A METHOD FOR GIS AND BIM INTEGRATION TO SUPPORT AUTOMATED ZONING CODE COMPLIANCE CHECKING

A Thesis Submitted to
the Graduate School of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in Architecture

by
Yelin DEMİR ALTINTAŞ

July 2020
İZMİR

ACKNOWLEDGMENTS

First, I would like to express my sincere thanks to my supervisor Assoc. Prof. Dr. Mustafa Emre İlal, for his sincere guidance, extensive assistance, insightful comments, and continuous support throughout my research. It has been a great privilege to learn from him. This research would not have been possible without his support.

I would like to thank to my committee members Assoc. Prof. Dr. Georg Suter and Prof. Dr. Serdar Kale for their great and valuable guidance, constructive comments and suggestions throughout my thesis study. I would also like to thank to Assist. Prof. Hasan Burak Çavka and Prof. Dr. Hüsnü Murat Günaydın for their contributions. I also owe particular thanks to Assist. Prof. Sibel Macit İlal, for her contribution and support.

Furthermore, I want to thank my friends İlknur Uygun, Livanur Erbil Altıntaş, Ece Ceren Engür, Onurcan Çakır and Nazlı Taraz for providing an encouraging and friendly environment that made working such a pleasant experience. Every interaction with them left me more encouraged and optimistic.

I also would like to express my sincere appreciation to my parents who have always encouraged and supported me during the realization process of this thesis and during all my lifetime. I thank them for their continuous love over all the years.

I owe a great debt of gratitude to my husband Efe Can Altıntaş for his patience, love, encouragement and support.

ABSTRACT

A METHOD FOR GIS AND BIM INTEGRATION TO SUPPORT AUTOMATED ZONING CODE COMPLIANCE CHECKING

This thesis constitutes a study in the field of BIM-GIS interoperability, with a concentration on code compliance checking. Code checking requires appropriate computer-based models of building codes in digital format, zoning plans in GIS and building designs in BIM environment. However, in existing design environments, it is not possible to access neighborhood data. Interoperability and geographical data transfer from GIS to BIM environments are problematic.

This study proposes a methodology for the creation of a third-party platform that draws IFC and GML data from both BIM and GIS environments separately and then combines them in a unified domain model appropriate to the field to be used in code checking processes. The proposed methodology includes the identification of the required data in the area of study; identification of how much of the data can be modelled with BIM model or GIS model; construction of the BIM and GIS domain models using the identified data; and development of an application that integrates the two domain models and manages them as a whole. For validation of the methodology and testing the developed platform, a proof of concept prototype is developed. The system retrieves building codes, reads the BIM file sent by the architect, and retrieves the information on the neighborhood the building is located in from GIS and conducts the checking process. The system demonstrates that BIM and GIS information can be compiled together during a code checking process creating a third-party platform.

The applicability of the model has been evaluated through use-case scenarios. The use-cases demonstrate that the zoning domain model developed in this thesis can be used as a database to manage geographical information. Coupling the zoning domain model with BIM data supports digital code checking, eliminating the need for hard copies and simplifying approval processes as well as enabling architects to self-check designs for compliance before submitting them to municipalities.

ÖZET

İMAR KURALLARININ OTOMATİK KONTROLÜNÜN DESTEKLENMESİNE YÖNELİK CBS VE YBM BÜTÜNLEŞMESİ İÇİN BİR YÖNTEM

YBM-CBS birlikte işlerliği alanında geliştirilen bu doktora tezinde, uyumluluk denetimi otomasyonu üzerine yoğunlaşan bir çalışma yürütülmüştür. Uyumluluk denetimi, dijital ortamda bina ile ilgili yönetmeliklerin bilgisayar tabanlı modellerini, CBS'de kentsel veriler ile imar planlarını ve YBM ortamında da bina tasarımının bulunmasını gerektirir. Oysa ki, mevcut tasarım ortamlarında çevre verilerine ulaşmak mümkün değildir. Birlikte işlerlik ve CBS'den YBM ortamlarına coğrafi veri aktarımı sorunludur.

Bu çalışma, bina ve çevre verilerini YBM ve CBS ortamlarından ayrı ayrı çeken ve daha sonra bunları alana özgü bir modelde bütünleştiren üçüncü, bağımsız bir platformun oluşturulması için bir yöntem önermiş ve geliştirilen modeli uyumluluk denetimi süreçlerine uygulamıştır. Bu yöntem, çalışılan alanda gerekli olan verinin tanımlanması; verinin ne kadarının YBM veya CBS ile modellenebileceğinin belirlenmesi; belirlenen veriyi kullanarak YBM ve CBS alan modellerinin oluşturulması; ve bu iki alan modelini entegre edip bir bütün olarak yöneten bir uygulamanın geliştirilmesi adımlarını içermektedir. Yöntemin uygulanabilirliğinin ve geliştirilen platformun test edilmesi için, bir prototip geliştirilmiştir. Bu prototip, imar yönetmeliğini, mimar tarafından gönderilen YBM dosyasını ve CBS'deki çevre verisini çekmekte ve uyumluluk denetimi sürecini yürütmektedir. Bu yaklaşım, uyumluluk denetiminin gerektirdiği YBM ve CBS verilerinin bağımsız, ayrı bir platformda birlikte derlenebileceğini göstermektedir.

Modelin uygulanabilirliği kullanım senaryoları ile değerlendirilmiştir. Kullanım örnekleri, bu tezde geliştirilen alana özgü modelin çevre verisini yönetmekte bir veritabanı olarak kullanılabileceğini göstermektedir. İmar yönetmeliklerine özgü bu modelin YBM verisiyle bütünleştirilmesi uyumluluk denetimi otomasyonu süreçlerini desteklemekte, basılı kopya ihtiyacını ortadan kaldırmakta ve onay süreçlerini basitleştirmekte, ayrıca mimarların belediyelere göndermeden önce tasarımları uygunluk açısından kendilerinin kontrol etmelerini sağlamaktadır.

TABLE OF CONTENTS

LIST OF FIGURES

Figure		**Page**

Figure		Page

LIST OF TABLES

LIST OF ABBREVIATIONS

AEC	Architecture Engineering Construction
BIM	Building Information Modeling
CAD	Computer Aided Design
GIS	Geographic Information System
GML	Geography Markup Language
IFC	Industry Foundation Class
IMHZCode	İzmir Municipality Housing and Zoning Code
QGIS	Quantum Geographic Information System
UML	Unified Modeling Language
XML	Extensible Markup Language

CHAPTER 1

INTRODUCTION

For effective collaboration between all project participants, interoperability is needed between different software. The ability of a system to access, interpret and exchange data and to work with other systems cooperatively is defined as interoperability. Building Information Modeling (BIM) and Geographic Information System (GIS) domains are two domains that need information from each other. Designers working in a BIM environment need GIS data for various tasks, e.g., site selection, creating solutions at the neighborhood level, and identifying relationships with the surroundings of the building (Akin 2010). On the other hand, GIS domain needs BIM data whenever more detailed information on buildings is required. BIM and GIS integration enables effective management of design data in various stages of a project's lifecycle, from planning and design to construction, operation, and maintenance. While BIM and GIS technologies are closely related, they remain largely isolated. Distinctive development of BIM and GIS tools, and professional divisions were the primary causes for this isolation (Karimi and Akinci 2010; Isikdag and Zlatanova 2009; El-Mekawy and Östman 2010).

There is an increasing demand for BIM and GIS integration (Akin 2010), but the interoperability problem causes many shortcomings in different fields of work. Evacuation planning and fire response analyses require both the interior plans and the locations of gathering points in the neighbourhood for planning an indoor and outdoor escape route. However, building data can not be transferred into geospatial environment and the indoor navigation can not be controlled using a GIS. In construction processes, planning the delivery routes for materials and storing component data requires usage of BIM and GIS together. Due to the lack of an integrated medium, trying to control the transportation process using a BIM tool, or to store component data using a GIS causes problems. Automated compliance checking of building projects against zoning codes is one of the areas that suffers from shortcomings in BIM-GIS integration. Code checking requires zoning data in GIS and its integration with building designs in BIM environment. The ideal automated code checking process should retrieve building codes from the authority publishing them, read the BIM file sent by the architect, and retrieve the information on the neighborhood the building is located in, directly from the municipality's GIS. Currently, there are many research efforts that study data import from BIM to geo-information domain, data transformation, and data extensions for BIM-GIS interoperability. However, considering the studies in literature, site related information at city scale is still not available to designers in current BIM systems and the integration of neighbourhood data with building data is problematic.

1.1. Problem Statement

During architectural design processes, architects need data on the surrounding context. While utilization of some contextual data is at the discretion of the architect, some contextual data represent constraints that all designs should meet and their utilization is mandatory. Local building regulations define and reference such data. Two examples are setbacks, which are minimum distances from the building to the parcel borders, and grade elevation of the main access road for determining the reference ground level for projects. Unfortunately, in existing design environments, it is not possible to access such data on the surroundings. Building Information Modelling (BIM) tools integrate and manage information from various disciplines including entire design, construction and maintenance processes, but they are limited in storing and managing geographical information on the surrounding landscape of buildings (Mignard and Nicolle 2014; Rafiee et al. 2014). Furthermore, BIM models are not able to query topological relationships and store topological data which is in fact required by automated code compliance checking processes (Isikdag, Underwood, and Aouad 2008; Karan and Irizarry 2015). Since designers do not need to model the surroundings to that level of detail for their projects, it is normal for BIM that this information does not exist. It is Geographic Information Systems (GIS) that have capabilities for integrating, analyzing and managing geographical information. Yet, interoperability and geographical information exchange between GIS and BIM environments are problematic. These two distinctive domains have clear professional divisions and each develops its own tools lacking a common data standard.

BIM software applications do not have the capability to fully populate data on the surrounding context that building regulations require. BIM is limited in modeling context, and placing the building within its macro-scale built environment (Malsane 2015). In this study, BIM and Industry Foundation Classes (IFC) data model are used interchangeably, and both refer to the building model rather than any BIM software. IFC data model acts as an international BIM standard for interoperability and is accepted as the standard most likely to succeed. Yet, it does not represent enough entities or attributes which are necessary for code checking. Many of the concepts mentioned in building codes, such as block, parcel, setback distance, roads, streets, sidewalks and topological relationships between geometric entities etc. are lacking in IFC. As a result, IFC, on its own, is not adequate for the use in compliance checking of building projects against zoning codes automatically (Malsane 2015). In fact, some of the concepts mentioned in building codes are not represented digitally anywhere. For example, consider the following rule;

> "IMHZCode, Clause 27: Setback distances where there is a front yard and setback distances from roads, green areas and parking lots should be at least 5.00 m."

Properly implementing this rule in an automated code checking system, requires buildings, roads, green areas, parking areas, adjacencies and distance relationships between these objects to be modelled as part of the data set.

Modeling the parcel and its relations to its neighbors is also not enough for a complete code checking process. For example, consider the following rule;

> "IMHZCode, Clause 27: In planned-unit ordered blocks, if there is an existing building in any of the parcels within the same block, setback distances are the same as setback distances used in that parcel."

For implementing this rule in an automated code checking system, the construction order and setback distance information of all parcels that face the same road in the same block should be modelled. First, containment relations should be queried to determine the block and the other parcels in the same block. Then, road-parcel relations should be queried for finding the parcels that face the same road in the same block. Finally, the identified parcels' construction orders should be queried to find the planned-unit ordered parcels.

CityGML, developed as an application schema for the Geography Markup Language (GML), is the commonly used open data format for representation, storage and exchange of city and landscape models. It includes many classes mentioned in building code texts (Groger et al. 2012), but not all of the required classes, attributes and relations exist in this data model. The "construction order" of the parcel, if the parcel has a "front garden" or not, if the parcel is a "corner parcel" or not, the "buildable area" of the parcel are all attributes and relations of the parcel that do not exist in CityGML. Similarly, “front distance,” side distance,” “rear distance” are attributes and relations related to setbacks that are currently missing in CityGML. For reliable code compliance checking, such data must be maintained in geographic information systems.

There is a need for defining how and where such building code concepts related to the surroundings should be stored and how this information can be made available to stakeholders in the AEC industry. At a minimum, zoning data stored in the GIS domain should be interoperable with BIM. If zoning data is modelled and described in a way that architects can easily reach during the design process, architects will be able to ensure during design process that every aspect of their design complies with the zoning regulations before submitting to municipalities. Architects will be able to combine the zoning data received from the municipality with their own building design data within an automated code-checking application and identify problematic points in their designs on their own. The same integrated digital model will be utilized by municipalities. Approval workflows can complete faster and results can be more accurate due to decreased manual labor.

1.2. Objective

The goal of this study is to 1) determine the GIS data on the surrounding urban context required for zoning regulations in İzmir and 2) propose a solution for integrating this data with BIM data in order to facilitate automated code compliance checking. In this context, the study aims to show that all geographical data related to a neighborhood and required for the automated code checking process can be modeled in GIS environment and this geographical data can be coupled with BIM data, constituting a complete data set for automated code checking. The study intends to pursue the process of integration between the BIM and the GIS environments by bringing BIM and GIS data to the same platform. IFC and GML data standards will be used for modeling buildings and their surroundings as they were both developed for data exchange purposes. This coupling of GIS and BIM data will help architects starting with the early design phase to achieve code compliant designs and later allow local authorities to conduct automated code compliance checking. The study aims to answer the below questions;

- What are the zoning concepts mentioned in building regulations?
- How much of the identified zoning concepts are missing in BIM?
- How can zoning data including classes, attributes and the relations required for a complete automated code checking process be modelled in GIS environment?
- How can topological relationships between zoning concepts be handled in GIS environment?
- How can data represented by IFC and data represented by GML be integrated in a unified domain model?
- How can unified data be used in a complete code checking process?

1.3. Methodology

For modeling the physical world, information from both BIM and GIS is needed and there are a significant number of areas that need BIM-GIS interoperability in Architecture, Engineering and Construction, and Facilities Management industry. To bring BIM and GIS data together, a methodology is described in this study that follows an approach based on the development of an independent third-party platform. Using an independent platform, integrating BIM and GIS data in one unified environment can take place without extending the scope of either domain to represent concepts already covered by the other. The study demonstrates that BIM and GIS data can be integrated and managed as a whole dataset in an independent environment. The first step of the developed methodology is to identify the required data in the area of study and then, to identify how much of the data can be modelled with BIM model or GIS model. Secondly, BIM and GIS domain models should be constructed by using the identified data in the previous step. Thirdly, an application should be developed that integrates the two domain models and manages them as a whole.

This research studies BIM-GIS interoperability in the area of automated compliance checking and for creating a unified model to be used in a checking process, the following stages have been conducted:

- Literature review; studying previous research for similar purposes, analyzing the approaches for solving BIM-GIS interoperability problem, addressing the importance of code checking domain in case of BIM-GIS interoperability

- Defining the problem of the study through all the information addressed through literature review
- Analyzing the current code checking processes and procedures followed in Bornova Municipality
- Analyzing Izmir Municipality Housing and Zoning Code (IMHZCode) in order to determine the data on the surroundings that is missing in BIM
- Deciding on the most appropriate approach to BIM-GIS integration
- Developing a zoning domain model for representing the zoning information
- Implementing the zoning domain model in GIS environment and modeling topological relations in GIS environment
- Developing an application for parsing and constructing BIM data in Java environment
- Developing an application for parsing and constructing GIS data in Java environment
- Developing an automated checking application using Java, that combines BIM and GIS data in a unique environment
- Testing the applicability of the model through use-case scenarios

The methodology of the study consists of seven stages:

Stage 1: The current code checking processes and procedures followed in municipalities should be understood initialy before envisioning the future of code checking processes. For analyzing the current situation, Bornova Municipality in İzmir is visited. Zoning status and construction permit procedures that are being carried out and existing geographic information systems are analyzed.

Stage 2: To define what should be represented explicitly for the purposes of automated compliance checking, an understanding and analysis of İzmir Municipality Housing and Zoning Code were of prime importance. Thus, İzmir Municipality Housing and Zoning Code is analyzed and concepts that are missing in BIM are identified. As buildings and most building parts are already represented by BIM data; concepts related to neighborhood scale information including parcel, block, road, sidewalk and surrounding buildings are considered and modelled in this study.

Stage 3: An analysis of how GIS concepts map to IFC (the BIM model) concepts is done revealing the concepts on the surroundings missing from IFC. Analysis results showed that a vast majority of building code classes with their properties should be modelled and integrated with BIM data for a complete code checking process.

Stage 4: Terms that are not covered by BIM data and identified in IMHZCode text are classified. Each term is modeled as either a class or a property. Classes are identified by analyzing if they contain members having certain attributes; and properties are identified by analyzing if they contain data related to a specific element. For example, “construction order” is a property of the “parcel” class mentioned in IMHZCode. Then the classes and their properties are brought together in an object hierarchy. A domain object model is developed using Unified Modeling Language (UML) class diagrams.

Stage 5: A GIS application is developed and the UML model is implemented in geographic information system QGIS. All the classes including block, road, parcel, sidewalk and buildings are modelled as layers. The properties of the classes are organized under layers as attribute tables. Then, topological relations between the UML diagram classes are modelled in QGIS. Building codes have rules with conditions depending on topological relations between building, parcel, block and road. Thus, modeling topological relations is crucial for a complete automated code checking process. In this stage, the relations (such as contain, touch and intersect) are defined, and queries are built in QGIS.

Stage 6: The geographical data is exported from QGIS using GML data model and parsed by a Java application. A building object with its surrounding is modelled using Graphisoft Archicad21, exported using IFC data model and then parsed by a Java application. Finally, rules in building code document are modelled and all the data sets are prepared for being used in the Java application.

Stage 7: A proof-of-concept Java application is developed for validating the developed zoning application in GIS and proving its applicability and authenticity. This validation is achieved by demonstrating a successful implementation that manages both GML data and IFC data together and conducts code checking on the integrated IFC and GML data against the modelled IMHZCode rules.

1.4. Scope

Recent years have seen an increasing number of technologies for compliance checking of building projects against zoning codes automatically. The developments in this innovative context require neighborhood scale data integrated with the building data in a common environment for a complete checking process. However, it is seen that BIM environment does not contain neighborhood data and there is an interoperability problem between BIM and GIS environments. The two distinctive domains develop their own tools lacking a common data standard. Thus, solving the data integration problem between BIM and GIS for an automated code checking process forms the main motivation for this study.

For solving the integration problem, the different approaches for interoperability in literature, which will be presented in detail in literature review part of the thesis, are studied and discussed within the scope of automated code compliance checking. The main goal of this research was not to develop a solution for every AEC field but to support automated code compliance checking processes.

As part of this research, a domain model is developed which combines both the data on the surroundings represented in GML and the building data represented in IFC. For testing its applicability, a proof of concept prototype is developed that successfully demonstrates the feasibility of the domain model as the base for exchanging information.

The protoype checks building data against İzmir Building Code, since it is representative of complex building codes in effect. İzmir is the third most populous city in Turkey and its housing and zoning code is representative of codes that are valid throughout Turkey. Hence, the effort for formalization of zoning information in İzmir Municipality Housing and Zoning Code can be adapted to regulations of other cities.

For testing the validity of the domain model and the proof of concept prototype, several use cases are organized. They are organized considering the possible behaviors of the system as a response to the potential circumstances. Considering the data on the surroundings during architectural design process is crucial as it effects both the checking results and the checking process itself. Thus, by varying the conditions within the context gradually, the correctness in checking results is tested and how the usage of different data on the surroundings affects the results of the compliance checking process are demonstrated. For this study, the use cases are limited to two-dimensional data and topological relationships of two-dimensional objects. The complex situations including the analysis of topological relations in 3D and usage of 3D GIS objects are left for future research. In addition, the proposed domain model is limited to neighborhood scale information including roads, sidewalks, blocks, parcels, buildings and their boundaries leaving the building related properties such as door, window, wall, stair out of scope.

1.5. Outline

The thesis is organized into seven chapters. First chapter provides a brief introduction of the thesis. It covers the problem statement, research objectives, methodology, and scope of this dissertation.

CHAPTER 2 contains the background information for this research. It reviews related literature about BIM and GIS interoperability and automated code checking studies. The review covers previous and current works related to this dissertation. Several models and approaches for BIM-GIS integration and compliance checking systems are examined, discussed with their advantages and limitations. The research gaps in the topic of BIM and GIS integration are identified.

CHAPTER 3 describes in detail the İzmir Municipality Housing and Zoning Code analysis and modeling approach. It begins by pointing out the current practices in municipalities including lack of zoning data stored in digital format and disadvantages of the manual code checking processes. In this chapter, the zoning information is studied, analysed and organized under building code classes, properties and relations using an UML model. Furthermore, this chapter makes a mapping between building code concepts with BIM concepts to identify the missing surrounding information in BIM.

CHAPTER 4 addresses the implementation of the developed model illustrating a GIS application and studies query developments for identification of topological relationships between domain model classes.

CHAPTER 5 contains the modeling of a project in BIM and the modeling of selected rules in the building code document to be used in the code checking prototype; as code checking requires building designs in BIM environment and rules modelled in digital environment in addition to the zoning plans in GIS environment.

CHAPTER 6 addresses the evaluation of the prototype system implementation performed for data integration between BIM and GIS for automated compliance checking. The system parses the BIM data and GIS data, combines them, and conducts automated checking on this coupled model using the modelled rules within this thesis. To test the applicability and validity of the developed prototype, use cases are organized considering the possible behaviors of the system as a response to the potential circumstances/problems. This chapter asserts that the modelled geographical data can be coupled with BIM data constituting a complete data set for an ideal automated code checking workflow.

CHAPTER 7 concludes the dissertation by providing a summary and conclusions. It also discusses the contributions and possible future research directions.

CHAPTER 2

BACKGROUND

This thesis constitutes a study on BIM-GIS interoperability, within the context of automated code compliance checking. Before proposing a method for interoperability, understanding the existing interoperability approaches and reviewing previous research on automated compliance checking systems are important. This chapter introduces the relevant background information. It is divided into two main sections: 1) Interoperability and 2) automated code compliance checking. The first section starts with a general introduction to interoperability and then focuses specifically on BIM-GIS interoperability. The second section presents an overview of previous automated code compliance checking systems and the existing platforms.

2.1. Interoperability

The term interoperability in the AEC/FM domain refers to the ability to exchange, interpret and use information with other systems in a coordinated manner. Eastman et al. (2011) defines interoperability as; "the ability to exchange data between applications, which smoothes workflows and sometimes facilitates their automation." BuildingSMART (2016), which is an international not-for-profit organization for developing open standards, defines interoperability as; "sharing of information between project team members and across the software applications that they commonly use for design, construction, procurement, maintenance and operations." Gallaher et al. (2004) in their report published by National Institute of Standards and Technology define interoperability as; "the ability to manage and communicate electronic product and project data between collaborating firms' and within individual companies' design, construction, maintenance, and business process systems."

For interoperability, the shared data should be in a common language/format to be manageable and understandable by different systems. Building Information Modeling (BIM) can complement interoperability studies with the adoption of a common model such as the IFC. With BIM, a digital building model can be constructed digitally and by using that information model, the technology of BIM makes the collaboration of multiple professions and applications possible (Eastman et al. 2011). BIM differs from Computer Aided Design (CAD) by going beyond visual representation of the building and building parts to an integrated semantic product and process model (Laakso and Kiviniemi 2012). BIM collects and manages all the project data in digital format through entire design, construction and maintenance processes in an integrated manner. It is not just a tool, but a communication and integration medium. It is also a process that contributes to the lifecycle of a facility from design to construction. The National Building Specification (NBS 2019), defines BIM as "a way of working; it is information modelling and information management in a team environment, all team members should be working to the same standards as one another."

BIM stores geometry, behavior, material property and constraints of building parts and physical and functional characteristics of a building in a single model, which can be accessed, read and edited at any time when needed by project stakeholders. Every actor can access and refine the model. Using BIM data, the entire building life-cycle process can be displayed in 3D including spaces, systems, products and design and construction sequences in relation to each other. BIM supports all stages of the lifecycle of a building and helps designers explore alternatives at the conceptual stage, make energy and performance analyses, incorporate sustainability measures, design heating, ventilation, and air conditioning systems, visualize construction process, identify material cost and quantities, make structural analyses, etc. (Jalaei and Jrade 2015).

2.1.1. BIM-GIS Interoperability

While BIM supports collaboration between all project participants during an architectural design and construction process, the integration of BIM with other systems, such as GIS, is becoming increasingly important. GIS is a system that stores, manages, queries, analyzes and presents geographically referenced spatial data. It is generally used for urban scale studies, topological, network and land use analyses, interactive spatial planning, locating the project using real world coordinates, analyzing, monitoring and managing large amounts of spatial data, working with site topography, land boundaries, and soil type etc. Even though they are developed for different purposes, BIM and GIS domains need data from each other and complement each other. They play crucial roles within the lifecycle of facilities. GIS data provides the context that is necessary to BIM data. For modeling the physical world, information from both BIM and GIS is needed. Integrating the two data models have a significant impact on solving problems in design, construction and infrastructure domains.

For example, Rafiee et al. (2014) carried out a study for determining view quality of windows by analyzing the amount of surrounding physical features appearing within the view from each window. For this study, both positions of windows from a BIM model and the type and physical features in the view such as 3D trees and 3D buildings from a GIS model are needed (Figure 2.1) (Rafiee et al. 2014). Or, in case of a fire, both fire escapes in the building and nearest safe open spaces out of the building should be kept in mind, to plan an effective evacuation plan. For evaluating the noise level for defined areas, traffic noise from the outdoor environment and sound absorption and material type of building elements within the buildings should be considered together. An urban infrastructure project requires both models from the building domain and information from the urban planning domain. For designing energy-efficient buildings in neighbourhood scale, the building envelope, interior layout, and the existing energy systems of surrounding buildings should be taken into account. Thus, there are a significant number of areas that need BIM-GIS interoperability such as real estate and property management, construction, utilities maintenance operations and infrastructure management, site selection, site circulation/parking/vehicle routing, security planning, evacuation routing, coverage and shadow analysis, height analysis, cost estimation, response analysis, and automated compliance checking.

Figure 2.1. Combining BIM and GIS data.
(Source: Rafiee et al. 2014)

While BIM and GIS technologies are closely related, they remain largely isolated. Distinctive development of BIM and GIS tools, and professional divisions were the primary causes for the differences between BIM and GIS that are listed by Akin (2010) as:

- BIM focuses on architectural design and construction processes, and facility operations and management; while GIS focuses on mapping, spatial analysis, data management, and geoprocessing.
- BIM is mostly used in AEC/FM domain; while GIS is used in areas of business (site location, delivery systems, marketing), government (local, state, federal, military), emergency services (fire and police), health (hospitals, health policy and research), economic development, politics (elections and reappointment), communication, transportation, mining, and urban planning (land use, housing studies, crime analysis).
- BIM has highly standardized structures; while GIS has user defined structures (Przybyla et al. 2010).
- BIM uses mathematical models for form creation, such as circles, arcs, and parallel lines. GIS uses only points, lines and polygons for storing and analyzing spatial data.
- BIM is based on vector data; while GIS is based on both vector data and raster data.
- BIM uses a rectangular Cartesian coordinate system which uses a set of axes for locating the object with respect to an arbitrary origin; while GIS uses Geographic Coordinate System which uses longitude and latitude for positioning the object in a physical world horizontally and vertically with reference to the center of Earth (Ghafourian and Karimi 2010). In Geographic Coordinate System, features are geographically referenced to the world.
- BIM visualizes single drawings great detail in larger scales; while GIS visualizes multiple sites with less detail in smaller scales (Irizarry and Karan 2012).
- BIM is used for modeling objects that do not exist prior to the generation of the model, GIS is used for modeling the existing geographical features.

Although BIM and GIS differ from each other, both can benefit from each other if they could exchange data effectively. For overcoming the differences between BIM and GIS and the interoperability problem, four main interoperability approaches are proposed in literature:

- integrating BIM data into GIS environment or vice versa
- developing extensions for data models
- developing transfer/conversion mechanisms
- developing independent domain models

For relevant data exchange and information sharing between BIM and GIS software applications, common data formats should be developed. Data transfer/ conversion and usage of a common data format is a critical step towards interoperability of BIM and GIS domains. Hence, the next section introduces the related background information on data standards and data converters.

2.1.2. Data Standards

Early studies that accomplished data exchange included only geometric data, failing to transfer non-graphical attribute data. AutoCAD DXF developed by Autodesk was one of the file formats used for geometry exchange purposes. During 1980's, data exchange studies incorporated object-oriented modeling concepts, capturing and exchanging semantic information. STandard for the Exchange of Product data model (STEP) is an example, which enables users to exchange semantic information in addition to geometric data and topology information, using object-oriented modeling concepts. Semantic information is the information of object with its attributes and relationships to other objects. For example, a wall with its semantic information (Figure 2.2) represents the material, the dimensions, the geometry, the thickness and location of the wall and its relation to other building elements such as walls, ceiling, floor, windows, doors etc. (Kiziltas et al. 2010; Eastman et al. 2011).

Semantics specify what each concept means and its relationship to other concepts. It provides an objective specification of domain information by representing a consensual agreement on the concepts and relations. By semantics, an object can be defined only once and reused numerous times. Therefore, it enables heterogeneous systems within the city or between the cities to seamlessly interoperate (El-Diraby and Osman 2010). Briefly, semantic information is objects with meaning.

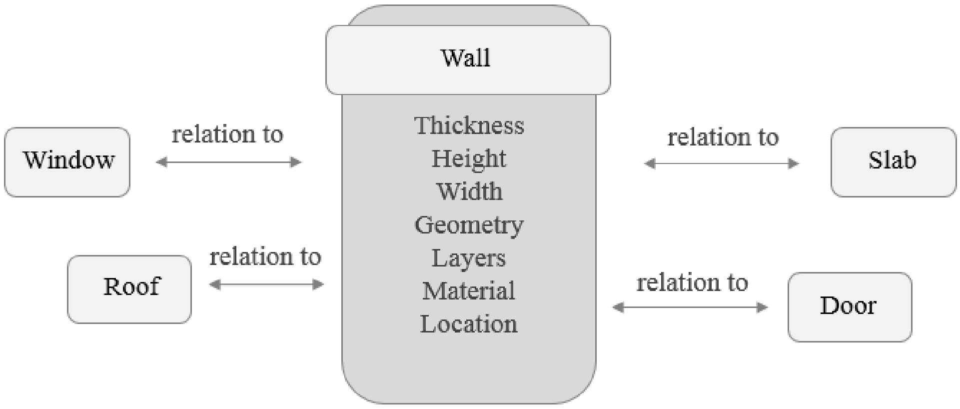


Figure 2.2. Conceptual structure of wall object.

For BIM-GIS interoperability to work, many software developers have to agree on common semantics, interfaces, information models, guidelines and schemas (Figure 2.3). Such agreements that are supported by community are typically documented as standards. The Open Geospatial Consortium (OGC) is an international non-profit organization that develops standards for supporting the industry consensus development in the area of geoprocessing and related information technologies. The standards that OGC develops are freely available for anyone to use (Opengeospatial 2017). OGC collaborates with many standard developers. For BIM-GIS interoperability, OGC collaborates with the National Institute of Building Sciences (NIBS) and the buildingSMART alliance. The close coordination, communication and collaboration of many organizations and developers strengthen standards-based interoperability between the geospatial and building communities. For example, upon approval by the OGC membership, 3D Information Management (3DIM) working group, previously the CAD-GIS Working Group was formed in 2005. The group was formed to improve interoperability of geospatial content and services across the AEC/FM, 3D, and GIS domains (Reed 2010). The OGC Land And Infrastructure Domain Working Group (LandInfraDWG) was formed in 2013. The group works closely with buildingSMART for integrating their proposed data model with BIM. The InfraGML data model includes information related to infrastructure facilities, road, railway, survey, drainage, water distribution systems, land features and land division (Liu et al. 2017; OGC 2019b). Another working group is IndoorGML Standard Working Group that proposed the IndoorGML data model. The data model will be an OGC GML application schema and aims to become a complementary standard to CityGML and IFC to support location based services for indoor navigation (Liu et al. 2017; OGC 2019a).

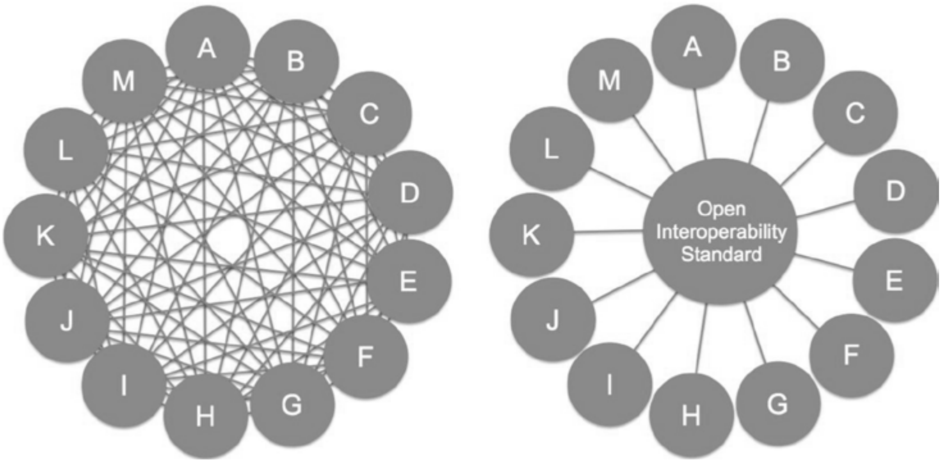


Figure 2.3. Interoperability: direct translators vs. an open interoperability standard. (Source: Laakso and Kiviniemi 2012)

The OGC developed many standards relevant to BIM-GIS integration including: OGC and ISO Web Map Service (WMS) Interface Standard; OGC Web Map Context Standard (WMC); OGC Web Feature Service (WFS) Interface Standard; OGC and ISO Geography Markup Language (GML) Encoding Standard; OGC CityGML Encoding Application Schema; and OGC KML 2.2 Encoding Standard. GML and CityGML that is an application schema of the GML are the widely used data models for interoperability comparing to others. They define classes, spatial properties and relations of urban objects such as buildings, water bodies, vegetation, transportation facilities and city furniture with their geometry, topology, semantics, and visualization properties (Reed 2010). Urban and landscape planning, architectural design, environmental simulations, disaster simulations, facility management and site surveying are some of the areas where GML and CityGML are used (Kiziltas et al. 2010).

buildingSMART is another international organization like the OGC that develops standards for supporting the industry consensus development. buildingSMART focuses on standardizing processes, workflows and procedures for BIM and has a worldwide network. IFC, BIM Collaboration Format XML, BIM Collaboration Format API, IFD: Framework for object oriented information, IFC4 Design Transfer View, IFC4 Reference

View are the common data models that buildingSMART developed for transporting information across the AEC/FM, 3D, and GIS domains (buildingSMART 2016b).

Industry Foundation Classes (IFC) is the widely accepted open and non-proprietary standard developed by buildingSMART association. It serves to the interoperability needs of various software and focuses on product and process modeling (Inyim, Rivera, and Zhu 2014). IFC represents a sharable data schema ensuring the same semantic understanding between domains and aims to define a common language to enhance the collaboration, efficiency, productivity, delivery time, cost, and quality both for AEC/FM domain, and cross domain and cross-disciplines (Isikdag, Underwood, and Aouad 2008; Kiziltas et al. 2010).

IFC is an object-oriented data model based on class definitions representing objects. It includes physical object types such as walls, floors, ceilings, doors, and furniture, and non-physical object types such as schedules, resources, activities, and organization and construction costs. Both 2D and 3D geometry of objects and semantic information including properties of objects and their relationships are the content of IFC (buildingSMART 2016b; Isikdag, Underwood, and Aouad 2008; Hijazi et al. 2009). IFC is currently the emerging standard in the industry. By its semantic information storage capability and ability for interoperability; IFC serves as a benchmark and testbed for new models (Akin 2010).

buildingSMART has a 'GIS/BIM Information Exchange' project that is under-development for providing the basis for information exchange between BIM and GIS and their underlying databases. The project is focusing on interactions between BIM and GIS that might occur during the lifecycle of a project (National Institute of Building Sciences 2016). Since 2010, the meetings for developing the project have been continuing with the aim of providing a site/surroundings dataset from GIS to BIM environment at the early stages of design, and providing building design data from a designer to GIS environment at building completion stage (Przybyla et al. 2010). The committee acknowledged the need for one or more BIM-GIS standards (National Institute of Building Sciences 2016).

Initiated by the Norwegian State Planning Authority, Industry Foundation Classes for GIS (IFG) was another effort developed for exchanging information between BIM and GIS-based applications. Rather than integrating complete GIS into the IFC data model, the aim of IFG was to use existing entities in IFC and map the information to GML. Finally, a mapping specification was created that could be used for data transformation from IFC to GML (Khosrow-Pour 2014). IFG enables exchange of building elements such as wall, window and door, maps, coordinate systems, networks, distribution systems (water, sewer, and power), terrain data and semantic definition of a building. Response management, disaster management, and utility services management are some of the areas where IFG can be used (Kiziltas et al. 2010).

2.1.3. Data Converters

Due to the fundamental differences in the origins and methodologies of BIM and GIS, some researchers believe that attaining a common data model is naturally difficult and inappropriate. These researchers mostly adopt an approach that makes use of data converters, which translate one data format into another data format. Most commercial BIM and GIS software have data translation tools and extensions to allow direct access to other formats (Casey and Vankadara 2010). For example, ESRI ArcGIS's Data Interoperability extension allows converting between formats using import/export functions within the system. The interoperability extension enables ArcGIS users to use and distribute data in various formats including Autodesk 3ds, Autodesk AutoCAD DWG/DXF, Bentley MicroStation Design V7/V8, CityGML and IFC (Esri 2015). However, the converted data is mostly read only and limited in function. In addition, not all GIS objects have definitions in IFC and vice versa. For example, an object defined as sidewalk in GIS, will not be able to find a response in BIM environment after a data conversion process, and that will end in data loss or mapping to a wrong object.

2.1.4. Recent Efforts on BIM-GIS Interoperability

Recent studies on BIM-GIS interoperability commonly focus on developing new ways of applying BIM data to GIS environment and vice versa, developing new extensions to data models such as CityGML and IFC, developing new methodologies for making data conversion and translation between data models, and developing independent domain models.

