Automation in Construction xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Computer representation of building codes for automated compliance checking

Sibel Macit İlal[a,*], H. Murat Günaydın[b]

[a] İzmir Democracy University, Faculty of Architecture, Department of Architecture, İzmir, Turkey
[b] İstanbul Technical University, Faculty of Architecture, Department of Architecture, İstanbul, Turkey

ARTICLE INFO

Keywords:
Building code representation
Automated compliance checking
Semantic modeling
BIM
IFC

ABSTRACT

Development of automated building code compliance checking systems requires appropriate representations for building codes. Building codes are complex documents written in natural languages, and the development of computable representations is challenging. This paper presents a new model and an accompanying modeling methodology for the representation of building codes that may be utilized in the development of future automated compliance checking systems.

The new model combines the semantic modeling approach of the SMARTcodes project with the theoretical foundations established by Nyman and Fenves, namely the four level representation. This hybrid model organizes the representation in four levels and allows for separate modeling of domain concepts, individual rule statements, relationships between rules, and the overall organization of the building code.

The model is evaluated with a case study. The İzmir Municipality Housing and Zoning Code is chosen as it is representative of complex building codes that are in effect throughout Turkey. The formalizable rules in the section of Izmir code that apply to all types of buildings, are represented in computer implementable format based on the new model. The research presented in this paper shows that decomposing a building code into four levels and modeling rules based on the semantic-oriented paradigm is an effective modeling strategy for representing building codes in a computable form.

1. Introduction

In the Architecture, Engineering, and Construction (AEC) industry, building projects must be checked against numerous building codes for compliance. They are allowed to be executed only when compliance with all applicable rules of the building code have been guaranteed. Compliance checking is a major task for both architects and building certifiers often involving ambiguities and inconsistencies in assessment, leading to delays in the overall construction process [1,2]. Failure to correctly assess projects for compliance can also have negative effects on building performance and allow errors that are expensive to correct. Today, although every building project is modeled in a digital environment, compliance checking is a manual process increasing the delays as well as the risk for errors in evaluation [3].

Automated compliance checking has long been an area of research that aims to provide computational support for accurate compliance checking of building projects against applicable building codes in a time and cost effective way. Research into developing automated compliance checking systems has focused mostly on three areas: Representation of building codes in computational format [4–6] definition of building model views [7,8], and compliance checking algorithms and reporting [1,9,10].

Automated compliance checking systems are expected to retrieve a set of building codes from related authorities and conduct compliance checking on submitted building projects. Compliance checking systems primarily require appropriate computer-based models of both building codes and building designs. Advances in BIM tools have finally established a standardized representation for building designs, even if it is currently deemed unsatisfactory. However, a standard representation for building codes is still not available.

The goal of the research presented in this paper is to develop a new computer representation for building codes that can be used in the development of future automated compliance checking systems. The aim is to develop a formal model that supports the creation of digital representations of build codes by following a corresponding methodology. To achieve this aim, the research has been conducted in the following stages: 1) Exploring and evaluating previous modeling approaches; 2) Analyzing building codes to understand the various types of information contained in them and identifying the components of rule statements as well as the organization of the documents; 3)

* Corresponding author.
E-mail addresses: sibelmacit@gmail.com, sibel.macit@idu.edu.tr (S. Macit İlal), gunaydinh@itu.edu.tr (H.M. Günaydın).

http://dx.doi.org/10.1016/j.autcon.2017.06.018
Received 4 May 2016; Received in revised form 9 April 2017; Accepted 11 June 2017

Developing a formal representation model for building codes; 4) Defining a building code modeling methodology for utilizing the representation model to build digital versions of existing building codes; 5) Modeling of an actual building code as a case study to demonstrate the feasibility, benefits, and limitations of the representation model; 6) Implementing a prototype system to demonstrate an application and to test the validity of the new model.

This research focuses on the issue of building code representations and its scope is limited to it. Other issues impeding the development of automated compliance checking systems exist, such as building model extensions for code checking, and efficient querying and checking methods but they are beyond the scope of this work. Additionally, the scope is also limited to representing rules that are formalizable, i.e. can be evaluated computationally. Semi-formalizable rules that contain ambiguous statements or based on relative judgment (e.g. enough, adequate, etc.) as well as non-formalizable rules that require qualitative evaluations (e.g. aesthetic judgment) are not considered.

As a conclusion, this research proposes a formal model for building code representation based on the analysis of building codes and theories established in literature. The theory embedded in this proposed model is evaluated through the development of an actual building code and a prototype implementation.

