Assessment of the Building Code of Australia to Inform the Development of BIM-enabled Code-checking Systems

Shan-Ying Shih[1], Willy Sher[2] and Helen Giggins[3]

Abstract

Building projects in Australia are traditionally checked manually against the Building Code of Australia (BCA) – a set of continuously changing and increasingly complex regulations. Manual certification processes are error-prone and time-consuming tasks (J. Jeong & G. Lee 2010; Tan et al. 2010). Technical developments in Building Information Modelling (BIM) provide the potential for a new-generation of software tools to assist the checking of compliance with building codes. These should improve efficiency and accuracy for designers as well as for governing bodies. This paper reviews the requirements of certification processes for commercial buildings with specific emphasis on fire codes. We describe the selection of building class, the assessment of fire rating and the interpretation of fire codes. The characteristics of these requirements are explored, and ways for BIM-enabled checking systems to access these data are identified.

Keywords: BCA, IFC, code-checking system, BIM

1. Introduction

Within the construction industries, building designs require approval by governing bodies before construction works can commence. In Australia, the approval documents that applicants need to obtain are Development Approvals (DA) and Construction Certificates (CC) (Building Professionals Board 2011). Several studies have identified the time-consuming and error-prone nature of checking building designs against building codes (J. Jeong & G. Lee 2010; Tan et al. 2010). These problems result largely from the manual certification processes conducted by certifying authorities and are compounded by the continuous changes and increasing complexities of the building codes (Tan et al. 2010; Greenwood et al. 2010). If building designs are found to be not compliant, delays and budget overruns ensue (Abrantes 2010; Tan et al. 2010). Moreover, this may compromise the quality of the construction works (Building Professionals Board 2012).

[1] Ph.D. Student; School of Architecture and Built Environment; University of Newcastle; University Drive, Callaghan NSW 2308, Australia; Shan-ying.shih@uon.edu.au.
[2] Associate Professor; School of Architecture and Built Environment; University of Newcastle; University Drive, Callaghan NSW 2308, Australia; willy.sher@newcastle.edu.au.
[3] Lecturer; School of Architecture and Built Environment; University of Newcastle; University Drive, Callaghan NSW 2308, Australia; Helen.giggins@newcastle.edu.au.

Some studies have suggested developing code-checking systems to improve the certification process for both certifying authorities and designers (L. Ding et al. 2004; Abrantes 2010; J. Jeong & G. Lee 2010), thereby improving efficiency and productivity. Currently, Building Information Models (BIM) are acknowledged as an effective platform for exchanging information between design team members (Babič et al. 2010; Grilo & Jardim-Goncalves 2010; Succar 2009). BIM has the potential to drive the development of applications that conduct rigorous analyses for team members before construction works start (Rogers 2012). Additionally, related research notes that BIM has the potential for new-generation software tools to facilitate the checking of compliance with building codes, hence improving the efficiency and accuracy of the checking processes for designers as well as for governing bodies (Greenwood et al. 2010; Yang & Li 2001). Examples of such BIM applications have been developed in several countries (including Singapore and Norway) and have been demonstrated to be beneficial for stakeholders (Eastman, J. Lee, et al. 2009). Moreover, the development of BIM-enabled code-checking systems for Australia has also been demonstrated in a buildingSMART Australasia report (buildingSMART Australasia 2012).

This present study seeks to extend earlier developments for the Australian construction industry. In next section, it reviews the characteristics of existing code-checking systems to identify the information required by BIM-enabled checking systems. Afterwards, the procedures of fire code-checking and the ways of interpreting fire codes are discussed in section 3. This section also sets out a methodology for the interpretation of building codes for future studies.

2. Current Code-Checking Systems

The concept of developing code-checking systems that use BIM is not new. However, the factors that affect the development of code-checking systems are complex and these applications are not yet mature. A leading exponent of BIM-enabled code-checking systems has been CORENET (Construction and Real Estate NETwork) in Singapore, where a significant proportion of the nation's building regulations can be checked through Industry Foundation Class (IFC) models (Eastman, J. Lee, et al. 2009; Khemlani 2011). CORENET was funded by the Ministry of National Development in Singapore. CORENET may be seen as the catalyst which promoted the development of code checking systems in countries such as Norway (Statsbygg), Australia (DesignCheck) and the United States (International Code Council, ICC) (Eastman, Jae-min Lee, et al. 2009). A comparison of these and three additional code-checking systems viz. LiCA (Portugal)(Martins & Monteiro 2013), ACCBEP (Canada) (Tan et al. 2010) and GTPPM (Korea) (J. Jeong & G. Lee 2010) is shown in Table 1. This table identifies the types of *BIM Models* used and the ways of interpreting *Building Codes* that these seven code-checking systems adopt (Shih & Sher 2012). The key attributes of *BIM Models* and *Building Codes* are described below.