Considering applying BIM data to GIS environment and vice versa; Isikdag, Underwood, and Aouad (2008) investigated the applicability of BIM in the geospatial environment by focusing specifically on two domains; site selection and fire response management. Hijazi et al. (2009) studied integration of the 3D BIM data into GIS to enable interoperability between IFC and CityGML for utility network management. Rafiee et al. (2014) proposed a method to integrate an IFC BIM model of a building into a GIS environment for effective and efficient spatial analysis. Elbeltagi and Dawood (2011) integrated a BIM model into Google Earth and tried to link their building model with a geographic information system to monitor and visualize construction performance of repetitive projects in relation to time control. Irizarry and Karan (2012) studied interoperability of BIM and GIS for identifying optimal solutions for selecting, and locating tower cranes on construction sites. They implemented BIM data in geospatial environment and then exported the analysis output of the GIS model to the BIM environment aiming to use the strengths of BIM and GIS environments in the context of the other. In the approach of Irizarry, Karan, and Jalaei (2013), detailed information about the materials modelled in BIM was inputted into GIS using a plugin interface embedded in the BIM software for improving the visual monitoring of construction supply chain management. Shi and Liu (2014) proposed a platform that transfers extracted IFC data from BIM environment into the 3D GIS environment to support evacuation planning in case of a fire. Different from the studies mentioned above, Park et al. (2014) proposed a system that extracts the GIS data and integrates it into the CAD system for determining the probable costs of a national road structure.

Data model extension studies add concepts that do not already exist in the IFC model and thus, expand the scope of the model by integrating various domain data. For example; Borrmann et al. (2015) studied the extension of the BIM data model IFC for modeling multi-scale representations of shield tunnels. Then, they studied mapping the building model to the geospatial model CityGML and its use in the context of geo-analysis for the planning of large infrastructure facilities. Karan and Irizarry (2015) studied the extension of the BIM authoring tools to preconstruction operations using geospatial analyses and semantic web services. Zhiliang et al. (2011) studied extension of the IFC data model for integrating construction related data. At the end of the study, the IFC standard became able to be used for expressing the information for the construction cost estimating for tendering in China. Lee and Kim (2011) extended IFC schema for road structures. As definitions for IFC specification are not enough to represent bridge and tunnel properties in detail; new entities and properties were defined including 'IfcRoadElement,' 'IfcRelCivilSpatialElementBoundary,' and 'IfcGround-ReinforcingElement' (S.-H. Lee and Kim 2011). Malsane et al. (2015) studied extension of IFC for integrating England and Wales Building Regulations that relate to fire safety for dwelling houses. With automated code checking processes, building models will be able to be checked by computational rules in digital environment such as GIS against fire safety (Malsane et al. 2015).

There are also data model extension studies for geographical data involving adding building related data that do not already exist and thus, expand the scope of the geographical model. In their study, de Laat and van Berlo (2011), developed an extension to CityGML for converting building data into GIS data and thus enriched CityGML with semantics from IFC data.

IFC and CityGML are the most comprehensive and representative data models in BIM and GIS domains, respectively. Thus, many studies aimed to convert and translate them to be compatible with each other. Wu and Hsieh (2007) proposed an approach to transform IFC model objects to the GML model objects for using building information data in GIS environment. They then developed IFC2GML tool to demonstrate the applicability of their proposed approach. In the study of Sebastian, Böhms, and Helm (2013), the CityGML standard was extended using an Application Domain Extension (ADE) for being able to transform the IFC model into a CityGML model. Amirebrahimi et al. (2015) proposed a method that firstly converted the exported IFC data from BIM to ESRI geodatabase feature classes and then to CityGML using the ArcGIS Interoperability Extension and integrated with geographical data in GIS environment to support detailed analysis and 3D visualization of the potential damages of flood to buildings. T. Kang (2018) proposed a conceptual mapping standard ISO N19166 and a mechanism for object information coming from IFC and CityGML data models. The mechanism defines mapping rules for transforming a BIM object into a GIS object and tries to represent BIM objects in GIS. Donkers et al. (2016) presented an automatic conversion algorithm that converts building models stored in IFC format into CityGML format. They make the conversion by applying geometric operations and semantic mapping of the attributes to keep the meaning of each attribute. Isikdag and Zlatanova (2009) proposed another semantic mapping approach between BIM and GIS environments for information transfer between IFC and CityGML models. They focused on unidirectional information transformation from BIM to the CityGML models. El-Mekawy, Östman, and Hijazi (2012b) evaluated the approaches in literature for unidirectional transformation between IFC and CityGML data and discussed the pros and cons of such an approach. They concluded that because of the semantic differences between IFC and CityGML data models, unidirectional approaches are not successful for translation of all the needed concepts. In addition to unidirectional transformation studies, bidirectional transformation of information was also studied by researchers such as in the study of Ohori et al. (2017). They started a research project called GeoBIM in the Netherlands that offers guidelines for bidirectional transformation of BIM and GIS data and reuses BIM data in the GIS domain and vice versa.

Even though many researchers have studied how to integrate BIM and GIS data and how to address all the differences, many challenges are still being encountered and it is still very hard to integrate data throughout the design and construction processes. Data access and conversion/translation problems between BIM and GIS cause data loss, incomplete and unreliable transformations of information, incorrect mapping, and manual data re-entry. IFC and GML data standards are incompatible as they are different in terms of representation, scale and aim. They are developed for different professionals and processes. For example, IFC data models contain much more detailed information from various disciplines than GML. Some concepts in GML have no corresponding entity definitions in IFC data model and vice versa. Thus, data loss from IFC to GML is inevitable.

Data extension studies are also problematic since, the aim of BIM is to support lifecycle of a building from design to construction and the aim of GIS is to store, analyze and manage geographically referenced spatial data for urban scale studies. Thus, it is impractical to expect BIM to explicitly define geographical information; and GIS to manage detailed building data, and extend information storage capability of the two environments except their scope.

Currently, there are various interpretations of how to best transform a BIM file into GIS data and there is no accepted standard. One approach that is proving to be effective for many fields, is developing independent applications that draw data from both BIM and GIS environments and combining them in a unified domain model appropriate to the field (Hor, Jadidi, and Sohn 2016; Gilbert et al. 2018; Knoth et al. 2018). The domain model manages both the surrounding data represented in the GML and the building data represented in the IFC and then can be the base for exchanging information. This approach does not require BIM or GIS systems to extend their data models but depends on new domain-specific models linking the two to be defined. In their study, Song et al. (2017) also propose creating a third-party platform as one of the future trends and opportunities of BIM-GIS integration, as this approach does not require to modify or convert any of the concepts, methods, systems and theories of BIM and GIS.

As an example, Hor, Jadidi, and Sohn (2016) studied data exchange between BIM and GIS domains and asserts that the syntactic and even the semantic approaches do not provide a full data sharing. Therefore, they propose a novel interoperability approach using semantic web technologies and Resource Description Framework (RDF) graphs for representing BIM and GIS objects, properties and relationships. The novelty of the proposed approach comes from the integration of BIM and GIS data into one unified model that is called Integrated Geospatial Information Model (IGIM). IGIM consists of three modules; BIM and GIS RDF graph construction, a semantically integrated model construction that contains all the classes, and properties from both BIM and GIS domains and query of information using SPARQL language. Gilbert et al. (2018) developed a scale-free and multi-format integrated system for infrastructure planning that includes modeling of buildings, the urban environment and infrastructure networks. The system uses a graph database to drive, integrate and query the utility network data represented in seperate models: the CityGML UtilityNetwork Application Domain Extension (ADE) and IFC. The data in disparate data models are integrated and formed a single network graph model without relying on the existing schemas. Knoth et al. (2018) proposed a model for interoperability where the common building elements amongst various data models are extracted and then combined into a common core model. The common elements are the most basic elements that a building should have for storing digital building information such as storey, wall, space etc. As the building is at the overlap of two disciplines: architecture and spatial planning, the model can be exchanged easily between BIM and GIS and has a use-case independent character. Kim et al. (2015) proposed a semantic data integration approach for BIM and GIS to perform spatial data analysis needed in earthwork calculations. The system includes infrastructure data extraction from BIM and geographic information extraction from GIS separately and their integration in the same platform. El-Mekawy, Östman, and Hijazi (2012a) developed a unified building model that collects data from both BIM and GIS and integrates them by mapping of objects in IFC and CityGML standards. Thus, all classes of IFC and CityGML data with their concepts were brought into a single unified model, overlapping concepts were merged and new objects were created to perform required spatial analyses.

Figure 2.4 shows diagrams of different interoperability approaches for integrating BIM and GIS data. In this thesis, an independent application is developed that draws data from both BIM and GIS environments and combines them in one unified domain model appropriate to the field. The merging of BIM and GIS creates a powerful tool, which is then applied to an automated compliance checking process in this thesis.

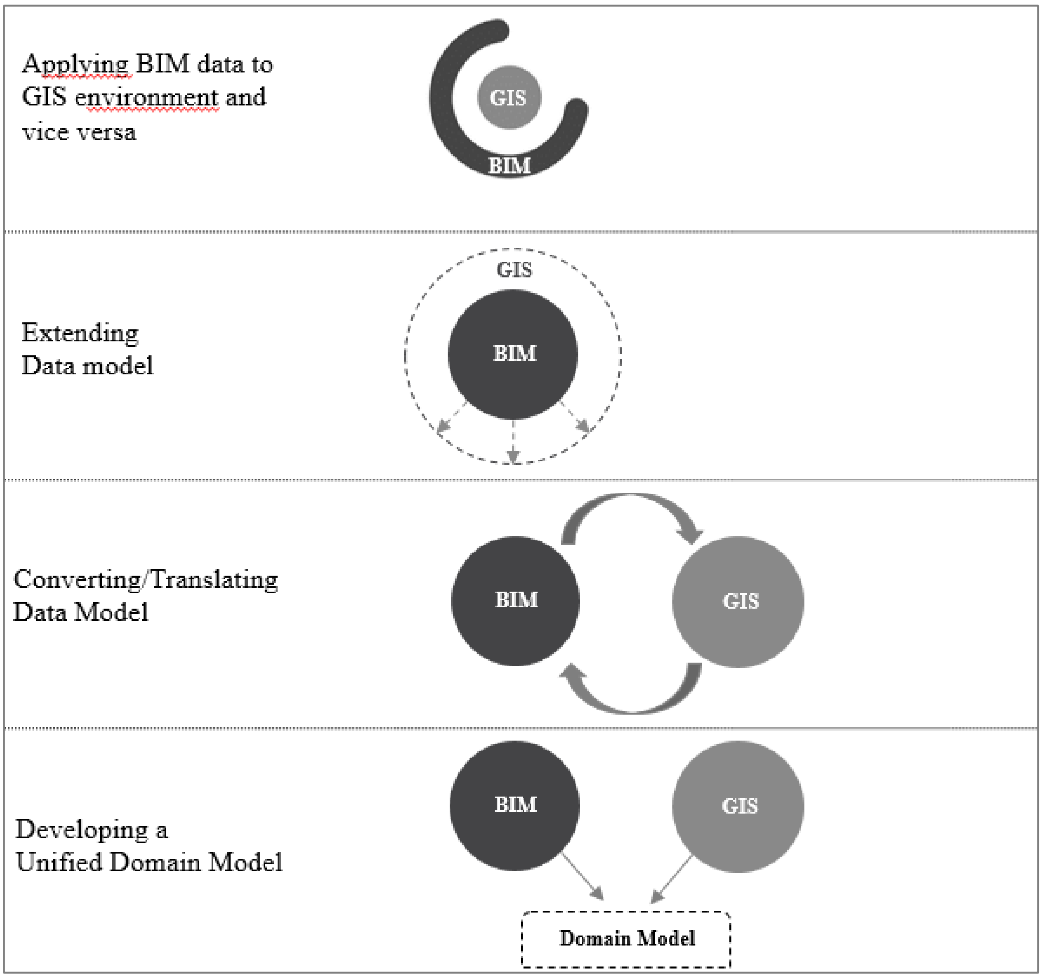


Figure 2.4. Interoperability approaches for integrating BIM and GIS data.

2.2. Automated Code Compliance Checking

Automated compliance checking needs BIM-GIS interoperability. It requires building codes, zoning information and building design in a digital interoperable environment for an ideal automated code checking process.

Automated compliance checking has long been an area of research that aims to provide computational support for compliance checking of building projects against national building codes developed by government agencies. Building code checking applies rules and conditions to a design, with results such as “pass”, “fail” or “unknown“

for cases where data is incomplete or missing (Eastman et al. 2009). If the results indicate that the model or the building components do not comply with regulations, they can be revised based on the results and checked again. Final result must comply with the regulations in the building code (Van Berlo, Dijkmans, and Stoter 2013).

Automating the checking process has attracted many researchers who were interested in how to represent codes written in natural languages in digital form. Garrett and Hakim (1992) and de Waard (1992) developed an object-oriented model of building codes. Yabuki and Law (1993) combined first order predicate logic with object-oriented modeling approach to building code processing. Kiliccote and Garrett Jr (1998) developed a context-oriented approach to represent building codes. However, these early efforts adopted a hard-coding approach, which required specialized programming knowledge and expertise for modeling the building codes. To overcome the challenges of hard-coding representation, researchers started to focus on semantic modeling approach for knowledge representation. The SMARTcodes project was an important project adopting a semantic approach for automating the generation of rules into a computer interpretable format (Conover 2007). Macit İlal and Günaydın (2017) developed a four level representation for building codes based on the same semantic modeling approach. A model for representing building regulations from Local Urban Planning Schema of France has been developed by Brasebin et al. (2018), Brasebin, Perret, and Haëck (2011) and Brasebin et al. (2016). They modeled the geographic elements in the regulations including urban zones, parcels, roads, buildings and then formalized the rules based on the Object Constraint Language. H. Lee et al. (2016) described an approach for translating the natural language text of the Korean Building Act into a computer-executable format for evaluating building permit requirements. In their rule based semantic approach, Beach et al. (2015) formulate a methodology that allows domain experts to create and maintain their own automated regulation checking systems. The system is validated by using regulations from construction domain. Recently, ontology-based approaches are adopted by researchers for interpreting and representing building code (Pauwels et al. 2011). Yurchyshyna et al. (2008) studied an ontology based approach and Dimyadi et al. (2015) developed a regulatory knowledge model for the formalization of building codes.

Over the last three decades, while some researchers were looking for the most appropriate digital representation of building codes, other researchers started work on building automated checking systems using whatever representation was available. The Singapore CORENET project is the first automated code compliance checking system that was used in AEC industry. Developed by the Singapore Building and Construction Authority, the project aimed to provide a web based electronic system to do code checking over submitted building plans (Liebich et al. 2002). The rules for checking were related to building control, fire code, environmental health, public housing and vehicle parking. Norwegian efforts for rule checking included checking BIM projects for evaluating spatial program requirements, and building accessibility (Lindberg 2006). The majority of the rules deal with the relationships between spaces, doors, ramps, stairs and windows. Later, the Cooperative Research Center for Construction Innovation in Australia funded a research for automated building code checking. This was the Design Check project aimed at automated checking of Australia's requirements related to disabled accessibility (Eastman et al. 2009). SMARTcodes project led by the International Code Council (ICC) focused largely on mapping the paper based building codes into computer interpretable code sets and, automate online code-compliance checking of building projects (Conover 2007). General Services Administration (GSA) in United States funded development of a rule checking system in the areas of circulation and security validation of buildings (Eastman et al. 2009). SEUMTER is a Korean electronic system that enables the automation of building code compliance checking for fire prevention (Choi, Choi, and Kim 2014). Malsane et al. (2015) developed a building regulation-specific object model for automated compliance checking according to England and Wales Building Regulations. They focused on fire safety for dwelling houses. Choi, Choi, and Kim (2014) developed an automated BIM-based system for checking high-rise and complex buildings against the evacuation regulation compliance. Yang and Xu (2004) studied the development of an online code checking process. In the study of Dimyadi et al. (2014), a two-part regulatory knowledge representation was studied for performance-based design of buildings related to fire safety.

Recently, automated or semi-automated transcriptions employing artificial intelligence techniques such as semantic modeling have been studied (J. Zhang and El-Gohary 2012). Ontology based building regulation representation on the semantic web is another research area as Pauwels et al. (2011) studied compliance checking considering acoustics on a BIM model. Wicaksono, Rogalski, and Kusnady (2010) studied compliance checking considering energy management system for buildings. Baumgärtel, Kadolsky, and Scherer (2015) studied building energy performance checking. S. Zhang, Boukamp, and Teizer (2015) used semantic web technologies for analysis of construction projects considering jobs, tasks, safety procedures etc.

The main topics these studies focus on are setback distance calculations (Yang and Xu 2004), accessibility (Ding et al. 2006; Lindberg 2006), fire safety (J. S. Kang 2008; S. H. Lee 2011; Malsane et al. 2015; Liebich et al. 2002), evacuation planning (Choi, Choi, and Kim 2014), environmental health, public housing and vehicle parking (Liebich et al. 2002), energy conservation (Conover 2007), circulation and security (Eastman et al. 2009). These studies represent a significant progress in this research area, yet they mostly deal with regulations of specific disciplines and focused on checking individual building designs isolated from their context. They do not provide a generic solution, which would be applicable to all types of clauses. Since the legal systems and the building regulations between countries vary, the developed solutions are not comprehensive enough to include the concepts mentioned in Building Code Document of Turkey. In addition, the present BIM tools used in these code checking processes are expected to hold regulation specific data. However, the BIM tools are not capable of containing all the required building code data for a fully automated code checking (Malsane 2015).

2.2.1. Recent Efforts on BIM-GIS Interoperability in the Area of Automated Code Compliance Checking

Recent studies on BIM-GIS interoperability in the area of automated compliance checking mostly focused on data import from BIM to geo-information domain. Olsson et al. (2018) proposed integration of BIM and GIS data for automating building permission process and checking building model against Swedish building permission regulations. They focused on the building height and the building footprint area criteria in detail to which extent they can be automatically checked. Their approach included importing BIM model into a digital city model.

Some studies focused on data transformation such as the study of Benner, Geiger, and Häfele (2010). They studied building information and geo information integration for automated checking during building licensing processes by transforming BIM data in IFC format into a CityGML city model where the rule checking is performed.

Several researches included both data transformation and data integration within BIM environment. Van Berlo, Dijkmans, and Stoter (2013) generated a BIM model and enabled architects to use geo information in early design phases and automate checking process against building regulations. They converted geo information objects as IFC, import these into BIM and conduct the rule checking process in a BIM environment. They focused on checking maximum allowed building volume, the maximum percentage of built-up area on the site and maximum allowed noise impact on each facade.

Other studies proposed development of domain extensions to IFC model. Malsane (2015) and Malsane et al. (2015) developed an IFC-compliant England and Wales building regulation specific object model focusing on fire safety and extended the building model as current IFC files are not rich enough for use in the automated compliance checking process. Salama and El-Gohary (2011) incorporated construction process information into BIM model to check a model's compliance with regulations. However, the number of studies that study BIM-GIS interoperability in the area of automated compliance checking is limited. In these studies, limited types of information were modeled from specific disciplines, and limited number of regulations were checked from specific areas, which did not cover building code data extensively required for a complete code checking process.

This thesis focuses on the integration of BIM and GIS data for supporting automated code compliance checking processes. Similar to other studies it will be limited to a specific code set, namely housing and zoning codes. However, in contrast to the existing studies in literature that apply BIM data to GIS environment and vice versa, develop new extensions to data models and develop new methodologies for making data conversion and translation between data models, this thesis proposes adoption of an approach to integration where code checking is not conducted in either the BIM or GIS environments, does not require the extension of existing models, but instead, is carried out by an independent application that draws data from BIM and GIS systems and constructs its own unified domain model. In this approach, the main goal of this independent, third-party domain model is to facilitate domain specific tasks (code checking processes in the context of this thesis). Yet, as a formal data schema, it will at the same time serve to inform GIS and BIM based applications as to the domain of interest's representational requirements. As an example, the unified model that includes neighborhood scale data developed as part of this thesis work can be utilized in implementing third-party add-ons that allow access to site related information at city scale for BIM based building design applications. Currently, such information on the immediate surroundings of buildings is still not available to designers.

CHAPTER 3

ZONING DATA ANALYSIS AND MODELING

Before proposing a solution for integrating the geographical information with BIM data in order to facilitate automated code compliance checking, the current code checking processes and procedures should be understood and analysed, and the required geographical information should be identified. Identifying the geographical information embedded in rule statements that are missing in BIM requires an understanding and analysis of İzmir Municipality Housing and Zoning Code. This chapter presents the work carried out during the following stages:

- Stage 1: Analyzing the current building code checking procedures followed in Bornova Municipality.
- Stage 2: Analysis of the building code document to define the missing concepts in BIM that should be represented explicitly for the purposes of automated compliance checking.
- Stage 3: Mapping GIS concepts to BIM concepts and determining the amount of missing zoning knowledge in BIM.
- Stage 4: Organization and representation of the building code concepts and relations and building the domain object model.

3.1. Identifying Current/Existing Practices

In order to envision the future of code checking processes, current situation has to be analyzed, initially. For observing how zoning status and construction permit procedures are being carried out in municipalities, examining existing geographic information systems, and understanding code checking processes; Bornova Municipality in İzmir was visited from 31 January 2017 to 10 February 2017 for two weeks. Bornova Municipality is selected since Bornova Municipality is known for its commitment to digital technologies and its effective use of information and communication channels. During the visit, nonparticipant observations were made in the Department of Construction Permits and the Department of Zoning. Notes were taken about ongoing procedures and geographic information systems used. During the first week, observations were made in the Department of Construction Permits and during the second week, observations were made in the Department of Zoning.

3.1.1. Zoning Status and Construction Permit Procedures

Department of Zoning is the first department that property owner consults for getting zoning status information of the property. Department of Construction Permits is the last department where the code compliance checking is done.

3.1.1.1. Department of Zoning

The procedure of obtaining a construction permit starts with a consultation at the municipality. Property owner first consults the Department of Zoning and takes the zoning status information about the property. Department of Zoning gives two documents to the owner, which are; a copy of the development plan /site plan and the zoning status (Appendix A). Development plan is a document that includes information for addressing the social and cultural needs of the people and improving the quality of life. The purpose of the document is to create a healthy and safe environment and protect general welfare. The document is prepared with a research based on the social, cultural, economic, demographic, historical and physical characteristics of the town and determines urban settlement and development trends by creating alternative solutions. It consists of layouts, reports, and notes that contains the principles of land use, protection, restriction, organization and application. Development plans are organized as master plan and implementation plan (Yıldız 2012).

Master plan is a plan document that includes general aspects of land use, main types of regions, future population densities of the regions, building density, development direction and size of various settlement areas and the main transportation network. It shows the main texture of the city, indicates general principles and does not determine exact boundary and shapes. Administrative boundaries, main transportation networks, areas of religion, trade, art, industry, green areas, infrastructure facilities such as high voltage networks, water networks are some of the information that a master plan contains (Yıldız 2012).

Implementation plan is a plan document that includes blocks of regions, their density and organization, roads, implementation stages and principles and other information in detail. In the implementation plans (Figure 3.1), blocks, construction order in these blocks, the heights of the buildings, the roads and their widths are specified. Types of usage (housing, trade, industry, education, health, religious, green areas), buildings (construction order, number of floors or height), infrastructure (electricity network, substation, water network, sewer network) are some of the information that an implementation plan contains (Yıldız 2012).

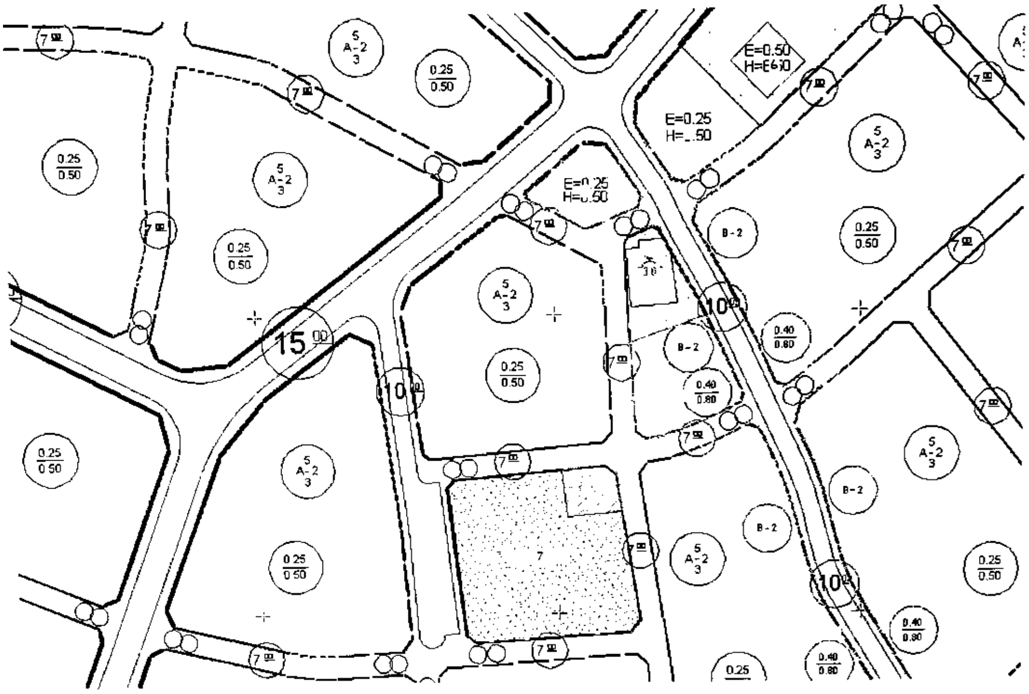


Figure 3.1. Implementation Plan Example.
(Source: Yıldız 2012)

Development plans are crucial as they are used for planning and organizing parcels within the blocks. However, a development plan cannot show all of the details aimed to be shown and to indicate all the decisions taken. It is not possible to transfer some information to the plan drawing. Special conditions and restrictions that are relevant for the specific property plan notes are written on the development plan and these are called “plan notes.” Plan notes include information about construction conditions that are prevalent only for the neighborhood where the property is located. These plan notes do not exist digitally. Instead they are written documents and these texts are written on the hardcopy of the Development Plan of Bornova as shown in Figure 3.2 (Izmimod 1997).

The other document that is given to the property owner is the zoning status document. Zoning status document also gives information about the property, however; it contains more detailed information according to the development plan including construction order, maximum number of storeys, maximum height of building, maximum depth of building and setback distances.

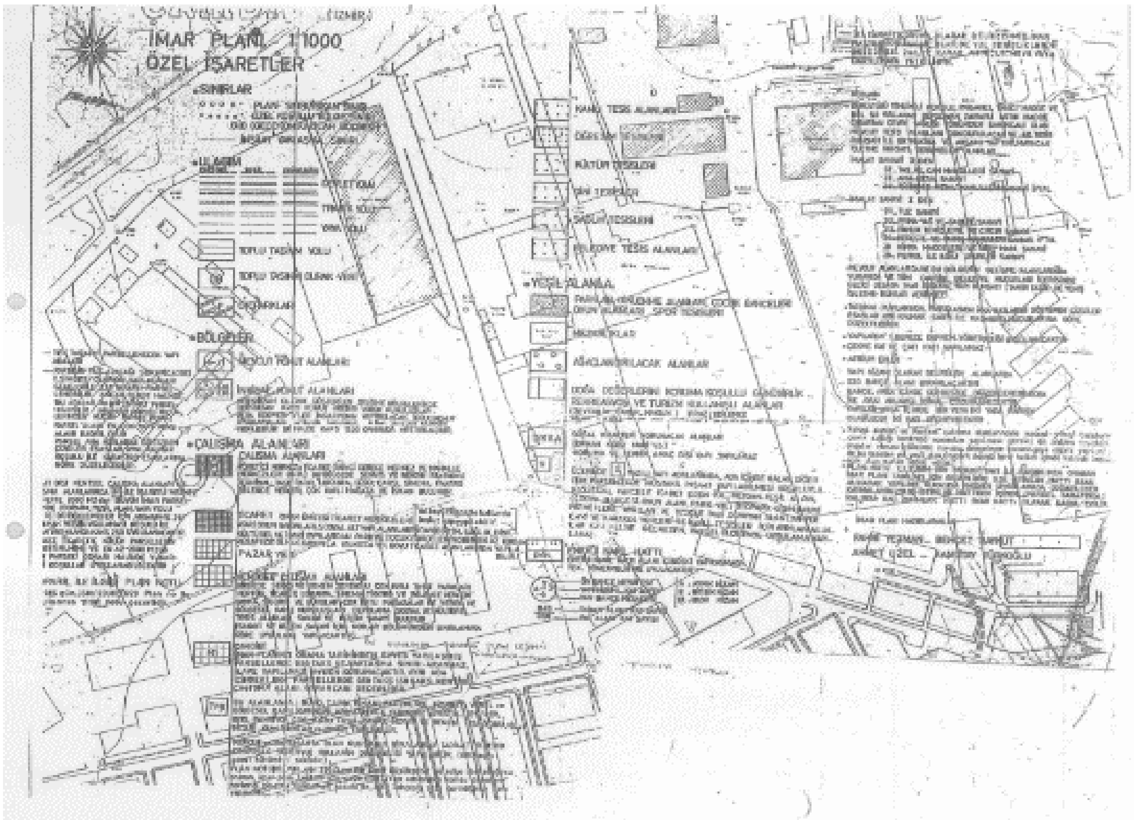


Figure 3.2. Plan Notes on the Development Plan of Bornova.
(Source: Izmimod 1997)

In Bornova Municipality, the development plan and zoning status documents are prepared using Netcad software and given to property owners as printouts. For producing these documents, plan drawings of Bornova in Netcad format and scanned development plan of Bornova in .tiff format are superimposed in Netcad. Plan of Bornova includes the drawing of existing buildings while the development plan includes construction order, number of storeys, maximum height of building, maximum depth of building and setback distances and surrounding information including roads, neighboring parcels and green areas. Neighborhood information also concerns the property owner because if the building order is determined as a row-building order in the selected parcel, placement of the buildings on neighboring parcels becomes important for determining the location of the new building. In some cases, there can be missing information related to location of the buildings on neighboring parcels on plan drawing of Bornova. In these situations, archives related to surrounding parcels are asked for from the archive section in the municipality and checked for whether there is an existing building or not in surrounding parcels. Thus, generally zoning knowledge is not recorded digitally, rather found on printouts and read from the scanned images of the development plan.

Different from Bornova Municipality, some municipalities including Bayraklı Muncipality has an online system. By entering neighborhood name, street number, door number or parcel number; the system displays an informative online zoning status document without a need for going to the municipality. However, property owner still needs to go to the zoning department of the municipality for more detailed and updated information.

3.1.1.2. Department of Construction Permits

After required zoning documents are gathered; architectural, structural, mechanical and electrical drawings are prepared and submitted to the construction permit department for approval. These projects are controlled considering the constraints mentioned on the development plan and zoning status documents, plan notes on development plan and building regulations in force as shown in Figure 3.3. Considering construction and organization on a parcel, development plan and plan notes are the initial documents to be used for checking the architectural project. For the missing details, Izmir Metropolitan Municipality Zoning Regulation is the prevailing document. If there are still missing details, Ministry of Environment and Urbanization's Zoning Regulation for Planned Areas is valid. In addition, in some cases parcel or building information can be defined differently in the development plan and in the Metropolitan Municipality Zoning Regulation. In those cases, the development plan has priority. For example, if the given building height on development plan is different from the value in the Metropolitan Municipality Zoning Regulation, the given height in the development plan is valid (Yıldız 2012).

For the code compliance checking process, the owner submits architectural projects as printouts as well as digitally on a CD. Project printouts are checked against building codes manually. The building regulations are written as texts and currently, there are three valid regulations, which are: İzmir Metropolitan Municipality Regulation, Ministry of Environment and Urbanization's Planned Type Zoning Regulation 08.09.2013 and Planned Type Zoning Regulation 01.06.2013. Personnel carrying out the manual checking memorizes - in-time - most of the regulations. Still, regulation documents are consulted frequently for verification and various details in codes. The information on CD is used for performing quantity takeoff. AutoCAD files are used for area calculations in dwellings, common spaces, parking areas, terraces, etc. During code checking, anything that does not comply with the building codes is noted down on the plan, section, and elevation printouts of the architectural project and sent back to the architect. The architect makes the required changes and resubmits the revised version of the architectural project. The project is checked again to see if the required changes are made. This manual code checking process can last up to one month and usually needs to be repeated more than once. The overall construction permit process is represented as a flow chart in Figure 3.4.

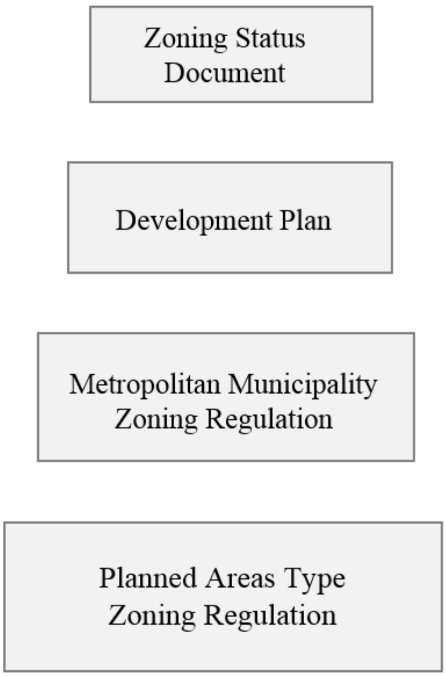


Figure 3.3. Required documents for architectural project control (general to specific).

3.1.2. GIS in Municipalities of Izmir

In recent years, digital city models play a central role for zoning information storage and communication. Currently, zoning information is generally kept as written texts and scanned plan documents in municipalities, which is unable to be processed digitally. Some of the zoning information is held digitally in isolated GIS environments. Neighborhood names, locations of authorized buildings, locations of landmarks, numbering of buildings, names of roads and streets are some of the information that are recorded digitally. However, there is no standard data model employed by all municipalities within the context of Turkey and existing digital information is inadequate for code checking. In addition, because of dependence on traditional methods of using information from printouts and archive documents and steep learning curves of newer systems; city information database systems are currently not being used.

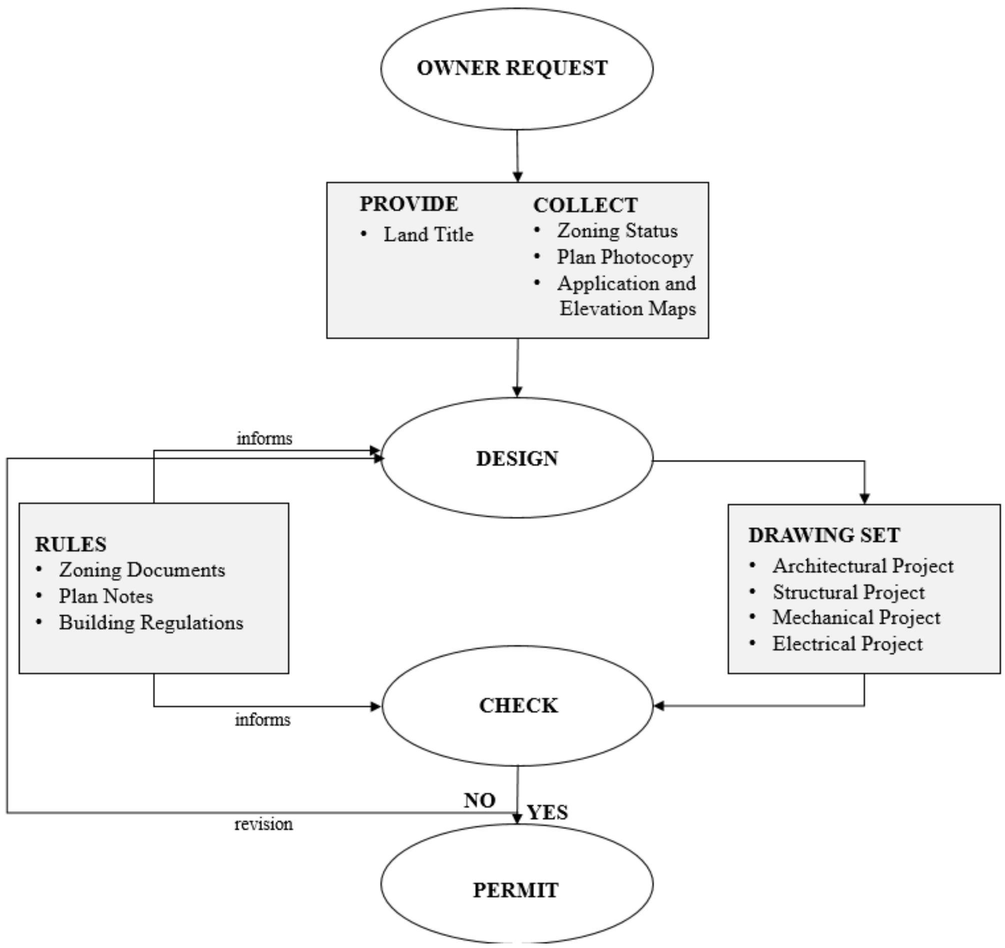


Figure 3.4. Flow chart for construction permit process

Lack of collaboration between local governments is another reason for the lack of a common city information database system. Usage of both digital and manual systems, and thus, the existence of various different systems that municipalities utilize, cause communication issues. There is no common system and every municipality organizes and systematizes its own data. There is a need for the development and adoption of standards for effective management and sharing of the ever-growing data on our cities. Currently, local governments aim to be a part of Europe and standardize spatial information database systems.

3.1.2.1. INSPIRE

In December 2016, a national education program was organized by Republic of Turkey Ministry of Environment and Urbanization for a new system named Infrastructure for Spatial Information in Europe (INSPIRE). Representatives from all municipalities including Bornova participated in this program. The aim of INSPIRE is to develop a spatial data infrastructure for the European Union. The INSPIRE project will encourage sharing of spatial data among public institutions, private sector and public, to form a spatial data infrastructure and ease accessing spatial data across boundaries. Current version of INSPIRE enables accessing and viewing limited spatial data sets and themes. However, topological databases and data themes are being developed to integrate information related to buildings, cadastral parcels, zoning, land use, transport networks etc. (European Comission 2016; Republic of Turkey Ministry of Environment and Urbanization 2016a).

