building Information Model [72]. Second, a method in terms of the usage of the Building Information Model over the whole life cycle of a building asset [72].

The productivity in construction industry has stagnated for years, unlike other nonfarm industries [73,74]. The usage of the Building Information Model over the whole life cycle of a building asset aims to increase the productivity of the construction industry through several aspects such as automation of time-intensive tasks, reduction of error-prone re-entering of information, or cost savings due to decisions about design changes in early project phases [72,75]. An obstacle to use the Building Information Model over the whole life cycle, is the heterogeneous landscape of vendor-specific software products used in a construction project. This kind of heterogeneous landscape hinders the software product interoperability what is metaphorically emphasized by though the notion “islands of automation” [54]. An approach to bridge these islands of automation is the use of the vendor-neutral information model IFC to exchange of information across heterogeneous software products.

2.4.2 Urban Information Modeling

Urban Information Modeling (UIM) aims to represent the urban environment, such as trees, terrains and buildings, through semantically enriched three-dimensional geometric models to support several use cases like solar potential analysis, building energy demand estimation, or disaster management in urban environments [4,76]. Like BIM, the acronym UIM denote either a subject in terms of Urban Information Model or a method in terms of Urban Information Modeling. Concepts used (partially) synonymously with UIM are City Information Modeling (CIM) [77], semantic 3D City Modeling [4], and Geospatial Information Modeling (GIM) [11]. The relevance of a digital representation is illustrated by Biljecki who emphasizes 100 use cases for the urban information models.

In general, Urban Information Modeling utilizes Geographic Information Systems (GIS). GIS is part of the geospatial field while the term *geospatial* refers to both geographic information like terrains and spatial information like digital models based on some geometrical representation. In general, the information systems belonging to GIS have in common that they represent georeferenced objects. As example, Bartelme [78] writes that GIS “serves for capturing, storing, analysis, and visualization of data that describe a part of the Earth’s surface, the technical and administrative entities, as well as findings of geoscience, economics, and ecological applications”. There are several terms that are closely related to GIS with corresponding examples being geomatics, geoinformatics, or geospatial engineering.

2.4.3 Comparison

In the field of BIM-GIS Integration, there are three different approaches to describing **differences between these domains**, namely *accumulation of differences*, *structured comparison*, and *comparison of specific entities*. The differences generally refer to either the domains BIM and GIS, or the information models IFC and CityGML.

- The first approach **accumulates differences** from a general perspective in an informal manner like using text or tables [5,11,12,14,79–84]. For example, IFC and CityGML models are often compared regarding the physical objects they aim to represent, their underlying modeling paradigm, the supported geometrical representation, utilized coordinate systems, or their detail levels (Table 2.1). Roughly spoken, IFC is intended to represent buildings including their physical objects from prescriptive view and supports both explicit and implicit geometrical representations in cartesian coordinate systems [4,85]. On the other side, CityGML is intended to represent cities including their buildings from descriptive view and support explicit geometrical representations in georeferenced coordinate reference systems [4,85]. A common comparison characteristic of BIM and GIS is the scale of the addressed physical objects: BIM generally deals with physical objects at a more granular scale than GIS. Noteworthy, the scale of the addressed physical objects overlaps so that both domains partly deal with similar objects represented from different perspectives (Figure 2.14).

Table 2.1: Comparison of IFC and CityGML following general approach with the example of Brüggemann and von Both [14].

	IFC (BIM)	**CityGML** (GIS)
Physical representation	Building structures, buildings, rooms, building components, materials, etc.	Topography, infrastructure, vegetation, buildings, rooms, etc.
Modeling paradigm	Prescriptive	Descriptive
Geometrical representation	Implicit or explicit	Explicit
Coordinate systems	Cartesian	Georeferenced
Detail levels	Not generally defined	5 Level of Details

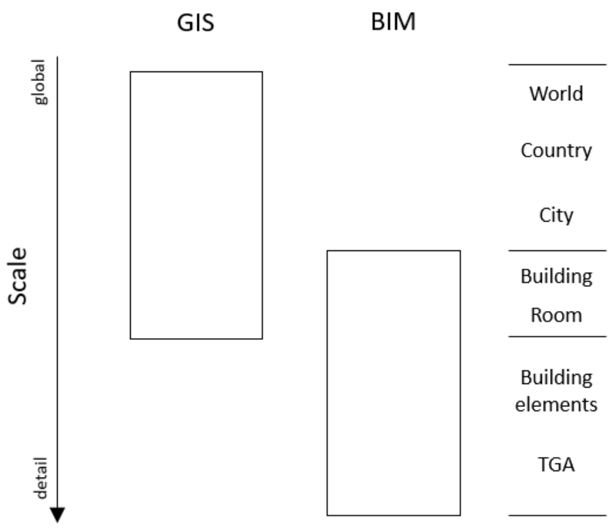


Figure 2.14: Different scales of the physical representations related to the domains BIM and GIS. Figure is adapted from Donaubauer et al. [86].

- The second approach is based on a **structured categorization** of the heterogeneities like the categorization into syntactical, structural, and semantical heterogeneities. Structural heterogeneity occurs when representing the same information using different structures, like representing information as a class versus as an attribute of a class. Semantical heterogeneity addresses the ambiguity of the used terms, like synonyms and homonyms. Other categorization approaches further differentiate between terminological, semiotic heterogeneities and differences in perspective [33,35]. Furthermore, syntactical heterogeneity refers to differences caused by the used syntactical/modeling language for the schema or instance model, like STEP Part 21 versus XML. Notably, some publications belonging to this approach do not explicitly refer to the heterogeneity of information models but to mismatches between the information models or interoperability levels [35]. In the reviewed literature there are solely four publications referring to the second approach [11,82,87,88], even though the second approach is often referenced in related research areas like database or ontology engineering.
- The third approach refers to the **comparison of specific entities** at instance or schema level of the information models [18,19,25,89–96]. This kind of comparison is either unformal and unstructured (like continuous text), unformal and structured (like tables) or formal and structured (like UML, Express-G). An often cited publication is from Nagel et al. [13] who visually illustrate common differences between IFC and CityGML models created from different perspectives, namely prescriptive and descriptive views respectively (Figure 2.15). For example, a beam modeled in the design phase is represented in IFC using their true dimensions (i.e., including support length) while beams created from photogrammetrically methods are represented in CityGML using the visible dimension (i.e., without support length). Furthermore, walls and slabs are represented by their outer surfaces in CityGML

which is why respective joints are not represented and a wall object in IFC generally refers to several wall surfaces in CityGML.

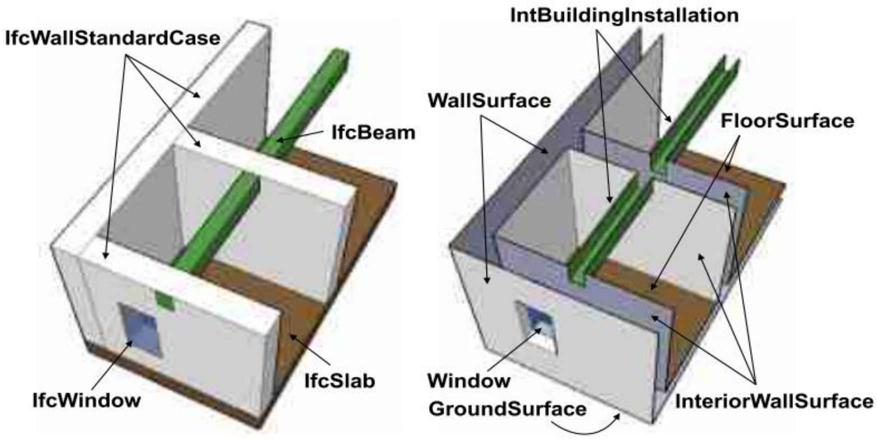


Figure 2.15: Common similarities and differences between IFC created from prescriptive (left figure) and CityGML created from descriptive view (right figure) [13].

2.5 The keyword „BIM-GIS Integration"

The keywords '***BIM-GIS Integration'*** or '*GeoBIM'* are commonly used when referring to information integration efforts related to information models from the domains BIM and GIS. However, the direct comparison of BIM and GIS is inappropriate since BIM generally refers to a method while GIS refers to information systems [4,85]. Thus, some authors use the keyword *BIM-UIM Integration* what overcomes this comparison issue but its subject is limited to urban areas [4,85]. In the following, the acronyms BIM, GIS, and UIM are described and the comparisons of *BIM versus GIS* and *BIM versus UIM* are subsequently discussed in more detail.

- The acronym **BIM** refers to the field of Computer-Aided Design (CAD) and denotes either a subject in terms of a *Building Information Model* or a method in terms of *Building Information Modeling*. Furthermore, the creation, modification, and maintenance of a Building Information Model [72]. Second, a method in terms of the usage of the Building Information Model over the whole life cycle of a building asset [72].
- The acronym **GIS** stands for Geographic Information System(s) and refers either to all information systems dealing with geospatial information (Geographic Information Systems) or to a single instantiation of such kind of information system (Geographic Information System).
- The acronym **UIM** refers to modeling of urban environments [85]. Like BIM, the acronym UIM can denote either a subject in terms of Urban Information Model or a method in terms of Urban Information Modeling.

In a nutshell, the method BIM describes the way how building information is used ("over the whole life cycle of a building asset") for a specific goal ("increase productivity in the construction industry"). In contrast to that, GIS rather refers to data structures, software tools, and processes dealing with geospatial information, but does neither describe the way how the information is used, nor the goal intended to be achieved. On the other hand, the methods **BIM and UIM** have in common that they utilize computer-based information systems (i.e. CAD and GIS) to reach some goal defined by a respective use case. The underlying computer-based information systems of BIM and UIM can be integrated to reach these goals what is further called *BIM-UIM Integration*. BIM, GIS and UIM are related to both applications and information, what is further referenced by the term *domain*. Thus, BIM-GIS Integration refers to the integration in terms of *domains* but not in terms of their actual interpretation as described previously.

For both ***BIM-GIS Integration*** and *BIM-UIM Integration* it is true that they do neither specify the integration subject nor the integration method. The integration subject can be at data, process, or application level (chapter 7.2.3). In general, the integration efforts referenced by *BIM-GIS Integration* address the integration at data level. Exemplary integration methods at data level are conversion or linking (chapter 7.2.2). Notably, the term *integration* generally refers to the way to reach a specific goal such as integrating internal information models to improve the interoperability of these software products.

2.6 Semantic Web and Linked Data

The **Semantic Web** is a design of the world wide web which aims to make the data in the internet machine readable and to connect the data in terms of a data web. The Semantic Web was coined by Tim-Berners Lee in 1999 by describing the vision of the future of the web [97]. The standards for the realization of the Semantic Web are defined by the World Wide Web consortium (W3C). These standards cover the specifications about languages such as RDF and OWL, which are generally referenced by the term Semantic Web technologies.

There are different interpretations of the term Semantic Web [98]. Similarly, the boundaries of the terms **Linked Data and Semantic Web** are often described blurry what makes a distinction difficult for persons new to the topic. As an example, the W3C writes on their webpage [99] that “The Semantic Web is a Web of Data” and further that the Web of Data “can also be referred to as Linked Data.” On the other hand, they also write on the same web page that “Linked Data lies at the heart of what Semantic Web is all about.” Thus, the terms Semantic Web and Linked Data are here considered to some degree synonymously since both *are* or *refer to* Web of Data, but also not synonymously since Linked Data lies at the heart of Semantic Web. Nevertheless, Semantic Web is generally considered as vision of a global Web of machine-readable data [98] .

In this dissertation, there is a **more granular distinction** between Linked Data and Semantic Web. Here, Linked Data refers to the interrelation of machine-readable data through the web and supports the querying of this kind of interrelated data. Linked Data do not assign any truth value to the linked data statements (i.e., RDF triples) but consider the data literally as linked data. On the other hand, Semantic Web makes use of Linked Data and enhances this data with extensional semantics and logics to provide meaning to the linked data. Thus, Semantic Web allows to evaluate the truthfulness of linked data statements in the sense of knowledge representations as defined above. Linked data is a requirement for the Semantic Web. The Semantic Web in terms of a vision refers to the idea of a global Semantic Web.

In 2006, Berners Lee has published the five star schema defining **different levels** to support Linked Data [100] (Figure 2.16). The five star scheme is a guideline to "make it progressively more powerful, easier for people to use." [100] but does not define whether the data should be open. Instead, Berners Lee recommend making the linked data open. Linked data being open is denoted by the term *Linked Open Data*. The most common example for datasets following Linked Open Data is the Linked Open Data Cloud which covers (in May 2020) 1255 datasets with 16174 links [101].

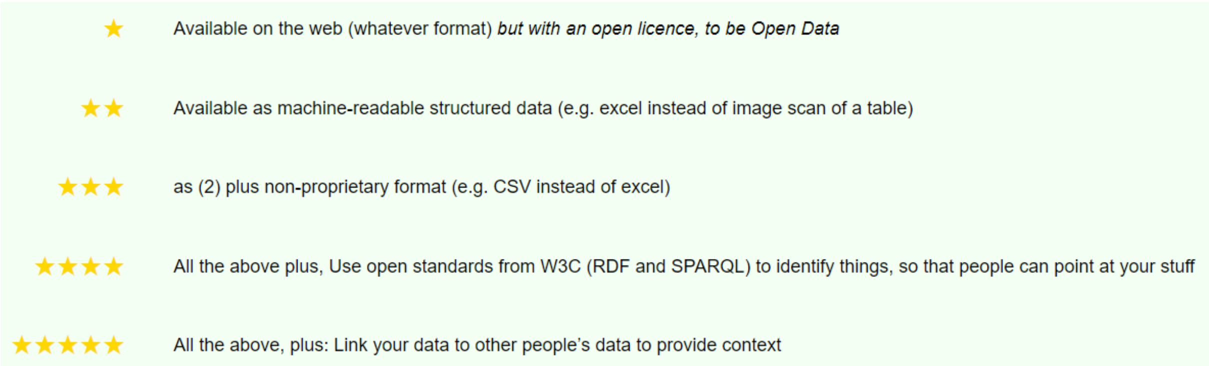


Figure 2.16: The five stars scheme as guideline for implementing Linked Data [100].

The relevance of Linked Data and Semantic Web in the field of BIM can be illustrated with the help of the BIM maturity levels as defined by Bew and Richards [102] (Figure 2.17). The use of **BIM** can be split into four maturity levels, each describing respective technological requirements. In contrast to the other level, there is no clear description of the final maturity level (Level 3). Following Rasmussen et al. [42], the final maturity level refers to the vision that "process and information is exchanged purely on a web-scale and fully integrated over disciplines and companies" [42]. And further, BIM maturity level 3 requires that "information is exchanged on the Web using open standards, and interoperable and decentralized model servers allow collaborative work on interoperable models and structured data" [42]. An approach to fulfill these requirements is the utilization of Semantic Web technologies for the data management. The realization of this vision is subject of the Linked Building Data (LBD) Community Group of the W3C [42]. A key challenge for the realization of the vision is the representation of geometrical descriptions through Semantic Web technologies. For example, Pauwels and Roxin [103] show that the neglection of geometrical descriptions and further simplification measures can reduce the file size of an ifcOWL file about ninety percent.

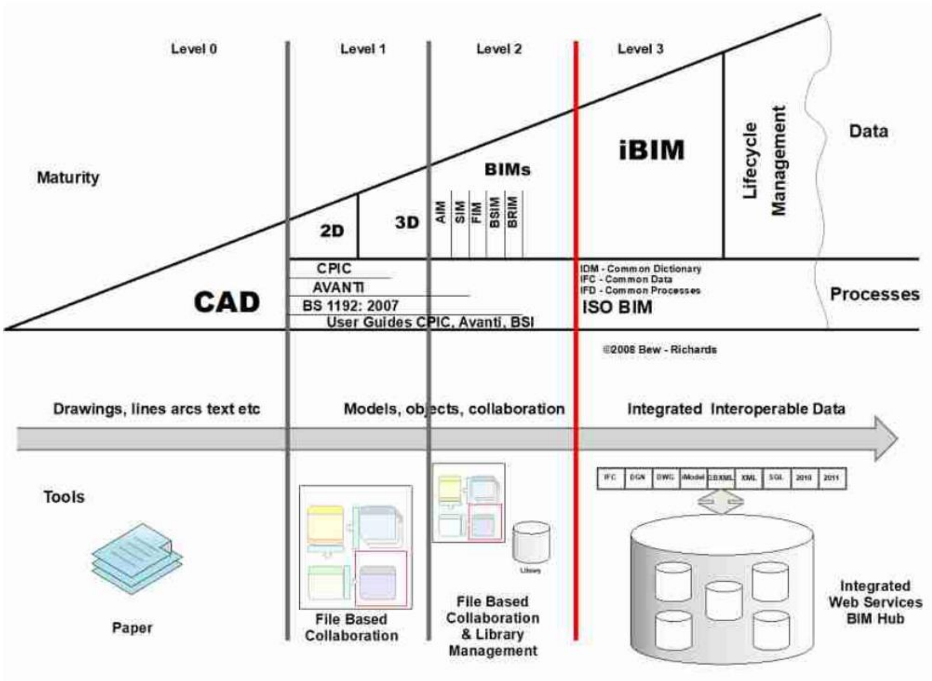


Figure 2.17: The four majority levels for the implementation of BIM in companies [102].

Pauwels et al. [104] discuss arguments for using Semantic Web technologies in AEC industries and emphasize “Linking across domains” as one of three main arguments. Here, "linking across domains is considered as the challenge to combine different content that is available in multiple applications (e.g., cost data, energy simulation data, geometrical data, GIS data). In this challenge, there is little to no need to ‘convert’ or ‘map’ data. Instead, the focus here is entirely on linking data, which can happen in a relatively loose fashion (linked data approach) or a formally rigid fashion (semantic web approach)" [104]. In other words, Pauwels et al. [104] consider the use of Semantic Web relevant for combining different contents of different domains such as **BIM and GIS.** Furthermore, Pauwels et al. [104] emphasize the connection of BIM and GIS as relevant use case of the argument linking across domains.

Example 1: RDF graph, Alignment, and SPARQL Query

A) RDF document

In the following, the IfcBeam and BuildingInstallation objects of the reference example are represented through RDF and turtle syntax. First, the namespace is defined what reduces the number of symbols in the documents and makes the document easier to read for humans. The prefix *beo* refers to an ontology that is based on the IfcBuildingElement subtree and is utilized instead of *ifcowl* due to lower complexity. The prefix *inst* refers to all instances created in this example. The prefix *ex* refers to an example ontology defining geometrical properties. The example ontology was chosen to attach the length value directly to the object to reduce the complexity of the RDF graph. The prefix *bldg* refers to classes related to buildings in the scope of CityGML.

```
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix beo:   <https://pi.pauwel.be/voc/buildingelement#> .
@prefix bldg: <http://www.opengis.net/citygml/building/2.0#> .
@prefix inst:  <http://instance.org/MyGeoBIMinstances#> .
@prefix ex:  <http://example.org/MyGeometricalProperties#> .
...
```

Afterward, the beam objects are instantiated based on the respective classes IfcBeam and BuildingInstallation. The classes belong to the namespaces *ifcowl* and *bldg* and their instantiation is achieved through the predicate *rdf:type*. Furthermore, the length value of the IfcBeam is attached to the instantiated beam object through the link predicate *ex:Length* and the literal “2.50”^^xsd:double. In general, a RDF statement is completed by a dot symbol. The semicolon at the end of an RDF statements indicates that the following RDF statement refers to the same subject.

```
...
inst:IfcBeam_23   rdf:type     beo:Beam ;
                  ex:Length    “2.50”^^xsd:double ;
                  ex:Id        "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
inst:BuildingInstallation_71   rdf:type   bldg:BuildingInstallation ;
                               ex:Id      "ad128f7c-2ce6-4996-894d-7e7e0659575e" .
```

B) SPARQL Query

The resulting RDF document can be queried through SPARQL. Similar to the RDF document, the namespaces have to be defined in the beginning indicated with PREFIX component.

```
PREFIX rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX beo:   <https://pi.pauwel.be/voc/buildingelement#>
PREFIX bldg: <http://www.opengis.net/citygml/building/2.0#>
PREFIX inst:  <http://instance.org/MyGeoBIMinstances#>
PREFIX ex:  <http://example.org/MyGeometricalProperties#>
...
```

Afterward, the object of interest is indicated by the notion SELECT which is followed by a WHERE block. The WHERE block covers RDF statements specifying information related to the object of interest. The object of interest is denoted with a term having a question mark at first place and referenced in the WHERE block. Similarly, a component of an RDF statement in the WHERE block can be designated with a question mark at first place so that it acts as kind of placeholder. Here, the length of those subjects which have the id "fa86005f-4ca3-437b-83af-1d03f5d5be19" is retrieved. In this example, the result is *“2.50”^^xsd:double*.

```
...
SELECT ?length
WHERE {
?subject    ex:Id        "fa86005f-4ca3-437b-83af-1d03f5d5be19" ;
            ex:Length    ?length  .
}
```

Result: “2.50”^^xsd:double

<u>Note</u>: The prefixes are used throughout all examples in this dissertation and are, due to practical reasons, not rewritten abbreviated through the symbols [@prefix] for RDF graphs and [PREFIX] for SPARQL queries and statements.

Example 2: RDF graph, Alignment, and SPARQL Query

In this example, the RDF graph of *Example 1* is separated into two distinct RDF graphs, namely ex:ifcModel and ex:citygmlModel. The alignment is created through a SPARQL statement and the relevant information are subsequently retrieved through a SPARQL query.

Graph: <u>ex:ifcModel</u>

```
[@prefix]
inst:IfcBeam_23   rdf:type     beo:Beam ;
                  ex:Length    “2.50”^^xsd:double ;
                  ex:Id        "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
```

Graph: <u>ex:citygmlModel</u>

```
[@prefix]
inst:BuildingInstallation_71   rdf:type   bldg:BuildingInstallation .
                               ex:Id      "ad128f7c-2ce6-4996-894d-7e7e0659575e" .
```

A) SPARQL statement - Alignment creation

The alignment creation is conducted through a SPARQL statement which is composed of an INSERT block, USING statements and additional RDF statements. The INSERT block specifies both the graph name of the alignment which is denoted with GRAPH, and the links of the alignment which are composed of the link predicated and the corresponding objects. The corresponding objects are further specified in the additional triples. Here, the corresponding objects are specified by their ID. The USING statement define the graphs relevant for the alignment creation.

```
[PREFIX]
INSERT {
  GRAPH ex:Alignment
  { ?citygmlObject    ex:equivalenceLink    ?ifcObject    . }
}
USING ex:ifcModel
USING ex:citygmlModel
?ifcObject          ex:Id     "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
?citygmlObject    ex:Id     "ad128f7c-2ce6-4996-894d-7e7e0659575e" .
```

B) RDF document – Integrated model

After applying the SPARQL statement, a graph is created specifying the alignment. Here, the objects inst:ifcBeam_23 and inst:BuildingInstallation_71 are related through ex:equivalenceLink in the graph with the name ex:Alignment.

Graph: <u>ex:Alignment</u>

```
[@prefix]
inst:BuildingInstallation_71    ex:equivalenceLink    inst:IfcBeam_23    .
```

C) SPARQL query

Afterward, information of interest is retrieved through a SPARQL query. The graphs providing the relevant information are defined through the FROM statements. Here, the length of the IfcBeam related to the BuildingInstallation_71 through ex:equivalenceLink is queried from the graphs ex:ifcModel, ex:citygmlModel and ex:Alignment.

```
[PREFIX]
SELECT ?length
FROM ex:ifcModel
FROM ex:citygmlModel
FROM ex:Alignment
WHERE {
```

```
        ?citygmlObject    ex:Id               "ad128f7c-2ce6-4996-894d-7e7e0659575e" .
        ?citygmlObject    ex:equivalenceLink  ?ifcObject .
        ?ifcObject        ex:Length           ?length .
Result: “2.50”^^xsd:double
```

3 Related Literature

3.1 Research structure

The dissertation follows knowledge-based DSR (chapter 1.2) which is why the **literature research** is the basis for the identification of the research problem. The literature research was mainly carried out using Scopus [105] which is a common database for scientific publications. The structure of the literature research is subdivided into two research areas, namely BIM-GIS Integration with a focus on linking information models and Semantic Web with a focus on contextual linking.

The literature research was complicated due to **ambiguous terminologies** used in the literature. This kind of ambiguity is majorly caused by two aspects:

- First, integrating information is the subject of several (mostly) independent research areas related to computer science like Database Engineering, Model-Driven Engineering, Ontology Engineering, and Semantic Web. The variety of these related research areas is accompanied by many publications and results in semantically incoherent terminologies.
- Second, the research area about ‘BIM-GIS Integration’ or ‘GeoBIM’ is an increasingly active since several years (Figure 3.1). This means that there is an increasing number of publications related to these keywords and that there is a variety of terminologies that have been not commonly defined yet.

To emphasize the difficulties accompanied by the ambiguity of terminologies, relevant terms are briefly discussed in Table 3.1 regarding *partial synonyms*, *polysemes,* and *hypernyms*. The term partial synonyms refer to terms that can be used interchangeably in some (but not all) situations, the term *polysemes* refers to terms with several meanings and the term *hypernyms* refer to umbrella terms.

3.2 BIM-GIS Integration

The literature research about BIM-GIS Integration address two pairs of keywords, namely ‘BIM GIS’ and ‘CAD GIS’ (chapter 2.1). Furthermore, the literature research was limited to the subject areas ‘Engineering’, ‘Computer Science’ and ‘Earth and Planetary Science’. In total, four different **queries** were applied with different combinations of the two keyword pairs (I – IV). The results are represented in Figure 3.1.

TITLE-ABS-KEY(CAD GIS) **OR** TITLE-ABS-KEY(BIM GIS) AND (LIMIT-TO (SUBJAREA,"ENGI") OR LIMIT-TO (SUBJAREA,"COMP") OR LIMIT-TO (SUBJAREA,"EART")) (I)

TITLE-ABS-KEY(CAD GIS) **AND** TITLE-ABS-KEY(BIM GIS) AND (LIMIT-TO (SUBJAREA,"ENGI") OR LIMIT-TO (SUBJAREA,"COMP") OR LIMIT-TO (SUBJAREA,"EART")) (II)

TITLE-ABS-KEY(CAD GIS) **AND NOT** TITLE-ABS-KEY(BIM GIS) AND (LIMIT-TO (SUBJAREA,"ENGI") OR LIMIT-TO (SUBJAREA,"COMP") OR LIMIT-TO (SUBJAREA,"EART")) (III)

TITLE-ABS-KEY(BIM GIS) AND (LIMIT-TO (SUBJAREA,"ENGI") OR LIMIT-TO (SUBJAREA,"COMP") OR LIMIT-TO (SUBJAREA,"EART")) (IV)

Table 3.1: Hypernyms, synonyms and polysemes of relevant keywords in the literature research.

Information integration	S	- Examples for partial synonyms commonly used in related research areas are *data integration*, *record linkage*, *entity resolution*, *object identification*, *model weaving*, *data matching*, *ontology alignment.* - The term interoperability is often used synonymously to information integration even though it is inexact.
	P	The term information integration can be divided into the *information* (subject) and *integration* (activity), both representing polysemes: - Information can refer to an *information model*, *relational database,* or *ontology*, etc.. Furthermore, information can refer to different levels like schema or instance-level. - Integration can be limited to specific integration methods like conversion or refer to all types of integration methods.
BIM-GIS Integration	H	The term BIM-GIS Integration is an umbrella term referring to: - different integration subjects like processes, applications, or information models. - all types of integration (as emphasized in the cell above).
	S	Examples for partial synonyms are *GeoBIM*, *CAD GIS Integration*, *BIM-UIM Integration, City Information Modeling.*
Ontology	P	The term ontology in the field of computer science generally refers to all kind of formal knowledge representations or is limited to those based on formal semantics.
	H	Depending on the definition, ontologies might refer to several types of computer-based knowledge representations like *information models*, *taxonomies*, or *relational databases.*
Alignment	S	As example, partial synonymous to the term alignment are *link model*, *linkage*, or *link set.*
Object	S	As example, partial synonymous to the term object are *instances*, *instance-level information*, *data*, or *individuals*.
Contextual linking	P	There is no common definition of contextual linking. For example, contextual linking might refer to the enrichment of links with metadata, or to the validity scope of a link.
	S	Some authors use different terms like context-dependent linking, contextual linking, or context-sensitive linking.
Semantics	P	For example, the term semantics in the field of computer science might refer to non-geometric data or to the meaning of the data.

S = partial synonyms; P = polyseme; H = hypernyms

The curve referring to ‘CAD GIS’ OR ‘BIM GIS’ (I) shows that the number of publications referenced by these keyword pairs is steady until 2004, then slightly increases and rapidly increases after 2018. When addressing ‘CAD GIS’ AND NOT ‘BIM GIS’ (III), the curve increases with 2004 and then again declines after 2009. On the contrary, the curve belonging to the keyword pair ‘BIM GIS’ (IV) starts in 2007 and increases from 2010, and further increases by 2018. Noteworthy, this curve also includes literature about ‘CAD GIS’ while the curve about ‘CAD GIS’ AND NOT ‘BIM GIS’ (III) excludes literature about ‘BIM GIS’. Nevertheless, there are only a few publications referencing both ‘CAD GIS’ AND ‘BIM GIS’ (II).

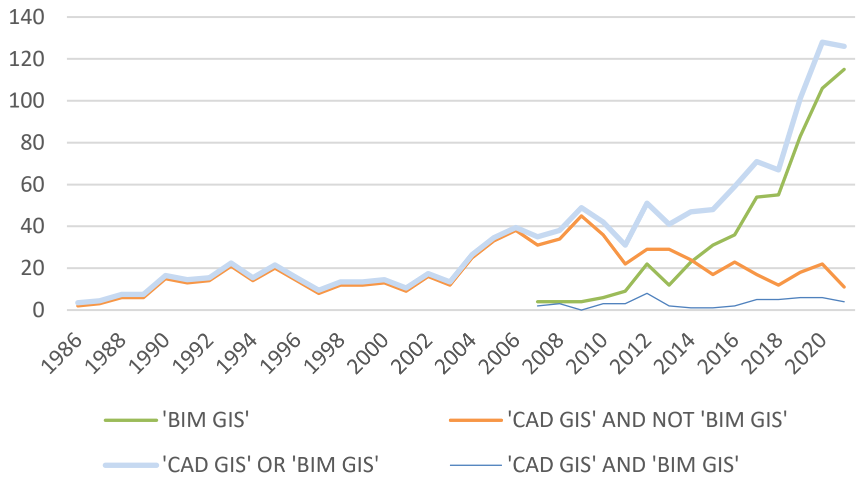


Figure 3.1: Number of publications based on the queries following four different combinations of the keyword pairs 'CAD GIS' and 'BIM GIS'.

The query belonging to the keyword pair ‘BIM GIS’ (IV) has resulted in 643 documents. From these **documents**, 205 documents were selected by inspecting the number of references, the quality of the publication platform, and the number of citations. Of these 205 documents, 54 documents were either not accessible or considered as irrelevant for this dissertation. Consequently, 151 publications about ‘BIM GIS’ were analyzed in more detail in this literature review (Figure 3.2) and are further referenced by the term *reviewed literature*.

The most common **publishing** platforms are *ISPRS IJGI*, *ISPRS Int. Annals*, *Int. Arch. Photogramm.*, *Remote Sens. Spatial Inf. Science*, and *Automation and Construction*. Further common publishing platforms are *Advanced Engineering Informatics*, *Journal of Spatial Science*, and *Linked data in Architecture and Construction* (*LDAC)*. The most common publishing institute is ISPRS with more than one-quarter of all reviewed publications. The rest of the publications belong to other media types and platforms like books, or less common scientific conferences or journals about the topic ‘BIM-GIS Integration’.

The authors with the most publications in the reviewed literature are Stoter (14 publications), Biljecki (9 publications), Arroyo (9 publications), Kolbe (9 publications), Hijazi (7 publications) and

Tauscher (7 publications). In Figure 3.4, all authors with more than four publications are represented through absolute numbers of publications and annual publications. In the scope of the reviewed literature, most of these authors published after 2014, while Kolbe, Hijazi, Donaubauer, and Borrmann published before 2014. Notably, some of these authors have also published papers related to the keywords 'CAD GIS' or 'BIM GIS' integration not part of the reviewed literature.

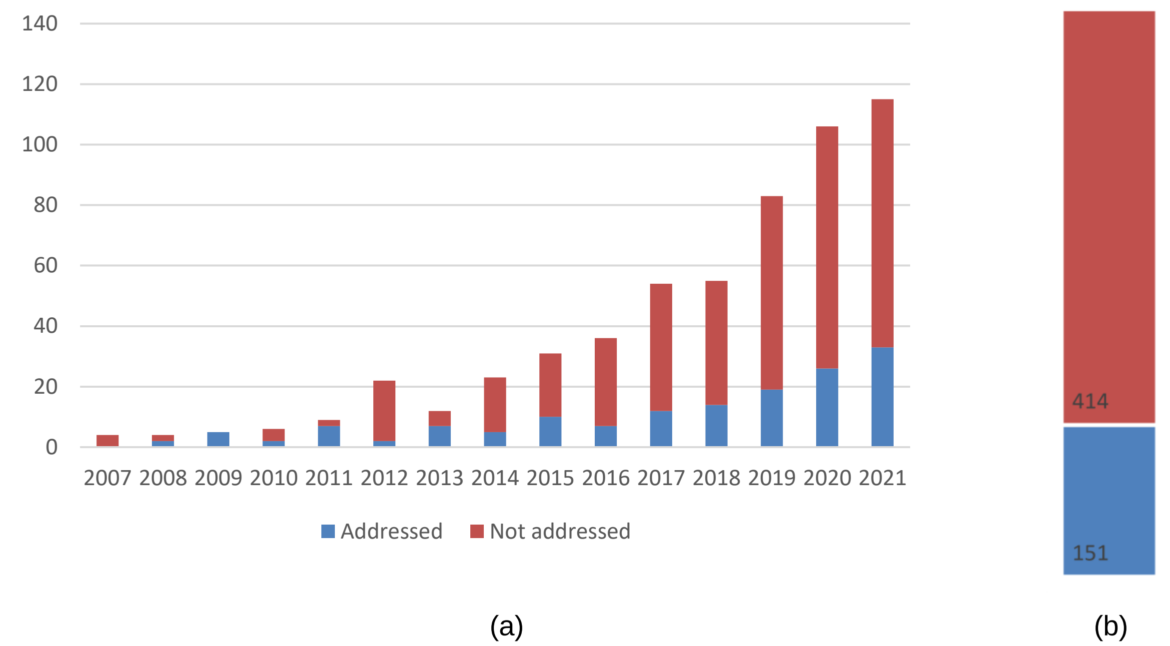


Figure 3.2: (a) Number of publications about BIM-GIS Integration and the reviewed publications with respect to the year; (b) Number of publications about BIM-GIS Integration and the reviewed publications in total.

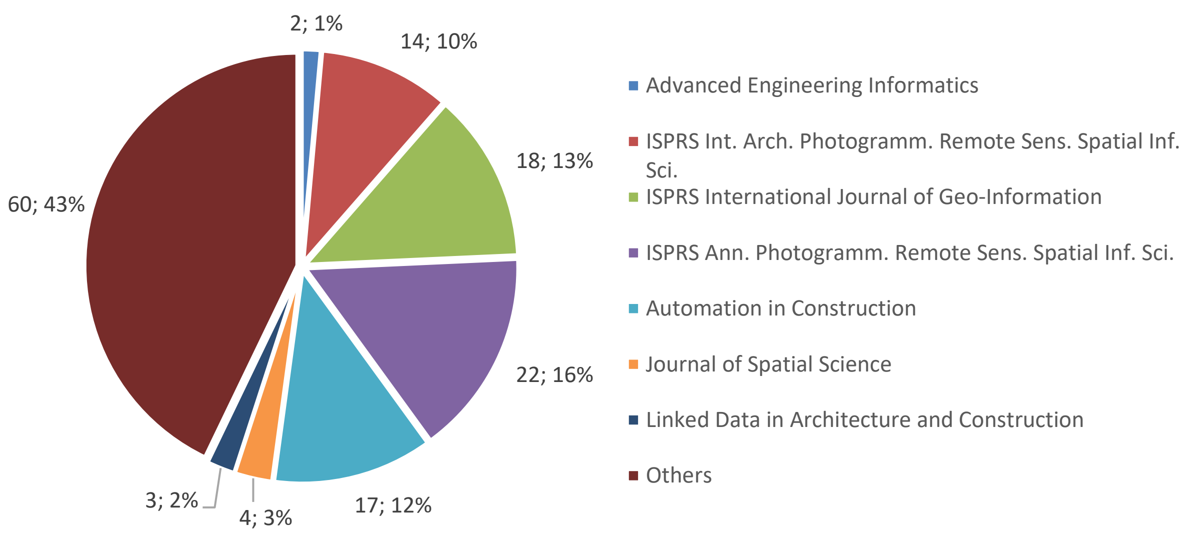
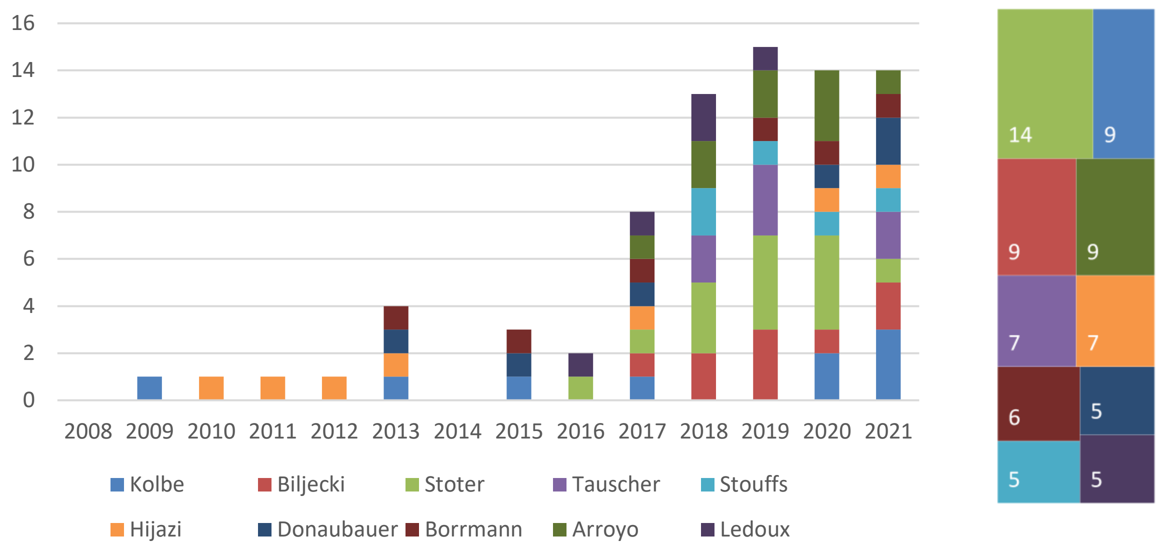


Figure 3.3: Publication platforms of the reviewed literature in absolute numbers and percentage.



(a) (b)

Figure 3.4: Authors with more than four publications in the reviewed literature about 'BIM GIS' represented through (b) absolute numbers of publications and (a) annual publications.

The absolute **number of authors** per year is represented in Figure 3.5 which refers to the sum of the authors of all papers per year including duplicates. The absolute number is further subdivided into the effective number and the duplicates. The absolute number of authors was constant until 2013, has afterward increased slightly, and increased rapidly from 2019 (similar to the number of reviewed publications). The number of duplicate authors per year is close to zero until 2015, when it slightly increases and then heavily increases from 2018 to 2019. Furthermore, Figure 3.5 shows the number of authors per publication per year based on the absolute number of authors and the number of reviewed publications. The jump in the start of the course is caused by the low number of publications at the beginning of the considered period and the consequently strong influence of single publications. In 2020 and 2021, the numbers are influenced by some publications with a number of authors above average, like Noardo et al. [79] with 16 authors and Biljecki [106] with 10 authors. Nevertheless, the curve about the number of authors per publication per year shows that there is a tendency toward an increasing number of authors participating in a publication.

In Figure 3.6, the value 'No. of publications per authors (effective)' illustrates how many publications were published in ratio to the effective number of authors: The higher the value, the more authors with many publications in that year. Notably, the value does not show the no. of publications an author publishes on average per year (what would be in all situations larger or equal to one when only including authors who have published in that year). The value is highest between 2009 and 2013, since the publications in this time span belong to a few numbers of authors. Afterward, the value slightly declines on average even though there are more authors with several publications per year. However, both the number of authors per publication and the number of authors with one publication is increasing (Figure 3.5).

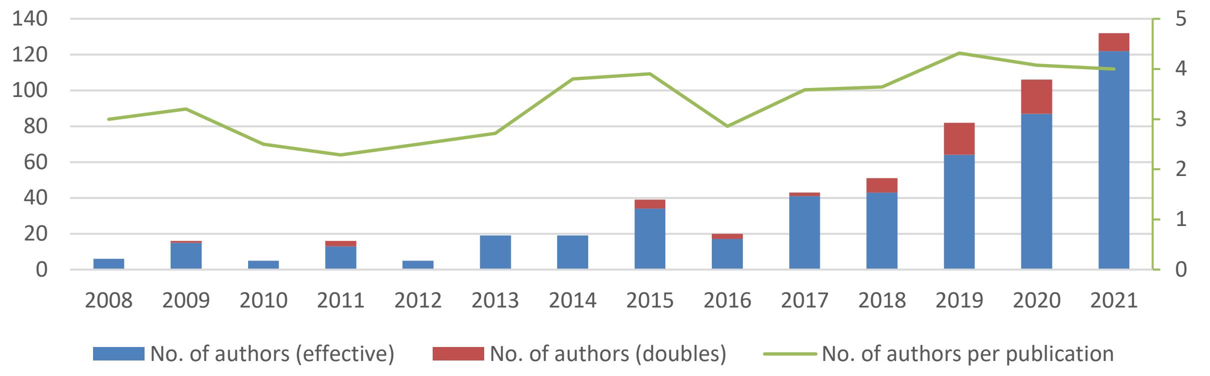


Figure 3.5: *Number of authors per year in the scope of the reviewed publications as effective number and difference to an absolute number. The number of authors per publication per year based on the absolute number of authors and number of reviewed publications.*

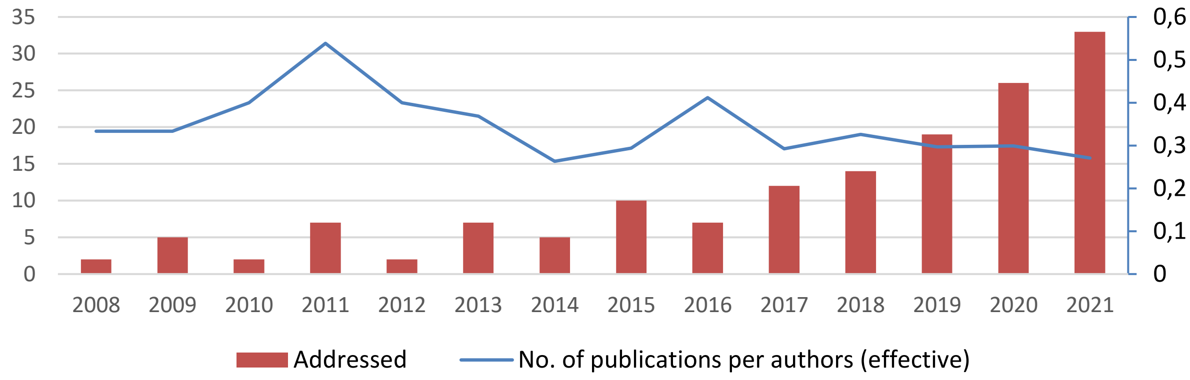


Figure 3.6: *Number of reviewed publications per year. The number of publications per authors per year in terms of the ratio of the number of reviewed publications per year to the effective number of authors in the same year.*

Consequently, the structure of the literature review is subdivided into five periods. The periods are deduced from the analysis of no. of publications, number of authors, publications per author of the reviewed literature, and the number of publications related to the keyword pair ‘CAD GIS’. Notably, the subdivision of periods does not consider the content of the publications. In summary, there are five periods, namely *until 2007*, *2007-2009*, *2010-2013*, *2014-2018*, and *2019-2021*, which are further described in Table 3.2. The period *until 2007* is not further subdivided since it is not of major interest for further literature research.

In Table 3.3, the reviewed literature is briefly summarized following the defined periods and the document type, namely *overview*, *review,* and *integration effort*. Additionally, the publications belonging to the publication type *integration effort* are further structured according to the integration methods *conversion*, *extension*, *merging,* and *linking* (the integration methods are described in chapter 7.2.2). For the period *until 2007* only some publications related to the keyword pair ‘CAD GIS’ which are referenced by publications related to the keyword pair ‘BIM GIS’ are described.

Table 3.2: Periods deduced from the analysis of no. of publications, no. of authors, and publications per authors based on the keywords 'CAD GIS' and 'BIM GIS'.

	'CAD GIS'	'BIM GIS' (reviewed literature)			
	No. of publications	No. of publications	No. of authors	Publications/ authors	Description
until 2007	o	n.a.	n.a	n.a	The acronym BIM is not part of research publications. Only few publications belonging to this period are referenced in current publications. From 2004, the number of research referencing both CAD and GIS has significantly increased.
2007 - 2009	+	-	-	o	The acronym BIM is addressed by the first publications related to CAD/BIM-GIS Integration. The number of publications referencing both CAD and GIS is increasing.
2010 – 2013	o	o	-	+	The number of publications related to BIM and GIS has slightly increased while the number of authors remained more or less the same. Consequently, the number of publications per authors has increased. The number of publications related to CAD and GIS has declined.
2014 – 2018	o	+	o	o	Both the number of publications related to BIM and GIS and the number of authors has increased, whereby the ratio publications per authors has declined. Thus, the publications are distributed among more authors than in the period before.
2019 – 2021	o	++	+	-	Both the number of publications related to BIM and GIS and the number of authors have rapidly increased. On the other side, the ratio publications per authors has declined again.

high number ++ + o - -- low number

The publications of the reviewed literature were categorized with respect to subcategories belonging to *document type*, *integration method*, *source model*, *integrated model*, *use case*, and *alignment*. The analysis of the use case and integration method are described in the respective chapters, chapter 5 and chapter 7.2.2 respectively. In total, the literature review covers 1171 assignments to respective categories. The consequent table can be accessed online [107]. The outcome with respect to the categories is discussed in the following sections and illustrated through diagrams. It is true for all categories that a clear categorization is not always possible which is why some publications are assigned to several categories.

The categorization of the reviewed literature with respect to the ***Document type*** covers the subcategories *Overview*, *Review*, *and Integration effort* (Figure 3.7). Overviews aim to provide literally an overview about the topic. Reviews summarize and analyze the state of the art on the topic. Integration efforts describe a particular integration approach. In total, the reviewed literature covers 15 overviews, 16 reviews, and 122 integration efforts. In the following, the publications and respective trends in the categories *Overview* and *Review* are described. Publications referring to the category *Integration effort* will be discussed in more detail in the next chapters. Interestingly, after 2017, there is a higher percentage of overviews and reviews compared to the years before.

One of the most often cited **overviews** is from Kolbe and Plümer [12] and Nagel et al. [13] who describe, among others, the heterogeneities between IFC and CityGML. Furthermore, the categorization of integration methods as described by Hijazi et al. [85] is the basis for further publications summarizing integration methods in the field of BIM-GIS Integration. In 2018 and 2019, the percentage of the number of overviews is relatively high compared to other years which is because of several publications related to the research project EuroSDR. The most extensive overview of technological challenges and opportunities of BIM-GIS Integration with respect to semantic 3D city modeling is provided by Kolbe and Donaubauer [4].

The first **review** in the reviewed literature was in 2015, then there was one year break, and in 2017 and 2018, there is a relatively high percentage of reviews. In a nutshell, some literature reviews are more extensive than others but there is no outstanding review. The reviews often discuss the same subject but generally utilize different categorization approaches. For example, several reviews consider different kinds of use cases to which the literature is assigned.

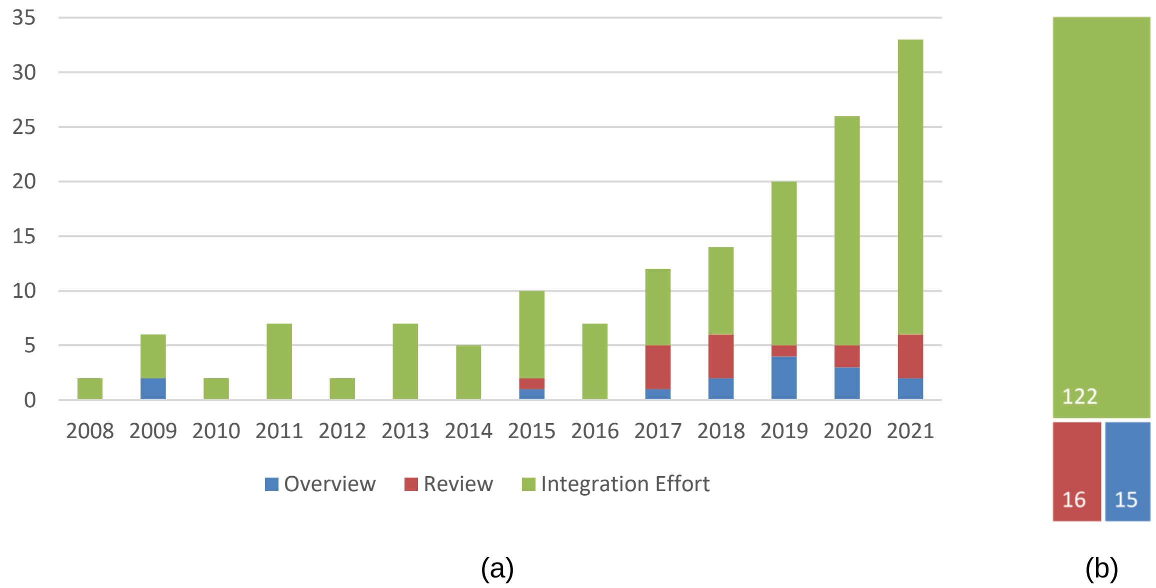


Figure 3.7: Number of publications of document types Overview, Review and Integration Effort (a) per year and (b) in absolute numbers.

Table 3.3: Summary of reviewed literature about 'BIM GIS' and relevant literature about 'CAD GIS'.

Overviews	
Until 2007	In 2004, Kolbe and Plümer [12] have published an article about opportunities and challenges for "Bridging the Gap of GIS and CAAD". One year later, van Osteroom et al. [108] have written a book chapter about a similar topic, namely "Bridging the worlds of CAD".
2007 - 2009	In 2009, Nagel et al [13] have discussed the opportunities and challenges of converting CityGML to IFC models. Casey and Vankadara [88] have discussed the syntactic and semantic interoperability of CAD and GIS models and emphasize some use cases for BIM-GIS Integration.
2010 – 2013	-
2014 – 2018	In 2015, Brüggemann and Both [14] have described the relevance of BIM in the scope of 3D City Modeling and emphasize the differences between IFC and CityGML. In 2017, Hijazi and Donaubauer [85] have concluded four integration methods for BIM-UIM Integration, namely Conversion, Linking, and Unified Model, and describe several use cases. One year later, Arroyo et al. [109] have briefly described the differences between IFC and CityGML models, and Ellul et al. [110] have summarized the "state-of-play" in the field of GeoBIM as a basis for the EuroSDR research project.
2019 – 2021	The EuroSDR research project has resulted in additional three publications in 2019 [6,111,112], two in 2020 [79,84] and one in 2021 [113]. In a nutshell, these documents summarize GeoBIM activities in several European countries, emphasize relevant use cases and tools, and describe respective opportunities and challenges. Apart from that, Biljecki and Tauscher [90] have described challenges for the conversion of IFC to CityGML models, Herle et al. [11] have discussed the integration of building and geospatial information models with respect to interoperability levels, and Kolbe and Donaubauer [4] have extensively described opportunities and challenges for the Integration of Semantic 3D City Modelling and BIM.
Reviews	
2004 – 2009	-
2010 – 2013	-
2014 – 2018	In 2015, Fosu et al. [114] have analyzed the publications related to BIM-GIS Integration with respect to the publication platform, the use case and the integration

approach. In 2017, Kurwi et al [115] have assigned publications about railway projects to respective project phases. Similarly, Song et al. [116] have related the publications to the subject city or construction phases in the scope of buildings and described respective use cases. Liu et al. [81] have discussed the state of the art with respect to several integration methods and use cases. In 2018, Zhu et al. [117] have described differences between IFC and CityGML models based on previous literature. Further literature reviews are provided by Ma and Ren [118], Sani and Abdul Rahman [83], and Wan et al. [119].