BIM Models: BIM data may be exported to code-checking applications in various formats such as IFC and Step (buildingSMART 2010). These provide geometric information (including points, lines and polygons) that code-checking applications may be instructed to interrogate. The BIM models of all code-checking systems shown in Table 1 adopt IFC-

Table 1 Comparison between code checking systems

	Singapore:CORENET	Norway: Statsbygg	United States: ICC	Australia: DesignCheck	Portugal: LicA	Canada: ACCBEP	Korea: GTPPM
Target rules	Building code	Accessibility	Building code	Disabilities (AS1428.1)	Water system	Building envelope	Fire resistance
Checking platform	FORNAX	SMC	DA's SMART codes for SMC, XABIO	EDM	LicA	Rule Engine	Checking engine
BIM Models							
Using IFC-based model	YES	YES	YES	YES	YES	YES	YES
Add new properties using enhanced objects	YES, called FORNAX	YES, Adding geometry data	YES, using DA's SMART codes for SMC, XABIO	YES, using internal model schema to define objects and properties	NO	NO	YES
Export IFC properties to new format	NO	NO	NO	NO	YES, using LicAXML to create XML-based model	YES, XML-based model	NO
Building codes							
Translating by programmer	YES	YES		YES	YES	YES	YES
Employs predicate logic or similar derivation process	YES		YES				
Rules coded in	Computer code	Parametric Tables	SMART code builder	Rule-based language	XML-based parametric Tables	XML based Decision Tables	Computer code
Reference	(Khemlani 2011)	(Sjøgren 2007)	(ICC 2006)	(Lan Ding et al. 2006)	(Martins & Monteiro 2013)	(Tan et al. 2010)	(J. Jeong & G. Lee 2010)

based format for communication with rule engines. Furthermore, LicA and ACCBEP have highlighted the creation of software components to extract the required information from IFC-based BIM models or to integrate IFC models into a new digital schema (e.g. ifcXML format) (Martins & Monteiro 2013; Tan et al. 2010). XML schemas are able to store and manage properties and definitions extracted from IFC files in the XML format, thereby enhancing exchangeability across applications.

Building Codes: The context and content of building codes need to be defined in logical and readable ways so that they can be related to the BIM data being checked. This involves an interpretation process where the semantic structure of each regulation is translated into rules or parametric tables. These are then interrogated and acted upon by bespoke software. Although the outcomes of these code-checking systems are similar, the techniques are varied between them (such as computer code, parametric tables, XML-based parametric tables). Among these systems, Statsbygg, LicA and ACCBEP advocate the use of XML-based parametric tables because this approach provides users with the flexibility to modify software rules without the input of professional programmers (Sjøgren 2007; Martins & Monteiro 2013; Tan et al. 2010). To date, DesignCheck (developed in 2005) is the only application that is specific to Australia. It checks for compliance against the disability codes incorporated in Australian Standard AS1428.1 of the Building Code of Australia (BCA) (L. Ding et al. 2004; Lan Ding et al. 2006). Although DesignCheck provides an advance in code compliance technology for the Australian construction industry, some challenges remain. For example, the IFC objects that are constructed and / or assembled can vary between software vendors. In addition, rule-based engines are used to interpret the building code and it is difficult for designers and/or non-computer experts to change the rules. Furthermore, DesignCheck focuses solely on disability compliance, with rules for other sections of the BCA yet to be developed. Finally, DesignCheck has no facilities for presenting checking reports in a visual format while visual information has potential to improve the design awareness and cognition (Gu et al. 2007).

The code-checking systems noted in Table 1 have significantly improved the efficiency and accuracy of code checking processes for governing bodies (e.g. CORENET has been used throughout Singapore for the compliance of building designs). However, some studies have investigated additional advantages of these systems. For example, there is merit in integrating code-checking into the design process (Rogers 2012; L. Ding et al. 2006). We are thus attempting to develop a framework for a code-checking system that is specific to Australia and will facilitate designers in various phases of the design process. This framework will target the compliance of commercial buildings with fire codes set out in the BCA. Fire codes provide a useful vehicle for such a framework because they demonstrate the integrated nature of design. Compliance requires close relationships between design geometry and the ways in which designs are interpreted to satisfy the rules contained within the code. This paper focuses on the interpretation of building codes to inform the development of Australian code-checking systems.