In Turkey, Ministry of Environment and Urbanization (MoEU) is the institution that works for spatial data development, utilization and registration into urban information systems. These spatial data information systems are used by local authorities for carrying out the tasks related to planning, mapping, and infrastructure. General Directorate of Geo-Information System within MoEU is in charge of for the implementation of the INSPIRE Directive and works for developing the national spatial data infrastructure for Turkey. Turkey's National Spatial Data Infrastructure Project (TRGIS) is one of the several projects of The General Directorate of Geo-Information System for defining national geographic information standards. In TRGIS, there are 10 geo-data themes defined at the moment at urban scale including address, land use, land cover, building, vegetation, geodetic establishments, public services, urban furniture, water body, and transportation (Republic of Turkey Ministry of Environment and Urbanization 2016b).

For effective spatial data exchange and establishing a standardized structure of urban information system for interoperability, TRGIS data specifications are defined with UML-Unified Modelling Language schemas and the application themes are encoded with XML schema based on GML data exchange format (Republic of Turkey Ministry of Environment and Urbanization 2016b). In addition to GML; CSV, JSON/GeoJSON, KML, and SHP are the other data standards that will be used if needed (INSPIRE 2016a).

3.2. Code Document Analysis

It is essential to document the various types of information contained in building codes for identifying the spatial concepts and properties that concern the compliance checking process. Thus, analysis of the building code is the second stage. The code that is used throughout this research is the İzmir Municipality Housing and Zoning Code (IMHZCode). This stage is completed in the following two steps:

1. Analyzing the structure of the code document.
2. Decomposing the building code rules.

3.2.1. Structuring of Izmir Municipality Housing and Zoning Code

Building codes consist of zoning information that include rules, standards, and specifications. They are legal documents for the evaluation of building projects. The regulatory knowledge specifies technical and functional criteria that buildings and their settlements must meet throughout buildings' intended lives (Dimyadi et al. 2014). Many countries have national building codes that are valid for all building design and construction projects. Government agencies develop the building codes. In some instances, local jurisdictions such as regions where there is a historic fabric can develop their own building codes. In Turkey, Planned Area Type Development Regulations – Construction Law No.3194 prevails across the country. This law draws a generic framework for plan making and implementation processes in the entire country.

Respecting the law is mandatory to receive a building permit from the local municipalities.

Building codes include various types of information hierarchically organized as parts, chapters, and clauses that contain a number of concepts and statements. There are three types of clauses; general provisions, definitions, and prescriptions. Provisions give information about the aim and scope of the building code. They state the legal regulations the code is related to. Definitions describe the specific terminology used in building code. Prescriptions give information about the restrictions with regard to a given condition. Prescriptions are the only clauses that are used for evaluating the compliance of a building project. They are composed of several statements that need to be satisfied by the proposed project. The statements can indicate acceptable and threshold values, clarify calculation methods, identify obligations, define conditions, etc. (Macit 2014).

İzmir is the third most populous city in Turkey and its housing and zoning code is representative of codes that are valid throughout Turkey. Being a metropolis, İzmir's housing and zoning code is comprehensive and includes a large set of rules that have high complexity. As all the municipalities' housing and zoning codes consist of rules that are similar, the effort for formalization of zoning information in IMHZCode is generalizable and applicable to a wide a range of zoning code documents. Hence, IMHZCode can be adapted to regulations of other cities of Turkey. In this respect, IMHZCode has been chosen for the case study and is analyzed for comprehending the building codes. IMHZCode is the legal document that specifies minimum conditions that are need to be satisfied by architectural projects with their settlements. It consists of six sections as illustrated in Table 3.1.

Table 3.1. The Structure of IMHZCode.

PartNo	**Part Heading**	**Clause No**
I	General Rules	1-10
II	Definitions	11-23
III	Rules Related to Buildings and Land Readjustment	24-66
IV	Rules Related to Construction Permit and Building Occupancy Permit	67-76
V	Buildings, Building Parts and Facilities Subject to Special Rules	77-86
VI	Rules Rescinded, Interim Provisions and Entry in to Force	87-89

Prescriptions related to buildings and parcels are found in section III whereas the rest of the building code is informative.

3.2.2. Decomposition of Izmir Municipality Housing and Zoning Code

Izmir Municipality Housing and Zoning Code includes information related to buildings, building parts and the surroundings. In this stage, in order to identify the information that is not covered by BIM data, all clauses of IMHZCode are decomposed into a list of terms. These terms are then classified according to which concept they are pertinent to including parcel, building, block, setback, road, sidewalk and arcade. Table 3.2 lists all objects and terms mentioned in Izmir Municipality Housing and Zoning Code with where they are mentioned, and what they are pertinent to. Each of the clauses consists of several rules and statements that define conditions related to specific terms such as parcels, blocks, roads, roofs, walls etc.

Table 3.2. Terms used in İzmir Building Code.

ClauseId	StatementNo	Object/Terms	Property of
24	4	Width of Parcels	Parcel
24	4	Zoning type	Parcel
24	4	Construction Order	Parcel
24	6	Depth of Parcels	Parcel
24	6	Order of Parcel with Front Garden	Parcel
24/42	6/7	Order of Parcel without Front Garden	Parcel
24	7	Area of Parcels	Parcel
25		Organization of Parcels	Parcel
26		Integration and Separation of Parcels	Parcel
27	1	Front Garden Distance	Setback
27	3-4-5	Front Garden Distance of Adjacent Parcels	Setback
27	1	Dimension of Garden Facing Road	Setback
27	1	Dimension of Garden Facing Green Area	Setback
27	1	Dimension of Garden Facing Parking Area	Setback
27/42	6	Side Garden Distance	Setback

(cont. on next page)

Table 3.2. (cont.)

ClauseId	StatementNo	Object/Terms	Property of
27	9	Rear Garden Distance	Setback
27	13-14-15	Rear Garden Distance of Adjacent Parcels	Setback
27	10	Number of Roads Facing Parcel	Parcel
27	10	Corner Parcel	Parcel
27/36	11 / 3	Distance Between Two Buildings	Parcel
27/28		Has Existing Building	Parcel
28	1-2-3-4-5-6-7-8-9-10-11	Depth of Building	Building
28	3-4-5-6	Building Depth of Adjacent Parcels	Building
28	2	Distance to Rear Neighbor Border	Setback
29	1	Maximum Building Façade Dimension	Building
30/31/35	1-4 / 3-6 / 4	Height of Buildings	Building
30/31	1-4 / 3	Number of Storeys	Building
31		Temporary Constructions	Building
31	3-6	Total Construction Area	Building
32		Dead End Streets and Closed Roads	Road
33		Disaster Area	Block
34		Non-Settlement Area	Block
34	3	Dimension of Building Façade Coincident to Road	Building
34	6	Eave height	Building
34	6	Distance of Building to Road	Setback
34	6	Distance of Building to Parcel Border	Setback
35		Cadastral Parcels	Parcel
35		Parcel façade length	Parcel
36		Multiple Construction Permits in a Parcel	Parcel
37	1	Grade Elevation of Road Facing the Parcel	Road
37/38	1 / 6-7	Sidewalk Top Level	Sidewalk
37	4	Distance to Road (Front Garden Distance)	Setback
37	5	Front Elevation Base Level	Building
37	7	Given Building Height for Road	Road
37	11	Road Level	Road

(cont. on next page)

Table 3.2. (cont.)

ClauseId	StatementNo	Object/Terms	Property of
37	18	Front Garden Level	Parcel
37	19	Lowest Level of Building	Building
37	19	Rear Garden Level	Parcel
37	20	Side Garden Level	Parcel
38	1-2-5	Ground Floor Level	Building
39	1-2-3-4	Construction Material	Building
40	1-2	Maximum Projection of Eave from Building Border	Building
40	2	Width of Eave	Building
40	2	Distance of Eave’s Outer Line to Neighbor Parcels	Building
40	3	Minimum Height of Eave from Sidewalk Top Level	Building
40	5	Width of Sun-Shadings	Building
41		Roofs	Building
42	2-8-11-12	Width of Cantilevers	Building
42	6-9	Adjacent Parcel’s Cantilever Width	Building
42	2	Height of Bottom of Cantilever from Ground	Building
42	2-3-4	Distance of Cantilever Border to Side Neighbor Border	Building
42	2	Distance of Cantilever Border to Rear Neighbor Border	Building
42	3	Length of Cantilever	Building
42	3	Length of Façade where Cantilever is Located	Building
42	7	Road Width	Road
42	8	Height of Bottom of Cantilever from Sidewalk Top Level	Sidewalk
42	10	Width of Ornament Cantilever	Building
43		Light/Air Shafts	Building
44		Dwelling Units and Dimensions	Building
45		Interior Heights	Building
46		Windows	Building
47		Doors	Building
48		Lifts	Building
49		Stairs	Building

(cont. on next page)

Table 3.2. (cont.)

ClauseId	StatementNo	Object/Terms	Property of
50		Fire Escape Stairs	Building
51		Balustrades	Building
52		Chimneys	Building
53		Fire Precautions	Building
54		Water Tank and Plumbing	Building
55	2-4	Evened Earth Level	Parcel
55	4	Ground Floor Slab Top Level	Parcel
56		Porter’s Flat	Building
57		Auxiliary Buildings	Building
58		Lightning rods, TV Antennas and Air Conditioners	Building
59		Walls	Building
60		Fences	Building
61		Construction Site Buildings	Building
62		Cesspools	Building
63	1	Height of Arcades	Arcade
63	1	Depth of Arcades	Arcade
64	1-2-6	Number of Trees to be Planted	Parcel
65		Shelter	Building
66		Dimensions for Handicapped	Building

3.3. GIS-BIM Concept Mapping

After identifiying the classes and their properties in IMHZCode, an analysis is made for determining which of these code concepts and properties are lacking in BIM. These missing entities that are the focus of this study needs to be modelled and managed along with BIM data to be used by the architect during the early design phase for code compliant design and later for automated code compliance checking. For the analysis, IFC is selected as the BIM representation as it is the only commonly used data schema aiming to define a common language to enhance the collaboration, both for AEC/FM domain, and cross domain and cross-disciplines. The analysis determined whether each a domain model concept maps to an entity, a property from a property set (Pset) or a quantity from a quantity set (Qto) in IFC. The result of the analysis is shown in Table 3.3.

Table 3.3. Building Code Classes – BIM Classes mapping

Class/Property	Correspondence in IFC
Block	-
blockId	-
zoningType	Pset_BuildingUse_IfcLabel_Category of use e.g. residential, commercial, recreation etc.
constructionOrder	-
hasExistingBuilding	-
facedRoadId	-
isSettlementArea	-
isZoningCompleted	-
%50ofAreaHas LicensedBuilding	-
Road	Entity_An IfcCivilElement is a generalization of all elements within a civil engineering works. It includes in particular all occurrences of typical linear construction works, such as road segments, bridge segments, pavements, etc. Entity_An IfcGeographicElement is a generalization of all elements within a geographical landscape. It includes occurrences of typical geographical element, often referred to as features, such as roads, zones, trees, etc.
roadId	-
gradeElevation	-
level	-
slope	-
width	-
isDeadEnd	-
Arcade	-
arcadeId	-
height	-

(cont. on next page)

Table 3.3. (cont.)

Class/Property	Correspondence in IFC
depth	-
Sidewalk	Entity_An IfcCivilElement is a generalization of all elements within a civil engineering works. It includes in particular all occurrences of typical linear construction works, such as road segments, bridge segments, pavements, etc. Entity_An IfcGeographicElementType is used to define an element specification of a geographic element. Geographic element types include for different types of element that may be used to represent information within a geographical landscape external to a building. (pavement, T junctions)
sidewalkId	-
topLevel	-
width	-
Parcel	Entity_IfcSite
parcelId	Pset_IfcIdentifier_LandID: Identification number assigned by the statutory registration authority to a land parcel.
area	Pset_SiteCommon_IfcAreaMeasure_Total planned area for the site. Used for programming the site space. Qto_SiteBaseQuantities_Q_AREA_Gross area for this site, measured in horizontal projections.
depth	-
width	-
hasExistingBuilding	-
isCadastralParcel	-
zoningType	Pset_BuildingUse_IfcLabel_Category of use e.g. residential, commercial, recreation etc.
constructionOrder	-
hasMoreThanOne ConstructionPermit	-
totalConstructionArea	Qto_BuildingBaseQuantities_Q_AREA_GrossFloorArea: Sum of all gross areas of spaces within the building. It includes the area of construction elements within the building.

(cont. on next page)

Table 3.3. (cont.)

Class/Property	Correspondence in IFC
totalConstructionBaseArea	Pset_SiteCommon_IfcAreaMeasure_The area of site utilization expressed as a maximum value according to local building codes.
maximumHeight	Pset_SiteCommon_IfcPositiveRatioMeasure_Allowed maximum height of buildings on this site - according to local building codes.
maximum StoryNumber	-
siteCoverageRatio	Pset_SiteCommon_IfcPositiveRatioMeasure_The ratio of the utilization, TotalArea / BuildableArea, expressed as a maximum value. The ratio value may be used to derive BuildableArea.
floorAreaRatio	Pset_SiteCommon_IfcPositiveRatioMeasure_The ratio of all floor areas to the buildable area as the maximum floor area utilization of the site as a maximum value according to local building codes.
distanceBetween BuildingsInParcel	-
isCornerParcel	-
isFacingRoad	-
numberofRoadsFaced	-
faceRoadId	-
twinParcelId	-
numberofExisting Trees	-
NumberofTreesToBe Planted	-
gradeElevation	Entity_IfcSite_reference height of site is provided by: IfcSite.RefElevation_ Datum elevation relative to sea level.
clearDepth	-

(cont. on next page)

Table 3.3. (cont.)

Class/Property	Correspondence in IFC
type	Pset_BuildingUse_IfcLabel_PlanningControlStatus: Label of zoning category or class, or planning control category for the site or facility.
Building	Entity_IfcBuilding
buildingId	Pset_BuildingCommon_IfcIdentifier_BuildingID: A unique identifier assigned to a building. A temporary identifier is initially assigned at the time of making a planning application. This temporary identifier is changed to a permanent identifier when the building is registered into a statutory buildings and properties database.
height	Qto_BuildingBaseQuantities_Q_LENGTH_Height: Standard gross height of this building, from the top surface of the construction floor, to the top surface of the construction floor or roof above.
depth	-
baseArea	Qto_BuildingBaseQuantities_Q_AREA_FootprintArea: Gross area of the site covered by the building(s).
numberofStorey	Pset_BuildingCommon_IfcInteger_The number of storeys within a building. Captured for those cases where the IfcBuildingStorey entity is not used.
BoundaryLine	Entity_IfcRelSpaceBoundary_The space boundary defines the physical or virtual delimiter of a space by the relationship IfcRelSpaceBoundary to the surrounding elements. (A: There is interior/exterior space on the other side / B: There is a building element on the other side.)
boundaryId	-
direction/orientation	-
length	Qto_SiteBaseQuantities_Q_Length_Perimeter of the site boundary, measured in horizontal projection.
parcelId	-
adjacentTo	-
setbackDistance	-
gardenLevel	-

(cont. on next page)

Table 3.3. (cont.)

Class/Property	Correspondence in IFC
buildingId	-
distanceToRoad	-
adjacentCantilever Width	-
adjacentCantilever SideOffset	-

3.4. Development of the Zoning Domain Model

After IMHZ code analysis and concept mapping process, terms related to the surroundings that are not covered by BIM data are identified and brought together in an object hierarchy. Classes are identified by analyzing if they contain members having certain attributes; and properties are identified by analyzing if they contain data related to a specific element. Properties are shared by all objects of a class. For example, “construction order” is a property of the “parcel” class mentioned in IMHZCode. There are also properties such as area, width, distance etc., which are mostly used to define requirements. This study focuses on entites that are missing in BIM models, namely the classes related to zoning, including block, road, sidewalk, parcel, setback and arcade. Entites of the building and building parts are already considered and modelled by BIM models and are not the major focus of the thesis.

This partial domain model that is constituted by classes and relationships between them is described using Unified Modeling Language (UML) class diagrams (Figure 3.5). UML is an object oriented modeling language used for purposes such as database design and software development.

There are four types of relationships in a UML diagram: dependency, association, generalization and realization. In this study, association and generalization are used to represent relations between classes. Generalization represents subclass-superclass relations. It relates a general class to a specific one. For example, bay window and patio window are subclasses of window superclass. Objects of child class share the structure and the behavior of objects of the parent class and can define additional properties and behaviours that are not in the superclass. Generalization relation is represented using a solid line with a large hollow arrowhead pointing to the parent as shown in Figure 3.6. (Booch, Rumbaugh, and Jacobson 1999).

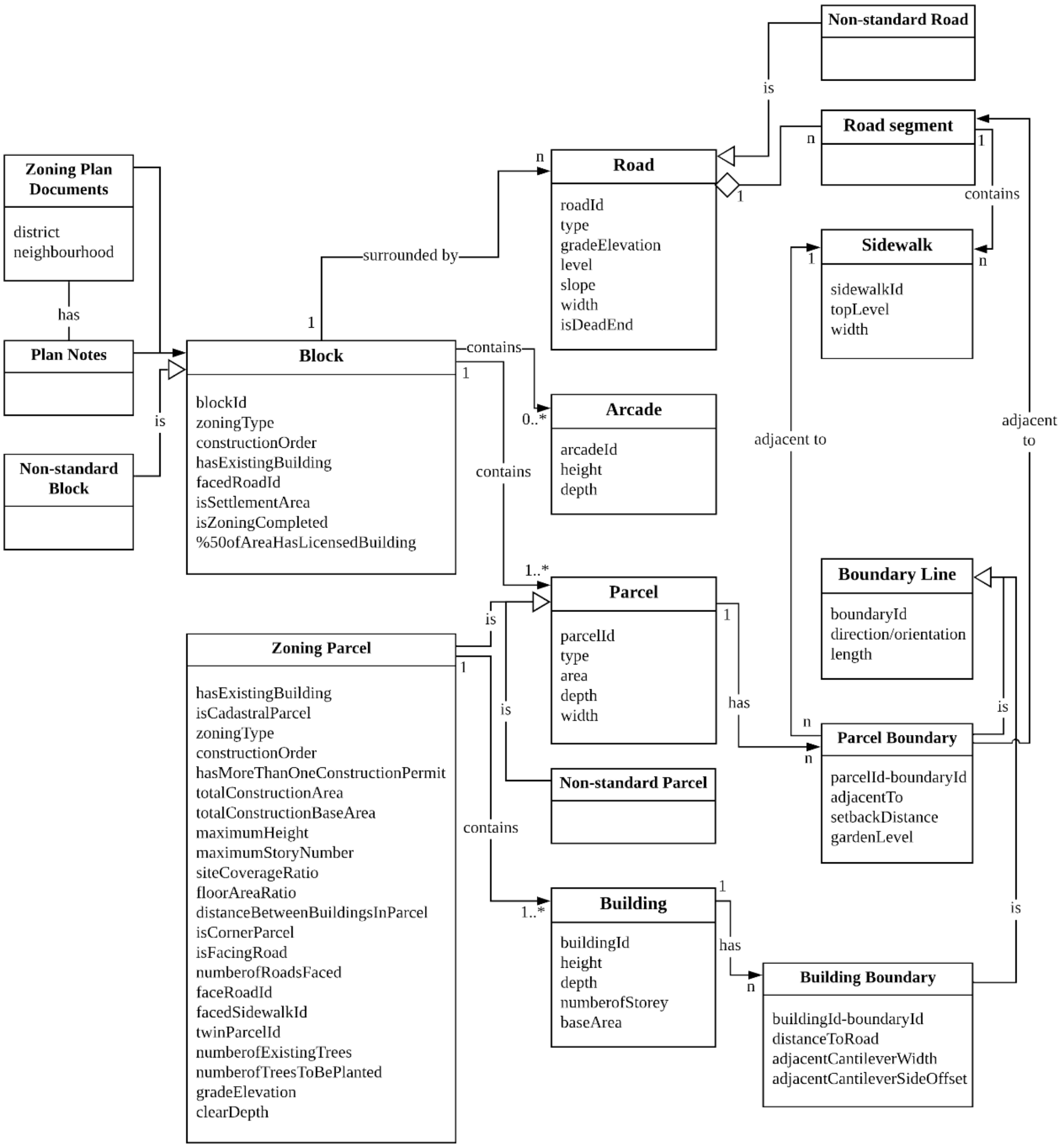


Figure 3.5. Building code rule set organized as a UML diagram.

Figure 3.6. Generalization arrow in UML diagram

Association describes links that are the connections among instances. By an association, an object of a class can be related to an object of another class. For example, a room and wall object can be related with an association relation as a room consists of walls. Associations sometimes have names for describing the meaning of the relation. The relation links are represented by solid lines and can indicate multiplicity when more than one object is associated for each object. There are five different associations between classes in a UML diagram; having one (1), zero or one (0..1), many (0..*), one or more (1..*) or an exact number n (Booch, Rumbaugh, and Jacobson 1999). Relations structured in the UML model are shown in the Figure 3.7.

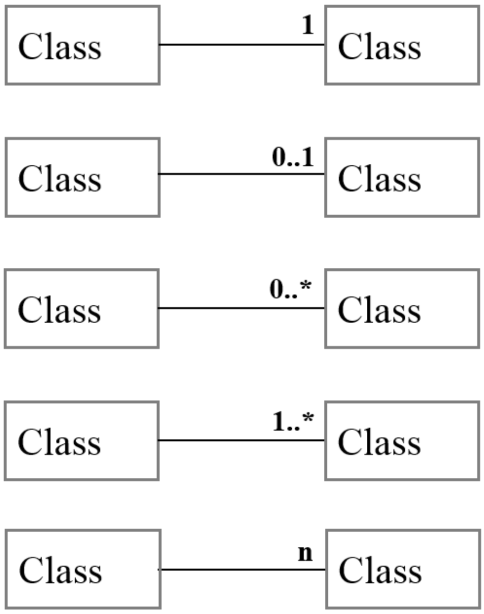


Figure 3.7. UML associations.

In the zoning domain model, “surrounded by” association is used for block-road/sidewalk relationships as roads surround blocks as shown in Figure 3.8. “Contain” association is used for block-parcel, parcel-building, block-arcade and road-sidewalk relationships. A block contains one or more parcels as shown in Figure 3.8 and zero or more arcades. A parcel may contain a number of buildings, and a road may contain more than one sidewalk. By removing the sidewalk width from the road width, the value of the vehicle road can be obtained as shown in Figure 3.9 (Portal Netcad 2019).

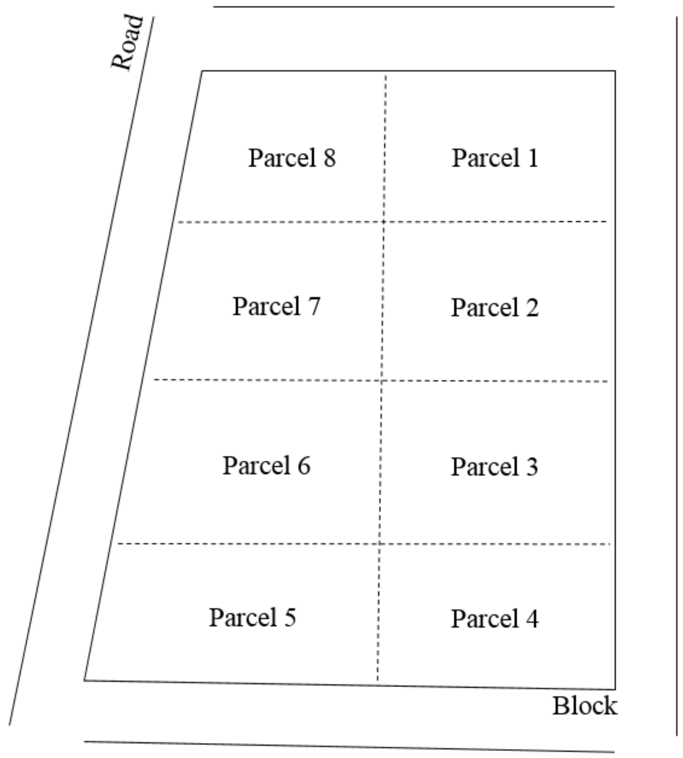


Figure 3.8. Parcels are contained by a block, which is surrounded by roads.

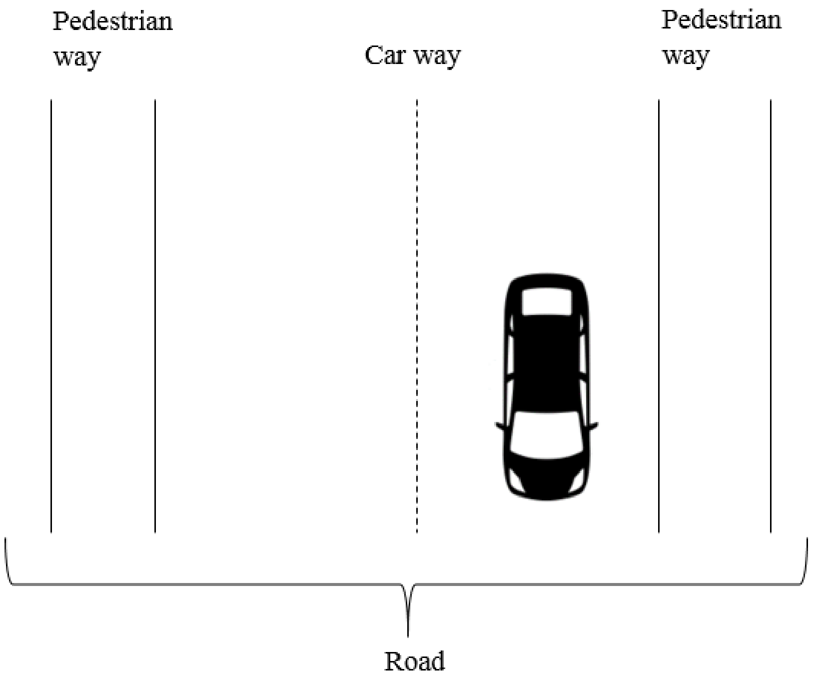


Figure 3.9. Containment association between roads and sidewalks.

“Has” association is used for defining the boundaries of parcel and building geometries as İzmir Building Codes consider lengths, relations and distances between parcel and building objects. Generalizations are “is” relationships in the UML model providing specializations for the parcel and boundary line classes. They are represented with a solid line having hollow arrowhead. Parcel parent class has two child classes: “zoning parcel” and “non-standard parcel.” “Non-standard parcel” is defined as a specialized parcel class for representing areas such as green-area, parking area, disaster area, market-place, recreation area and gas station area. Construction in these areas are generally not allowed and rather than İzmir Building Code document, specialized documents are valid for the organization of these kind of areas. Boundary line is the other parent class from which “parcel boundary” and “building boundary” classes are derived.

A detailed UML model of the Road class is shown in Figure 3.10. To deal with the elevation information of sloped roads, roads are divided into segments and subsegments. Roads are divided into segments at junction points. Each road segment can further be divided into sub-segments and each sub-segment is represented by nodes at both ends as shown in Figure 3.11. Nodes hold the elevation information, which is needed for the levelling of the building located in the parcel facing that road.

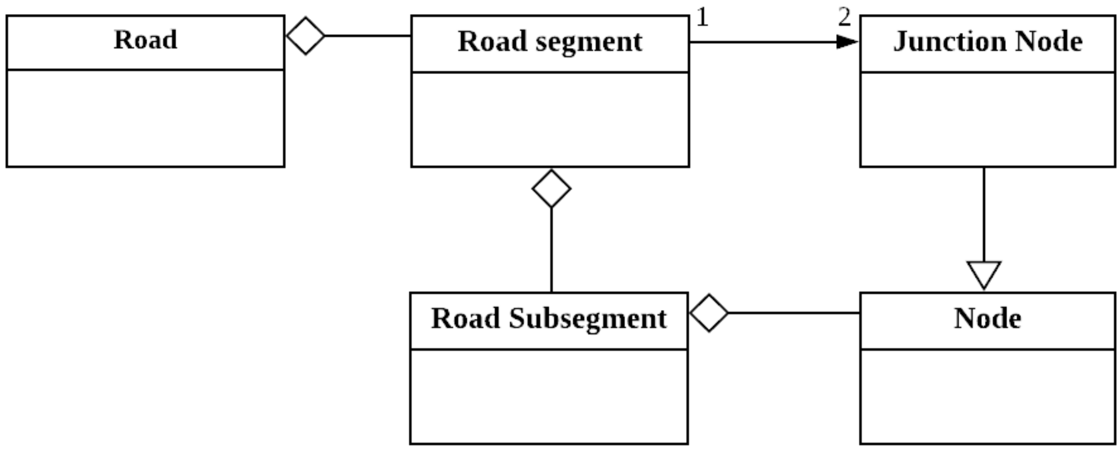


Figure 3.10. Road classes within Zoning Domain Model.

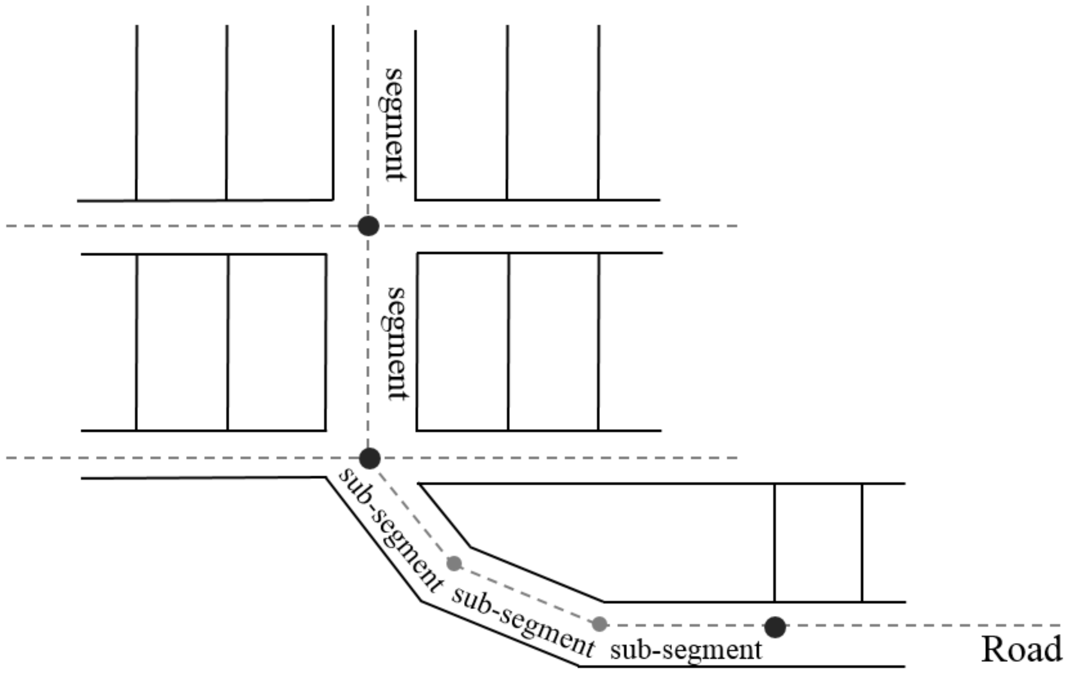


Figure 3.11. Road modeling example

3.5. Summary

In this chapter, IMHZ code document analysis and the building code rules' decomposition into code concepts and properties have been presented. Before bringing the code concepts and properties together in an object hierarchy, a mapping study is conducted between BIM domain and GIS domain and the results are listed in a table. The table provides a basis to identify the data on the surroundings required for code checking and their corresponding representations in the BIM domain. The analysis results showed that %70 of the data (48 of 69 terms) on the surroundings that was identified with IMHZ code analysis, are lacking representation in BIM, but can be modelled using GIS. As it is impractical to expect BIM to integrate geographical information in its domain, all the identified code data on the surroundings regardless of whether the data has a correspondence in BIM or not is modelled in GIS. Thus, when data on the surroundings are modelled using GIS, the GIS data can then be coupled with BIM data in a unified domain model appropriate to the field and constitute a complete dataset for the purpose of automated code checking.

CHAPTER 4

ZONING DATA IMPLEMENTATION IN GIS

This chapter describes how the zoning domain model presented in Chapter 3 is implemented in a GIS application with all the classes, properties and relations. The GIS application will be a complete data set that complements building design in BIM for a compliance checking process.

4.1. Geographic Information Systems

Geographic Information Systems (GIS) are the systems that store, model, query, analyze and present geographical data. They manage geographical information, query attributes of features based on defined constraints, analyze data referring to the questions than has been asked, compute new information based on the queries and analyses, and produce map displays and data reports. By using GIS, users connect geographical information with geospatial entities. A GIS is a spatial database that accumulates and manages different types of data on digital maps. It relates spatial objects with spatially referenced data. The stored data can be converted into various formats, which enables data interoperability between different systems.

Geographical data is stored and organized as layers in GIS. Each layer contains a group of real world objects such as rivers, buildings, roads etc. On a GIS map, each object has a geographically referenced location and shape. Based on the shape and characteristics that are in common, objects are grouped into layers as shown in Figure 4.1. For example, a water layer can include rivers and lakes. These objects are represented using vector data models and the boundaries of the objects are defined by x, y coordinate pairs.

In a vectoral GIS map, geographical data can be represented by one of the three main geometries, which are point, line and polygon, depending on its shape. Points are used for representing objects that are too small such as trees and electric poles. They are defined by a single x, y coordinate pair. Lines are used for representing objects that are long, narrow and continuous such as streets, roads and rivers. Two or more coordinate pairs define lines and each line has one start and one end point defined by coordinate pairs. Polygons are used for representing objects that are large enough to have boundaries such as cities, soils and blocks. They define areas and are defined by a closed loop of coordinates, which gather and constitute a polygon. In other words, polygons are constructed from a series of lines.

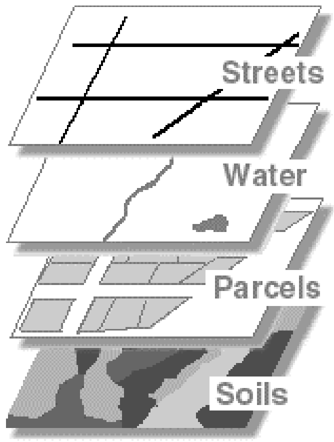


Figure 4.1. Spatial objects grouped as layers in GIS.
(Source: Esri Virtual Campus 2015)

In addition to using vector data models and representing objects by points, lines and polygons, spatial objects in GIS can be also represented using raster data. In a raster data model, the earth is divided into equally sized two-dimensional cells organized into columns and rows. Each cell corresponds to a portion of the earth and the value in the cell represents a characteristic of that geographical location such as temperature or rainfall. Raster data can be obtained from digital aerial photographs or satellite imagery.

The geographical data stored in GIS is also associated with non-graphic attribute data. Every object holds attribute information together with the graphical ones. Spatial data of objects describes the geometry and location in geographical space, whereas attribute data of objects identifies their characteristics. The attribute information of objects, are stored as attributes in GIS and are organized under attribute tables. The columns in the table refer to the attributes of objects, whereas rows in the table refer to the records. For example, width, height and name are the attributes of a road object organized in columns, whereas each row stores information of a different road object.

There are various GIS software packages available today such as ArcGIS, MapInfo, MicroStation GeoGraphics, GeoTools and QGIS. Most of the software packages are commercial software that are expensive and require expertise for managing geographical data. However, open and free software have been developing in recent years, which increase the usage of open software. Quantum GIS (QGIS) is an open and free software and was developed by Open Source Geospatial Foundation (OSGeo) that is a non-for-profit foundation established to prepare and support the use of open source geospatial software (OSgeo 2019). QGIS has numerous capabilities that other commercial geographical system alternatives have and supports numerous data formats. It has a wide range of plugin support and can visualize, manage, edit, update and analyze spatial data (QGIS 2019; INSPIRE 2016b). In this study, in order to provide a proof-of-concept implementation for the new zoning domain model required by automated code checking, an application is developed using the open source QGIS software.

4.2. Modeling Zoning Classes and Properties

Çamiçi neighborhood of the Bornova district in Izmir was selected for the case study. Its 2D map was obtained from the municipality. The CAD file was imported into QGIS environment and transformed into shapefiles - the native format for GIS software. Figure 4.2 shows the modelled vector data of the neighborhood.

Figure 4.2. Modeling Çamiçi Neighbourhood of Bornova.

First, for each class in the domain model a layer has been defined in QGIS. Features (instances of a class) are drawn using vector-based tools on the appropriate layer and their associated semantic data is inputted into the attribute table of each corresponding layer. Then, spatial analysis queries have been developed for automated creation of topological relationships between features.

Each class including block, road, parcel, sidewalk, building and arcade is handled as individual layers. The geographical features (objects) belonging to each layer are modeled using vector data tools. Vector data types used by each class are listed in Table 4.1.

Table 4.1. Vector data model of UML classes.

Classes/Features	**Vector data model**
Block	Polygon
Parcel	Polygon
Road	Line
Sidewalk	Line
Building	Polygon
Boundary Line	Line

A block is a piece of land, which is already built-up or open to new developments, surrounded by roads. The block, and the parcels located in each block, are assigned numbers and letters as identifiers. A parcel (or lot) refers to a building site. Moyer and Fisher (1973) define a parcel as: ”a unit of real property with rights and interests.” It is a subdivided property that can be purchased and developed by the property owner within boundaries defined by local regulations. Each building located in a parcel has both a parcel Id and a block Id for identifying its location. For modeling block, parcel and buildings in QGIS, polygons are used as shown in Figure 4.3, Figure 4.4, and Figure 4.5. Building polygons are contained within the parcel polygons, and parcel polygons are contained within the block polygons.

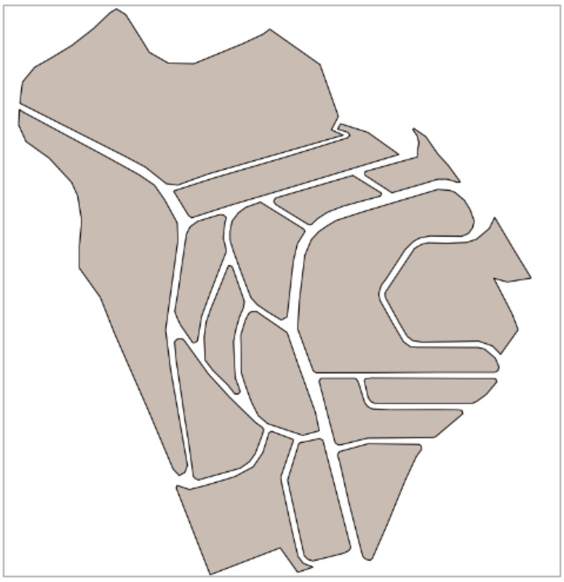


Figure 4.3. Block layer modelled using polygons.