2019 – 2021 In 2019, Wang et al [120] have reviewed publications with respect to sustainability and BIM and GIS Integration, and categorize integration efforts with respect to their sovereignty, like GIS leads and BIM supports. In 2020, Beck et al. [2] have emphasized a differentiation between integration efforts in BIM-GIS Integration and described categories of real-world objects, integration method, and purpose. Hbeich et al. [82] have categorized interoperability approaches with respect to literature about BIM-GIS Integration. In 2021, Shkundalov and Vilutienė [121] have addressed the identification of use cases and integration methods for 4D BIM and GIS. Liu et al. [81] have published a bibliometric analysis of BIM, GIS, and Web Environments. Further reviews are provided by Kurwi et al. [115] and Wei et al. [122]. In 2022, Souza et al. [123] categorize the literature by its contribution and the utilized tools.

Integration Method

Conversion

2004 - 2009 -

2010 – 2013 From 2010 to 2013, El-Mekawy et al. [89,124–126] have addressed the bidirectional conversion of CityGML and IFC through Unified Building Model (UBM) what is a merged model at schema level. In the same time span, Irizarry and Ebrahim [127] have imported BIM data to ArcGIS to optimize the site layout with respect to tower cranes and, in another publication [128], GIS data to Revit for construction supply chain management. In 2010, Hijazi et al. [129] have investigated the conversion of IFC to some database to visualize and query through a 3D web viewer. Furthermore, Donkers [130] has addressed the automatic conversion from IFC to CityGML and was later the first author of a related paper [91]. Rizal et al. [131] have conceptually addressed the conversion of IFC to CityGML for low-disturbance construction, and Bansal et al. [132,133] and Elbeltagi et al. [134]

have addressed the conversion based on proprietary software products like ArcGIS and Revit.

2014 – 2018 In 2014, Kim et al. [135] have described the conversion of IFC to Civil3D for infrastructure alignment construction, and Rafiee et al. [136] the conversion of IFC to geographic vector format for shadow analysis. In 2015, Kang and Hong [80] have addressed the conversion of IFC to CityGML for Facility Management, Niu et al. [137] the conversion of gbXML to KML for energy planning, and Yaagoubi et al. [138] the conversion of several information models to InfraWorks for heritage management. In 2016, Deng et al. [139] have addressed noise mapping simulation and Wu and Zhang [140] the evacuation scenarios through ArcGIS and Revit. Furthermore, Deng et al. [24] and Donkers et al. [91] have investigated the IFC to CityGML conversion. After that, Arroyo et al. [141,142] have addressed the conversion of IFC and CityGML to CGAL structures. In 2018, Tauscher et al. [96] propose the use of triple graph grammar for the conversion of IFC to CityGML and Chen et al. [143] have discussed the conversion of IFC to 3D Tiles for visualization purposes. Further publications are from Yamamura et al. [144], Niu et al. [137], Kardinal Jusuf et al. [145], and Kang and Hong [146].

2019 – 2021 In 2019, Tauscher [95] have further discussed IFC-to-CityGML conversion rules based on triple graph grammar, Zhou et al. [147] have developed an algorithm for the conversion of outer surfaces from IFC to CityGML, McGlinn et al. [148] and O'Donovan et al. [149] have investigated the ifcOWL/ IFC to GeoSPARQL conversion for visualization purposes. Furthermore, Esser and Borrmann [150] have discussed the conversion of Plan Pro models to IFC, and Zhu et al. [151–153] addressed the IFC to shapefile conversion. In 2020, Chen et al. [154] have investigated the conversion of IFC to Virtual Globes for adjacency analysis of indoor spaces, Jetlund et al. [155] have discussed the conversion of IFC EXPRESS to UML, and Xu et al. [156] have investigated the conversion of IFC to 3D Tiles for visualization purposes. In 2021, Chen et al. [157] have discussed the conversion of IFC to osgEarth for visualization purposes, Yang et al. [158] have addressed the conversion of IFC and other data to ArcGIS for urban flooding simulation, and Moretti et al. [159] and Zhu et al. [153] have investigated the conversion of IFC to shapefile for asset management and visualization purposes respectively. Further publications are from Salheb et al. [160], Noardo et al. [161], Clemen et al. [162], Akob et al. [163], Ding et al. [164], Vacca & Quaquero [165], Bayat et al. [166], Bansal [167], Gabriele [168], Gobeawan et al. [169], Andrianesi et al. [170], AlSaggaf et al. [171], Barazetti [172], Floros et al. [173], Guntel et al. [174], Basir et al. [175], and Barzegar et al. [176]

Extension

2004 – 2009 In 2009, Tegtmeier et al. [177] have developed a CityGML ADE, called 3DIM, to represent information from IFC, CityGML, and other geo-databases.

2010 – 2013 In 2011, De Laat and van Berlo [178] have developed a CityGML ADE, called GeoBIM, which supports the representation of building information based on IFC by the extension. Hijazi et al. [179] have utilized UtilityNetworkADE which is an ADE for CityGML and aims to support the representation of information about utility infrastructure.

2014 – 2018 In 2013 and 2015, Bormann et al. [180,181] have developed an IFC extension for shield tunnels to support the representation of these kinds of objects in IFC. In 2018, Stouffs et al. [182] have developed the ifcADE which is an extension of CityGML and aims to support the representation of information from IFC.

2019 – 2021 In 2019, Wang et al. [183] have developed the schema called Underground utilities which is intended to be an extension for either IFC or CityGML. One year later, Jetlund [184] has designed an UML Profile to bridge UML and IFC EXPRESS. In 2021, Biljecki et al. [106] have further developed the ifcADE from Stouffs et al. [182], Petronijevic et al. [185] have designed an extension of IFC to support the Land Administration Domain Model and Wilhelm et al. [186] have developed the EnvPlan ADE for CityGML to represent environmental planning data and some IFC data.

Merging

2004 - 2009 In 2005, Benner et al. [187] have developed the QUASY model which is structurally similar to CityGML but also intended to be capable of representing information based on IFC. In 2008, Choi et al. [188] have proposed the Ubiquitous Space Information Model (USIM) which is largely based on IFC and intended to support Indoor GIS simulations. One year later, Ghafourian and Karimi [189] have published an information model intended to support indoor and outdoor navigation.

2010 – 2013 In this time span, El-Mekawy et al. [124–126] have developed the Unified Building Model to support the bidirectional conversion between IFC and CityGML

2014 – 2018 In 2014, Karan and Irizarry [190] have proposed the Facility Management ontology (FM ontology), and Mignard and Nicolle [191] have published the Urban Information Model (UIM) bridging IFC and CityGML. One year later, Amirebrahimi et al. [192] have developed a merged model based on IFC and CityGML for the evaluation of flood damages. Aien et al. [193] have proposed the 3D cadastral data model (3DCDM) to represent both physical and cadastral data based on IFC, CityGML and Land Administration Domain Model (LADM) in one information model. Furthermore, Shojaei [194] have used WebGL as a merged model for IFC, COLLADA, and other data for visualization purposes. In the following years, Hor et al. [20,195] have proposed the Integrated Geospatial Information Model (IGIM) intended to support the alignment of IFC and CityGML as RDF or Labelled Property Graphs. In 2016, Deng et al.[24] have suggested the Semantic City Model to achieve bidirectional conversion similar to El-Mekawy et al. [89,124,125] and Teo and Cho [196] have proposed the Multi-purpose Geometric Network Model (MGNM) to combine indoor and outdoor networks. In 2017, Peckiene et al. [197] have proposed to merge models for site layout planning, and Sicilia et al. [198] have investigated the so-called District Data Model. Furthermore, Arroyo et al. [141] and Arroyo et al. [142] have discussed the use of CGAL structures as merged model to convert IFC to CityGML models.

2019 – 2021 In 2019, Kumar et al. [199] have evaluated the suitability of LandInfra overarching model between IFC and CityGML with respect to transport infrastructure. One year later, Chen [154] have suggested an Indoor GIS Model (IGSM) to convert IFC to Virtual Globes for indoor network analysis, and Li et al. [200] have proposed the Integrated Building Model (IBM) to merge IFC and CityGML as basis for the smart built environment. Furthermore, Lee et al. [201] and Lv et al. [202] have merged building and geospatial data to some database. In 2021, Demir Altıntaş et al. [203] have proposed the Zoning Domain Model to merge IFC and GML data for automated checking of zoning codes. Furthermore, Karimi et al. [204] have investigated the Building Information Robotic System (BIRS) Ontology for robot navigation on sites, and Hor and Sohn [205] have further developed the integration approach using IGIM.

Linking

2004 – 2009 Peachavanish et al. [16] (2006) and Akinci et al. [17] (2009) have suggested to link CAD and GIS data using Semantic Web technologies and emphasized site layout planning and project management as use cases. In 2008, Li and He [206] have created topological graphs for navigation purposes.

2010 – 2013 In 2013, Esfahani [207] have linked IFC, CityGML and further data using XML files called Multi Model Method to support transport infrastructure design.

2014 – 2018 In 2016, both Karan and Irizarry [19] and Hor et al.[195] have discussed linking IFC and CityGML models using Semantic Web technologies. One year later, Sicilia and Costa [198] have aimed to link IFC and CityGML models using SW technologies. Furthermore, Vilgertshofer et al. [23] have linked IFC and CityGML models using SW technologies and applied SPARQL queries to retrieve data about tunnels and their infrastructure alignments. Zheng [208] has linked IFC and OKSTRA using Semantic Web technologies and applied queries to evaluate the distances of objects to infrastructure alignments. In 2018, Esfahani [209] and Hor et al. [20] have developed further their proposed linking approaches, Multi Model Method and Semantic Web technologies respectively.

2019 – 2021 In 2019, Zhao et al. [210] have linked IFC and CityGML models to identify the optimal route for a transport infrastructure alignment. McGlinn et al. [18] have linked geospatial and building data based on BOT using SW technologies and emphasized heritage management and project management use cases. Djuedja et al. [211] have suggested to integrate environmental data and building data for environmental assessment. In 2020, Hbeich and Roxin [93] have published a paper about linking relevant concepts of IFC Schema and GIS standard ontologies based on SW technologies, Huang et al. [212] have proposed to link IFC and CityGML models using Semantic Web technologies for solar energy simulation, and Hijazi et al. [213] have linked IFC and CityGML at building level using relational structure mainly for visualization purposes. In 2021, McGlinn et al. [21] have aimed to link IFC models with geo-databases and DBPedia. Furthermore, Beck et al. [3] have linked IFC and CityGML models using Semantic Web technologies and subsequently applied queries in order to demonstrate the context-dependency of the links, and Stepien et al. [22] have linked IFC tunnel alignments with cadastral and geospatial data for finding an optimal tunnel alignment.

Since the instance-level linking is in the major focus in this dissertation, the alignments used in the related literature to like information models from the domains BIM and GIS are further analyzed. Alignments between information models from the domains BIM and GIS can be defined at instance and schema level. These alignments are expressed either formally through modeling languages like Semantic Web technologies [18,19,93,195,210] and UML/ EXPRESS-G [92,155,182] or informally, e.g. using tables [95,149,181,214] or text/ figures [89]. In an alignment, the correspondence between two entities is expressed through some relation type. The used relation types of instance-level alignment and the syntactical representation of the alignment of the reviewed literature are represented in Table 3.4. As an example, Hijazi et al. [213] utilize a relation table for linking GMLIDs of CityGML buildings and IFC models. Hor et al. [195] propose to link IFC and CityGML models through the concepts Equivalent, As-is, and Has an attribute following Semantic Web technologies. Huang et al. [212] propose to link window objects represented as ifcOWL or BOT and CityGML using the link predicate *skos:exactMatch* from the SKOS vocabulary. Vilgertshofer et al. [23] use Semantic Web technologies to link the globalID of IFC elements to the gml:id of CityGML element but without specifying any link predicate. Stepien et al. [22] link several different models like city models and cadastral maps to an infrastructure alignment of a tunnel using Semantic Web technologies through topological relationships. Overall, there is no common relation type utilized in research work about instance-level alignments relating information models from the domains BIM and GIS.

Table 3.4: Relation types and their syntactical representations for linking information models of the domains BIM and GIS identified n the reviewed literature.

Publication	Relation type	Representation
[13]	n:m, 1:m, n:1, 1:1	Figure, Text
[89]	Full, partial, none	Figure, Table, Text
[20,195]	Equivalent, Is-as, Has an attribute	SW technologies
[198]	Shared entity	SW technologies
[23]	lnk:Link, owl:equivalentClass	SW technologies
[208]	"has closest road section"	SW technologies
[211]	*n.a.*	SW technologies
[210]	1:1 (Similarity Matrix)	SW technologies
[213]	*n.a.*	Relational
[212]	skos:exactMatch	SW technologies
[3]	skos:related	SW technologies
[22]	DE-9IM	SW technologies

3.3 Semantic Web: Contextual linking

Semantic Web is one of several research areas dealing with the alignment of ontologies or data models, like data warehousing. The research area Semantic Web was investigated in more detail due to two aspects: First, Semantic Web technologies are also addressed by the majority of publications about linking information models from the domains BIM and GIS [11,18,19,104,215]. Second, contextual linking is a current research topic in the field of Semantic Web [216–218] that is particularly relevant for this dissertation. The literature research with respect to Semantic Web focuses on the topic of contextual linking and is based on document search in Scopus following several variations of the keyword pairs “Context” and “Semantic Web”. Furthermore, the literature research is based on literature reviews about the problem of identity links in the scope of the Semantic Web [216] and alternative alignment languages to OWL from Euzenat et al. [35] since these topics are strongly related to contextual linking as described in the following.

In general, there are two different understandings of the term ***contextual linking*** in the scope of the Semantic Web: First, contextual linking refers to the enrichment of links/ alignments with metadata like authorship or matching algorithms which will be further called *metadata approaches*. Second, contextual linking is referenced by research work aiming to link heterogeneous ontologies created from different perspectives. Here, the term *context* either refers to the related models or to the alignment in terms of contextual alignment. In the following, common contextual linking in terms of heterogeneous ontologies are discussed in more detail in Table 3.5.

Table 3.5: Common contextual linking approaches addressed in the literature review about contextual linking in the field of Semantic Web technologies.

Year	Approach	Description
1995	IST	In 1995, Guha [219] has investigated the dependency of statements in knowledge bases on context and has proposed the expression *ist*, which refers to a statement *“is true in”* some context. The expression *ist* was further adopted by several authors, e.g. by Bao et al. [220] and Aljalbout et al. [217]
2003	C-OWL	In 2003, Bouquet et al. [218] have proposed Context OWL (C-OWL) to relate contextual ontologies. C-OWL covers so-called bridge rules describing relations between entities of the contextual ontologies through five relation types, namely disjoint (⊥), equivalence (≡), related (∗), more abstract (⊆), and more specific (⊇).
2004	ε -Connections	In 2004, Kutz et al. [221] have developed so-called ε -Connections which are *n*-ary relations between disjoint ontologies and are based on description logics.
	Lifting	Furthermore, Guha [222] has extended the proposal about the expression *ist* through so-called *lifting rules*. Lifting rules describe the transfer of information from one context to another.
	SKOS	In the same year, the Simple Knowledge Organization System (SKOS) core 1.0 guide was published which was continuously further developed and provides a data model for sharing and linking ontologies. Among others, SKOS provides several link predicates like *skos:related* or *skos:exactMatch*.
2006	Compatibility Relations	In 2006, Bouquet et al. [223] have proposed *compatibility relations* which describe the compatibility between contexts. In more detail, a compatibility relation indicates whether an entity refers to the same interpretation in different contexts.

2007	EDOAL	The *Expressive and Declarative Ontology Alignment Language (*EDOAL*)* from [224] extends the so-called alignment format through several vocabularies to express complex relations between entities of ontologies like restriction or transformations. In EDOAL, a relation between two entities is expressed through a *cell* which can be considered as a container for the relation.
2010	SO	The *Similarity Ontology (SO)* was proposed by Halpin et al. [68] and provides vocabularies for linkage of entities of different ontologies less strict than *owl:sameAs*, like *so:related*, or *so:claimsMatches*.
2017	Lenticular Lenses	*Lenticular lenses* were proposed by Idrissou et al. [225] and are a further development of scientific lenses from Batchelor et al. [226]. Lenticular lenses extend Vocabulary of Interlinked Datasets (VoID) [227] vocabulary to relate at entity level and enhance link sets with metadata like data about the matching algorithm. The concept of lenticular lenses is based on the idea that two related entities refer to a limited set of properties that identify a match and a limited set of properties that hold for the related entity.
2017	Identi-ConTo	Raad et al. [228] have developed the algorithm DECIDE aiming to reduce two graphs so that the owl:sameAs relation holds between corresponding entities. Like lenticular lenses, this approach is based on the idea to categorize properties of corresponding entities into those which hold for the corresponding entity and those which do not. The relation between entities of the reduced graphs is expressed by the expression *identiConTo*.
2019	OWL^C	Aljalbout et al. [217] propose OWL^C to limit the validity scope of graphs similar to the IST approach from Guha [219] but differ between validity context and additional context. OWL^C is developed as an extension of OWL.

In general, **research works about contextual linking** in the field of Semantic Web are initiated based on two aspects:

- First, corresponding objects of heterogeneous ontologies are generally not the same thing even though they refer to the same real-world object.
- Second, the ontologies are often aligned through the link predicate *owl:sameAs* which states that the linked objects “have the same identity” [229].

The consequent **issue** caused by both aspects heterogeneity of ontologies and strict semantics of owl:sameAs is commonly known in the field of Semantic Web: Is a ship before the replacement of some components the same as the ship after this replacement [230]? Is Tim Berners-Lee as a child the same as Tim Berners-Lee as an adult [68]? Is drug A the same as drug B when having the same structure but different vendors [68,225,226,228,231]? These examples illustrate that the use of *owl:sameAs* is prone to be misleading for linking heterogeneous ontologies.

Research works about contextual linking address the development of alternatives to *owl:sameAs* for linking heterogeneous ontologies created from different perspectives. In summary, these linking approaches can be **categorized** regarding the *Alignment level*, the *Context type,* and the *Linking approach* and are summarized in Table 3.6. Notably, some integration approaches explicitly refer to the term *context* while others do not. For example, C-OWL addresses the alignment of “contextual ontologies”, while EDOAL provides expressivity like other contextual linking approaches but do not explicitly mention the term *context*.

The **alignment level** refers to alignments at entity or model level. In the scope of information models, alignment at entity level means that instance-level or schema-level entities are linked to each other. For example, Simple Knowledge Organization System (SKOS) [67] and Similarity Ontology (SO) [68] refer to vocabularies describing the relation between two entities like *skos:exactMatch* or *so:related*. Alignment at model level means that the link relates whole models. For example, compatibility relations are vocabularies specifying the compatibility between two ontologies. Noteworthy, lifting and compatibility relations refer to the alignment of interpretations which is why they cannot clearly be assigned to entity and model level.

The reviewed publications address different **context types**, here called contextual model and contextual alignment. A contextual model means that the corresponding entities or models are heterogeneous and created from different perspectives. Contextual alignment means that the alignment depends on contextual variables like the perspective of the user. In other words, the validity scope of the alignment is limited to a specific situation. All investigated approaches deal with contextual models while few explicitly consider these models as contextual. The relation between alignment levels and context types is illustrated in Figure 3.8.

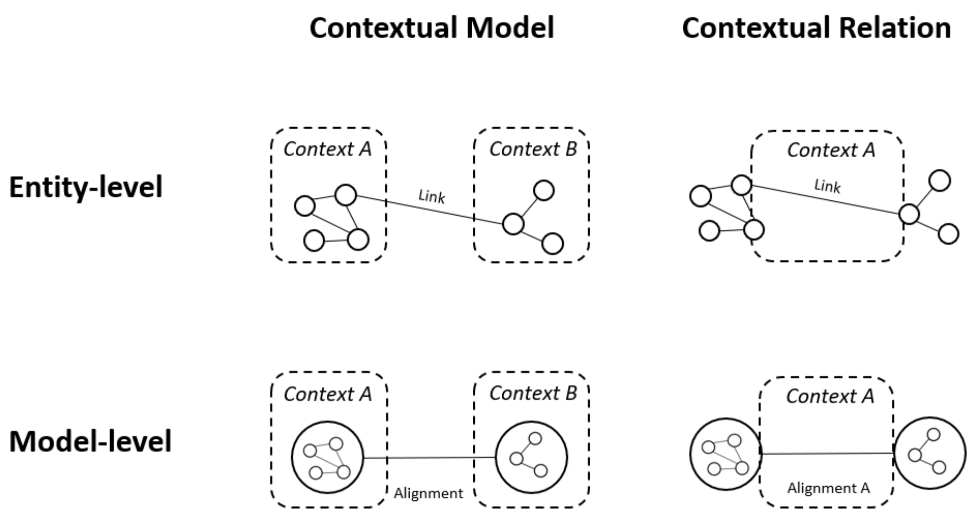


Figure 3.8: Illustration of the relation of alignment levels (entity vs. model level) and context type (contextual model vs. contextual relation)

There are two major **contextual linking approaches**, namely *alternative relation types* and *restriction of validity scope.* Alternative relation types can be further subdivided into *alternative link predicates* and *transformation rules*. Restriction of validity scope can be further subdivided into limiting validity scope in terms of a *restriction to specific situation* and in terms of *vocabulary restriction*.

- **Alternative link predicates** refer to contextual linking approaches providing alternative vocabularies to *owl:sameAs*. An example of alternative link predicates are vocabularies provided by Contextualized OWL (C-OWL) [218] which covers five different relation types, namely *disjoint* ($\perp$), *equivalence* ($\equiv$), *related* ($*$), *more abstract* ($\subseteq$), and *more specific* ($\supseteq$). Further alternative link predicates are provided by the Simple Knowledge Organization System (SKOS) [67] like *skos:exactMatch* and Similarity Ontology (SO) like *so:related* [68]. Furthermore, Raad et al. [228] and Idrissou et al. [225] propose to use alternative link predicates to *owl:sameAs* like *link:identiConTo* or *my:sameAs*.
- Some contextual linking approaches make use of **transformation rules** describing the relation between entities. An example is a transformation rule relating two properties of temperature based on Kelvin and Celsius which is, among others, supported by EDOAL [224]. Furthermore, the integration approach *lifting* defines rules to lift entities from one context to another. Noteworthy, compatibility relations [223] and OWL^C [217] refer to mapping rules. However, these mapping rules rather define criteria for mapping with respect to the interpretation.
- Guha [219] suggested restricting the validity scope with respect to a **specific situation** through the expression *ist* which means 'is true in' a certain context. OWL^C [217] adapts the concept of *ist* and uses two-dimensional description logics to make this kind of validity scope explicit. The contextual linking approaches *link:IdentiConTo* [228] and Lenticular Lenses [225,226] assign subgraphs or sets of links to specific situations respectively. C-OWL [218] and EDOAL [232] do not explicitly refer to this kind of validity scope but allow to define links for specific situations through so-called bridge rules and cells respectively.
- The restriction of the validity scope in terms of **vocabulary restriction** generally refers to the categorization of properties belonging to the related entities (i.e. following Leibniz's laws of identity of indiscernible). For instance, Raad et al. [228] and Beek et al. [228,233] limit the alignment to subgraphs so that an identity link becomes valid between related entities. Idrissou et al. [225] propose to categorize properties of related entities into those used as identity criteria and those valid for the related entity. Furthermore, EDOAL [224] provides rules to define the restriction of entities belonging to an alignment.

The described linking approaches often make use of approaches to enriching links/alignments with metadata. Similarly, these **metadata approaches** can be categorized regarding entity and model level (Table 3.6). Overviews about metadata approaches are provided by Frey et al. [234] and Sen et al. [235] who evaluate six and twelve metadata approaches respectively. One of the most common metadata approaches in the field of Semantic Web is Singleton Property [236], Named graphs [237], and RDF* [238]. Singleton property and RDF* are generally utilized to enrich single RDF statements, while Named graphs address metadata of whole graphs. Noteworthy, named graphs can also be used to enrich single RDF statements with metadata but this is not common practice. Furthermore, an approach to defining relations between datasets and enriching this relation with links is provided by the Vocabulary of Interlinked Datasets (VoID) [227] vocabulary. As an example, Idrissou et al. [225] use Singleton Properties and VoID to enrich links with information about matching methods so that the links become more detailed.

Table 3.6: Categorization of contextual linking approaches regarding level, context type and approach, and categorization of common metadata approaches regarding level.

		Contextual linking										**Metadata**			
		IST	C-OWL	Lifting	Comp. relations	EDOAL	SKOS	SO	IdentiConTo	Lenticular Lenses	OWL^C	VoID	Named graphs	Singl. Properties	RDF*
Level	Entity	x	x	(x)	(x)	x	x	x	x	x	x		(x)	x	x
	Model			(x)	(x)							x	x		
Context	Model	x	x	x	x	x	x	x	x	x	x				
	Alignment	x	x	x	x	x			x	x	x				
Approach	Link predicates		x				x	x	x	x					
	Transf. rules			x		x									
	Restr. to situation	x	x	x	x	x			x	x	x				
	Vocabulary restr.					x			x	(x)					

3.4 Summary

To the best of the author's knowledge, the literature review is the **most extensive** one in research about BIM-GIS Integration and covers 155 publications which were analyzed based on 1171 category assignments. In more detail, the publications referring to integration efforts were categorized with respect to the *document type, source model, integrated model, integration method, use case*, and *alignment*. The literature review related to the integration method and the use case is described in the chapter 5 and chapter 7.2.3 respectively.

In the beginning of the literature review, the **number of publications** related to BIM-GIS Integration per year was analyzed. In contrast to other literature reviews on BIM-GIS Integration, the respective diagram shows the publications related to the keywords CAD AND GIS beside the keywords BIM AND GIS. As a result, research related to the topic on BIM-GIS Integration reaches back to mid of 80s, and the number of publications has rapidly increased in the last decade. Furthermore, the number of publications per authors, number of publications per year and further aspects were analyzed to identify patterns in the reviewed literature. Based on the identified patterns four phases on research related to BIM-GIS Integration were categorized to structure the literature research. Nevertheless, these patterns are not considered as significant, but the consequent categories are good enough to structure the literature research in this dissertation.

Furthermore, the integration efforts referring to the integration method ***linking*** were analyzed regarding the link predicate and the related objects. There is no link predicate that is commonly utilized in information integration efforts in the field of BIM-GIS Integration. The links are often only described conceptually, or only specific objects are linked such as windows. Conclusively, a discourse about the adequate link predicates in the field of BIM-GIS Integration and the semantics of these links is missing. This kind of discourse is generally neglected since the link creator and the link retriever are often the same person, so the meaning of the link is clear to the link retriever, and the semantic problem on the meaning of the link is not relevant. Furthermore, linking approaches to BIM-GIS Integration are generally limited to specific instance models and a specific purpose so that the created solutions are often applicable for only a specific situation.

The reviewed publications about contextual linking in the field of Semantic Web have in common that they address the issue caused by the heterogeneity of ontologies and **strict semantics of *owl:sameAs***. However, these kinds of publications do not share a common understanding of contextual linking and use different terms. For example, some publications consider ontologies created from different perspectives as context, while other publications consider the creation of the alignment in different contexts.

In summary, the reviewed publications about contextual linking refer to **four different approaches** aiming to overcome the issue caused by the heterogeneity of ontologies and strict semantics of owl:sameAs.

- First, the linkage through alternative link predicates semantically weaker than identity links like owl:sameAs. This approach is accompanied by the advantage that they are less prone to be misleading. However, on the downside, alternative link predicates are often too weak for making respective conclusions without having knowledge about the linked objects.
- Second, transformation rules that are, without further ado, limited to numerical values.
- Third, restricting the validity scope of an alignment to a specific situation. This approach requires the definition of these situations and may result in many additional triples when dealing with several validity scopes.
- Fourth, restricting the validity scope through the categorization of properties specifying which properties of one object hold for the linked object. This approach requires the categorization of all relevant properties, and results in many additional triples when dealing with several validity scopes.

All reviewed contextual linking approaches provide reasonable concepts to overcome the issue caused by the heterogeneity of ontologies and strict semantics of *owl:sameAs* but are accompanied by several drawbacks.

4 Research problem & objectives

4.1 Research gap

The *research problem* arises from both the *research gap* identified in related literature and the *task problem* caused by the lack of knowledge required to accomplish an innovative task (Section 1.2). The **research gap** relevant for this dissertation emerges from the literature research on both BIM-GIS Integration and contextual linking in the field of Semantic Web.

In the **field of BIM-GIS Integration**, there are several research works aiming to link information models related to the domains BIM and GIS. Among others, these kinds of research works are limited regarding following two aspects:

- The proposed linking approaches are limited to certain situations so that they link specific instance models for a specific purpose.
- The proposed linking approaches use a variety of different link predicates but do not discuss their suitability in more detail.

For example, Stepien et al. [22] link several different models like city models and cadastral maps to an infrastructure alignment of a tunnel but the demonstration of the proposed approach is limited to specific instance models and specific purpose. Similarly, Zheng [208] link buildings to their closest road sections using Semantic Web technologies but demonstration of the proposed approach is limited to specific instance models and specific purpose. Both Stepien et al. and Zheng use some link predicate, such as topological links, but do not discuss their suitability in more detail.

The research works related to contextual linking in the field of Semantic Web has shown that the use of identity links such as owl:sameAs is not suitable for linking heterogeneous ontologies. Instead, several linking approaches are proposed as alternative to owl:sameAs such as introducing alternative link predicates or specifying the situation in which a link holds. Currently, there is no contextual linking approach outstanding from the others due to approach-specific flaws. For example, specifying the situation in which a link holds requires that the particular situations are made explicit. Here, the meaning of the term "situation" is not defined, and the dependence of the linkage on situational variables is not exploited.

In summary, the discourse about the dependency of the semantics of the link and the situation in which the link holds with the example of BIM-GIS Integration is a relevant research gap for both BIM-GIS Integration and contextual linking in Semantic Web and further referenced by the term *context-sensitive linking*.

4.2 Task problem

The **task problems** are based on the dependency of semantics of the link and the situation. In the following, exemplary task problems with respect to the referring example are illustrated through the perspective of link creator and querier (*Example 3* and *Example 4*).

In *Example 3*, the consequent task problems from the perspective of the link creator are illustrated. Here, the comparison of two integration scenarios are compared, while the integration scenarios differ from each other with respect to the used source models and the application. As a result, the different source models or application may require different links.

In *Example 4,* the consequent task problems from the perspective of the link retriever are illustrated. Here, the query is performed on the linked models without further knowledge about these information models or the semantics of the link. Querying linked heterogeneous information models without additional knowledge about the linked models or the semantics of the links may be prone for misinterpretations

Example 3: Task problem – Perspective of link creator

An IFC model represent the building as designed, while a CityGML model represent the building as built. A facility manager has the IFC model and wants two know the exact length of the beam without support length due to refurbishment measures (Figure 4.1).

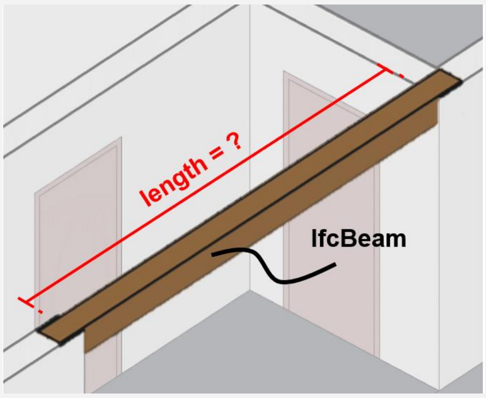


Figure 4.1: Missing length value which needs to be queried from CityGML model in Example 3.

A) Different variants

- **Situation A**

The corresponding beam objects of the CityGML and IFC model are matched based on spatial identification and schema level alignment, and the following SPARQL statement for alignment creation is generated.

```
[PREFIX]
INSERT {
  GRAPH ex:Alignment
  { ?ifcObject      ex:equivalenceLink     ?citygmlObject   .  }
}
USING ex:ifcModel
USING ex:citygmlModel
?ifcObject          ex:Id      "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
?citygmlObject     ex:Id      "ad128f7c-2ce6-4996-894d-7e7e0659575e" .
```

Afterward, the length of the beam object from the CityGML model is retrieved through the following SPARQL query.

```
[PREFIX]
SELECT ?length
FROM ex:ifcModel
FROM ex:citygmlModel
FROM ex:Alignment
WHERE {
  ?ifcObject          ex:Id      "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
  ?ifcObject          ex:equivalenceLink     ?citygmlObject    .
  ?citygmlObject    ex:Length                   ?length  .
}
```

Result: “2.03”^^xsd:double

- **Situation B**

Like in situation A, the corresponding beam objects of the CityGML and IFC model are matched based on spatial identification and schema level alignment, and the following SPARQL statement for alignment creation is generated.

```
[PREFIX]
INSERT {
  GRAPH ex:Alignment
  { ?ifcObject      ex:equivalenceLink     ?citygmlObject   .  }
```

```
}
USING ex:ifcModel
USING ex:citygmlModel
?ifcObject        ex:Id     "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
?citygmlObject    ex:Id     "694ff972-a884-4760-a495-67d5266e087e" .
```

Like situation A, the length of the beam object from the CityGML model is retrieved through the following SPARQL query.

```
[PREFIX]
SELECT ?length
FROM ex:ifcModel
FROM ex:citygmlModel
FROM ex:Alignment
WHERE {
  ?ifcObject        ex:Id               "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
  ?ifcObject        ex:equivalenceLink  ?citygmlObject   .
  ?citygmlObject    ex:Length           ?length  .
}
```

Result: "2.50"^^xsd:double

However, this time, the retrieved length value does *not* match the expected result. The reason for the different results comes from the fact that the two situations refer to the alignment to different variants of the CityGML model (Figure 4.2).

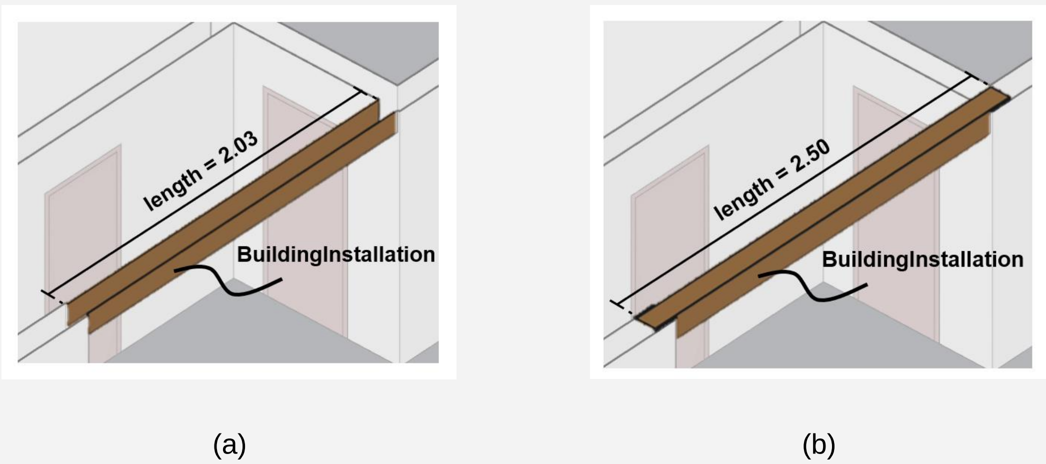


Figure 4.2: Retrieved length values of different variants of a CityGML model in Example 3A in (a) situation 1 and (b) situation 2.

B) Different use cases

Like in situation A, the corresponding beam objects of the CityGML and IFC model are matched based on spatial identification and schema level alignment, and the following SPARQL statement for alignment creation is generated.

```
[PREFIX]
INSERT {
  GRAPH ex:Alignment
  { ?ifcObject     ex:equivalenceLink     ?citygmlObject   .  }
}
USING ex:ifcModel
USING ex:citygmlModel
?ifcObject          ex:Id      "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
?citygmlObject    ex:Id      "694ff972-a884-4760-a495-67d5266e087e" .
```

Like situation A, the length of the beam object from the CityGML model is retrieved through the following SPARQL query with result 2.00 (Figure 4.3).

```
[PREFIX]
SELECT ?length
FROM ex:ifcModel
FROM ex:citygmlModel
FROM ex:Alignment
WHERE {
  ?ifcObject          ex:Id      "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
  ?ifcObject          ex:equivalenceLink     ?citygmlObject    .
  ?citygmlObject    ex:Length                    ?length  .
}
```

Result: “2.00”^^xsd:double

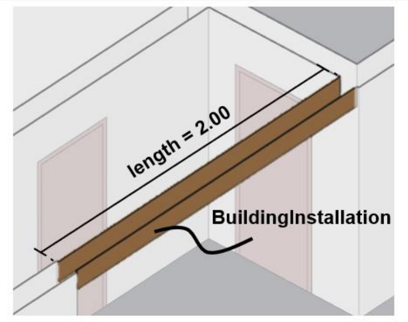


Figure 4.3: Retrieved length values of the CityGML model in Example 3B.

- **Situation A**

In the first situation the facility manager requires a rough estimation of the length value without support length to check whether the new furniture can be transported on a specific path. Thus, the length value about 2.00 m is accurate enough for the path analysis.

- **Situation B**

In the second situation, the facility manger wants to order new paneling for the beam so that the exact length value of the beam without support length is required. Thus, the length value about 2.00 m is not accurate enough for the task at hand.

Example 4: Task problem - Perspective of link querier

In this example, the IFC model and CityGML model are aligned in the beginning based on an RDF graph. From this RDF graph, the length of the IfcBeam object from the corresponding object in CityGML model is queried with the result 2.00. Here, the retriever of the information does not know to which variant of the beam object the length value refers.

```
[PREFIX]
SELECT ?length
FROM ex:ifcModel
FROM ex:citygmlModel
FROM ex:Alignment
WHERE {
  ?ifcObject        ex:Id        "fa86005f-4ca3-437b-83af-1d03f5d5be19" .
  ?ifcObject        ex:equivalenceLink    ?citygmlObject   .
  ?citygmlObject    ex:Length             ?length  .
}
```

Result: "2.00"^^xsd:double

4.3 Objectives

The objective of this dissertation refers to the creation of a solution to the research problem in terms of an artifact. Here, a solution to the research problem are **design principles** that support the implementation of information integration environments to overcome problems caused by the dependence of the link on the situation. In more detail, these design principles are intended to describe which *kind of linking mechanism* must be considered when facing these problems.

Design principles are **rules** specifying *how* a software product shall be implemented in certain situations. Rules are composed of premises and consequences. Thus, the objective of the dissertation is to develop premises and respective consequences:

- The *premises* specify the situation of an information integration environment in which the problems caused by the dependence of the link on the situation occur.
- The *consequences* specify the kind of linking mechanisms which must be implemented in this kind of information integration environment to overcome these problems.

Noteworthy, design requirements specify *what* to implement. Specifying *how* to implement in detail is closely related to specifying *what* to implement so that there is no clear boundary between design principles and design requirements [239].

For clarification purposes, **the objective is not** to develop linking mechanisms overcoming these problems. Moreover, the objective is not to develop an algorithm for automated matching, which might be a research topic associated with linking information in first place. Furthermore, the development of the design principles is limited to several aspects as defined in chapter 1.4.

5 Use Cases

5.1 Categorization approaches

The literature research has shown that there is no clear understanding of use cases related to BIM-GIS Integration. There are several publications with different approaches aiming to categorize **use cases** related to BIM-GIS Integration [5,110,114–116,120]. For example, Liu et al. [5] distinguish between 3D Cadastre, Location-Based Services and Navigation, Asset Management, Heritage Management, Site Selection and Layout Plan, Urban Environment Analysis, and Safety. As another example, Hijazi and Donaubauer [85] distinguish between Controlling and Monitoring Energy, Facility Management, 3D Spatial Analysis, Utility Networks and Building Service System, Construction Management, Emergency Management and Indoor Navigation. Further categorizations address the integration direction such as BIM leads and GIS supports, GIS leads and BIM supports, and BIM and GIS are equally involved [120]. In this dissertation, the use cases are categorized with respect to two aspects: First, the (physical) subjects describing the information addressed by the use case. Second, the Activity describing the use of this kind of information.

5.2 Physical subject

The *subjects* of a use case related to BIM-GIS Integration refer to both domains BIM and GIS. The subject of the domain **BIM** refers to a building which can be categorized with respect to its type. Here, building means a construction object and exemplary categories are following:

- Residential buildings
- Industrial buildings
- Transport infrastructure
- Utility infrastructure

Furthermore, the subject of the domain BIM refers to specific phases of the lifetime of the building. Thus, exemplary phases are the following:

- Building in construction phase
- Constructed Building
- Building in demolition phase

Building in construction or demolition phase generally refers to the site of the building, while constructed building can be the thought-ahead building in design phase or the existing building in maintenance phase. As an example, the objects of a BIM model representing transport infrastructure alignment could be integrated in a GIS model representing the geological data to calculate the mass of earth works. On the other hand, a site of industrial building could be integrated in a GIS model representing geological data to plan the site layout like the drainage.

In the domain **GIS**, the *subject* refers to georeferenced information. As an example, an infrastructure alignment could be integrated in a GIS model representing geological data to calculate mass of earth works or tree cadastre data to calculate the tree stock on that path, Further examples are following:

- Cadastre data
- City models (urban buildings)
- Zoning and land use
- Hydrography
- Soils

In both domains BIM and GIS, the model describing the subject can refer to different ***states*** of the model, such as following:

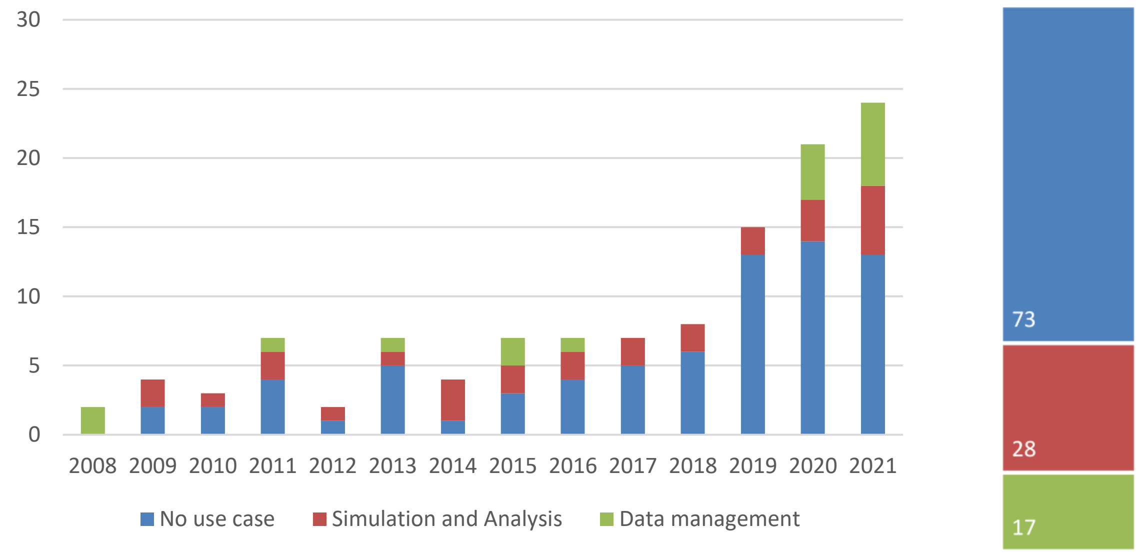
- planned subject
- existing subject
- former subject

A model representing the planned subject means that the thought-ahead subject is modelled, while a model representing the former subject means that subject ‘as-it-was’ is modelled. As an example, a BIM model representing a building is integrated to a city model to simulate its influence on the wind flow at an adjacent public square. This kind of BIM model could represent the planned building, the existing building, or the former building. Models representing former subjects are particularly relevant for heritage management which is not further addressed in this dissertation.

Furthermore, in both domains BIM and GIS, the model describing the subject can be represented in different levels of detail. As described in chapter 2.4, there are different types of Level of Details in the domain BIM and in the representation of CityGML models. The description of Level of Details according to the specification of CityGML is used to roughly estimate the detailedness of the model representing the subject.

5.3 Activity of use case

The use cases in the field of BIM-GIS Integration can be further categorized with respect to the activity. The *Activity* describes the use of the information relevant in the BIM-GIS Integration use case. Here, there are two major categories describing the activity, namely analysis and simulation, and data management. In the reviewed literature, most integration efforts do not refer to a specific use case or only relate to use cases in terms of a buzzword (62%). These kinds of integration efforts are represented in Figure 5.1 through the category *No use case.* There are 28 integration efforts (23%) referring to *Simulation and Analysis*, and 17 integration efforts (14%) referring to *Data management*.



(a) (b)

Figure 5.1: Number of publications in the related literature referring to the categories No use case, Simulation and analysis, and Data management (a) per year and (b) in absolute numbers.

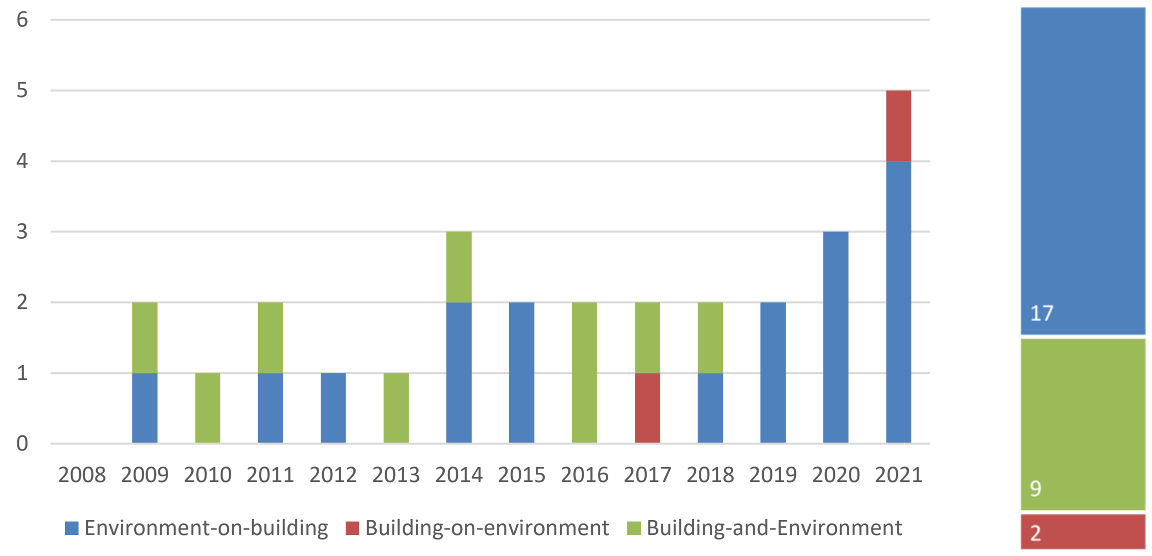
The *Analysis and simulation* means processing the information to satisfy the requirements of a task in an operative planning process. Use cases related to BIM-GIS Integration can be further categorized with respect to the direction of the simulation:

- The building influences the environment (*Building-on-environment*)
- The environment influences the building (*Environment-on-building*)
- The building and environments are equally involved (*Building-and-environment*)

The use cases related to category *Environment-on-building* majorly refers to tasks relevant for planning the building. For example, for analyzing the thermal insulation of a building, the thermal influence of the surroundings on the building needs to be considered. On the other hand, use cases related to the category *Facility-on-environment* majorly refers to tasks relevant for geospatial domains like urban planning. For example, analyzing the influence of a planned building on wind flow in an urban area.

Notably, Wang et al. [120] emphasize categorization which seems to be similar at first glance: BIM leads and GIS supports, GIS leads and BIM supports, and BIM and GIS are equally involved. However, the categorization from Wang et al. is based on the integration or conversion direction. For example, BIM leads and GIS supports means “to import or integrate data from the GIS model to the BIM model” [120]. In contrast to that, the proposed categorization in this dissertation addresses the direction of the simulation independent of the integration approach or direction.

The most integration efforts in the reviewed literature which is related to simulation and analysis refers to the category *Environment-on-building* (61%). There are nine publications in the related literature which address *Building-and-environment*, and only two which address integration efforts related to the category Building-on-environment (Figure 5.2).



(a) (b)

Figure 5.2: Number of publications in the related literature referring to the categories Environment-on-building, Building-on-environment, and Building-and-environment (a) per year and (b) in absolute numbers.

In Table 5.1 several use cases are described with respect to the categories and the publications about BIM-GIS Integration of the related literature are assigned to the respective use cases. Notably, visualization is here not considered as a use case on its own. This is because visualization serves for gaining insights for a task at hand but does not represent the task itself. For example, the evaluation whether the window in the living room of the planned building has lake view can be achieved through visualization or through mathematical calculations.

Furthermore, in Table 5.1 the Level of Detail required for the use case is roughly estimated. The required LOD also depends on the required accuracy of the analysis so that there may be several LODs assigned to a use case which are separated by the comma symbol. The '+' symbol indicates the minimum LOD necessary for the use case. For example, the required LOD for checking buildings against building codes depends on the type of building code. Querying the number of elevations which are not handicapped accessible in a certain district requires a higher LOD than querying the number of buildings whose height does not match with the building code. Notably, some models can be reduced in complexity so that a low detail model is enough for the respective use case. For example, checking the used insulation of buildings in an urban area does not require a model representing the insulation in detail but to enrich the buildings at low LOD level with data about the insulation.

Table 5.1: Exemplary use cases related to BIM-GIS Integration and related to the category Analysis and simulation. The use cases are categorized with respect to the analysis direction and the required detailedness is roughly estimated according to the Level of Details as specified in CityGML.

	Required LOD	
	BIM	GIS
Building-on-environment		
Shadow analysis Analyzing the influence of the shadow caused by building on its environment, such as - shadow of the planned high-rise on the adjacent buildings.	1,2	1,2
Wind flow simulation Analyzing the influence of a building on the wind flow such as - the influence of a new shopping center on the wind flow at the adjacent public square.	1,2	1,2
Microclimate analysis Analyzing the influence of a building on the microclimate such as - the influence of a large residential building on the urban microclimate. [145]	1,2	1,2
Water flow simulation Analyzing the influence of the building on the water flow such as - influence of the planned tunnel on the underground water flow. - influence of the drainage of the planned road on urban flooding. [158]	1,2	1,2
Ground settlements Analyzing the influence of the building on ground settings such as - the ground setting of the planned high-rise tower.	1	1,2
People flow simulation Analyzing the influence of the building on people flow such as - the planned pedestrian bridge on the access to a festival.	1,2	2
Environment-on-building		
Quantity estimations Analyzing the quantity estimations for the planned building, such as - calculation of the earthwork mass for a planned transport infrastructure alignment. [135,163,209,210,240]	0,1	0,1

<u>Site scheduling</u> Analyzing the scheduling for the planned construction site, such as - analyzing the traffic related to the access of the construction site.	1,t	1,t
<u>Thermal insulation analysis</u> Analyzing the thermal measures of surroundings on a building, such as - the influence of adjacent high-rise building on daylight simulations of the planned building - solar energy simulations [212]	1,2	1,2
<u>Sound insulation analysis</u> Analyzing the sound measures of surroundings on a building, such as - noise emissions of adjacent main road on sleeping rooms of adjacent residential buildings [139]	1,2	1,2
<u>Aesthetical design</u> Analyzing the influence of the surroundings on the aesthetics of a building, such as - the view from a window in the living room on the adjacent lake. - the view from the windows of adjacent building on the garden of the planned building.	3	1,2,3
<u>Wind loading analysis</u> Analyzing the influence of the wind loading to the building such as - the urban wind flow on the high-rise building to calculate wind loading - the wind flow at the sea to calculate energy potential for planned wind turbines	1,2	1,2
<u>Geological analysis</u> Combined analysis of the building and geological measures such as - analyzing the grounding of the building through visualizing both building and boreholes in one view. - analyzing the geological surroundings of a site with respect to the excavation safety. [201]	1,2	1,2
<u>Ground settings analysis</u> Analyzing the influence of the ground settings on the building, such as - the influences of the ground settings caused by the planned high-rise on the structure of the adjacent buildings.	1	1,2
<u>Building codes</u> Check weather planned building fulfills building codes in respective environment, such as	1+	1,2

- analyzing the setback distance of the planned building to the street. [203] - checking the overhang of a building to the maximum limits. [161]		
<u>Conflicts detection</u> Analyzing whether the building spatially conflicts with its surroundings, such as - the identification of the buildings or cadastral zone above the planned tunnel alignment [22] - analyzing whether the time schedule for planned transport infrastructure alignment conflicts with time restrictions of spatially related environmental zones (like incubation restriction zones) [186] - analyzing the amount and type of trees on the planned transport infrastructure alignment - analyzing whether the planned tower cranes conflict with the adjacent buildings [17,127,171] - analyzing whether the planned workspace conflicts with surroundings [241]	1,(t)	1,2
Environment-and-building		
<u>Path planning</u> Analyzing the shortest path related to both building and its surroundings such as - combination of indoor and outdoor paths to support navigation [20,129,189,195,196]	3*	1*
<u>Building codes</u> Check whether buildings fulfills building codes in urban environment, such as - new public standards to building insulation in specific area. - distance of Kindergarten to street [208]	1+	1+

* = possibly alternative model, t = time, + = min LOD

In particular, the most common use cases are conflict detection such as the identification of the geometrical conflict of an infrastructure alignment with cadastral zones, and path planning requiring the combination of indoor and outdoor paths. Except of the use cases *building codes* and p*ath planning*, all the listed use cases require building information at detail level less than LOD3. Consequently, the integration of the hull of the building, such as the alignment of a tunnel or the outer shell of the residential building, to the respective environment is often enough for the simulation or analysis. The use case *Building codes* may require more detailed building information and the use case *Path planning* for combined indoor and outdoor paths requires at least spatial indoor information.