3. The requirements of fire code checking

The BCA, part of the National Construction Code (NCC), consists of various codes included in Volume One (Class 2 to Class 9 buildings) and Volume Two (Class 1 and Class 10 buildings) (Australian Building Codes Board 2010). The Australian Building Codes Board (ABCB), the main body responsible for the BCA, states “The goal of the BCA is to enable the achievement of nationally consistent, minimum necessary standards of relevant, health, safety (including structural safety and safety from fire), amenity and sustainability objectives efficiently.” (http://www.abcb.gov.au/)

The fire resistance of commercial buildings (class 5 to 9) is regulated through section C of the BCA Volume 1. Fire resistance deals with fire resisting construction, compartmentalisation of buildings into “fire resisting cells” and the “protection of openings” in elements that are required to be fire resisting. Section C also contains a performance hierarchy including “deemed to satisfy (DTS)” provisions, “alternative solutions” and the combinations of both of these methods. However, alternative solutions or combined methods vary from project to project. This study therefore focuses on the “deemed to satisfy” provisions.

Ensuring the safety of building occupants by catering for the eventuality of fire is most important for certifiers and developers. This involves complex procedures including the identification of fire codes for each building project. Initially a building class needs to be selected, followed by an assessment of fire ratings. Finally, building designs need to be checked to ensure they are in accordance with the codes relating to the assessed fire ratings. The following sections of this paper introduce the ways that each procedure is applied in BIM-enabled code-checking systems.

3.1 Building classes

Fire code requirements vary from building to building. The building classifications in Table 2 need to be evaluated first so that building designs can be checked against appropriate fire codes. The classes shown in Table 2 can generally be divided in two main groups: residential buildings (Class 1 to Class 4) and non-residential buildings (Class 5 to Class 10). The latter includes commercial buildings from Class 5 to Class 9. The determination of building classes is a challenge for code-checking systems because the proposed use of a building is technically difficult to determine. Although the information embedded in BIM models can represent the characteristics of geometric objects (such as walls, doors etc.) and the functions of spaces (such as meeting rooms, parking spaces etc.), it is challenging to provide criteria which will allow code-checking systems to identify the use of buildings. For example, a building consisting of meeting rooms can be used for many different purposes (e.g. an office or a laboratory). A pragmatic solution is to require users to manually define the class of building. This means users need to identify the class of buildings manually before code-checking commences.

Table 2 Classification Summary of Buildings and Structures defined in the BCA

Class		Definitions
Class 1	Class 1a	A single dwelling being a detached house, or one or more attached dwellings, each being a building, separated by a fire-resisting wall, including a row house, terrace house, town house or villa unit.
	Class 1b	A boarding house, guest house, hostel or the like with a total area of all floors not exceeding 300m2, and where not more than 12 reside, and is not located above or below another dwelling or another Class of building other than a private garage.
Class 2	A building containing 2 or more sole-occupancy units each being a separate dwelling.	
Class 3	A residential building, other than a Class 1 or 2 building, which is a common place of long term or transient living for a number of unrelated persons. Example: boarding-house, hostel, backpacker's accommodation or residential part of a hotel, motel, school or detention centre.	
Class 4	A dwelling in a building that is Class 5, 6, 7, 8 or 9 if it is the only dwelling in the building.	
Class 5	An office building used for professional or commercial purposes, excluding buildings of Class 6, 7, 8 or 9.	
Class 6	A shop or other building for the sale of goods by retail or the supply of services direct to the public. Example: café, restaurant, kiosk, hairdressers, showroom or service station.	
Class 7	Class 7a	A building which is a car park.
	Class 7b	A building which is for storage or display of goods or produce for sale by wholesale.
Class 8	A laboratory, or a building in which a handicraft or process for the production, assembling, altering, repairing, packing, finishing, or cleaning of goods or produce is carried on for trade, sale or gain.	
Class 9	A building of a public nature -	
	Class 9a	A health care building, including those parts of the building set aside as a laboratory.
	Class 9b	An assembly building, including a trade workshop, laboratory or the like, in a primary or secondary school, but excluding any other parts of the building that are of another class.
	Class 9c	An aged care building.
Class 10	A non-habitable building or structure -	
	Class 10a	A private garage, carport, shed or the like.
	Class 10b	A structure being a fence, mast, antenna, retaining or free standing wall, swimming pool or the like.