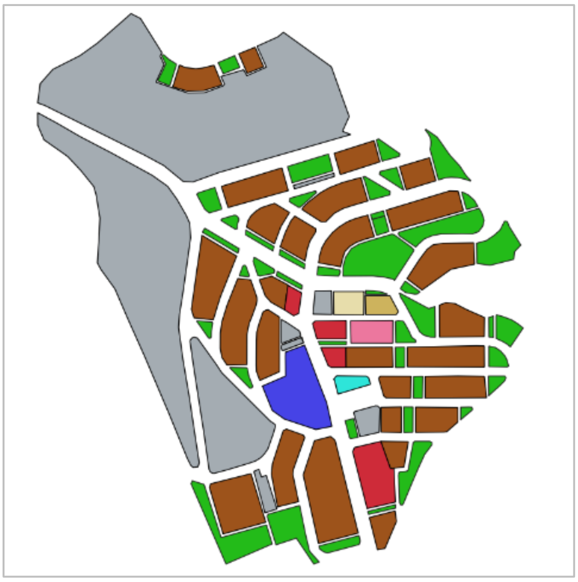


Figure 4.4. Parcel layer modelled using polygons.

The reason of using polygons for modeling parcel, block, and buildings is related to Izmir Municipality Housing and Zoning Code. Building codes require information on dimensions of features. How much area features cover and how they are related with each other and with the surrounding features are necessary for the required topological analyses in QGIS. For example, consider the following rule statement:

> “IMHZCode, Clause 27: Parcel areas, where high buildings are allowed to be constructed, cannot be less than 3000m^2.”

Figure 4.5. Building layer modelled using polygons.

BIM data that reflects the design that is being checked will of course include data on the site and this data can represent the parcel and supply the area information. However, the authority that is checking the design should rely on its own information on the parcel. Another example is the following statement:

> “IMHZCode, Clause 27: In blocks using contiguous order, if more than 50% of the block area consists of licensed buildings, than the rear setback distance is the rear setback distance of any building on the block with the same height.”

For this rule, information on the whole block is required in order to carry out the necessary calculations. Naturally, BIM data should not be expected to hold such information, but it should be supplied by GIS systems maintained by the city.

Parcel and building feature boundaries are explicitly represented with boundary line classes. İzmir Building Codes query adjacency relationships and differentiate between front, side, and rear borders of buildings and parcels. Boundary lines are vector-based lines automatically derived from polygons. Each boundary line can be associated with the feature it is adjacent to and have its own setback distance, allowing processing of rules such as the following:

"IMHZCode, Clause 27: Minimum setback distance of wooden buildings to surrounding parcel borders with detached order can not be less than 5m."

Setback distances are the minimum distances, established by city authorities, between the building and the road, or the neighboring parcel. The distance is calculated by identifying the closest point of the building to the parcel boundary. The purpose of setback distances is to ensure necessary space for visibility, access, ventilation, sunlight and privacy. Each parcel has a front, side and rear setback distances parallel to the property lines as shown in Figure 4.6. Setback distances define the buildable area for the parcel. Buildings cannot be constructed within setback distances; however, overhead projections such as cantilevers and eaves are allowed to extend into the setback distances.

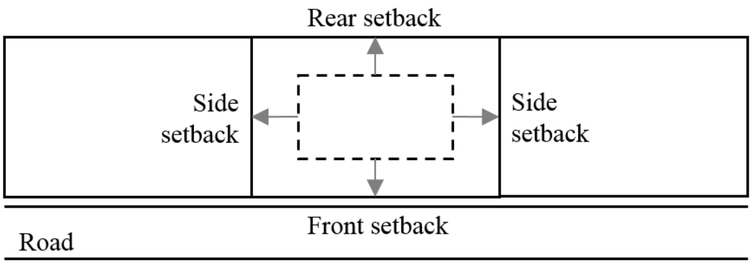


Figure 4.6. Setback distance shown in parcel.

Setback boundary is an imaginary line. For representing and identifying front, side and rear facades of parcels and buildable areas, line vector data is used. By using the "polygon to line" feature of QGIS, a line dataset is created which contains all the borders of parcels and buildings. Each parcel object has at least one street front and the rest of the borders are adjacent to other parcels. If the parcel faces more than one road, then the border that faces wider road is the front façade of the parcel. If the roads have same widths, then the shortest frontage is the front façade (İzmir Büyükşehir Belediyesi 2013). Side borders are the borders that intersect with the front borders. The remaining borders are the rear borders.

Road is a route for accessibility and travel including roadways for vehicles and sidewalks for pedestrians. A sidewalk (pavement) is a paved and elevated path by the side of a roadway for the use of pedestrians. Sidewalks are part of the road and sidewalk width is included in the road width. In QGIS, roads and sidewalks are linear features. Roads are modeled using lines and nodes going through road centerlines as shown in Figure 4.7.

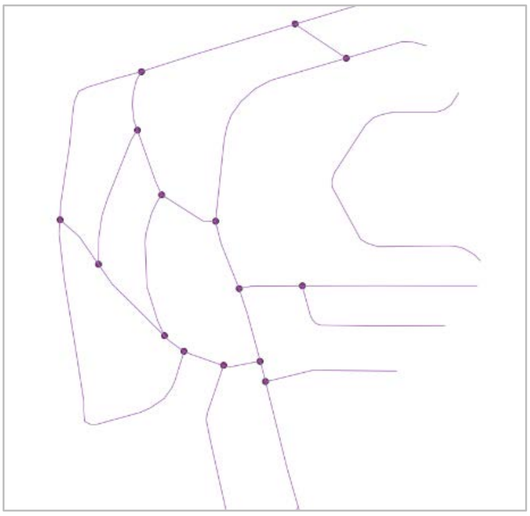


Figure 4.7. Road layer modelled using lines and points.

Sidewalks are vector-based lines automatically derived from polygon layer of block as shown in Figure 4.8. Each edge of sidewalk along the block can be associated with the parcel it is adjacent to. When needed, thickness can be given from the block boundary towards the road. However, representing road with centerline and sidewalk with the edge facing parcel are adequate for automated code checking based on IMHZCode regulations. Building code requires information only on how roads and sidewalks are related to surrounding features, area information and geometry is not necessary for roads and sidewalks.

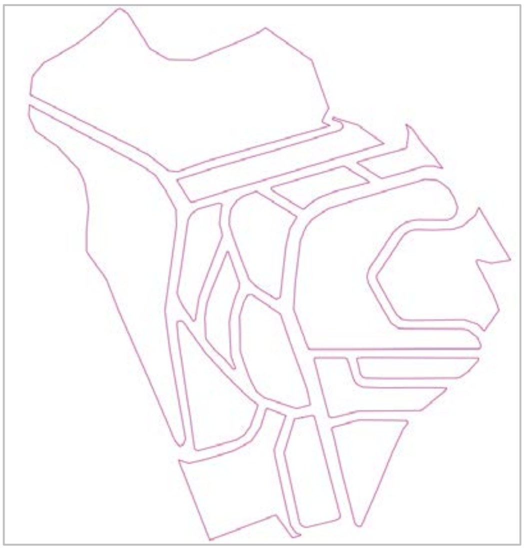


Figure 4.8. Sidewalk layer modelled using lines.

An arcade is a covered pedestrian path open to public use that may have a building over it. At least two sides of it is open. The location and use of arcades are specified in the zoning plan and an arcade is organized in accordance with the provisions of Izmir Municipality Housing and Zoning Code. If there is no specification about arcades in zoning plans, the height and depth dimensions are defined by the Zoning Code. Arcades are not common and the case study does not include one but arcades should be modeled using polygons.

As a next step, an attribute table is organized for each layer including the attributes defined in the domain model. The parcel layer attribute table is shown in Figure 4.9. The attribute tables contain semantic data on geographical features. Each feature is represented as a different record in the tables. Data entry into tables is manual but partly automated. GIS provides tools that enable calculations such as width, height, length, area and thus, decreases the required time for entry into tables. “Required area for locating trees,” “height difference between ground floor base level and sidewalk top level,” and “height difference between evened earth and ground floor slab top level” properties are some examples of attributes that are automatically calculated.

	id	cnstOrder	onCorner	zoningType	faceRoad	dstncBuild	depth	AREA_1	width	gradeElev	block_id
1	67	semi-detached	No	housing	Yes	0	26	5761	19	0	12
2	66	semi-detached	Yes	housing	Yes	0	26	5910	23	0	12
3	65	semi-detached	No	housing	Yes	0	26	7267	25	0	12
4	64	semi-detached	No	housing	Yes	0	26	6128	21	0	12
5	63	detached	No	housing	Yes	0	25	6566	23	0	13
6	62	detached	No	housing	Yes	0	25	5574	20	0	13
7	61	detached	No	housing	Yes	0	25	5828	21	0	13
8	60	detached	No	housing	Yes	0	25	6411	23	0	13
9	40	seperated	Yes	trade	Yes	0	40	30521	48	0	10
10	39	seperated	Yes	housing	Yes	18	27	14350	52	0	13

Figure 4.9. Zoning Parcels layer attribute table

The results of the digitizing of the classes and attributes formed the data representing the spatial organization of the neighborhood for the study. Henceforth, geographical analyses and inquiries can be performed using the QGIS features. When a spatial object is queried, number of entered attribute information can be accessed instantly.

4.3. Modeling Spatial Queries and Topological Relationships

Building codes have rules, which require querying spatial relations between building, parcel, block and road. For example, consider the rule:

> "IMHZCode, Clause 28: The depth of building in corner parcels is determined by the depths to be given to the neighboring parcels on the roads that the parcel faces."

The road id's that the parcel faces should be known for determining the other parcels that face the same roads. Likewise, consider the rule:

> "IMHZCode, Clause 37: Generally, buildings' maximum elevations are determined based on the road's grade elevation that the parcel, where the building is located, faces."

The parcel id where building is located and the road's id that the parcel faces should be known for determining the grade elevation. Such information can be defined manually during data entry as the city model is built in GIS but geoprocessing and spatial analysis tools of GIS applications are useful in determining such topological relationships between features automatically. The connection between the non-graphic attribute information and graphical information of an object makes GIS a powerful tool in handling various spatial queries. Topological data structure of GIS enables defining spatial relations between spatial objects based on their geographical and attribute data sets. By using the geographical location information, spatial analyses in GIS enables to answer questions that cannot be answered in traditional information management systems including relational databases (Blake 2007).

Analyses such as finding maximum-minimum values, calculating length, making summations, querying containment, intersection and aggregation relations allow constructing complex relationships required by building code processing. Analysis results can be recorded as new property columns in attribute tables. Spatial analysis features are used for both automating attribute table data entry as well as filtering the city information for the neighborhood information relevant for the design to be checked during the export of GML files.

Three basic spatial relations between zoning objects in IMHZCode are containment, adjacency and intersection shown in Figure 4.10. For containment relation, one object is being contained within another. There is an included-in relation and one object lies inside of an area entirely. This type of topological relationship is between a building located in a parcel; or a parcel located in a block. Adjacency is used for two features that touch each other. When two geometries are adjacent to each other, they share a common border. For example, this relationship is between two neighboring parcels of a block. For intersection, spaces share common geometries. There is a common region within two features. It is useful in identifying relationships between an arcade running underneath several buildings, and the buildings over it. Intersection also refers to containment and adjacency relations. If the shared geometry is the entire geometry of one space, then it is containment. If the shared geometry is a boundary, then it is adjacency.

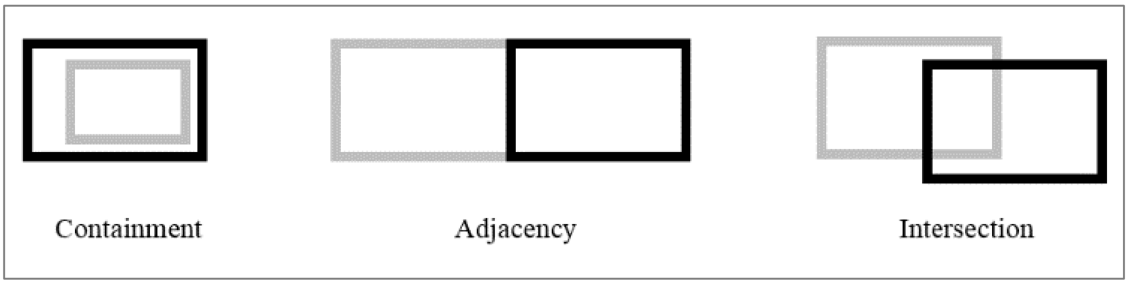


Figure 4.10. Containment, adjacency and intersection relations.

4.3.1. Identification of Zoning Concepts for Topological Analysis

Before studying topological relations of QGIS classes, clauses of Izmir Municipality Housing and Zoning Code are analyzed to identify the data type of the terms; value /enumeration / query (Table 4.2). Value represents an amount and is a resultant of a calculation or measurement. For example “area” is a property of parcel class and its value is gained through a calculation. Enumeration is a data type that enables users to define their items as a list. For example, “zoning type” is a property of parcel class and includes data types such as residential, commercial, industrial and mixed use. Query enables defining spatial relations between spatial objects based on their geographical and attribute data sets. Thus, the terms that require spatial queries are determined firstly, and then studied.

Table 4.2. Analyzing type of IMHZCode terms.

Terms	Type	Related to
Width of Parcels	Value	Parcel
Zoning type	Enumeration	Parcel
Construction Order	Enumeration	Parcel
Depth of Parcels	Value	Parcel
Order of Parcel with Front Garden	Query/Procedure	Parcel
Order of Parcel without Front Garden	Query/Procedure	Parcel
Area of Parcels	Value	Parcel
Front Garden Distance	Value	ParcelBoundary
Front Garden Distance of Adjacent Parcels	Query/Procedure	ParcelBoundary
Dimension of Garden Facing Road	Value	ParcelBoundary
Dimension of Garden Facing Green Area	Value	ParcelBoundary
Dimension of Garden Facing Parking Area	Value	ParcelBoundary
Side Garden Distance	Value	ParcelBoundary
Rear Garden Distance	Value	ParcelBoundary
Rear Garden Distance of Adjacent Parcels	Value	ParcelBoundary
Number of Roads Facing Parcel	Query/Procedure	Parcel
Is Corner Parcel	Query/Procedure	Parcel
Distance Between Two Buildings	Value	Parcel
Has Existing Building	Query/Procedure	Parcel/ Block
Depth of Building	Value	Building
Building Depth of Adjacent Parcels	Query/Procedure	Building
Distance to Rear Neighbor Border	Value	ParcelBoundary
Building Façade Dimension	Value	Building
Height of Buildings	Value	Building
Number of Storeys	Value	Building
Total Construction Area	Value	Building
Disaster Area	Enumeration	Block
Non-Settlement Area	Enumeration	Block

(cont. on next page)

Table 4.2. (cont.)

Terms	Type	Related to
Dimension of Building Façade Coincident to Road	Value	Building
Distance of Building to Road	Value	ParcelBoundary
Distance of Building to Parcel Border	Value	ParcelBoundary
Parcel Façade Length	Value	Parcel
Grade Elevation of Road Facing the Parcel	Value	Road
Sidewalk Top Level	Value	Sidewalk
Distance to Road (Front Garden Distance)	Value	ParcelBoundary
Given Building Height for Road	Value	Road
Road Level	Value	Road
Front Garden Level	Value	Parcel
Rear Garden Level	Value	Parcel
Side Garden Level	Value	Parcel
Sidewalk Top Level	Value	Sidewalk
Height Difference between Ground Floor Base Level and Sidewalk Top Level	Query/Procedure	Sidewalk
Adjacent Parcel's Cantilever Width	Query/Procedure	Building
Road Width	Value	Road
Evened Earth Level	Value	Parcel
Height difference between Evened Earth and Ground Floor Slab Top Level	Query/Procedure	Parcel
Height of Arcades	Value	Arcade
Depth of Arcades	Value	Arcade
Number of Trees to be Planted	Query/Procedure	Parcel

After the analysis, the spatial queries identified in Table 4.2 have been developed in QGIS.

4.3.2. Developing the Spatial Queries

In order to develop the queries identified in previous section, relationships structured in UML domain model should be also structured in QGIS environment. For example, to determine the "front garden distance of adjacent parcel," the parcel that is adjacent to that parcel in the same block should be known; to determine "number of roads facing parcel," the roads that are adjacent to that parcel should be kept as a list; to determine "height difference between ground floor base level of the building and sidewalk top level," the parcel where the building is located and the sidewalk that is adjacent to that parcel should be identified. Thus, the three basic spatial relations including containment, adjacency and intersection that were identified in section 4.3 are structured firstly between zoning objects identified in the zoning domain model. Containment relationship is structured for block-parcel, parcel-building, road-sidewalk and adjacency relationship is structured for block-road and parcel-sidewalk. In some queries, intersection relationship is also used as in the parcel-block relation, the shared geometry is the entire geometry of the parcel object. The queries are constructed using SQL-like expression syntax of QGIS.

Building-parcel relation is established on building layer by using an aggregation relationship (4.1), which queries the parcel and building classes together using "intersects" filter. By developing the query, the id of the parcel that contains the building is identified. Building belongs to only one parcel, and a parcel can contain more than one building.

aggregate(layer:='parcel', aggregate:='sum',expression:="id", filter:=intersects ($geometry, geometry(@parent))) (4.1)

Parcel-block relation is established in GIS by querying the parcel and block classes together using "intersects" filter. If the analyses is made over parcel layer (4.2), a new block column is added to the parcel layer's attribute table. Each parcel pairs with a block id, where the parcel is contained in. If the analysis is made over block layer (4.3), a new parcel column is added to the block layer's attribute table containing a list of parcel ids. Each block pairs with more than one parcel id as blocks can contain more than one parcels.

aggregate(layer:='block', aggregate:='concatenate',expression:="blockid", filter:=intersects ($geometry, geometry(@parent))) (4.2)

aggregate(layer:='parcel', aggregate:='sum',expression:="id", filter:=intersects ($geometry, geometry(@parent)), concatenator:='-') (4.3)

The other containment relationships in the UML domain model including block-arcade containment relationship and road-sidewalk containment relationships are also queried using the same containment formulas that were used for block-parcel and parcel-building queries.

Assessing the relation of parcel to surrounding roads is important for Izmir Municipality Housing and Zoning Code that should be studied for solving the entrance and access to buildings, planning service routes and parking areas. By knowing the id of the road, information related to that road can be known. For example, by comparing the widths of the roads, front façade of the building can be determined. By knowing the id of the road, the level of the road can also be known for determining the levels of the building.

The queries related to road include generally adjacency relations and it is necessary to know the geometry of the borders of the roads. However, as road layer is a line vector layer in current QGIS application, buffer analysis should be made firstly in case of road relation queries. A buffer analysis generates buffer zones and obtains information in specified distance from the determined object or point. Thus, buffers are created around each road object whose size corresponds to the width property of the roads and then, topological queries are applied (Figure 4.11).

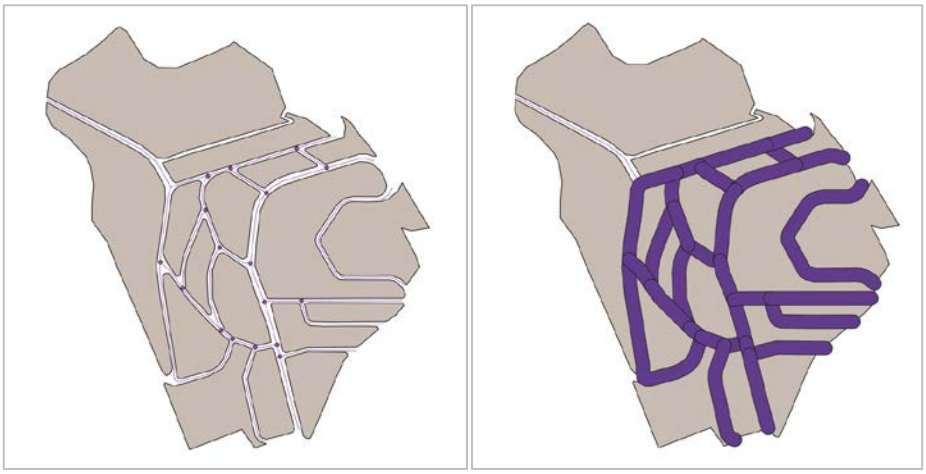


Figure 4.11. Buffers created around road-line layer.

Then, block-road adjacency relationship is queried using the road’s buffer layer and then, the road id’s that intersect with the block are identified with formula 4.4.

```
aggregate( layer:='road_line_bufferr', aggregate:='concatenate',
expression:="idString", filter:=intersects ($geometry, geometry(@parent)) ,
concatenator:='-')
```
(4.4)

There are also boundary line classes in the UML diagram and the boundary line-object relationships should be also structured. However, rather than drawing lines, the already drawn parcel and buildable area polygons in QGIS are translated into line vectors and new boundary line layers are created automatically for each of the parcel and buildable area class. The boundary classes include the information of the related parcel id and boundary ids together in one layer.

After the relationships are structured between zoning domain model classes using containment, intersection and adjacency, the identified terms that require procedures are studied. The following terms are queried;

- Order of parcel with/without front garden
- Front garden distance of adjacent parcels
- Number of roads facing parcel
- Is corner parcel
- Has existing building
- Building depth of adjacent parcels
- Height difference between ground floor base level and sidewalk top level
- Adjacent parcel’s cantilever width
- Height difference between evened earth and ground floor slab top level
- Number of trees to be planted

“Order of parcel with/without front garden” is a property of parcel class. Front garden is the setback distance from the front façade boundary of the parcel to the front façade boundary of the building. Thus, rather than querying the relation, it can be inferred that if a parcel has front setback distance, then the parcel has front garden.

The terms that require adjacency relations such as front garden distance of adjacent parcels, building depth of adjacent parcels and adjacent parcel’s cantilever width can be easily queried. As adjacency relations are structured, the properties of the buildings located in adjacent parcels can be retrieved.

For identifying the number of roads that are adjacent to parcel, the roads should be counted that touch to the parcel. Thus, the aggregation relationship, which queries the building and road buffer layer, should be studied within parcel layer. Thus, a query (4.5) is developed that counts the roads, which touch to parcel layer by using “touch” filter, and then a new property column is added into the attribute table (Figure 4.12).

aggregate(layer:='road_line_bufferr', aggregate:='count',expression:="id", filter:=intersects ($geometry, geometry(@parent))) (4.5)

	cnstOrder	onCorner	zoningType	faceRoad	dstncBuild	depth	AREA_1	width	gradeElev	block_id	idString	faceRoadNm	faceRoadId	buildgArea
1	semi-detached	No	housing	Yes	0	26	5761	19	0	12	67	2	8-11	1539
2	semi-detached	Yes	housing	Yes	0	26	5910	23	0	12	66	2	8-11	1535
3	semi-detached	No	housing	Yes	0	26	7267	25	0	12	65	2	8-11	1539
4	semi-detached	No	housing	Yes	0	26	6128	21	0	12	64	2	8-11	1527
5	detached	No	housing	Yes	0	25	6566	23	0	13	63	2	5-11	1556
6	detached	No	housing	Yes	0	25	5574	20	0	13	62	2	5-11	1556
7	detached	No	housing	Yes	0	25	5828	21	0	13	61	2	5-11	1556
8	detached	No	housing	Yes	0	25	6411	23	0	13	60	2	5-11	1556
9	seperated	Yes	trade	Yes	0	40	30521	48	0	10	40	2	9-9	9298
10	seperated	Yes	housing	Yes	18	27	14350	52	0	13	39	2	10-11	3112
11	undefined	No	problemarea	No	0	91	426229	349	0	16	38	0	0	0
12	seperated	No	housing	Yes	15	38	19804	50	0	14	37	1	5	3112
13	seperated	No	housing	Yes	20	37	27501	75	0	14	36	1	5	4668
14	seperated	No	housing	No	0	33	5809	28	0	16	35	0	0	1556

Figure 4.12. Number of roads adjacent to each parcel.

The building code has properties related to corner parcels. Parcels at the corner of a block can have two or three front facades and are called corner parcels. However, to query corner parcel attributes, the parcel should be identified firstly, if it is on corner or not. “Is corner parcel” query can be done with a buffer analysis. Roads are modelled using lines and every road junction is identified with a point in QGIS. Hence, point buffers are created as shown in Figure 4.13 and the parcels that touch buffers are identified as corner parcels. The following code (4.6) queries road node buffer layer and block layer and if these two layers intersect, then the code results yes.

CASE WHEN aggregate(layer:='road_node_buffer', aggregate:='count', expression:="id", filter:=intersects ($geometry, geometry(@parent))) THEN 'Yes' ELSE 'No' END (4.6)

Querying if there is an existing building within the parcel or within the block is required for checking Izmir Municipality Housing and Zoning Code regulations. For example, for querying the clause “in attached ordered parcels, building depth is the depth of any building in a parcel within the same block,” existing buildings in the parcels within the same block should be identified. By querying the containment relation, if parcel has an existing building the query results yes; if not, the query results no (4.7).

```
CASE WHEN aggregate( layer:='building', aggregate:='count', expression:="id",
filter:=intersects ($geometry, geometry(@parent)))
THEN 'Yes' ELSE 'No' END                                                    (4.7)
```

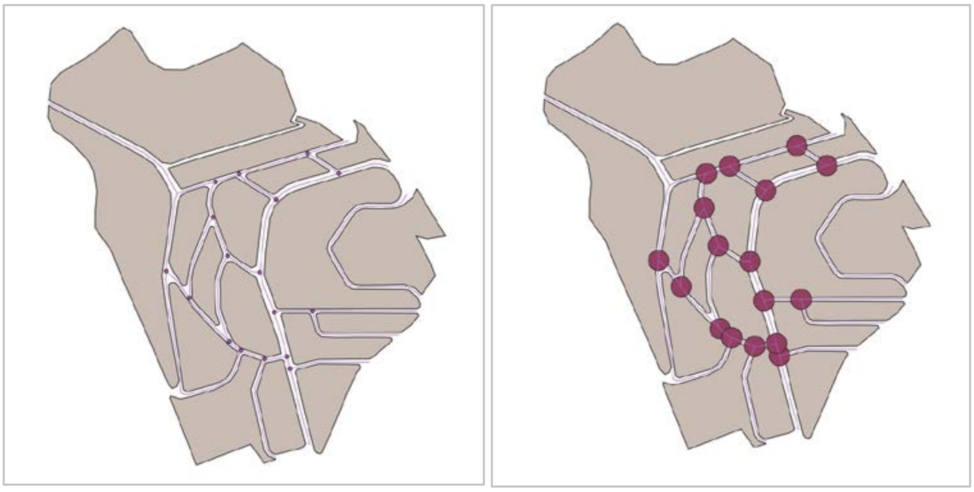


Figure 4.13. Buffers created around road-point layer.

In addition to spatial queries, some of the terms require operations between the zoning domain model class attributes. For example, for identifying the “height difference between ground floor base level and pavement top level;” and “height difference between evened earth and ground floor slab top level” the numeric property values should be subtracted from each other.

“Number of trees to be planted” is a property of parcel class. For identifying the the number of trees to be planted in a parcel, total building footprint area should be substracted from the parcel area and the result should be divided to 25. Thus, as a first step, the total building footprint area in every parcel is calculated structuring intersection relation between parcel and building and then, using sum aggregation (4.8). As a second step, the calculated area is subtracted from the total area of parcels and divided to 25 (4.9) according to the rule for finding the required number of trees that should be planted in each parcel.

aggregate(layer:='building', aggregate:='sum',expression:="baseArea", filter:=intersects ($geometry, geometry(@parent))) (4.8)

("area" - "buildgArea") / 25 (4.9)

In this chapter, a digital city model is developed and presented through defining zoning objects, properties, geometry and topological relations of the objects. The GIS application will be a complete data set that complement building design in BIM for a compliance checking process.

4.4. Exporting Zoning Data from QGIS

Currently, digital building models can be taken from BIM models and computer-based representations of building codes can also be taken from developed systems (Yabuki and Law 1993; Kiliccote and Garrett Jr 1998; Kerrigan and Law 2003; Eastman et al. 2009; Dimyadi and Amor 2013; Macit İlal and Günaydın 2017). And it appears that the complementary data required for compliance checks can be provided by GIS. Relations between objects can be resolved by "spatial query" in GIS and all objects and properties that concern any parcel can be extracted and saved as GML file to be sent to programs that will do automated code checking.

During the code checking process, the design information is submitted as a BIM file that includes the block and parcel number of the building, the neighborhood name and the building id. The GIS should complement this data with information only on the relevant neighborhood and not the whole city. This is done by filtering the GIS data based on the parcel id. The block containing the parcel as well as a number of neighboring blocks and surrounding roads are filtered and can be saved to a single GML file for use during code checking. GIS allows modeling as layers. All layers (block-road-parcel-building) are joined and exported as a single file. By using “package layers” tool of QGIS, an algorithm collected the selected layers and packaged them together into a single GeoPackage database. Then the gpkp format file was converted into GML file.

XML based Geography Markup Language Encoding Standard (GML) format is used for data export because it is an open and vendor-independent geographic information exchange standard developed by The Open Geospatial Consortium (OGC) whose mission is to promote the industry consensus development in the area of geoprocessing. It provides easy integration of spatial data during data transport. It is a data model for defining, modeling, transporting, and storing the data types and constructs for describing the geographic features and their locations (Reed 2010; Opengeospatial 2017; Peachavanish et al. 2006). GML is a component-based data format storing information about each geographical object separately. Hence, each feature's attribute information can be read and extracted separately. A section of the GML file that includes data on two parcels is shown in Figure 4.14.


```
<gml:featureMember>
  <ogr:parcel fid="parcel.0">
    <ogr:geometryProperty><gml:MultiPolygon srsName=
    <ogr:ID>62</ogr:ID>
    <ogr:cnstOrder>attached</ogr:cnstOrder>
    <ogr:onCorner>no</ogr:onCorner>
    <ogr:zoningType>housing</ogr:zoningType>
    <ogr:faceRoad>yes</ogr:faceRoad>
    <ogr:dstncBuild>0</ogr:dstncBuild>
    <ogr:depth>25</ogr:depth>
    <ogr:AREA_1>550</ogr:AREA_1>
    <ogr:width>22</ogr:width>
    <ogr:gradeElev>0</ogr:gradeElev>
    <ogr:block_id>1</ogr:block_id>
    <ogr:faceRoadNm>2</ogr:faceRoadNm>
    <ogr:buildgArea>330</ogr:buildgArea>
    <ogr:buildingId>23</ogr:buildingId>
    <ogr:bldngHeigt>18</ogr:bldngHeigt>
    <ogr:exstngBuld>yes</ogr:exstngBuld>
    <ogr:cnstrPermt>no</ogr:cnstrPermt>
    <ogr:cnstrcArea>5940</ogr:cnstrcArea>
    <ogr:cnstBaseAr>330</ogr:cnstBaseAr>
    <ogr:twnParclId>0</ogr:twnParclId>
    <ogr:nmbrExTree>0</ogr:nmbrExTree>
    <ogr:nmbrTreePl>1</ogr:nmbrTreePl>
    <ogr:type>null</ogr:type>
    <ogr:faceRoadId>62-63</ogr:faceRoadId>
    <ogr:clearDepth>15</ogr:clearDepth>
    <ogr:maxHeight>18</ogr:maxHeight>
    <ogr:maxStryNmb>6</ogr:maxStryNmb>
    <ogr:siteCoverR>0</ogr:siteCoverR>
    <ogr:floorAreaR>0</ogr:floorAreaR>
  </ogr:parcel>                                         1
</gml:featureMember>
<gml:featureMember>
  <ogr:parcel fid="parcel.1">
    <ogr:geometryProperty><gml:MultiPolygon srsName=
    <ogr:ID>63</ogr:ID>
    <ogr:cnstOrder>attached</ogr:cnstOrder>
    <ogr:onCorner>no</ogr:onCorner>
    <ogr:zoningType>housing</ogr:zoningType>
    <ogr:faceRoad>yes</ogr:faceRoad>
    <ogr:dstncBuild>0</ogr:dstncBuild>
    <ogr:depth>25</ogr:depth>
    <ogr:AREA_1>550</ogr:AREA_1>
    <ogr:width>22</ogr:width>
    <ogr:gradeElev>0</ogr:gradeElev>
    <ogr:block_id>1</ogr:block_id>
    <ogr:faceRoadNm>2</ogr:faceRoadNm>
    <ogr:buildgArea>330</ogr:buildgArea>
    <ogr:buildingId>24</ogr:buildingId>
    <ogr:bldngHeigt>18</ogr:bldngHeigt>
    <ogr:exstngBuld>yes</ogr:exstngBuld>
    <ogr:cnstrPermt>no</ogr:cnstrPermt>
    <ogr:cnstrcArea>5940</ogr:cnstrcArea>
    <ogr:cnstBaseAr>330</ogr:cnstBaseAr>
    <ogr:twnParclId>0</ogr:twnParclId>
    <ogr:nmbrExTree>0</ogr:nmbrExTree>
    <ogr:nmbrTreePl>1</ogr:nmbrTreePl>
    <ogr:type>null</ogr:type>
    <ogr:faceRoadId>2-4</ogr:faceRoadId>
    <ogr:clearDepth>15</ogr:clearDepth>
    <ogr:maxHeight>18</ogr:maxHeight>
    <ogr:maxStryNmb>6</ogr:maxStryNmb>
    <ogr:siteCoverR>0</ogr:siteCoverR>
    <ogr:floorAreaR>0</ogr:floorAreaR>
  </ogr:parcel>                                         2
</gml:featureMember>
```


Figure 4.14. GML data format representation of two parcels with their properties

A parcel with its surrounding information (Figure 4.15) can be also delivered to architects at the early stages of design both by text files and drawing files including the surrounding parcel, road and block information to ensure that their design complies with the regulations before submit to municipalities.

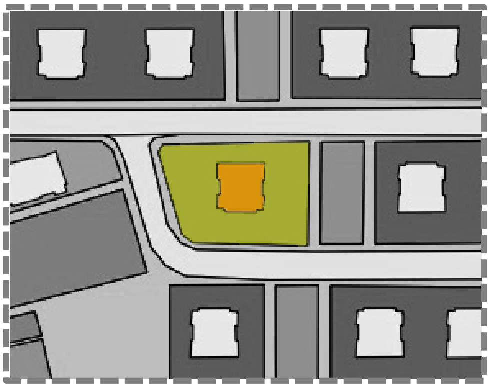


Figure 4.15. Parcel drawing with its surrounding information.

CHAPTER 5

MODELING RULES AND BIM MODEL

For the test case of the study, urban data from QGIS should be retrieved, the BIM data should be retrieved and the checking should be conducted based on the computationally modelled Izmir Municipality Housing and Zoning Code regulations and the retrieved GIS data. Thus, for developing prototype and showing the implementation of BIM and GIS integration in an automated compliance checking process, the following processes are described in this section;

1. Modeling of Izmir Municipality Housing and Zoning Code regulations.
2. Modeling of a BIM project.

5.1. Modeling Izmir Municipality Housing and Zoning Code Regulations

For representing the building codes, it is essential to determine the various type of information that building codes contain and analyze the organization of codes. As Izmir Municipality Housing and Zoning Code is analyzed already in stage two by identifying the building code concepts, properties and their relations, organization of codes are analyzed, decomposed and modelled in this stage.

5.1.1. RASE Methodology

Automated compliance checking systems require rules in computable format for checking of projects that are also in digital format. There are various methodologies in literature that are being used for rule representation and modeling in computable format for the purpose of automated compliance checking. Decision tables, rule-based models, logic-based models, object-oriented models and semantic models are some of them. However, it is hard to model comprehensive building code models with decision tables. On the other hand, rule-based models include hard-coding that makes it difficult to update and revise the model due to the building code amendments (change of setback distances or increase/decrease of building height limits). In addition, the codes are modelled by system programmers and hard to comprehend by the building code authors, which can cause misunderstandings between system programmers and the building code authors. Logic-based models and object-oriented models require code modelers to have specialized programming knowledge and expertise in case of modeling the rules (Macit 2014).

Maintainability is a crucial criteria in case of modeling building codes because the rules are updated and changed continuously. Another important criteria is that building code modeling should not require any programming knowledge and expertise. The rules should be adaptable with no further expertise in programming. Semantic modeling is the approach that meets all the criteria for efficient modeling of building codes. It is easy to understand, build and maintain building code representations for non-programmers using semantic modeling. By using a semantic approach, an AEC domain professional can identify and implement computable rules for model checking (Hjelseth and Nisbet 2010).

RASE (Requirement, Applicability, Selection, Exception) is a methodology that is being used for semantic modeling of building codes. It is similar to characteristic of building codes where each *check* should be in some way satisfied. There are four operators in RASE that structure a rule which are requirement, applicability, selection, exception. Requirements are conditions that should be satisfied by the building or building parts and include imperatives such as shall or must. There should be at least one requirement in a check. Applicability is the definition of the building or building parts associated with the requirement check. There should be also at least one applicability in a check. Selection operator is applied when there is a specialized condition among the applicable elements within the rule. Exception defines conditions where the rule can not be applied to the building or building elements. Exceptions are the opposite of applicabilites. They include clauses such as unless, and work by exclusion. Each of the four operator has a topic, property, comparator, value and unit properties (Macit 2014; Hjelseth and Nisbet 2010; Hjelseth and Nisbet 2011).

5.1.2. Four Level Representation

In this thesis, semantic modeling approach is used for developing a representation for the Izmir Municipality Housing and Zoning Code regulations. Rules are modeled using the four level representation paradigm (Macit 2014). Initially, concepts, properties and their relations referenced in building code text should be extracted and identified in domain level. As extracting and identifying processes are completed in chapter 3, modeling and representing are processed in this stage. The identified objects, properties and relations in chapter 3 are associated with the applicability operators of RASE methodology and utilized for modeling rules.

In Izmir Municipality Housing and Zoning Code, there are various concepts and some of them are related to each other such as “attached,” “detached” and “constructionOrder.” Rather than modeling attached and detached as seperate classes, a concept mapping table is created and they are modelled as “relatedSpace” property values of “constructionOrder” concept. This table includes all the concepts and identifies them as classes or filtered set of instances belonging to a class. Thus, complexity of the domain model is decreased by avoiding high number of concepts. The concept list in domain level is always editable for adding or removing new concepts if there is an update in the building code (Macit 2014).