The use case category *data management* refers to data consistency measures with related computer-based information systems such as the data transfer to asset management systems. The need for information integration result from that fact that the required information needs to be transferred to the information system adequate for the respective analysis. For example, indoor navigation could be performed based on a BIM model, but an information model based on network description might be rather adequate. The use cases of the reviewed literature belonging to this category are represented in Table 5.2.

Table 5.2: Exemplary use cases related BIM-GIS Integration and related to the category Data management.

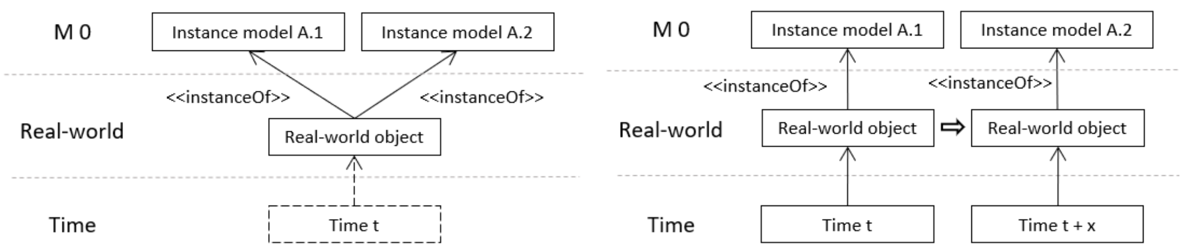
<u>3D property cadastre</u> - import BIM models to a 3D property cadastre system [170,194,242]
<u>Indoor property cadastre</u> - import BIM models to system for indoor property cadastre. [176]
<u>Supply chain monitoring</u> - monitoring the supply chain of building products of the construction site at global scale. [128,133,190]
<u>Asset and Facility management</u> - importing information about building products to asset management system which is based on GIS [80,159,200]
<u>Indoor navigation</u> - creating a system for indoor navigation based on GIS and import data to this kind of system. [106,140,166,188,206,243] - creating system for construction site navigation and import data to this kind of system.[204]

6 Instance-level heterogeneity

To understand the difficulties when linking heterogeneous models at instance level, one needs to understand the differences between instance models. To define the differences between instance models, the concepts **variants and versions** in the scope of information modeling are introduced.

- Variants are alternative representations of a real-world object.
- Versions are different representations of a real-world object at different times.

In information modeling, the object which is represented is a real-world object. Thus, two variants are here two alternative instance models both representing the same real-world object, and two versions are two different instance models both representing the same real-world object at different times *[244,245]*. For example, an IFC models representing a wall as “bearing” is a different variant than an IFC model representing the same wall as “non-bearing”. Furthermore, a CityGML model that represents the tree cadastre of a city is a different version after adopting latest landscape measures than before this kind of adaption.



(a) (b)

Figure 6.1: (a) Different variants and (b) different versions representing the same real-world object. Variants might be limited to the representations of a real-world object at a specific time.

In BIM-based design or planning two variants are often considered as two instance models representing thought-ahead alternatives to the design or planning problem. Here, the represented object is the solution to a problem, and the use of the term variant refers to the domains design and planning, but not to the domain information modeling. Furthermore, the term *version* often refers to different instance models resulting from simultaneous, distributed modeling operations [246]. In the following, the terms variants and versions are used in terms of information modeling as defined above.

The informational distance or the difference between the two variants or versions is here called instance-level heterogeneity. In the field of computer science, there is no uniform understanding about the term heterogeneity so that different terminologies are used, such as *model differences* or *data conflicts*, and there are different **categorizing approaches**. For example, Euzenat et al. [35] emphasize that there are a variety of different categorization approaches to heterogeneity and refers to more than 21 publication.[3] Two types of categorizations based on different approaches are described below, namely those from Stuckenschmidt [33] and Batini et al. [247]. Afterward, heterogeneity is defined for the scope of this thesis based on these categorization approaches.

- Common categorization approaches of heterogeneity differ between syntactical, structural and semantical heterogeneity or similar categories. Following Stuckenschmidt [33], structural conflicts are caused by different representations of equivalent information at schema level. Examples are the representation of some information as class versus as attribute or the representation of some information as "float" versus as "integer". Semantical conflicts are caused by different intended interpretations on the attribute or class. For example, a geometrical attribute can be expressed through different measurement units. As another example for semantical conflicts, the covered objects by the class wall may depend on the user's interpretation. A structural engineer might see the vertical load bearing as characteristic for a wall while an architect may refer to rather optical characteristics.
- A categorization approach focusing on instance-level heterogeneity is provided by Batini et al. [247] who address data quality and integration with respect to databases. Batini et al. differ between *schema heterogeneities* and *instance-level heterogeneities*. Schema heterogeneities may be caused by the use of different data models like XML versus relational data model, or structural differences as described above. Instance-level heterogeneities refer to different, conflicting data values due to different data sources and are caused by quality errors like accuracy or currency.

The categorization approaches similar to the one of Stuckenschmidt generally do not explicitly differentiate between *schema-level* and *instance-level heterogeneity* according to the **levels of the MOF**. The categorization approach of Batini et al. considers solely instance-level heterogeneities caused by differences at schema level (*schema heterogeneities*) and caused by quality errors (*instance-level heterogeneities*). Instance-level heterogeneities are of major interests in this dissertation while schema-level heterogeneities are considered rather as a source of instance-level heterogeneities.

The categorization approaches of heterogeneities have in common that they refer to the *sources of instance-level heterogeneities*. For example, structural differences, different data sources and quality errors are sources for instance-level heterogeneities. Thus, the categorizations approaches of heterogeneities often do not describe the actual ***nature of the instance-level heterogeneities***. Here, nature of instance-level heterogeneities means the characteristics of the heterogeneities relevant for the integration of the information models. For example, can the difference between two

[3] Some authors compare information models or ontologies based on *interoperability levels* what is also emphasized by Euzenat et al. [35]. In this dissertation the use of the concept interoperability with respect to information models is critically considered since information models do not "operate" with each other.

objects be bridged through some transformation rule? Is the one of the conflicting information correct while the other is not, or do the objects represent different perspectives? In the following, instance-level heterogeneities are categorized with respect to its nature and exemplary sources for these kinds of instance-level heterogeneities are described based on the categorization approaches from Stuckenschmidt and Batini et al.

- *Contradicting data* which means that one data value is correct while the other one is not.
- *Transformable data* which means that the data can be bidirectionally transformed without information loss.
- *Contextual data* which means that the data is not bidirectional transformable without information loss. Furthermore, the data values are not considered correct or incorrect in an absolute sense but are created from different perspective or based on different data sources.

Contradicting data may occur due to errors in parallel data management like contradicting entries in two information models about the material encoding of a building object (*quality errors*). Examples for transformable data are attributes based on different measurement units, like Kelvin and Celsius (*semantical heterogeneity*). Another example are different geometrical representations which are bidirectional transformable (*structural heterogeneity*). Examples for contextual data are accuracies of length values due to different data acquisition methods (different data sources),or different expectations on the used vocabulary (semantical differences).

In summary, the differences between variants and versions representing the same real-world object are described by instance-level heterogeneities, while aspects like differences of the underlying schema models, different data acquisition methods, quality errors are the sources for instance-level heterogeneities. For example, different variants may occur due to instantiations of different schema models or instantiations based on different data acquisition methods. Variants and versions and their relation to same and different schema models with respect to MOF are represented in Figure 6.2. In this dissertation instance-level heterogeneity in terms of contextual data is of major relevance and further addressed by the concept *contextual data differences.* Instance objects which are based on contextual data differences are further called *contextually different*.

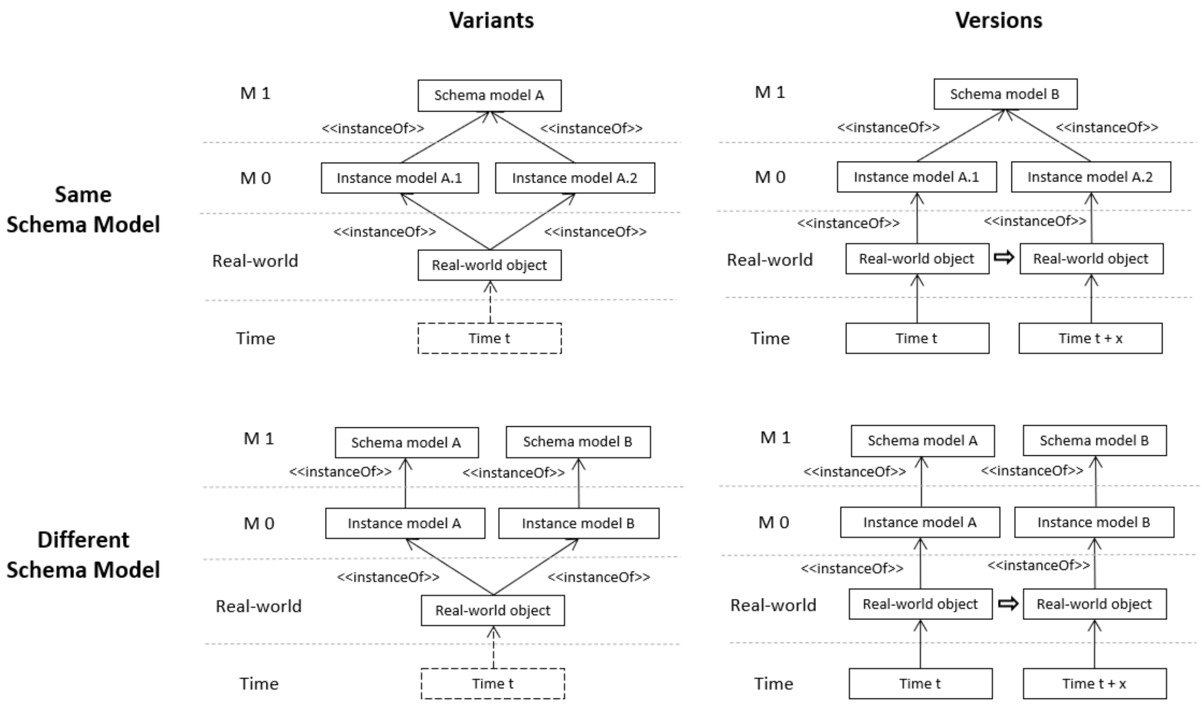


Figure 6.2: Variants and versions of the instance model based on the same schema model and different schema models. Versions might refer to both instance-level heterogeneities due to temporal aspects and due to different alternative representations (variants).

7 Information integration systems

7.1 Computer-based information system

Both methods BIM and UIM share and process information through ***computer-based information systems***. In general, a computer-based information system is considered as a socio-technical system that aims to support operational tasks through technological resources [248–250]. In the following, two interpretations of computer-based information systems are described that are most relevant for further research work, namely those from Ropohl [248] and Herden and Zwanziger [249]. The interpretation of Ropohl is often cited in expert literature when referring to foundations of general system theory while Herden and Zwanziger address integration subjects of computer-based information systems in the scope of business informatics. This section is not intended to provide an exhaustive discussion about computer-based information systems but to describe the fundamentals of computer-based information systems relevant for the further discourse about *BIM-GIS Integration*.

In more detail, Ropohl [248] describes foundations in general system theory with a focus on technological systems and illustrates these foundations using the example of computers. Following Ropohl, the meaning of the concept *system* can be briefly illustrated through three **conceptual perspectives**: *functional*, *structural*, and *hierarchical* concepts (Figure 7.1).

- The *functional concept* associates the system with a “black box” which has an input, output, and a function transforming the input to an output.
- The *structural concept* focuses on the relations between the systems whereby the relation of the systems refers to the flow of energy, material or information.
- The *hierarchical concept* addresses the hierarchy of a system whereby a system covers subsystems and is part of some super system.

These three **conceptual perspectives** complicate the discourse about *systems* because the perspective is often not specified during the discourse. An approach representing these conceptual perspectives through a single illustration is provided by Schmid [251] who has utilized system theoretical foundations to optimize façade structures (Figure 7.1d).

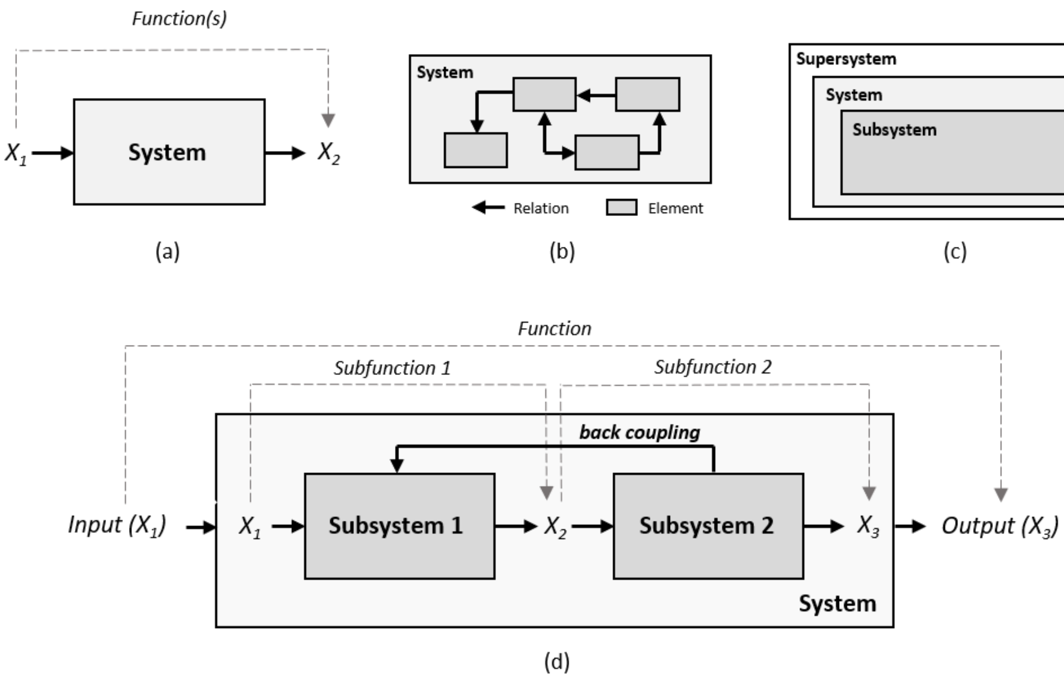


Figure 7.1: (a) Functional, (b) structural, and (c) hierarchical concept of a system following Ropohl [248] and (d) simplified representation of the combined representation following Schmid [251].

A **system** can be described in terms of both *socio-technical system* and *behavioral system* (original: *Verhaltenssystem*) [248]:

- A ***socio-technical system*** is composed of both social and technological subsystems, like user and computer respectively (Figure 7.2a). The interaction between these subsystems is based on the concept of labor division between computer and human. This kind of labor division can occur when functions of the technical system are equivalent to functions of the social system.
- A ***behavioral system*** is composed of three major subsystems, namely *objectives system* (original: *Zielsystem*), *information system,* and *execution system* (original: *Ausführungs-system*) (Figure 7.2b). Roughly spoken, the *objectives system* defines the objectives for the consequent actions. The *information system* addresses the utilization of information. The *execution system* refers to the actual 'working' system.

A **computer-based information system** refers to the *information system* of a behavioral system where the functions of this behavioral system are distributed between computer and human. The technological cores of the information system are the *processing system* (original: *Verarbeitungs-system*) and the *information storage system* (original: *Informationsspeicherungssystem*). The processing system evaluates and transforms information and defines the commands for the execution system. The information storage system provides the information in structured representation formats which Ropohl calls *internal model*. The information system of a computer-based system is limited to the flow of information while the execution system addresses the flow of energy and material.

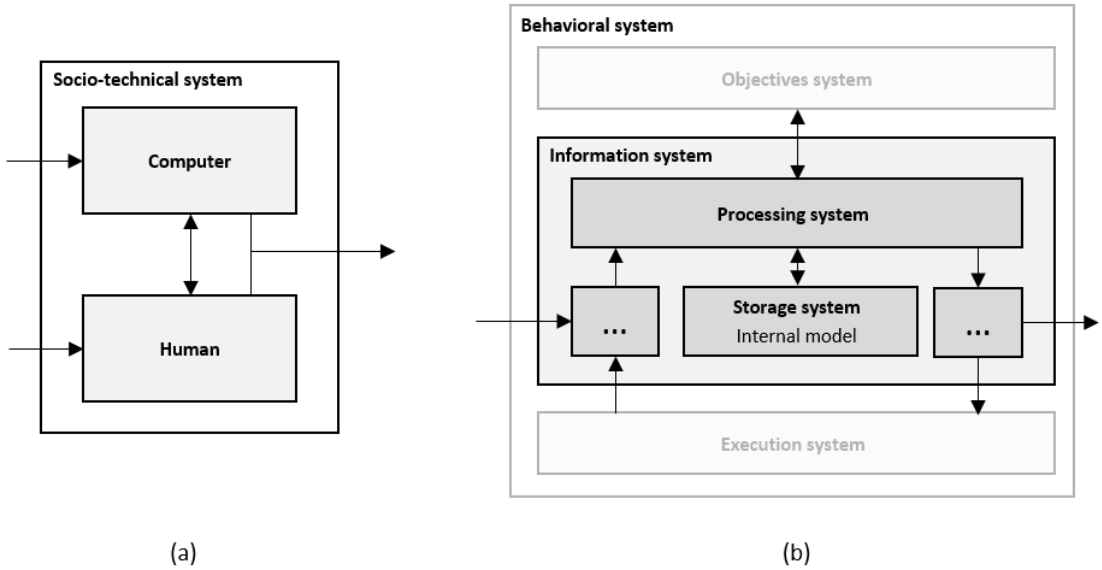


Figure 7.2: Computer-based system represented as (a) socio-technical system and (b) behavioral system.

Herden and Zwanziger [249] describe the **integration subjects** of a computer-based information system from the perspective of business informatics (Figure 7.3). Following Herden and Zwanziger, integration is the combination of corresponding subsystems of different computer-based information systems to a single entity. The respective subsystems are called integration subjects. Integration subjects can be categorized vertically into *organizational perspective* and *technical perspective*, and horizontally into *procedural level* and *structural level*. The integration subjects are *data* from a technical perspective and *objects/subjects* from an organizational perspective. The integration subjects at the procedural level are *program* from a technical perspective and the *process* from an organizational perspective. *Program* refers to the executable sequence of functions. Analogous, the term *process* refers to the logical, temporal sequence of tasks.

In the following, the two interpretations of Herden & Zwanziger and Ropohl are briefly **compared** to each other to relate the interpretations for further discourse. The interpretations correspond to each other in several aspects: Among others, both interpretations subdivide computer-based systems into *technical* and *social system* (Ropohl), or *technical* and *organizational layer* respectively (Herden and Zwanziger). Furthermore, the integration subject *application* following Herden and Zwanziger is similar to the concept *processing system* according to Ropohl and both interpretations state that a function transforms input to output. On the other hand, the interpretations differ from each other: Among others, Ropohl considers *information* as input and output of a subsystem while Herden & Zwanziger represent *data* as a structural component in terms of an integration subject. Similarly, while Ropohl describes the function as an integral part of each system, Herden and Zwanziger consider the function as a component of the integration subject *program*.

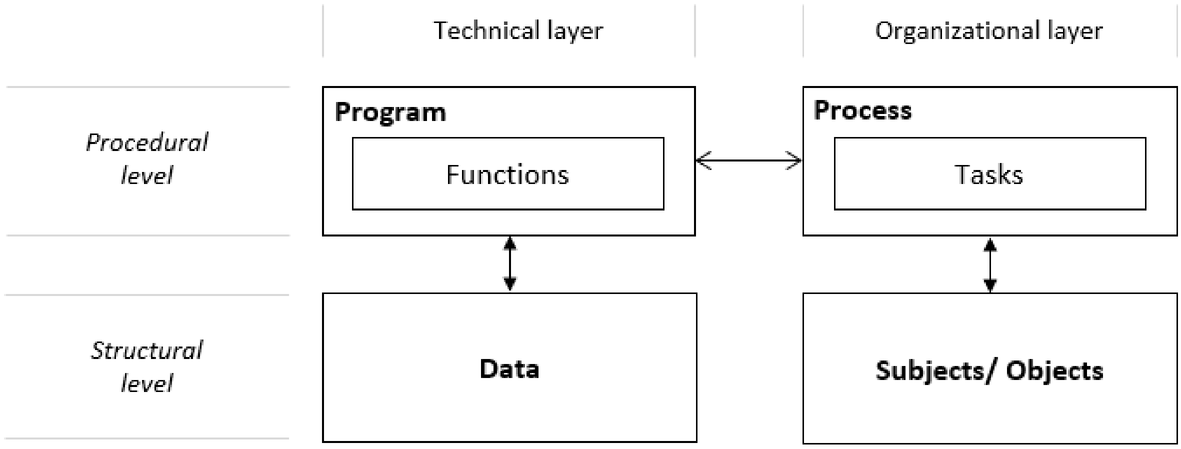


Figure 7.3: Integration subjects of a computer-based information system according to Herden and Zwanziger [249].

In this dissertation, the concepts ***application*** **and** ***information model*** are introduced and adopted to the interpretation of Ropohl and Herden, and Zwanziger. Here, the reasons are twofold: First, both concepts are commonly used in the field of BIM-GIS Integration and relevant for further discourse about information integration. Second, the introduction of these concepts aims to bridge some gaps between the interpretations of Ropohl and Herden and Zwanziger to make them consistent in the scope of this dissertation.

- The concept *application* refers to both subsystem of the *processing system* according to Ropohl and the integration subject *program* according to Herden and Zwanziger. An *application* covers subsystems whose output directly addresses the fulfillment of the objectives coming from the objectives system. More roughly spoken, an application refers to a set of functions relevant for some use cases, whereby a function transforms some information from input to output. The requirements on the input are further called information requirements. These information requirements define the relevant information for the application specifying what and how the information must be represented.
- The information relevant for an application is formally represented through *information models*. Here, an *information model* represents the *input/ output* within an information system according to Ropohl and refers to the integration subject *data* according to Herden and Zwanziger. An information model supports the functions of the application for which it is created.

The concepts *application* and *information model* are further illustrated through **two models**, which will be further referenced throughout this dissertation:

- The first model represents the technical processing system of a behavioral model and is primarily based on the interpretation from Ropohl (Figure 7.4).
- The second model represents the integration subjects at technical layer of a computer-based information system following the interpretation from Herden and Zwanziger.

The second model is represented in terms of integration subjects (Figure 7.5a) and set theory (Figure 7.5b), whereby an application refers to a set of functions and an information model refers to a set of information supporting these functions.

As an **example**, the authoring tool Revit [252] is composed of both an internal information model and an application. The application refers to a set of functions and the internal information model is designed to support these functions (see second model). The functions process the internal information model in terms of input and output of the processing system, like modeling operations creating or modifying a building object (see first model). The application and information models are integration subjects since they can be integrated with other software products through adopting Revit functionalities or enabling data exchange with Revit.

Noteworthy, the integration subjects as conceptualized in Figure 7.3 correspond to publications in the field of BIM-GIS Integration stating that there are three levels of integration: application, process, and information/ data level *[2,5,85,119,192]*. Interestingly, the integration at process level is interpreted differently by some of these publications since they consider web-based integration methods as process-level integration *[5,119]*.

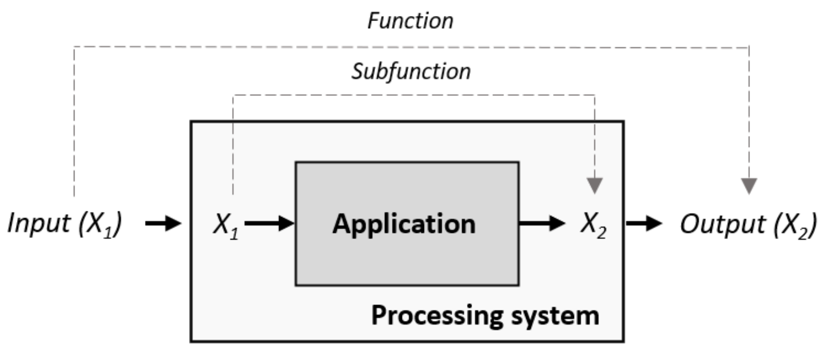


Figure 7.4: Application as subsystem of a processing system with respective input, output and functions.

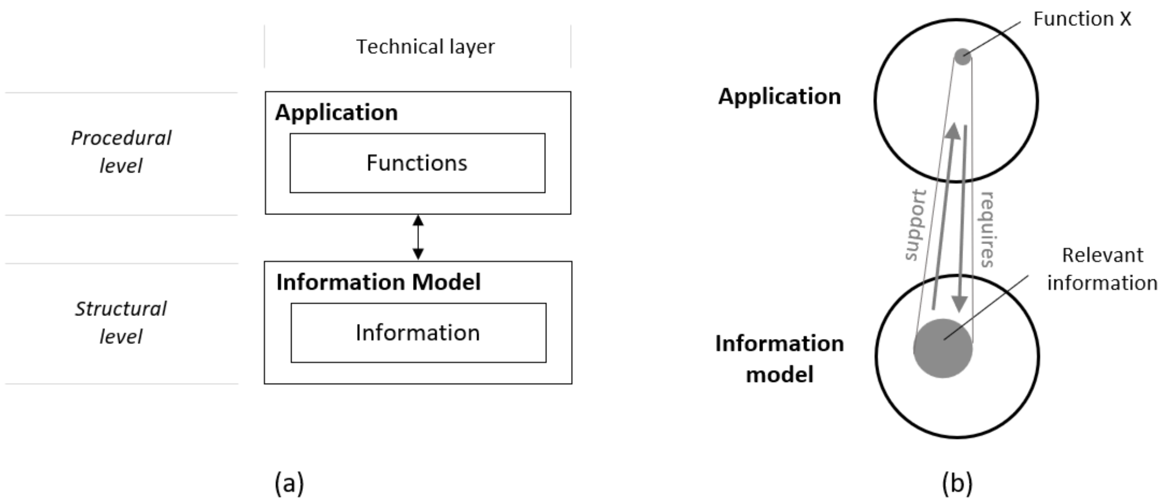


Figure 7.5: (a) Application and information model as integration subjects at technical layer of a computer-based information system. (b) Relation between application and information model represented through Boolean sets.

7.2 System structure

7.2.1 Overview

The term ***information integration*** is not commonly defined in the field of BIM-GIS Integration or research fields related to computer science like ontology engineering or data warehouses. On one hand, this is because different research fields use different terms partially synonymously for information integration like data matching or record linkage. On the other hand, neither the meaning of *information* nor the meaning of *integration* is well defined in the scope of information integration. In other words, neither the integration subject specifying *what* is integrated nor the integration method specifying *how* it is integrated is well defined. While some publications refer to the integration of databases, others deal with the integration of ontologies. While some publications limit the meaning of the term *integration* to the combination of one information model to another one, others solely address the linkage of two information models. These different understandings complicate the discourse about information integration.

Here, information integration is achieved through transforming one or more source model(s) to an integrated model with the aim to fulfill the information requirements coming from an application. Thus, an information integration effort addresses two interrelated subsystems of a processing system (Figure 7.6): First, the *integration system* that transforms the source information (input) to the integrated information (output). Second, the *application system* that transforms the integrated information (input) to the application result (output). This kind of processing system is further called ***information integration system***.

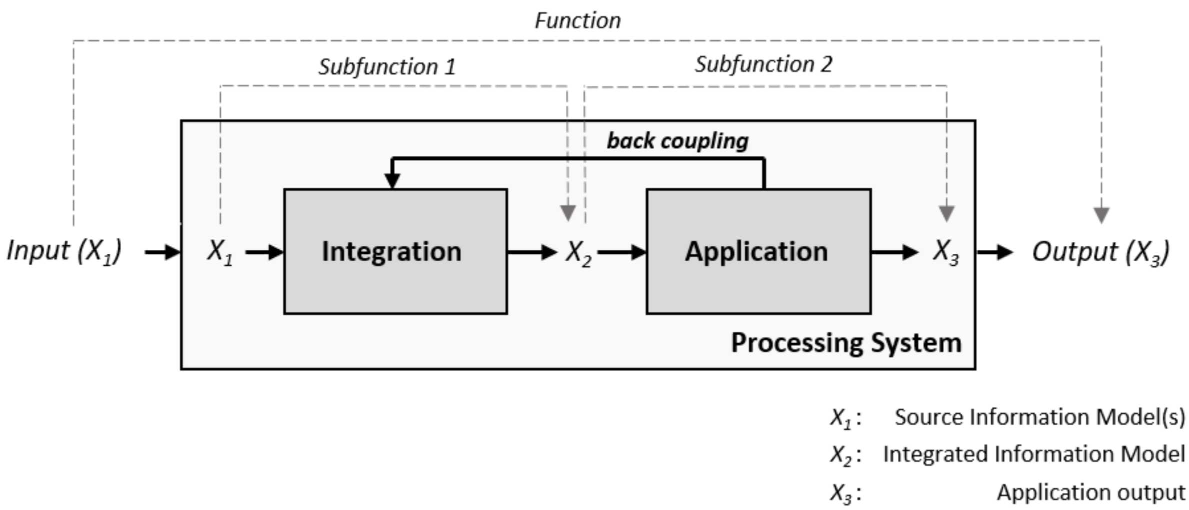


Figure 7.6: Information integration system as processing system which is composed of integration subsystem and application subsystem.

The integrated information model is intended to meet the **expectations** coming from an application**,** such as the information requirements. The information requirements refer to both an output of the application system and an input of the integration system. Thus, the expectations refer to the *back coupling* aspect relating the two systems (chapter 7.1). An information integration system that is implemented without considering the expectations of an application is called *incomplete*. Interestingly, the research on the use cases in the field of BIM-GIS Integration in chapter 5 has shown that the major number of integration efforts do not refer to a specific use case and are, therefore, either incomplete or the information integration system is insufficiently described.

In summary, there are two types of information that need to be transferred between integration and application system: the integrated information from the integration to the application system, and the expectations from application to integration system. This kind of information transfer is further referenced by the concept ***communication system***. The three components of an information integration system, namely *integration*, *application,* and *communication system* are described in more detail in the following chapters.

7.2.2 Integration system

An integration system addresses the transformation of one or more source model(s) to an integrated model with the aim to fulfill respective information requirements. In Figure 7.7, the source model(s) are represented through the symbol X_1 and the integrated model is represented through the symbol X_2.

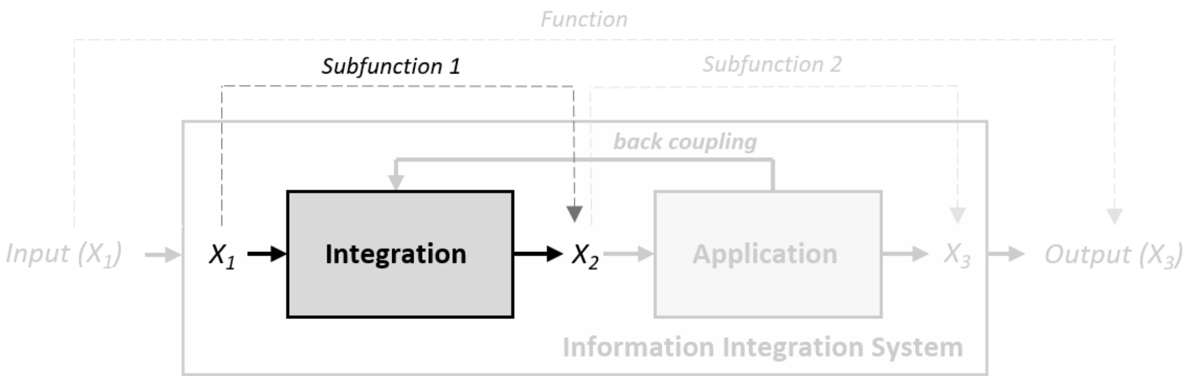


Figure 7.7: Integration system transforming the source model(s) to an integrated model through a function.

The purpose of information integration is often **software product interoperability** through the integration of respective internal models. The integration of the internal models is achieved either directly or through intermediate steps like the transformation of the information to external models (Figure 7.8). Here, each interface between two models, external or internal models, refers to an information integration process for itself.

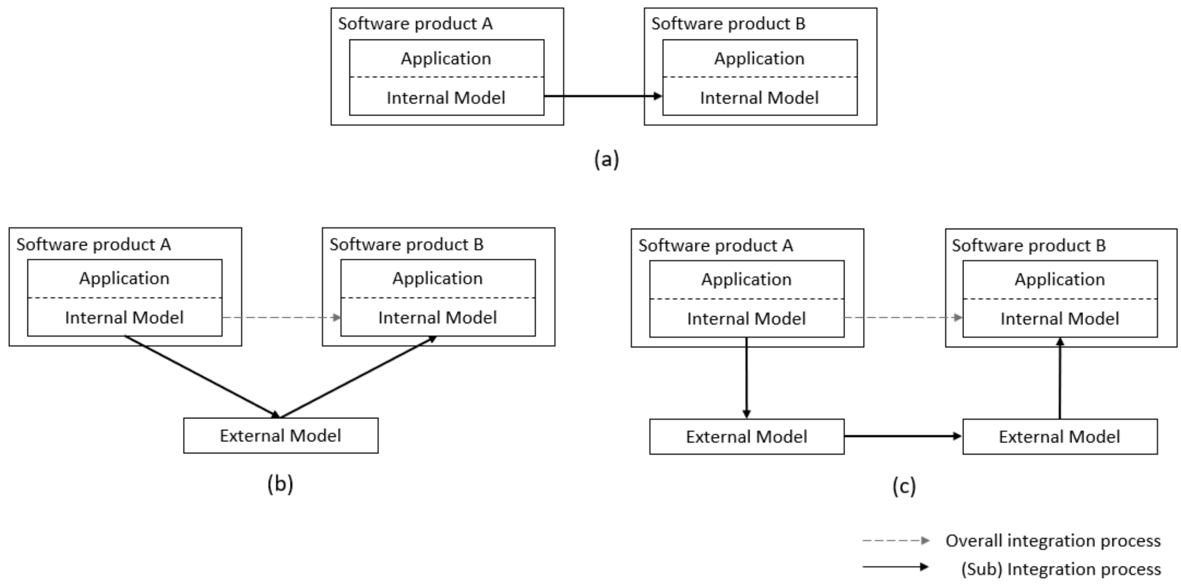


Figure 7.8: Common information integration approaches in the field of BIM-GIS Integration aiming to enable software product interoperability: (a) integrating internal models directly; (b) integrating internal models using an external model; (c) integrating internal models using two external models. The figure is adapted from Beck et al. [3].

Information models are integrated by means of an **integration method** at the instance or schema level. In the field of BIM-GIS Integration, the most common integration methods are *conversion*, *extension*, *merging*, and *linking* (Figure 7.9) [2,11,85].

- *Conversion* means the conversion or transformation of one information model to the other information model. Common examples are Extract Transform Load (ETL) processes, where relevant data is extracted, syntactically transformed, and afterward loaded to the respective application. In the field of BIM-GIS Integration, common examples are the conversion of IFC to CityGML models [24,90,146], or the conversion of IFC to internal models of a proprietary software product like ArcGIS [127,139,172]
- *Extension* means the conversion of an information model to another information model which is extended at schema level. The extension of the information model aims to prevent information loss in case the required information cannot be represented in the original target model. Examples are the extension of CityGML through Application Domain Extension (ADE) such as the Utility Network ADE [253] or IfcADE [106].
- *Merging* refers to the combination of one or more source models to one merged information model. The integration method *merging* also refers to integration efforts that create a new schema model with the purpose to represent information from two source models. Examples are the creation of a new schema model like Unified Building Model (UBM) [125] or Semantic City Model [24].
- *Linking* refers to the linking of corresponding entities at schema or instance level. The resulting integrated model is composed of both source models and the set of links. Examples are provided by Zheng [208] and Stepien et al. [22] who link infrastructure alignments to their surroundings using Semantic Web technologies.

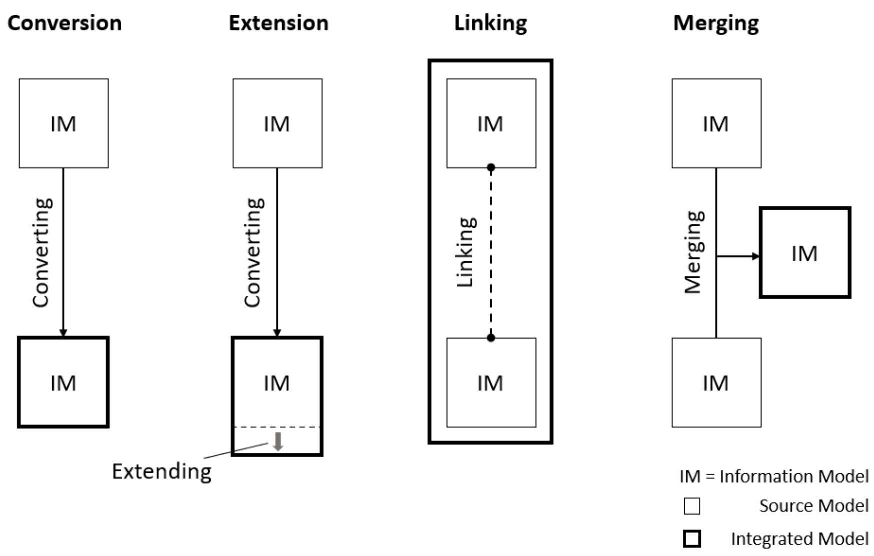


Figure 7.9: Common information integration methods in the field of BIM-GIS Integration, namely conversion, extension, linking, and merging. The figure is adapted from Beck et al. [2].

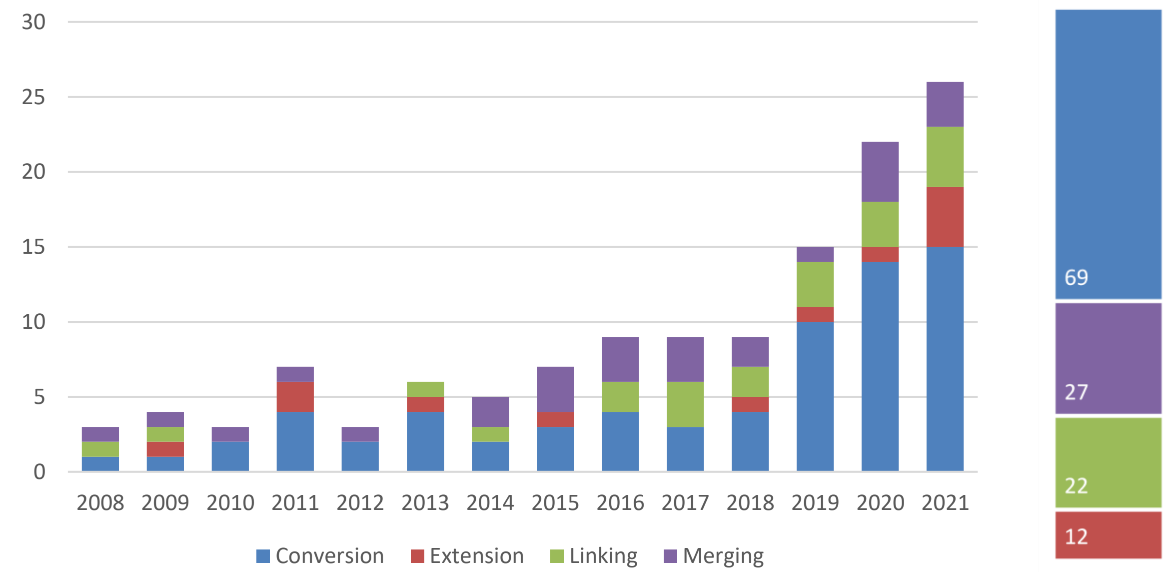
To demonstrate the validity of the categorization of the integration methods, the **reviewed literature** about information integration efforts in the field of BIM GIS is categorized with respect to these integration methods. An information integration effort can refer to a combination of different integration methods at instance and schema level. For example, merging at schema level can support the linking or conversion at instance level [2,89]. However, a clear differentiation concerning schema- and instance-level is in some cases difficult since information integration efforts at instance level often implicitly deal with integration at schema level. For example, schema-level alignments are often used as matching criteria for instance-level alignments, also when they are not explicitly formulated. Integration efforts that use two different integration methods at schema and instance-level were assigned to both integration methods. For instance, some integration efforts merge at schema level and convert at instance level [2,89]. Furthermore, some integration efforts could not be assigned to any integration method since they do not describe the integration method clearly enough.

- **Conversion** is the most addressed integration method in the reviewed literature (69 publications). From 2019 there is a high percentage of integration efforts using conversion, while the first integration efforts from 2005 and 2006 belong to merging and linking respectively. Among others, often cited publications referring to conversion are from El-Mekawy et al. [89,125,126] who address bidirectional conversion approaches and emphasize their scientific relevance. Furthermore, Tauscher [92,95,96] is an often cited author who introduced the conversion based on triple-graph-grammar. Noteworthy, several publications solely address the information exchange between proprietary software products using their native interfaces which are generally contributions of minor relevance to the scientific discourse about BIM-GIS Integration using conversion.
- The **extension** for BIM-GIS Integration is the category with the fewest number of publications (12 publications). The most integration efforts belonging to *Extension* address the

extension of CityGML through an ADE to support IFC. Often cited publications about CityGML ADEs are from De Laat and van Berlo [178], Hijazi et al. [179]. Other integration efforts address the extension of IFC or the extension of CityGML. Integration efforts about extension often propose an extended schema model without applying it for instance-level integration.

- There are 28 integration efforts using the integration method **merging** in the review literature. One of the most cited integration approaches belonging to *Merging* is provided by El-Mekawy et al. [124–126] who have developed the so-called Unified Building Model to support bidirectional conversion of IFC and CityGML. Like the extension, the integration method merging is often applied at schema but not at instance level.
- The integration method **linking** is addressed by 22 publications in the reviewed literature. Within the review publications, there is no researcher with more than two publications. Nevertheless, Pauwels [18,211] and Beetz [208,254] refer to two publications each and are commonly cited researchers in the field of AEC with respect to the topic linked data. Relevant publications about linking in the field of BIM-GIS Integration are from Zheng [208] and Stepien et al. [22] since they demonstrate the applicability and utility of the linking method with instance models.

In total, 117 publications were categorized regarding their integration method out of 122 referring to the category *Integration effort*. These 117 publications refer to 130 assignments to these integration methods.



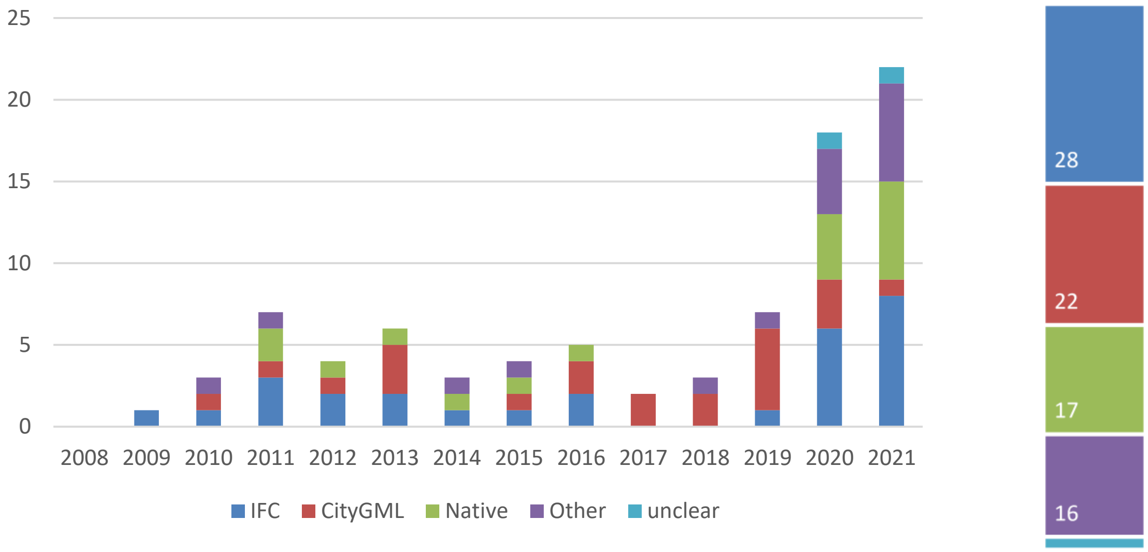
(a) (b)

Figure 7.10: Number of publications related to integration methods conversion, extension, linking or merging (a) per year and (b) in absolute numbers.

Furthermore, the information integration efforts were categorized with respect to the information models with respect to both **source and integrated models**.

- For integration efforts on **conversion**, the most common approach is the conversion of CityGML to IFC or vice versa (Figure 7.11). Similarly, the most common integrated models (or target models) are IFC (28 publications), and CityGML (22 publications). Furthermore, 17 integration efforts based on conversion address the translation of some information model to a native information model of a proprietary software product like ArcGIS [127,139,158,172], Revit [128,139,172], Civil3D [135] or InfraWorks [138,172] what is generally achieved by native import functionalities provided by the respective software product. Other target models are based on formats like shapefile [151–153,159,168], 3Dtiles [143,156] or GeoSPARQL [148,149]
- Integration efforts about extension generally address the **extension** of either CityGML (8 publications) or IFC (4 publications). Exemplary ADEs extending CityGML are 3DIM [177], GeoBIM [178], Utility Network ADE [253], ifcADE [106,182], CDM [242] and EnvPlan ADE [186]. On the other side, extensions of IFC are proposed for shield tunnels and underground utilities [183]. The source information models are generally either CityGML or IFC, further models are only addressed by two publications [186,242].
- The integration efforts about **merging** are related to 22 different merged models. The resulting integrated models do generally merge IFC and CityGML while only a few publications address further data sources like native information models or other databases. An often-cited merged model is the Unified Building Model [124–126] which merges IFC and CityGML at schema level and is intended to support the conversion at instance level. Further exemplary merged models are the Quasy [187], BO-IDM [243], USIM [188], IGIM [20,195,205]. Furthermore, some authors utilize existing information models in terms of a merged model. For example, Kumar et al. [199] propose to utilize the LandInfra model as a merged model for IFC and CityGML in the scope of transport infrastructure.
- The integrated models resulting from information integration efforts on **linking** are generally not named except of Multi-Modell [207,209] and District Data Model [198]. Like the other integration methods, the source models of integration efforts about linking generally refer to IFC and CityGML. Some publications further source model like LandXML [209], Okstra [208], BOT [18] or other databases [21,22,93,198,207,211].

Furthermore, the information integration efforts with an integrated model related to BIM are analyzed in more detail with respect to the use case and the use case category as defined in chapter 5 (Table 7.1). In total, there are 12 integration efforts with an integrated model related to BIM. Seven integration efforts do not refer to a specific use case, two integration efforts refer to the use case category *Data management*, and three use cases refer to the use case category *Environment-on-building*. Two integration efforts related to the category *Environment-on-building* import the terrain model to Revit, while the other one sends the analysis result from ArcGIS back to Revit. Conclusively, there is no integration effort in the review literature about BIM-GIS Integration with a use case about the simulation and analysis which uses a BIM model as integrated model, except of integration efforts which aim to represent the terrain model in the BIM model.



(a) (b)

Figure 7.11: Integrated models of integration method conversion (a) per year and (b) absolute numbers.

Table 7.1: Information integration efforts dealing with the conversion to an integrated model related to BIM with respect to the use case.

Publication	**Integration method**	**Integrated model**	**Use Case**	**Category**
[162]	Conversion	IFC	Terrain model e.g., for quantity estimation	*Environment-on-building*
[24]	Conversion, Merging	IFC, CityGML (bidirectional)	No use case	-
[89,124–126]	Conversion, Merging	IFC, CityGML (bidirectional)	No use case	-
[150]	Conversion	IFC	Change management	*Data management*
[13]	Conversion	IFC	No use case	
[160]	Conversion	IFC	No use case	
[172]	Conversion	Revit	Terrain model e.g., for quantity estimation	*Environment-on-building*
[139]	Conversion	ArcGIS, Revit (bidirectional)	Feedback of noise emission analysis on building from GIS system	*Environment-on-building*
[128]	Conversion	Revit	Retrieve information about manufacturer/ supplier	*Data management*

7.2.3 Application system

The application system transforms the integrated model to the application output that is intended to fulfill some task specification. In a more technical sense, the application system refers to an application that is composed of several **functions** transforming input information to output information. In Figure 7.12, the integrated model is represented through the symbol X_2 and the application output is represented through the symbol X_3.

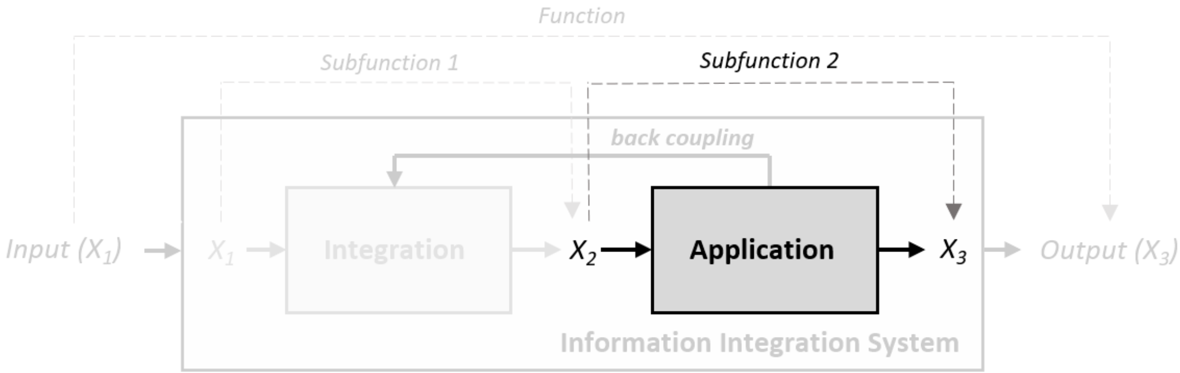


Figure 7.12: The application system of an information integration system which take the integrated model (X_2) as input and transforms it to the application output (X_3).

In Figure 7.13, source models and the applications to which they belong are represented, according to the representation of computer-based information systems in chapter 7.1. Here, it is assumed that a function belonging to application A requires information in the scope of the respective information model A. For example, the functions implemented in the software product Revit require information which can be represented by the internal information model of Revit. Thus, there are three different scenarios to which a function of an application can relate in an information integration system: to both source models, to one of the source models but not to the other, or to none of the source models. In the following, these three scenarios are described in more detail.

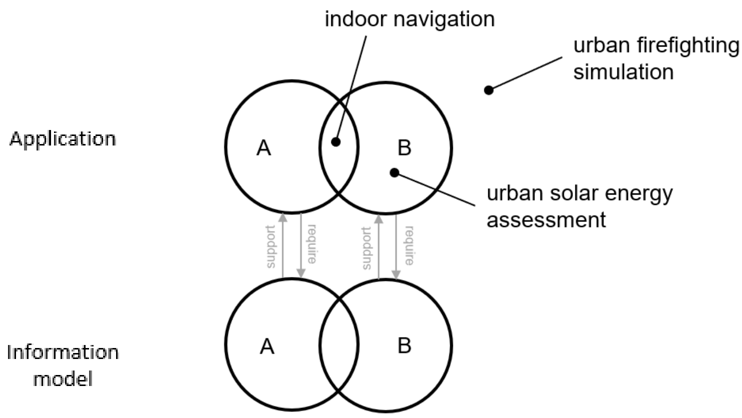


Figure 7.13: Scope of information models and applications related to domains A and B and exemplary use cases. The figure is adapted from Beck et al. [3].