(Source from: Australian Building Codes Board 2010)

3.2 Fire ratings

In determining the fire code requirements for a commercial building it is necessary to assess the appropriate type of fire resisting construction for that building. There are three types of construction, being type A (the most fire resistant), B and C (the least fire resistant) (Australian Building Codes Board 2010). In addition to the aforementioned classification, the required type of fire resisting construction is determined based on the rise in storeys, floor area and volume of that building. The first step is determining the "preliminary type of construction" through the use of a building classification and rise in storeys. Table 3 (source from BCA Table C1.1) shows the types of construction required.

Table 3 Rise in Storeys X Class of building.

Rise in storeys	Class of building	
	2, 3, 9	5, 6, 7, 8
4 or more	A	A
3	A	B
2	B	C
1	C	C

(Source from: Australian Building Codes Board 2010)

The next step in determining the fire code requirements of a specific building is to check the floor area and volume against the building classes. Table 4 (source from BCA Table C2.2) shows that the size of any fire compartment or atrium in a Class 5 to 9 building must not exceed the relevant maximum floor area or the relevant maximum volume (floor area times internal height). In terms of measuring the floor area of a fire compartment in a BIM model, some important properties need to be defined: heating, ventilation, lift equipment, water tanks, or similar service units are not counted. The volume of the target space is required to be calculated because it effectively relates to the fire load. In the manual certification process, certifying authorities normally use the average ceiling height or average height to the underside of trusses to calculate the volume. BIM-enabled code-checking systems can improve the accuracy of these measurements when BIM models are correctly defined. After checking the building classifications, storeys, floor areas and volume, the required type of construction (fire ratings) can be generated, followed by checking the BIM model according to the appropriate fire codes. The methods of interpreting fire codes are discussed in the next section.

Table 4 Maximum size of fire compartments or atria

Classification		Type of construction of building		
		Type A	Type B	Type C
5, 9b or 9c aged care building	Max floor area -	8 000 m^2	5 500 m^2	3 000 m^2
	Max volume -	48 000 m^3	33 000 m^3	18 000 m^3
6, 7, 8 or 9a (except for patient care areas)	Max floor area -	5 000 m^2	3 500 m^2	2 000 m^2
	Max volume -	30 000 m^3	21 000 m^3	12 000 m^3
Note: See C2.5 for maximum size of compartments in patient care areas in Class 9a health care buildings				

(Source from: Australian Building Codes Board 2010)

3.3 Interpretation of fire codes

Most building-related codes and regulations are presented as well-structured documents. However their semantic complexities make it difficult to devise rules that accurately represent code clauses and the subtleties of their meaning. To address this challenge, several studies have developed different methods of analysing the characteristics of regulations or codes. In this section, two practical approaches for interpreting building codes are described.

3.3.1 RASE semantic rules (Hjelseth & Nisbet 2010)

This semantic approach enables AEC professionals to develop rules that can be applied to the semantic content of IFC-based BIM models. The general rules contained in regulations consist of more than one “check” that typically represents a section of a regulation. A check can be analysed into four constructs: *Requirement, Applicability, Selection* and *Exception (RASE)*. *Requirement* is related to the imperatives “Shall” or “Shall Not”, and a check needs to contain at least one requirement. Some specific texts are identified as the *Applicability* of the check. For instance, “internal walls” compound the “internal” and “walls” concepts. The construct *Selection* is similar but distinct from *Applicability*, which is used for alternative subjects (e.g. doors, windows and other openings). The last construct, *Exception*, is the opposite of the *Applicability*. These can be summarized as a regulation that includes more than one “check” and each check contains a number of the four constructs described above. The formulations and an example clause are shown as below (Table 5):

Table 5 RASE formulations and examples

Formulations:	Check: C0 = R0 or NOT A0 or NOT S0 or E0 Regulation: Regulation0 = C0 and C1 and C2…Cn
Example Clause: (ICC IECC 2006 502.5 Moisture control)	All framed (A) walls, floors (S) and ceiling (S) not ventilated (E) to allow moisture to escape shall be provided with an approved vapour retarder (R) having a permeance rating of 1 perm (R) or …

Legend: R – Requirement; A – Applicability; S – Selection; E – Exception; C – Check