Secondly, rules are modelled in computable format in rule level. Rules are modelled using the concepts constructed at domain level. Each rule statement is represented as a rule object and defines a single requirement that must be satisfied by a single property of a concept. Requirements include comparators (e.g. greater than, equal to), value (numeric, descriptive) and a unit measure. If there is a specialized condition among the applicable concepts, selection information is included in rule statements. If selection is true, then requirement is checked for compliance (Figure 5.1). Every rule object should include at least one requirement and zero or more selection objects. Macit’s approach modifies the RASE constructs. Exception information is represented within the selection construct and applicability information is embedded within the requirement construct (Macit 2014).

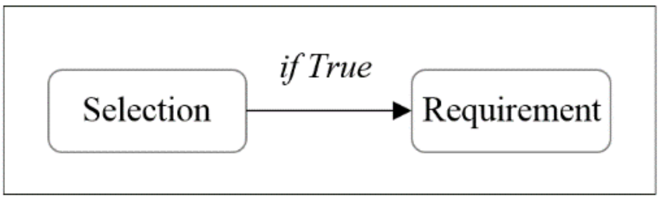


Figure 5.1. RASE constructs in rule level.

Thirdly, relationship hierarchy of rule objects (each representing a single requirement statement) are modelled with rule-sets at the rule-set level. In many cases, multiple rule statements define requirements for a property of a concept. The conditions of which one is applicable in any given context is modeled at this level. The rules are connected to each other with logical conjunctions such as OR and AND. OR conjunction is used for rules indicating alternatives for the same property depending on defined conditions. With OR conjunction, only one rule should be satisfied. AND conjunction is used for combining the rules indicating different conditions for the same property (Figure 5.2). With AND conjunction, all rules that indicate different requirements should be satisfied by the property of an object. At the end of this step, rule objects are grouped based on the property of the class they are associated with and constitute rule-sets of the representation model (Macit 2014).

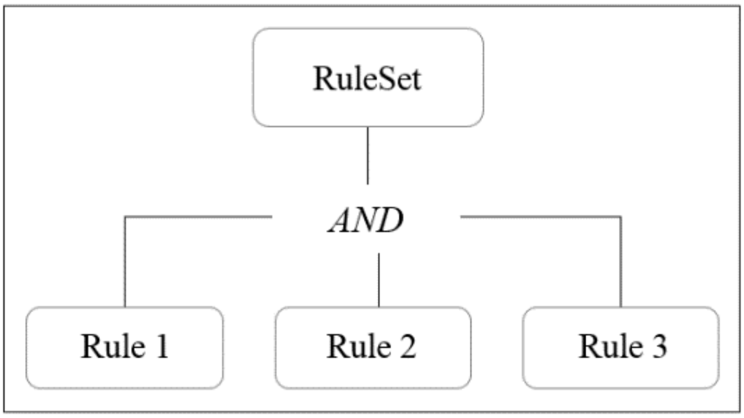


Figure 5.2. Rule objects connected with an AND relation in rule-set level.

Fourthly in management level, building code information is organized systematically by relating and grouping the rule-sets modelled in rule-set level. In a code checking environment, it is important for the user to determine which building codes should be applied for a given design problem. Therefore, rule-sets are grouped according to the concepts they are related to and this grouping enabled to access all the rules that apply to a determined concept (Figure 5.3). The groups are modeled using rule-set group objects (Macit 2014).

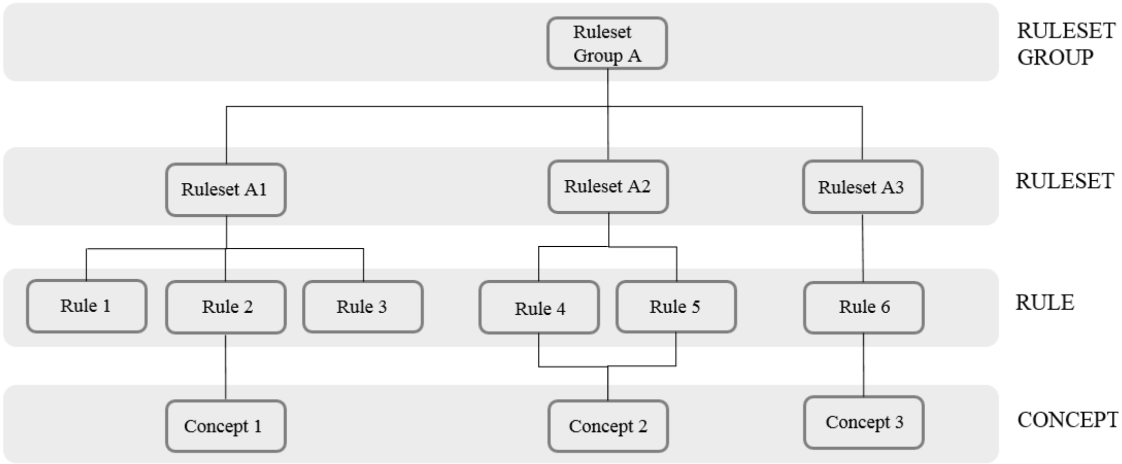


Figure 5.3. RuleSet group objects constituting rules

5.1.3. Analyzing Rule Structure

IMHZCode includes various type of rule statements including value checking rules, attribute checking rules, applicability clarifying rules, descriptive rules and qualitative rules. Analyzing the structure of the rules and identifying the type of each statement is needed for determining which of the rules can be and should be modelled with the four levelled methodology.

Value checking rule statements check correctness based on value and implies strict constraints. They indicate how things should be. The rules related to the distance, height, storey number, length are examples of this type. Such as the following:

> “IMHZCode, Clause 27: Rear garden distances are calculated by dividing building height into two.”
>
> “IMHZCode, Clause 63: Height of arcades are 3.50meters.”

Attribute checking rule statements check correctness based on an attribute. The attribute can both be a value or an enumeration. Such as the following:

> “IMHZCode, Clause 28: In semi-detached construction order, building depth is the depth of semi-detached building.”

Third type of rule statements clarify applicability conditions and identifies what to do based on specified conditions. Such as the following:

“IMHZCode, Clause 24: Parcel areas, where number of building storeys are ten or more, can not be less den 2000m^2.”

The first three type of statements are declarative statements, which declare information unambigously. They are measurable and clear in their meaning. These kind of statements can be transferred into computer processable format easily (Malsane 2015). Fourth type of rule statements are descriptive rules. They give general information and do not indicate acceptable and threshold values, clarify calculation methods or identify obligations. They just identify probabilistic situations such as the following examples;

“IMHZCode, Clause 26: Unification of parcels whose clearance height are different is not allowed according to zoning plans.”

“IMHZCode, Clause 31: Permanent constructions are not allowed in areas which are defined as residential development areas by the zoning plan.”

“IMHZCode, Clause 32: Parcels which are located near closed road or dead-end streets or located in the middle of a block are not allowed to be separated according to zoning plans.”

Fifth type of rule statements include qualitative evaluations, which are open to interpretation and require human judgement. They are mostly related to aesthetic values or under the initiatives of local authority. Such as the following:

“IMHZCode, Clause 37: In exceptional situations, where parcel has a high slope from road or there are different elevations in the same parcel, technical committee’s decisions based on their surveys in the area are valid for defining the elevations in site.”

Fourth and fifth type of rule statements are informative clauses. These type of rules are subjective rules and direct meaning cannot always be taken from them. They require human judgement for understanding the meaning. Thus, they are not checkable and can not be transferred into computer processable format easily. These statements are outside the scope of the thesis and are not modelled since they cannot be integrated into such a system. The classification of the rule statement of Clause-24 is given in Table 5.1 as an example.

Table 5.1. Classification of the rule statement of Clause-24.

Clause Id	Statement Id	Rule Type
C24	ST24.4	Clarify applicability conditions
	ST24.5	Check correctness based on attribute
	ST24.6	Clarify applicability conditions
	ST24.7	Clarify applicability conditions

Dealing with rules that are open to interpretation is another research topic and is not within the scope of this thesis. Since only value checking rules, attribute checking rules, applicability clarifying rules can be represented in digital format, they are identified and extracted from the Izmir Municipality Housing and Zoning Code. These rules are represented clearly by the four-levelled representation method.

5.1.4. Rule Database

The rules that require neighborhood data for the checking process are selected and modelled in this thesis. Clause 27 (setback distances), 28 (depth of buildings), 29 (façade of buildings), 37 (garden levels) and 38 (ground floor levels), of Izmir Municipality Housing and Zoning Code are used as examples for testing and demonstration of the developed prototype. There are 43 clauses in section III of IMHZCode document. Section III is analysed as the clauses that are used for evaluating the compliance of a building project are found in this section. It is seen that 26 of 43 clauses are only related to building data; hence, the clauses that require only building data are out of scope for the study. Even though all the other 17 clauses require data on the surroundings from GIS environment for checking the building, 4 of 17 require human judgement and cannot be modelled. Out of the remaining 13 clauses, 6 of them involve only simple checking of a value or values. Out of the remaining 7 clauses of interest, the three clauses on setbacks, eaves, and cantilevers (clauses 27, 40, and 42, respectively) use very similar constructs with regard to checking relationships to adjacent parcels and buildings. In order to avoid repetitions, only the most complex clause, which was clause 27 on setbacks, was selected for modeling. Thus, from the rules that can be transferred into computer processable and require data on the surroundings from GIS environment for checking the building, the clauses that are representative of the most complex rule constructs found in IMHZCode are selected and modeled.

Rules are modeled independent of the checking engine and stored in a database. Microsoft's relational database application Access is used to store and model the rules. Whenever IMHZCode is updated, the rules in the database can be modified independently without requiring any changes to the checking engine. Following tables are created within the database and provided in Appendix B:

- Domain object table
- Concept mapping table
- Rule table
- Node-list table
- Rule-set and rule-set group table.

As Macit's (2014) study is used as a reference for creating the tables and modeling the rules, the tables should be updated firstly based on the field of work. The data on the tables should be modified and domain concepts with their properties should be updated using the required field data. For this study, the DomainObject table is organized to contain information on IMHZ code classes/objects and their properties. Each record contains an object and property pair. BimBuilding and BimParcel objects from BIM; Block, Building, Parcel, FrontBoundary, SideBoundary and RearBoundary objects from GIS in the DomainObject table represent concepts that exist in the building code text. These classes with their defined properties are required for checking the unified BIM-GIS sample model. These pairs are used by multiple rules in Izmir Municipality Housing and Zoning Code.

The ConceptMapping table is another table that contains information required for constructing the code clauses. It lists all concepts referred to in the Izmir Municipality Housing and Zoning Code texts. Different from the DomainObject table, while some of the concepts are domain objects, some of them are specialized domain objects. For example, semiDetached and attached objects are specialized objects of block domain object. The *selection* construct of each rule in the Rule table contains a concept defined in the ConceptMapping table.

The Rule table contains the data on rule statements of the code clauses selected for this study. 24 rule records are listed in the table with their Id, description, concepts, properties, comparators and value information. The rules are organized under two parts.

Concepts and properties of the first *requirement* part comes from the DomainObject table while concepts and properties of the second *selection* part comes from the ConceptMapping table. If there is any update in Izmir Municipality Housing and Zoning Code (such as change of setback distances or increase/decrease of building height limits), just by changing the related information within a cell in the rule table updates the whole system. It is also possible to add new rules just by adding a new row into the table.

The Node_List and RuleSet tables contain information that structure the rule-set trees. For example, the Rule-set table contains dependency relations between rules about same property of the same object by using AND and OR conjunctions. While some rule sets have one level relation; some rules have complicated multilevel relations between rule objects which constitute hierarchical tree structures. For example ruleset RS28 shown in Figure 5.4, which is related to building object's depth property, has complicated multilevel relations and the rules are related with each other using OR and AND conjunctions.

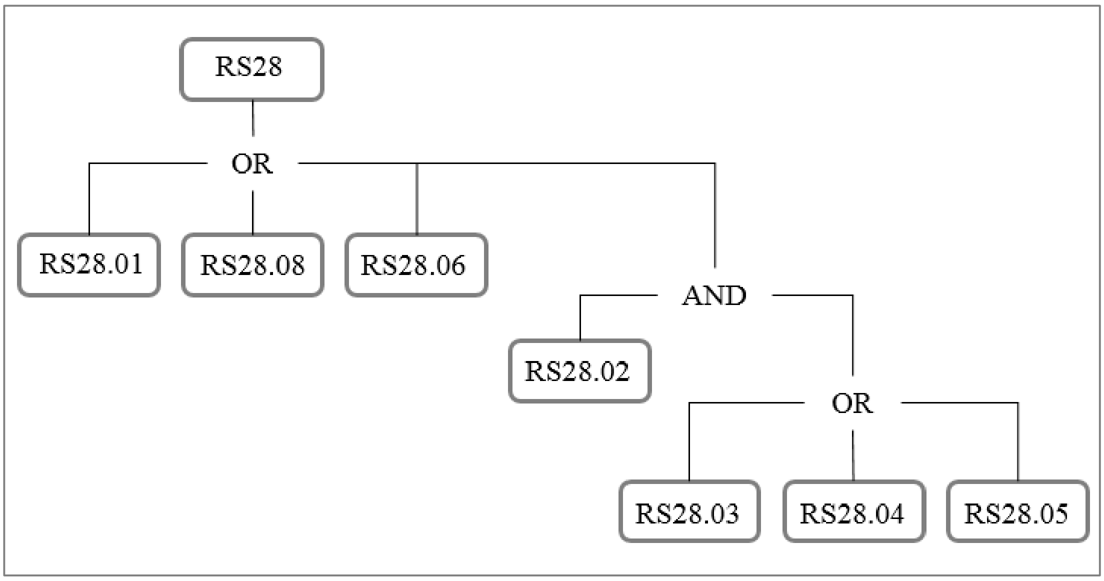


Figure 5.4. Representation of rule set RS.28.

Lastly, the RuleSet-Group table contains domain objects and rule-sets, and groups the rules based on the domain object they are related to.

5.2. Modeling BIM project

For creation of a unified domain model, a building project should also be modelled in BIM environment to be coupled with GIS data. Then, this coupled data can be managed in a third-party platform. For this study, a BIM project is modelled to conduct BIM-GIS data coupling and create the dataset required for compliance checking process that uses the code data modelled in Access database. As automated code checking can work if the building elements and their relations are modelled in an intelligent format, creating a building model in BIM application is fundamental. A building object with its surroundings is modelled using Graphisoft Archicad v21 as shown in Figure 5.5. As explained in the study, a BIM model does not include information related to the surroundings for a complete code compliance checking. Hence, only a parcel object could be modelled as the data in the surroundings and then exported together with the building object using IFC data model.

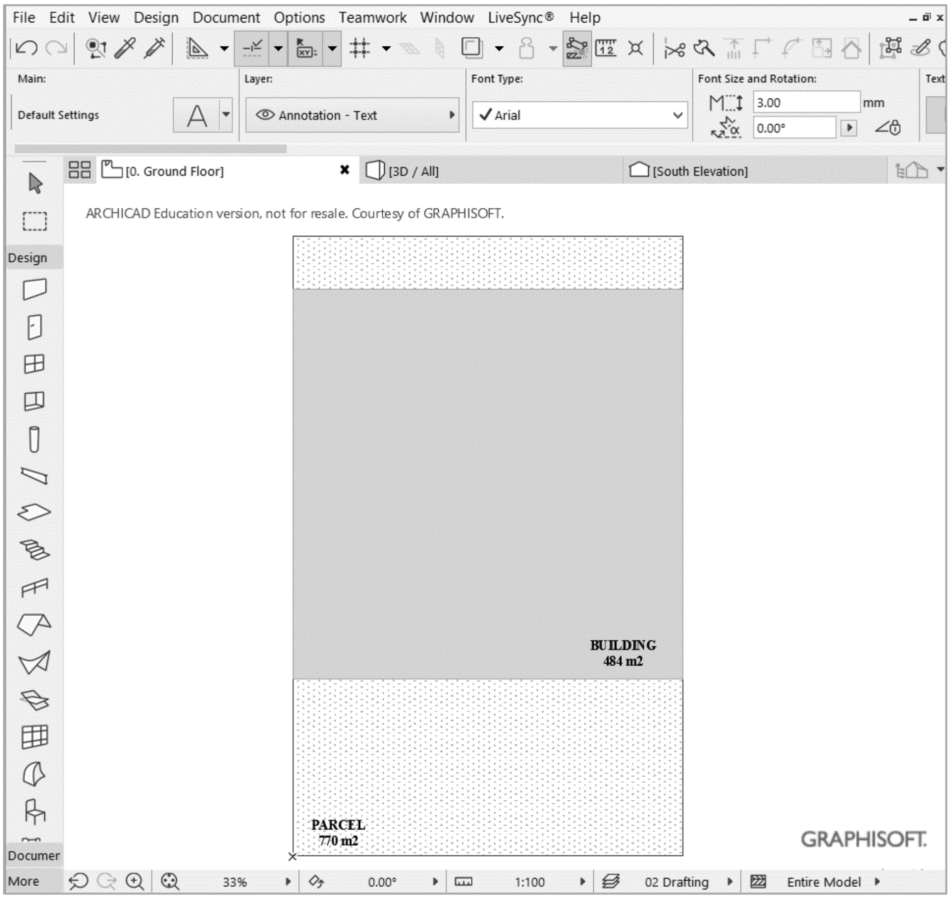


Figure 5.5. BIM model of the sample project.

Parcel is modelled using the “mesh” tool in Archicad and recorded as ifcSite entity in the IFC data model. Building is modelled using “zone” tool of Archicad and recorded as ifcSpace entity in the IFC data model. For demonstrative purposes, the building is modelled as a simple object and not modelled in detail as compliance checking of a detailed building model was conducted and studied in the study of Macit (2014) already. In addition, modeling the building’s borders is adequate for the thesis that studies the queries that relate building to its surrounding.

The IFC data model exported from Archicad includes building related IFC entities and IFC properties. The geometrical representation and location of the objects in Archicad environment are also defined in the IFC data using Cartesian coordinates. By using the coordinate information, numeric information that do not exist in BIM model can be calculated easily such as area, length, width, depth and height. Since making such calculations to obtain the missing building data is not within the scope of this thesis, the missing numeric data is manually encoded into IFC building data model using property-sets (PSet). Archicad enables to extend the IFC schema using property-sets. In IFC Project Manager, new custom properties can be added to the standard IFC schema. This enables adding properties to BIM objects that are not included in the IFC schema. Therefore, only for the prototype system, required numeric information, which can be obtained by making calculations from raw Archicad BIM model, is defined as extended property sets of ifcBuilding and ifcSite added to the model.

Using property-set extensions of the IFC schema, building related objects such as eave and cantilever can also be modelled in Archicad. One can even foresee that IFC could include these domain objects in future data schemas because they are related to building design. However, the remaining required neighborhood related information such as block, arcade, road, sidewalk and setback distance do not exist in BIM. Developed by buildingSMART, IFC 4 is the most current version and includes building controls domain, plumbing fire protection domain, structural elements domain, structural analysis domain, HVAC domain, electrical domain, architecture domain and construction domains (buildingSMART 2016a). There is a potential for the zoning domain objects to be included in future IFC versions because the IFC schema is updated continuously to new releases as new entities, properties and domains that represent data from different domains are added. IfcCivilElement class of IFC4 has evidential value. Currently, IfcCivilElement includes a general definition of elements within a civil engineering works including road segments, bridge segments, pavements, etc. Thus, IfcCivilElement class indicates that future versions of IFC standard can include specially defined classes and properties of road and pavement in detail.

CHAPTER 6

COUPLING BIM-GIS DATASETS AND DEVELOPING CODE CHECKING APPLICATION

For the test case of the study, urban data from QGIS will be retrieved, the BIM data will be retrieved, the datasets will be coupled and the checking will be conducted using the computationally modelled Izmir Municipality Housing and Zoning Code regulations. For constructing a test case for the study;

1. A Java application is developed that parses the GIS model, the BIM model and Izmir Municipality Housing and Zoning Code rules.
2. BIM and GIS datasets are coupled based on the parsed data.
3. A system is developed that checks coupled BIM and GIS data against Izmir Municipality Housing and Zoning Code data as a proof of concept for BIM-GIS interoperability and data integration.
4. The system is tested for its applicability and validity using use-case scenario.

6.1. Developing Code Checking Application

Before proposing an automated code checking application, understanding the existing approaches and reviewing previous checker environments are important. The following section reviews the current approaches and compares them with their advantages and disadvantages. Then, section 6.1.2 presents the developed code checking application for this thesis.

6.1.1. Rule-based Checking Systems

Efforts for developing rule-based checking systems for building models date back to 1980s. Development of Industry Foundation Classes (IFC) contributed to this process by being the building model schema for code checking (Eastman et al. 2009).

FORNAX, EXPRESS Data Manager (EDM), Solibri Model Checker (SMC) and SMARTcodes are examples to the commonly used platforms in literature that support implementation aspects of automated compliance checking systems. They are all object-based checkers, which apply building code regulations to BIM data.

Developed by novaCITYNETS Pte. Ltd, FORNAX derives required BIM information in IFC format and applies building code regulations for automated code checking. It is an object library written in C++. It includes objects and attributes from Singapore codes and the rules that apply to them. FORNAX also adds higher level semantics to BIM data for corresponding to building code checking requirements. For example, a kitchen drawing in IFC is not adequate for processing code checking of Singapore codes (a kitchen must be compartmentalized with minimum 1 hr fire rating). By FORNAX schema, IFC information can be extended to provide the needed data for checking of Singapore building codes. FORNAX structure building components as FORNAX objects by including related properties and methods to perform calculations and define bahaviours. These objects are extensible for covering the requirements of other countries (Greenwood et al. 2010; Khemlani 2005).

Developed by Jotne EPM Technology, a Norwegian IT company, EDM uses EXPRESS language to define the rule schema and enables interaction with BIM data in STEP or IFC format. It retrieves objects, object properties and relations from building information and translates the building model from IFC format into a Design Check model by applying an EXPRESS-X mapping (André Borrmann, Hyvärinen, and Rank 2009). For interpreting the domain-specific knowledge, EDM uses pseudo-code encoded into the system. It supports open development of rule checking and enables editing and writing of new rules and thus, open to user extensions (Eastman et al. 2009).

Developed by Solibri, a Finnish software company, SMC is a java-based software. It reads required BIM information in IFC format and carries out automated code checking against pre-defined rules in parametric format. Solibri also checks the model against simple geometrical relations such as “Beam must touch slab surface above” (André Borrmann, Hyvärinen, and Rank 2009). It has a library of parameterized rules and by changing parameters, users can modify the building code rules. Solibri includes a variety of checking functions including accessibility checking, space program checking and fire code exit path distance checking. As accessibility requires modeling of building objects and their relations, Solibri also retrieves building objects and their geometric data from building model for code checking (Macit 2014; Eastman et al. 2009).

Developed by The International Code Council (ICC), SMARTcodes is a platform that also reads required BIM information in IFC format. It enables to translate written language rules to computer codes by dictionary of domain-specific terms and semi-formal mapping methods. Rule statements and regulations are interpreted using the dictionary, which consists of properties, data types and units associated with each term, during checking process. The dictionary is also used for relating checker system to the IFC building model (Eastman et al. 2009).

Currently, EDM and SMC are the most widely known and commonly used checking systems. However, rule-writing capabilities in EDM using EXPRESS language is complex and is limited to a small group of people who have high level of expertise. Checking process in SMC only proceeds using the restricted range of objects and rules supplied by Solibri. The application programming interface is not publicly available and it does not propose an open environment for writing rules. Solibri Inc. provides a consultancy service for integration of user-specific rules (Eastman et al. 2009; Macit 2014; André Borrmann, Hyvärinen, and Rank 2009).

Therefore, neither of these systems were found suitable as platforms for code compliance checking. For this reason, a new checker platform is developed and implemented as a protoype for demonstrating the feasibility of the proposed model for BIM-GIS information coupling.

6.1.2. The Prototype

The prototype system has been developed and implemented using the Java language and consists of four components:

- Building Model Parser
- Geographical Information Parser
- Building Code Parser
- Building Code Checker.

Figure 6.1 illustrates the conceptual framework for the compliance checking system.

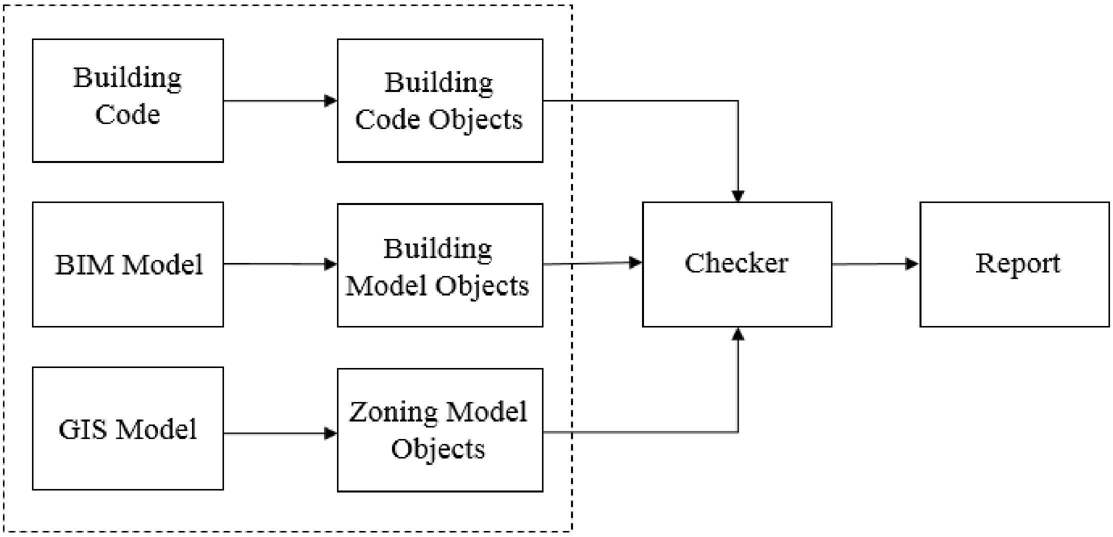


Figure 6.1. The developed code checking system in Java environment.

Building Model Parser accesses and reads the IFC data exported from BIM environment, constructs classes and properties, structures relations between classes, and assigns the values of the properties to domain object properties in Java. It structures a model to be checked. For accessing and extracting BIM data, JSDAI (Java Standard Data Access Interface) is used. JSDAI is an application programming interface (API) for parsing information from models written in EXPRESS language – the ISO standard product data modeling language (J. Zhang and El-Gohary 2017). In this thesis, BIM data exported as IFC2x3 is parsed using the JSDAI library and IFC objects are created in Java environment.

Geographical Information Parser reads the GML data exported from GIS environment, constructs classes and properties, structures relations between classes, and assigns the values of the properties to domain object properties in Java. For accessing and extracting GIS data, The Document Object Model (DOM) Parser is used. It is a way for accessing, manipulating and structuring XML documents. DOM holds a tree structure of objects from the document it parses. During parsing, a top-down approach is adopted which reaches to the root of the file first and then finds its child entities. Classes and attributes of zoning data are the nodes and thus, children of the tree structure (W3schools 2018). In this thesis, geographical information parser firstly reads the GML data with the help of DOM, and creates the GML objects in Java environment. Figure 6.2 shows block class with properties.

```java
package DomainZoning;

import java.util.ArrayList;
import java.util.TreeMap;

public class Block extends DomainObject {

    public String zoningType;
    public String constructionOrder;
    //public double touchRoadId;
    public double hasParcelId;
    public String facedRoadId;
    public String isSettlementArea;
    public String isZoningCompleted;
    public String halfOfAreaHasLicensedBuilding;
    public String hasExistingBuilding;

    public ObjectGeometry blockgeometry = null;

    public ArrayList<Parcel> parcels;

    public Block() {

    TreeMapping tmb = new TreeMapping();
    tmb.BlockMapping();

    parcels = new ArrayList<Parcel>();

    }
```

Figure 6.2. Block class, properties and relations created in Java environment.

After the Geographical Information Parser creates GML objects and assigns values to their properties, it constructs the necessary associations between objects using the relationship tables or properties in the database schema. It uses ID properties and as an example relates the parcel with the block it is located in; with the roads and sidewalks that it is adjacent to; with the other parcels it is adjacent to and the buildings that are located on it, etc. A more detailed example is, when the 'block_id' property of a parcel is parsed, first, the block object with that ID is searched for and a reference to is placed in the 'block' property of the parcel; and second, a reference to the parcel is added to the 'parcels' array of that block. Thus, the bi-directional object associations are constructed from the relational data.

As the third step, the Geographical Information Parser assigns the transient properties, which do not yet have values; but need to be calculated based on the values of other properties. These transient properties do not come from GML, but are required for building code concepts during code checking. While some of these properties can be calculated and assigned dynamically as needed in future systems, the prototype assigns all during this preprocessing stage. Reference building depth, reference setback distance and reference sidewalk top level are such transient properties that are required for checking the 28^{th}, 27^{th}, 37^{th} and 38^{th} clauses of the Izmir Municipality Housing and Zoning Code.

For clause 28, if there are existing surrounding buildings having the same construction order, the building depth is the depth given to the existing buildings in the parcels that face same road within the same block. Thus, for checking the 28^{th} clause, a 'reference building depth' needs to be calculated for the empty parcel of interest only for the purposes code checking. This value is not required to reside in either the BIM or GIS models. It is only created for the building code. The Java application firstly parses the construction order property and checks if the surrounding parcels have the same construction order with the parcel that will be checked. Based on the type of the construction order, the application runs the related method. For example, if the construction order is defined as attached, the system runs the findAttachedParcel method and identifies the parcels that face the same road within the same block. Then, application identifies the building depths on the parcels as a second step using contains topological relationship constructed between parcel and building objects. Finally, the checking process proceeds using the depth information gained from the data on the surroundings.

For clause 27, if there are existing surrounding buildings, the front and rear setback distances are the distances given to the existing buildings in the parcels that face same road within the same block. Thus, for checking 27^{th} clause, the Java application identifies the setback distances defined for the surrounding parcels and conducts the checking process using that data.

For clause 37, front garden level can be levelled by taking the top level of the sidewalk that the parcel faces as a reference; and for clause 38, ground floor base level of a building should be planned between + 0.50m to +1.00 m. from the upper level of the sidewalk. However, if the parcel faces more than one road, the levelling should be referenced from the widest road and if the roads have same widths, then the shortest frontage is the front façade. Thus for checking 37^{th} and 38^{th} rules, the application identifies the widest road and the sidewalk it faces for checking the front garden level and ground floor base level of the building. Table 6.1 shows the list of transient properties that are calculated for checking the 28^{th}, 27^{th}, 37^{th} and 38^{th} clauses of the Izmir Municipality Housing and Zoning Code.

Table 6.1. Transient properties list.

Classes	Property
Block	refFrontSetbackDist
Block	refSideSetbackDist
Block	refRearSetbackDist
Parcel	widestRoadWidth
Parcel	widestRoadSidewalkTopLevel
Building	refBuildingDepth
Parcel Boundary	setbackDistance

Building Code Parser reads the building code data from the MS Access database and instantiates rule objects for the checking process. Building Code Checker checks the building data extracted from BIM against building code data, using the data on the surroundings extracted from GIS. Building Code Checker ensures that the design complies with the regulations. At the end of the process, it presents a report. The checker firstly accesses the rule-set group objects table defined in section 5.1.4 of the thesis. It makes use of the rule-sets grouped according to the class they are related to. For each class, it applies all rules in the related rule-set group. If the rule-set is constructed using an AND conjunction, every applicable rule in the rule-set should be checked; if the rule-set is constructed using an OR conjunction, passing one of the rules in the rule-set is adequate.

6.1.2.1. Coupling of BIM and GIS Data

The main purpose of the study is to use the building information provided by IFC, integrate it with required data on the surroundings from GML and to create a dataset that will be used in an automated code checing process. Thus, data coupling is needed between BIM and GIS datasets to create a unified model. For creating a unified model, BIM and GIS data models are analyzed and compared in detail, and overlapping parts are coupled. The new integrated semantic model merges BIM and GIS data into one unique model.

For data coupling, firstly BIM and GIS data models are analyzed to reveal the classes that are similar in semantics and contain similar properties. This has been done using UML diagrams of BIM and GIS data models where the classes and properties are matched. The classes that exist in both UML diagrams and reference the same object have been considered "coupling elements." For semantic interoperability, the common concepts having the same meaning from BIM and GIS are coupled by using the common properties which are IDs. As the site/parcel object is at the overlap of BIM and GIS, parcel objects from BIM and GIS are matched/mapped using their IDs for unique identification as shown in Figure 6.3. For the GML file, ID of parcel object is used for unique identification; for the IFC file, the name of the IfcSite element is used.

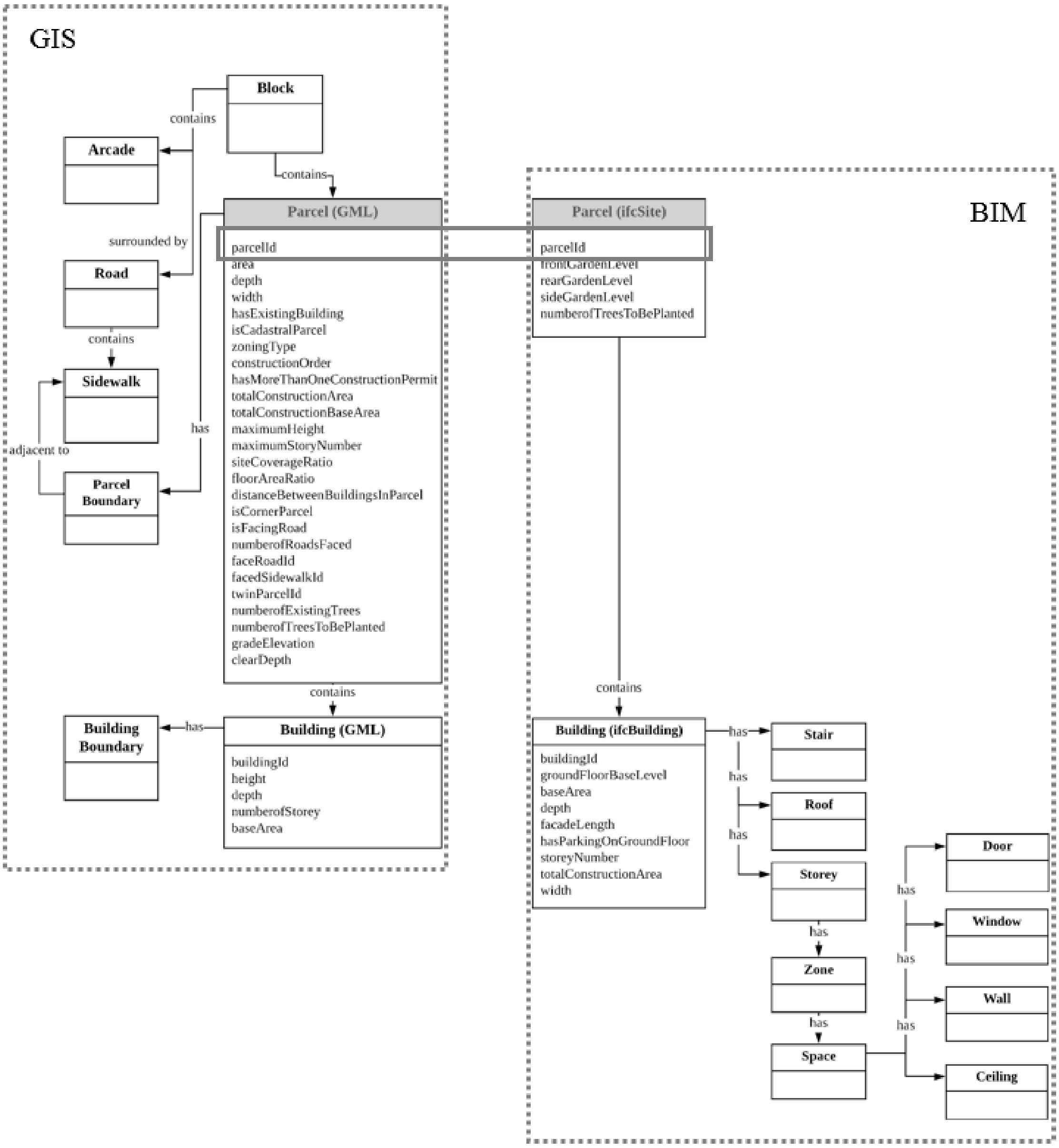


Figure 6.3. BIM and GIS data coupling.

At the end of the coupling process, parcel object in BIM and parcel object in GIS become a single parcel object that holds all the information recorded seperately in IFC and GML data models as shown in Figure 6.4.

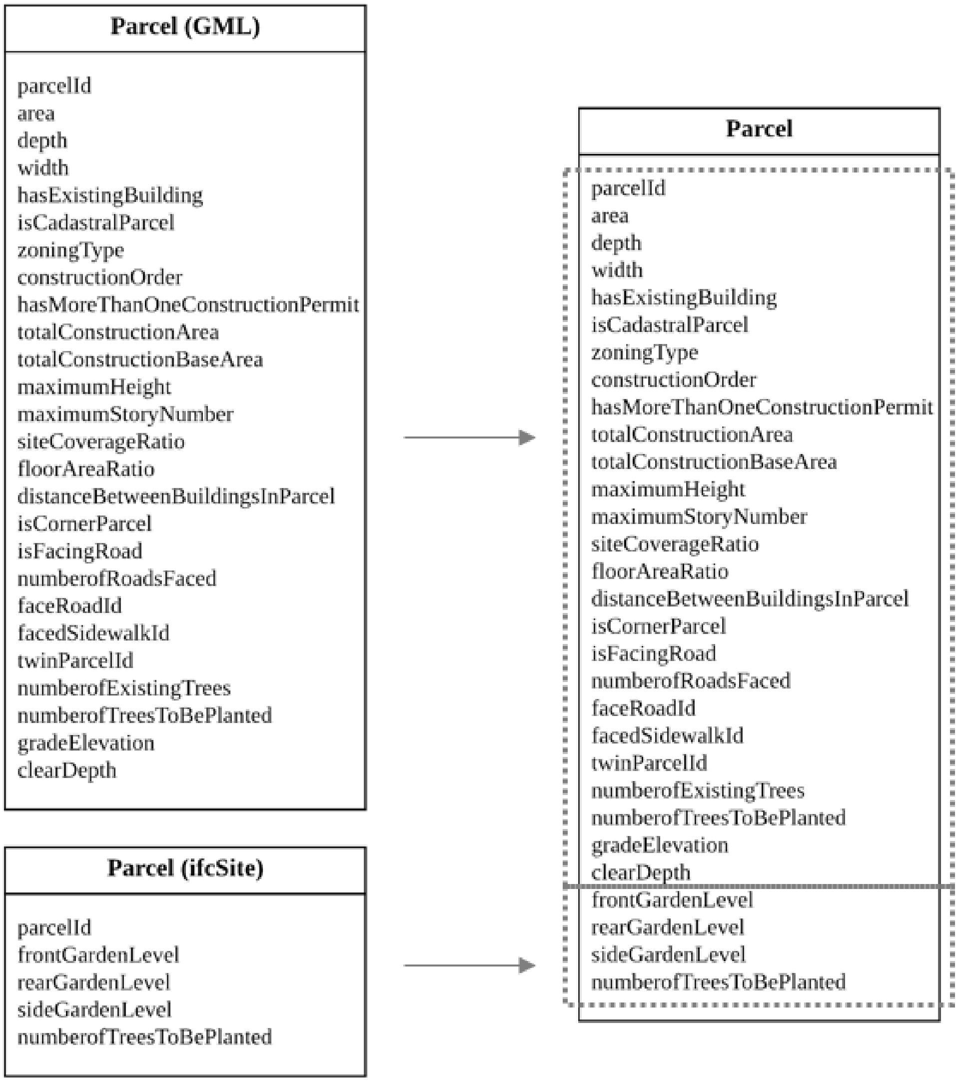


Figure 6.4. Coupling of BIM and GIS parcel objects.