- The function refers to the applications related to both source model A and source model B, written $\boldsymbol{Function} \in \boldsymbol{A} \cap \boldsymbol{B}$ (Figure 7.14). Here, the relevant information refers to the overlapping area of both information models. Examples are indoor navigation scenarios in which the relevant indoor information can be represented by both IFC and CityGML [154,255]. Another example are consistency check of two information models to communicate model modifications. For instance, consistency checks after the handover of building information represented in IFC to a city model like CityGML [245] or in the field of railway infrastructure [256].
- The function refers to the application related to one of the source models but not to the other, written $\boldsymbol{Function} \in \boldsymbol{A} \setminus \boldsymbol{B}$ (Figure 7.14). Here, all relevant information for this kind of function can be represented by one source model but not by the other information model. Examples are alignment optimization scenarios in which information about the infrastructure alignment of the BIM model and the topology or geology from a GIS model is required which can be represented in a GIS software product [135,163]. This kind of function is the most common one in the field of BIM-GIS Integration.
- The function refers to an application related to neither information model A nor B, written $\boldsymbol{Function} \in (\boldsymbol{A} \cup \boldsymbol{B})^{\boldsymbol{C}}$ (Figure 7.14). Such kind of function requires information that is within the scope of both instance models. An example is checking building projects against building codes provided by government agencies which require both building and geospatial information [203]. This is because some information required for compliance checking rules does not exist in CityGML (like “front yard”) while others lack in IFC (like “setback distance”). Thus, the compliance checking rule saying that the setback distance where there is a front yard should be at least 5.00 m demands the integration of both IFC and CityGML models.

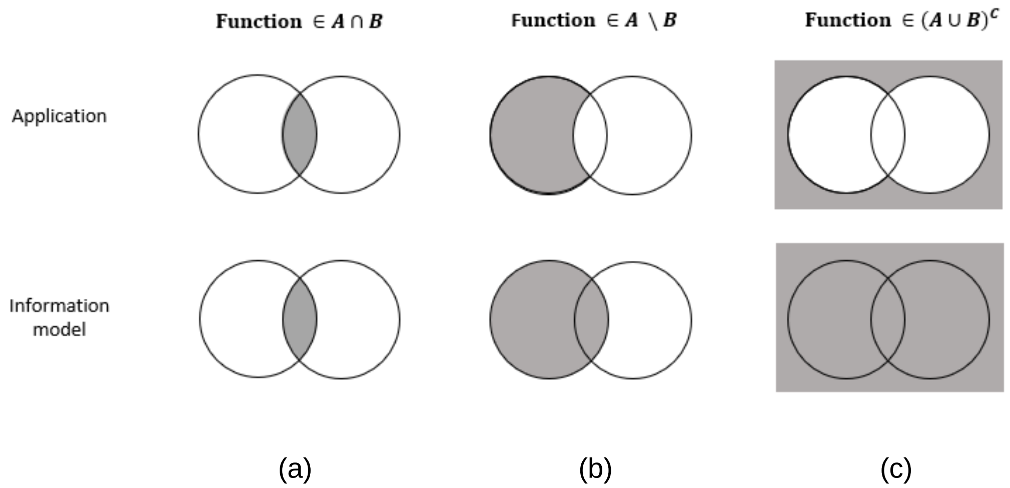


Figure 7.14: Different information integration scenarios in which the function refers (a) to both source models, (b) to one of the source models but not to the other, (c) or to none of the source models. The figure is adapted from Beck et al. [3].

The type of integration scenario influences the choice of the **integration method**. The functions $\boldsymbol{Function} \in \boldsymbol{A} \cap \boldsymbol{B}$ and $\boldsymbol{Function} \in \boldsymbol{A} \setminus \boldsymbol{B}$ demand the transfer of information limited to the overlapping space (Figure 7.15a). This kind of information transfer is generally achieved through the conversion of the information models *[90,96,143,148,150]*, but can also be achieved through other integration methods. On the other side, functions referring to $\boldsymbol{Function} \in (\boldsymbol{A} \cup \boldsymbol{B})^{\boldsymbol{C}}$ require information out of the scope of the overlapping space from both domains (Figure 7.15b). Here, the integration method conversion is not sufficient for these kinds of functions so one of the integration methods extension, merging or linking must be applied. The former type of functions is further referenced by the term *Conversion use cases* (Figure 7.15a) while the latter type of functions will be further referenced by the term *Non-conversion use cases (*Figure 7.15b*).*

The **need for information integration** arises from the situation in which a function requires instance-level information represented in two different information models. Similarly, some authors argue that there is a need for 'BIM-GIS Integration' because both domains deal with similar information [85,108,109], like *Conversion use cases* ($Function \in A \setminus B, Function \in A \cap B$). Other authors argue that these domains deal with different information [85,109,115–117,257], like *Non-conversion use cases* ($Function \in (A \cup B)^C$). As an example of the latter type of argumentation, Rich and Davies [257] state that utility networks do not stop at the outside of the building so the exterior and the interior need to be integrated.

The integration method **linking** supports *Non-conversion use cases* ($\mathrm{Function} \in (\mathrm{A} \cup \mathrm{B})^{\mathrm{C}}$) since it enables queries beyond the overlapping scope of both information models. Furthermore, integration method linking allows for meeting high requirements for *data sovereignty* or for the *timeliness* of information [258,259]. For instance, Hijazi et al. [213] link two building models represented in CityGML and IFC to avoid time-consuming preprocessing of files and prevail control about privacy rights on the data (*timeliness of information*). Another example is the administration of the release of links to specific stakeholders (*data sovereignty).*

The information required by a function can be expressed in terms of a query. There are four integral components of this kind of query describing the relevant information supporting the function.

- Objects related to information model A but not to B (Blue).
- Objects related to information model B but not to A (Orange)
- Objects related to both information models A and B (<u>underlined</u>)
- The adjacency relation between the objects (*cursive*)

Queries belonging to *Conversion use cases* address either objects related to A but not to B *or* vice versa. Queries belonging to *Non-conversion use cases* belong to both objects related to A but not to B *and* vice versa. In Table 7.2, exemplary queries with respect to BIM and GIS are represented and the components are colorized as defined above. Furthermore, the components defining an adjacency correspondence are made cursive while components defining an equivalence correspondence refer to those which are underlined.

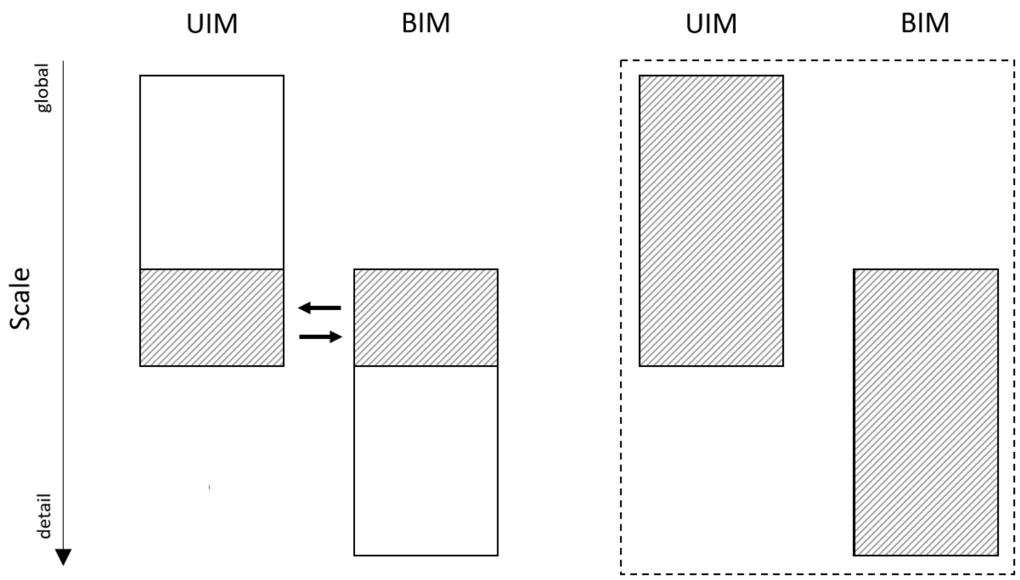


Figure 7.15: Dependency between function type and integration method in terms of (a) conversion use cases and (b) non-conversion use cases.

Table 7.2: Exemplary queries about BIM-GIS Integration scenarios with analysis of their components and assignment to intersection, difference, and complement information integration scenarios.

	Query description
Conversion	Find all buildings that are *located directly next to* the road alignment.
Non-conversion	Give me the insulation material of the walls that are *adjacent to* other buildings with a height of more than 20m and industrial usage.

Orange = objects related to GIS; Blue = objects related to BIM; underlined = objects related to both (equivalence correspondence); *cursive* = local preposition (adjacency correspondence)

To support the thesis that functions can expressed through a query, the integration efforts in the related literature on BIM-GIS Integration using Non-conversion integration methods are analyzed in more detail. First, the identified integration efforts are analyzed with respect to the detailedness of the use case description, and whether the use case was applied or not. Afterward, the underlying function of the detailed use case descriptions were expressed through a query as described above and analyzed whether they refer to Conversion or Non-conversion use cases.

The detailedness of the use case description is evaluated following the categories *Detailed*, *Rough*, *Undefined*.

- *Detailed* means that the publication exactly describes which objects are integrated in either formal (like queries) or unformal (like text) manner. The detailed use cases are further described below.

- *Rough* means that the publication describes a specific use case, but not exactly which objects are integrated. An example for a rough use case is the question “How will a newly constructed building impact existing nearby units?” or the question “What is the impact on the outdoor thermal comfort (e.g. facade material reflectivity)?” [106].
- *Undefined* means that the publication emphasizes some use case using some keyword but does not further describe the use case.

In summary, there are seven publications assigned to the category *Detailed*, 32 publications assigned to the category *Rough,* and 14 assigned to the category *Undefined* (Figure 7.16). Noteworthy, there are solely seven publications related to *Detailed* use case descriptions out of 53 publications, whereby five of these seven publications were published in 2021.

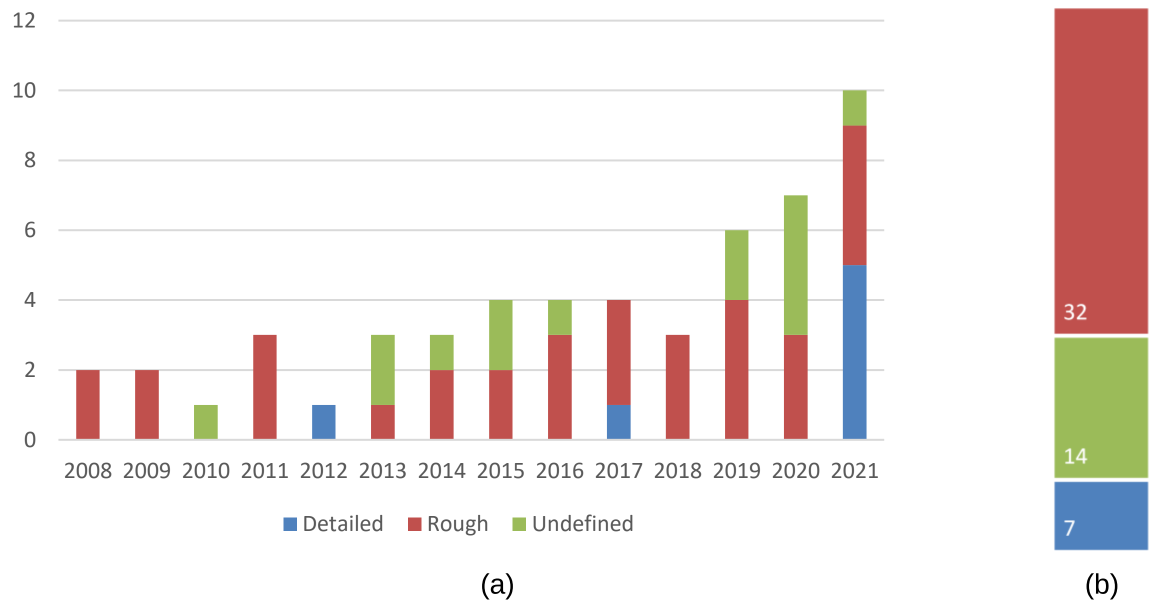


Figure 7.16: Number of detailed, rough and undefined use case descriptions (a) per year and (b) in absolute numbers.

Furthermore, the integration efforts are evaluated whether the use case was applied so that there are three categories, namely *Applied, Conceptual,* and *Unclear*. Notably, the assignment to these categories is based on some perception of the reader since the publications often do not allow a certain conclusion whether the use case has really been applied.

- *Applied* means that the integration method was demonstrated through the use case.
- *Conceptual* means that the integration method was not demonstrated through the use case.
- *Unclear* means that the application is described in a rough manner so that it is unclear whether they are *applied* or *conceptual*.

In summary, 22 publications were assigned to the category *Applied*, 17 publications to the category *Conceptual,* and 11 to the category *Unclear* (Figure 7.17).

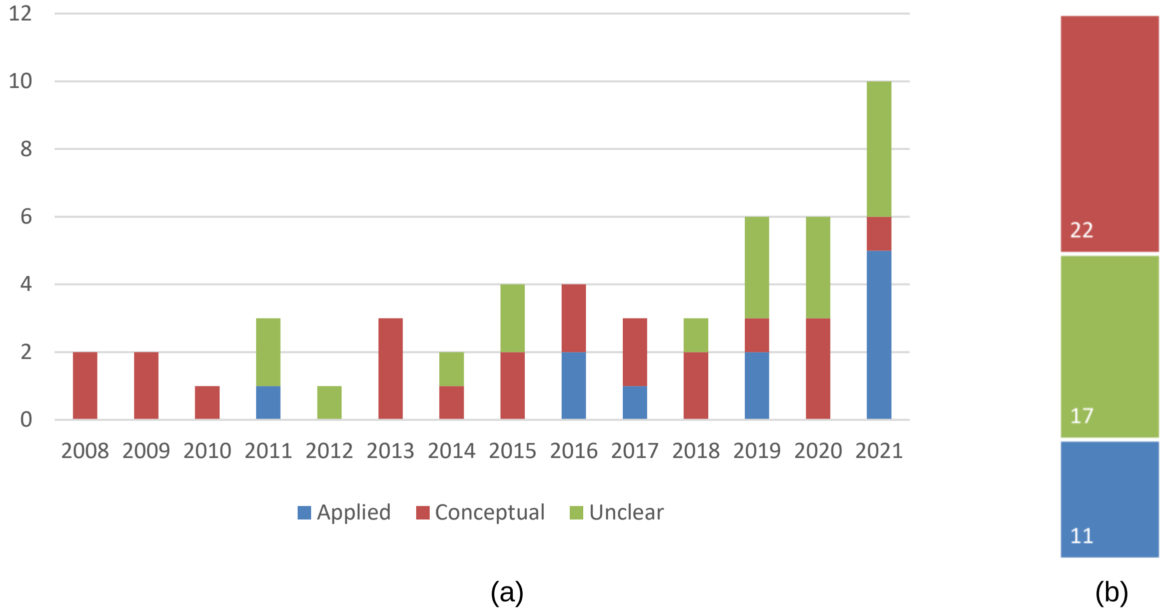


Figure 7.17: Number of use cases that were applied, conceptual or unclear regarding their application (a) per year and (b) in absolute numbers.

The eight publications referring to the category *Detailed* describe 21 different use cases which are summarized in terms of queries in Table 7.3. The queries were further analyzed and subsequently categorized into *Conversion use cases* and *Non-conversion use cases* (chapter 7.2.3). In total, 19 use cases out of these 21 use cases are *Conversion use cases*. Only two use cases are *Non-conversion use cases* which are analyzed below.

- *Use case #1*: In this use case, the possible windows on a specific floor that can be extended as doors in emergency situations or have access to the roof of an adjacent building in emergency situations shall be identified. Here, the need for non-conversion integration method results from the fact that the two subqueries are merged in one: First, querying the possible windows in a specific floor that can be extended as doors in emergency situations. Second, querying the possible windows in a specific floor that have access to the roof of an adjacent building. However, the question arises as to how practically relevant this type of query is since the openings relevant for emergency situations are well standardized and are usually planned before modeling the building.

- *Use case #2*: Here, the environmental time restrictions of some environmental area in the range of a construction area related to the start day of that construction shall be queried. In other words, it shall be checked whether the start day of the construction conflicts with the environmental time restrictions in that area. The need for non-conversion integration method results from the assumption that the schedule data about the start date of the construction is solely represented in the BIM system, while the construction area and the environmental area can be represented in the GIS system.

All other detailed use cases of the reviewed publications referring to non-conversion integration methods are categorized as *Conversion use cases*. These kinds of *Conversion use cases* have in common that they all require the conversion of data from an information model related to BIM to an information model related to GIS.

Table 7.3: Detailed uses cases identified in the reviewed literature expressed as queries and categorized with respect to non-conversion and conversion use cases.

#	Description	Author	Method
Non-conversion use cases			
1	Define the possible windows in a specific floor that can be extended as doors in emergency situations or have access to the roof of an *adjacent* building in emergency situations.	[125]	*M*
2	Query the environmental time restrictions of some environmental area *in the range* of a construction area related to the start day of that construction.	[186]	*E*
Conversion use cases			
3	Find the doors that *have access* to the outer environment of a building and define the use of the accessed outer space.	[125]	*M*
4	If there is an existing building *in* any of the parcels within the same block, setback distances should be the same as setback distances used in that parcel.	[203]	*M*
5	Querying private buildings *above* the tunnel alignment and *built on* specific property (other information model, but could be represented in GIS).	[22]	*L*
6	Find gas stations and car wash service buildings in the target area and *their closest* road section information	[260]	*L*
7	Which height have the buildings being *in the range* of the planned crane tower.	[17]	*L*
8	Select outdoor areas that have trees or the applicability/ability to be planted and furnished with a water fountain or noise inhibitors.	[125]	*M*
9	Find the surrounding buildings that have industrial use and *are within* a 300 meter distance from the development site of the hospital.	[125]	*M*
10	How much greenery is visible from specific windows and viewpoints of a building?	[106]	*E*
11	What is the total floor area of all constructions *in a* certain plot?	[106]	*E*
12	What is the area covered with vegetation on residential buildings?	[106]	*E*
13	Query the area which affected by the construction area and is biotope area.	[186]	*E*
14	Query the "environmental measures" that affect the construction areas.	[186]	*E*
15	Find all buildings that are *located directly above* the tunnel.	[22]	*L*
16	Find buildings under historical preservation that *are in range* of the alignment.	[22]	*L*
17	Find the areas with time restrictions in the scope of the site area.	[186]	*E*
18	Get the height of the building story and add it to another ontology.	[198]	*L*

L = Linking, M = Merging, E = Extension

Orange = objects related to GIS; Blue = objects related to BIM; underlined = objects related to both (equivalence correspondence); *cursive* = local preposition (adjacency correspondence)

7.2.4 Communication system

The communication system refers to the information flow between both integration and application system (Figure 7.18). In an information integration environment, there are two subjects that need to be communicated between the integration and the applications system. First, the expectations on the integrated information coming from the application system which is here expressed through the *back coupling*. Second, the integrated model which is sent from the integration system to the application system and is here denoted by the symbol X_2.

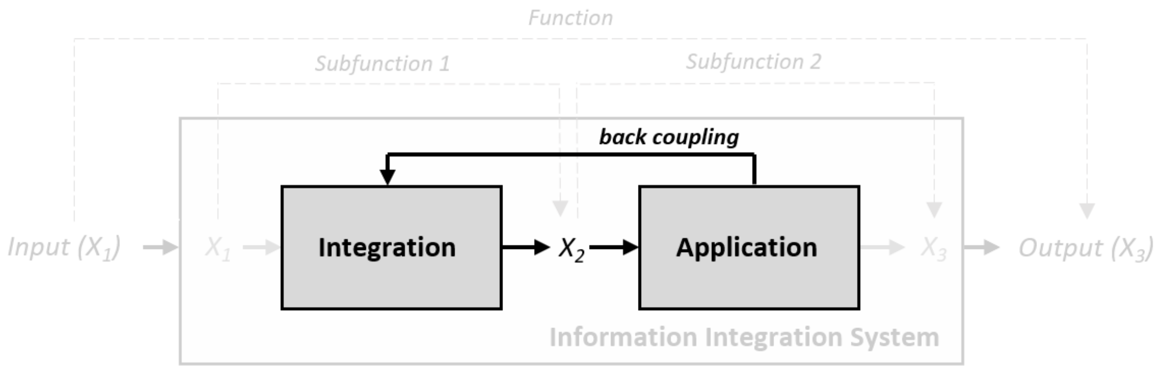


Figure 7.18: The communication system of an information integration system describing the information exchange between the integration and application system.

The process of communicating these subjects can be represented by a procedural model. There are several of these kinds of procedural models (see Weaver [261], Gerbner [262], Lasswell [263], Newcomb [264], Westley and MacLean [265], Jakobson [266]). One of the most influential models is provided by Shannon and Weaver's Mathematical Theory of Communication. The model results from an investigation during the Second World War aiming to send a maximum amount of information along a given channel. In a nutshell, the model of Shannon and Weaver represents the exchange of information from sender to receiver through a channel while the noise within this channel refers to different communication problems (Figure 7.19).

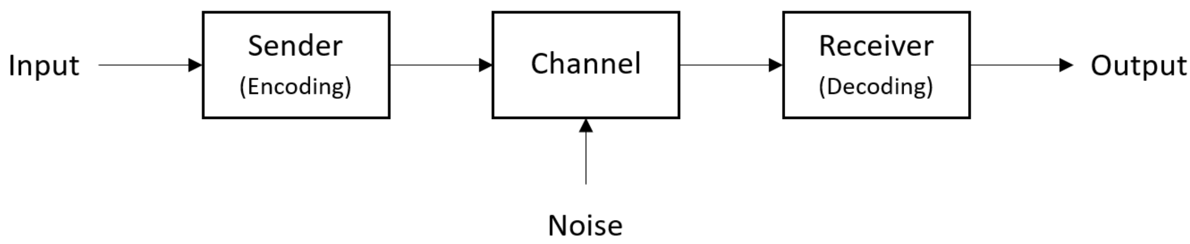


Figure 7.19: Procedural model of communication process according to Weaver [261].

In their study, Shannon and Weaver identified three types of **communication problems**:

- The *technical problem* concerns the issue of how accurately the symbols of communication can be transmitted.
- The *semantic problem* refers to the issue of how precisely the transmitted symbols convey the desired meaning.

- The *efficacy problem* addresses the question of how much noise affects the message in the communication channel.

This dissertation majorly addresses the semantic problem with respect to links that is described in the following chapter. The **semantic problem** can be further illustrated through the structural model about the relation between a symbol and its meaning. Common structural models are developed by C.S. Pierce [267], Ogden and Richards [268], and Ferdinand des Saussure [269]. In particular, the models of Ogden and Richards, and Peirce are similar and often cited. Both models are based on a triangle structure (Figure 7.20a). The three vertices of this triangle structure represent the *symbol* itself, the *referent* to which the symbol refers, and the *thought* which represents the meaning. The edges refer to the relationships between these vertices. Notably, the naming of these edges and vertices is not commonly defined so there are also other names in literature like *sign*, *object,* and *construct* respectively. The edges are commonly represented as *one-to-one* relationships even though *n-to-m* relationships would be rather adequate. These *n-to-m* relationships cause semantic problems in the communication process which can be further illustrated through the model about the *direction of fit*.

The **direction of fit** adopts the semiotic triangle and includes process directions, called *world-to-word* and *word-to-world* respectively (Figure 7.20b). Here, the concept *world* stands for the referent while the concept *word* stands for the symbol. The semantic problem relevant for further research work occurs when the referent intended by the sender does not match the referent decoded by the receiver. For example, Person A (sender) writes that a "beam has length 2.50 meters" and refers to the length of the beam without support length, while Person B (receiver) understands by reading this sentence that the value 2.50 meters refers to the true length of the beam. The obstructive consequences of these kinds of semantic problems with respect to planning processes are, among others, investigated by Schönwandt [270].

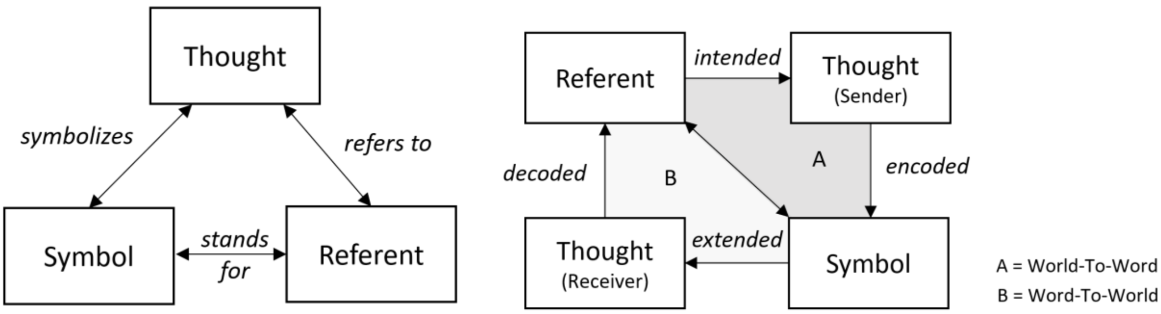


Figure 7.20: (a) Structural model of the semantics of information called semiotic triangle. (b) Procedural model illustrating the direction of fit in the communication process based on the concept of the semiotic triangle.

In total, there are two interfaces in the communication system relating the information system and the application system which are prone for semantic problems. First, the misinterpretation of the expectations on the integrated information coming from the application system. Second, the misinterpretation of the integrated model coming from the integration system. These kinds of semantic problems are not addressed in research about linking information models from the domains BIM and GIS. This is because the link creation and querying are performed by the same person, or the respective algorithms are implemented by the same person. For example, Stepien et al. [22] link several different models like city models and cadastral maps to an infrastructure alignment and subsequently query the linked model. As another example, Zheng links buildings to their closest road sections using Semantic Web technologies and subsequently query the linked model.

7.3 Implementation

The implementation of an information integration system refers to the implementation of an information system, the implementation of the application system and respective interfaces. The integration systems refers to the procedural steps *link creation* and *maintenance of the integrated model*. In BIM-GIS Integration, the link creation is based on three major steps, namely spatial placement, matching and alignment formulation (Figure 7.21). First, the source models must be spatially placed since the matching in the field of BIM and GIS is generally based on spatial identification. Afterward, the corresponding objects are matched based on the previously defined matching criteria. Last, the alignment is formulated which can be performed, for example, through a SPARQL statement. The single steps in the link creation process can be performed manually, automatically, or semi-automatically. Furthermore, the relevant information must be parsed to the integration system and the links must be transferred to the system in which the integrated model is maintained.

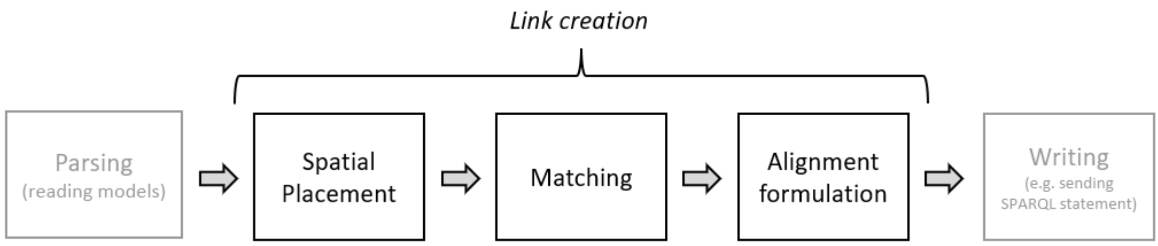


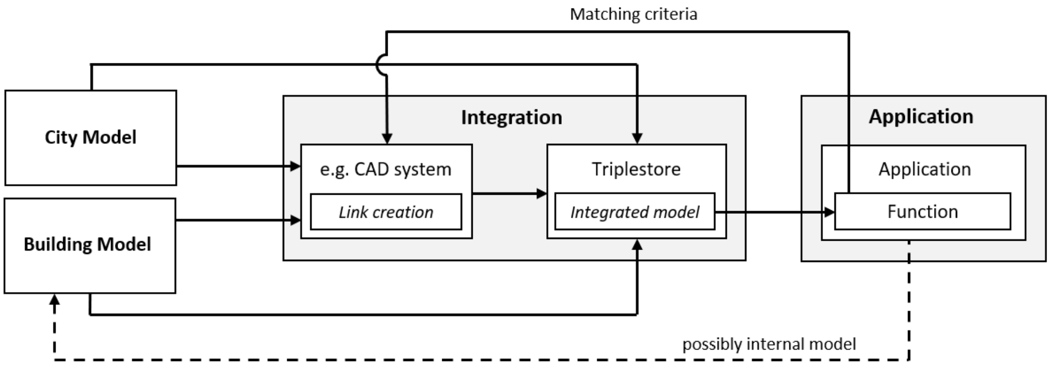
Figure 7.21: The link creation process for spatial matching covers spatial placement, matching and alignment formulation.

In the application system the respective functions process the integrated information which is expressed through a query retrieving the relevant data from the integrated information model. As example, an integration system can be based on a triplestore, and the application system can be a BIM authoring tool such as Revit from which the query (function) is performed. Notably, throughout the dissertation the term *querier* (or *link user*) is used to denote the subject creating and applying the query. This kind of subject can be a human or an algorithm. Nevertheless, the term *querier* emphasized that interpretation of the queried information is conducted by a human, independent whether the query is manually written by a human or automatically by an algorithm which is implemented by a human.

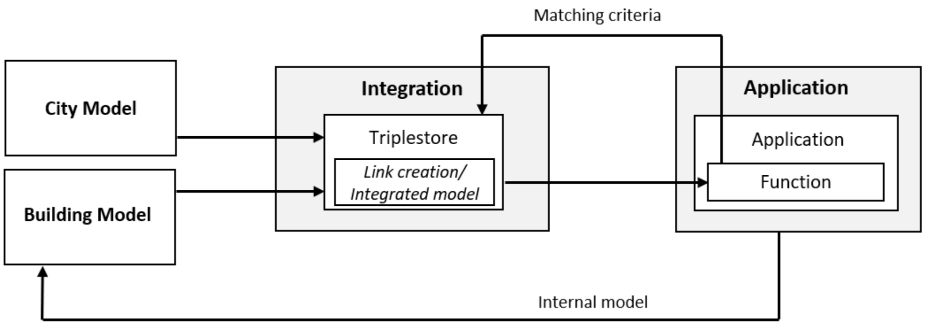
There are two major types for information integration system in the scope of linking information models from the domains BIM and GIS. The two **types of information integration systems** differ with respect to the system in which the link creation and the maintenance of the integrated model is performed, namely in the *same system* or in *different systems* (Figure 7.22). The difference between the two types of information integration systems becomes clear when using Semantic Web technologies for the integrated model. Here, the link creation and maintenance of the integrated model in the same system means that the building and the city model are converted to RDF graphs and the link creation must be performed based on these RDF graphs. However, the identification of corresponding objects generally based on spatial identification and RDF graphs are not appropriate for representing geometry due to the large number of triples. Thus, the spatial identification might be performed in a different system like a CAD or GIS system which is rather adequate for representing geometrical objects. In this kind of information system, the link creation and the maintenance of links is performed in different systems. On the downside of the information integration systems in which the link creation and maintenance is performed in different systems, the source models must be converted twice: First, to the representation in the system of the link creation. Second, to the representation in the system of the maintenance of the integrated model.

Furthermore, the link creation and the query of the integrated model can be performed either *directly sequential*, or *timely separated.* Directly sequential means that integrated model is queried directly after the link creation. Here, the link is not permanent but only created for performing the related query. Each time a query shall be performed, the respective link must be created. On the other hand, *timely separated* means that the link creation and the query are not directly sequential. Thus, the link is created once so that several queries can utilize the link timely independent.

In summary, the implementation of information integration systems refers to several design choices such as performing the link creation and maintenance in *different* or the *same system*, performing the single steps of the link creation *manually* or *automatically*, and performing the link creation and query *directly sequential or timely separated*.



(a)



(b)

Figure 7.22: Structure of implemented information integration system (a) with separated systems, and (b) with combined system for link creation and integrated model.

8 The rationale behind linking

8.1 Modell-theoretic semantics

In the domains BIM and GIS, interpretations of the term ***semantics*** are twofold: In the scope of object-oriented information models, as they are generally utilized in the domains BIM and GIS, the term *semantics* refers to non-geometrical information [4,72]. Coming from the perspective of communication science, semantics refer to the meaning of information, or in other words, to the concept of what a symbol communicates [271,272]. For further research work, the term *semantical information* will be used when referring to semantics in the sense of non-geometrical information, the term *semantics* will be used when referring to the meaning of symbols.

The approach to expressing the semantics of an ontology through some formal language is called ***formal semantics***. Formal semantics aim to constrain the meaning of the used terms of an ontology through the mapping to respective interpretations depending on the domain of discourse [273,274]. Among others, this kind of mapping enables the entailment of implicit information of an ontology. This is because the entailment of implicit information requires a precise meaning of the used terms due to the semantic ambiguities (chapter 7.2.4). A particular type of formal semantics is model-theoretic semantics which is further described and subsequently discussed with respect to instance-level links relating corresponding objects of information models.

Model-theoretic semantics comes from mathematics and was adopted in linguistic and knowledge representation systems such as Semantic Web. Among others, model-theoretic semantics is rooted in Tarski's semantic theory of truth [275] which is based on the idea that formal logical statements in terms of a metalanguage are entailed through some object language. Roughly spoken, the metalanguage is the language, which is used when talking, while the object language is the language one is talking about. For example, the statement "a window is an opening" is true if and only if a window is an opening, whereby the statement in metalanguage is marked with quotation marks. Similarly, model-theoretic semantics address two different levels, namely vocabulary and interpretation. The vocabulary refers to syntactical statements and its components, while the interpretations can be understood as 'realities' or 'worlds' [273,274,276]. Model-theoretic semantics allows evaluating the truth of a statement under a particular interpretation.

In **RDF(S)**, statements are syntactically expressed through triples $s\ p\ o$. The subject s and object o of these triples are mapped to so-called *resources* (IR) while the predicate is mapped to so-called *properties* (IP) (Figure 8.1). *Resources* and *properties* can be considered as the interpretation of the syntactical components of a triple and set up the interpretation domain Δ^{I}. The mapping from the syntactical components to resources and properties refers to the interpretation function $\cdot^{I}$. Additionally, resources are related to each other through the function I_{EXT} which refers to the extension of the predicate p, written $I_{EXT}\ (p^{I})$. In summary, an interpretation I consist of an interpretation domain Δ^{I}, an interpretation function $\cdot^{I}$ and the extension function I_{EXT} (Figure 8.1), written $I =$

$(\Delta^I, \cdot^I, I_{EXT})$. Noteworthy, a more detailed discussion would require the consideration of further aspects, like empty nodes of RDF(S) graphs or the distinctions between literals and URIs [276]. However, the detailedness of model-theoretic semantics for RDF(S) as described above is sufficient for the following discourse.

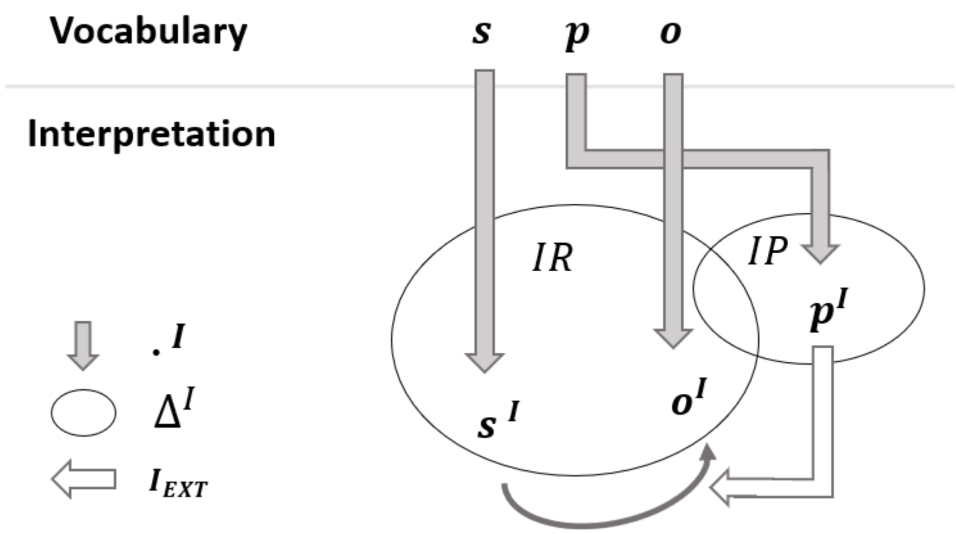


Figure 8.1: Vocabulary and interpretation in terms of model-theoretic semantics for RDFS with the example of an RDF triple according to Hitzler [276].

Model-theoretic semantics provide structures allowing the evaluation of the truthfulness of statements based on some vocabulary. To be more precise, a statement is not called true but ***satisfied*** under a particular interpretation. A particular interpretation under which a statement is satisfied is also called *model* of the statement. A statement under a particular interpretation, written $s\ p\ o.^I$, is called *satisfied* when the following conditions are fulfilled [276]:

- s, p, and o are part of the Vocabulary *V*
- s^I and o^I can be related to each other through $I_{EXT}\ (p^I)$ written $\langle s^I,\ o^I \rangle \in I_{EXT}\ (p^I)$.

The investigation of the latter condition refers to two procedural steps: First, the interpretation of the individual components of a statement, s^I, o^I and p^I. Second, the interpretation of the relation of s^I and o^I through $I_{EXT}\ (p^I)$.

In general, object-oriented information models are not based on some formal semantics which is why information represented by these models cannot be evaluated regarding its truthfulness. Instead, information is generally called *compatible* with object-oriented information models. Nevertheless, the described concepts about model-theoretic semantics are applied to object-oriented information models from BIM and GIS based on the following **assumptions**:

- First, an information model that is syntactically represented as RDF(S) graph refers to the vocabulary *V*, whereby the triples are considered as statements.
- Second, a function of an application refers to a particular interpretation I, written I_C.

Roughly spoken, an RDF statement is called satisfied under a particular interpretation when the statement meets the information requirements of the application, i.e., the information is conceived as ‘good enough’ for the function. The concepts about model-theoretic semantics with respect to the domains BIM and GIS are illustrated in *Example 5*.

8.2 Semantics of links

An alignment relates two ontologies in terms of a set of links [35]. The creation of an alignment is denoted by keywords like *record linkage*, *ontology matching*, *database integration*, *schema mediation,* or *entity resolution* [277]. In the scope of instance models, a link explicitly relates two **corresponding** objects each representing a real-world object. Here, *corresponding* means that the explicit formulation of an implicit relation between two entities serves a specific purpose, such as the fulfilment of a task specification. Similarly, a *correspondence* refers to the implicit relation between corresponding entities.

In this thesis, the entities addressed by a correspondence are objects of an instance model representing a physical object. These kinds of objects are based on a semantical and geometrical description. Following these limitations, there are three different **types of correspondences**:

- *Equivalence correspondence* refers to the relation of two objects representing the same real-world objects.
- *Hierarchy correspondence* refers to the relation of two objects representing real-world objects at different hierarchy levels.
- *Adjacency correspondence* refers to the relation of two objects representing different real world objects, which are to some extent topological adjacent.

An example for equivalence correspondence is the relation of two objects representing the same window. An example for hierarchy correspondence is the relation of objects representing rooms at one building level to an object representing the respective building level. An example for adjacency correspondence is the relation of an object representing an entry of a building and an object representing the street next to that entry.

Noteworthy, the difference between **hierarchy and equivalence correspondence** is not always clear since there is no clear definition whether the related objects represent the same real-world object or real-world objects at different hierarchy levels. For example, a wall object represented as solid 3D geometry and a wall object represented by its outer surface might represent the same real-world object (the wall) or real-world object at different hierarchy levels (the wall and its outer surface). Thus, whether related real-world objects refer to different or same hierarchy level might depend on the perspective. A more detailed discourse about the differentiation between equivalence and hierarchy correspondence is not part of this dissertation but considered as relevant future research topic. Instead, this dissertation solely focuses on equivalence links.

Adjacency correspondences are particularly **relevant** when the information models do not share the objects representing the same real-world objects. For example, the adjacent street linked to the planned building to get information about the traffic. As another example, the buildings observable from the window view might be linked to the window to get information about their height. In case the information models do not share objects representing the same real-world objects, then these objects can be linked through hierarchy or equivalence links.

A major challenge in process of alignment creation is the identification of a correspondences which refers to keywords like *object identification*, ***data matching****,* or *ontology matching* [35,277]. A correspondence is identified based on respective matching criteria. Common matching criteria for the identification of instance-level correspondences are schema-level alignments. Additionally, in geometry-oriented information models, like IFC and CityGML, the identification of correspondences the instance level is often achieved through spatial analysis [258]. For example, two objects representing the same window might be matched because they share the same space and are both instances of classes about windows.

A correspondence is syntactically expressed through a ***link***. The act of making the implicit relation of a correspondence explicit is called *link creation*. Like the correspondence types, there are three types of links. The link types are illustrated based on the MOF in Figure 8.2.

- *Equivalence links* relate two objects representing the same real-world objects.
- *Hierarchical links* relate two objects representing real-world objects at different hierarchy levels.
- *Adjacency links* relate two objects representing different real-world objects, which are to some extent topological adjacent.

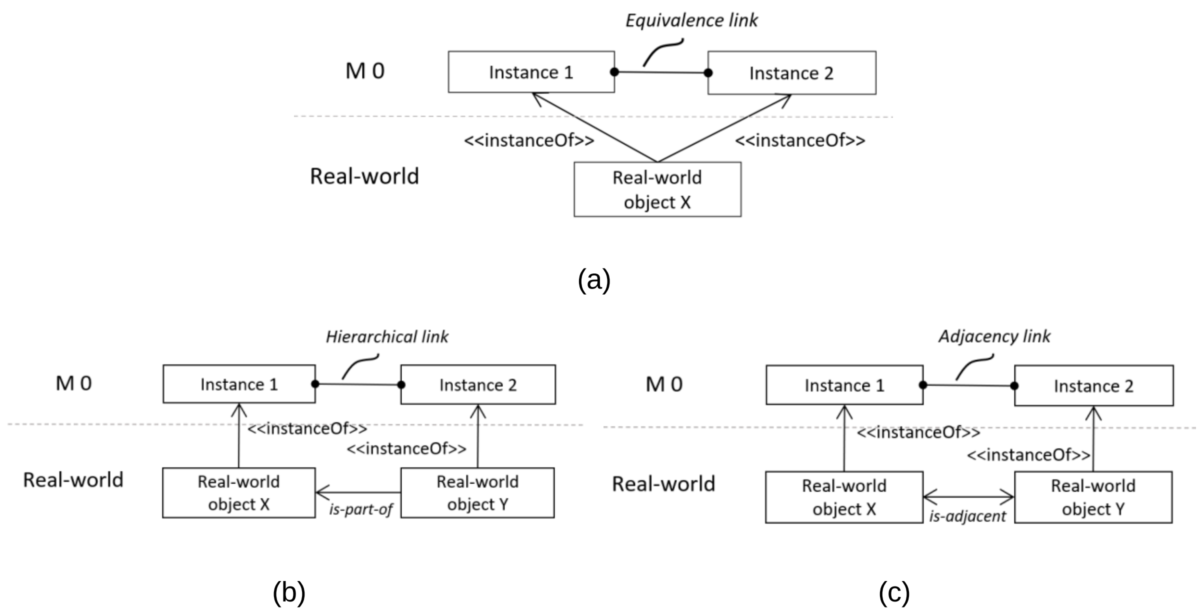


Figure 8.2: (a) Equivalence link, (b) hierarchical link, and (c) adjacency link relating instances of respective real-world object illustrated based on the MOF model.

Equivalence links and **identity links** are often used synonymously. However, *Identity links* relate two objects which are exactly the same thing. Following Leibniz, this means that all of their properties are the same [278] so that everything stated about one object also holds for the other. In contrast to that, equivalence links relate two objects which represent the same real object. Thus, the objects related through an equivalence can be different variants or versions of the same real-world object. In other words, an equivalence link allows to relate heterogeneous objects (in terms of instance-level heterogeneity).

Noteworthy, in the field of BIM-GIS Integration, topological links are often used to link geometrical objects. Topological links describe the topological relationship of these objects. Common topological relationships are provided by the 9-intersection model (9IM) [279] which is utilized in several research works linking BIM and GIS models [280,281] and also adopted in the ontology language GeoSPARQL [282]. An example for a predicate of topological links is geo:sfIntersects which states that two objects geometrically intersect [283]. However, a topological relation does not describe the type of correspondence. This means that a topological relation does not describe whether the related objects represent the same real-world object, real-world objects at different hierarchy levels or different real-world objects which are adjacent. Instead, a topological link relating two objects indicates the type of spatial correspondence. Thus, the use of topological links to link heterogeneous instance models is, therefore, not exact but may be 'good enough' for the task at hand. Consequently, topological links are not further addressed in this thesis, but the topological relation or spatial relation is considered as relevant matching criteria for defining correspondences.

In the field of Semantic Web, a link relates two resources of different RDF(S) vocabularies v_{C1} and v_{C2} based on a common interpretation I_C. A set of these links is called alignment a_C and the different vocabularies v_{C1} and v_{C2} are referenced by v_C. Thus, the integrated information model is composed of the vocabulary v_C, the interpretation I_C, and the alignment a_C (Figure 8.3). Noteworthy, there are further approaches to describing the semantics of links. For example, in another approach, the vocabulary does not refer to a common interpretation, but vocabularies are interpreted independently and their interpretations are mapped to each other [284,285]. In this dissertation, the vocabularies v_{C1} and v_{C2} as part of the integrated information model refer to a common interpretation I_C since they are both interpreted by the same application. Another approach aims to map the alignment itself to an interpretation [285]. In this dissertation, the alignment is not mapped to an interpretation since the meaning of equivalence as described in this chapter has strict semantics and the mapping to some interpretation is therefore not necessary.

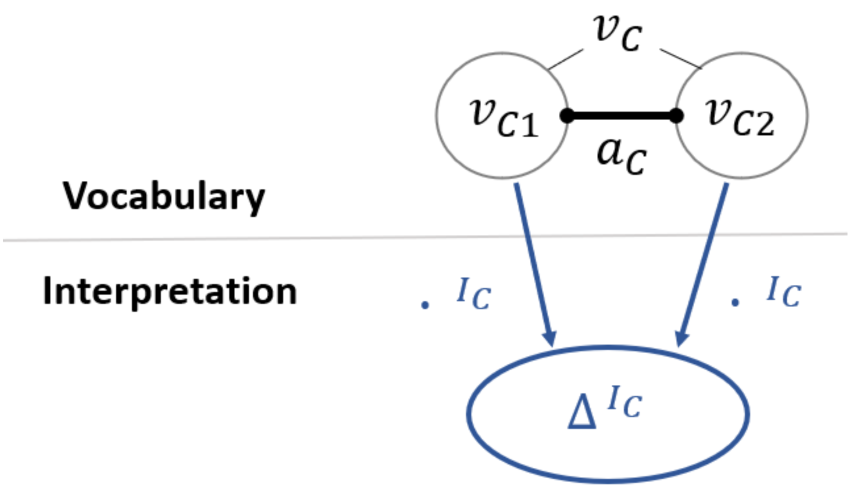


Figure 8.3: Vocabularies that are linked through an alignment and mapped to common interpretation.

An **equivalence link** relates two resources (*here*: s and s') referring to the same real-world object and suggests that something stated about s' also holds for s (Figure 8.4). In other words, the link suggests that the inferred statement $s\ p'\ o'.$ is satisfied under a particular interpretation I_C (Figure 8.4). The statement $s\ p'\ o'.$ is satisfied under the interpretation I_C when following conditions are fulfilled:

- $s\ p'$ and o' are part of the Vocabulary v_C
- s^{I_C} and o'^{I_C} can be related to each other through p'^{I_C}, written $\langle s^{I_C},\ o'^{I_C}\rangle \in I_{EXT}\ (p'^{I_C})$

The evaluation of $\langle s^{I_C},\ o'^{I_C}\rangle \in I_{EXT}\ (p'^{I_C})$ is achieved through the interpretation of all components of the inferred statement, s^{I_C}, p'^{I_C} and o'^{I_C}, and through the interpretation of the whole statement, $s\ p'\ o'.^{I_C}$. The described semantics of an equivalence link with respect to the reference example is illustrated in *Example 5*.

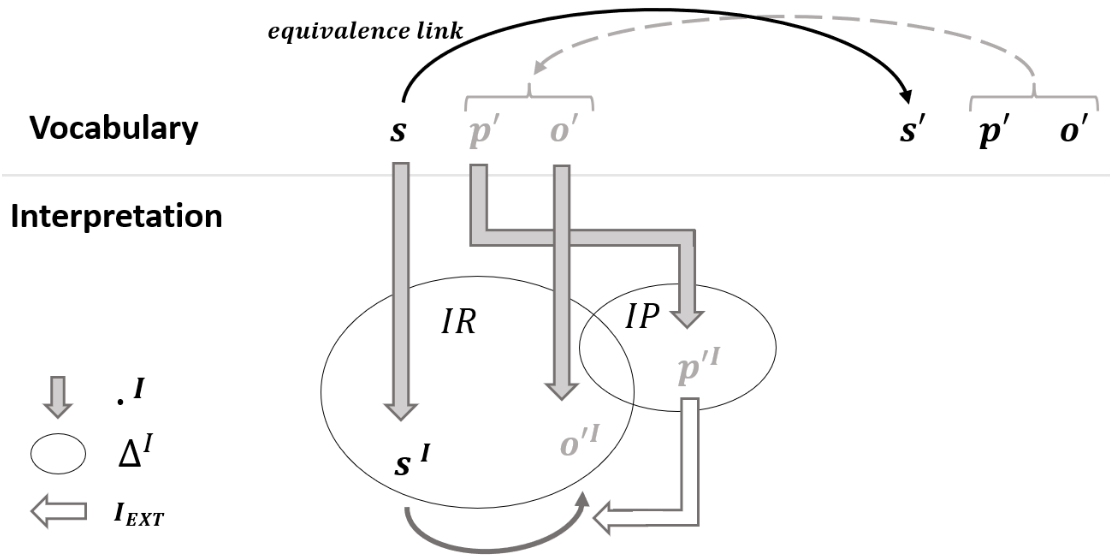


Figure 8.4: Representation of the suggestion of an equivalence link that something stated about s' holds for s following model-theoretic semantics.

Equivalence inks relating two RDF(S) vocabularies can be **categorized** into those which follow truth-conditional semantics, and those which do not. The former can be further categorized into *universal* and *contextual links* while the latter is further referenced by the term *forwarding links* (Table 8.1).

Table 8.1: Exemplary equivalences links according to categories forwarding, universal and contextual links.

Forwarding	**Universal** (truth conditional)	**Contextual** (truth conditional)
skos:exactMatch	*owl:sameAs*	*link:identiConTo, so:claimsIdentical*

A **universal equivalence link**, like *owl:sameAs*, is generally interpreted in terms of identity, or logical equality. Two things are logically equal or identical when they are exactly the same thing so that everything stated about the relating object holds also for the related object, what is also known as Leibniz' principles of indiscernible [225,228,231,278]. Thus, a universal equivalence link relating two things that are *not* exactly the same thing is prone to be misleading [216,218,228,231]. Noteworthy, this misleading character of universal equivalence links for relating non-identical objects is not caused by a lack of semantics but by the misuse of the links [231].

Contextual equivalence links follow the concept of truth-conditional semantics with respect to a specific context, further called validity scope [217]. Here, the validity scope of a contextual link can be defined through both the limitation of the linked vocabulary v_C and the specification of the interpretation I_C for which the link holds (Figure 8.5). The limitation of the linked vocabulary refers to a subset of the vocabulary of the actual linked vocabularies v_A and v_B, what is called v_{C1} and v_{C2} as described previously. An example of *contextual equivalence links* is *so:claimsIdentical* which is used to treat identity assertions "as claims, where the statement of identity is not necessarily true, but only stated by a particular agent." [68] In other words, the identity assertion claimed by the link is limited to a specific context.