2. Related work

There has been an extensive amount of research conducted internationally over the last four decades in the area of representing building codes in a computable format for automated compliance checking. The introduction of decision tables by Fenves [11] is the initial effort on building code representation. In this effort, building code provisions are represented in a precise and unambiguous decision table form. A decision table is a concise tabular representation of the conditions applicable in a given situation and of the appropriate actions to be taken as a result of the values of the conditions [12]. A follow-up project by the same group of researchers investigated the restructuring of the AISC (American Institute of Steel Construction) Specification [13]. In this project, the content of the building code is modeled in four levels. This abstract model of the logical structure of building codes is used as a modeling methodology in a software system called SASE (Standard Analysis, Synthesis and Expression), to represent individual provisions, relationships among provisions, and the organization of the building code [14]. This system aimed to provide tools for creating decision tables and for structuring building code representations.

Several researchers [15–18] have proposed methods based on a rule-based modeling approach for representing building codes as rules/clauses in processing systems. In these models, the clauses of the building code are represented as a set of rules in the form of IF [condition] THEN [action] statements instead of decision tables. Later works focused on structuring building codes in a predicate logic structure [19,20]. A commercial expert system was developed in Australia by the Commonwealth Scientific and Industrial Research Organization (CSIRO) called BCAider [21]. This system was available between 1991 and 2005.

Many other models for representing building codes in a computable format have been proposed. Garrett and Hakim [22] developed an object-oriented model of building codes, which allows organizing a building code around building objects pertinent to the building code. Waard [23] offered another object-oriented approach to building code processing. In this study, an object model for residential buildings and another object model for building codes were developed, and the two models were linked for compliance checking. Yabuki and Law [24] combined first order predicate logic and object-oriented modeling approaches to represent and process building codes. Kiliccote and Garrett [25] developed a context-oriented model for representing building codes. This model uses the object-oriented modeling approach and organizes building code around "contexts" which are a collection of subclasses used to define conditional parts of the provisions for which they are applicable. Given a predicate logic structure, Kerrigan and Law [26] developed the REGNET application to determine the applicability of various codes under given building conditions, based on a question-and-answer user interface.

Previous building code representation research efforts mainly focused on the hard-coding approach. The main disadvantage to this approach is that it requires a high-level of expertise in computer programming to define, write and maintain building codes. To overcome the deficiencies of hard-coded representation approaches, recently, attention has been directed towards the study of semantic modeling approach, which is a relatively new method for knowledge representation. The SMARTcodes project [27] is a semantic approach which proposes to mark-up building codes in such a way that rules are dynamically generated in a computable format. The project provides a protocol and a software program (SMARTcodes Builder) for creating smart versions (tagged representations) of actual building code texts that reflects building codes with schema and tags used for automated compliance checking applications [28].

Recently, the application of an ontology-based approach has been investigated as a possible computable framework for building code representation. Yurchyshyna et al. [29] developed a formal ontology-based approach for the formalization and semantic organization of building codes. Ontology-based building code representation using semantic web technologies has also been explored by researchers [30]. Dimyadi et al. [31] developed a regulatory knowledge model and a higher level query language for designers' access to this model. Beach et al. [32] developed a rule-based semantic methodology for the specification of a regulatory compliance checking system. There have been also some projects using semantic modeling approach and the application of industry specific taxonomies and ontologies in combination with Artificial Intelligence (AI) and Natural Language Processing (NLP) techniques to allow systems to interpret building code by automated or semi-automated data extraction [33–35].

Exploration of building code compliance checking systems for building models began following the development of the Industry Foundation Classes (IFC) in the 1990s. The Singapore Construction and Real Estate NETwork (CORENET) project is the earliest production of building code compliance checking effort initiated in 1995 [36]. Initial work was based on electronic 2D drawings, but later on IFC [37] was used. The CORENET project developed the FORNAX platform to capture needed building code information. Another effort is the DesignCheck system from Australia, initiated in 2006 [38]. In this effort, the EXPRESS Data Manager (EDM) [39] platform was used for encoding barrier-free accessibility rules. A more recent effort, led by USA International Code Council (ICC), developed SMARTcodes Model Checking System [40,41]. It is a platform providing methods of translation from written, natural language rules to computer code. The platform targets energy conservation rules. USA General Services Administration (GSA) have supported development of a rule checking system for circulation and security validation of U.S. Court houses [42].

Although several researches have already proposed various building code models, and software environments to operate on these models and check building projects for compliance with building codes, for a variety of reasons, employing these environments in AEC industry practice has been limited. Literature review reveals that the reasons for this failure are related to the building code models used in these environments [43,44]. Previous building code models do have several limitations. One limitation is not being comprehensive enough compared to the complex nature of building codes and thus lacking the capability to represent all of the various types of information in building codes. A second limitation is that some building codes are hard-coded into the systems, thus lacking flexibility, maintainability, and user control (i.e. non-programmer users cannot add/modify the rules embedded in the system). Furthermore, any change in the building code necessitates changes in all such systems. A third