Secondly, geometries are coupled. A Java algorithm is developed that compares the parcel geometry of BIM and GIS objects to identify if the corresponding objects have the same geometry and location with each other. The algorithm will be explained in detail in section 6.1.2.2.

After the coupling process, the building in IFC is parsed and added to the buildings list in the Java application where all the other existing buildings on the surrounding parcels are also stored as shown in Figure 6.5. Thus, the final unified domain model contains all the classes and properties from both BIM and GIS domains provided in detail in Appendix C. The automated code checking application then has access to process this unified dataset - the developed domain model.

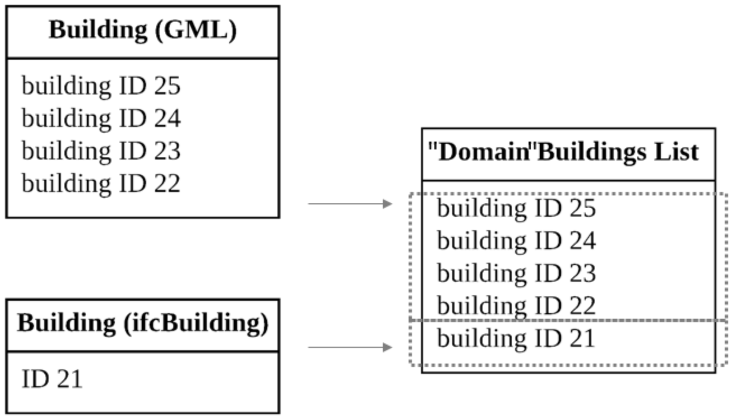


Figure 6.5. Integration of ifcBuilding to GML dataset.

6.1.2.2. Geometric Reasoning

Geometric reasoning is required of the prototype for creating some of the transient data that rules require for code checking. This is carried out during preprocessing after all data is read and before code checking starts.

Geometric reasoning enables analysis of geometric aspects of physical objects that are structured directly in the BIM and GIS data models. The prototype identifies the location of objects, queries relations and calculates distances between objects by parsing coordinates and dimensions of domain objects. For example, parsing the coordinates of the parcel geometry in BIM and the parcel geometry in GIS is crucial for coupling the two geometries. At the beginning of the design process, the architect should retrieve the parcel information from GIS and design the building by remaining within the boundaries of the setback distances. At the end of the design process, the parcel geometry submitted by the architect must match the GIS data before any code checking process takes place.

This is to prevent unintended transformations that may have been applied to the parcel in the BIM environment. As shown in Figure 6.6, the position may have changed because of translation; size may have changed because of scaling and orientation may have changed because of rotation.

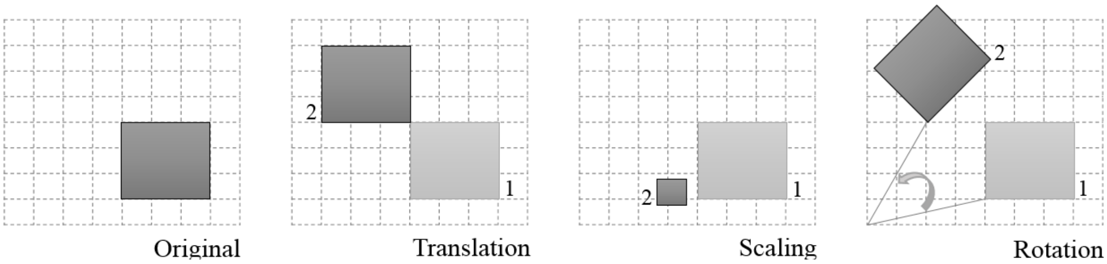


Figure 6.6. 2-dimensional transformation processes.

BIM and GIS environments rely on different coordinate systems. GIS usually uses a world coordinate system that uses longitude and latitude for positioning the object in the real world horizontally and vertically with reference to the center of Earth (Ghafourian and Karimi 2010). BIM tools often use local coordinate system for locating objects and the objects are located with regard to each other. For example, the local placement system of a wall refers to a building's local placement system (Deng 2015). This is because the focus is to design architectural components one by one at the scale of a single building. However, the lacking geographical reference information of BIM data causes problems in exchange with geospatial information systems (Ohori et al. 2017; Lapierre and Cote 2007).

Even though BIM uses a local coordinate system, IFC standard allows storing geographic coordinate information with RefLatitude, RefLongitude and RefElevation attributes of the ifcSite class. IfcSite identifies an area of land where the project will be constructed. RefLatitude attribute is the world latitude coordinate with respect to the world geodetic system WGS84. RefLongitude attribute is the world longitude coordinate with respect to the world geodetic system WGS84. RefElevation attribute is the datum elevation relative to sea level (buildingSMART 2016a). In addition to longitude, latitude and height information, the true north definition of the project can also be obtained from IfcGeometricRepresentationContext attribute of the IfcProject entity.

RefLatitude, RefLongitude, RefElevation and TrueNorth attributes provided by IFC standard provides a geographic placement of an object in relation to the real world. However, exact georeferencing is not crucial for code checking. What is crucial is the geometry of the site. The two sites, ifcSite from BIM and parcel from GIS must have identical ID's and geometries in order to make sure the code checking for various distances is accurate. Building designers should start the design process importing parcel information they receive from GIS into their BIM environments. They should be careful to avoid deformations in site boundaries. At the beginning of the checking process, for ensuring the parcel geometry in BIM sent by the architect matches the parcel geometry in GIS, what is needed is, extracting parcel geometry coordinates from GML and IFC files, proving that they are the same geometries and coupling these two data sets. This is fundamental for the reliability of an automated code checking process that uses both building data and geographical data.

For the study, the Java application parses the geometry of the parcel in BIM model and the parcel in GIS model for obtaining the x, y coordinate pairs of the parcel's corner points. Then, the application brings the two parcel geometries extracted from BIM and GIS into the common Java environment to be checked if they overlap for the accuracy of the design and the checking process. For identifying if there is scaling, the application calculates the growth rates of the length of the boundary lines (6.1); calculates the distances between coordinate points to identify if there is translation (6.2) and then draws the translated points to origin to find the rotation angle about the origin (6.3).

The new point: (x', y')

$$x' = x \cdot S_x \qquad (6.1)$$
$$y' = y \cdot S_y$$

$$x' = x + t_x \qquad (6.2)$$
$$y' = y + t_y$$

$$x' = x\cos(\theta) - y\sin(\theta) \qquad (6.3)$$
$$y' = y\cos(\theta) + x\sin(\theta)$$

Therefore, coupling and matching the data is the first crucial step before starting the automated compliance checking process. Figure 6.7 shows an example where the architect applies no two-dimensional transformation to the parcel geometry during design process.

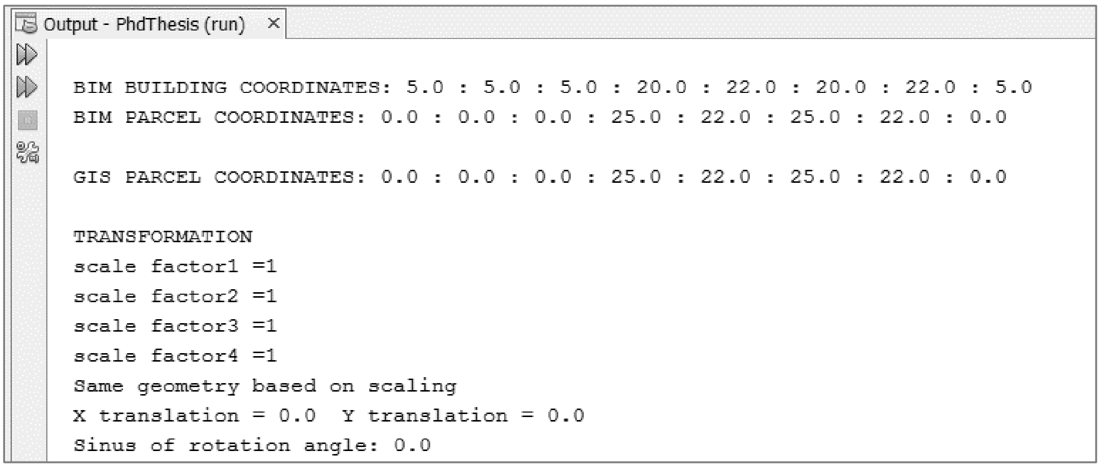


Figure 6.7. A screenshot of the application result.

The properties that do not exist in BIM but have to be checked are also studied with geometric reasoning which uses geometric aspects of physical objects including dimensions and locations to identify relationships. A code is developed in Java that queries spatial relations that are structured directly in product data models. For example, setback distance property does not exist in BIM. However, it is a property that needs to be checked based on Izmir Municipality Housing and Zoning Code to get building permission. Therefore, the prototype after parsing the input files, during preprocessing, calculates setback distances using coordinates, dimensions and locations of building and parcel objects and assigns the calculated values to domain objects' transient properties to be used during the checking process.

As an example, setback distances are calculated using the geometric formula (6.4). That distance formula is structured in Java environment for finding the perpendicular distance of each boundary line to each building point as shown in Figure 6.8.

Distance: (6.4)

$$\frac{\text{Math.abs}(((y2 - y1) * x3 + (x1 - x2) * y3 + (x2 * y2 + y1 * x2 - x2 * y2 - x1 * y2)))}{\text{Math.sqrt}((y2 - y1) * (y2 - y1) + (x1 - x2) * (x1 - x2))}$$

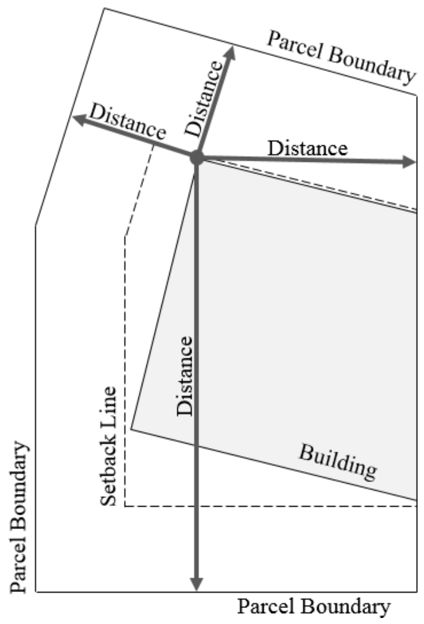


Figure 6.8. The distance formula applied to each parcel boundary line.

For the calculation process, the prototype calculates distances from all building corners to each parcel boundary and assigns the minimum distance for each parcel boundary as the designed setback distance for that boundary line since setbacks are defined as parallel distances. If the designed setback is less than the setback distance required by code for that boundary, building exceeds the limits of the setback distance in that parcel and the checker results fail.

The developed prototype in this thesis is a only a proof-of-concept for BIM and GIS data coupling but shows that geometric reasoning is required for most code checking applications.

6.1.2.3. Prototype Interface

The prototype developed in this thesis enables the user to select and open the GML and IFC files separately, one after another. From the interface, the user clicks to “Open” button and selects the GML file that he/she will, in future, obtain from the municipality. The GML file includes the information of the construction site (parcel) with its surroundings including the adjacent parcels, the block where the parcel is located, the buildings on the adjacent parcels, the surrounding roads and adjacent sidewalks. After opening the GML file, the user selects and opens the IFC file that includes information on the building design. When a BIM file is opened, IFC data model and the classes and properties of the BIM model are seen on the interface. When a GIS file is opened, GML data model and the classes and properties of the GIS model are seen on the interface. Classes and properties of both files are seen under domain model component of the system as shown in Figure 6.9.

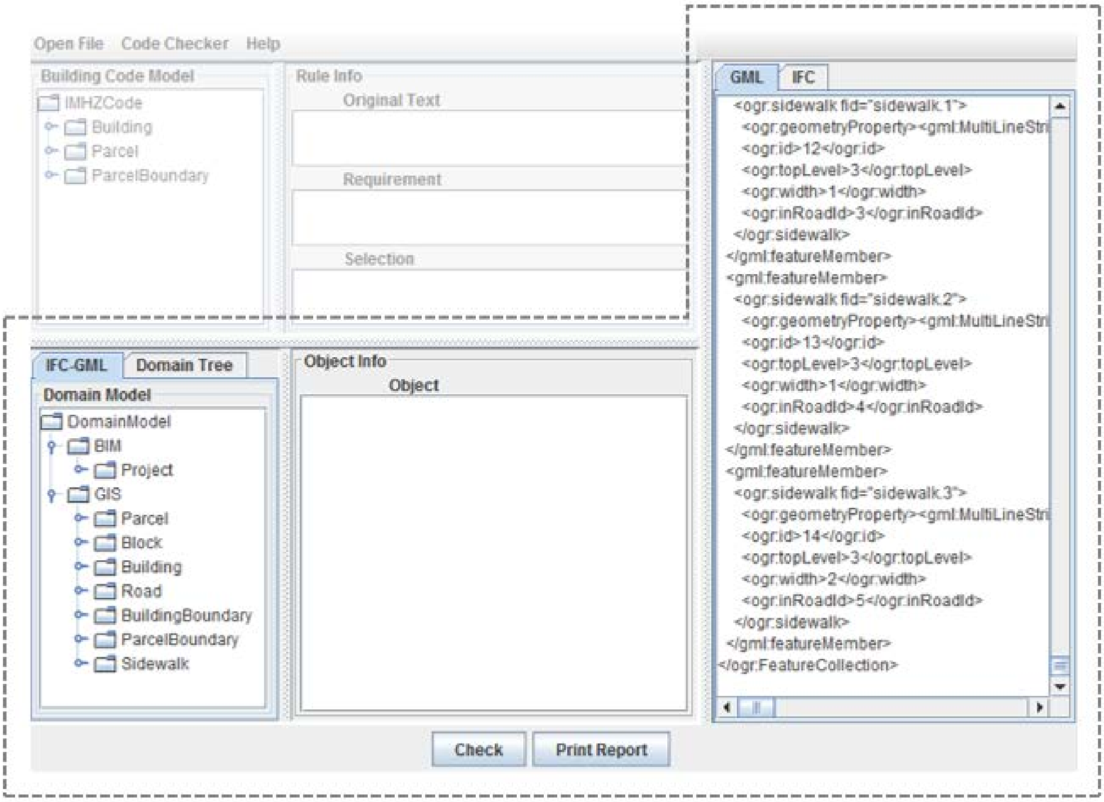


Figure 6.9. BIM and GIS data on the prototype screen.

The system also enables to select an object from a class and bring the textual information related to that object on screen. For example, if parcel number 60 is selected, its defined attributes are displayed on the interface as shown in Figure 6.10.

As well as the objects coming from BIM and GIS data models are identified separately in separate domain models; they are also presented as a unified dataset under the GIS domain model. As the dissertation focuses on developing a parcel based domain model, BIM data is integrated into GIS data. To identify the coupled parcel object that exists in both BIM and GIS, and the BIM-building object that is added to the existing buildings array in GIS, coloring is used. That parcel and building are marked with a different color on the interface as also shown in Figure 6.10.

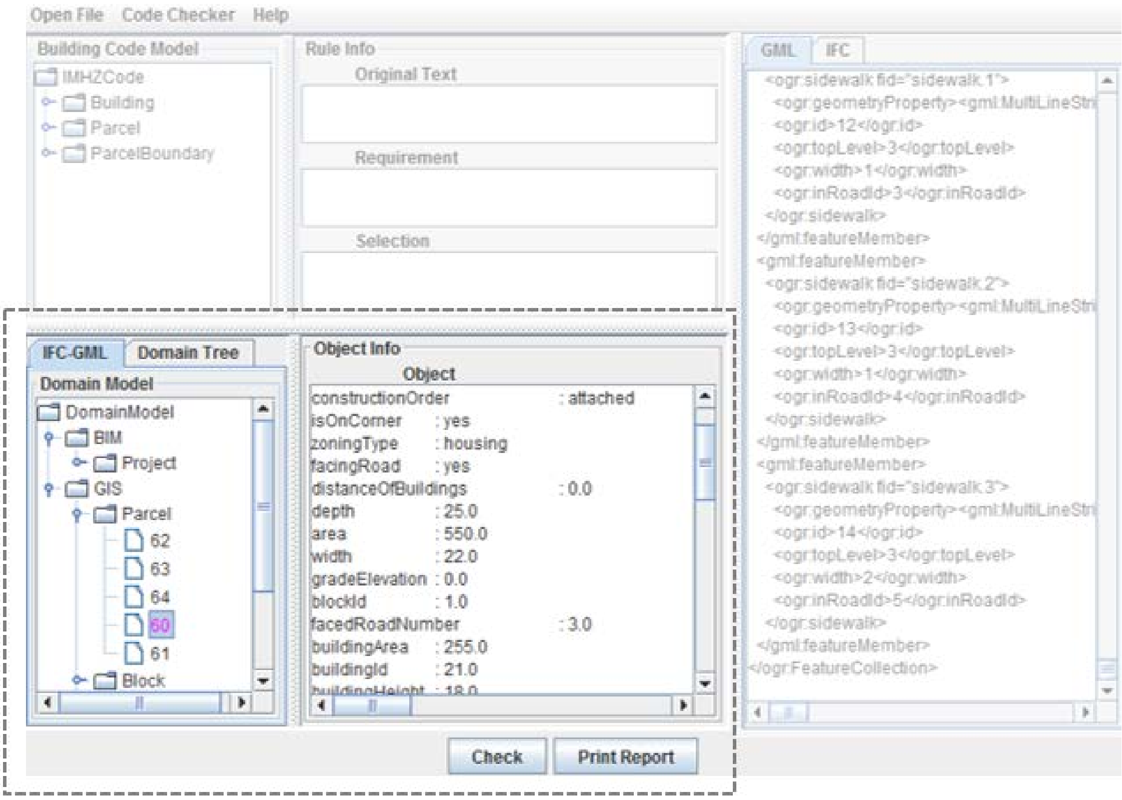


Figure 6.10. Prototype interface displaying the coupled parcel with its attributes.

In addition to presenting the BIM and GIS objects with their properties, they are also presented by their topological relationships under domain tree component of the prototype as shown in Figure 6.11. Under the domain tree, the parcels a block contains, the buildings a parcel contains, the roads and sidewalks a parcel is adjacent to and the boundaries of the parcels and blocks are organized as a tree structure.

The prototype interface also brings the Izmir Municipality Housing and Zoning Code regulations on screen and enables user to select among the rules, view the selected rule information and identify which rule is related to which domain object. If front setback is to be checked, 27th clause under frontBoundary class is parsed from the Access database for the checking process and all the information related to that rule can be seen on the interface as shown in Figure 6.12.

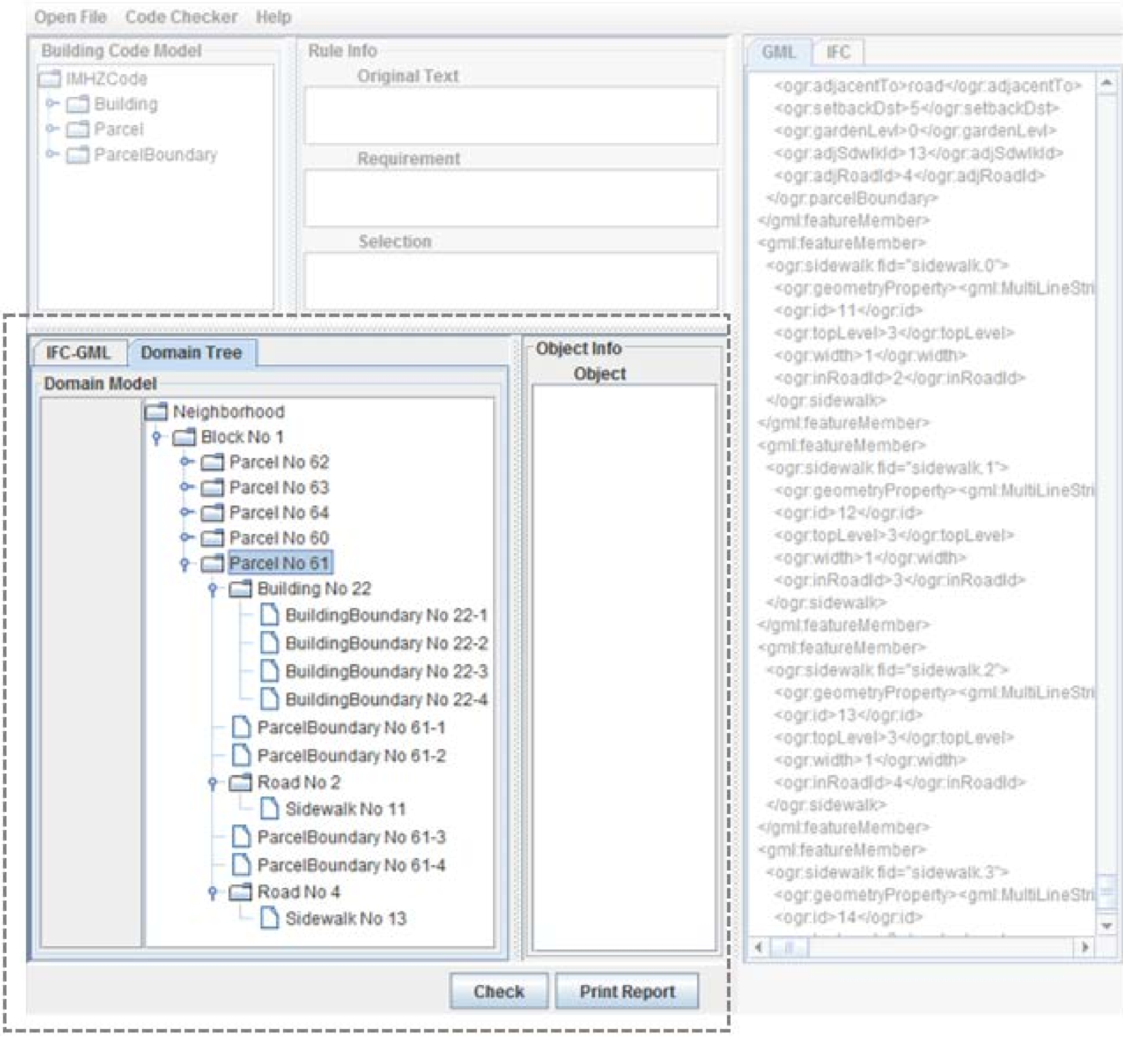


Figure 6.11. Tree structure of the unified domain model.

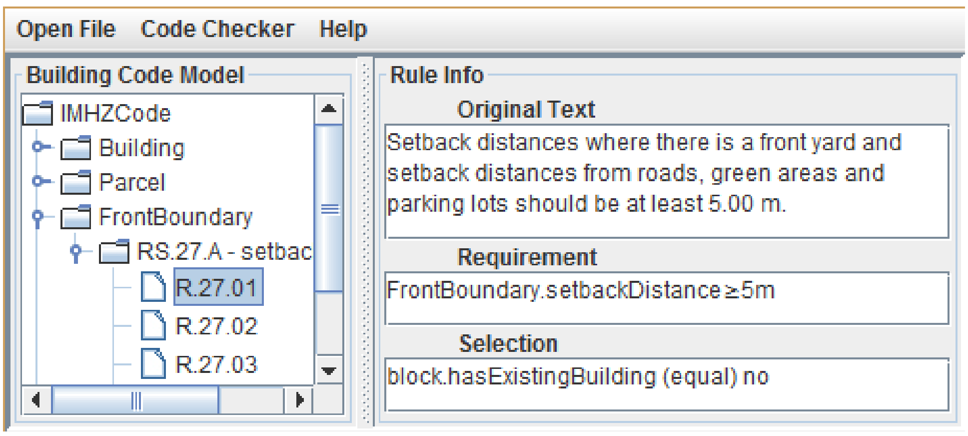


Figure 6.12. Building Code Model section of the prototype system.

6.2. Testing and Validation

This research aims to answer two questions: (1) how can the data on the surrounding urban context required for automated compliance checking be modelled and stored (2) how can data integration of BIM and GIS be achieved? To answer these questions, a domain model is developed and a GIS application based on this model is developed, a sample BIM file is prepared and a Java based digital platform is developed that extracts data from BIM and GIS seperately, and integrates and stores them in a unified domain model representation. The testing of this prototype is carried out making use of use-case scenarios demonstrating how automated compliance checking processes are carried out with various levels of complexity. The use case scenarios are specifically designed for evaluation and validation of the BIM-GIS unified model.

The use-case scenarios look for correctness in checking results. Correctness is tested by varying the conditions within the context gradually. The goal is to eliminate false negative and false positive results and ensure that the unified domain objects are able to represent the missing data on the surroundings correctly. The correctness of the coupling has been evaluated by using the prototype to carry out testing and validation. Results indicate correct integration of BIM and GIS data.

6.2.1. Use Case Scenarios

Managing and checking the coupled BIM-GIS data with Izmir Municipality Housing and Zoning Code rules in a compliance-checking prototype is illustrated in this section. The building from BIM, coupled with its data on the surroundings from GIS are checked by a set of checking scenarios using the developed prototype. This proof of concept prototype demonstrates the feasibility and applicability of the proposed unified BIM-GIS model through use-cases.

Use cases are organized considering the possible behaviors of the system as a response to the potential circumstances. The data on the surroundings and its variation is considered in the use cases, as it effects both the checking results and the checking process itself. Thus, the importance of considering environmental data during architectural design and code compliance checking processes is emphasized with the use cases.

The coupled dataset is checked with Clause 27 (setbacks), 28 (depth of buildings), 37 (garden levels) and 38 (ground floor levels) of Izmir Municipality Housing and Zoning Code document to show how checking results vary based on the data coming from the data on the surroundings. 27th rule is modelled as it requires front, side and rear garden distance information for that parcel from the GIS file. If there are existing buildings within the block, the prototype also identifies the setback distances of the adjacent parcels facing the same road within the same block to be used during the checking process as references. 28th rule is modelled as if there are existing buildings within the block, building depths in the surrounding parcels facing the same road within the same block are needed from GIS for checking the clause. 37th and 38th clauses are modelled, as they require the data of the sidewalk that faces the parcel from GIS. If the parcel faces more than one road, the prototype firstly identifies the widest road and then identifies the sidewalk that faces that road to be used during the checking process.

Five use case scenarios are developed and presented in order to demonstrate the processes (associated with the various neighborhood data) in a detailed manner. Each use case comes with a use case scenario that aims to determine the possible response of the system and the changing results in given situations. For all scenarios, the IFC file of the sample building project and different GML file alternatives that correspond to different conditions are imported to the prototype for automated compliance checking of coupled data against building codes.

For demonstration, the use cases are organized based on the most complex clauses of IMHZCode that require data on the surroundings from GIS environment for checking the building project. The clauses that require only building data are out of scope for the study. The existence of specialized conditions, exceptions, and requirements based on spatial data or topological relations during the checking process are the major reasons that increase the level of complexity in clauses of IMHZCode. Clause 27 (setbacks) and clause 28 (building depth) cover rules on setbacks and building depth and are the most complex clauses in IMHZCode. Hence, the second case and third case are organized around these clauses. They are also representative of the clauses that identify the specialized conditions in case of any exceptions. All the exception statements contained in 27th and 28th clauses are modeled individually in the Access database and checked in use cases. The checking of 27th and 28th clause also require data from adjacent parcels in the same block facing the same road, thus requires the query of topological relations of domain objects. Fourth case and fifth cases are organized and checked with clause 37 (garden levelling) and clause 38 (ground floor levelling) since rules related to levelling requires another topological query of the data on the surroundings. In case of the specialized condition that if the parcel faces two or more roads, the widest road that faces the parcel should be queried and used as a reference for the checking process.

6.2.1.1. The Scenario 1 – Matching Parcel Geometries

For the reliability of an automated code checking process and coupling the two data sets, first step should be to prove that the parcel geometry used by the architect matches the GIS data. For demonstration, two identical parcels were used for the first maching process. When the architect selects the GML file and IFC file for the automated code checking process, the application gives a message about whether the two parcels match or not. Thus, for the first condition where the two parcels are identical (Figure 6.13), the application brings a message on screen that states, "BIM parcel size and location is correct" (Figure 6.14).

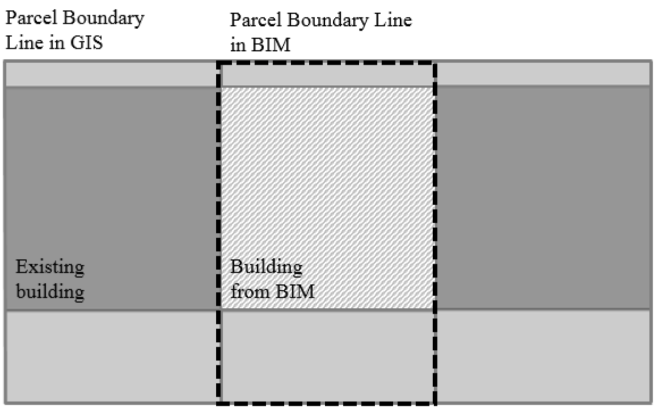


Figure 6.13. The condition where BIM parcel and GIS parcel match.

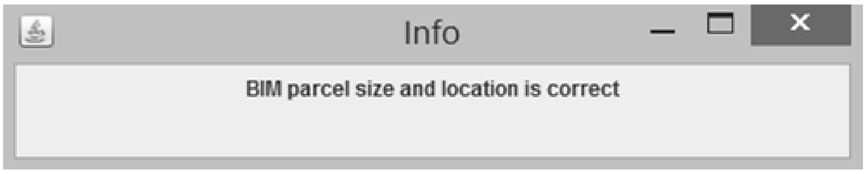


Figure 6.14. The message for matching parcels.

In the second condition as shown in Figure 6.15, the architect moved one edge of the parcel in BIM while conducting a move operation during architectural design process and increased the area of the parcel by accident. When the user runs the program, the application brings a message on screen and that message states" BIM parcel size and location is not correct!” as shown in Figure 6.16. In such a condition when the parcels do not match, the architect should run the system over again and upload the correct file for conducting an accurate code checking process.

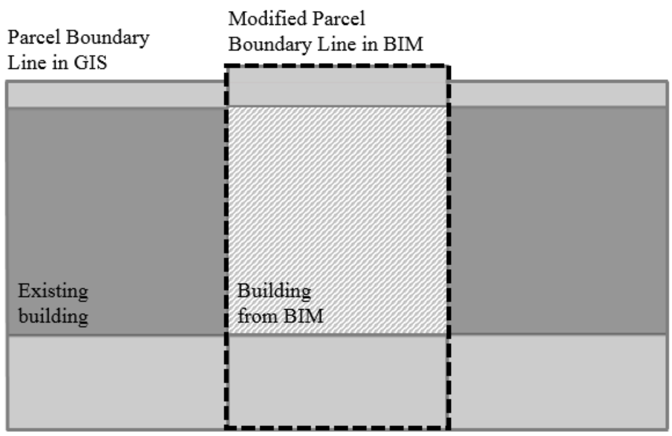


Figure 6.15. The condition where BIM parcel and GIS parcel do not match.

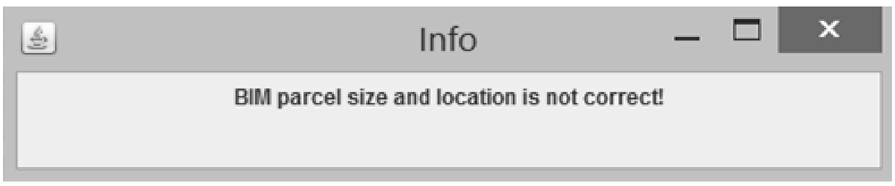


Figure 6.16. The message for the non-matching parcels.

Therefore, coupling and matching the data should be the first crucial step before starting the automated compliance checking process.

6.2.1.2. The Scenario 2 – Setbacks

Considering the data on the surroundings during architectural design process is crucial as it effects both the checking results and the checking process itself. In some cases, there can be existing buildings within the block where the building site is located. In those cases, exception rules are valid and the prototype identifies the existing buildings within the block for getting the required information for the code checking process. For demonstration, clause 27 (setbacks) have been selected as the rule on setbacks is one of the most complex clauses in İzmir Building Code in addition to its requirement of data on the surroundings from GIS during the checking process.

Clause 27 requires allowed front, side and rear setback information for the parcel from the GIS file. If there are existing buildings within the block, the prototype also identifies the setback distances of the adjacent parcels within the same block to be used during the checking process as references. Ruleset RS.27 includes three sections that indicate the conditions for determining setback distances for a parcel. First section is for front setback distance and it indicates that front setback distances should be at least 5.00 meters. 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. Second section is for side setback distance and it indicates that side setback distances should be 3.00 meters. Third section is for rear setback distance and it indicates that rear setback distances should be half of the building height. 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. Thus, 27th clause requires setback distances defined for that parcel and existing setback distances applied for the surrounding parcels from GIS for the checking process.

For the case, the building is located in a parcel whose construction order is attached. The architect planned the front setback distance as 3.00 meters since there are existing buildings in the surrounding parcels having front setback distances of 3.00 meters and the new building is subject to the existing established building line (Figure 6.17). Thus based on the rule, “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,” the project passes RS.27 with the defined dataset.

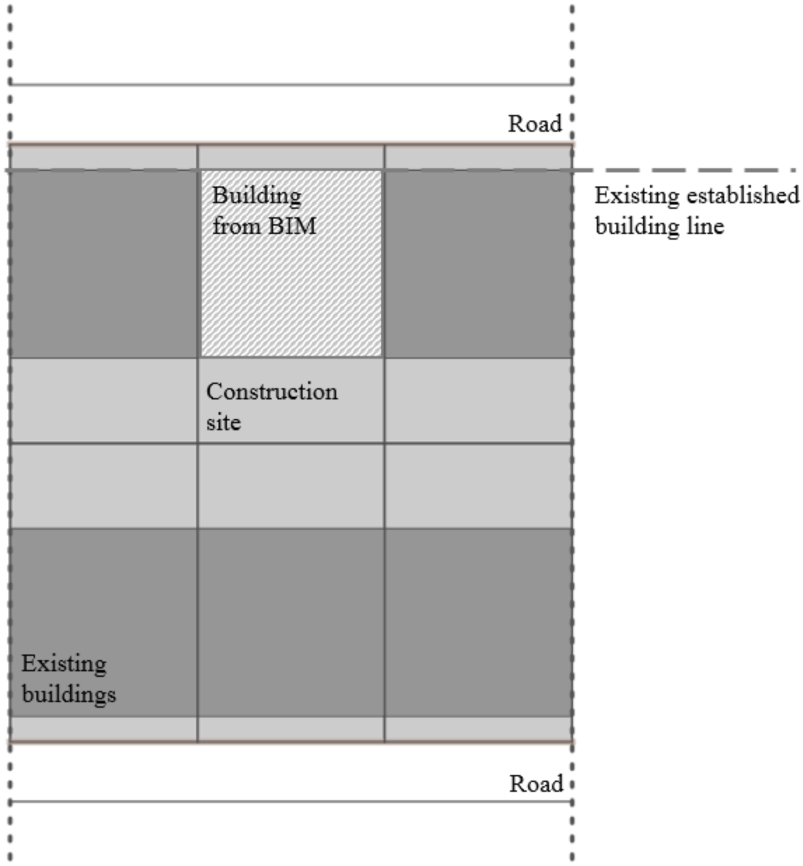


Figure 6.17. First condition where there are existing buildings on surroundings.

However if there are no existing buildings in the surrounding parcels as shown in Figure 6.18, the condition defined for setbacks changes. Ruleset RS.27 indicates that if there are no existing buildings in the block, front setback distances should be at least 5.00 meters. Thus, the 27th clause fails in second condition since the design does not meet the setback distance requirements.

Ruleset 28 (depth of buildings) is also considered for scenario 2 as building depth is related to the allowable setbacks defined for that parcel. As the building depth is defined as 22.00 meters for scenario 2, the dataset passes the check on Ruleset 28 that indicates building depths should not exceed 22.00 meters. The checking results of the scenario 2 is given in Figure 6.19.

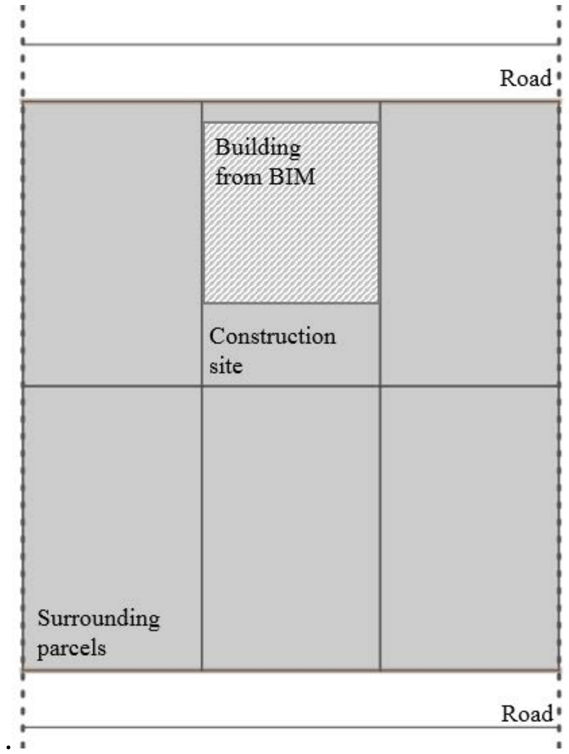


Figure 6.18. Second condition where there are no existing buildings on surroundings.