Forwarding equivalence links do not refer to a truth value and are, therefore, 'weaker' than *truth conditional links*. An example of *forwarding equivalence links* is the link predicate *skos:closeMatch* which indicates two entities that "can be used interchangeably in some information retrieval applications" [286]. In the following, the semantics of *universal* and *contextual equivalence links* are described in more detail.

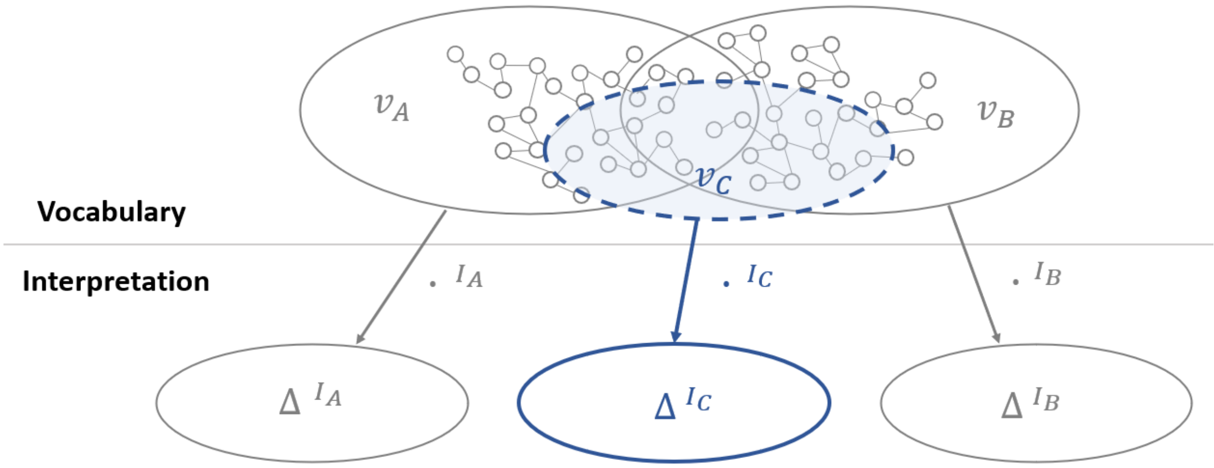


Figure 8.5: Limiting the validity scope of the alignment relating two vocabularies v_A and v_B through specifying the vocabulary v_C or domain of discourse I_C.

Example 5: Model-theoretic semantics

A) Source models, alignment, and interpretation

The two information models provided in the reference example are represented as RDF graphs belonging to the vocabulary v_A and v_B

v_A: inst:IfcBeam_23 rdf:type beo:Beam .

inst:IfcBeam_23 ex:Length "2.50"^^xsd:double .

v_B: inst:BuildingInstallation_71 rdf:type bldg:BuildingInstallation .

The vocabulary v_A is interpreted following the interpretation $I = (\Delta^I, \cdot^I, I_{EXT})$. Thus, the statement specifying the length of the IfcBeam in v_A can be expressed as follows:

$IR = \{\Pi, 2.50\}$ $IP = \{\varepsilon\}$ $LV = \{2.50\}$

$I_{EXT} = \varepsilon \rightarrow \{\langle \Pi, 2.50 \rangle\}$ $I_S =$ $\text{inst:IfcBeam\_23} \rightarrow \Pi$

$\text{ex:Length} \rightarrow \varepsilon$

(I) $\langle \text{inst:IfcBeam\_23}^I, \text{"2.50"}^I \rangle = \langle \Pi, 2.50 \rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\text{ex:Length}^I)$

The statement specifying that the IfcBeam has the length "2.50" is satisfied under I when the condition (I) is true.

Now, the RDF graph is extended with the alignment a_c that is another triple relating *inst:BuildingInstallation_71* and *inst:IfcBeam_23* through an equivalence link. The equivalence link is here expressed through the link predicate *ex:equivalenceLink.*

a_C: inst:BuildingInstallation_71 ex:equivalenceLink inst:IfcBeam_23

The equivalence link suggests that the statements holding for *inst:IfcBeam_23* also hold for *inst:BuildingInstallation_71* like the statements specifying the length of the IfcBeam. This kind of suggestion with respect to the interpretation I is expressed as follows:

$\mathrm{IR} = \{\,\Psi, 2.50\,\}$ $\mathrm{IP} = \{\,\varepsilon\,\}$ $LV = \{\,2.50\,\}$

$I_{EXT} = \varepsilon \rightarrow \{\,\langle \Psi, 2.50\,\rangle\,\}$ $I_S = \begin{array}{l}\mathrm{ex{:}\,Length} \rightarrow \varepsilon \\ \mathrm{inst{:}\,BuildingInstallation\_71} \rightarrow \Psi\end{array}$

(II) $\langle \mathrm{inst{:}\,BuildingInstallation\_71}^I, \text{"2.50"}^I\,\rangle = \langle \Psi, 2.50\,\rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\mathrm{ex{:}\,Length}^I)$

The statement that the BuildingInstallation has the length "2.50" is satisfied under I when condition (II) is true.

B) Validity scope of the alignment

In the following the two types of limiting the validity scope are described, namely *vocabulary* and *interpretation*.

Vocabulary: The vocabulary v_A is now extended with another triple describing the author of the IfcBeam object.

v_A: inst:IfcBeam_23 rdf:type beo:Beam .

inst:IfcBeam_23 ex:Length "2.50"^^xsd:double .

inst:IfcBeam_23 ex:Author "Smith"^^xsd:string .

v_B: inst:BuildingInstallation_71 rdf:type bldg:BuildingInstallation

A contextual equivalence link may suggest that the length value holds for the inst: BuildingInstallation_71 but does not address the author property. Here, the validity scope of the equivalence link is limited to the subset of the vocabularies v_A and v_B on the beam object and its length value. These kinds of subsets refer to v_{C1} and v_{C2} which compose the vocabulary v_C.

Interpretation: Different applications refer to different interpretations. The interpretation of a particular application is here expressed through I_C: The resource inst: BuildingInstallation_71 is interpreted as a representation of the real-world beam representing the true length of the beam (Ψ). The predicate ex: Length is interpreted as the true length of the beam (ε).

$IR_C = \{\,\Psi, 2.50\,\}$ $IP_C = \{\,\varepsilon\,\}$ $LV_C = \{\,2.50\,\}$

$I_{EXT,C} = \varepsilon \rightarrow \{\,\langle \Psi, 2.50\,\rangle\,\}$ $I_{C,S} = \begin{array}{l}\mathrm{ex{:}\,Length} \rightarrow \varepsilon \\ \mathrm{inst{:}\,BuildingInstallation\_71} \rightarrow \Psi\end{array}$

(II) $\langle \mathrm{inst{:}\,BuildingInstallation\_71}^{I_C}, \text{"2.50"}^{I_C}\,\rangle = \langle \Psi, 2.50\,\rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\mathrm{ex{:}\,Length}^{I_C})$

Following the application, the length value “2.50” represents the true length and holds for both the inst: IfcBeam_23 and the inst: BuildingInstallation_71. Thus, the conditions (I) and (II) are satisfied under interpretation I_C.

8.3 Information integration systems

The model-theoretic perspective on information models and alignments as described above can be transferred to the representation of information integration systems as described in chapter 7 (Figure 8.6). The source models are described by the vocabularies, v_A and v_B, and the respective interpretations, I_A and I_B,. The integrated model is described by the alignment a_C and its validity scope which is composed of the vocabulary v_C and he interpretation I_C.The vocabulary v_C refers to a subset of both v_A and v_B. The interpretation I_C refers to the expectation on the integrated information and communicated from the application system to the integration system. The interpretation I_C defines how the vocabulary v_C is interpreted and which statements in the integrated information are ‘true’.

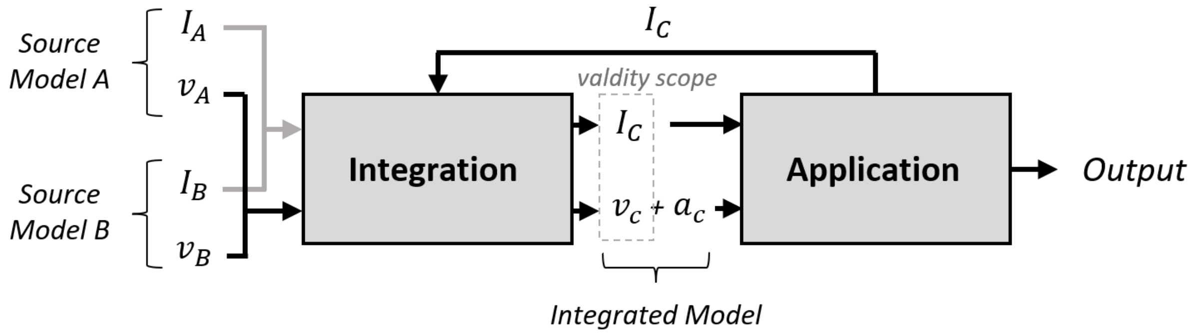


Figure 8.6: Information integration environment based on linking in which the source models and integrated model are represented from the perspective of model-theoretic semantics.

9 Context-sensitive linking

Note: The chapters Context-sensitive linking extends the publication Beck et al. [3]. Beck et al. have not further differentiated between context-sensitive, contextual, and context-dependent linking. Instead, Beck et al. have discussed a specific type of context-sensitive linking what is here called context-dependent linking.

9.1 Idealized information integration system

The implementation of information integration systems can be idealized, further called **idealized information integration systems**. Idealized information integration systems are characterized by following two constraining situational aspects.

- First, the input variables of the integration system are well-defined (*Specified variables*).
- Second, the semantics of the link is transparent to the querier (*Link interpretation).*

Accordingly, *non-idealized information integration systems* are information integration systems which are not constrained with respect to both aspects. Here, the integration is not restricted to specific variants and versions of the source model or specific application (see *Specified variables*), or the semantics of the link is opaque to the user (see *Link interpretation*). As example for the *Link interpretation*, the semantics of the link is transparent to the querier in case the link is created and queried by the same person.

Instance-level linking approaches in research on **BIM-GIS Integration** generally deal with idealized information integration systems. For example, Stepien et al. [22] link several different models like city models and cadastral maps to an infrastructure alignment of a tunnel and Zheng links buildings to their closest road sections using Semantic Web technologies. Both Stepien et al. and Zheng do neither discuss the link creation with respect to different variants or versions of the source models and different applications (see *Specified variables*) nor the interpretation of the link by other persons than the authors (see *Link interpretation*). Conclusively, the related literature about BIM-GIS Integration does not address the identification and management of challenges in non-idealized information integration systems.

Instance-level linking approaches in non-idealized information integration may face more complicated **challenges** for purposefully creating alignments than in idealized information integration systems: What does it mean for the linking process when the links are created and queried by different persons? Do different applications require different alignments, and if so, what are the consequences for the linking process? This kind of discourse is insufficiently addressed in the related literature on both BIM-GIS Integration and contextual linking in the field of Semantic Web.

The potential problems in non-idealized information integration systems related to the two aspects refer to **different perspectives** on the information integration process. The problems related to the aspect *Specified variables* can be illustrated through the perspective on link creation (prospective) and refers to the task problem as described in *Example 3* in chapter 4.2. The problems related to the aspect *Link interpretation* can be illustrated through the perspective on link querying (retrospective) and refers to the task problem as described in *Example 4* in chapter 4.2. These two perspectives on information integration system are illustrated in Figure 9.1. The discourse on these two aspects refers to context-sensitive information integration systems which is described in the next chapter.

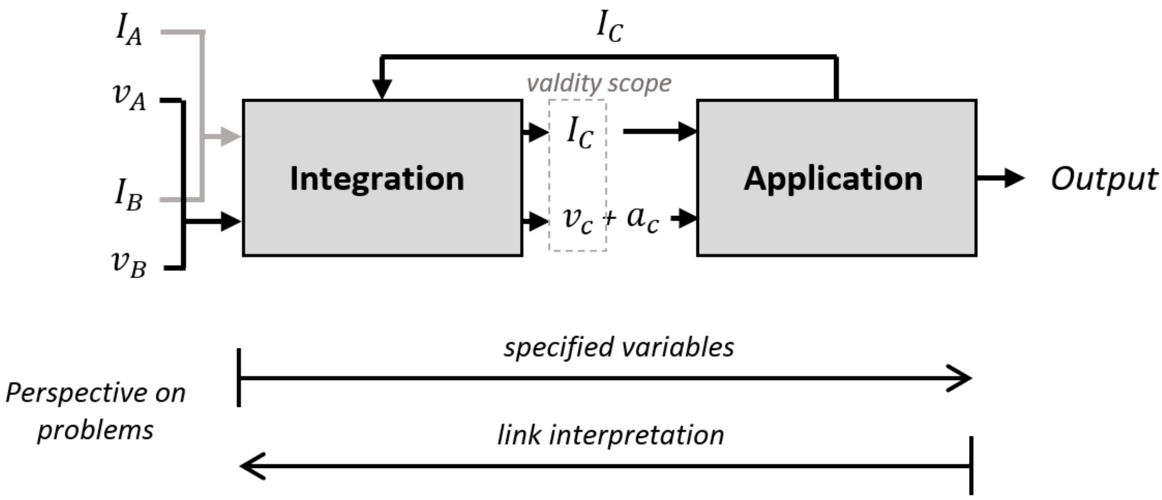


Figure 9.1: Different perspectives on information integration systems describing the problems in non-idealized information integration systems caused by the aspects Specified variables and Link interpretation.

9.2 Context-sensitive information integration systems

The term **context** has different meanings across and within research fields related to information integration in computer-based information systems. As described previously, context might refer to either the source models or the integrated model. For example, Bouquet et al. [218] consider context as locally created information models which encode a party's view of a domain while Aljalbout et al. [217] see context as limited validity scope of an alignment. From a more general perspective, Abowd et al. [287] define context as "any information that can be used to characterize the situation of an entity, where an entity can be a person, place, or physical or computational object." Here, the entity of interest is the alignment relating two information models at the instance level and the situation is characterized by the form of the information integration system. In more detail, the information integration system influences the vocabulary and interpretation relevant for the alignment creation (Figure 9.2). This kind of influence is referenced by the term context-sensitivity of an alignment. Similarly, context-sensitive linking refers to the act of alignment creation depending on situational aspects describing the information integration system.

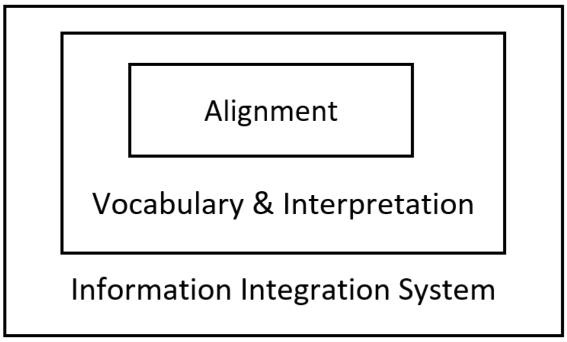
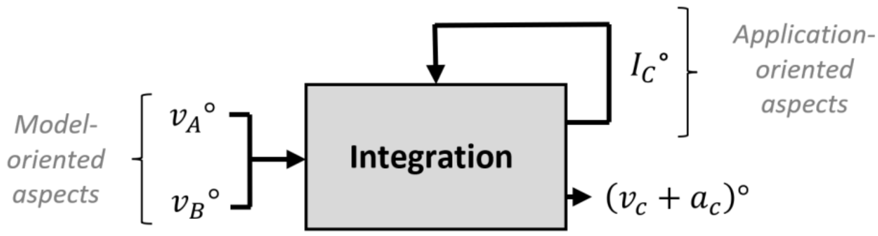


Figure 9.2: Hierarchical structure of information integration system, vocabulary & interpretation representing the context of an alignment.

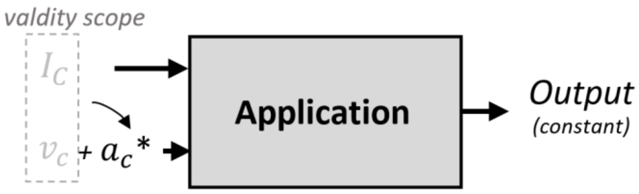
The concept of context-sensitive linking is based on the fact that the output of a system depends on the respective input. In more detail, there are two different types of subsystems of an information integration system, namely *integration system* and *application system*. Similarly, there are two **types of context-sensitive linking** related to these subsystems, further called context-dependent linking which is related to the integration system, and contextual linking which is related to the application system. Figure 9.3 illustrate context-dependent linking and contextual linking from the perspective of model-theoretic semantics and system theory.

- **Context-dependent linking** results from the fact that the output of the integration system depends on its input (Figure 9.3a). Here, the input is the vocabulary v_A , v_B (further called *model-oriented aspects*) and the interpretation of the back coupling process I_C (further called *application-oriented aspects*). The output refers to the vocabulary v_C or the alignment a_C.
- **Contextual linking** results from the fact that the input of the application system depends on its output. Here, the output of application system is constant which means that modifying one input variable leads to modifying another input variable. The output refers to the application result and the modified input variable refers to the implicit validity scope of the alignment, v_C and I_C, which is opaque to the user (Figure 9.3b). The subsequently modified input variable refers to the alignment which must be enriched so that the implicit validity scope is made explicit, and the user can interpret the alignment correctly.

In the following, both kinds of context-sensitive linking are further discussed with respect to equivalence links. The discussion addresses the evaluation of whether a statement $s\ \ p'\ \ o'$. is satisfied under the interpretation I_C. In more detail, the discussion addresses the evaluation whether s^{I_C} and o'^{I_C} can be related to each other through p'^{I_C}, written $\langle s^{I_C},\ o'^{I_C}\rangle \in I_{EXT}\ (p'^{I_C})$.



(a)



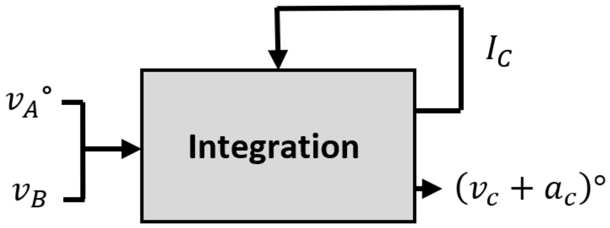
(b)

Figure 9.3: (a) Context-dependent linking and (b) contextual linking represented from perspective of system theory and model-theoretic semantics.

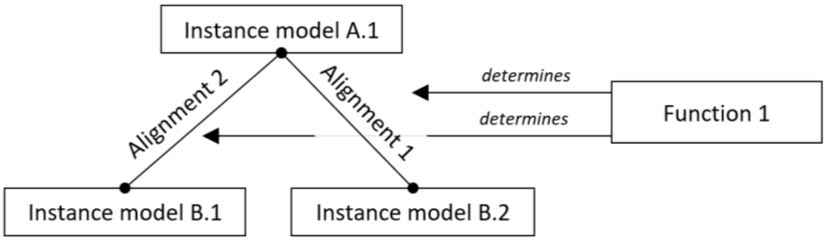
9.3 Context-dependent linking

Context-dependent linking refers to two types of changing input variables for the integration system, namely *model-oriented aspects* (v_A, v_B) and *application-oriented aspects* (I_C). Context-dependent linking caused by application-oriented aspects is further called *application-driven context-dependent linking* while context-dependent linking caused by model-oriented aspects is further called *model-driven context-dependent linking*.

Model-driven context-dependent linking results from the fact that modifying an entity of the vocabulary v_A, v_B leads to a different validity scope v_C or different alignment a_C (Figure 9.4). Here, a component of the statement $s\ p'\ o'$. changes so that the condition $\langle s^{I_C},\ o'^{I_C}\rangle \in I_{EXT}\ (p'^{I_C})$ is satisfied under the interpretation I_C either before or after the change. For example, the object o' changes, written o'°, so that $\langle s^{I_C},\ o'^{\circ I_C}\rangle \in I_{EXT}\ (p'^{I_C})$ is not satisfied under interpretation I_C while $\langle s^{I_C},\ o'^{I_C}\rangle \in I_{EXT}\ (p'^{I_C})$ is satisfied. These kinds of changes can be caused by different variants or versions of source models representing the same real-world object. *Model-driven context-dependent linking* due to different variants and versions is further illustrated in *Example 6*.



(a)



(b)

Figure 9.4: Model-driven context-dependent linking caused by different variants or versions of the source models illustrated from the perspective of (a) model-theoretic semantics and system theory and (b) general perspective

Example 6: Model-driven context-dependent linking

A. Variants

In the following, two variants representing the BuildingInstallation (s, s°) are related to the IfcBeam (s') through an equivalence link. As described previously, the IfcBeam represents the true length, s' p' o' with $s' = \text{inst:IfcBeam\_23}, \mathrm{p}' = \text{ex:Length}, o' = \text{"2,50"}$.

v_A: inst:IfcBeam_23 rdf:type beo:Beam .

inst:IfcBeam_23 ex:Length “2.50”^^xsd:double .

In the first variant, the BuildingInstallation $(s = \text{inst:BuildingInstallation\_71})$ represents the true length since the UIM model was created through conversion from the BIM model.

v_{B1}: inst:BuildingInstallation_71 rdf:type bldg:BuildingInstallation

a_{C1}: inst:BuildingInstallation_71 ex:equivalenceLink inst:IfcBeam_23

The respective interpretations I_C and the condition $\langle s^{I_C}, o'^{I_C} \rangle \in I_{EXT}\ (p'^{I_C})$ are described as follows:

$IR_C = \{ \Psi, 2.50 \}$ $IP_C = \{ \varepsilon \}$ $LV_C = \{ 2.50 \}$

$I_{EXT,C} =$ $\varepsilon \rightarrow \{ \langle \Psi, 2.50 \rangle \}$ $I_{S,C} =$ $\text{ex: Length} \rightarrow \varepsilon$
$\text{inst: BuildingInstallation\_71} \rightarrow \Psi$

(II) $\langle \text{inst: BuildingInstallation\_71}^{I_C}, \text{"2.50"}^{I_C} \rangle = \langle \Psi, 2.50 \rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\text{ex: Length}^{I_C})$

Here, the statement $s\ p'\ o'$. is satisfied under the interpretation I_C since $\langle s^{I_C}, o'^{I_C} \rangle \in I_{EXT}\ (p'^{I_C})$. In other words, the length value of about 2.50m is considered as “true” for the BuildingInstallation.

In the second variant, the BuildingInstallation ($s° = \text{inst: BuildingInstallation\_62}$) represents the visible length since the data was acquired through photogrammetry methods.

v_{B2}: inst:BuildingInstallation_62 rdf:type bldg:BuildingInstallation

a_{C2}: inst:BuildingInstallation_62 ex:equivalenceLink inst:IfcBeam_23

In the following, the BuildingInstallation is interpreted in terms of a beam object representing the visible length ($\text{inst: BuildingInstallation\_62} \rightarrow \Phi$) so that only the visible length can be related to this kind of BuildingInstallation with $\nu \rightarrow \{ \langle \Phi, 2.00 \rangle \}$).

$IR_C = \{ \Psi, 2.50 \}$ $IP_C = \{ \nu \}$ $LV_C = \{ 2.50 \}$

$I_{EXT,C} =$ $\nu \rightarrow \{ \langle \Phi, 2.00 \rangle \}$ $I_{S,C} =$ $\text{ex: Length} \rightarrow \nu$
$\text{inst: BuildingInstallation\_62} \rightarrow \Phi$

(II) $\langle \text{inst: BuildingInstallation\_62}^{I_C}, \text{"2.50"}^{I_C} \rangle = \langle \Phi, 2.50 \rangle \notin I_{EXT}(\nu) = I_{EXT}(\text{ex: Length}^{I_C})$

Here, the statement $s°\ p'\ o'$. is not satisfied under the interpretation I_C since $\langle s°^{I_C}, o'^{I_C} \rangle \notin I_{EXT}\ (p'^{I_C})$. In other words, the length value of about 2.50 is not considered as “true” for the BuildingInstallation.

Thus, an application based on an interpretation I_C might consider the statement $s\ p'\ o'$. as “true” but not the statement $s°\ p'\ o'$. Consequently, the beam objects in the latter case are either corresponding which results in a different alignment a_C or the length property is not considered as part of the validity scope which results in a different validity scope v_C.

B. Versions

In the following, the BuildingInstallation (s) is related to the IfcBeam (s'). The length of the IfcBeam refers to two different versions: During the refurbishment measures the length of the Beam has been increased so that length ($o'° = \text{"2.55"}$) of the IfcBeam of the second version is larger than the length ($o' = \text{"2.50"}$) of the IfcBeam of the first version.

v_B: inst:BuildingInstallation_71 rdf:type bldg:BuildingInstallation

a_C: inst:BuildingInstallation_71 ex:equivalenceLink inst:IfcBeam_23

In the first version, the IfcBeam represents the beam before refurbishment measures.

v_{A1}: inst:IfcBeam_23 rdf:type beo:Beam .
inst:IfcBeam_23 ex:Length "2.50"^^xsd:double .

The BuildingInstallation represents the length of the beam before refurbishment measures with $\text{ex:Length} \rightarrow \varepsilon$ and $\varepsilon \rightarrow \{ \langle \Psi, 2.50 \rangle \}$.

$IR_C = \{ \Psi, 2.50 \}$ $IP_C = \{ \varepsilon \}$ $LV_C = \{ 2.50 \}$

$I_{EXT,C} = \varepsilon \rightarrow \{ \langle \Psi, 2.50 \rangle \}$ $I_{S,C} =$ $\text{ex:Length} \rightarrow \varepsilon$
$\text{inst:BuildingInstallation\_71} \rightarrow \Psi$

(II) $\langle \text{inst:BuildingInstallation\_71}^{I_C}, \text{"2.50"}^{I_C} \rangle = \langle \Psi, 2.50 \rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\text{ex:Length}^{I_C})$

Here, the statement $s\ p'\ o'$. is satisfied under the interpretation I_C since $\langle s^{I_C}, o'^{I_C} \rangle \in I_{EXT}\ (p'^{I_C})$. In other words, the length value of about 2.50m is considered as true for the BuildingInstallation.

In another version, the IfcBeam represents the beam after refurbishment measures.

v_{A1}: inst:IfcBeam_23 rdf:type beo:Beam .
inst:IfcBeam_23 ex:Length "2.50"^^xsd:double .

Again, the BuildingInstallation represents the length of the beam before refurbishment measures with $\text{ex:Length} \rightarrow \varepsilon$ and $\varepsilon \rightarrow \{ \langle \Psi, 2.50 \rangle \}$. However, the length value linked to the BuildingInstallation is 2.55m.

$IR_C = \{ \Psi, 2.55 \}$ $IP_C = \{ \varepsilon \}$ $LV_C = \{ 2.55 \}$

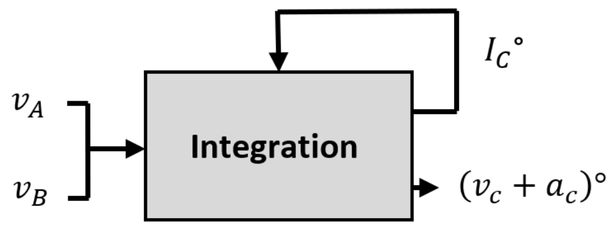
$I_{EXT,C} = \varepsilon \rightarrow \{ \langle \Psi, 2.50 \rangle \}$ $I_{S,C} =$ $\text{ex:Length} \rightarrow \varepsilon$
$\text{inst:BuildingInstallation\_71} \rightarrow \Psi$

(II) $\langle \text{inst:BuildingInstallation\_71}^{I_C}, \text{"2.55"}^{I_C} \rangle = \langle \Psi, 2.55 \rangle \notin I_{EXT}(\varepsilon) = I_{EXT}(\text{ex:Length}^{I_C})$

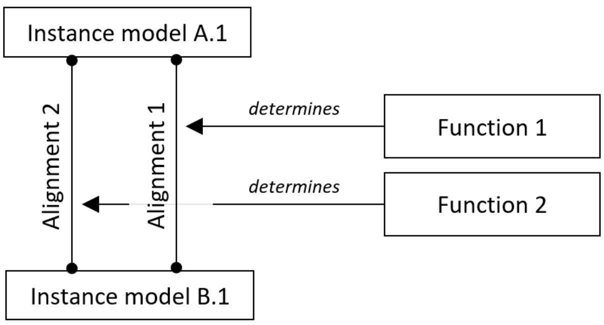
Here, the statement $s\ p'\ o'^{\circ}$. is not satisfied under the interpretation I_C since $\langle s^{I_C}, o'^{\circ I_C} \rangle \notin I_{EXT}\ (p'^{I_C})$. In other words, the length value of about 2.55m is not considered as "true" for the BuildingInstallation.

Thus, an application based on an interpretation I_C might consider the statement $s\ p'\ o'$. as "true" but not the statement $s\ p'\ o'^{\circ}$. Consequently, the beam objects in the latter case are either not related to each other which results in a different alignment a_C or the length property is not considered as part of the validity scope which results in a different validity scope v_C.

Application-driven context-dependent linking results from the fact that modifying the interpretation, written $I_{C°}$, leads to a different validity scope v_C or different alignment a_C (Figure 9.5). In more detail, a statement $s\ p'\ o'$. might be satisfied under interpretation I_C, but not under the interpretation $I_{C°}$. As described in chapter 8.2, the 'truthfulness' of a statement refers to the evaluation of I_{EXT} based on two steps: First the interpretation of the single components s^I, p^I, and o^I. Second, the interpretation of the whole statement $s\ p\ o.^I$. Similarly, application-oriented aspects can be distinguished between those influencing the interpretation of the single components and those influencing the interpretation of the whole statements. Examples of different interpretations of the whole statements are different requirements on the timeliness, accuracy of the information, or different granularity levels. Application-driven context-dependent linking due to different interpretations is further illustrated in *Example 7*.



(a)



(b)

Figure 9.5: Application-driven context-dependent linking caused by different interpretations related to the application illustrated from the perspective of (a) model-theoretic semantics and system theory and (b) general perspective

Example 7: Application-driven context-dependent linking

In the following, the BuildingInstallation is related to the IfcBeam object through an equivalence link. Both the BuildingInstallation ($s = \text{inst:BuildingInstallation\_71}$) and the IfcBeam represent the true length. The length value of the IfcBeam is expressed through the statement $s'\ \text{p}'\ o'$ with $s' = \text{inst:IfcBeam\_23}, \text{p}' = \text{ex:Length}, o' = \text{"2,50"}$. Here, the true length of the IfcBeam (2.50m) is slightly different than the true length of the real-world beam (2.51m). The link and the respective statements are interpreted through different applications. These kinds of different interpretations refer to different interpretations of the single components and to the interpretation of the whole statement $s\ p'\ o'$.

v_A: inst:IfcBeam_23 rdf:type beo:Beam .
inst:IfcBeam_23 ex:Length "2.50"^^xsd:double .

v_B: inst:BuildingInstallation_71 rdf:type bldg:BuildingInstallation

a_C: inst:BuildingInstallation_71 ex:equivalenceLink inst:IfcBeam_23

<u>A. Single components</u>:

In the first application, the length defined through the statement

inst:IfcBeam_23 ex:Length "2.50"^^xsd:double .

is interpreted as the true length of the real-world beam with $\mathrm{ex{:}\,Length} \rightarrow \nu$ and $\nu \rightarrow \{\langle\Psi, 2.51\rangle\}$.

$IR_C = \{\Psi, 2.51\}$ $IP_C = \{\nu\}$ $LV_C = \{2.50\}$

$I_{EXT,C} = \nu \rightarrow \{\langle\Psi, 2.51\rangle\}$ $I_{S,C} =$ $\mathrm{ex{:}\,Length} \rightarrow \nu$
$\mathrm{inst{:}\,BuildingInstallation\_71} \rightarrow \Psi$

(II) $\langle \mathrm{inst{:}\,BuildingInstallation\_71}^{I_C}, \text{"2.50"}^{I_C}\rangle = \langle\Psi, 2.50\rangle \notin I_{EXT}(\nu) = I_{EXT}(\mathrm{ex{:}\,Length}^{I_C})$

Here, the statement $s\ p'\ o'$. is not satisfied under the interpretation I_C since $\langle s^{I_C}, o'^{I_C}\rangle \notin I_{EXT}\ (p'^{I_C})$. In other words, the length value of 2.50m is not considered as true for the BuildingInstallation. This is because the length value of the IfcBeam $\langle\Psi, 2.50\rangle$ differs from the expected length value of the BuildingInstallation $\langle\Psi, 2.51\rangle$.

In the second application, the length defined through the statement

inst:IfcBeam_23 ex:Length "2.50"^^xsd:double .

is interpreted literally as the true length of the IfcBeam with $\mathrm{ex{:}\,Length} \rightarrow \varepsilon$ and $\varepsilon \rightarrow \{\langle\Psi, 2.50\rangle\}$.

$IR_{C^\circ} = \{\Pi, \Psi, 2.50\}$ $IP_{C^\circ} = \{\varepsilon\}$ $LV_{C^\circ} = \{2.50\}$

$I_{EXT,C^\circ} = \varepsilon \rightarrow \{\langle\Psi, 2.50\rangle\}$ $I_{S,C^\circ} =$ $\mathrm{ex{:}\,Length} \rightarrow \varepsilon$
$\mathrm{inst{:}\,BuildingInstallation\_71} \rightarrow \Psi$

(II) $\langle \mathrm{inst{:}\,BuildingInstallation\_71}^{I_{C^\circ}}, \text{"2.50"}^{I_{C^\circ}}\rangle = \langle\Psi, 2.50\rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\mathrm{ex{:}\,Length}^{I_{C^\circ}})$

The statement $s\ p'\ o'$. is satisfied under the interpretation I_{C° since $\langle s^{I_{C^\circ}}, o'^{I_{C^\circ}}\rangle \in I_{EXT}\ (p'^{I_{C^\circ}})$. In other words, the length value of 2.50m is considered as "true" for the BuildingInstallation. This is because the expected length value *is* the length value of the IfcBeam.

Thus, an application based on an interpretation I_{C° might consider the statement $s\ p'\ o'$. as "true" but not an application based on an interpretation I_C. Consequently, the beam objects in the latter case are either not related to each other which results in a different alignment a_C or the length property is not considered as part of the validity scope which results in a different validity scope v_C.

<u>B. Whole statements</u>:

In the first application (I_C), the true length of the Beam value is interpreted as “close enough” (e.g., for quantity estimation).

$$IR_C = \{\Pi, \Psi, 2.50\} \qquad IP_C = \{\varepsilon\} \qquad LV_C = \{2.50\}$$

$$I_{EXT,C} = \begin{array}{l} \varepsilon \rightarrow \{\langle \Pi, 2.50 \rangle\} \\ \varepsilon \rightarrow \{\langle \Psi, 2.497 \rangle\} \end{array} \qquad I_{S,C} = \begin{array}{l} \text{inst: IfcBeam\_23} \rightarrow \Pi \\ \text{ex: Length} \rightarrow \varepsilon \\ \text{inst: BuildingInstallation\_71} \rightarrow \Psi \end{array}$$

(I) $\langle \text{inst: IfcBeam\_23}^{I_C}, \text{"2.50"}^{I_C} \rangle = \langle \Pi, 2.50 \rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\text{ex: Length}^{I_C})$

(II) $\langle \text{inst: BuildingInstallation\_71}^{I_C}, \text{"2.50"}^{I_C} \rangle \approx \langle \Psi, 2.497 \rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\text{ex: Length}^{I_C})$

Here, the length value of the IfcBeam "2.50" is considered as close enough to the expected value $\langle \Psi, 2.497 \rangle$, what is expressed through the symbol ‘≈’ in expression (II).

In the second application (I_{C°), the precise length values are required (e.g., for prefabrication purposes).

$$IR_C = \{\Pi, \Psi, 2.50\} \qquad IP_C = \{\varepsilon\} \qquad LV_C = \{2.50\}$$

$$I_{EXT,C} = \begin{array}{l} \varepsilon \rightarrow \{\langle \Pi, 2.50 \rangle\} \\ \varepsilon \rightarrow \{\langle \Psi, 2.497 \rangle\} \end{array} \qquad I_{S,C} = \begin{array}{l} \text{inst: IfcBeam\_23} \rightarrow \Pi \\ \text{ex: Length} \rightarrow \varepsilon \\ \text{inst: BuildingInstallation\_71} \rightarrow \Psi \end{array}$$

(I) $\langle \text{inst: IfcBeam\_23}^{I_C}, \text{"2.50"}^{I_C} \rangle = \langle \Pi, 2.50 \rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\text{ex: Length}^{I_C})$

(II) $\langle \text{inst: BuildingInstallation\_71}^{I_C}, \text{"2.50"}^{I_C} \rangle \neq \langle \Psi, 2.497 \rangle \in I_{EXT}(\varepsilon) = I_{EXT}(\text{ex: Length}^{I_C})$

Here, the length value of the IfcBeam "2.50" is *not* considered as close enough to the expected value $\langle \Psi, 2.497 \rangle$, what is expressed through the symbol ‘≠’ in expression (II).

Thus, the statement $s\ p'\ o'.$ might be true for quantity estimation (I_C) but not prefabrication purposes (I_{C°). Consequently, the beam objects in the latter case are either not related to each other which results in a different alignment a_C or the length property is not considered as part of the validity scope which results in a different validity scope v_C.

<u>Further aspects</u>

In the described example, the interpretation I_C and I_{C° refer to different requirements on the geometrical accuracy that result in application-driven context-dependent linking. Further aspects causing application-driven context-dependent linking are requirements on timeliness or granularity levels.

Timeliness: Information requirements on timeliness is especially relevant when dealing with (semi-) dynamic information like damage symptoms. As a brief example, in one situation, the damage symptoms must be on time due to project management purposes (I_C), while in another situation the damage symptoms from weeks ago are relevant due to legal reasons (I_{C°). Thus, a statement $s\ p'\ o'.$ describing the current damage symptoms might be 'true' for project management purposes (I_C) but not for juristic purposes (I_{C°).

Granularity: Information requirements on granularity levels are exceptional in the sense that the statements inferred from the equivalence link may be true but irrelevant to the application. For example, a query regarding the number of beam objects in the whole building requires solely linkage at building level, while a query about the fire resistance of beams on a specific evacuation route requires linkage of the single beams located on that route. In the first query, the linkage of the single beams is not necessarily misleading but not relevant for the task. Similarly, in the second query, the linkage of the buildings through an equivalence link is not necessarily misleading but not relevant for the task.

9.4 Contextual linking

Contextual linking refers to making the implicit validity scope of an alignment explicit and occurs due to the opaqueness of the validity scope for the user of the alignment (Figure 9.3b). In other words, the user cannot evaluate whether the statement $s\ p'\ o'.$ is satisfied under his/ her interpretation I_C since he/ she does not know the validity scope of the statement $s\ p'\ o'.$ In contextual linking, the implicit validity scope needs to be made explicit to reduce the risk that the link is misinterpreted by the user. Two approaches aiming to make the validity scope explicit are discussed in chapter 10.3 and contextual linking is further illustrated in *Example 8*.

Example 8: Contextual linking

In the following, the BuildingInstallation is related to the IfcBeam object through an equivalence link. The following RDF statements are provided and the length value of the BuildingInstallation shall be queried for structural calculations.

v_A: inst:IfcBeam_23 rdf:type beo:Beam .
inst:IfcBeam_23 ex:Length "2.50"^^xsd:double .

v_B: inst:BuildingInstallation_71 rdf:type bldg:BuildingInstallation

a_C: inst:BuildingInstallation_71 ex:equivalenceLink inst:IfcBeam_23

Here, the user does not know whether the IfcBeam represents the true or visible length of the beam. In other words, the user does not know the exact meaning of ex: Length, which is further represented by the symbol $?\,\varepsilon$.

$$IR_C = \{\, \Psi, 2.50 \,\} \qquad IP_C = \{\, \varepsilon \,\} \qquad LV_C = \{\, 2.50 \,\}$$

$$I_{EXT,C} = \begin{array}{l} ?\,\varepsilon \rightarrow \{\, \langle \Pi, 2.50 \,\rangle \,\} \\ ?\,\varepsilon \rightarrow \{\, \langle \Psi, 2.50 \rangle \,\} \end{array} \qquad I_{S,C} = \begin{array}{l} \text{ex: Length} \rightarrow ?\,\varepsilon \\ \text{inst: BuildingInstallation\_71} \rightarrow \Psi \end{array}$$

(II) $\langle \text{inst: BuildingInstallation\_71}^{I_C}, \text{"2.50"}^{I_C} \rangle = \langle \Psi, 2.50 \,\rangle \; ?\!\in \; I_{EXT}(?\,\varepsilon) = I_{EXT}(\text{ex: Length}^{I_C})$

Thus, the user cannot evaluate whether the interpretations of a subject and object can be related through $?\,\varepsilon$, written $\langle s^{I_C}, \; o'^{I_C} \rangle \in I_{EXT}\,(?\,\varepsilon)$. Consequently, the validity scope of alignment must be made explicit so that the user can evaluate whether $s \; p' \; o'.$ is satisfied under the interpretation I_C .

10 Design Principles

10.1 Implementation

The design principles are demonstrated through the implementation of the following **information integration system** and its utilization (Figure 10.1).

- The *source models* are an IFC model and a CityGML model. In this dissertation, three instance models are used which represent parts of the campus of the Technical University of Munich (Figure 10.2). An IFC model which was modeled using BIM authoring tool Revit from Autodesk [252]. A CityGML model based on LoD3 which was created through converting the IFC model to CityGML. A CityGML model based on LoD2 which represents the whole TUM campus in the inner city. In Figure 10.2, these three models are represented through FZKViewer [288].
- The *integration system* is based on the software products Rhinoceros 3D [289] and Grasshopper [290] both provided by Robert McNeel & Associates. Rhinoceros 3D is a CAD tool for modeling 2D and 3D geometries. Grasshopper is a visual programming environment within Rhinoceros 3D. The IFC model is imported in Grasshopper through the plug-in GeometryGym [291] and the CityGML model is imported through a C# script. Noteworthy, GeometryGym did not import the ground level of the IFC model. In Grasshopper, corresponding objects are matched and the consequent alignment is pushed to Apache Jena triplestore [292] through the creation of a respective SPARQL statement (Figure 10.3, Figure 10.6, and Figure 10.7). In the Apache Jena triplestore, the integrated model is maintained.
- The *integrated model* originates from the source models and the alignment created in Grasshopper. The integrated model are RDF graphs which are located in the Apache Jena triplestore [292]. The IFC model is converted to RDF through the *IFCtoRDFConverter* [293] while the CityGML model is converted to RDF through the *GMLImporter* [294]. The models and the alignment are pushed to a triple store through the SPARQL endpoint Apache Jena Fuseki [292]. The source models are pushed by means of the web user interface provided by Fuseki while the alignment is pushed through accessing the SPARQL endpoint within Grasshopper. Noteworthy, *IFCtoRDFConverter* did not convert the ground level of the IFC model to RDF. Furthermore, the resulting RDF-based CityGML models (44.2 and 25.3 Megabytes) were too large for loading to the triplestore. Thus, the models were reduced to the triples relevant for the demonstration scenarios. In Figure 10.8, the web user interface provided by Fuseki is represented in which the integrated models composed of source models and alignments (link models) are maintained.

- The *application system* refers to applications that utilize the resulting triple store through the SPARQL endpoint Fuseki which can be achieved by extending the respective software product or by the web user interface provided by Fuseki. In this dissertation, the queries were performed through the web user interface provided by Fuseki (Figure 10.9 and Figure 10.10).

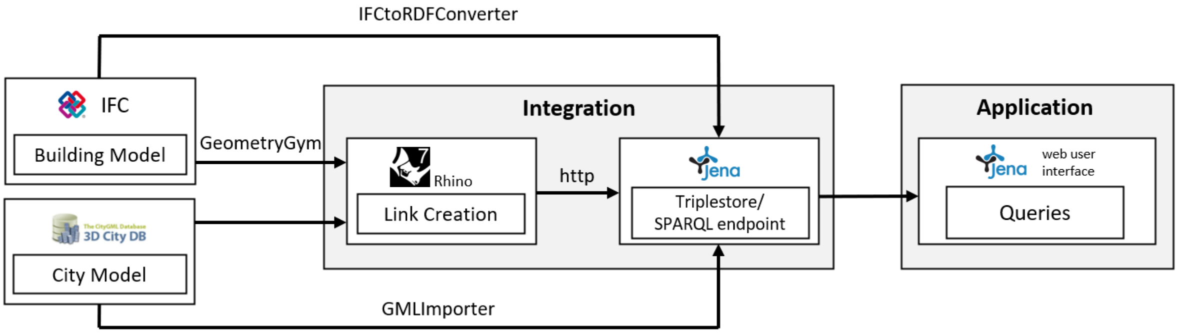
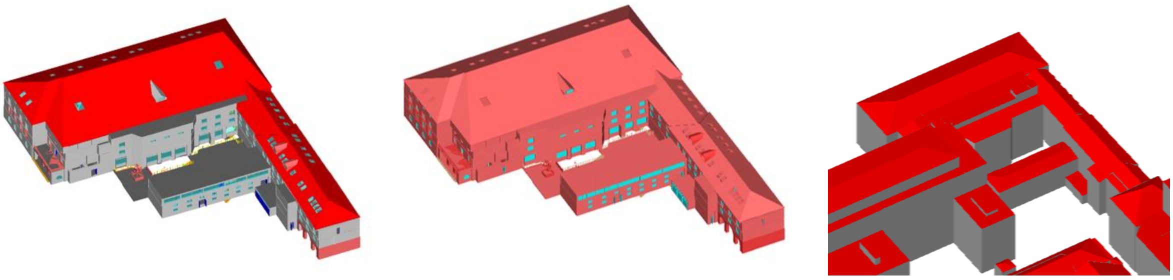


Figure 10.1: Information integration system as instantiated and utilized in the demonstration of the design principles.



(a) (b) (c)

Figure 10.2: Building of campus of Technical University Munich as (a) IFC model (b) CityGML model LOD3, and (c) CityGML model LOD2 visualized with FZKViewer.

The software products **Rhinoceros 3D and Grasshopper** were chosen for the integration of the source models due to three major reasons: First, Rhinoceros 3D allows to represent both implicit and explicit geometries so that geometrical objects of both IFC and CityGML can be represented. Second, the matching of the objects is mainly based on their geometrical description which can be visualized in Rhinoceros 3D so that the user receives instant visual feedback about the created links. Third, the visual programming environment provided by Grasshopper allows to easily create user interfaces for specifying variables relevant for the matching of geometric objects.

The **developed script** (Figure 10.3) for the link creation in Grasshopper is subdivided into three groups of components.

- The first group (light grey) covers the components for specifying the input variables. Here, the components refer to variables about the import like folder paths, CityGML version, and building ID, variables for the placement and dimensioning of the CityGML model, and variables for the alignment creation like the connection details to the SPARQL endpoint, used prefix in the alignment and name of the alignment. Furthermore, variables relevant for context-sensitive linking are defined which is further described in the chapters on the demonstration of the developed design principles (chapters 10.2 and 10.3).
- The second group (dark grey) covers the components of object matching and link creation. These components refer to five procedural steps: First, retrieving the geometrical objects from the respective models. Second, placement of the geometrical objects from the CityGML model with respect to those from the IFC model. Third, geometry creation is based on the retrieved geometrical objects. Fourth, matching of corresponding geometrical objects. Fifth, alignment creation and sending the alignment through the SPARQL endpoint to the triple store Apache Jena.
- The third group (blue) covers the components for representing and visualizing the aligned objects. On the left side of the group, the aligned objects are represented through their GUIDs. On the right side, an individual triple of the alignment can be selected and the colors for the representation of the related objects by the individual triple in Rhinoceros 3D can be chosen.

Noteworthy, the purpose of the script is the demonstration of the developed design principles but not its use in the industry which is why some drawbacks like processing time were not further investigated. The developed script can be accessed online [295].

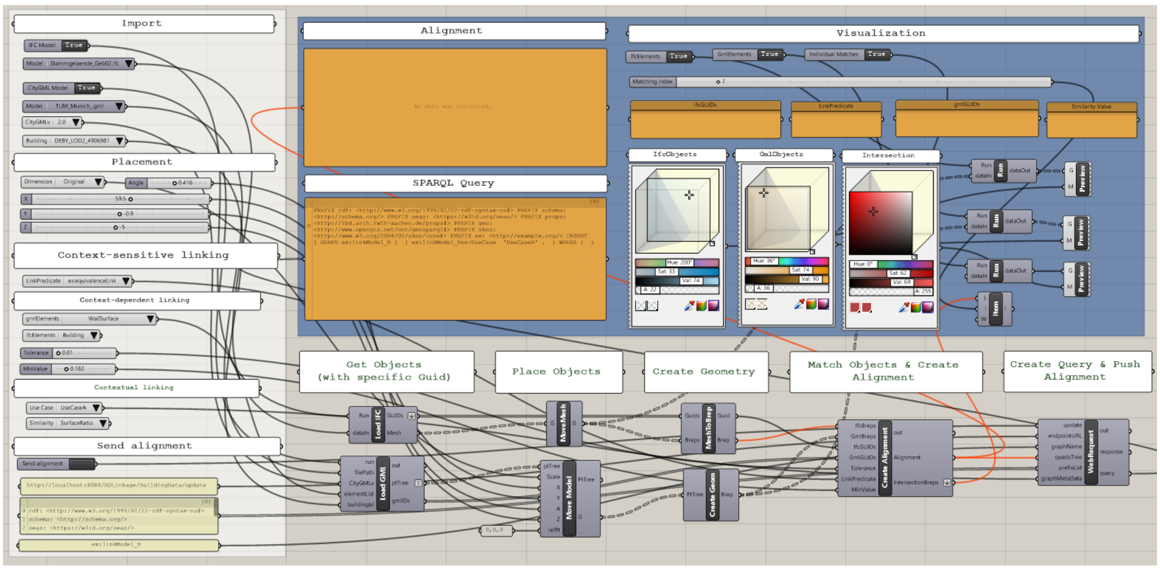


Figure 10.3: The developed grasshopper script for importing data from IFC and CityGML files, matching corresponding objects, SPARQL alignment creation and pushing the alignment based on the SPARQL endpoint.

The matching criteria used for the link creation refer to a schema-level alignment and a similarity value. The schema-level alignment is specified through the class of the CityGML and the IFC object which shall be matched. The schema-level alignment used in the demonstrations are represented in Table 10.1. The similarity value defines the spatial similarity of the corresponding objects and further described in chapter 10.3. The similarity value is used as matching criteria since the thesis is limited to equivalence correspondences. In this dissertation, the similarity measurement is based on the proportion between the overlap of corresponding wall objects in IFC and CityGML and the maximum surface of the wall object in IFC (Figure 10.4). Thus, a similarity value about one means ‘full match’ and zero means ‘no match’.

In the link creation, the object placement is performed manually, the matching is performed automatically, and the query formulation is performed semi-automatically.

- In the step *Object placement*, the user places the IFC and CityGML models in Rhinoceros.
- In the step *Matching,* the corresponding objects are automatically identified based on the matching criteria.
- In the step *Query formulation,* the correspondence is made explicit through a SPARQL statement for the alignment creation. In some circumstance, the created SPARQL statement needs to be adapted manually.