(Adapted from: Hjelseth & Nisbet 2010)

3.3.2 Dialogue Language (DL) (Omari & Roy 1993)

According to Omari and Roy’s (1993) study, a Dialogue Language (DL) has been developed to interpret Life Safety Codes (LSC) for Australia in an expert system. It adopts a consistent interpretation to represent the code clauses as well as the interactions between users and the expert systems. The DL provides systematic structures that organize the hierarchical dialogue of codes. These structures contain eight primary items (Table 6). Among them, the *comment* is used to explain the semantic meaning of the clause text. It is related to the object defined by the code violation and assists in explaining the noncompliance of the building design to the codes. Figure 1 provides an example of the structure of a building code using the DL approach.

Table 6 the components of Dialogue Language (DL) structure

Dialog_id	An identifier which references a dialog.
Parent_id	An identifier which points to the dialog from which the current dialog was referenced.
Code_violation	The id of the object to which evaluation error messages are to be attached.
Clause	The actual text of the clause from the BCA which is embodied within the current dialog.
Condition	The DL interpretation of the conditions which must be satisfied for this dialog to be applicable.
Action	The DL interpretation of the actions to be carried out if the conditions for application of the dialog are met.
Comment	An explanatory note which describes the reasons for the application of this dialog. This field is primarily to indicate to the user, why a particular dialog has failed, in simpler terms than can normally be available from the raw code clauses. This text field is attached to the frame identified by the Code violation field to indicate non-conformance of building model elements of the BCA.
Dependency	A list of property value identifiers which is used during the evaluation of a dialog. This field can be used to indicate the values of the properties used by the dialog. As such they provide a reference by which the user can determine the exact property which is not valid.

(Adapted from Omari & Roy 1993)