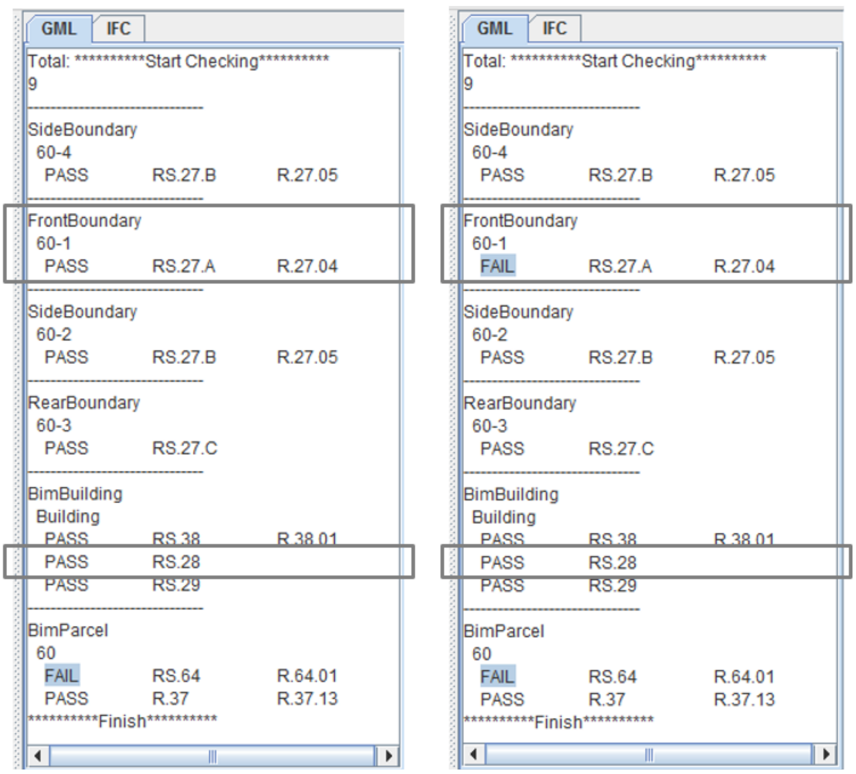


Figure 6.19. The checking result for scenario 2 a) There are existing buildings in block b) There are no existing buildings in block.

6.2.1.3. The Scenario 3 – Construction Order

Clause 27 (setbacks) of Izmir Municipality Housing and Zoning Code document is used again as a demonstrative example in the third scenario. Different from the second scenario, this time the importance of considering construction order for the application of Clause 27 is studied. Ruleset RS.27 includes three sections. Each section indicates the rules for determining setback distances for a parcel and the applicability of each rule is dependent on the construction order of the block where the project will be built. For example, first section is for front setback distance that indicates front setback distances should be at least 5.00 meters, and third section is for rear setback distance that indicates rear setback distances should be half of the building height. In semi-detached ordered blocks, if there is an existing building in one of the two parcels then front/rear setback distances should be the same as front/rear setback distances used in that parcel. Or, in attached ordered blocks, if there is an existing building in any of the parcels within the same block then front/rear setback distances should be the same as front/rear setback distances used in that block.

For the case, the building is located in a parcel whose construction order is attached. Based on the existing buildings, the architect planned 3.00 meters front setback distance, 10.00 meters rear setback distance and no side setback distances as shown in Figure 6.20. As the parcel has a construction order, which is attached, the rule about side setback distances do not apply to this project and the project passes RS.27 with the defined dataset.

However, if construction order of the block is changed into detached, a problem occurs about side setback distances. As there are no defined side setback distances in attached ordered parcels, the building should have setback distances from side parcel boundaries in detached ordered parcels as shown in Figure 6.21. Thus, the project fails 28th clause in the second condition since the design does not meet the side setback requirement. The checking results of the scenario 3 is given in Figure 6.22.

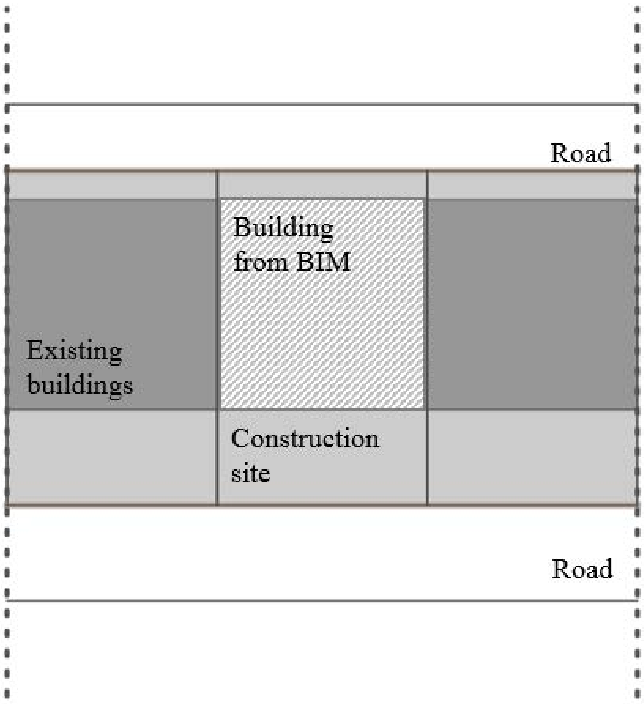


Figure 6.20. First condition where the construction order is attached.

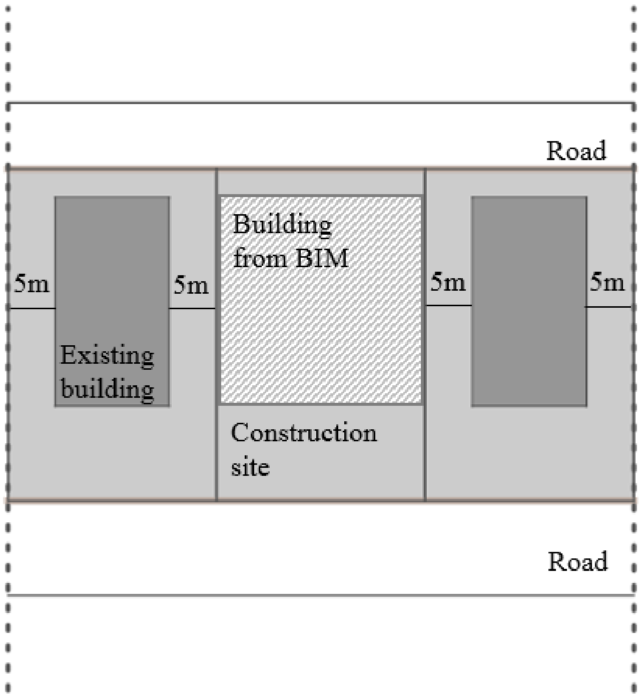


Figure 6.21. Second condition where the construction order is detached.

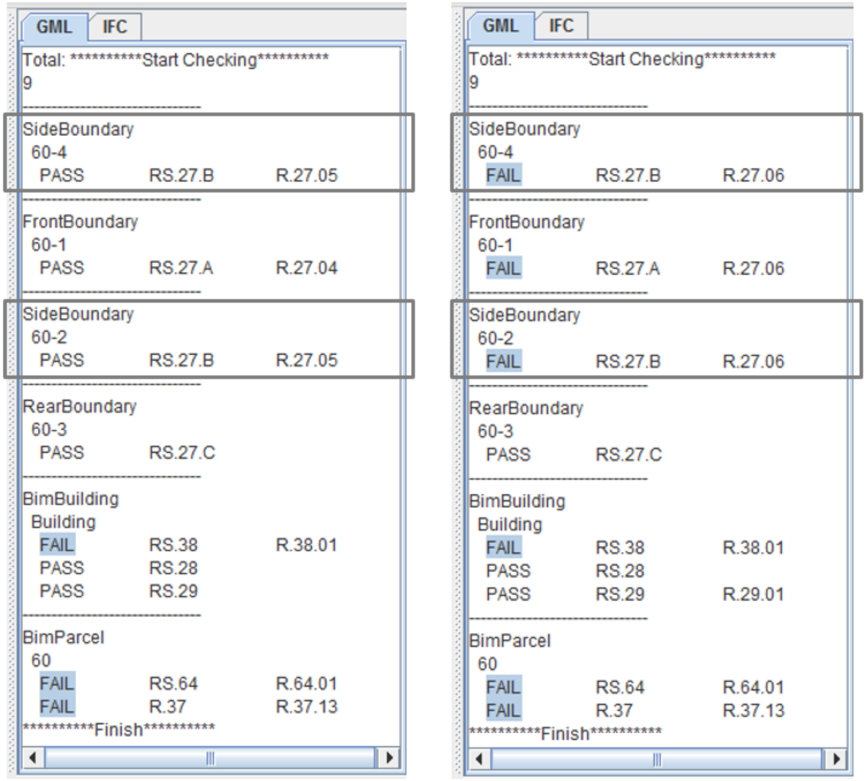


Figure 6.22. The checking result for scenario 3 a) The construction order is attached b) The construction order is detached.

6.2.1.4. The Scenario 4 – Garden Levels

Ruleset RS.37 includes rules that indicate the conditions for determining elevations of buildings and their gardens. The third part of the 37th rule, which is section C, indicates three conditions for garden levelling. First one is for front garden levelling and it indicates that front garden level should be levelled by taking the top level of the sidewalk that the parcel faces as a reference.

For the case, the architect defines the front garden level as +3.00 meters, while top-level property of the sidewalk that faces the parcel is also defined as +3.00 meters in the GIS file. As the front garden level is equal to the sidewalk level, 37th clause passes with the defined dataset. However if the sidewalk level had a different value, the dataset would fail with the 37th clause since the design does not meet the levelling requirement for front garden. The two conditions are shown in Figure 6.23 and the checking results of both conditions for the scenario 4 is given in Figure 6.24.

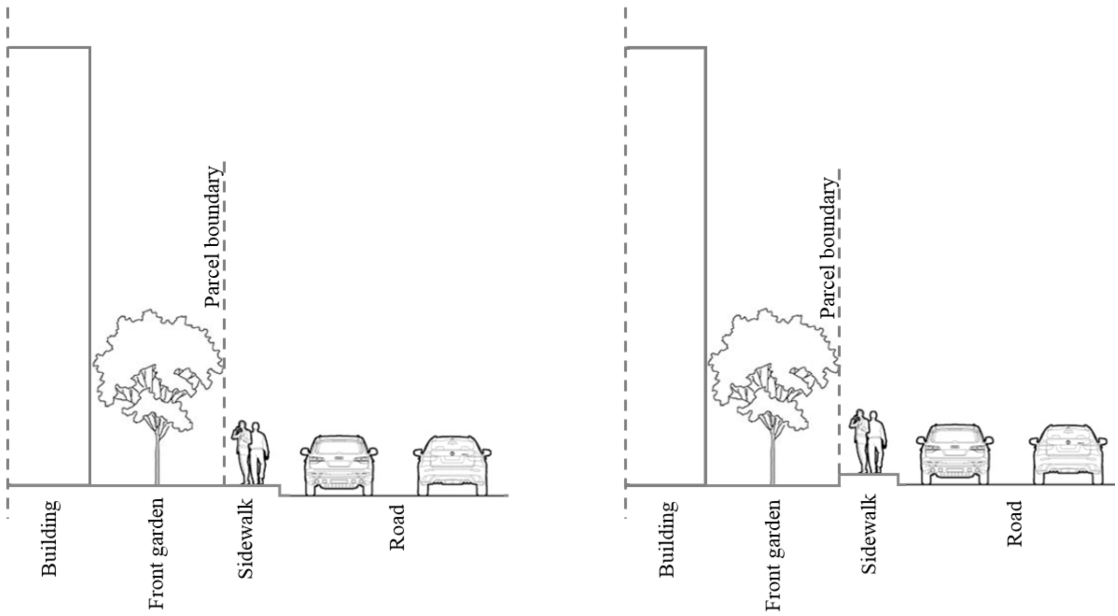


Figure 6.23. Two conditions showing different front garden levels.

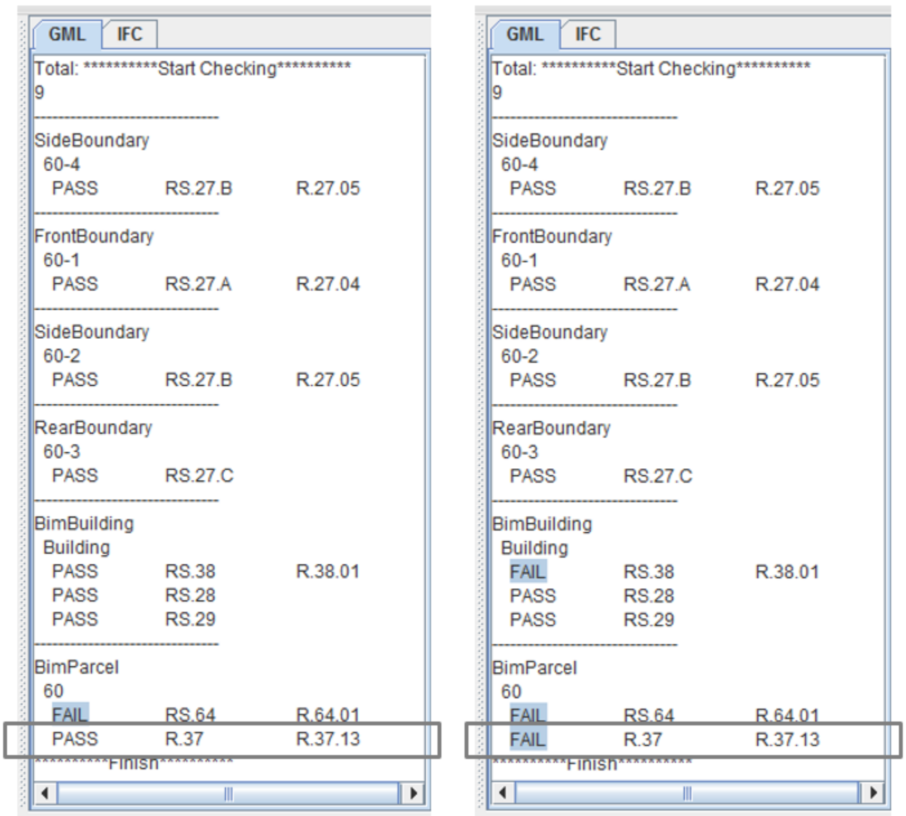


Figure 6.24. The checking result for scenario 4 a) Front garden level is equal to sidewalk level b) Front garden level is not equal to sidewalk level.

6.2.1.5. The Scenario 5 – Ground Floor Levels

Clause 38 (ground floor levels) of Izmir Municipality Housing and Zoning Code document is used as demonstrative example for the scenario. The rule is selected for the checking process, as it requires the data of the sidewalk that faces the parcel from GIS. If the parcel faces more than one road, the prototype firstly identifies the widest road and then identifies the sidewalk that faces that road to be used during the checking process.

Ruleset RS.38 includes rules that indicate the conditions for determining the ground floor base levels. The rule indicates that ground floor base level of a building should be planned between + 0.50 meters to +1.00 meters from the upper level of the sidewalk. For the case, the site is a middle parcel and faces only one road. As the top-level of the sidewalk that faces the parcel is defined as +2.00 meters in the GIS file, the architect planned the ground floor level of the building as +3.00 meters based on the sidewalk top level. The project passes RS.38 with the defined dataset shown in Figure 6.25.

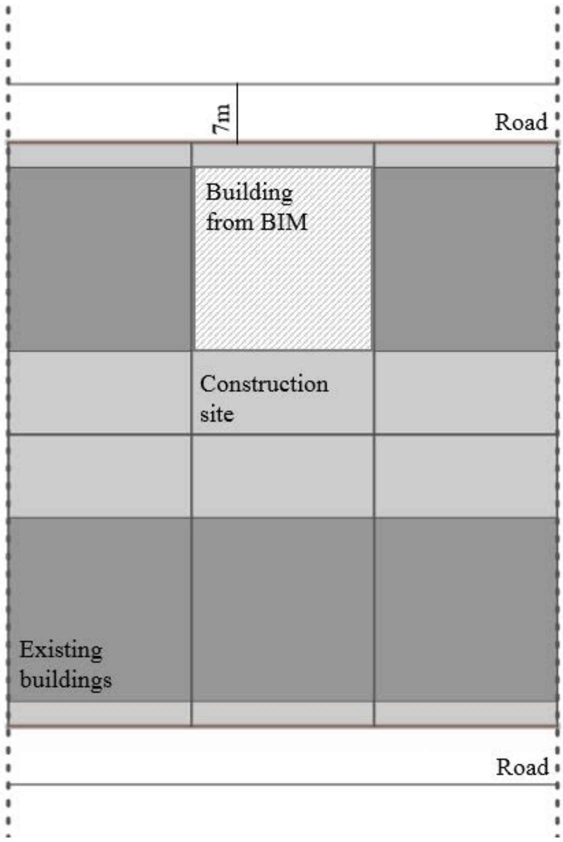


Figure 6.25. First condition where the parcel faces two roads.

However, when the surrounding context changes, the checking process with RS.38 results differently. In the second condition as shown in Figure 6.26, the parcel faces two roads having different widths and the road that the architect should use as a reference for ground floor levelling is different from the previous condition. The top level of the sidewalk that faces the wider road is defined as +3.00 meters in the GIS file. As the architect defined ground floor level of the building as +3.00 meters, the project fails RS.38, because the ground floor level of the building should have been planned higher from the upper level of the sidewalk.

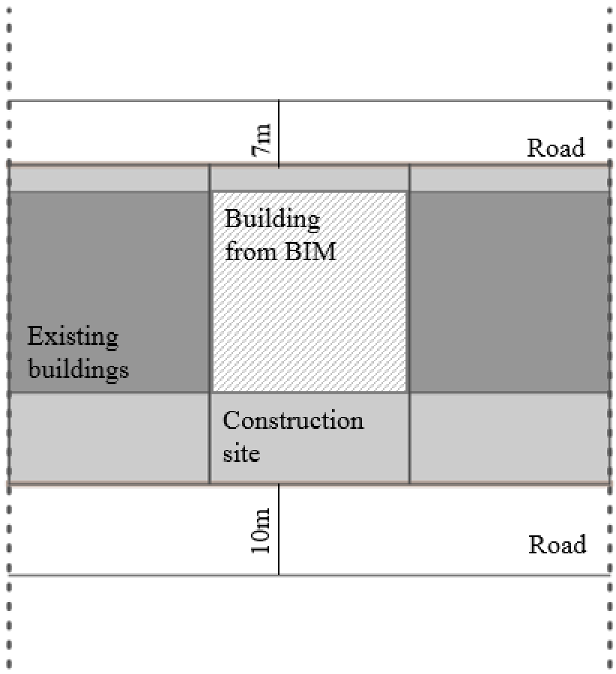


Figure 6.26. Second condition where the parcel faces two roads.

The second condition that fails the check when the parcel faces a wider road is shown using a section drawing in Figure 6.27; and the checking results of scenario 5 is given in Figure 6.28.

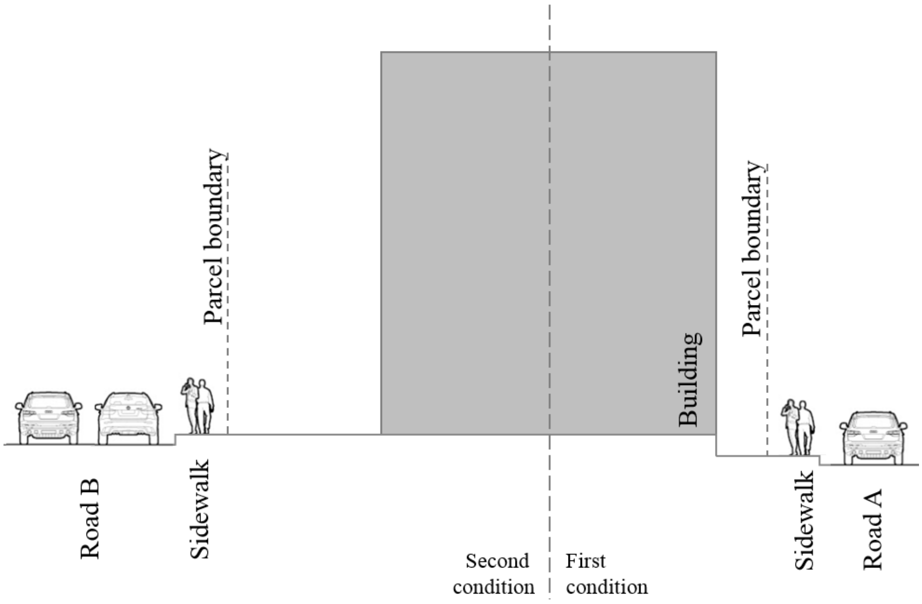


Figure 6.27. Second condition where the ground floor fails to be at least 0.50 m higher than the sidewalk facing the wider road.

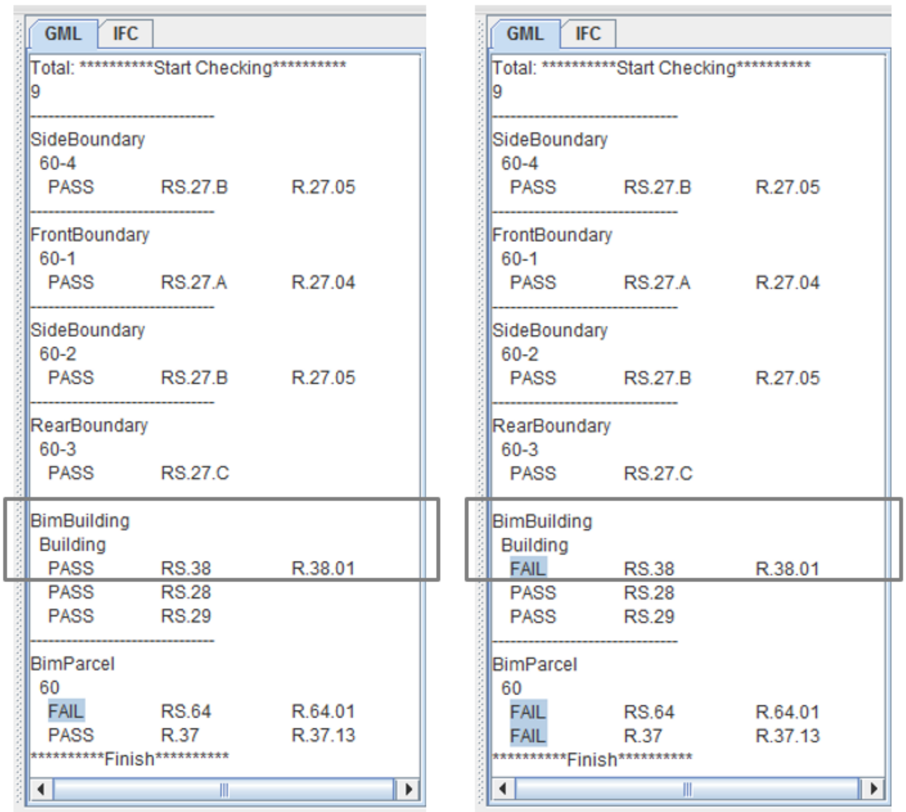


Figure 6.28. The checking result for scenario 5 a) The parcel facing one road b) The parcel facing two roads.

CHAPTER 7

CONCLUSION

In this thesis, a methodology for BIM and GIS data integration is described and applied in an automated code checking context. Through studying and reviewing different code checking studies, it is found that the integration of BIM and GIS data can be very beneficial for facilitating automated code checking, as BIM model does not contain the required neighborhood data by itself. Therefore, this study firstly identified the neighborhood data mentioned in building regulations required for automated code compliance checking of building projects, resulting in a list of zoning concepts mentioned in building regulations, which constituted the answer for the first research question: "What are the zoning concepts mentioned in building regulations?" Then, to answer the question of, “How much of the identified zoning concepts are missing in BIM?” a mapping was defined between the IFC model as a BIM representation and the identified concepts. The analysis results showed that %70 of the data (48 of 69 terms) on the surroundings that was identified with IMHZ code analysis, are not covered by the IFC data model. A follow-up analysis showed that they can be modelled using GIS. Thereafter, the crux of the study focused on developing a domain model for the representation of the identified missing zoning data required for a automated code checking processes, in search of answers for the third and fourth research questions: "How can zoning data required for automated code checking be modelled in GIS?", and "How can topological relationships between zoning concepts be handled in GIS?" Here, first, an object-oriented model for the zoning concepts was developed that could be implemented using GIS constructs. Then, this domain model was implemented in QGIS. The classes and attributes became layers and properties while topological relationships were structured using spatial queries. It was shown that GIS is able to adequately represent zoning concepts relating to the surroundings of a building project within the scope of IMHZCode. Finally, the last two research questions on BIM-GIS integration were explored: "How can IFC and GML data be integrated in a domain model?", and "How can unified data be used in code checking?" For this purpose, a running prototype was developed that conducts automated code compliance checking. The GIS data was combined with BIM data using a semantic mapping approach in a unified environment independent of any BIM-GIS platform or compliance checking system. The prototype populates the domain model by only extracting what is necessary from both the IFC data from BIM systems, and the GML data from the QGIS application. The rules that also exist independently, are then used to check the project's compliance with IMHZCode. Development of an independent environment does not require any data conversion or extension process. It aims to avoid any data loss, incomplete and unreliable transformations of information, incorrect mapping, and increase in information storage capability of the two environments during integration.

The reseach conducted for the thesis was of an interdisciplinary type in which various working areas were involved, including building information modeling, geographic information systems, interoperability, building regulations, automated compliance checking systems etc. Goal of this study was to 1) determine the data on the surrounding urban context required for automated compliance checking and 2) propose a solution to how data from GIS can be integrated with BIM. While conducting the research, various issues were covered, such as the lack of a common city information database system, problem of keeping zoning data in the form of written texts and scanned plan documents in municipalities, the lack of required surrounding data in BIM and GIS, interoperability issues, and solutions were developed for these issues. This chapter discusses contributions of this dissertation along with directions for future work.

7.1. Contributions

While carrying out the work, several key contributions to knowledge were made in the following areas:

- Development and implementation of a methodology and defining the steps for integration of BIM and GIS data for automated code compliance checking.
- Analysis and organization of building code data of İzmir for identifying any form of neighborhood data that is missing in BIM and defining a model for future GIS applications supporting the automated compliance checking against Izmir Municipality Housing and Zoning Code.
- Showing that GIS with its topological analysis functions is adequate to model necessary neighborhood data, including classes, attributes and the relations required for a complete automated code checking process against Izmir Municipality Housing and Zoning Code.

- Demonstrating the effectiveness of the data integration approach where independent applications draw data separately from both BIM and GIS environments.

The methodology described for BIM and GIS data integration is the first contribution of this research. Although the methodology is not tested for domains other than automated code checking, it is believed that the approach can be applied to other fields of research that requires the management of BIM data and GIS data together. Regarding the data integration approach, the phases are identified and applied to automated code checking process in this thesis. The first step of the developed methodology is to identify the required data in the domain of interest and then, to identify the part of the data that resides within the BIM model and the part that resides within the GIS model. Secondly, BIM and GIS domain models should be constructed by using the identified data in the previous step. Thirdly, an application should be developed that integrates the two domain models and manages them as a whole.

Analysis and organization of building code data of İzmir for identifying any form of surrounding data that is missing in BIM is the second contribution of this research. The understanding of the types of zoning knowledge explicitly present in building codes is very important for the development of successful models for automated code checking processes. Organizing the analysed zoning data is also important for a clear presentation of the data. Developing a zoning data model provides ability to represent and manage any form of geographical information contained in building code regulations. Thus, an object-oriented data modeling framework is developed for organizing all the classes with their attributes and relations to each other. This framework is developed in response to the research question: "What types of surrounding information exist in building regulations that do not exist in BIM?" Thus, one of the major contributions of this research is a thorough analysis, specification and organization of zoning knowledge through a model formalizing spatial knowledge contained in Izmir Housing and Zoning Code.

Currently, zoning information is generally kept as written texts and scanned plan documents, which is unable to be processed digitally. Although some of the zoning information is held digitally, it is inadequate for code checking. This study demonstrates the importance of storing zoning data digitally, and introduces the dependence on traditional ways of using information from printouts and archive documents as a critical problem in building code checking procedures. It aims to improve the utilization of digital city plans modelled in geographic information systems. It is also possible to add new classes and attributes or to make changes to the model in GIS in case of updates in the building code. The GIS environment lets the user make updates. The implementation of zoning data model in GIS environment also allows zoning data to be queried with topological relations and hence, construct a data model that can be queried for a complete automated code checking process. The implementation of the zoning domain model in GIS environment precludes manual code checking processes, saves time and reduces human intervention during checking process. The developed GIS model can be used as a database to manage geographical information in municipalities. It can reduce time spent searching for paper-based information. Proving that topological analysis functions of GIS is adequate to model a complete data model including objects, attributes and relations for a complete automated code checking process is the third contribution of this research. The data modeling framework and the studied reasoning mechanisms for querying topological relations are believed to be an important piece for the future zoning data modeling studies and code compliance checking processes.

The study also aimed to increase the automation of the building permit processes by utilising BIM and GIS data together. Thus, coupling the zoning data modelled in GIS with BIM data and evaluating designs for conformance to governing design standards in a digital environment is the forth contribution of the study. The thesis demonstrates the importance and effects of integrating GIS data with BIM data for automated code checking processes. In order to evaluate the applicability of the developed zoning model in GIS, and test its integration with BIM data in the area of code checking, building regulations and a building model are also modelled as code checking requires appropriate computer-based models of building codes in digital format, zoning plans in GIS and building designs in BIM environment. Thus, zoning data model in GIS is integrated with these digital models in a unique checker environment. An independent environment that draws data from both BIM and GIS environments and combining them in a unified domain model appropriate to the field is developed. The domain model manages both the surrounding data represented in the GML and the building data represented in the IFC and then becomes the base for exchanging information. Thus, uploding the BIM data in IFC format and GIS data in GML format into the checker environment and integrating BIM and GIS into a unique system enable architects to self-check designs for compliance before submitting them to municipalities and local authorities to conduct automated code compliance checking processes in digital environment.

Processing BIM and GIS in the same environment independently avoids data translations between the models, incorrect mapping and thus, loss of information. The independent nature of the coupling environment also does not increase data sets and workloads of either BIM or GIS environments. After all, it is impractical to expect BIM to explicitly define geographical information; and GIS to manage detailed building data. The integrated environment developed in this thesis can manage data coming from BIM and GIS worlds in the same structure and with the same tools. Considering the vast amount of attempts at developing conversion algorithms, this study does not address this type of problem. The study focuses on the conceptual integration of concepts defined in IFC and GML standards within a new domain-specific model. While this approach and methodology proved to be effective in the context of automated code checking, it may not be the case in every domain. However, it is proof that the approach is viable and should be considered as an option for efforts in other domains as well.

Modeling zoning data in GIS, integrating it with BIM data, and combining it with building codes in digital format, brings many advantages in the area of code checking processes in municipalities. There are various studies for developing computer-based models of building codes in digital format (Yabuki and Law 1993; Kiliccote and Garrett Jr 1998; Kerrigan and Law 2003; Eastman et al. 2009; Dimyadi and Amor 2013; Macit İlal and Günaydın 2017). Digital checking of projects, even with partial automation, simplifies the work of approving authorities, leads to faster managing and processing. The ability to superpimpose GIS data and BIM data in an application provides automated visual control, faster turnaround in feedback, faster approvals for buildings; hence, prevents errors due to manual checking and inconsistenties in the interpretation of codes.

The checker environment is developed independent of any BIM-GIS platform or compliance checking system in this thesis. It is foreseen that the developed checker platform can be integrated as a plug-in into a BIM platform in future studies and will be able to provide architects immediate feedback with regard to code compliance, even in early design phases.

7.2. Future Work

The methodology that has been developed has been successful in producing an integrated BIM-GIS data model to be used in a code compliance checking of building projects. However, there are limitations to this proof of concept. First, only two-dimensional data has been modelled and the data on the surroundings is constructed based on two-dimensional relations. While two-dimensional relations were sufficient in the case of the Izmir Building codes, three-dimensional geometric reasoning might be needed with more complicated codes. A second important limitation is the fact the methodology developed in this thesis includes a mapping process between IFC and GML data, which requires manual work. These limitations are also seen as future research directions. Considering future research possibilities, the following are proposed:

• The methodology developed in this thesis can be applied to other fields of research that requires the management of BIM data and GIS data together.

• The unified model, which holds the two-dimensional object data, can be further developed to integrate objects from 3D building and 3D city models and a 3D BIM model can be visualized in a 3D geospatial environment.

• Domain mapping is manual work. It should be investigated if mapping can be an automated process.

REFERENCES

Akin, Omer. 2010. “CAD/GIS Integration: Rationale and Challenges.” In *CAD and GIS Integration*, edited by Hassan A Karimi and Burcu Akıncı, 51–72. New York, NY: CRC Press.

Amirebrahimi, Sam, Abbas Rajabifard, Priyan Mendis, and Tuan Ngo. 2015. “A Data Model for Integrating GIS and BIM for Assessment and 3D Visualisation of Flood Damage to Building.” In *CEUR Workshop Proceedings*, edited by B. Veenendaal and A. Kealy, 1323:78–89. Brisbane, Australia.

Baumgärtel, Ken, Mathias Kadolsky, and Raimar J Scherer. 2015. “An Ontology Framework for Improving Building Energy Performance by Utilizing Energy Saving Regulations.” In *Proceedings of European Conference on Product and Process Modelling*, 519–26.

Beach, T. H., Y. Rezgui, H. Li, and T. Kasim. 2015. “A Rule-Based Semantic Approach for Automated Regulatory Compliance in the Construction Sector.” *Expert Systems with Applications* 42 (12): 5219–31.

Benner, Joachim, Andreas Geiger, and Karl-Heinz Häfele. 2010. “Concept for Building Licensing Based on Standardized 3d Geo Information.” In *5th International 3D GeoInfo Conference*, XXXVIII:9–12. Berlin, Germany.

Berlo, L. Van, T. Dijkmans, and J. Stoter. 2013. “Experiment for Integrating Dutch 3D Spatial Planning and BIM for Checking Building Permits.” In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2:279–84.

Blake, Landon. 2007. “Spatial Relationships In GIS - An Introduction.” *OSGeo Journal* 1 (May).

Booch, Grady, James Rumbaugh, and Ivar Jacobson. 1999. *Unified Modeling Language User Guide*. Addison-Wesley Longman Inc.

Borrmann, A., Thomas H. Kolbe, A. Donaubauer, H. Steuer, J. R. Jubierre, and M. Flurl. 2015. “Multi-Scale Geometric-Semantic Modeling of Shield Tunnels for GIS and BIM Applications.” *Computer-Aided Civil and Infrastructure Engineering* 30: 263–81.

Borrmann, André, Juha Hyvärinen, and Ernst Rank. 2009. “Spatial Constraints in Collaborative Design Processes.” In *Proceedings of the International Conference on Intelligent Computing in Engineering (ICE’09)*, 1–8. Berlin, Germany.

Brasebin, M., J. Perret, S. Mustière, and C. Weber. 2016. “A Generic Model to Exploit Urban Regulation Knowledge.” *ISPRS International Journal of Geo-Information* 5 (2).

Brasebin, M., J. Perret, S. Mustière, and C. Weber. 2018. “3D Urban Data to Assess Local Urban Regulation Influence.” *Computers, Environment and Urban Systems* 68 (December): 37–52.

Brasebin, M, J Perret, and C Haëck. 2011. "Towards a 3D Geographic Information System for the Exploration of Urban Rules: Application to the French Local Urban Planning Schemes." In *28th Urban Data Management Symposium*.

buildingSMART. 2016a. "IFC4 Addendum 1 – Introduction." Accessed April. http://www.buildingsmart-tech.org/ifc/IFC4/Add1/html/

buildingSMART. 2016b. "The Basic Standards." Accessed April. http://buildingsmart.org/standards/technical-vision/open-standards-101/

Casey, Michael J, and Sriharsha Vankadara. 2010. "Semantics in CAD/GIS Integration." In *CAD and GIS Integration*, edited by Hassan A Karimi and Burcu Akinci, 143–70. New York, NY: CRC Press.

Choi, Jungsik, Junho Choi, and Inhan Kim. 2014. "Development of BIM-Based Evacuation Regulation Checking System for High-Rise and Complex Buildings." *Automation in Construction* 46: 38–49.

Conover, D. 2007. "Development and Implementation of Automated Code Compliance Checking in the U.S." *International Code Council*.

Deng, Yichuan. 2015. "Mapping of BIM and GIS for Interoperable Geospatial Data Management and Analysis for the Built Environment." The Hong Kong University of Science and Technology.

Dimyadi, Johannes, and Robert Amor. 2013. "Regulatory Knowledge Representation for Automated Compliance Audit of BIM-Based Models." In *Proceedings of the 30th W78 International Conference: Move Towards Smart Buildings, Infrastructures and Cities*, 68–78. Beijing, China.

Dimyadi, Johannes, Charles Clifton, Michael Spearpoint, and Robert Amor. 2014. "Regulatory Knowledge Encoding Guidelines for Automated Compliance Audit of Building Engineering Design." In *Computing in Civil and Building Engineering*, 536–43.

Dimyadi, Johannes, Pieter Pauwels, Michael Spearpoint, Charles Clifton, and Robert Amor. 2015. "Querying a Regulatory Model for Compliant Building Design Audit." In *Proc. of the 32nd CIB W78 Conference*, 139–48. Eindhoven, The Netherlands.

Ding, Lan, Robin Drogemuller, Mike Rosenman, David Marchant, and John Gero. 2006. "Automating Code Checking for Building Designs-DesignCheck." *CRC for Construction Innovation, (Clients Driving Innovation: Moving Ideas into Practice)*, 1–16.