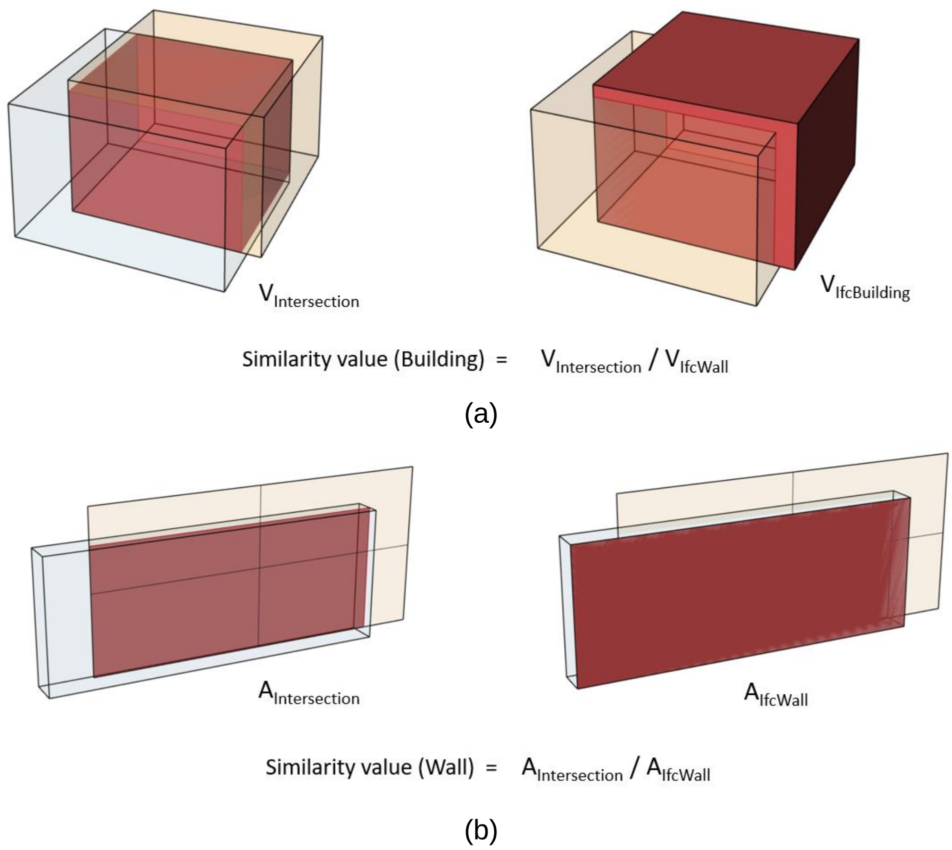


Figure 10.4: Similarity measurement describing the relation of two corresponding (a) building objects and (b) wall objects.

Table 10.1: Schema level alignments used for the instance level matching in the respective integration scenarios

Alignment		Integration Scenario
IFC class	**CityGML class**	
IfcBuilding	*Building*	*A.1, C.1-3*
IfcStair	*BuildingInstallation*	*A.2*
IfcDoor	*Door*	*B.1*
IfcWall	*BuildingConstructiveElement*	*B.2*
IfcWall	*WallSurface*	*D.1-3*

The alignment is created through a SPARQL statement in which the corresponding objects of the source models are specified through their GUIDs (Figure 10.7). In more detail, the query is based on PREFIX, INSERT, USING, and WHERE blocks. The PREFIX block specifies the utilized prefix of the vocabulary. The PREFIX block used throughout the examples is described below and referenced through the symbol [PREFIX]. In the INSERT block, the links are defined and assigned to a named graph.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX ex: <http://example.org/>
PREFIX schema: <http://schema.org/>
PREFIX seas: <https://w3id.org/seas/>
PREFIX props: <http://lbd.arch.rwth-aachen.de/props#>
PREFIX bot: <https://w3id.org/bot#>
```

Figure 10.5: PREFIX Block used in the SPARQL queries throughout the demonstration and further referenced by [PREFIX].

The link relating to two objects is expressed through the link predicate *ex:equivalenceLink*. The USING block specifies the named graphs which shall be accessed through the SPARQL queries. The WHERE block identifies the relating objects based on their GUIDs. Notably, the SPARQL queries are manually adapted after their automatic creation.

The integrated model is queried through SPARQL queries created through the web user interface provided by Fuseki (Figure 10.10). the query is based on PREFIX, SELECT, FROM, and WHERE blocks. The PREFIX and WHERE blocks are similar to the SPARQL query for the alignment creation. The SELECT block specifies the objects of interest and the FROM block specifies the named graphs which shall be accessed through the SPARQL queries, like the USING block as described above.

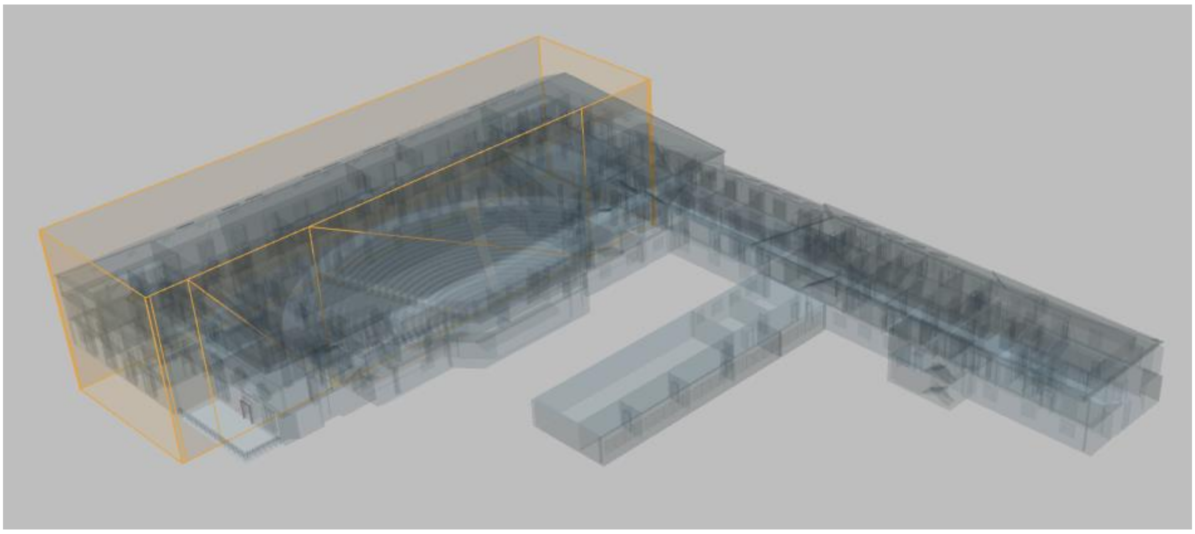


Figure 10.6: Matched IFC model and building of CityGML model LOD2 in Rhinoceros 3D.

```
INSERT {
  GRAPH ex:linkModel_A1 { ?ifcbuildingelement ex:equivalenceLink ?gmlbuildingelement . }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD3
WHERE {
   ?state_globalIdIfcRoot schema:value "AjTu972_dyO001000009iq" .
   ?globalIdIfcRoot seas:evaluation ?state_globalIdIfcRoot .
   ?ifcbuildingelement props:globalIdIfcRoot ?globalIdIfcRoot .
   ?gmlbuildingelement rdfs:label     "id090b25a1-fc5f-4a45-bde1-40fa09ee7ca7"@en .
}
```

Figure 10.7: Exemplary SPARQL query creating an alignment between the building and city model. The query was created through the script but manually adapted for visualization purposes.

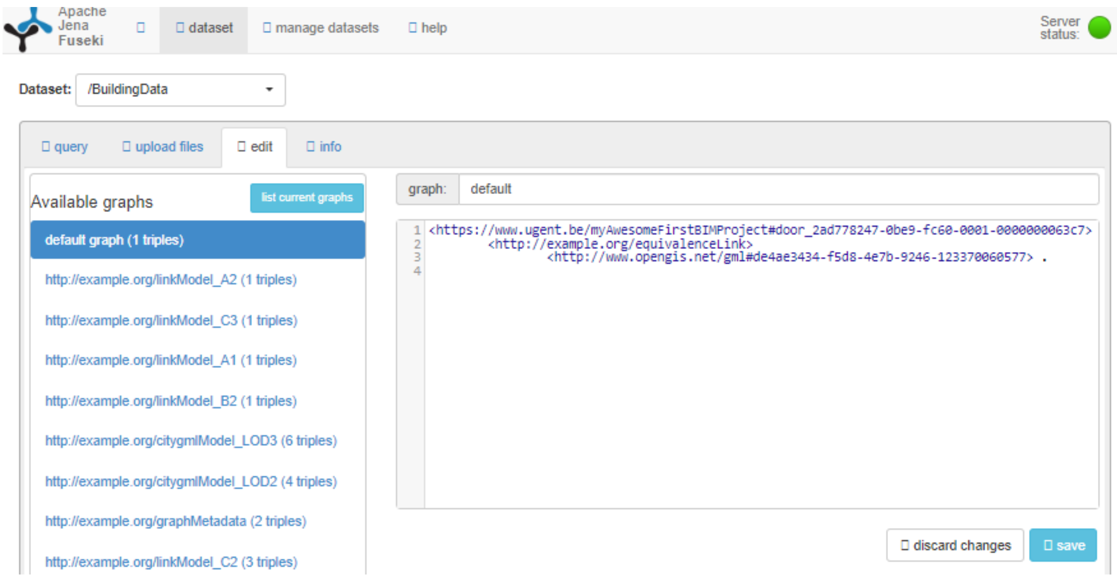


Figure 10.8: The integrated models as utilized in the demonstration scenarios composed of source models and alignment. Source models refer to ifcModel, citygmlModel_LOD2 and citygmlModel_LOD3. Alignments refer to the respective linkModels.

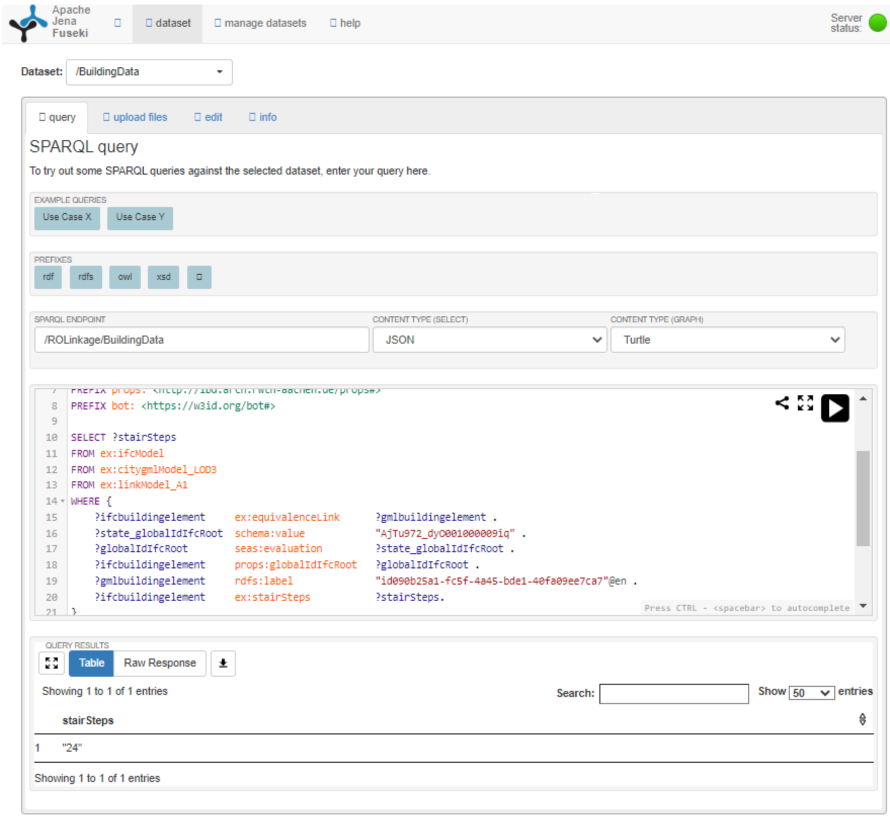


Figure 10.9: SPARQL query in the web user interface provided by Apache Jena as utilized in the demonstration.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX ex: <http://example.org/>
PREFIX schema: <http://schema.org/>
PREFIX seas: <https://w3id.org/seas/>
PREFIX props: <http://lbd.arch.rwth-aachen.de/props#>
PREFIX bot: <https://w3id.org/bot#>
PREFIX example: <http://www.example.org/rdf#>

SELECT ?stairSteps
FROM ex:ifcModel
FROM ex:citygmlModel_LOD3
FROM ex:linkModel_A1
WHERE {
    ?ifcbuildingelement     ex:equivalenceLink      ?gmlbuildingelement .
    ?state_globalIdIfcRoot  schema:value            "AjTu972_dy0001000009iq" .
    ?globalIdIfcRoot        seas:evaluation         ?state_globalIdIfcRoot .
    ?ifcbuildingelement     props:globalIdIfcRoot   ?globalIdIfcRoot .
    ?gmlbuildingelement     rdfs:label              "id090b25a1-fc5f-4a45-bde1-40fa09ee7ca7"@en .
    ?ifcbuildingelement     ex:stairSteps           ?stairSteps.
}
```

Figure 10.10: Exemplary SPARQL query for retrieving relevant data composed of PREFIX, SELECT, FROM, and WHERE blocks represented in the web user interface by Apache Jena.

10.2 Design Principle 1: Context-dependent linking

Note: The chapters Demonstration and Evaluation are adapted to a large extent from Beck et al. [3]. Beck et al. have described three demonstration scenarios while in this chapter only two demonstration scenarios are described. This is because of the different understanding of contextual/ context-dependent linking compared to Beck et al.

10.2.1 Description

As described in chapter 4.3, the **design principles** are composed of premises and consequences. The premises describe the situation of an information integration system in which the design principle holds. The consequences describe how to design software products aiming at linking information at instance level with respect to this situation. In the following, two design principles are described following the previous discourse about context-sensitive linking, namely context-dependent linking and contextual linking. Both design principles are limited to information integration systems in which heterogeneous source models are linked at instance level, and to equivalence links relating corresponding objects of these source models.

The first design principle (DP1) refers to **context-dependent linking** which applies in context-sensitive information integration systems with variable input for the integration system. The variable input refers to both the vocabulary of the source models and the interpretation belonging to the function from the application system. In such kind of information integration systems, the alignment and its validity scope must be adapted with respect to these variables. Therefore, *context-dependent linking* requires integration systems which the creation of alignments depending on the variable input, namely vocabulary, and interpretation. This kind of requirement can be met through an adaptive matching mechanism.

Table 10.2: Premises and consequences for the design principles on context-dependent linking based on different terminologies.

Premises	Variable input of integration system, such as variable vocabulary (source models) or variable interpretation (functions)
Consequence	Adaptive matching mechanism

10.2.2 Implementation

The design principle of context-dependent linking is instantiated in the demonstration environment. Here, the integration system was extended to allow adaptive matching depending on different application and model-oriented aspects. In more detail, a user interface was developed which allows to define schema-level alignments relating two classes of both the IFC model and CityGML model. The user interface allows to select a class of the IFC model and the corresponding class of the CityGML, such as *IfcStair* and *BuildingInstallation*, and based on this schema-level alignment, the matching is performed. The similarity value is not in the major focus in the following integration scenarios so that a similarity value larger than zero indicates a match. Instead, the following demonstration shows that different situations require different schema-level alignments.

10.2.3 Demonstration

The developed design principles on context-dependent linking are demonstrated through conducting integration scenarios differing regarding application or model-oriented aspects. The instance-level alignments of two integration scenarios differing regarding model-oriented or application-oriented aspects are compared to show the availability of context-dependent linking. The comparison of two integration scenarios is further called *Demo*. In the following, two Demos are described, namely *application-driven contextual linking (Demo A)* and *model-driven contextual linking (Demo B).*

In Demo A, the IFC model and the converted CityGML LOD3 model representing the same building of the campus of Technical University Munich were linked. In both models, the stairs are represented but only the IFC model covers the number of stair steps of the respective stairs. Noteworthy, the number of stair steps was manually added to the respective objects in the RDF graph of the BOT model through the predicate *ex:hasStairSteps*.

- *Integration scenario A.1:* In the first integration scenario (A.1), the total amount of all stair steps from a specific building of the building complex was queried. This integration scenario refers to the use case *Asset management*. The query requires the linkage of the *IfcBuilding* instance represented in the IFC model to the respective *Building* instance of the CityGML model. In Figure 10.11 the SPARQL statement which was applied for the link creation is represented. In Figure 10.12, the relevant subset of the consequent RDF graph and the building objects are represented, while the corresponding object are colorized. In Figure 10.13, the SPARQL query which was applied to retrieve the relevant information is represented.

```
INSERT {
  GRAPH ex:linkModel_A1 { ?ifcbuildingelement   ex:equivalenceLink    ?gmlbuildingelement . }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD3
WHERE {
        ?state_globalIdIfcRoot      schema:value            "3gh05gIW9CX8bf7zut6IhH"  .
        ?globalIdIfcRoot            seas:evaluation         ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot   ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label              "id6601bc23-ed03-4869-b9c3-68f25ef91360"@en .
}
```

Figure 10.11: SPARQL query for linking corresponding IfcBuilding instance of the IFC model and Building instance of the CityGML LOD3 model (Integration scenario A.1).

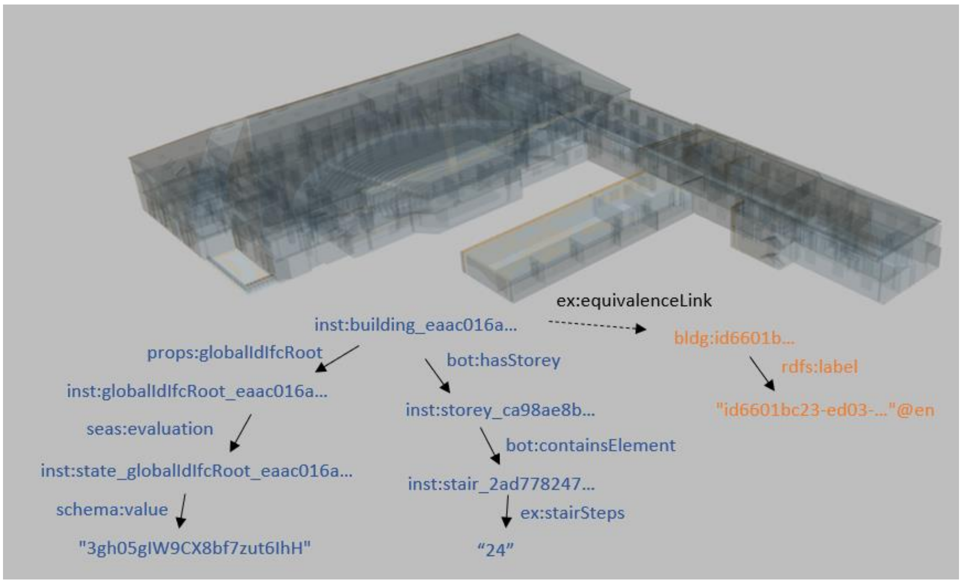


Figure 10.12: Linked IfcBuilding instance of the IFC model and Building instance of the CityGML LOD3 model for integration scenario A.1 and relevant subsets of the respective RDF graphs.

```
SELECT ?stairSteps
FROM ex:ifcModel
FROM ex:citygmlModel_LOD3
FROM ex:linkModel_A1
WHERE {
        ?ifcbuildingelement         ex:equivalenceLink        ?gmlbuildingelement .
        ?state_globalIdIfcRoot      schema:value              "3gh05gIW9CX8bf7zut6IhH" .
        ?globalIdIfcRoot            seas:evaluation           ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot     ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label                "id6601bc23-ed03-4869-b9c3-68f25ef91360"@en .
        ?ifcbuildingelement         bot:hasStorey             ?storey .
        ?storey                     bot:containsElement       ?ifcbuildingelement2 .
        ?ifcbuildingelement2        ex:stairSteps             ?stairSteps .
}

Result: "24"
```

Figure 10.13: SPARQL query for retrieving the number of stair steps of the building object of the CityGML LOD3 model (Integration scenario A.1).

- *Integration scenario A.2*: In the second integration scenario (A.2), the number of stair steps on a specific path in the building complex was queried. This integration scenario refers to the use case *Navigation & Evacuation* and requires linking the *IfcStair* instances of the IFC model to the corresponding *BuildingInstallation* instances of the CityGML model located at the respective path. In Figure 10.14 the SPARQL statement which was applied for the link creation is represented. In Figure 10.15, the relevant subset of the consequent RDF graph and the building objects are represented, while the corresponding object are colorized. In Figure 10.16, the SPARQL query which was applied to retrieve the relevant information is represented.

```
INSERT {
  GRAPH ex:linkModel_A2 { ?ifcbuildingelement   ex:equivalenceLink   ?gmlbuildingelement . }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD3
WHERE {
        ?state_globalIdIfcRoot        schema:value              "AjTu972_dyO001000009iq" .
        ?globalIdIfcRoot              seas:evaluation           ?state_globalIdIfcRoot .
        ?ifcbuildingelement           props:globalIdIfcRoot     ?globalIdIfcRoot .
        ?gmlbuildingelement           rdfs:label                "id090b25a1-fc5f-4a45-bde1-40fa09ee7ca7"@en .
}
```

Figure 10.14: SPARQL statement for linking IfcStair instances of the IFC model to the corresponding BuildingInstallation instances of the CityGML LOD3 model (Integration scenario A.2).

.

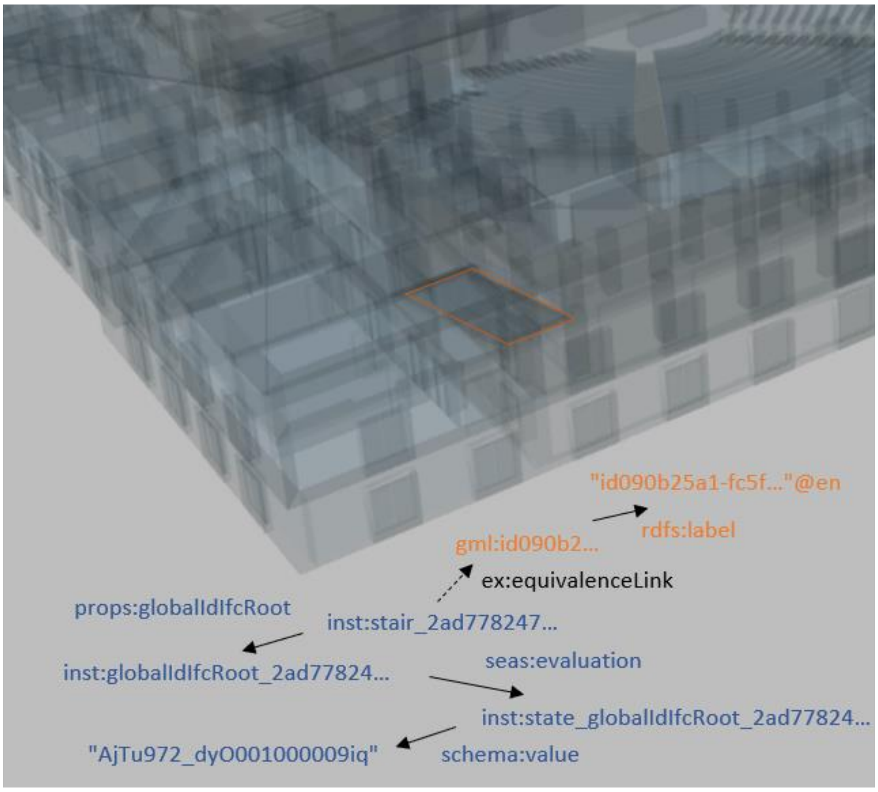


Figure 10.15: Linked IfcStair instance of the IFC model and BuildingInstallation instance of CityGML LOD3 model for integration scenario A.2 and relevant subsets of the respective RDF graphs.

```
SELECT ?stairSteps
FROM ex:ifcModel
FROM ex:citygmlModel_LOD3
FROM ex:linkModel_A2
WHERE {
        ?ifcbuildingelement          ex:equivalenceLink          ?gmlbuildingelement .
        ?state_globalIdIfcRoot       schema:value                "AjTu972_dyO001000009iq" .
        ?globalIdIfcRoot             seas:evaluation             ?state_globalIdIfcRoot .
        ?ifcbuildingelement          props:globalIdIfcRoot       ?globalIdIfcRoot .
        ?gmlbuildingelement          rdfs:label             "id090b25a1-fc5f-4a45-bde1-40fa09ee7ca7"@en .
        ?ifcbuildingelement          ex:stairSteps               ?stairSteps.
}

Result: "24"
```

Figure 10.16: SPARQL query for querying the number of stair steps of the BuildingInstallation instance of CityGML LOD3 model which represent the stair (Integration scenario A.2).

In Demo B, indoor information provided by the IFC model and outdoor information provided by a CityGML LOD3 model were linked for *Navigation & Evacuation* simulation. This kind of simulation requires both indoor and outdoor information and is a common example of BIM-GIS Integration.

- *Integration scenario B.1*: In the first integration scenario (B.1), the CityGML model is based on LOD3 what means that the openings of the buildings are represented in the model. Thus, the exit doors represented in the IFC model were linked to the corresponding door objects represented in the CityGML model. In more detail, *IfcDoors* instances of the IFC model were linked to the corresponding *Door* instances of the CityGML model. In Figure 10.17 the SPARQL statement which was applied for the link creation is represented. In Figure 10.18, the relevant subset of the consequent RDF graph and the building objects are represented, while the corresponding object are colorized. In Figure 10.19, the SPARQL query which was applied to retrieve the relevant information is represented.

```
INSERT {
  GRAPH ex:linkModel_B1 { ?ifcbuildingelement   ex:equivalenceLink   ?gmlbuildingelement .  }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD3
WHERE {
        ?state_globalIdIfcRoot       schema:value                "AjTu972_dyO001000006F7" .
        ?globalIdIfcRoot             seas:evaluation             ?state_globalIdIfcRoot .
        ?ifcbuildingelement          props:globalIdIfcRoot       ?globalIdIfcRoot .
        ?gmlbuildingelement          rdfs:label          "ide63a5a6d-5f9f-4f50-aff3-5d3162255fb0"@en .
}
```

Figure 10.17: SPARQL statement for linking IfcDoors instances of the IFC model to the corresponding Door instances of the CityGML LOD3 model (Integration scenario B.1). The Guid of IfcObject refers to a different door since not all Objects are provided in the .ttl file of the IFC Model.

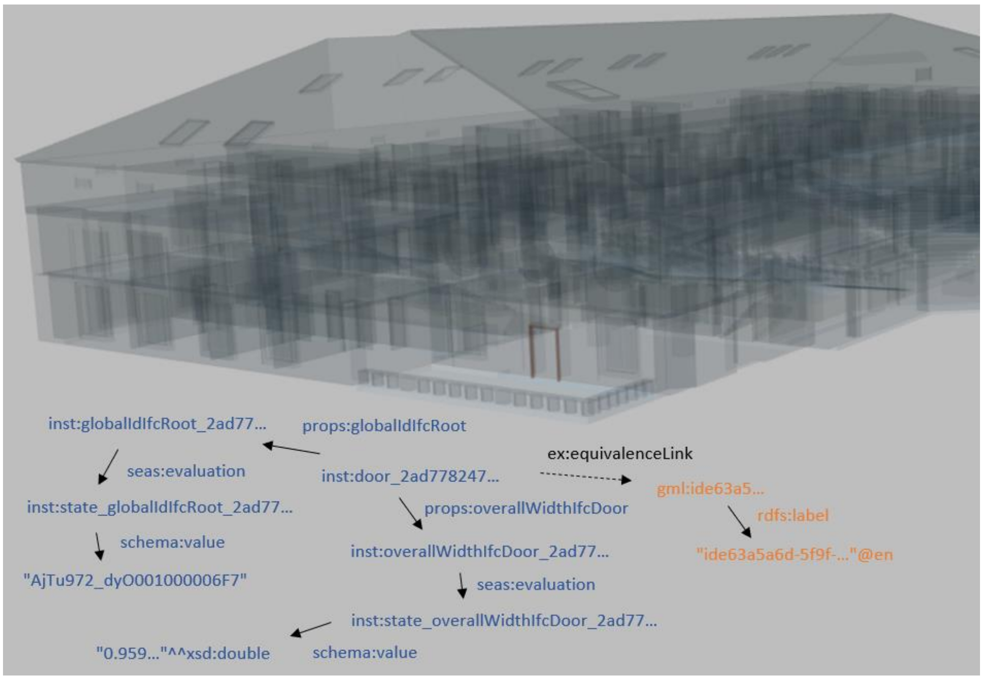


Figure 10.18: Linked IfcDoors instances of the IFC model to the corresponding Door instances of the CityGML LOD3 model for integration scenario B.1, and relevant subsets of the respective RDF graphs.

```
SELECT ?widthDoor
FROM ex:ifcModel
FROM ex:citygmlModel_LOD3
FROM ex:linkModel_B1
WHERE {
        ?ifcbuildingelement          ex:equivalenceLink          ?gmlbuildingelement .
        ?state_globalIdIfcRoot       schema:value                "AjTu972_dyO001000006F7" .
        ?globalIdIfcRoot             seas:evaluation             ?state_globalIdIfcRoot .
        ?ifcbuildingelement          props:globalIdIfcRoot       ?globalIdIfcRoot .
        ?gmlbuildingelement          rdfs:label          "ide63a5a6d-5f9f-4f50-aff3-5d3162255fb0"@en .
        ?ifcbuildingelement          props:overallWidthIfcDoor   ?overAllWidthIfcDoor .
        ?overAllWidthIfcDoor         seas:evaluation             ?state_overAllWidthIfcDoor .
        ?state_overAllWidthIfcDoor   schema:value                ?widthDoor .
}

Result: "1.439523661459987"^^xsd:double
```

Figure 10.19: SPARQL query for querying the width of Door object of the CityGML LOD3 model (Integration scenario B.1).

- *Integration scenario B.2:* In the second integration scenario (B.2), the CityGML model is based on LOD2 which means that it does not cover openings like doors. Therefore, the wall object represented in the IFC model were linked to the respective wall object in the CityGML model. In more detail, *IfcWall* instances of the IFC model were linked to the corresponding *BuildingConstructiveElement* instances of the CityGML model. In Figure 10.20, the SPARQL statement which was applied for the link creation is represented. In Figure 10.21, the relevant subset of the consequent RDF graph and the building objects are represented, while the corresponding object are colorized. In Figure 10.22, the SPARQL query which was applied to retrieve the relevant information is represented.

```
INSERT {
  GRAPH ex:linkModel_B2 { ?ifcbuildingelement   ex:equivalenceLink   ?gmlbuildingelement .  }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD2
WHERE {
        ?state_globalIdIfcRoot      schema:value             "AjTu972_dyO001000005xb" .
        ?globalIdIfcRoot            seas:evaluation          ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot    ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label               "DEBY_LOD2_4906978_c19dff78-452e-4ed7-
8eaa-cf1f76c2afe4"  .
}
```

Figure 10.20: SPARQL statement for linking corresponding IfcWall instances of the IFC model to the corresponding BuildingConstructiveElement instances of the CityGML LOD2 model (Integration scenario B.2).

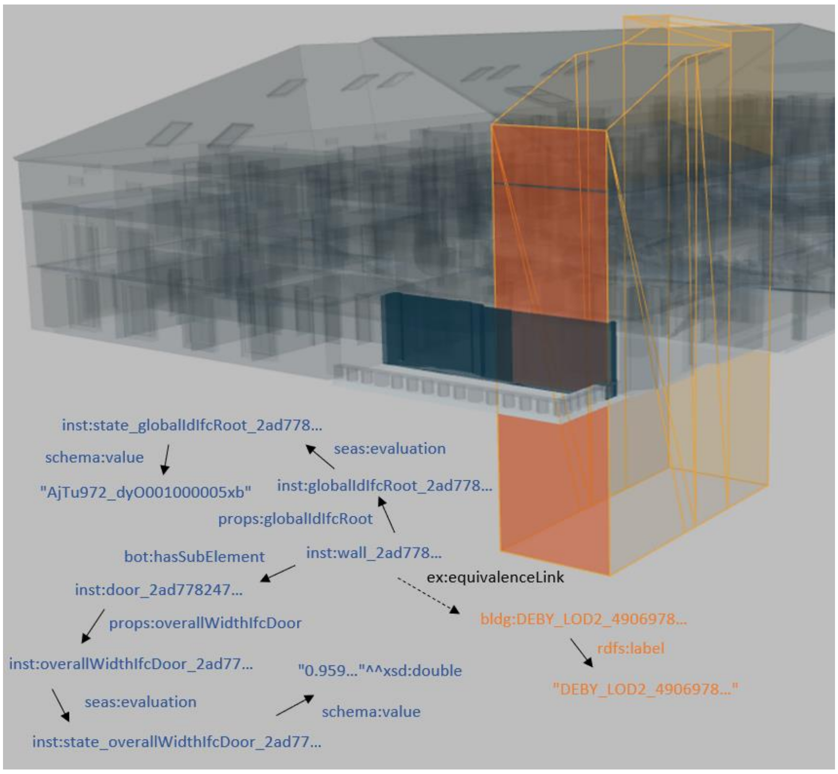


Figure 10.21: Linked IfcWall instances of the IFC model to the corresponding BuildingConstructiveElement instances of the CityGML LOD2 model for integration scenario B.2, and relevant subsets of the respective RDF graphs.

```
SELECT ?widthDoor
FROM ex:ifcModel
FROM ex:citygmlModel_LOD3
FROM ex:linkModel_B2
WHERE {
	?ifcbuildingelement	ex:equivalenceLink	?gmlbuildingelement .
	?state_globalIdIfcRoot	schema:value	"AjTu972_dyO001000005xb" .
	?globalIdIfcRoot	seas:evaluation	?state_globalIdIfcRoot .
	?ifcbuildingelement	props:globalIdIfcRoot	?globalIdIfcRoot .
	?gmlbuildingelement	rdfs:label	"DEBY_LOD2_4906978_c19dff78-452e-4ed7-
8eaa-cf1f76c2afe4"  .
	?ifcbuildingelement	bot:hasSubElement	?ifcDoor .
	?ifcDoor	props:overallWidthIfcDoor	?overAllWidthIfcDoor .
	?overAllWidthIfcDoor	seas:evaluation	?state_overAllWidthIfcDoor .
	?state_overAllWidthIfcDoor	schema:value	?widthDoor .
}

Result: "1.439523661459987"^^xsd:double
```

Figure 10.22: SPARQL query for querying the width of the Door object related to the BuildingConstructiveElement instance of the CityGML LOD2 model (Integration scenario B.2).

10.2.4 Evaluation

The design principle of context-dependent linking are evaluated through the discussion of whether the premises and respective consequences were met. The **premise** of the design principle is that different input variables (model-oriented or application-oriented aspects) result in different alignments between heterogeneous instance models. In the following, this kind of premise is discussed with respect to Demo A and Demo B.

- Demo A: The integration scenarios refer to the same model-oriented aspects since they link the same source models. On the other hand, the integration scenarios differ regarding the application-oriented aspect *granularity*. While querying the total amount of stair steps requires the link between building objects (A.1), querying the number of stair steps of a single building requires relations between particular stair objects (A.2). Thus, the first integration scenario considers $\langle s^{I_C}, o'^{I_C} \rangle \in I_{EXT}(p'^{I_C})$ as “true” with s being *inst:Building*, o' representing the number of stair steps of *inst:IfcProject_25* and p' relating the number of stairsteps to the building. In contrast to that, the second integration considers $\langle s^{I_{C^\circ}}, o'^{I_{C^\circ}} \rangle \in I_{EXT}(p'^{I_{C^\circ}})$ as “true” with s representing the single stairs in terms of *inst:BuildingInstallation*, o' representing the number of stair steps of single stairs in *inst:IfcProject_25* and p' relating the number of stairsteps to the stair objects.
- Demo B: The use case is the same for both integration scenarios so that the application-oriented aspects do not differ between the integration scenarios. However, the integration scenarios are based on different variants of the CityGML model. While the first integration scenario (B.1) refers to a CityGML model with LOD2, the second integration scenario (B.2) refers to a CityGML model with LOD3. The change from LOD2 to LOD3 allows linking at a

higher detail level to get more accurate results for the navigation scenario. Thus, the first integration scenario considers $\langle s^{I_C}, o'^{I_C}\rangle \in I_{EXT}(p'^{I_C})$ as ‘true’ with s representing *inst:door*, o' representing the door type of *inst:IfcDoor,* and p' relating door type to the door object. The second scenario considers $\langle s^{\circ I_C}, o'^{I_C}\rangle \in I_{EXT}(p'^{I_C})$ as “true” with s° representing *inst:WallSurface*, o' representing the door type of inst:IfcDoor, and p' relating door type to the door object.

In Table 10.3, Demo A and Demo B are described with respect to the model-oriented and application-oriented aspects and their alignment differences. In summary, all Demos refer to context-dependent linking since their integration scenarios differ regarding some model or application-oriented aspects and their alignments.

The **consequence** of the design principle is the need for an adaptive matching approach to creating the alignments in these integration scenarios. This adaptive matching approach was prototyped through the development of a user interface allowing the user to define the instance-level alignment depending on the respective situation. In summary, the developed user interface has enabled link creation, so that all integration scenarios could be carried out successfully.

Table 10.3: Comparison of integration scenarios of the Demos regarding model-oriented and application-oriented aspects [3].

	Demo A		**Demo B**	
	Scenario 1	Scenario 2	Scenario 1	Scenario 2
Model-oriented			CityGML LOD3 (with doors)	CityGML LOD2 (without doors)
Application-oriented	Query stair steps of building	Query stair steps on specific path		
Alignment	*IfcBuilding - Building*	*IfcStair - BuildingInstallation*	*IfcDoor - Door*	*IfcWall - BuildingConstructiveElement*

10.2.5 Conclusion

As a conclusion, different model-oriented and application-oriented aspects of the integration scenario require different instance-level alignments which refers to context-dependent linking. In context-dependent linking, there is no general matching approach, so that the alignment needs to be created depending on the situation. The alignment creation depending on the situation was achieved through the development of a user interface in which the user can adapt the matching criteria for the instance-level alignment. This design principle of contextual linking was successfully demonstrated through two demonstration scenarios. Thus, context-sensitive information integration systems in the field of BIM-GIS Integration based on the integration method linking require integration systems that allow adaptive matching when dealing with different source models or different functions.

10.3 Design Principle 2: Contextual linking

10.3.1 Description

The second design principle (DP2) refers to **contextual linking** which applies in context-sensitive information integration systems in which the implicit validity scope of the alignment is fixed and opaque to the user. Here, the links are prone to be misleading since the user does not know the validity scope of the alignment. In this kind of information integration system, the implicit validity scope must be made explicit so that the user can interpret the alignment correctly. Thus, contextual linking refers to making the implicit validity scope of an alignment or link explicit to reduce the risk of misinterpreting the link. More roughly spoken, the implicit semantics of the links must be made explicit in case the implicit semantics of the link are opaque to the user. The respective premise and consequence of the design principle is represented in Table 10.4.

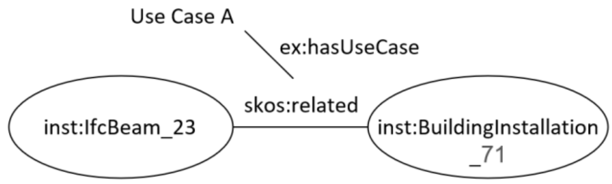
Table 10.4: Premises and consequences for the design principles on contextual linking based on different terminologies.

Premises	Implicit validity scope (semantics) of the link is opaque to the
Consequence	Making the validity scope (semantics) of the link explicit

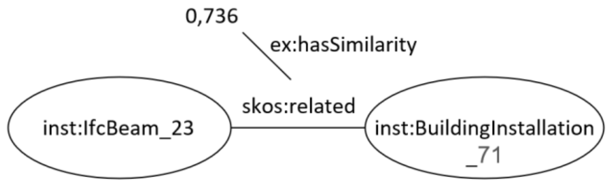
10.3.2 Implementation

The design principle of contextual linking is based on the premise that the validity scope of the link is opaque for the user and on the consequence that the implicit validity scope must be made explicit so that the user can interpret the link correctly. To the best of the authors' knowledge, making the implicit validity scope of a link explicit cannot be achieved in an absolute sense. This is because there is neither an approach making the exact meaning of the link explicit nor an approach specifying the exact situation to which the validity scope refers to. Thus, the **approaches** to contextual linking rather provide additional information aiming to reduce misinterpreting the link than making the validity scope explicit in an absolute sense. In general, there are two different types of these approaches:

- First, defining pre-defined use cases for which the link holds, further called *Use case-oriented approach* (Figure 10.23a).
- Second, defining the similarity between two objects, further called *Similarity-oriented approach* (Figure 10.23b).



(a)



(b)

Figure 10.23: Describing the validity scope of the links through (a) Use case-oriented approach (b) Similarity-oriented approach.

Both approaches have in common that they do not specify which information holds for the corresponding object (see section 8.2) but provide additional information to facilitate this kind of evaluation by the user. Furthermore, both approaches have in common that they require an approach to enrich the alignment (Use case-oriented approach) or the link (Similarity-oriented approach) with **metadata**. Here, the enrichment of the link is achieved through the metadata approach *SingletonProperties* [236]. Following the approach of *SingletonProperties*, the link is expressed through a proxy relation (Figure 10.24. and Figure 10.31). This proxy relation is the subject of two additional triples: First, the triple relating to the actual link predicate *ex:equivalenceLink* through the predicate *rdf:singletonPropertyOf*. Second, the triple specifying the metadata of the link what is here achieved through the predicate *ex:hasSimilarityValue* or *ex:hasUseCase*.

The ***Use case-oriented approach*** is implemented through a user interface allowing to manually enrich the alignment with some vocabulary specifying a use case. The standardization of respective use cases for the use case-oriented approach is not addressed in this research work but is considered as a relevant future work. The use case-oriented approach is related to contextual linking approaches in the field of Semantic Web aiming to limit the validity scope of the graph to a specific situation [217,219,225,226,228], as described in chapter 3.3.

The ***Similarity-oriented approach*** is implemented through a user interface allowing to specify similarity values relevant to the interpretation of the alignment. In research fields related to CAD, there are several approaches aiming to characterize the similarity between geometrical objects also referenced by the keywords *feature recognition* or *shape signature* [296]. Most commonly, these approaches were developed to support the retrieval of geometrical objects from CAD model libraries. An often-cited approach is the Optiz code which is a hybrid code describing the geometrical features of a CAD object based on a maximum of 14 digits [297]. Further similarity recognition methods for geometrical objects of CAD models are summarized by Zhetaban et al. [296].

10.3.3 Demonstration

The **demonstration** covers two different integration scenarios called Demos. Here, a Demo is based on three different linking approaches: First, linking two objects without further specifying the meaning of the link through ex:*equivalenceLink*. Second, enriching the link with information about the use case for which the link holds (Use case-oriented approach). Third, enriching the link with information specifying the similarity of the related objects (Similarity-oriented approach). In both Demos, the related objects refer to the same physical object, but their geometrical representations do not match exactly. Furthermore, it is assumed that the linked information is opaque to the user of the link (the person who queries the linked information) which is the premise for contextual linking.

In **Demo C**, the *IfcBuilding* instance of the IFC model is related to the corresponding *Building* instance of the CityGML model LOD2 which is based on different geometrical dimensions (Figure 10.24). The building instance of the CityGML model refers to a part of the *IfcBuilding* instance of the IFC model. In Demo C, the total number of stair steps of the building of the CityGML model shall be queried while the number of stair steps is attached to the IFC model. Noteworthy, the triple specifying the number of stair steps in the IFC model was manually created.

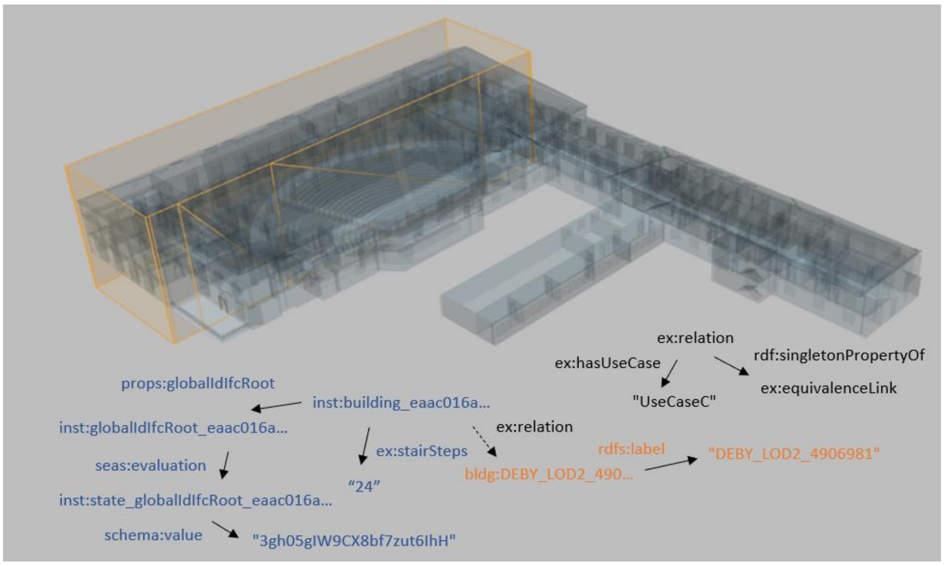


Figure 10.24: Linked IfcBuilding instance of the IFC model to the corresponding Building instance of the CityGML LOD2 model utilized for the Demo C, and relevant subsets of the respective RDF graphs.

- *Integration scenario C.1*: In the first integration scenario (C.1), the corresponding *IfcBuilding* instance of the IFC model and the *Building* instance of the CityGML model are linked through the link predicate *ex:equivalenceLink*. The applied SPARQL statement for the link creation is represented in Figure 10.25 and the applied SPARQL query for querying the relevant information is represented in Figure 10.26.

```
INSERT {
  GRAPH ex:linkModel_C1 { ?ifcbuildingelement   ex:equivalenceLink   ?gmlbuildingelement . }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD2
WHERE {
        ?state_globalIdIfcRoot      schema:value              "3gh05gIW9CX8bf7zut6IhH" .
        ?globalIdIfcRoot            seas:evaluation           ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot     ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label                "DEBY_LOD2_4906981" .
}
```

Figure 10.25: SPARQL statement for linking the IfcBuilding instance of the IFC model to the corresponding Building instance of the CityGML LOD2 model (Integration scenario C.1).

```
SELECT ?stairSteps
FROM ex:ifcModel
FROM ex:citygmlModel_LOD2
FROM ex:linkModel_C1
WHERE {
        ?ifcbuildingelement         ex:equivalenceLink        ?gmlbuildingelement .
        ?state_globalIdIfcRoot      schema:value              "3gh05gIW9CX8bf7zut6IhH" .
        ?globalIdIfcRoot            seas:evaluation           ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot     ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label                "DEBY_LOD2_4906981" .
        ?ifcbuildingelement         ex:stairSteps             ?stairSteps.
}

Result:   "230”
```

Figure 10.26: SPARQL query for querying the number of stairsteps of the Building instance of the CityGML LOD2 model (Integration scenario C.1).

- *Integration scenario C.2*: In the second integration scenario (C.2), the link relating the objects is enriched through the similarity measurement specifying the proportion of the volumes of both objects. The applied SPARQL statement for the link creation is represented in Figure 10.27 and the applied SPARQL query for querying the relevant information is represented in Figure 10.28.

```
INSERT {
  GRAPH ex:linkModel_C2 { ?ifcbuildingelement   ex:relation      ?gmlbuildingelement .
                          ex:relation    rdf:singletonPropertyOf   ex:equivalenceLink ;
                                         ex:hasSimilarityValue          "0.32"^^xsd:double . }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD2
WHERE {
        ?state_globalIdIfcRoot      schema:value              "3gh05gIW9CX8bf7zut6IhH" .
        ?globalIdIfcRoot            seas:evaluation           ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot     ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label                "DEBY_LOD2_4906981" .
}
```

Figure 10.27: SPARQL statement for linking the IfcBuilding instance of the IFC model to the corresponding Building instance of the CityGML LOD2 model, and enriching the link with similarity value (Integration scenario C.2).

```
SELECT ?stairSteps  ?similarityValue
FROM ex:ifcModel
FROM ex:citygmlModel_LOD3
FROM ex:linkModel_C2
WHERE {
        ?ifcbuildingelement          ?equivalenceLink              ?gmlbuildingelement .
        ?state_globalIdIfcRoot       schema:value                  "3gh05gIW9CX8bf7zut6IhH" .
        ?globalIdIfcRoot             seas:evaluation               ?state_globalIdIfcRoot .
        ?ifcbuildingelement          props:globalIdIfcRoot         ?globalIdIfcRoot .
        ?gmlbuildingelement          rdfs:label                    "DEBY_LOD2_4906981" .
        ?ifcbuildingelement          ex:stairSteps                 ?stairSteps .
        ?equivalenceLink             rdf:singletonPropertyOf       ex:equivalenceLink .
        ?equivalenceLink             ex:hasSimilarityValue         ?similarityValue .
        }

Result:   "230"; "0.32"^^xsd:double
```

Figure 10.28: SPARQL query for querying the height of Building instance of the CityGML LOD2 model and the similarity value specifying the link relating the building objects (Integration scenario C.2).

- *Integration scenario C.3*: In the third integration scenario (C.3), the link relating the objects is enriched with information about the use case for which the link holds. Here, the querier wants to retrieve the number of stair steps belonging to the whole building of the IFC model for asset management purposes. The applied SPARQL statement for the link creation is represented in Figure 10.29 and the applied SPARQL query for querying the relevant information is represented in Figure 10.30.

```
INSERT {
  GRAPH ex:linkModel_C3 { ?ifcbuildingelement   ex:equivalenceLink   ?gmlbuildingelement . }
                          ex:relation     rdf:singletonPropertyOf      ex:equivalenceLink ;
                                          ex:hasUseCase                "UseCaseC" . }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD2
WHERE {
        ?state_globalIdIfcRoot       schema:value                  "3gh05gIW9CX8bf7zut6IhH" .
        ?globalIdIfcRoot             seas:evaluation               ?state_globalIdIfcRoot .
        ?ifcbuildingelement          props:globalIdIfcRoot         ?globalIdIfcRoot .
        ?gmlbuildingelement          rdfs:label                    "DEBY_LOD2_4906981" .
}
```

Figure 10.29: SPARQL statement for linking the IfcBuilding instance of the IFC model to the corresponding Building instance of the CityGML LOD2 model, and enriching the alignment through specifying the use case (Integration scenario C.3).

```
SELECT ?stairSteps
FROM ex:ifcModel
FROM ex:citygmlModel_LOD2
FROM ex:linkModel_C3
WHERE {
        ?ifcbuildingelement         ex:equivalenceLink          ?gmlbuildingelement .
        ?state_globalIdIfcRoot      schema:value                "3gh05gIW9CX8bf7zut6IhH" .
        ?globalIdIfcRoot            seas:evaluation             ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot       ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label                  "DEBY_LOD2_4906981" .
        ?ifcbuildingelement         ex:stairSteps               ?stairSteps .
        ?equivalenceLink            rdf:singletonPropertyOf     ex:equivalenceLink .
        ?equivalenceLink            ex:hasUseCase               "UseCaseC" .
}

Result:   “230”
```

Figure 10.30: SPARQL query for querying the height of the Building instance of the CityGML LOD2 model with respect to the use case (Integration scenario C.3).

In **Demo D**, a *IfcWall* instance of the IFC model is related to the corresponding *WallSurface* instance in the CityGML model LOD2 (Figure 10.31). The related objects of the IFC and CityGML models do not match exactly: While the *IfcWall* instance in the IFC model is represented as a solid object, the *WallSurface* instance in the CityGML model is represented through the wall surfaces. Furthermore, these wall objects are represented through different length and height values.

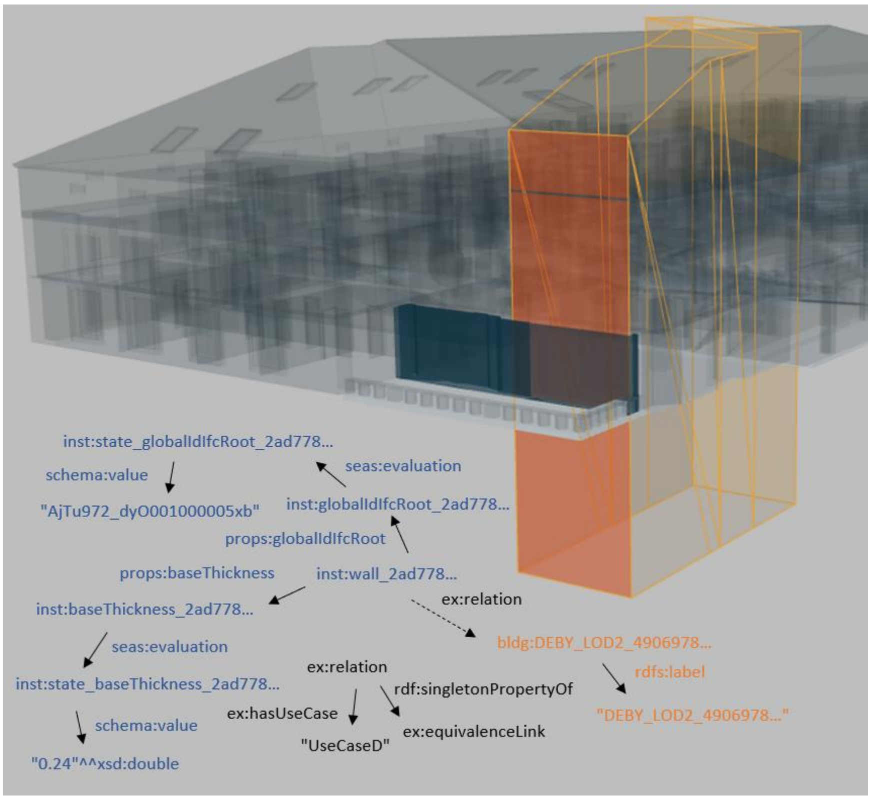


Figure 10.31: Linked IfcWall instance of the IFC model to the corresponding WallSurface instance in the CityGML LOD2 model as utilized for the Demo D and relevant subsets of the respective RDF graphs.