Donkers, Sjors, Hugo Ledoux, Junqiao Zhao, and Jantien Stoter. 2016. "Automatic Conversion of IFC Datasets to Geometrically and Semantically Correct CityGML LOD3 Buildings." *Transactions in GIS* 20 (4): 547–69.

Eastman, Chuck, Jae min Lee, Yeon suk Jeong, and Jin kook Lee. 2009. "Automatic Rule-Based Checking of Building Designs." *Automation in Construction* 18 (8): 1011–33.

Eastman, Chuck, Paul Teicholz, Rafael Sacks, and Kathleen Liston. 2011. *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors*. 2nd ed. John Wiley and Sons.

El-Diraby, Tamer, and Hesham Osman. 2010. "Ontologies for Linking CAD/GIS." In *CAD and GIS Integration*, edited by Hassan A Karimi and Burcu Akinci, 171–98. New York, NY: CRC Press.

El-Mekawy, M, A Östman, and I Hijazi. 2012a. "A Unified Building Model for 3D Urban GIS." *ISPRS International Journal of Geo-Information* 1 (3): 120–45.

El-Mekawy, M, A Östman, and I Hijazi. 2012b. "An Evaluation of IFC-CityGML Unidirectional Conversion." *International Journal of Advanced Computer Science and Applications* 3 (5): 159–71.

El-Mekawy, M, and Anders Östman. 2010. "Semantic Mapping: An Ontology Engineering Method for Integrating Building Models in IFC and CITYGML." In *Proceedings of the 3rd ISDE Digital Earth Summit*. Nessebar, Bulgaria.

Elbeltagi, Emad, and Mahmoud Dawood. 2011. "Integrated Visualized Time Control System for Repetitive Construction Projects." *Automation in Construction* 20 (7): 940–53.

Esri. 2015. "ArcGIS Data Interoperability." Accessed March. http://www.esri.com/software/arcgis/extensions/datainteroperability.

Esri Virtual Campus. 2015. "Learning ArcGIS Desktop." Accessed March. http://training.esri.com/Courses/LearnArcGIS_10/index.cfm?c=283.

European Comission. 2016. "About INSPIRE." Accessed February. http://inspire.ec.europa.eu/about-inspire/563.

Gallaher, Michael P., Alan C. O'Connor, John L. Dettbarn, Jr., and Linda T. Gilday. 2004. "Cost Analysis of Inadequate Interoperability in the U.S. Capital Facilities Industry." *NIST (National Institute of Standards and Technology).*

Garrett, James H Jr, and M Maher Hakim. 1992. "Object-Oriented Model of Engineering Design Standards." *J. Comput. Civ. Eng.* 6 (3): 323–47.

Ghafourian, Mahsa, and Hassan A Karimi. 2010. "CAD/GIS Integration Issues for Seamless Navigation between Indoor and Outdoor Environments." In *CAD and GIS Integration*, edited by Hassan A Karimi and Burcu Akinci, 129–42. New York, NY: CRC Press.

Gilbert, Thomas, Stuart Barr, Philip James, Jeremy Morley, and Qingyuan Ji. 2018. "Software Systems Approach to Multi-Scale GIS-BIM Utility Infrastructure Network Integration and Resource Flow Simulation." *ISPRS International Journal of Geo-Information* 7 (8): 310.

Greenwood, David, Stephen Lockley, Sagar Malsane, and Jane Matthews. 2010. “Automated Compliance Checking Using Building Information Models.” In *The Construction, Building and Real Estate Research Conference of the Royal Institution of Chartered Surveyors*, 363–71. Paris.

Groger, G, Thomas H Kolbe, Claus Nagel, and Karl-Heinz Haefele. 2012. “Open Geospatial Consortium OGC City Geography Markup Language (CityGML) En-Coding Standard.”

Hijazi, Ihab, Manfred Ehlers, Sisi Zlatanova, and Umit Isikdag. 2009. “IFC to CityGML Transformation Framework for Geo-Analysis: A Water Utility Network Case.” In *4th International Workshop on 3D Geo-Information*, 123–27. Ghent, Belgium.

Hjelseth, Eilif, and Nick Nisbet. 2010. “Exploring Semantic Based Model Checking.” In *Proceedings of the CIB W78 2010: 27th International Conference*. Cairo, Egypt.

Hjelseth, Eilif, and Nick Nisbet. 2011. “Capturing Normative Constraints by Use of the Semantic Mark-up RASE Methodology.” In *Proceedings of the CIB W78-W102 2011: International Conference*. Sophia Antipolis, France.

Hor, A. H., A. Jadidi, and G. Sohn. 2016. “BIM-GIS Integrated Geospatial Information Model Using Semantic Web and RDF Graphs.” In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3:73–79. Prague, Czech Republic.

INSPIRE. 2016a. “Coğrafi / Kent Bilgi Sistemi (CBS / KBS).” In *INSPIRE Direktifinin Uygulanmasına Yönelik Yatay Sektörde Kapasite Geliştirme Için Teknik Destek Projesi*.

INSPIRE. 2016b. “Şema Eşleştirme ve Veri Dönüşümü I Quantum GIS: Genel İşleyiş ve Fonksiyonlar.” In *INSPIRE Direktifinin Uygulanmasına Yönelik Yatay Sektörde Kapasite Geliştirme Için Teknik Destek Projesi*.

Inyim, Peeraya, Joseph Rivera, and Yimin Zhu. 2014. “Integration of Building Information Modeling and Economic and Environmental Impact Analysis to Support Sustainable Building Design.” *Journal of Management in Engineering* 31 (1): 1–10.

Irizarry, Javier, Ebrahim P. Karan, and Farzad Jalaei. 2013. “Integrating BIM and GIS to Improve the Visual Monitoring of Construction Supply Chain Management.” *Automation in Construction* 31: 241–54.

Irizarry, Javier, and Ebrahim P Karan. 2012. “Optimizing Location of Tower Cranes on Construction Sites Through Gis and Bim Integration.” *Journal of Information Technology in Construction (ITcon)* 17: 351–66.

Isikdag, Umit, Jason Underwood, and Ghassan Aouad. 2008. “An Investigation into the Applicability of Building Information Models in Geospatial Environment in Support of Site Selection and Fire Response Management Processes.” *Advanced Engineering Informatics* 22 (4): 504–19.

Isikdag, Umit, and Sisi Zlatanova. 2009. "Towards Defining a Framework for Automatic Generation of Buildings in CityGML Using Building Information Models." In *3D Geo-Information Sciences*, 79–96. Berlin, Germany: Springer Verlag.

Izmimod. 1997. "Bornova Plan Notları." Accessed February 2017. http://www.izmimod.org.tr/docs5/bornova-plannotlari.pdf

İzmir Büyükşehir Belediyesi. 2013. *İzmir Büyükşehir Belediyesi İmar Yönetmeliği*. Turkey.

Jalaei, Farzad, and Ahmad Jrade. 2015. "Integrating Building Information Modeling (BIM) and LEED System at the Conceptual Design Stage of Sustainable Buildings." *Sustainable Cities and Society* 18: 95–107.

Kang, J S. 2008. "Future of SEUMTER." In *BuildingSMART Forum*. Korea.

Kang, Taewook. 2018. "Development of a Conceptual Mapping Standard to Link Building and Geospatial Information." *ISPRS International Journal of Geo-Information* 7 (5): 162.

Karan, Ebrahim P., and Javier Irizarry. 2015. "Extending BIM Interoperability to Preconstruction Operations Using Geospatial Analyses and Semantic Web Services." *Automation in Construction* 53: 1–12.

Karimi, Hassan A., and Burcu Akinci, eds. 2010. *CAD and GIS Integration*. *CAD and GIS Integration*. Taylor & Francis Group (CRC Press).

Kerrigan, Shawn, and Kincho H Law. 2003. "Logic-Based Regulation Compliance-Assistance." In *Proceedings of the 9th International Conference on Artificial Intelligence and Law*, 126–35. Scotland, United Kingdom: ACM.

Khemlani, Lachmi. 2005. "CORENET E-PlanCheck : Singapore ' s Automated Code Checking System." *AECbytes*, 1–8.

Khosrow-Pour, Mehdi, ed. 2014. *Encyclopedia of Information Science and Technology*. 3rd ed. IGI Global.

Kiliccote, Han, and James H Garrett Jr. 1998. "Standards Modeling Language." *J. Comput. Civ. Eng.* 12 (3): 129–35.

Kim, Hyunjoo, Zhenhua Chen, Chung Suk Cho, Hyounseok Moon, Kibum Ju, and Wonsik Choi. 2015. "Integration of BIM and GIS: Highway Cut and Fill Earthwork Balancing." *Computing in Civil Engineering 2015*, no. January: 468–74.

Kiziltas, Semiha, Fernanda Leite, Burcu Akinci, and Robert R Lipman. 2010. "Interoperable Methodologies and Techniques in CAD." In *CAD and GIS Integration*, edited by Hassan A. Karimi and Burcu Akinci, 73–110. New York, NY: CRC Press.

Knoth, Laura, Johannes Scholz, Josef Strobl, Manfred Mittlböck, Bernhard Vockner, Caroline Atzl, Abbas Rajabifard, and Behnam Atazadeh. 2018. "Cross-Domain Building Models—a Step towards Interoperability." *ISPRS International Journal of Geo-Information* 7 (9): 363.

Laakso, Mikael, and Arto Kiviniemi. 2012. "The IFC Standard - A Review of History, Development, and Standardization." *Journal of Information Technology in Construction (ITcon)* 17 (May): 134–61.

Laat, Ruben de, and Léon van Berlo. 2011. "Integration of BIM and GIS: The Development of the CityGML GeoBIM Extension." In *Advances in 3D Geo-Information Sciences*, edited by C. Kolbe, T. H.; König, G.; Nagel, 211–25. Berlin, Heidelberg: Springer.

Lapierre, A., and P. Cote. 2007. "Using Open Web Services for Urban Data Management: A Testbed Resulting from an OGC Initiative for Offering Standard CAD/GIS/BIM Services." *Urban and Regional Data Management. Annual Symposium of the Urban Data Management Society*, 381–93.

Lee, Hyunsoo, Jin Kook Lee, Seokyung Park, and Inhan Kim. 2016. "Translating Building Legislation into a Computer-Executable Format for Evaluating Building Permit Requirements." *Automation in Construction* 71: 49–61.

Lee, S H. 2011. "Development of SEUMTER Legality System Using BIM Model." In *BuildSMART Construction IT&BIM User Conference*. Korea.

Lee, Sang-Ho, and Bong-Geun Kim. 2011. "IFC Extension for Road Structures and Digital Modeling." *Procedia Engineering* 14: 1037–42.

Liebich, Thomas, Jeffrey Wix, James Forester, and Z Qi. 2002. "Speeding-up the Building Plan Approval-the Singapore e-Plan Checking Project Offers Automatic Plan Checking Based on IFC." In *European Conferences on Product and Process Modelling (ECPPM) 2002 - EWork and EBusiness in Architecture, Engineering and Construction*, 467-471 Portoroz, Slovenia.

Lindberg, K. 2006. "MATL, Development of a New ICT-System for Registration and Assessment of Accessibility to Public Buildings." In *Property Management, Statsbygg, BAS Conference*.

Liu, Xin, Xiangyu Wang, Graeme Wright, Jack Cheng, Xiao Li, and Rui Liu. 2017. "A State-of-the-Art Review on the Integration of Building Information Modeling (BIM) and Geographic Information System (GIS)." *ISPRS International Journal of Geo-Information* 6 (2): 53.

Macit İlal, Sibel, and H. Murat Günaydın. 2017. "Computer Representation of Building Codes for Automated Compliance Checking." *Automation in Construction* 82 (April): 43–58.

Macit, Sibel. 2014. "Computer Representation of Building Codes for Automated Compliance Checking." Izmir Institute of Technology.

Malsane, Sagar. 2015. “The Application of Automated Rule Checking to Existing Uk Building Regulations Using BIM Technologies.” University of Northumbria at Newcastle.

Malsane, Sagar, Jane Matthews, Steve Lockley, Peter E.D. Love, and David Greenwood. 2015. “Development of an Object Model for Automated Compliance Checking.” *Automation in Construction* 49: 51–58.

Mignard, Clement, and Christophe Nicolle. 2014. “Merging BIM and GIS Using Ontologies Application to Urban Facility Management in ACTIVe3D.” *Computers in Industry* 65 (9): 1276–90.

Moyer, D David, and Kenneth P Fisher. 1973. *Land Parcel Identifiers for Information Systems*. Chicago: American Bar Foundation.

National Institute of Building Sciences. 2016. “GIS/BIM Information Exchange.” Accessed January. https://www.nibs.org/?page=bsa_gisbimie

NBS. 2019. “What Is Building Information Modelling (BIM)?” Accessed June. https://www.thenbs.com/

OGC. 2019a. “IndoorGML SWG.” Accessed May. http://www.opengeospatial.org/projects/groups/indoorgmlswg

OGC. 2019b. “Land and Infrastructure DWG.” Accessed May. http://www.opengeospatial.org/projects/groups/landinfradwg

Ohori, Ken Arroyo, Filip Biljecki, Abdoulaye Diakité, Thomas Krijnen, Hugo Ledoux, and Jantien Stoter. 2017. “Towards an Integration of GIS and BIM Data: What Are the Geometric and Topological Issues?” In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:1–8. Melbourne, Australia.

Olsson, Per-Ola, Josefine Axelsson, Martin Hooper, and Lars Harrie. 2018. “Automation of Building Permission by Integration of BIM and Geospatial Data.” *ISPRS International Journal of Geo-Information* 7 (8): 307.

Opengeospatial. 2017. “Geography Markup Language.” Accessed March. http://www.opengeospatial.org/standards/gml

OSgeo. 2019. “The Open Source Geospatial Foundation.” Accessed June. https://www.osgeo.org/

Park, T, T Kang, Y Lee, and K Seo. 2014. “Project Cost Estimation of National Road in Preliminary Feasibility Stage Using BIM/GIS Platform.” In *The Sixth Annual International Conference on Computing in Civil and Building Engineering*, 423–30.

Pauwels, P., D. Van Deursen, R. Verstraeten, J. De Roo, R. De Meyer, R. Van De Walle, and J. Van Campenhout. 2011. “A Semantic Rule Checking Environment for Building Performance Checking.” *Automation in Construction* 20 (5): 506–18.

Peachavanish, Ratchata, Hassan A. Karimi, Burcu Akinci, and Frank Boukamp. 2006. "An Ontological Engineering Approach for Integrating CAD and GIS in Support of Infrastructure Management." *Advanced Engineering Informatics* 20 (1): 71–88.

Portal Netcad. 2019. "Drawing Road." Accessed May. http://portal.netcad.com.tr/pages/viewpage.action?pageId=99614816

Przybyla, J, PE, GISP, and Woolpert Inc. 2010. "Introduction to BIM-GIS Integration." *Ecobuild America 2010*.

Quantum GIS. 2019. "Quantum GIS." A Free and Open Source Geographic Information System. Accessed June. https://qgis.org/en/site/

Rafiee, Azarakhsh, Eduardo Dias, Steven Fruijtier, and Henk Scholten. 2014. "From BIM to Geo-Analysis: View Coverage and Shadow Analysis by BIM/GIS Integration." *Procedia Environmental Sciences* 22: 397–402.

Reed, C. 2010. "Interoperable Methodologies and Techniques in CAD-GIS Integration Standardization Efforts, The Open Geospatial Consortium Perspective." In *CAD and GIS Integration*, edited by Hassan A Karimi and Burcu Akıncı, 111–28. New York, NY: CRC Press.

Republic of Turkey Ministry of Environment and Urbanization. 2016a. "Inspire Nedir?" Accessed February. http://www.csb.gov.tr/gm/cbs/index.php?Sayfa=sayfa&Tur=webmenu&Id=36621

Republic of Turkey Ministry of Environment and Urbanization. 2016b. "National Spatial Data Infrastructure." Accessed February. http://www.csb.gov.tr/gm/cbsen/index.php?Sayfa=sayfa&Tur=webmenu&Id=13829

Salama, D. M., and N. M. El-Gohary. 2011. "Semantic Modeling for Automated Compliance Checking." *Computing in Civil Engineering*, no. 413: 641–48.

Sebastian, Rizal, Michel Böhms, and Pim Van Den Helm. 2013. "BIM and GIS for Low-Disturbance Construction." In *Proceedings of the 13th International Conference on Construction Applications of Virtual Reality*, 469–79. London, UK.

Shi, Jianyong, and Pai Liu. 2014. "An Agent-Based Evacuation Model to Support Fire Safety Design Based on an Integrated 3D GIS and BIM Platform." In *Computing in Civil and Building Engineering*, 1893–1900.

Song, Yongze, Xiangyu Wang, Yi Tan, Peng Wu, Monty Sutrisna, Jack C.P. Cheng, and Keith Hampson. 2017. "Trends and Opportunities of BIM-GIS Integration in the Architecture, Engineering and Construction Industry: A Review from a Spatio-Temporal Statistical Perspective." *ISPRS International Journal of Geo-Information* 6 (12): 1–32.

W3schools. 2018. "XML DOM." Accessed March. https://www.w3schools.com/xml/xml_dom.asp

Waard, Marcel de. 1992. “Computer Aided Conformance Checking.” In *Proceedings of CIB W78 Conference*, 177–88.

Wicaksono, Hendro, Sven Rogalski, and Enrico Kusnady. 2010. “Knowledge-Based Intelligent Energy Management Using Building Automation System.” *2010 Conference Proceedings IPEC*, 1140–45.

Wu, I. Chen, and Shang Hsien Hsieh. 2007. “Transformation from IFC Data Model to GML Data Model: Methodology and Tool Development.” *Journal of the Chinese Institute of Engineers* 30 (6): 1085–90.

Yabuki, Nobuyoshi, and Kincho H. Law. 1993. “An Object-Logic Model for the Representation and Processing of Design Standards.” *Engineering with Computers* 9 (3): 133–59.

Yang, Q. Z., and Xingjian Xu. 2004. “Design Knowledge Modeling and Software Implementation for Building Code Compliance Checking.” *Building and Environment* 39 (6): 689–98.

Yıldız, Ferruh. 2012. *İmar Bilgisi: Planlama Uygulama Mevzuat*. 8th ed. Ankara: Nobel Yayıncılık.

Yurchyshyna, Anastasiya, Catherine Faron-Zucker, Nhan Le Thanh, and Alain Zarli. 2008. “Towards an Ontology-Enabled Approach for Modeling the Process of Conformity Checking in Construction.” In *Proceedings of CAiSE’08 Forum*, 344:21–24.

Zhang, Jiansong, and Nora El-Gohary. 2012. “Automated Regulatory Information Extraction from Building Codes : Leveraging Syntactic and Semantic Information.” In *Construction Research Congress 2012*, 622–32.

Zhang, Jiansong, and Nora M. El-Gohary. 2017. “Integrating Semantic NLP and Logic Reasoning into a Unified System for Fully-Automated Code Checking.” *Automation in Construction* 73: 45–57.

Zhang, Sijie, Frank Boukamp, and Jochen Teizer. 2015. “Ontology-Based Semantic Modeling of Construction Safety Knowledge: Towards Automated Safety Planning for Job Hazard Analysis (JHA).” *Automation in Construction* 52: 29–41.

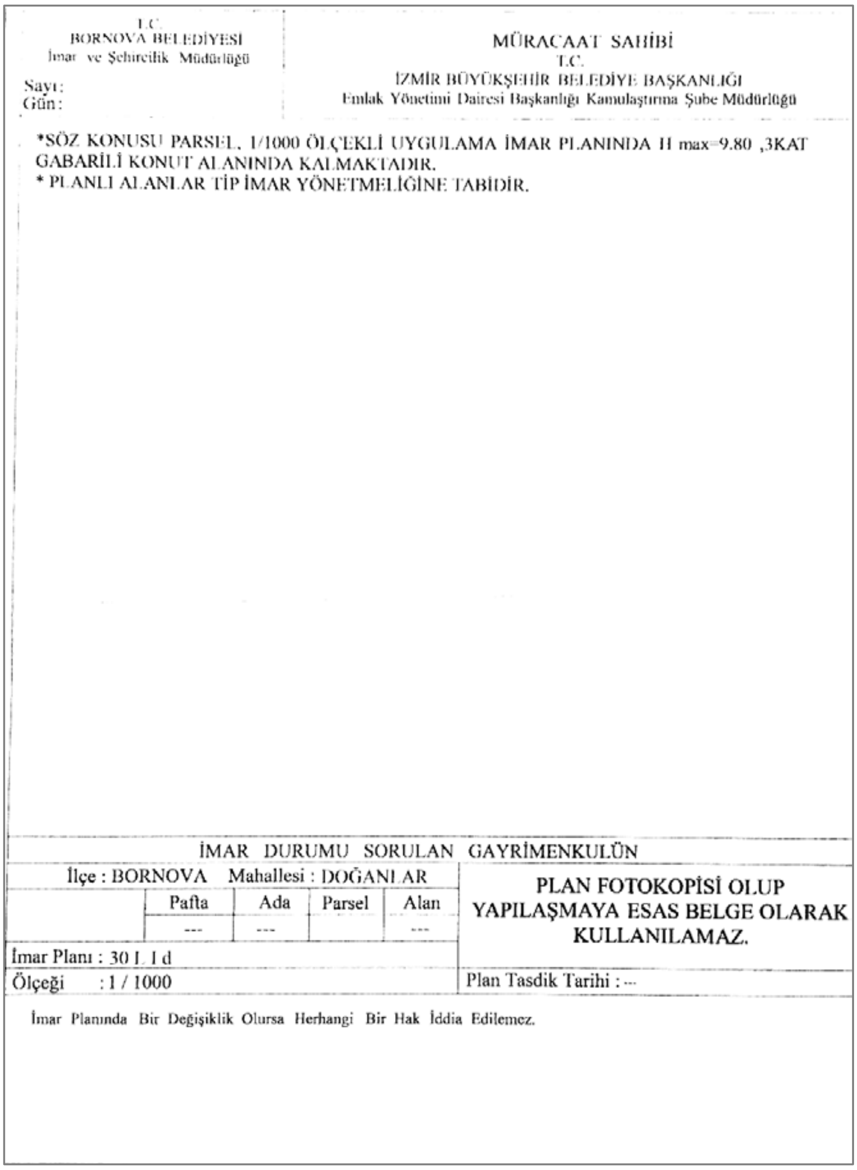
Zhiliang, Ma, Wei Zhenhua, Song Wu, and Lou Zhe. 2011. “Application and Extension of the IFC Standard in Construction Cost Estimating for Tendering in China.” *Automation in Construction* 20 (2): 196–204.

APPENDICES

APPENDIX A

ZONING DOCUMENT SAMPLES

This appendix shows the zoning document samples. Site Plan document contains spatial notes, which were read and taken from the Bornova development plan hardcopy, and information related to parcel including parcel number, lot number, name of the neighborhood and area.



T.C.
BORNOVA BELEDİYESİ
İmar ve Şehircilik Müdürlüğü

Sayı :
Gün :

MÜRACAAT SAHİBİ
T.C.
İZMİR BÜYÜKŞEHİR BELEDİYE BAŞKANLIĞI
Emlak Yönetimi Dairesi Başkanlığı Kamulaştırma Şube Müdürlüğü

*SÖZ KONUSU PARSEL, 1/1000 ÖLÇEKLİ UYGULAMA İMAR PLANINDA H max=9.80 ,3KAT GABARİLİ KONUT ALANINDA KALMAKTADIR.
* PLANLI ALANLAR TİP İMAR YÖNETMELİĞİNE TABİDİR.

İMAR DURUMU SORULAN GAYRİMENKULÜN

İlçe : BORNOVA Mahallesi : DOĞANLAR

	Pafta	Ada	Parsel	Alan
	---	---		---

PLAN FOTOKOPİSİ OLUP YAPILAŞMAYA ESAS BELGE OLARAK KULLANILAMAZ.

İmar Planı : 30 L I d
Ölçeği : 1 / 1000
Plan Tasdik Tarihi : ---

İmar Planında Bir Değişiklik Olursa Herhangi Bir Hak İddia Edilemez.

Figure A.1. Site Plan document.

Zoning status document contains spatial notes and detailed information about the property considering the Site Plan document.

T.C.
BORNOVA BELEDİYESİ
İmar ve Şehircilik Müdürlüğü

Sayı: 310.05.02-_____
Gün: __/__/201_

MÜRACAAT SAHİBİ

____

* Söz konusu parsel, 1/1000 ölçekli Uygulama İmar Planında______________________ alanında kalmakta olup, aşağıda belirtilen şekilde yapılaşabilir.
* Bu imar durumu __/__/____ gün, ____yevmiye nolu Tapu Senedine ve __/__/___ gün ___ sayılı Bornova ___ nolu LİHKAB'ın hazırladığı Aplikasyon Krokisine istinaden hazırlanmıştır.

PTT ONAYI
GEREKMEKTEDİR

OTOPARK SORUNU
PARSELİ İÇİNDE
ÇÖZÜMLENECEKTİR.

YAPI YERİ UYGULAMA VE
KOT KROKİSİ ALINMASI
GEREKMEKTEDİR

RUHSAT ALINMADAN
HAFRİYAT YAPILAMAZ

________ KİTLE HATTI
\- - - - - - - - YAPI YAKLAŞMA

Çizen: ____________

İMAR DURUMU SORULAN GAYRİMENKULÜN

İlçe: BORNOVA		Mahallesi: ÇAMDİBİ			İnşaat Nizamı	_____
	Pafta	Ada	Parsel	Alan	Kat Adedi	_____
	___	___	___	---	Bina Yüksekliği	_____
İmar Planı : _____					Bina Derinliği	_____
Ölçeği : 1 / 1000					Ön Bahçe Mesafesi	_____
Tasdik Tarihi :---					Komşu Mesafeler	_____
					Arka Bahçe Mesafesi	_____

1. Bu İmar Durumu Proje Tanzimine Esas Olmak Üzere Verilmiştir.
2. İmar Planında Bir Değişiklik Olursa Herhangi Bir Hak İddia Edilemez.
3. Bu imar durumuna esas aplikasyon krokisi sayfanın arka yüzünde olup, bu aplikasyon krokisinde değişiklik olması durumunda imar durumu belgesinin yeniden irdelenmesi gerekmektedir.

İMAR DUR. Ş.B. ŞEFİ

İMAR ve ŞEHİRCİLİK MÜDÜRÜ

Figure A.2. Zoning status document.

APPENDIX B

ACCESS DATABASE TABLES

This appendix shows the Microsoft Access database tables organized to store and model the rules as demonstrative examples to conduct the checking process in this thesis.

class	property
BimBuilding	constructionTechnique
BimBuilding	numberofStorey
BimBuilding	groundFloorBaseLevel
BimBuilding	hasParkingOnGroundFloor
BimBuilding	height
BimBuilding	area
BimBuilding	facadeLength
BimBuilding	depth
BimParcel	area
BimParcel	constructionOrder
BimParcel	clearDepth
BimParcel	facedRoadNumber
BimParcel	isOnCorner
BimParcel	numberofTreesToBePlanted
BimParcel	frontGardenLevel
Block	refFrontSetbackDist
Block	constructionOrder
Block	refRearSetbackDist
Block	halfOfAreaHasLicensedBuilding
Block	hasExistingBuilding
Building	refBuildingDepth
FrontBoundary	setbackDistance
Parcel	widestRoadSidewalkTopLevel
RearBoundary	setbackDistance
SideBoundary	setbackDistance

Figure B.1. Domain object table of IMHZ code clauses.

concept	domainObject	property	comparator	value
sideBoundary	SideBoundary			
semiDetachedBlock	Block	constructionOrder	equal	semiDetached
rearBoundary	RearBoundary			
plannedUnitBlock	Block	constructionOrder	equal	plannedUnit
parcelBoundary	ParcelBoundary			
parcel	Parcel			
frontBoundary	FrontBoundary			
detachedBlock	Block	constructionOrder	equal	detached
building	Building			
block	Block			
bimParcel	BimParcel			
bimBuilding	BimBuilding			
attachedBlock	Block	constructionOrder	equal	attached

Figure B.2. Concept-mapping table of IMHZ code clauses.

Requirement Selection

id	text	r_concept	r_property	r_com	r_value	r_uni	s_concept	s_property	s_comp	s_value
R.27.01	Setback distance	FrontBoundary	setbackDistance	≥	5	m	block	hasExistingBuildin	equal	no
R.27.02	İkiz yapı nizamı	FrontBoundary	setbackDistance	=	{Block_refFrontSetbackDist}	m	block	constructionOrder	equal	semiDetached
R.27.03	Blok yapı nizam	FrontBoundary	setbackDistance	=	{Block_refFrontSetbackDist}	m	block	constructionOrder	equal	plannedUnit
R.27.04	Bitişik yapı niza	FrontBoundary	setbackDistance	=	{Block_refFrontSetbackDist}	m	block	constructionOrder	equal	attached
R.27.05	Bitişik nizamda	SideBoundary	setbackDistance	=	0	m	bimParcel	constructionOrder	equal	attached
R.27.06	Bundan yüksek	SideBoundary	setbackDistance	≥	(((:{BimBuilding_storeyNumber}:-4)/2)+3)	m	bimbuilding	storeyNumber	≥	4
R.27.07	Yan bahçe mes	SideBoundary	setbackDistance	≥	3	m	bimbuilding	storeyNumber	<	4
R.27.08	Arka bahçe mes	RearBoundary	setbackDistance	≥	(:{BimBuilding_height}:/2)	m	block	hasExistingBuildin	equal	no
R.27.09	Hiçbir yerde 3.0	RearBoundary	setbackDistance	≥	3	m	block	hasExistingBuildin	equal	yes
R.27.10	a) İkiz yapı niza	RearBoundary	setbackDistance	=	{Block_refRearSetbackDist}	m	block	constructionOrder	equal	semiDetached
R.27.11	Blok yapı nizam	RearBoundary	setbackDistance	=	{Block_refRearSetbackDist}	m	block	constructionOrder	equal	plannedUnit
R.27.12	Bitişik yapı niza	RearBoundary	setbackDistance	=	{Block_refRearSetbackDist}	m	block	constructionOrder	equal	attached
R.28.01	Bina derinlikleri	BimBuilding	depth	≤	22	m	block	hasExistingBuildin	equal	no
R.28.02	Mevcut yapılaşı	BimBuilding	depth	≤	22	m				
R.28.03	a) İkiz yapı niza	BimBuilding	depth	=	{Building_refBuildingDepth}	m	semiDetachedB	hasExistingBuildin	equal	yes
R.28.04	b) Blok yapı niz	BimBuilding	depth	=	{Building_refBuildingDepth}	m	plannedUnitBlo	hasExistingBuildin	equal	yes
R.28.05	c) Bitişik yapı n	BimBuilding	depth	=	{Building_refBuildingDepth}	m	attachedBlock	halfOfAreaHasLic	equal	yes
R.28.06	a) Bitişik yapı n	BimBuilding	depth	=	{Building_refBuildingDepth}	m	bimparcel	isOnCorner	equal	yes
R.28.08	b) Köşe başında	BimBuilding	depth	≤	{BimParcel_clearDepth}	m	bimparcel	facedRoadNumber	=	2
R.29.01	Ayrık yapı nizam	BimBuilding	facadeLength	≤	30	m	bimparcel	constructionOrder	equal	detached
R.37.13	Ön bahçelerin t	BimParcel	frontGardenLevel	=	{Parcel_widestRoadSidewalkTopLevel}					
R.38.01	Zemin kat döşe	BimBuilding	groundFloorBaseLevel	=	(:{Parcel_widestRoadSidewalkTopLevel}:+1)	m				
R.38.02	Binaların zemin	BimBuilding	groundFloorBaseLevel	≥	{Parcel_widestRoadSidewalkTopLevel}	m	bimBuilding	hasParkingOnGrou	equal	yes
R.64.01	İmar planlarında	BimParcel	numberofTreesToBePla	=	(:{BimParcel_area}:-:{BimBuilding_area}:/25)					

Figure B.3. Rule table of IMHZ code clauses.

node	list
R.27.01	27.A
R.27.02	27.A
R.27.03	27.A
R.27.04	27.A
R.27.05	27.B
R.27.06	27.B
R.27.07	27.B
R.27.08	27.C
RS.27.C.1	27.C
R.27.09	27.C.1
RS.27.C.1.1	27.C.1
R.27.10	27.C.1.1
R.27.11	27.C.1.1
R.27.12	27.C.1.1
R.28.01	28
R.28.06	28
R.28.08	28
RS.28.1	28
R.28.02	28.1
RS.28.1.1	28.1
R.28.03	28.1.1
R.28.04	28.1.1
R.28.05	28.1.1
R.29.01	29
R.37.13	37
R.38.01	38
R.38.02	38
R.64.01	64

Figure B.4. Node_List table of IMHZ code clauses.

id	conjunction	nodeList
RS.27.A	OR	27.A
RS.27.B	OR	27.B
RS.27.C	OR	27.C
RS.27.C.1	AND	27.C.1
RS.27.C.1.1	OR	27.C.1.1
RS.28	OR	28
RS.28.1	AND	28.1
RS.28.1.1	OR	28.1.1
RS.29	NULL	29
R.37	NULL	37
RS.38	OR	38
RS.64	NULL	64

Figure B.5. RuleSet table of IMHZ code clauses.

domainObject	ruleSet
FrontBoundary	RS.27.A
SideBoundary	RS.27.B
RearBoundary	RS.27.C
BimBuilding	RS.28
BimBuilding	RS.29
BimParcel	RS.37
BimBuilding	RS.38
BimParcel	RS.64

Figure B.6. RuleSet-Group table of IMHZ code clauses.

APPENDIX C

UNIFIED ZONING DOMAIN MODEL

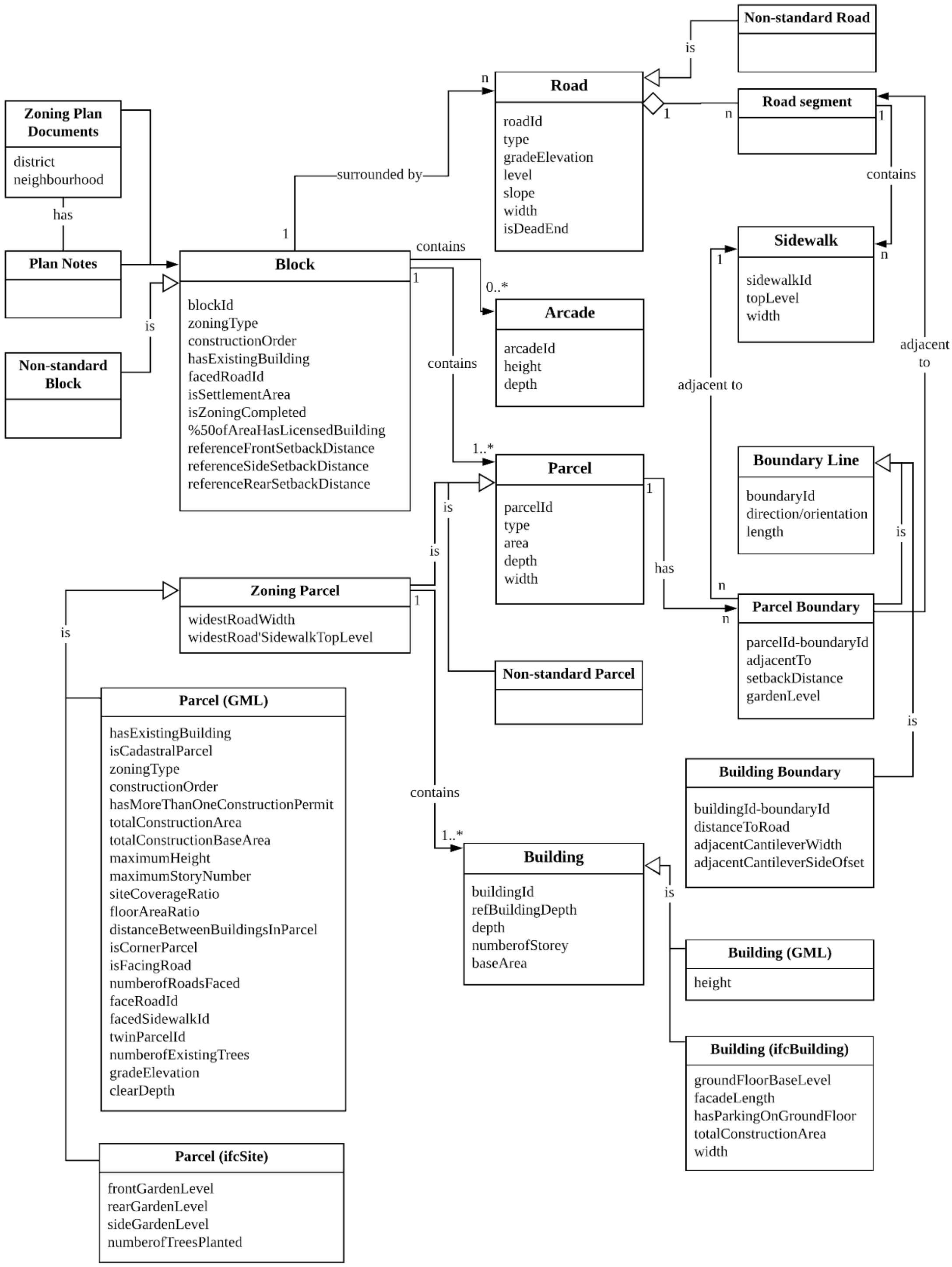


Figure C.1. Unified zoning domain model.

VITA

PERSONAL

Surname, Name: DEMİR ALTINTAŞ, Yelin

Date and Place of Birth: 29.03.1989 – İzmir (Turkey)

E-mail: dmryelin@gmail.com

EDUCATION

PhD., İzmir Institute of Technology, Graduate School of Engineering and Sciences, Department of Architecture (2013-2020)

Thesis: "A Method for GIS and BIM Integration to Support Automated Zoning Code Compliance Checking"

M.Sc., İzmir Institute of Technology, Graduate School of Engineering and Sciences, Department of Architecture (2010-2013)

Thesis: "The Assessment of a New Relationship between Form and Structure in Digital Architectural Design After 1990's"

B.Arch., İzmir Institute of Technology, Faculty of Architecture, Department of Architecture (2006-2010)

ACADEMIC EXPERIENCES

Research Assistant, İzmir Institute of Technology, Graduate School of Engineering and Sciences, Department of Architecture (2010-2019)

Part-time Lecturer, Yaşar University, Faculty of Architecture, Department of Architecture (2019-continuing)