- *Integration scenario D.1*: In the first integration scenario (D.1), the wall objects are linked through the link predicate *ex:equivalenceLink*. The applied SPARQL statement for the link creation is represented in Figure 10.32 and the applied SPARQL query for querying the relevant information is represented in Figure 10.33.

```
INSERT {
  GRAPH ex:linkModel_D1 { ?ifcbuildingelement   ex:equivalenceLink    ?gmlbuildingelement . }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD2
WHERE {
        ?state_globalIdIfcRoot      schema:value              "AjTu972_dyO001000005xb" .
        ?globalIdIfcRoot            seas:evaluation           ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot     ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label                "DEBY_LOD2_4906978_c19dff78-452e-4ed7-
8eaa-cf1f76c2afe4"  .
}
```

Figure 10.32: SPARQL statement for linking the IfcWall instance of the IFC model to the corresponding WallSurface instance in the CityGML LOD2 model (Integration scenario D.1).

```
SELECT ?thickness
FROM ex:ifcModel
FROM ex:citygmlModel_LOD2
FROM ex:Alignment_D1
WHERE {
        ?ifcbuildingelement         ex:equivalenceLink        ?gmlbuildingelement .
        ?state_globalIdIfcRoot      schema:value              "AjTu972_dyO001000005xb" .
        ?globalIdIfcRoot            seas:evaluation           ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot     ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label                "DEBY_LOD2_4906978_c19dff78-452e-4ed7-
8eaa-cf1f76c2afe4"  .
        ?ifcbuildingelement         props:baseThickness       ?baseThickness .
        ?baseThickness              seas:evaluation           ?state_baseThickness .
        ?state_baseThickness        schema:value              ?thickness .
}

Result: "0.24"^^xsd:double
```

Figure 10.33: SPARQL query for querying the thickness of the WallSurface instance of the CityGML LOD2 model (Integration scenario D.1).

- *Integration scenario D.2*: In the second integration scenario (D.2), the link relating the objects is enriched through the similarity measurement specifying the proportion of the surfaces of the related wall objects. The applied SPARQL statement for the link creation is represented in Figure 10.34 and the applied SPARQL query for querying the relevant information is represented in Figure 10.35.

```
INSERT {
  GRAPH ex:linkModel_D2 { ?ifcbuildingelement   ex:relation      ?gmlbuildingelement .
                                    ex:relation    rdf:singletonPropertyOf  ex:equivalenceLink ;
                                                   ex:hasSimilarityValue          "0.27"^^xsd:double . }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD2
WHERE {
        ?state_globalIdIfcRoot        schema:value              "AjTu972_dyO001000005xb" .
        ?globalIdIfcRoot              seas:evaluation           ?state_globalIdIfcRoot .
        ?ifcbuildingelement           props:globalIdIfcRoot     ?globalIdIfcRoot .
        ?gmlbuildingelement           rdfs:label                "DEBY_LOD2_4906978_c19dff78-452e-4ed7-
8eaa-cf1f76c2afe4"  .
}
```

Figure 10.34: SPARQL statement for linking the IfcWall instance of the IFC model to the corresponding WallSurface instance in the CityGML LOD2 model and enriching the alignment through specifying the use case (Integration scenario D.2).

```
SELECT ?thickness  ?similarityValue
FROM ex:ifcModel
FROM ex:citygmlModel_LOD2
FROM ex:linkModel_D2
WHERE {
        ?ifcbuildingelement           ?equivalenceLink          ?gmlbuildingelement .
        ?state_globalIdIfcRoot        schema:value              "AjTu972_dyO001000005xb" .
        ?globalIdIfcRoot              seas:evaluation           ?state_globalIdIfcRoot .
        ?ifcbuildingelement           props:globalIdIfcRoot     ?globalIdIfcRoot .
        ?gmlbuildingelement           rdfs:label                "DEBY_LOD2_4906978_c19dff78-452e-4ed7-
8eaa-cf1f76c2afe4"  .
        ?ifcbuildingelement           props:baseThickness       ?baseThickness .
        ?baseThickness                seas:evaluation           ?state_baseThickness .
        ?state_baseThickness          schema:value              ?thickness .
        ?equivalenceLink              rdf:singletonPropertyOf   ex:equivalenceLink .
        ?equivalenceLink              ex:hasSimilarityValue     ?similarityValue .
}

Result: "0.24"^^xsd:double; "0.27"^^xsd:double
```

Figure 10.35: SPARQL query for querying the thickness of the WallSurface instance of the CityGML LOD2 model and the similarity value specifying the link relating the wall objects (Integration scenario D.2).

- *Integration scenario D.3*: In the third integration scenario (D.3), the link relating the objects is enriched with information about the use case for which the link holds. Here, the querier aims to retrieve the thickness of the wall object in the IFC model for structural engineering purposes. The applied SPARQL statement for the link creation is represented in Figure 10.36 and the applied SPARQL query for querying the relevant information is represented in Figure 10.37.

```
INSERT {
  GRAPH ex:linkModel_D3 { ?ifcbuildingelement   ex:equivalenceLink   ?gmlbuildingelement . }
  GRAPH ex:graphMetadata  { ex:linkModel_D3    ex:hasUseCase        "UseCaseD" . }
}
USING ex:ifcModel
USING ex:citygmlModel_LOD2
WHERE {
        ?state_globalIdIfcRoot      schema:value               "AjTu972_dyO001000005xb" .
        ?globalIdIfcRoot            seas:evaluation            ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot      ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label                 "DEBY_LOD2_4906978_c19dff78-452e-4ed7-
8eaa-cf1f76c2afe4"  .
}
```

Figure 10.36: SPARQL statement for linking the IfcWall instance of the IFC model to the corresponding WallSurface instance in the CityGML LOD2 model and enriching the alignment through specifying the use case (Integration scenario D.3).

```
SELECT ?thickness
FROM ex:ifcModel
FROM ex:citygmlModel_LOD2
FROM ex:linkModel_D3
WHERE {
        ?ifcbuildingelement         ex:equivalenceLink         ?gmlbuildingelement .
        ?state_globalIdIfcRoot      schema:value               "AjTu972_dyO001000005xb" .
        ?globalIdIfcRoot            seas:evaluation            ?state_globalIdIfcRoot .
        ?ifcbuildingelement         props:globalIdIfcRoot      ?globalIdIfcRoot .
        ?gmlbuildingelement         rdfs:label                 "DEBY_LOD2_4906978_c19dff78-452e-4ed7-
8eaa-cf1f76c2afe4"  .
        ?ifcbuildingelement         props:baseThickness        ?baseThickness .
        ?baseThickness              seas:evaluation            ?state_baseThickness .
        ?state_baseThickness        schema:value               ?thickness .
        ?equivalenceLink            rdf:singletonPropertyOf    ex:equivalenceLink .
        ?equivalenceLink            ex:hasUseCase              "UseCaseD" .
}

Result: "0.24"^^xsd:double
```

Figure 10.37: SPARQL query for querying the width of the WallSurface instance of the CityGML LOD2 model with respect to the use case (Integration scenario D.3).

10.3.4 Evaluation

As described previously, the design principles are evaluated through the discussion on whether the premises and respective consequences were met. The **premise** of the design principle of contextual linking is that the implicit validity scope of alignment is opaque for the user of the link. More roughly spoken, the user of the link does not know about the queried information. In both Demos, it was assumed that the user does not know whether the geometrical representations of the related objects match exactly. Therefore, the premise is considered as fulfilled for both Demos.

The **consequence** refers to making the implicit validity scope of the alignment explicit. In more detail, the interpretation and vocabulary for which the semantics of the equivalence link holds need to be attached to the alignment. To the best of the authors' knowledge, making the implicit validity scope of a link explicit cannot be achieved in an absolute manner. This is because there is neither an approach making the exact meaning of the link explicit nor an approach specifying the exact situation to which the validity scope refers to. Thus, two approaches aiming to reduce the risk of misinterpreting a link due to its opaque validity scope were applied, namely *Use case-oriented approach* and *Similarity-oriented approach*. In the *Use case-oriented approach* information about situation for which the link holds is attached to the alignment such as the specification of the use case. In the *Similarity-oriented approach* information about the similarity of linked objects is attached to the link. Both Demos have shown that the usage of these approaches can reduce misinterpretations of the link, and therefore support the design principle of contextual linking.

10.3.5 Conclusion

As a conclusion, an opaque validity scope of the alignment for the user of the alignment is prone for misinterpretations. To overcome the issue of misinterpretation, the implicit validity scope of the alignment must be made explicit. To the best of the author's knowledge, there is neither an approach to making the exact meaning of a link explicit nor an approach that specifies the exact situation to which the validity scope applies. Instead, the design principle was approximately implemented through two approaches aiming to reduce the risk of misinterpreting a link due to its opaque validity scope, namely *Use case-oriented approach* and *Similarity-oriented approach*. This approximated instantiation of the design principle of contextual linking was successfully demonstrated through two demonstration scenarios. More roughly speaking, this means that information integration systems in which the user does not know about the linked information require the enrichment of the alignment so that the risk of misinterpretations is reduced.

11 Discussion & future research

11.1 Summary

At the beginning of this dissertation, the research method ***Design Science Research*** and its application within the dissertation are described. In general, the goal of a research work according to Design Science Research is to acquire and communicate new research knowledge by creating an artifact. There are several different understandings on Design Science Research resulting in different procedural models in describing the operative research steps. This dissertation is based on the steps *Problem identification*, *Objective definition*, *Design & Development*, *Demonstration & Evaluation* and *Conclusion*. In this dissertation, the distinction between task-initiated and knowledge-initiated DSR is introduced. Most often, DSR research is task-oriented, meaning that the research problem is triggered by a problem in an innovative task. In contrast to that, in research works following knowledge-initiated DSR, the research problem results from the research gap identified by analyzing the state-of-the-art. This dissertation is based on knowledge-initiated DSR so that the description of the research problem follows the description of the state-of-the-art.

In the chapter ***Fundamentals***, the foundations for the further discourse on BIM-GIS Integration is described and relevant terms are defined. In the beginning of this chapter the concepts ontology, knowledge representation and information model are defined for the scope of the dissertation. In a nutshell, both information models and knowledge representations are ontologies but based on different formal languages, while ontologies refer to all kinds of representations of the real world. Moreover, common definitions of ontology, the term *formal,* and information models are defined. Afterward, formal languages, such as XML, EXPRESS, RDF, and information models relevant for the dissertation are described such as IFC, BOT and CityGML. Additionally, Building Information Modeling and Urban Information Modeling are described and subsequently compared. The comparison is based on the reviewed literature and can be categorized into *accumulation of differences*, *structured categorization of differences*, and *comparison of specific entities*. Notably, respective comparison in the reviewed literature often address the information models generally associated with BIM and GIS, such as IFC and CityGML. The most commonly described difference is the granularity of the physical objects that can be represented, such as buildings and cities. Furthermore, the keyword “BIM-GIS Integration” is critically discussed resulting in three major points of criticism:

- First, BIM and GIS are not comparable with respect to their actual meaning since BIM generally refers to the use of building information to reach a goal (“increase productivity”), while GIS refers to the information systems dealing with georeferenced information. In this regard, the keyword BIM-UIM Integration is rather adequate but limited to urban information. Nevertheless, BIM, GIS and UIM are comparable in terms of domains since they refer to specific functions and information.

- Second, the keyword “BIM-GIS Integration” is fuzzy what impedes an effective discourse in research. This is because the keyword does neither specify the integration subject (such as data, process, or application) nor the integration method (such as linking or conversion).
- Third, the term “Integration” refers to the way, but not to the actual goal. For example, the goal of research works related to “BIM-GIS Integration” is often the interoperability of respective software products which can be achieved through integrating their internal models. The association of the keyword with the way leads to the fact that, in related research works, the actual goal falls into the background or is even not addressed.

In the end of the chapter, the concepts Semantic Web and Linked Data, their relevance within BIM and for BIM-GIS Integration is described based on the related literature. Roughly spoken, Semantic Web refers to the vision on the use of linked data in the web based on knowledge representation, while Linked Data literally refers to linked data without any specification on the formality. From the perspective of BIM, the idea of Linked Data and Semantic Web becomes particularly relevant when facing the fourth of the majority levels as defined by Bew and Richards. According to current research, the concepts of Semantic Web and Linked Data are particularly relevant when the conversion of information models is not sufficient, highlighting BIM-GIS Integration as a relevant example.

In the chapter ***Related Literature***, literature about BIM-GIS Integration and contextual linking in the field of Semantic Web was analyzed to identify the research gap.

- To the best of the authors' knowledge, the literature review on BIM-GIS Integration is the most extensive one in this research field and covers 155 publications and 1171 category assignments. The literature review is also referenced in the following chapters to support the statements in this dissertation. In the chapter *Related Literature*, an overview on the publications and authors is provided, the document types are analyzed, and the used alignments in the field of BIM-GIS Integration are discussed. In a nutshell, there is no link predicate that is commonly used for linking information models of the domains BIM and GIS. Moreover, the links are often described only conceptually, or only specific objects are linked such as windows.
- The literature research on contextual linking in the field of Semantic Web refers to linking methods overcoming the limitation of owl:sameAs. In the field of Semantic Web, it is well known that owl:sameAs is generally too ‘strong’ to relate objects of different ontologies, since the link predicate suggests that the related objects are exactly the same object. In summary, there are four major types of contextual linking approach, namely alternative link predicates, transformation rules, restriction to specific situation, and vocabulary restriction. However, all of these approaches have drawbacks which is why there is no approach clearly outstanding from the others.

In the chapter ***Research problem & objectives,*** the research gap was deduced based on the reviewed literature on both BIM-GIS Integration and Contextual linking in the field of Semantic Web. The research gap results from two gaps in the reviewed literature on BIM-GIS Integration.

- First, the proposed linking approaches are limited to certain situations so that they link specific instance models for a specific purpose.
- Second, the proposed linking approaches use a variety of different link predicates but do not discuss their suitability in more detail.

On the other hand, the research works related to contextual linking in the field of Semantic Web has shown that the use of identity links such as owl:sameAs is not suitable for linking heterogeneous ontologies, and that there is no contextual linking approach overcoming this issue without drawbacks. Consequently, the research gap refers to exploring the dependence of the link on the situation in which the link holds with the example of BIM-GIS Integration. Furthermore, the identified research gap was illustrated through task problems from the perspective of the link creation and link query. In the first task problem, different variants of the source models were linked, and different use cases were applied. In the second task problem, the user of the link predicate of the created alignment is too 'weak' so that the link is prone to misinterpretation. Additionally, this chapter describes the objective of this dissertation. The objective of this dissertation is the creation of design principles that support the implementation of information integration environments to overcome problems caused by the dependence of the link on the situation. A design principle is composed of premises describing the situation in which the design principles apply and respective consequences supporting the implementation of these kinds of information systems.

In the next chapter, a new perspective on ***use cases*** demanding the integration of information models related to BIM and GIS is introduced. Here, the use cases can be described with respect to both the *physical subject* and the *activity*.

- The subject can be further described with respect to the *type* such as residential building or transport infrastructure alignment (in case of BIM), or such as cadastral maps or transportation infrastructure at city scale (in case of GIS). Furthermore, the subject can be further described with respect to the *state* such as the subject as-planned in terms of a thought-ahead model or the subject as-built.
- There are two major types of activities, namely *Data management* and *Analysis and simulation*. Data management refers to data consistency measures with related computer-based information systems such as the data transfer to Asset management systems, or to systems adequate for indoor navigation simulation. Simulation and analysis can be further subdivided into the categories *Environment-on-building*, *Building-on-environment*, and *Building-and-environment*. These categories describe the direction of the analysis or simulation. Examples are the analysis of the noise influence of the environment on the building (*Environment-on-building*), the influence of the planned high-rise building on the solar gain of the adjacent buildings (*Building-on-environment)*, or combined indoor and outdoor path

planning (*Building-and-environment*). Exemplary use cases were accumulated based on the reviewed literature and further analyzed with respect to the required level of detail.

Noteworthy, more than 60 percent of the integration efforts in the related literature do not emphasize a specific use case or only refer to buzzwords, and more than 60 percent of the integration efforts of the category *Analysis and Simulation* refer to the subcategory *Environment-on-building.* Moreover, the integration of the hull of the represented subject is often enough for the use cases. In the identified use cases, only the use cases *indoor and outdoor navigation* and *building codes of the category Building-and-environment* may require higher level of detail.

In the chapter ***Instance-level heterogeneity***, a new perspective on differences between instance models is introduced. Instance-level heterogeneity, as defined in this dissertation, occur between different variants or versions of instance models. *Variants* are alternative representations of a real-world object, while *Versions* are different representations of a real-world object at different times. Furthermore, common categorization on heterogeneity were analyzed, with the result that these kinds of categorizations refer to the sources of instance-level heterogeneity but do not address the meaning of instance-level heterogeneity for the integration process. In this dissertation, instance-level heterogeneity is categorized with respect to *conflicting data*, *transformable data*, and *contextual data*. Contextual data means that the instance models are created from different perspective but are not considered as ‘true’ or ‘false’ in an absolute sense. The categories are further illustrated with examples of the categorization approaches, which belong to the sources of instance-level heterogeneity, as described above. The further discourse in this dissertation is limited to instance-level heterogeneity with respect to contextual data, and objects based on this kind of instance-level heterogeneity are called *contextually different*.

In the chapter ***Information integration systems***, a new perspective on information integration systems was introduced which is based on the discourse on computer-based information systems from system-theoretical perspective. In the beginning, the foundations of system theory are described with respect to computer-based information systems. In a nutshell, a computer-based information can be represented through a *functional*, *structural* or *hierarchical concept*. Following the functional concepts, a system is represented through functions transforming the input to an output. Following the structural concept, systems are represented by structural components and their relation to each other. In this dissertation, two models on computer-based information systems are utilized based on both the functional and structural concept aiming to describe information integration systems. The first model addresses the transformation of the source models to an integrated model and forms the integrated model to an application output. The second model represents the integration subjects, namely *application* and *information*, and their relation to each other. The application and information are represented through a set of functions and a set of information required by these functions. Two information systems sharing similar information and functions can be represent by respective overlapping sets of functions and information. In this dissertation, an information integration system is based on the subsystems *integration system*, *application system,* and their relation through the *communication system*.

- The *integration system* transforms the source model(s) to the integrated model which can be achieved through four different integration methods, namely *conversion*, *extension*, *merging*, and *linking*. The integration efforts in the related literature on BIM-GIS Integration were analyzed with respect to both integration method and integrated model. As a result, more than 50% of the integration efforts in the reviewed literature address the integration method conversion. Moreover, there are no integration efforts related to the category *Analysis and simulation* with an integrated model related to BIM, except those related to simulation feedback or transferring terrain models to a BIM-authoring tool.
- The *application system* transforms the integrated information to some application output. An application system refers to different types of functions depending on the required information of two information systems *A* and *B*, namely $Function \in A \setminus B$, $Function \in A \cap B$ and $Function \in (A \cup B)^C$. These former two types of functions refer to *Conversion use cases* since conversion of the information is enough for these types of functions, while the latter type of function refer to *Non-conversion use cases*. The information function required by a function can be expressed as query which was further analyzed with respect to BIM-GIS Integration. In a nutshell, a query belonging to *Non-conversion use cases* covers both a subject related to BIM, and a subject related to GIS. Consequently, integration efforts based on *Non-conversion integration methods* were analyzed in more detail with the result that few of these integration efforts address *Non-conversion use cases*.
- The information exchange between the integration and application system is described by the *communication system* and addressed two communication subjects, namely the integrated model, and the expectations on the integrated model. The integrated model is communicated from the integration system to the application system, while the information requirements are communicated in the opposite direction. The communication system is further described by common procedural and structural models emphasizing the semantic problem that also occurs in the communication between integration and application system.

Additionally, the implementation of information integration systems are discussed with respect to the integration method linking. There are two major types of information integration systems: First, information integration systems in which the link creation and maintenance of the integrated model is performed in different systems. Second, information integration systems in which the link creation and maintenance of the integrated model is performed in the same system. Furthermore, the link creation in the field of BIM-GIS Integration covers the steps *spatial placement*, *matching* and *alignment formulation*. Each of these steps can be performed either manually, automatically, or semi-automatically. Last, the link creation and the query can be performed either *directly sequential*, or *timely separated*.

In chapter ***The rationale behind linking***, the semantics of links with respect to BIM-GIS Integration are discussed. First, the term semantics in scope of computer science is discussed and model-theoretic semantics are described in more detail. Following model-theoretic semantics for RDF(S), the vocabulary specified in an RDF graph are mapped to interpretations which allows the evaluation of the 'truthfulness' of the represented triples. Afterward, a *link* is defined as the explicit relation of two corresponding object, while *corresponding* means that the explicit formulation of an implicit relation between two entities serves a specific purpose, such as the fulfilment of a task specification. In the scope of BIM-GIS Integration, there are three major types of correspondences:

- First, an *equivalence correspondence* which refers to the relation of two objects representing the same real world objects.
- Second, a *hierarchy correspondence* which refers to the relation of two objects representing real-world objects at different hierarchy levels.
- Third, an *adjacency correspondence* which refers to the relation of two objects representing different real world objects, which are to some extent topologically adjacent.

Similarly, there are three link types, namely *equivalence links*, *hierarchy links* and *adjacency links.* Furthermore, topological links are discussed with respect to these link types since topological links are often used to research for linking in the scope of BIM and GIS. In a nutshell, topological relation do not describe whether the related objects represent the same real-world object, but state that the objects are topologically related. Thus, a topological link relating two objects indicates the type of correspondence. Afterward, the semantics of equivalence are further discussed with respect to model-theoretic semantics. In this dissertation, there is the assumption that the information models which are syntactically expressed as RDF graph can be related to model-theoretic semantics, whereby the interpretation of the vocabulary refers to respective applications. Moreover, an equivalence link suggests that something stated about one object also may hold for the related object. Consequently, there are three types of equivalence links, namely *forwarding equivalence links*, *universal equivalence links* and *contextual equivalence links*.

- *Forwarding equivalence links* do not follow truth conditional semantics but solely describe that the related objects represent the 'same real-world object'.
- *Universal equivalence links* suggest that everything stated about one objects holds for the related object independent of the situation.
- *Contextual equivalence links* suggest that something stated about one object holds for the related object depending on the situation.

In more detail, contextual equivalence links are based on a validity scope that defines the vocabulary addressed and the interpretation of the vocabulary. Last, the model-theoretic perspective on the alignment is transferred to the system-theoretic representation of information integration system as defined in the previous chapter.

In the next chapter, the concept of **context-sensitive linking** is introduced and further discussed. In the beginning of the chapter, the concept of idealized information integration systems is introduced. Idealized information integration system are characterized by two constraining situational aspects:

- First, the input variables of the integration system are well-defined.
- Second, the semantics of the link is transparent to the querier.

In the reviewed literature on BIM-GIS Integration, the proposed linking approaches are restricted to idealized information integration systems. For example, the linking approaches are restricted to specific source models, or the link creation and querying are performed by the same person so that the semantics of the link is transparent to the querier. *Non-idealized information integration systems* are information integration systems which are not constrained with respect to these two aspects, which is a requirement for context-sensitive linking. Afterward, the meaning of context in scope of computer science is discussed and context-sensitivity with respect to information integration systems is described. Context-sensitive linking refers to the act of link creation depending on situational aspects which describe the information integration system. In more detail, there are two types of context-sensitive linking, namely *context-dependent linking* and *contextual linking*.

- *Context-dependent linking* occurs when different input variables of the integration system result in different alignments or validity scopes of the alignment.
- *Contextual linking* occurs when the validity scope of the alignment is opaque to the user so that the alignment is prone for misinterpretation.

In *context-dependent linking*, the input variables refer either to *model-oriented aspects,* such as different variants or versions of the vocabulary*,* or to *application-oriented aspects,* such as different interpretations (expectations) from the application. In more detail, *application-oriented aspects* refer either to different interpretations of single vocabularies or different interpretations of whole statements. Both *context-dependent linking* and *contextual linking* are illustrated through respective examples.

In the following chapter, the **design principles** are defined and implemented, and their validity is demonstrated and evaluated. In the beginning of the chapter, the implementation of the demonstration environment is described. The demonstration environment is an information integration system based on the integration system and application system. In the integration system, the link creation is semi-automatically performed using Rhinoceros 3D and the integrated model is maintained in an Apache Jena triplestore. The application system refers to the queries which are applied through the web user interface based on Apache Jena.

- The first design principle relates to *context-dependent linking* and states that context-sensitive information integration systems with variable input of the integration system require an adaptive matching mechanism. Variable input of the integration system refers to *model-*

oriented and *application-oriented aspects*, such as variable vocabulary or variable interpretation. The design principle is implemented through the development of a user interface allowing the adaption of the matching criteria such as the schema-level alignment. The design principle is demonstrated through two demonstration scenarios each composed of two integration scenarios. Two integration scenarios of one demonstration scenario differ regarding either by some *model-oriented* or *application-oriented aspects*. The integration scenarios of a demonstration scenario result in different alignments which corresponds to the concept of context-dependent linking. The instantiation of the different alignments and conducting the respective queries have successfully demonstrated the validity of the design principle of context-dependent linking. As a conclusion, information integration systems based on the integration method linking require an adaptive linkage mechanism when dealing with variable input of the integration system such as different vocabularies and interpretations. More roughly spoken, information integration systems based on the integration method linking require an adaptive linkage mechanism when dealing with variable variants or versions of the source models, or variable functions of an application. In general, a function can also be understood as a specific use case in a wider sense.

- The second design principle relates to *contextual linking* and states that the implicit validity scope of the links must be made explicit in case the implicit validity scope is opaque to the user. Making the validity scope explicit in an absolute manner is not feasible with current approaches because there is neither an approach making the exact meaning of the link explicit nor an approach specifying the exact situation to which the validity scope refers to. Instead, the instantiation is approximated by implementing two approaches that aim to reduce the risk of misinterpreting a link due to its opaque validity scope, namely *Use case-oriented approach* and *Similarity-oriented approach*. Following the *Use case-oriented approach*, the link is enriched by data defining the use cases for which the link holds. Following the *Similarity-oriented approach*, the link is enriched by a value that defines the similarity of the linked objects. The design principle is successfully demonstrated through conducting two different demonstration scenarios. Both demonstration scenarios illustrate the design principle based on three integration scenarios, namely without additional information, following the *Use case-oriented approach*, and following the *Similarity-oriented approach*. As a conclusion, information integration systems in which the implicit validity scope of the alignment is opaque to the user, the implicit validity scope must be made explicit to reduce the risk of misinterpretation. More roughly spoken, the design principle of *contextual linking* means that in case the implicit semantics of the link is opaque to the user, the implicit semantics of the link must be made explicit.

In addition, both context-dependent linking and contextual linking are based on the premise that the objects of the source models must be contextually different, as described previously. In Table 11.1, the premises and consequences are summarized for both the design principle of *context-dependent linking* and the design principle of *contextual linking* using ‘less complex’ terms than those based on model-theoretic semantics. Through this kind of description, the design principles are formulated less exact but are rather suitable for the adoption in respective projects.

Table 11.1: Summary of the premises and consequences for the design principles on context-dependent and contextual linking.

	Context-dependent linking	**Contextual linking**
Premise	Source models with contextually different objects	
	Variable source models or variable functions (use cases)	Implicit semantics of the link is opaque to the user
Consequence	Adaptive matching mechanism	Making the semantics of the link explicit

11.2 Discussion

In the following, the entry questions of this dissertation are discussed (see chapter 1.1):

> *“Does* the mismatch of corresponding objects cause problems for linking heterogeneous information models from the domains BIM and UIM? […]

The answer to this question requires the discourse about the meaning of the mismatch of corresponding objects and the semantics of links.

- In more detail, the mismatch of corresponding objects refers to instance-level heterogeneity and occurs between different variants or versions of instance models. The instance-level heterogeneity can be further categorized in *contradicting*, *transformable* and *contextual data*. Instance-level heterogeneity in terms of contextual data means that the objects represent the same real-world object but are created from different perspectives.
- There are three types of links in the field of BIM-GIS Integration, namely *equivalence link*, *hierarchy link* and *adjacency link*. In this dissertation, the discourse on information integration is limited to equivalence links. An equivalence link, as defined in this dissertation, relates two objects representing the ‘same real-world object’ so that something stated about one object may hold for the related object.

An equivalence link relating two contextually different objects is prone to be misleading since not everything stated about one object also holds for the corresponding object. In general, there are three different types of equivalence links dealing with this issue.

- First, *forwarding equivalence links* state that the linked objects refer to the same real-world object. Forwarding equivalence links do not describe the contextual difference of these objects, and are, therefore, prone to be misleading when linking contextually different objects.
- Second, *universal equivalence links* suggest that everything stated about one object holds for the corresponding object independent of the situation. Universal equivalence links are

misleading for linking contextually different objects, since not everything stated about one object does also hold for the corresponding object that is contextually different.

- Third, contextual equivalence links restrict the validity scope of a link with respect to a specific situation. Here, the validity scope is defined through vocabulary and interpretation. However, making the validity scope of a link explicit cannot be achieved seamlessly since there is no approach to make the exact meaning of the link explicit, nor is there an approach to specify exactly the situation to which the validity scope applies.

Thus, the mismatch of corresponding objects is prone to cause problems for linking heterogeneous information models from the domains BIM and UIM since the semantics of equivalence links are generally not sufficient for relating contextually different objects.

> […] If yes, *when* does it cause these problems, […]

The mismatch of corresponding objects causes problems for the linking process when linking contextually different data in non-idealized but *context-sensitive information integration systems*.

- information integration systems are *non-idealized* if they are characterized by at least one of the following situational aspects: First, the integration systems has variable input, such as variable source models or variable functions. Second, the implicit validity scope of the alignment is opaque to the user.
- Information integration systems are *context-sensitive* when these kinds of different situational aspects require different alignments of the integrated model. Thus, information integration systems being non-idealized is the premise for being context-sensitive.

Here, there are two different types of context-sensitive information integration systems according to the two aspects characterizing non-idealized information integration systems, namely those dealing with *context-dependent* and those dealing with *contextual linking*.

- *Context-dependent linking* occurs when different input variables, such as different source models or different functions, of the integration system result in different alignments such as different correspondences or validity scopes.
- *Contextual linking* occurs when the implicit validity scope of the alignment is opaque to the user so that the validity must be made explicit to reduce the risk of misinterpretation.

Thus, the mismatch of corresponding objects causes problems for the linking process when the linked data is contextually different and the information integration systems are context-sensitive, so that context-dependent linking or contextual linking becomes relevant for the alignment creation.

> […] *which* problems does it cause […]

The problems caused by non-idealized, context-sensitive information integration systems for linking heterogeneous information models with contextually different data refer to the concepts of *context-dependent linking* and *contextual linking*.

- The problem related to *context-dependent linking* can be illustrated from perspective of the link creator on the information integration systems by the following question: How to link the instance models *X1* and *X2* when dealing with function *Y*?
- The problem related to *contextual linking* can be illustrated from the perspective of the link querier on the information integration systems by the following question: Is the link valid for my function?

Thus, the problems in the linking process caused by contextually different objects refer to both the link creation depending on the situation, and the link interpretation depending on the situation.

> […] and *how* can these problems be overcome?"

The solution approaches to the problems above refer to design principles supporting the implementation of information integration environments dealing with *context-dependent linking* and *contextual linking*.

- Context-sensitive information integration environments with variable input for the integration system require an adaptive matching mechanism (*context-dependent linking*). This is because the matching criteria depends on the input of the integration system such as the source models and the function. For example, in integration systems based on semi-automated linking, the link creator should be able to adapt the matching criteria such as the schema-level alignment.
- In context-sensitive information integration environments, in which the implicit validity scope of the links is opaque to the user, the implicit validity scope of the links must be made explicit (*contextual linking*). There are two major approaches aiming to make the validity scope explicit, namely *Use case-oriented approach* and *Similarity-oriented approach*. In the use case-oriented approach, the use case in which the alignment is valid is made explicit. For example, use cases can be standardized and respective description can be attached to the alignment. In the similarity-oriented approach, the similarity of the linked objects is made explicit. For example, similarity values or codes based on standardized similarity measures can be attached to the links.

In summary, the described problems related to context-dependent and contextual can be addressed by the implementation of information integration systems considering the respective design principles as defined above.

11.3 Future research

This dissertation provides entry points for several research topics related to BIM-GIS Integration and Semantic Web. Here, two major topics relevant for future research works related to the topic on context-sensitive information integration systems are described.

- First, this dissertation is limited to context-sensitive information integration systems with the example of **equivalence links**. Future research needs to discuss context-sensitive information integration systems using the example of other link types such as hierarchy and adjacency links. Here, the semantics of both hierarchy and adjacency needs to be defined to understand their relevance in context-sensitive information integration system. Among others, the discourse about the semantics of hierarchy links refer to the question how hierarchy and equivalence correspondences can be distinguished.
- Second, the **implementation of the design principles** of both context-dependent linking and contextual linking need to be investigated in more detail. As an example, respective use cases need to be standardized for establishing the Use case-oriented approach. Furthermore, similarity measurements need to be further investigated for establishing the Similarity-oriented approach.

In summary, this dissertation on *context-sensitive linking of heterogeneous information models from the building and the urban domain* contributes to the research knowledge of both BIM-GIS Integration and Semantic Web and provides entry points for future research work.

References

[1] A. Smith, The Wealth of Nations. An inquiry into the nature and causes of the Wealth of Nations, Harriman House Limited, 2010.

[2] F. Beck, A. Borrmann, T.H. Kolbe, The need for a differentiation between heterogeneous information integration approaches in the field of "BIM-GIS Integration": a literature review, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VI-4/W1-2020 (2020) 21–28. https://doi.org/10.5194/isprs-annals-VI-4-W1-2020-21-2020.

[3] S.F. Beck, J. Abualdenien, I.H. Hijazi, A. Borrmann, T.H. Kolbe, Analyzing Contextual Linking of Heterogeneous Information Models from the Domains BIM and UIM, IJGI 10 (2021). https://doi.org/10.3390/ijgi10120807.

[4] T.H. Kolbe, A. Donaubauer, Semantic 3D City Modeling and BIM, in: W. Shi, M.F. Goodchild, M. Batty, M.-P. Kwan, A. Zhang (Eds.), Urban Informatics, Springer Singapore, Singapore, 2021, pp. 609–636. https://doi.org/10.1007/978-981-15-8983-6_34.

[5] X. Liu, X. Wang, G. Wright, J.C. Cheng, X. Li, R. Liu, A State-of-the-Art Review on the Integration of Building Information Modeling (BIM) and Geographic Information System (GIS), IJGI 6 (2017) 53. https://doi.org/10.3390/ijgi6020053.

[6] F. Noardo, F. Biljecki, G. Agugiaro, K.A. Ohori, C. Ellul, L. Harrie, J.E. Stoter, GeoBIM benchmark 2019: intermediate results, IJGI XLII-4/W15 (2019) 47–52. https://doi.org/10.5194/isprs-archives-XLII-4-W15-47-2019.

[7] Y. Bar-Yam, Complexity Rising: From Human Beings to Human Civilization, a Complexity Profile, UNESCO Encyclopedia of Life Support Systems (2002).

[8] Y.T. Lee, Information modeling: From design to implementation, in: Proceedings of the second world manufacturing congress, 1999, pp. 315–321 (accessed 21 December 2020). https://www.nist.gov/publications/information-modeling-design-implementation.

[9] G. Booch, R.A. Maksimchuk, M.W. Engle, B.J. Young, J. Connallen, K.A. Houston, Object-oriented analysis and design with applications, ACM SIGSOFT software engineering notes 33 (2008) 29.

[10] M. Uschold, M. Gruninger, Ontologies and semantics for seamless connectivity, SIGMOD Rec. 33 (2004) 58. https://doi.org/10.1145/1041410.1041420.

[11] S. Herle, R. Becker, R. Wollenberg, J. Blankenbach, GIM and BIM, Journal of Photogrammetry, Remote Sensing and Geoinformation Science (2020) 33–42. https://doi.org/10.1007/s41064-020-00090-4.

[12] T.H. Kolbe, L. Plümer, Bridging the gap between GIS and CAAD: Geometry, referencing, representations, standards and semantic modelling, GIM International (2004) 12–15 (accessed 20 December 2020). https://www.researchgate.net/publication/293296915_Bridging_the_gap_between_GIS_and_CAAD_Geometry_referencing_representations_standards_and_semantic_modelling.

[13] C. Nagel, A. Stadler, T.H. Kolbe, Conceptual Requirements for the Automatic Reconstruction of Building Information Models from Uninterpreted 3D Models, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. 38 (2009) 46–53 (accessed 21 December 2020). https://www.isprs.org/proceedings/XXXVIII/3_4-C3/.

[14] T. Brüggemann, P. von Both, 3D-Stadtmodellierung: CityGML, in: A. Borrmann, M. König, C. Koch, J. Beetz (Eds.), Building Information Modeling: Technologische Grundlagen und industrielle Praxis, Springer Vieweg, Wiesbaden, 2015, pp. 177–192. https://doi.org/10.1007/978-3-658-05606-3_10.

[15] H.B. Newcombe, J.M. Kennedy, S.J. Axford, A.P. James, Automatic linkage of vital records, Science 130 (1959) 954–959. https://doi.org/10.1126/science.130.3381.954.

[16] R. Peachavanish, H.A. Karimi, B. Akinci, F. Boukamp, An ontological engineering approach for integrating CAD and GIS in support of infrastructure management, Advanced Engineering Informatics 20 (2006) 71–88. https://doi.org/10.1016/j.aei.2005.06.001.

[17] B. Akinci, H.A. Karimi, A. Pradhan, C.-C. Wu, G. Fichtl, CAD and GIS Interoperability through Semantic Web Services, in: H.A. Karimi, B. Akinci (Eds.), CAD and GIS Integration, Auerbach Publications, Boca Raton, FL, USA, 2009, pp. 199–222. https://doi.org/10.1201/9781420068061-c9.

[18] K. McGlinn, A. Wagner, P. Pauwels, P. Bonsma, P. Kelly, D. O’Sullivan, Interlinking geospatial and building geometry with existing and developing standards on the web, Autom. Constr 103 (2019) 235–250. https://doi.org/10.1016/j.autcon.2018.12.026.

[19] E.P. Karan, J. Irizarry, J. Haymaker, BIM and GIS Integration and Interoperability Based on Semantic Web Technology, J. Comput. Civ. Eng. 30 (2016) 4015043. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000519.

[20] A.-H. Hor, G. Sohn, P. Claudio, M. Jadidi, A. Afnan, A Semantic Graph Database for BIM-GIS Integrated Information Model for an Intelligent Urban Mobility Web Application, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. IV-4 (2018). https://doi.org/10.5194/isprs-annals-IV-4-89-2018.

[21] K. McGlinn, R. Brennan, C. Debruyne, A. Meehan, L. McNerney, E. Clinton, P. Kelly, D. O'Sullivan, Publishing authoritative geospatial data to support interlinking of building information models, Autom. Constr 124 (2021) 103534. https://doi.org/10.1016/j.autcon.2020.103534.

[22] M. Stepien, A. Vonthron, M. König, An Approach for Cross-Data Querying and Spatial Reasoning of Tunnel Alignments, in: Proceedings of the EG-ICE 2021, Berlin, Germany, 2021.

[23] S. Vilgertshofer, J. Amann, B. Willenborg, A. Borrmann, T.H. Kolbe, Linking BIM and GIS Models in Infrastructure by Example of IFC and CityGML, in: International Workshop on Computing in Civil Engineering, Seattle, ASCE, Reston, VA, USA, 2017, pp. 133–140. https://doi.org/10.1061/9780784480823.017.

[24] Y. Deng, J.C. Cheng, C. Anumba, Mapping between BIM and 3D GIS in different levels of detail using schema mediation and instance comparison, Autom. Constr 67 (2016) 1–21. https://doi.org/10.1016/j.autcon.2016.03.006.

[25] J. Sun, P. Olsson, H. Eriksson, L. Harrie, Evaluating the geometric aspects of integrating BIM data into city models, Journal of Spatial Science 65 (2020) 235–255. https://doi.org/10.1080/14498596.2019.1636722.

[26] A. Hevner, S. March, J. Park, S. Ram, Design Science in Information Systems Research, Management Information Systems Quarterly 28 (2004) 75-105 (accessed 21 December 2020). https://www.researchgate.net/publication/201168946_Design_Science_in_Information_Systems_Research.

[27] K. Peffers, T. Tuunanen, M. Rothenberger, S. Chatterjee, A design science research methodology for information systems research, Journal of Management Information Systems 24 (2007) 45–77.

[28] V. Vaishnavi, B. Kuechler, Design Science Research in Information Systems, Association for Information Systems (2004).

[29] J. Conklin, M. Basadur, G.K. VanPatter, Rethinking wicked problems–Unpacking paradigms, bridging universes, NexD Journal 10 (2007).

[30] H.W. Rittel, M.M. Webber, Wicked problems, Man-made Futures 26 (1974) 272–280.

[31] S.T. March, G.F. Smith, Design and natural science research on information technology, Decision Support Systems 15 (1995) 251–266. https://doi.org/10.1016/0167-9236(94)00041-2.

[32] R.L. Ackoff, From Data to Wisdom, Journal of Applied Systems Analysis (1989) 3–9.

[33] H. Stuckenschmidt, Ontologien: Konzepte, Technologien und Anwendungen, second. Aufl., Springer, Heidelberg, 2011.

[34] F.J. Furrer, Eine kurze Geschichte der Ontologie, Informatik Spektrum 37 (2014) 308–317. https://doi.org/10.1007/s00287-012-0642-3.

[35] J. Euzenat, P. Shvaiko, Ontology Matching, Springer Berlin Heidelberg, 2013. https://doi.org/10.1007/978-3-642-38721-0_2.

[36] T.R. Gruber, A translation approach to portable ontology specifications, Knowledge Acquisition 5 (1993) 199–220. https://doi.org/10.1006/knac.1993.1008.

[37] H. Stachowiak, Allgemeine Modelltheorie, Springer, 1973 (accessed 20 December 2020). https://archive.org/details/Stachowiak1973AllgemeineModelltheorie.

[38] M. Gagnon, Ontology-based integration of data sources, in: 10th International Conference on Information Fusion, Quebec City, IEEE, 2007, pp. 1–8. https://doi.org/10.1109/ICIF.2007.4408086.

[39] J. E. Rumbaugh, I. Jacobson, G. Booch, The Unified Modelling Language Reference Manual, Addison-Wesley (1999).

[40] J. Davies, Lightweight Ontologies, in: R. Poli, M. Healy, A. Kameas (Eds.), Theory and Applications of Ontology: Computer Applications, Springer Netherlands, Dordrecht, 2010, pp. 197–229. https://doi.org/10.1007/978-90-481-8847-5_9.

[41] F. Giunchiglia, I. Zaihrayeu, Lightweight Ontologies, in: L. Liu, M.T. ÖZSU (Eds.), Encyclopedia of Database Systems, Springer US, Boston, MA, 2009, pp. 1613–1619. https://doi.org/10.1007/978-0-387-39940-9_1314.

[42] M.H. Rasmussen, M. Lefrançois, G. Schneider, P. Pauwels, BOT: the Building Topology Ontology of the W3C Linked Building Data Group, Semantic Web (2020) 1–19. https://doi.org/10.3233/SW-200385.

[43] Object Management Group, OMG Meta Object Facility (MOF) Core Specification, 2019. https://www.omg.org/spec/MOF (accessed 14 July 2021).

[44] Object Management Group (OMG). https://www.omg.org/ (accessed 7 October 2022).

[45] International Organization for Standardization, ISO 10303-11:2004: Industrial automation systems and integration: Product data representation and exchange - Part 11: Description methods: The EXPRESS language reference manual, Genf, Schweiz 25.040.40, 2004. https://www.iso.org/standard/38047.html (accessed 9 October 2022).

[46] W3C RDF Working Group, RDF 1.1 Primer, 2020. https://www.w3.org/TR/rdf11-primer/ (accessed 14 July 2021).

[47] W3C OWL Working Group, OWL 2 Web Ontology Language: Document Overview (Second Edition). https://www.w3.org/TR/owl2-overview/ (accessed 20 November 2020).

[48] J. Zedlitz, Konzeptuelle Modellierung mit UML und OWL–Untersuchung der Gemeinsamkeiten und Unterschiede mit Hilfe von Modelltransformationen, Christian-Albrechts-Universität zu Kiel, 2013.

[49] P.D. Karp, V.K. Chaudhri, J. Thomere, XOL: An XML-Based Ontology Exchange Language, Stanford, 1999.

[50] S. Cox, P. Daisey, R. Lake, C. Portele, A. Whiteside, Geography Markup Language (GML) Encoding Specification v3.1.1, Open Geospatial Consortium, 2004.

[51] International Organization for Standardization, ISO 10303-21:2002 Industrial automation systems and integration – Product data representation and exchange – Part 21: Implementation methods: Clear text encoding of the exchange structure, Genf, Schweiz.

[52] RDF 1.1 Turtle. https://www.w3.org/TR/turtle/ (accessed 10 March 2021).

[53] P. Pauwels, T. Krijnen, W. Terkaj, J. Beetz, Enhancing the ifcOWL ontology with an alternative representation for geometric data, Autom. Constr 80 (2017) 77–94. https://doi.org/10.1016/j.autcon.2017.03.001.

[54] M. Hannus, Island of Automation in Construction, 1998. http://cic.vtt.fi/hannus/islands/ (accessed 30 May 2018).

[55] IEEE 610 working group, IEEE Standard Glossary of Software Engineering Terminology, IEEE, Piscataway, NJ, USA. http://ieeexplore.ieee.org/xpls/absall.jsp?arnumber=159342 (accessed 15 September 2021).

[56] International Organization for Standardization, ISO 16739-1:2018: Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries: Part 1: Data schema 25.040.40, 2018. https://www.iso.org/standard/70303.html (accessed 9 October 2022).

[57] Open Geospatial Consortium, CityGML. https://www.ogc.org/standards/citygml (accessed 20 November 2020).

[58] G. Gröger, L. Plümer, CityGML – Interoperable semantic 3D city models, ISPRS Journal of Photogrammetry and Remote Sensing 71 (2012) 12–33. https://doi.org/10.1016/j.isprsjprs.2012.04.004.

[59] J. Beetz, J.P. Van Leeuwen, B. de Vries, IfcOWL: A Case of Transforming Express Schemas into Ontologies, Artif. Intell. Eng. Des. Anal. Manuf. 23 (2009) 89–101. https://doi.org/10.1017/S0890060409000122.

[60] P. Pauwels, W. Terkaj, EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology, Autom. Constr 63 (2016) 100–133. https://doi.org/10.1016/j.autcon.2015.12.003.

[61] International Organization for Standardization, ISO/IEC 19505-1:2012: Information technology — Object Management Group Unified Modeling Language (OMG UML), 2nd ed. 35.060, 2012. https://www.iso.org/standard/32624.html (accessed 9 October 2022).

[62] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, F. Yergeau, Extensible Markup Language (XML) 1.0 (Fifth Edition). https://www.w3.org/TR/xml/ (accessed 7 October 2022).

[63] International Organization for Standardization, ISO 10303-21:2016: Industrial automation systems and integration — Product data representation and exchange — Part 21: Implementation

methods: Clear text encoding of the exchange structure 25.040.40, 2016. https://www.iso.org/standard/63141.html (accessed 9 October 2022).

[64] R. Cygniak, D. Wood, M. Lanthaler, RDF 1.1 Concepts and Abstract Syntax, 2014. http://www.w3.org/TR/rdf11-concepts/ (accessed 16 May 2018).

[65] E. Prud'hommeaux, A. Seaborn, SPARQL Query Language for RDF, 2008. https://www.w3.org/TR/rdf-sparql-query/ (accessed 22 June 2018).

[66] D. Brickley, R.V. Guha, RDF Schema 1.1, 2014. http://www.w3.org/TR/rdf-schema/ (accessed 16 May 2018).

[67] A. Miles, S. Bechhofer, SKOS Simple Knowledge Organization System Reference. https://www.w3.org/TR/skos-reference/ (accessed 22 April 2021).

[68] H. Halpin, P.J. Hayes, J.P. McCusker, D.L. McGuinness, H.S. Thompson, When owl: sameas isn’t the same: An analysis of identity in linked data, in: ISWC 2010, Proceedings of the 9th International Semantic Web Conference, Shanghai, China, Springer, Berlin/Heidelberg, Germany, 2010, pp. 305–320. https://doi.org/10.1007/978-3-642-17746-0_20.

[69] buildingsmart, Industry Foundation Classes 4.0.2.1, 2022. https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/.

[70] M.H. Rasmussen, P. Pauwels, M. Lefrançois, G.F. Schneider, Building Topology Ontology. https://w3c-lbd-cg.github.io/bot/ (accessed 2 October 2022).

[71] Building Element Ontology. https://pi.pauwel.be/voc/buildingelement/index-en.html (accessed 2 October 2022).

[72] A. Borrmann, M. König, C. Koch, J. Beetz, Die BIM-Methode im Überblick, in: A. Borrmann, M. König, C. Koch, J. Beetz (Eds.), Building Information Modeling, Springer Fachmedien Wiesbaden, Wiesbaden, 2021, pp. 1–31. https://doi.org/10.1007/978-3-658-33361-4_1.

[73] C. Eastman, K. Liston, R. Sacks, P. Teicholz, BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors, Wiley, 2008.

[74] McGraw Hill, The Business Value of BIM: Getting Building Information Modeling to the Bottom Line, McGraw Hill, 2009.

[75] P. MacLeamy, Collaboration, Integrated Information, and the Project Lifecycle in Building Design and Construction and Operation, Construction User Roundtable WP-1202, 2004.

[76] F. Biljecki, J.E. Stoter, H. Ledoux, S. Zlatanova, A. Çöltekin, Applications of 3D City Models: State of the Art Review, IJGI 4 (2015) 2842–2889. https://doi.org/10.3390/ijgi4042842.

[77] C.M.O. Yosino, S.L. Ferreira, Using BIM and GIS Interoperability to Create CIM Model for USW Collection Analysis, in: Proceedings of the 18th International Conference on Computing in Civil and Building Engineering, Springer International Publishing, Cham, 2021, pp. 248–271. https://doi.org/10.1007/978-3-030-51295-8_19.

[78] N. Bartelme, GIS-Technologie: Geoinformationssysteme, Landinformationssysteme und ihre Grundlagen, Springer, Berlin, Heidelberg, 1989.

[79] F. Noardo, L. Harrie, K. Arroyo Ohori, F. Biljecki, C. Ellul, T. Krijnen, H. Eriksson, D. Guler, D. Hintz, M.A. Jadidi, M. Pla, S. Sanchez, V.-P. Soini, R. Stouffs, J. Tekavec, J.E. Stoter, Tools for BIM-GIS Integration (IFC Georeferencing and Conversions): Results from the GeoBIM Benchmark 2019, IJGI 9 (2020) 502. https://doi.org/10.3390/ijgi9090502.

[80] T.W. Kang, C.H. Hong, A study on software architecture for effective BIM/GIS-based facility management data integration, Autom. Constr 54 (2015) 25–38. https://doi.org/10.1016/j.autcon.2015.03.019.

[81] A.H. Liu, C. Ellul, M. Swiderska, Decision Making in the 4th Dimension—Exploring Use Cases and Technical Options for the Integration of 4D BIM and GIS during Construction, IJGI 10 (2021) 203. https://doi.org/10.3390/ijgi10040203.

[82] E. Hbeich, A. Roxin, N. Bus, Previous BIM-GIS Integration Approaches: Analytic Review and Discussion, Le BIM et l'évolution des pratiques: Ingénierie et architecture, enseignement et recherche (2020) 47.

[83] M.J. Sani, A. Abdul Rahman, Gis and BIM Integration at Data Level: A Rieview, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. (2018) 299–306. https://doi.org/10.5194/isprs-archives-XLII-4-W9-299-2018.

[84] C. Ellul, F. Noardo, L. Harrie, J.E. Stoter, The EuroSDR GeomBIM Project - Developing Case Studies for the Use of GeoBIM in Practice, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLIV-4/W1-2020 (2020) 33–40. https://doi.org/10.5194/isprs-archives-XLIV-4-W1-2020-33-2020.

[85] I.H. Hijazi, A. Donaubauer, Integration of Building and Urban Information Modeling – Opportunities and Integration Approaches, in: T.H. Kolbe (Ed.), Geoinformationssysteme 2017: Beiträge zur 4. Münchner GI-Runde, Wichmann, Berlin, Offenbach, 2017, pp. 42–53.

[86] A. Donaubauer, R. Kaden, R. Seuß, U. Gruber, T.H. Kolbe, CAD, BIM und GIS – digitale Modelle der gebauten Umwelt, in: Leitfaden Geodäsie und BIM, Version third.0, Bühl/München, 2021.

[87] C. Clemen, Trends in BIM and GIS Standardization - Report from the joint ISO/TC59/SC13–ISO/TC211 WG: GIS-BIM, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLVI-5/W1-2022 (2022) 51–58. https://doi.org/10.5194/isprs-archives-XLVI-5-W1-2022-51-2022.

[88] M.J. Casey, S. Vankadara, Semantics in CAD/GIS integration, in: H.A. Karimi, B. Akinci (Eds.), CAD and GIS Integration, Auerbach Publications, Boca Raton, FL, USA, 2009, pp. 155–182.

[89] M. El-Mekawy, A. Östman, I.H. Hijazi, An evaluation of IFC-CityGML unidirectional conversion, IJACSA 3 (2013) 159–171. https://doi.org/10.14569/IJACSA.2012.030525.

[90] F. Biljecki, H. Tauscher, Quality of BIM-GIS conversion, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. IV-4/W8 (2019) 35–42. https://doi.org/10.5194/isprs-annals-IV-4-W8-35-2019.

[91] S. Donkers, H. Ledoux, J. Zhao, J.E. Stoter, Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings, Transactions in GIS 20 (2016) 547–569. https://doi.org/10.1111/tgis.12162.

[92] J. Lim, H. Tauscher, F. Biljecki, Graph transformation rules for IFC-to-CityGML attribute conversion, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. IV-4/W8 (2019) 83–90. https://doi.org/10.5194/isprs-annals-IV-4-W8-83-2019.

[93] E. Hbeich, A. Roxin, Linking BIM and GIS Standard Ontologies with Linked Data, in: Proceedings of the 8th Linked Data in Architecture and Construction Workshop, Dublin, Ireland, CEUR-WS, Aachen, Germany, 2020 (accessed 20 December 2020). http://ceur-ws.org/Vol-2636/.

[94] K. Adouane, R. Stouffs, P. Janssen, B. Domer, A model-based approach to convert a building BIM-IFC data set model into CityGML, Journal of Spatial Science (2019) 1–24. https://doi.org/10.1080/14498596.2019.1658650.

[95] H. Tauscher, Creating and Maintaining IFC-CityGML Conversion Rules, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. IV-4/W8 (2019) 115–122. https://doi.org/10.5194/isprs-annals-IV-4-W8-115-2019.

[96] H. Tauscher, R. Stouffs, An IFC-to-CityGML Triple Graph Grammar, in: ECAADe 2018 proceedings of the 36th International Conference on Education and Research in Computer Aided Architectural Design in Europe, Łódź, Poland, firstst edition, eCAADe, Brussels, Łódź, 2018, pp. 517–524 (accessed 20 December 2020). http://papers.cumincad.org/cgi-bin/works/paper/ecaade2018_265.

[97] T. Berners-Lee, Weaving the web: The original design and ultimate destinity of the world wide web by its inventor, first. ed., HarperCollins, San Francisco, 1999.

[98] C. Bizer, T. Heath, T. Berners-Lee, Linked Data - The Story So Far, International Journal on Semantic Web and Information Systems 5 (2009) 1–22. https://doi.org/10.4018/jswis.2009081901.

[99] W3C, Linked Data, 2022. https://www.w3.org/standards/semanticweb/data.

[100] T. Berners-Lee, Linked Data, 2009. https://www.w3.org/DesignIssues/LinkedData.html.

[101] J.P. McCrae, Linked Open Data Cloud. http://lod-cloud.net/ (accessed 2 October 2020).

[102] M. Bew, M. Richards, Bew Richards BIM Maturity Model, in: BuildingSMART Construct IT Autumn Members Meeting, Brighton, UK, 2008.

[103] P. Pauwels, A. Roxin, SimpleBIM: From full ifcOWL graphs to simplified building graphs, in: ECPPM, Limasol, Cyprus, 2016.

[104] P. Pauwels, S. Zhang, Y.-C. Lee, Semantic web technologies in AEC industry: A literature overview, Autom. Constr 73 (2017) 145–165. https://doi.org/10.1016/j.autcon.2016.10.003.

[105] Elsevier, Scopus - Abstract and citation database, 2022. https://www.scopus.com/.

[106] F. Biljecki, J. Lim, J. Crawford, D. Moraru, H. Tauscher, A. Konde, K. Adouane, S. Lawrence, P. Janssen, R. Stouffs, Extending CityGML for IFC-sourced 3D city models, Autom. Constr 121 (2021) 103440. https://doi.org/10.1016/j.autcon.2020.103440.

[107] Literature analysis - BIM GIS Integation. https://1drv.ms/u/s!AoSgdjb6RJMZhekoyKVn39ttDMSfIQ?e=qaDp6Y (accessed 24 November 2022).

[108] P. van Oosterom, J.E. Stoter, E. Jansen, Bridging the Worlds of CAD and GIS, in: S. Zlatanova, D. Prosperi (Eds.), Large-scale 3D Data Integration, CRC Press, Boca Raton, FL, USA, 2005, pp. 9–36. https://doi.org/10.1201/9781420036282.pt1.

[109] K. Arroyo Ohori, F. Biljecki, K. Kumar, H. Ledoux, J.E. Stoter, Modeling Cities and Landscapes in 3D with CityGML, in: A. Borrmann, M. König, C. Koch, J. Beetz (Eds.), Building Information Modeling: Technology Foundations and Industry Practice, Springer International Publishing, Cham, 2018, pp. 199–215. https://doi.org/10.1007/978-3-319-92862-3_11.

[110] C. Ellul, J.E. Stoter, L. Harrie, M. Shariat, A. Behan, M. Pla, Invetigating the State of Play of GeoBIM across Europe, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-4/W10 (2018) 19–26. https://doi.org/10.5194/isprs-archives-XLII-4-W10-19-2018.

[111] F. Noardo, C. Ellul, L. Harrie, E. Devys, K. Arroyo Ohori, P. Olsson, J.E. Stoter, EuroSDR GeoBIM Project: A Study in Eirope on how to use the potentials of Geo data in Practice, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-4/W15 (2019) 53–60. https://doi.org/10.5194/isprs-archives-XLII-4-W15-53-2019.

[112] F. Noardo, C. Ellul, L. Harrie, I. Overland, M. Shariat, K. Arroyo Ohori, J.E. Stoter, Opportunities and challenges for GeoBIM in Europe: developing a building permits use-case to raise awareness and examine technical interoperability challenges, Journal of Spatial Science 42 (2019).

[113] F. Noardo, T. Krijnen, K. Arroyo Ohori, F. Biljecki, C. Ellul, L. Harrie, H. Eriksson, L. Polia, N. Salheb, H. Tauscher, J. Liempt, H. Goerne, D. Hintz, T. Kaiser, C. Leoni, A. Warchol, J. Stoter, Reference study of IFC software support: The GeoBIM benchmark 2019—Part I, Transactions in GIS 25 (2021) 805–841. https://doi.org/10.1111/tgis.12709.

[114] R. Fosu, K. Suprabhas, Z. Rathore, C. Cory, Integration of Building Information Modeling (BIM) and Geographic Information Systems (GIS), in: Proceedings of the 32nd CIB W78 Conference, Eindhoven, 2015 (accessed 20 December 2020). https://itc.scix.net/paper/w78-2015-paper-020.

[115] S. Kurwi, P. Demian, T.M. Hassan, Integrating BIM and GIS in railway projects: A critical review, in: 33rd Annual ARCOM Conference, Cambridge, UK, 2017, pp. 4–6 (accessed 20 December 2020). https://core.ac.uk/display/288367404?recSetID=.

[116] Y. Song, X. Wang, Y. Tan, P. Wu, M. Sutrisna, J.C. Cheng, K. Hampson, Trends and Opportunities of BIM-GIS Integration in the Architecture, Engineering and Construction Industry: A Review from a Spatio-Temporal Statistical Perspective, IJGI 6 (2017) 397. https://doi.org/10.3390/ijgi6120397.

[117] J. Zhu, G. Wright, J. Wang, X. Wang, A Critical Review of the Integration of Geographic Information System and Building Information Modelling at the Data Level, IJGI 7 (2018) 66. https://doi.org/10.3390/ijgi7020066.

[118] Z. Ma, Y. Ren, Integrated Application of BIM and GIS: An Overview, Procedia Engineering 196 (2017) 1072–1079. https://doi.org/10.1016/j.proeng.2017.08.064.

[119] W.N.F. Wan Abdul Basir, Z. Majid, U. Ujang, A. Chong, INTEGRATION OF GIS AND BIM TECHNIQUES IN CONSTRUCTION PROJECT MANAGEMENT - A REVIEW, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-4/W9 (2018) 307–316. https://doi.org/10.5194/isprs-archives-XLII-4-W9-307-2018.

[120] H. Wang, Y. Pan, X. Luo, Integration of BIM and GIS in sustainable built environment: A review and bibliometric analysis, Autom. Constr 103 (2019) 41–52. https://doi.org/10.1016/j.autcon.2019.03.005.

[121] D. Shkundalov, T. Vilutienė, Bibliometric analysis of building information modeling, geographic information systems and web environment integration, Autom. Constr 128 (2021) 103757. https://doi.org/10.1016/j.autcon.2021.103757.

[122] J. Wei, G. Chen, J. Huang, L. Xu, Y. Yang, J. Wang, A.-M. Sadick, BIM and GIS Applications in Bridge Projects: A Critical Review, Applied Sciences 11 (2021) 6207. https://doi.org/10.3390/app11136207.

[123] L. Souza, C. Bueno, City Information Modelling as a support decision tool for planning and management of cities: A systematic literature review and bibliometric analysis, Building and Environment 207 (2022) 108403. https://doi.org/10.1016/j.buildenv.2021.108403.

[124] M. El-Mekawy, A. Östman, Semantic Mapping: an Ontology Engineering Method for Integrating Building Models in IFC and CITYGML (2010).

[125] M. El-Mekawy, A. Östman, I.H. Hijazi, A Unified Building Model for 3D Urban GIS, IJGI 1 (2012) 120–145. https://doi.org/10.3390/ijgi1020120.

[126] M. El-Mekawy, A. Östman, K. Shahzad, Towards Interoperating CityGML and IFC Building Models: A Unified Model Based Approach, in: T.H. Kolbe, G. König, C. Nagel (Eds.), Advances in 3D Geo-Information Sciences, Springer, Berlin, Heidelberg, 2011, pp. 73–93. https://doi.org/10.1007/978-3-642-12670-3_5.

[127] J. Irizarry, E.P. Karan, Optimizing location of tower cranes on construction sites through GIS and BIM integration, Electronic Journal of Information Technology in Construction 17 (2012).

[128] J. Irizarry, E.P. Karan, F. Jalaei, Integrating BIM and GIS to improve the visual monitoring of construction supply chain management, Autom. Constr 31 (2013) 241–254. https://doi.org/10.1016/j.autcon.2012.12.005.

[129] I.H. Hijazi, M. Ehlers, S. Zlatanova, BIM for geo-analysis (BIM4GEOA): Set up of 3D information system with open source software and open specification (OS), Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. (2010).

[130] S. Donkers, Automatic generation of CityGML LoD3 building models from IFC models. Master Thesis, 2013.

[131] R. Sebastian, M. Böhms, P. van den Helm, BIM and GIS for Low Disturbance Construction, in: 13th International Conference on Construction Applications of Virtual Reality ConVR, London, UK, 2013.

[132] V.K. Bansal, Application of geographic information systems in construction safety planning, International Journal of Project Management 29 (2011) 66–77. https://doi.org/10.1016/j.ijproman.2010.01.007.

[133] V.K. Bansal, M. Pal, Construction Projects Scheduling Using GIS Tools, International Journal of Construction Management 11 (2011) 1–18. https://doi.org/10.1080/15623599.2011.10773158.

[134] E. Elbeltagi, M. Dawood, Integrated visualized time control system for repetitive construction projects, Autom. Constr 20 (2011) 940–953. https://doi.org/10.1016/j.autcon.2011.03.012.

[135] H. Kim, K. Orr, Z. Shen, H. Moon, K. Ju, W. Choi, Highway Alignment Construction Comparison Using Object-Oriented 3D Visualization Modeling, J. Constr. Eng. Manage. 140 (2014) 5014008. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000898.

[136] A. Rafiee, E. Dias, S. Fruijtier, H. Scholten, From BIM to Geo-analysis: View Coverage and Shadow Analysis by BIM/GIS Integration, Procedia Environmental Sciences 22 (2014) 397–402. https://doi.org/10.1016/j.proenv.2014.11.037.

[137] S. Niu, W. Pan, Y. Zhao, A BIM-GIS Integrated Web-based Visualization System for Low Energy Building Design, Procedia Engineering 121 (2015) 2184–2192. https://doi.org/10.1016/j.proeng.2015.09.091.

[138] R. Yaagoubi, A. Baik, J. Boehm, Integration of Jeddah Historical BIM and 3D GIS for Documentation and Restoration of Historical Monument, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XL-5/W7 (2015). https://doi.org/10.5194/isprsarchives-XL-5-W7-29-2015.

[139] Y. Deng, J.C. Cheng, C. Anumba, A framework for 3D traffic noise mapping using data from BIM and GIS integration, Structure and Infrastructure Engineering 12 (2016) 1267–1280. https://doi.org/10.1080/15732479.2015.1110603.

[140] B. Wu, S. Zhang, Integration of GIS and BIM for Indoor Geovisual Analytics, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLI-B2 (2016) 455–458. https://doi.org/10.5194/isprs-archives-XLI-B2-455-2016.

[141] K. Arroyo Ohori, F. Biljecki, A.A. Diakité, T. Krijnen, H. Ledoux, J.E. Stoter, Towards an integration of GIS and BIM data: what are the geometric and topological issues?, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. IV-4/W5 (2017) 1–8. https://doi.org/10.5194/isprs-annals-IV-4-W5-1-2017.

[142] K. Arroyo Ohori, A.A. Diakité, T. Krijnen, H. Ledoux, J.E. Stoter, Processing BIM and GIS Models in Practice: Experiences and Recommendations from a GeoBIM Project in The Netherlands, IJGI 7 (2018) 311. https://doi.org/10.3390/ijgi7080311.

[143] Y. Chen, E. Shooraj, A. Rajabifard, S. Sabri, From IFC to 3D Tiles: An Integrated Open-Source Solution for Visualising BIMs on Cesium, IJGI 7 (2018) 393. https://doi.org/10.3390/ijgi7100393.

[144] S. Yamamura, L. Fan, Y. Suzuki, Assessment of Urban Energy Performance through Integration of BIM and GIS for Smart City Planning, Procedia Engineering 180 (2017) 1462–1472. https://doi.org/10.1016/j.proeng.2017.04.309.

[145] S. Kardinal Jusuf, B. Mousseau, G. Godfroid, V. Soh Jin Hui, Integrated modeling of CityGML and IFC for city/neighborhood development for urban microclimates analysis, Energy Procedia 122 (2017) 145–150. https://doi.org/10.1016/j.egypro.2017.07.329.

[146] T.W. Kang, C.H. Hong, IFC-CityGML LOD mapping automation using multiprocessing-based screen-buffer scanning including mapping rule, KSCE J Civ Eng 22 (2018) 373–383. https://doi.org/10.1007/s12205-017-0595-9.

[147] X. Zhou, J. Zhao, J. Wang, D. Su, H. Zhang, M. Guo, M. Guo, Z. Li, OutDet: an algorithm for extracting the outer surfaces of building information models for integration with geographic information systems, International Journal of Geographical Information Science 33 (2019) 1444–1470. https://doi.org/10.1080/13658816.2019.1572894.

[148] K. McGlinn, D. Blake, D. O'Sullivan, GViz - An Interactive WebApp to Support GeoSPARQL over Integrated Building Information, in: Companion Proceedings of The 2019 World Wide Web Conference on - WWW '19, San Francisco, USA, ACM Press, New York, New York, USA, 2019, pp. 904–912. https://doi.org/10.1145/3308560.3316536.

[149] J. O'Donovan, D. O'Sullivan, K. McGlinn, A method for converting IFC geometric data into GeoSPARQL, in: Proceedings of the 7th Linked Data in Architecture and Construction Workshop, Lisbon, Portugal, CEUR-WS, Aachen, Germany, 2019 (accessed 20 December 2020). http://ceur-ws.org/Vol-2389/.

[150] S. Esser, A. Borrmann, Integrating Railway Subdomain-Specific Data Standards into a common IFC-based Data Model, in: 26th International Workshop on Intelligent Computing in Engineering, Leuven, Belgium, 2019 (accessed 20 December 2020). http://ceur-ws.org/Vol-2394/.

[151] J. Zhu, X. Wang, P. Wang, Z. Wu, M.J. Kim, Integration of BIM and GIS: Geometry from IFC to shapefile using open-source technology, Autom. Constr 102 (2019) 105–119. https://doi.org/10.1016/j.autcon.2019.02.014.

[152] J. Zhu, Y. Tan, X. Wang, P. Wu, BIM/GIS integration for web GIS-based bridge management, Annals of GIS (2020) 1–11. https://doi.org/10.1080/19475683.2020.1743355.

[153] J. Zhu, P. Wu, Towards Effective BIM/GIS Data Integration for Smart City by Integrating Computer Graphics Technique, Remote Sensing 13 (2021) 1889. https://doi.org/10.3390/rs13101889.

[154] Q. Chen, J. Chen, W. Huang, Method for Generation of Indoor GIS Models Based on BIM Models to Support Adjacent Analysis of Indoor Spaces, IJGI 9 (2020) 508. https://doi.org/10.3390/ijgi9090508.

[155] K. Jetlund, E. Onstein, L. Huang, IFC Schemas in ISO/TC 211 Compliant UML for Improved Interoperability between BIM and GIS, IJGI 9 (2020) 278. https://doi.org/10.3390/ijgi9040278.

[156] Z. Xu, L. Zhang, H. Li, Y.-H. Lin, S. Yin, Combining IFC and 3D tiles to create 3D visualization for building information modeling, Autom. Constr 109 (2020) 102995. https://doi.org/10.1016/j.autcon.2019.102995.

[157] Q. Chen, J. Chen, W. Huang, Visualizing Large-Scale Building Information Modeling Models within Indoor and Outdoor Environments Using a Semantics-Based Method, IJGI 10 (2021) 756. https://doi.org/10.3390/ijgi10110756.

[158] Y. Yang, S.T. Ng, J. Dao, S. Zhou, F.J. Xu, X. Xu, Z. Zhou, BIM-GIS-DCEs enabled vulnerability assessment of interdependent infrastructures – A case of stormwater drainage-building-road transport Nexus in urban flooding, Autom. Constr 125 (2021) 103626. https://doi.org/10.1016/j.autcon.2021.103626.

[159] N. Moretti, C. Ellul, F. Re Cecconi, N. Papapesios, M.C. Dejaco, GeoBIM for built environment condition assessment supporting asset management decision making, Autom. Constr 130 (2021) 103859. https://doi.org/10.1016/j.autcon.2021.103859.

[160] N. Salheb, K. Arroyo Ohori, J.E. Stoter, Automatic Conversion of CityGML to IFC, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLIV-4/W1-2020 (2020) 127–134. https://doi.org/10.5194/isprs-archives-XLIV-4-W1-2020-127-2020.

[161] F. Noardo, T. Wu, K. Arroyo Ohori, T. Krijnen, H. Tezerdi, J.E. Stoter, GeoBIM for digital Building Permit Process: Learning from a Case Study in Rotterdam, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VI-4/W1-2020 (2020) 151–158. https://doi.org/10.5194/isprs-annals-VI-4-W1-2020-151-2020.

[162] C. Clemen, M. Schröder, T. Kaiser, E. Romanschek, IfcTerrain - A free and open source Tool to convert Digital Terrain Models (DTM) to OpenBIM Industry Foundation Classes (IFC), ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VIII-4/W2-2021 (2021) 145–151. https://doi.org/10.5194/isprs-annals-VIII-4-W2-2021-145-2021.

[163] Z. Akob, M. Abang Hipni, A. Abd Razak, Deployment of GIS + BIM in the construction of Pan Borneo Highway Sarawak, Malaysia, IOP Conf. Ser.: Mater. Sci. Eng. (2019). https://doi.org/10.1088/1757-899X/512/1/012037.

[164] X. Ding, J. Yang, L. Liu, W. Huang, P. Wu, Integrating IFC and CityGML Model at Schema Level by Using Linguistic and Text Mining Techniques, IEEE Access 8 (2020) 56429–56440. https://doi.org/10.1109/ACCESS.2020.2982044.

[165] G. Vacca, E. Quaquero, BIM-3D GIS: an integrated system for the knowledge process of the buildings, Journal of Spatial Science 65 (2020) 193–208. https://doi.org/10.1080/14498596.2019.1601600.

[166] H. Bayat, M.R. Delavar, W. Barghi, S.A. EslamiNezhad, P. Hanachi, S. Zlatanova, Modeling of emergency Evacuation in Building Fire, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLIII-B4-2020 (2020) 321–327. https://doi.org/10.5194/isprs-archives-XLIII-B4-2020-321-2020.

[167] V.K. Bansal, Use of GIS to consider spatial aspects in construction planning process, International Journal of Construction Management 20 (2020) 207–222. https://doi.org/10.1080/15623599.2018.1484845.

[168] M. Gabriele, M. Previtali, HBIM-GIS Integration with an IFC-To-Shapefile Approach: The Palazzo Trotti Vimercate Pilot Case Study, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VIII-4/W2-2021 (2021) 167–174. https://doi.org/10.5194/isprs-annals-VIII-4-W2-2021-167-2021.

[169] L. Gobeawan, S.E. Lin, X. Liu, S.T. Wong, C.W. Lim, Y.-F.L. Gaw, N.H. Wong, P.Y. Tan, C.L. Tan, Y. He, IFC-Centric Vegetation Modelling for BIM, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VIII-4/W2-2021 (2021) 91–98. https://doi.org/10.5194/isprs-annals-VIII-4-W2-2021-91-2021.

[170] D.E. Andrianesi, E. Dimopoulou, An Integrated BIM-GIS Platform for representing and visualizing 3D Cadastral Data, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VI-4/W1-2020 (2020) 3–11. https://doi.org/10.5194/isprs-annals-VI-4-W1-2020-3-2020.

[171] A. AlSaggaf, A. Jrade, ArcSPAT: an integrated building information modeling (BIM) and geographic information system (GIS) model for site layout planning, International Journal of Construction Management (2021) 1–25. https://doi.org/10.1080/15623599.2021.1894071.

[172] L. Barazzetti, Integration between Building Information Modeling and Geographic Information System for historic Building and Sites: Historic-BIM-GIS, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VIII-M-1-2021 (2021) 41–48. https://doi.org/10.5194/isprs-annals-VIII-M-1-2021-41-2021.

[173] G.S. Floros, C. Ellul, Loss of information during design & construction for highways asset management: A GeoBIM perspective, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VIII-4/W2-2021 (2021) 83–90. https://doi.org/10.5194/isprs-annals-VIII-4-W2-2021-83-2021.

[174] A. Guntel, A.C. Aydinoglu, Producing and Visualizating 3D Building Geodatabase as a part of 3D Cadastre Project, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLVI-4/W5-2021 (2021) 239–243. https://doi.org/10.5194/isprs-Archives-XLVI-4-W5-2021-239-2021.

[175] W.N.F.W.A. Basir, U. Ujang, Z. Majid, S. Azri, Data Interoperability of Building Information Modeling and Geographic Information System in Construction Industry, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLVI-4/W4-2021 (2021) 111–117. https://doi.org/10.5194/isprs-archives-XLVI-4-W4-2021-111-2021.

[176] M. Barzegar, A. Rajabifard, M. Kalantari, B. Atazadeh, An IFC-based database schema for mapping BIM data into a 3D spatially enabled land administration database, International Journal of Digital Earth 14 (2021) 736–765. https://doi.org/10.1080/17538947.2021.1875062.

[177] W. Tegtmeier, P. Oosterom, S. Zlatanova, H. Hack, Information management in civil engineering infrastructural development; With focus on geological and geotechnical information, 2009.

[178] R. De Laat, L. van Berlo, Integration of BIM and GIS: The development of the CityGML GeoBIM extension, in: Advances in 3D Geo-Information Sciences, Springer Berlin Heidelberg, 2011, pp. 211–225. https://doi.org/10.1007/978-3-624-12670-3_13.

[179] R. El Meouche, M. Rezoug, I.H. Hijazi, Integrating and managing BIM in GIS, software review, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. XL-2/W2 (2013). https://doi.org/10.5194/isprsarchives-XL-2-W2-31-2013.

[180] A. Borrmann, T.H. Kolbe, A. Donaubauer, H. Steuer, J.R. Jubierre, M. Flurl, Multi-Scale Geometric-Semantic Modeling of Shield Tunnels for GIS and BIM Applications, Computer-Aided Civil and Infrastructure Engineering 30 (2015) 263–281. https://doi.org/10.1111/mice.12090.

[181] A. Borrmann, T.H. Kolbe, A. Donaubauer, H. Steuer, J.R. Jubierre, Transferring multi-scale approaches from 3D city modeling to IFC-based tunnel modeling, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. II-2/W1 (2013) 75–85. https://doi.org/10.5194/isprsannals-ii-2-w1-75-2013.

[182] R. Stouffs, H. Tauscher, F. Biljecki, Achieving Complete and Near-Lossless Conversion from IFC to CityGML, ISPRS Int. J. Geo Inf. 7 (2018) 355. https://doi.org/10.3390/ijgi7090355.

[183] M. Wang, Y. Deng, J. Won, J.C. Cheng, An integrated underground utility management and decision support based on BIM and GIS, Autom. Constr 107 (2019) 102931. https://doi.org/10.1016/j.autcon.2019.102931.

[184] K. Jetlund, A structure of UML profiles for modelling of geospatial information in GIS, ITS and BIM, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VI-4/W1-2020 (2020) 101–108. https://doi.org/10.5194/isprs-annals-VI-4-W1-2020-101-2020.

[185] M. Petronijević, N. Višnjevac, N. Praščević, B. Bajat, The Extension of IFC For Supporting 3D Cadastre LADM Geometry, IJGI 10 (2021) 297. https://doi.org/10.3390/ijgi10050297.

[186] L. Wilhelm, A. Donaubauer, T.H. Kolbe, Integration of BIM and Environmental Planning: The CityGML EnvPlan ADE, Journal of Digital Landscape Architecture (2021) 323–324.

[187] J. Benner, A. Geiger, K. Leinemann, Flexible generation of semantic 3D building models, in: Proceedings of the 1st International Workshop on Next Generation 3D City Models. EuroSDR Publication #49, Bonn, 2005 (accessed 20 December 2020). https://citese-erx.ist.psu.edu/viewdoc/summary?doi=10.1.1.531.3713&rank=1.

[188] J.W. Choi, S.A. Kim, J. Lertlakkhanakul, J.H. Yeom, Developing Ubiquitous Space Information Model for Indoor GIS Service in Ubicomp Environment 2 (2008). https://doi.org/10.1109/NCM.2008.97.

[189] M. Ghafourian, H.A. Karimi, CAD/GIS Integration Issues for Seamless Navigation between Indoor and Outdoor Environments, in: H.A. Karimi, B. Akinci (Eds.), CAD and GIS Integration, Auerbach Publications, Boca Raton, FL, USA, 2009.

[190] E.P. Karan, J. Irizarry, Developing a Spatial Data Framework for Facility Management Supply Chains, in: Construction Research Congress, Atlanta, Georgia, ASCE, Reston, 2014, pp. 2355–2364. https://doi.org/10.1061/9780784413517.239.

[191] C. Mignard, C. Nicolle, Merging BIM and GIS using ontologies application to urban facility management in ACTIVe3D, Computers in Industry 65 (2014) 1276–1290. https://doi.org/10.1016/j.compind.2014.07.008.

[192] S. Amirebrahimi, A. Rajabifard, P. Mendis, T. Ngo, A data model for integrating GIS and BIM for assessment and 3D visualisation of flood damage to building, CEUR Workshop Proceedings 1323 (2015) 78–89.

[193] A. Aien, A. Rajabifard, M. Kalantari, D. Shojaei, Integrating Legal and Physical Dimensions of Urban Environments, IJGI 4 (2015) 1442–1479. https://doi.org/10.3390/ijgi4031442.

[194] D. Shojaei, A. Rajabifard, M. Kalantari, I.D. Bishop, A. Aien, Design and development of a web-based 3D cadastral visualisation prototype, International Journal of Digital Earth 8 (2015) 538–557. https://doi.org/10.1080/17538947.2014.902512.

[195] A.-H. Hor, A. Jadidi, G. Sohn, BIM-GIS Integrated Geospatial Information Model using Semantic Web and RDF Graphs, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. III-4 (2016) 73–79. https://doi.org/10.5194/isprsannals-III-4-73-2016.

[196] T.-A. Teo, K.-H. Cho, BIM-oriented indoor network model for indoor and outdoor combined route planning, Advanced Engineering Informatics 30 (2016) 268–282. https://doi.org/10.1016/j.aei.2016.04.007.

[197] A. Peckienė, L. Ustinovičius, Possibilities for Building Spatial Planning using BIM Methodology, Procedia Engineering 172 (2017) 851–858. https://doi.org/10.1016/j.proeng.2017.02.085.

[198] Á. Sicilia, G. Costa, Energy-Related Data Integration Using Semantic Data Models for Energy Efficient Retrofitting Projects, in: Proceedings of the Sustainable Places Conference, Middlesbrough, UK, 2017. https://doi.org/10.3390/proceedings1071099.

[199] K. Kumar, A. Labetski, K.A. Ohori, H. Ledoux, J.E. Stoter, The LandInfra standard and its role in solving the BIM-GIS quagmire, Open Geospatial Data, Software and Standards (2019). https://doi.org/10.1186/s40965-019-0065-z.

[200] W. Li, S. Zlatanova, A.A. Diakité, M. Aleksandrov, J. Yan, Towards Integrating Heterogeneous Data: A Spatial DBMS Solution from a CRC-LCL Project in Australia, IJGI 9 (2020) 63. https://doi.org/10.3390/ijgi9020063.

[201] P.-C. Lee, L.-L. Zheng, T.-P. Lo, D.-B. Long, A Risk Management System for Deep Excavation Based on BIM-3DGIS Framework and Optimized Grey Verhulst Model, KSCE J Civ Eng 24 (2020) 715–726. https://doi.org/10.1007/s12205-020-1462-7.

[202] Z. Lv, X. Li, H. Lv, W. Xiu, BIM Big Data Storage in WebVRGIS, IEEE Trans. Ind. Inf. 16 (2020) 2566–2573. https://doi.org/10.1109/TII.2019.2916689.

[203] Y. Demir Altıntaş, M.E. Ilal, Loose coupling of GIS and BIM data models for automated compliance checking against zoning codes, Autom. Constr 128 (2021) 103743. https://doi.org/10.1016/j.autcon.2021.103743.

[204] S. Karimi, I. Iordanova, D. St-Onge, Ontology-based approach to data exchanges for robot navigation on construction sites, ITcon 26 (2021) 546–565. https://doi.org/10.36680/j.itcon.2021.029.

[205] A.-H. Hor, G. Sohn, Design and evaluation of a BIM-GIS integrated information model using RDF graph database, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VIII-4/W2-2021 (2021) 175–182. https://doi.org/10.5194/isprs-annals-VIII-4-W2-2021-175-2021#.

[206] Y. Li, Z. He (Eds.), 3D indoor navigation: a framework of combining BIM with 3D GIS, 2008.

[207] N.N. Esfahani, Interoperability methods for Infrastructure Design and Information Systems, Forum Bauinformatik 2013 (2013) 161–171.

[208] Y. Zheng, Improving the interoperability between city and road semantics an integration of cityGML and OKSTRA data based on semantic web technologies. Master Thesis, Eindhoven University of Technology, Department Built Environment, 2017.

[209] N.N. Esfahani, Interoperability of traffic infrastructure planning and Geospatial Information Systems, Dissertation, Institut für Bauinformatik, Fakultät Bauingenieurwesen, TU Dresden; Saechsische Landesbibliothek- Staats- und Universitaetsbibliothek Dresden, Dresden, 2018.

[210] L. Zhao, Z. Liu, J. Mbachu, Highway Alignment Optimization: An Integrated BIM and GIS Approach, IJGI 8 (2019) 172. https://doi.org/10.3390/ijgi8040172.

[211] J.F.T. Djuedja, H.A. Fonbeyin, B. Kamsu-Foguem, M.H. Karray, C. Magniont, P. Pauwels, Integration of environmental data in BIM tool & linked building data, in: Proceedings of the 7th Linked Data in Architecture and Construction Workshop, Lisbon, Portugal, CEUR-WS, Aachen, Germany, 2019, pp. 78–91 (accessed 20 December 2020). http://ceur-ws.org/Vol-2389/.

[212] W. Huang, P. Olsson, J. Kanters, L. Harrie, Reconciling city models with BIM in knowledge graphs: A feasibility study of data integration for solar energy simulation, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. VI-4/W1-2020 (2020) 93–99. https://doi.org/10.5194/isprs-annals-VI-4-W1-2020-93-2020.

[213] I.H. Hijazi, T. Krauth, A. Donaubauer, T.H. Kolbe, 3DCityDB4BIM: A system architecture for linking BIM Server and 3D CityDB for BIM-GIS Integration, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. V-4-2020 (2020) 195–202. https://doi.org/10.5194/isprs-Annals-V-4-2020-195-2020.

[214] G.S. Floros, P. Ruff, C. Ellul, Impact of Information Management during Design & Construction on downstream BIM-GIS Interoperability for Rail Infrastructure, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. (2020) 61–68. https://doi.org/10.5194/isprs-annals-VI-4-W1-2020-61-2020.

[215] B. Godager, Critical review of the integration of bim to semantic web technology, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-4 (2018) 233–240. https://doi.org/10.5194/isprs-archives-XLII-4-233-2018.

[216] J. Raad, N. Pernelle, F. Saïs, W. Beek, F. Harmelen, The sameAs Problem: A Survey on Identity Management in the Web of Data, arXiv (2019) (accessed 20 December 2020).

http://www.semantic-web-journal.net/content/sameas-problem-survey-identity-management-web-data. PREPRINT.

[217] S. Aljalbout, D. Buchs, G. Falquet, OWL^ C: A Contextual Two-Dimensional Web Ontology Language, in: 2nd Conference on Language, Data and Knowledge (LDK 2019), Dagstuhl, Germany, seventh ed., 2019.

[218] P. Bouquet, F. Giunchiglia, F. Harmelen, L. Serafini, H. Stuckenschmidt, C-OWL: Contextualizing ontologies, in: ISWC 2003, Proceedings of the 2nd International Semantic Web Conference, Sanibel Island, USA, Springer, Berlin/Heidelberg, Germany, 2003.

[219] R.V. Guha, Contexts: a formalization and some applications, Stanford University Stanford, CA, 1995.

[220] J. Bao, J. Tao, D. Mcguinness, P. Smart, Context Representation for the Semantic Web (2013).

[221] O. Kutz, C. Lutz, F. Wolter, M. Zakharyaschev, E-connections of abstract description systems, Artificial Intelligence 156 (2004) 1–73. https://doi.org/10.1016/j.artint.2004.02.002.

[222] R.V. Guha, R. McCool, R. Fikes, Contexts for the Semantic Web, in: D. Hutchison, T. Kanade, J. Kittler, J.M. Kleinberg, F. Mattern, J.C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M.Y. Vardi, G. Weikum, S.A. McIlraith, D. Plexousakis, F. Van Harmelen (Eds.), The Semantic Web – ISWC 2004, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 32–46. https://doi.org/10.1007/978-3-540-30475-3_4.

[223] P. Bouquet, L. Serafini, H. Stoermer, Introducing context into Semantic Web knowledge bases, in: Proceedings of the CAiSE*06 Doctoral Consortium DC@CAISE '06, Luxemburg, 2006, pp. 1144–1151.

[224] J. Euzenat, F. Scharffe, A. Zimmermann, Expressive alignment language and implementation, Tech. rep., INRIA, 2007 (accessed 20 December 2020). https://hal.inria.fr/hal-00822892.

[225] A.K. Idrissou, R. Hoekstra, F. Van Harmelen, A. Khalili, P. van den Besselaar, Is my:sameAs the same as your:sameAs?, in: Proceedings of the K-CAP, Austin, USA, ACM Press, New York, 2017, pp. 1–8. https://doi.org/10.1145/3148011.3148029.

[226] C. Batchelor, C.Y.A. Brenninkmeijer, C. Chichester, M. Davies, D. Digles, I. Dunlop, C.T. Evelo, A. Gaulton, C. Goble, A.J.G. Gray, P. Groth, L. Harland, K. Karapetyan, A. Loizou, J.P. Overington, S. Pettifer, J. Steele, R. Stevens, V. Tkachenko, A. Waagmeester, A. Williams, E.L. Willighagen, Scientific Lenses to Support Multiple Views over Linked Chemistry Data, in: P. Mika, T. Tudorache, A. Bernstein, C. Welty, C. Knoblock, D. Vrandečić, P. Groth, N. Noy, K. Janowicz, C. Goble (Eds.), The Semantic Web – ISWC 2014, Springer International Publishing, Cham, 2014, pp. 98–113. https://doi.org/10.1007/978-3-319-11964-9_7.

[227] K. Alexander, R. Cyganiak, M. Hausenblas, J. Zhao, Describing Linked Datasets with the VoID Vocabulary, 2011. https://www.w3.org/TR/void/ (accessed 29 November 2021).

[228] J. Raad, N. Pernelle, F. Saïs, Detection of Contextual Identity Links in a Knowledge Base, in: Proceedings of the K-CAP, Austin, USA, ACM Press, New York, 2017, pp. 1–8. https://doi.org/10.1145/3148011.3148032.

[229] OWL Working Group W3C, OWL 2 Web Ontology Language Document Overview (Second Edition), 2012. https://www.w3.org/TR/owl2-overview/ (accessed 09th December 2020).

[230] D.K. Lewis, On the plurality of worlds, Blackwell Publishers, Malden, Mass., 2001.

[231] H. Halpin, P.J. Hayes, H.S. Thompson, When owl:sameAs isn't the Same Redux: Towards a Theory of Identity, Context, and Inference on the Semantic Web, in: H. Christiansen, I. Stojanovic, G.A. Papadopoulos (Eds.), Modeling and Using Context, Springer International Publishing, Cham, 2015, pp. 47–60. https://doi.org/10.1007/978-3-319-25591-0_4.

[232] EDOAL: Expressive and Declarative Ontology Alignment Language. http://alignapi.gforge.inria.fr/edoal.html (accessed 15th April 2021).

[233] W. Beek, S. Schlobach, F. Van Harmelen, A Contextualised Semantics for owl:sameAs, in: ISWC 2016, Springer International Publishing, Cham, 2016, pp. 405–419. https://doi.org/10.1007/978-3-319-34129-3_25.

[234] J. Frey, K. Müller, S. Hellmann, E. Rahm, M.-E. Vidal, Evaluation of metadata representations in RDF stores, Semantic Web 10 (2019) 205–229. https://doi.org/10.3233/SW-180307.

[235] S. Sen, M.C. Malta, B. Dutta, A. Dutta, State-of-the-Art Approaches for Meta-Knowledge Assertion in the Web of Data, IETE Technical Review (2020) 1–38. https://doi.org/10.1080/02564602.2020.1819891.

[236] V. Nguyen, O. Bodenreider, A.P. Sheth, Don't Like RDF Reification? Making Statements about Statements Using Singleton Property, Proc. Int. World Wide Web Conf. (2014) 759–770. https://doi.org/10.1145/2566486.2567973.

[237] J.J. Carroll, C. Bizer, P. Hayes, P. Stickler, Named graphs, provenance and trust, in: Proceedings of the 14th international conference on World Wide Web - WWW '05, Chiba, Japan, ACM Press, New York, USA, 2005, p. 613. https://doi.org/10.1145/1060745.1060835.

[238] O. Hartig, Foundations of RDF and SPARQL (An Alternative Approach to Statement-Level Metadata in RDF), in: 11th Alberto Mendelzon International Workshop on Foundation of Databases and the Web (AMW), Montevideo, Uruguay, 2017.

[239] H. Partsch, Requirements-Engineering systematisch: Modellbildung für softwaregestützte Systeme, Springer, Berlin, 2010. https://doi.org/10.1007/978-3-642-05358-0.

[240] H. Kim, Z. Chen, C.-S. Cho, H. Moon, K. Ju, W. Choi, Integration of BIM and GIS: Highway Cut and Fill Earthwork Balancing, in: Computing in Civil Engineering 2015, Austin, Texas,

American Society of Civil Engineers, Reston, VA, 2015, pp. 468–474. https://doi.org/10.1061/9780784479247.058.

[241] V.K. Bansal, Use of GIS and Topology in the Identification and Resolution of Space Conflicts, J. Comput. Civ. Eng. 25 (2011) 159–171. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000075.

[242] R. Hajji, R. Yaagoubi, I. Meliana, I. Laafou, A.E. Gholabzouri, Development of an Integrated BIM-3D GIS Approach for 3D Cadastre in Morocco, IJGI 10 (2021) 351. https://doi.org/10.3390/ijgi10050351.

[243] U. Isikdag, S. Zlatanova, J. Underwood, A BIM-Oriented Model for supporting indoor navigation requirements, Computers, Environment and Urban Systems 41 (2013) 112–123. https://doi.org/10.1016/j.compenvurbsys.2013.05.001.

[244] S. Vitalis, A. Labetski, K. Arroyo Ohori, H. Ledoux, J.E. Stoter, A data structure to incorporate versioning in 3D city models, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. IV-4/W8 (2019) 123–130. https://doi.org/10.5194/isprs-annals-IV-4-W8-123-2019.

[245] H. Eriksson, J. Sun, V. Tarandi, L. Harrie, Comparison of versioning methods to improve the information flow in the planning and building processes, Transactions in GIS 28 (2020) 21. https://doi.org/10.1111/tgis.12672.

[246] C. Koch, B. Firmenich, An approach to distributed building modeling on the basis of versions and changes, Advanced Engineering Informatics 25 (2011) 297–310. https://doi.org/10.1016/j.aei.2010.12.001.

[247] C. Batini, M. Scannapieco, Data Quality Issues in Data Integration Systems, in: C. Batini, M. Scannapieco (Eds.), Data and Information Quality, Springer International Publishing, Cham, 2016, pp. 279–307. https://doi.org/10.1007/978-3-319-24106-7_10.

[248] G. Ropohl, Allgemeine Technologie: Eine Systemtheorie der Technik, KIT Scientific Publishing, 2009.

[249] S. Herden, A. Zwanziger, Der Integrationsbegriff in der Wirtschaftsinformatik: Literaturanalyse, Begriffsexplikation und Modell der Integrationsgegenstände, Institut für Wirtschaftsinformatik, Universität Bern, 2009. https://doi.org/10.7892/boris.58011 (accessed 29 November 2021).

[250] R.A. Teubner, Organisations- und Informationssystemgestaltung: Theoretische Grundlagen und integrierte Methoden, Deutscher Universitätsverlag, Wiesbaden, s.l., 1999. https://doi.org/10.1007/978-3-322-99957-3.

[251] F.C. Schmid, Methodisches Gestalten und systematisches Entwickeln am Beispiel zukünftiger Fassadenlösungen, Springer Fachmedien Wiesbaden, Wiesbaden, 2015. https://doi.org/10.1007/978-3-658-12213-3.

[252] Autodesk - Homepage. https://www.autodesk.com/ (accessed 21 July 2022).

[253] I.H. Hijazi, M. Ehlers, S. Zlatanova, T. Becker, L. van Berlo, Initial Investigations for Modeling Interior Utilities Within 3D Geo Context: Transforming IFC-Interior Utility to CityGML/UtilityNetworkADE, in: T.H. Kolbe, G. König, C. Nagel (Eds.), Advances in 3D Geo-Information Sciences, Springer, Berlin, Heidelberg, 2011. https://doi.org/10.1007/978-3-642-12670-3_6.

[254] J. Beetz, W. van den Braak, R. Botter, S. Zlatanova, R. de Laat, Interoperable data models for infrastructural artefacts – a novel IFC extension method using RDF vocabularies exemplified with quay wall structures for harbors, in: A. Mahdavi, B. Martens, R.J. Scherer (Eds.), ECPPM 2014, CRC Press, 2014, pp. 135–140. https://doi.org/10.1201/b17396-26.

[255] L. Liu, B. Li, S. Zlatanova, P. van Oosterom, Indoor navigation supported by the Industry Foundation Classes (IFC): A survey, Autom. Constr 121 (2021) 103436. https://doi.org/10.1016/j.autcon.2020.103436.

[256] S. Esser, A. Borrmann, A system architecture ensuring consistency among distributed heterogeneous information models for civil infrastructure projects, in: Proceedings of the 13th European Conference on Product and Process Modeling (ECPPM), Moscow, 2020. PRE-PRINT.

[257] S. Rich, K.H. Davis, Geographic information systems (GIS) for facility management, 2010 (accessed 20 December 2020). http://blog.dinamika.ac.id/romeo/files/2013/02/GIS_WP_FINAL.pdf.

[258] J. Beetz, A. Borrmann, Benefits and Limitations of Linked Data Approaches for Road Modeling and Data Exchange, in: I.F.C. Smith, B. Domer (Eds.), Advanced Computing Strategies for Engineering, Springer International Publishing, Cham, 2018, pp. 245–261. https://doi.org/10.1007/978-3-319-91638-5_13.

[259] P. Pauwels, S. Törmä, J. Beetz, M. Weise, T. Liebich, Linked Data in Architecture and Construction, Autom. Constr 57 (2015) 175–177. https://doi.org/10.1016/j.autcon.2015.06.007.

[260] X. Zheng, Research on the Railway Engineering Informationization Construction Scheme Based on Knowledge Engineering, Journal of Railway Engineering Society 36 (2019) 90–97.

[261] W. Weaver, Recent Contributions to the Mathematical Theory of Communication, ETC: A Review of General Semantics 10 (1953) 261–281.

[262] G. Gerbner, Toward a General Model of Communication 4 (1956) 171–199.

[263] H.D. Lasswell, The structure and function of communication in society, The communication of ideas 37 (1948) 215–228.

[264] T.M. Newcomb, An Approach to the Study of Communicative Acts 60 (1953) 393–404.

[265] B.H. Westley, M.S. MacLean, A conceptual model for communications research, ETR&D 3 (1955) 3–12. https://doi.org/10.1007/BF02713344.

[266] R. Jakobson, Closing Statements: Linguistics and Poetics (1960).

[267] C.S. Peirce, Collected Papers, Paladin, London, 1931-58.

[268] C.K. Ogden, I.A. Richards, The Meaning of Meaning, Harcourt, Brace and World, 1923.

[269] F.d. Saussure, Course in general linguistics, Columbia University Press, New York, 2011.

[270] W.L. Schönwandt, Planung in der Krise?: Theoretische Orientierungen für Architektur, Stadt- und Raumplanung, Vieweg+Teubner Verlag, Wiesbaden, 2002.

[271] J. Brodeur, The Semantics of Geospatial Information, in: IGARSS, Quebec, Canada, 2014.

[272] R. Rambow, Entwerfen und Kommunikation, in: A. Hahn (Ed.), Ausdruck und Gebrauch. Dresdner wissenschaftliche Halbjahreshefte für Architektur - Wohnen - Umwelt/Ausdruck und Gebrauch. Themenheft: Raum und Erleben: Über Leiblichkeit, Gefühle und Atmosphären in der Architektur, first., Aufl., Shaker, Herzogenrath, 2012, pp. 103–124.

[273] J. Farrugia, Model-Theoretic Semantics for the Web, in: G. Hencsey (Ed.), Proceedings of the 12th international conference on World Wide Web, ACM, New York, NY, 2003.

[274] P.J. Hayes, RDF Semantics, 2004. https://www.w3.org/TR/2004/REC-rdf-mt-20040210/#intro (accessed 28 April 2021).

[275] A. Tarski, The concept of truth in formalized languages (1936).

[276] P. Hitzler, Semantic web: Grundlagen, first. Aufl., Springer Berlin, Berlin, 2008.

[277] P. Christen, Data Matching, Springer Berlin Heidelberg, 2012. https://doi.org/10.1007/978-3-642-31164-2_1.

[278] G.W. Leibniz, L.E. Loemker, Philosophical Papers and Letters, Springer, Dodrecht, Netherlands, 1976. https://doi.org/10.1007/978-94-010-1426-7.

[279] M.J. Egenhofer, R.D. FRANZOSA, Point-set topological spatial relations, International journal of geographical information systems 5 (1991) 161–174. https://doi.org/10.1080/02693799108927841.

[280] S. Zlatanova, Topological Relationships and Their Use, in: S. Shekhar, H. Xiong, X. Zhou (Eds.), Encyclopedia of GIS, Springer International Publishing, Cham, 2015, pp. 1–21. https://doi.org/10.1007/978-3-319-23519-6_1548-1.

[281] A. Borrmann, E. Rank, Topological analysis of 3D building models using a spatial query language, Advanced Engineering Informatics 23 (2009) 370–385. https://doi.org/10.1016/j.aei.2009.06.001.

[282] M. Jovanovik, T. Homburg, M. Spasić, A GeoSPARQL Compliance Benchmark, IJGI 10 (2021) 487. https://doi.org/10.3390/ijgi10070487.

[283] M. Perry, J. Herring, OGC GeoSPARQL - A Geographic Query Language for RDF Data, 2012. https://www.ogc.org/standards/geosparql.

[284] A. Zimmermann, J. Euzenat, Three Semantics for Distributed Systems and Their Relations with Alignment Composition, in: D. Hutchison, T. Kanade, J. Kittler, J.M. Kleinberg, F. Mattern, J.C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M.Y. Vardi, G. Weikum, I.F. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, L.M. Aroyo (Eds.), The Semantic Web - ISWC 2006, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 16–29. https://doi.org/10.1007/11926078_2.

[285] S. Klai, A. Zimmermann, M. Khadir, Networked Ontologies with Contextual Alignments, Computing and Informatics 38 (2019) 115–150. https://doi.org/10.31577/cai_2019_1_115.

[286] SKOS Simple Knowledge Organization System Reference, 2018. https://www.w3.org/TR/skos-reference/ (accessed 22 April 2021).

[287] G. Abowd, Towards a Better Understanding of Context and Context-Awareness, in: International Symposium on Handheld and Ubiquitous Computing, Springer Berlin Heidelberg, 1999. https://doi.org/10.1007/3-540-48157-5_29.

[288] KIT Downloads, 2021. https://www.iai.kit.edu/1302.php (accessed 29 November 2021).

[289] Robert McNeel & Associates, Rhino - Rhinoceros 3D. https://www.rhino3d.com/ (accessed 18 June 2022).

[290] Robert McNeel & Associates, Grasshopper 3D. https://www.grasshopper3d.com/.

[291] GeometryGym. https://geometrygym.wordpress.com/ (accessed 18 June 2022).

[292] Apache Jena, 2022 (accessed 18 June 2022).

[293] P. Pauwels, IFCtoRDF, 2021. https://github.com/pipauwel/IFCtoRDF (accessed 29 November 2021).

[294] Institute for Spatial Information- and Surveying-Technology, University of Applied Sciences Mainz, GMLImporter. https://github.com/i3mainz/GMLImporter (accessed 18 June 2022).

[295] Grasshopper script for matching BIM and UIM models. https://1drv.ms/u/s!AoSgdjb6RJMZhekmaGUHt9WA4RQ_hg?e=aoSomW (accessed 24 November 2022).

[296] L. Zehtaban, O. Elazhary, D. Roller, A framework for similarity recognition of CAD models, Journal of Computational Design and Engineering 3 (2016) 274–285. https://doi.org/10.1016/j.jcde.2016.04.002.

[297] H. Opitz, Verschlüsselungsrichtlinien und Definitionen zum werkstuckbeschreibenden Klassifizierungssystem, Verlag W. Girardet, 1966.