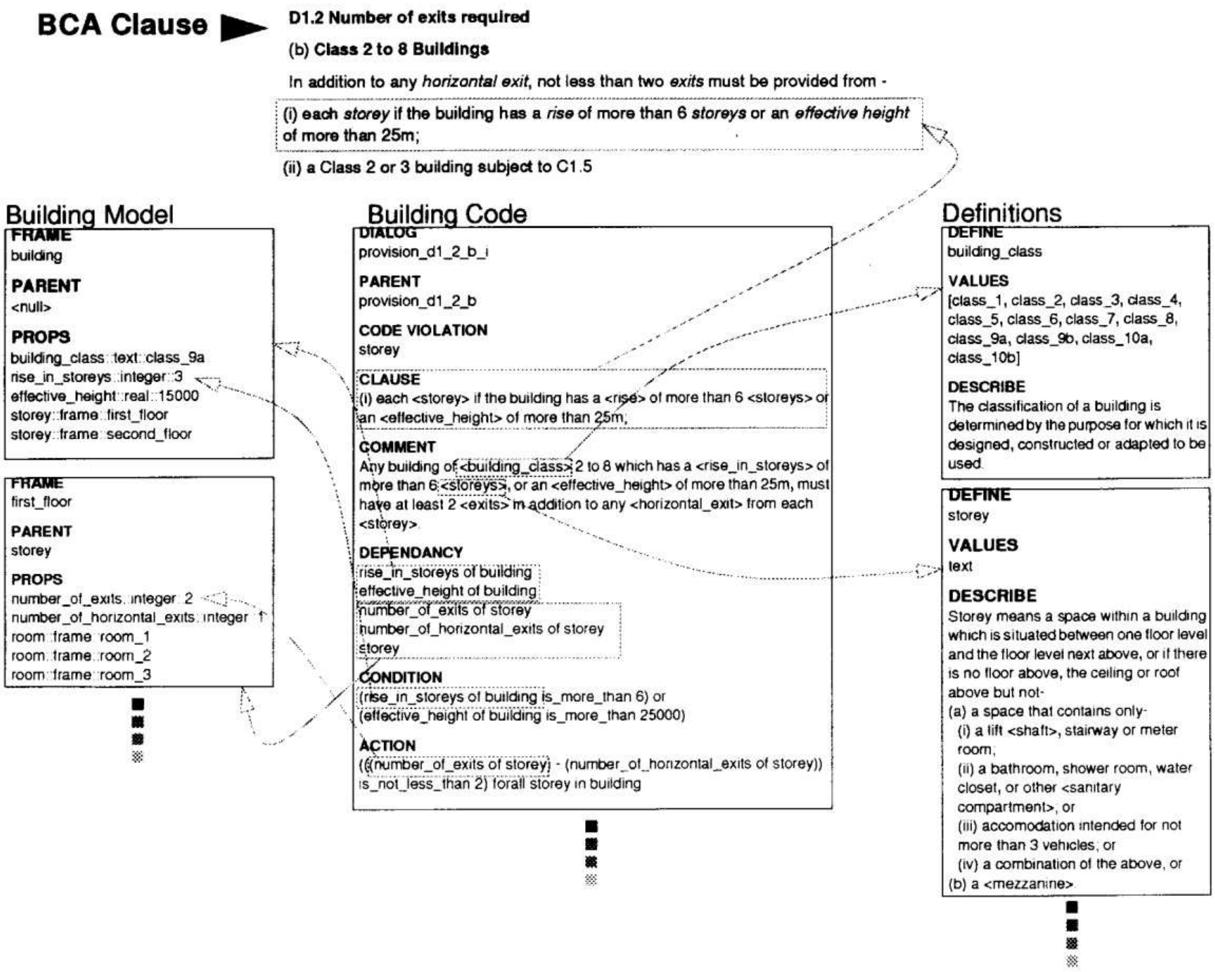

Figure 1 An example of relationships between DL elements

(Source from Omari & Roy 1993)

3.4 Discussion

The DL provides a clear structure that enables users to manage the hierarchical building codes and their interactions with BIM models. In addition, the hierarchical structure of the codes can assist in generating meaningful reports for designers. The RASE approach emphasizes the semantic logic of code clauses thereby enhancing the in-depth analysis of DL's *clause* and *comment*. Therefore, this study will adopt a combination of these two approaches as the primary tool for interpreting fire codes in our code-checking system. This will support the use of XML-based parametric tables. In addition the definition of conditions will be conducted logically using the semantic theory approach described above. Besides the straightforward rules within the building code, cross-reference rules, which result from the exceptional descriptions in the regulations, can be represented in decision tables. An XML-based parametric table provides a flexible mechanism to represent complex temporal data, and can be used to express the logic of building codes and their dependency (Noh & Gadia 2006).

Several significant challenges need to be overcome when defining the information that needs to be extracted from BIM models to allow for code-checking. The definitions of BIM models vary between software vendors. The ifcXML schema is beneficial in communicating information with XML-based interpretation of building codes. Some studies report the use of "Enhanced Objects" to add information to BIM models (Sjøgren 2007; Lan Ding et al. 2006), although this requires designers to spend additional time and effort on this task.

It is important to note that we focus on the 'deemed to satisfy' provisions of the BCA. However, assessing the compliance of building projects that require alternative solutions presents additional difficulties, particularly with commercial buildings. Alternative solutions are necessarily varied, having to accommodate the peculiarities of different performance requirements. Opportunities exist for relevant cases to be collected and arranged in a database as reference for future projects.

Finally, many studies have noted that BIM and its applications have the potential to improve collaboration between team members. The implementation of BIM-enabled code-checking systems for Australian construction industries is expected to enhance the collaborative performance of designers and certifying authorities. Further research into assessing the effectiveness of these collaborations would be beneficial.

4. Conclusion

This paper has provided an overview of existing BIM-enabled code-checking systems. It has identified the challenges of using BIM models and interpreting fire codes of the BCA. Moreover, the procedures involved in assessing the extent to which designs comply with fire codes have been investigated to inform the development of approaches of translating building codes into XML-based tables. This provides a foundation for the development of code-checking systems using BIM to assess compliance with the BCA. Not only do code-checking systems have the potential to enhance designers' awareness of building codes, they have the potential to improve collaboration and communication among project stakeholders.

References

Abrantes, V., 2010. Automated code-checking as a driver of BIM adoption. In World Congress on Housing. Santander, Spain.

Australian Building Codes Board, 2010. BCA 2010: Building Code of Australia.

Babič, N.Č., Podbreznik, P. & Rebolj, D., 2010. Integrating resource production and construction using BIM. *Automation in Construction*, 19(5), pp.539–543.

Building Professionals Board, 2012. *Case Study - Structural engineering design and construction defects (CLass 2-9 buildings)*, Sydney, Australia: Building Professional Board.

Building Professionals Board, 2011. *Guide to the building approvals process*, Building Professional Board.

buildingSMART, 2010. Model - Industry Foundation Classes (IFC).

buildingSMART Australasia, 2012. *National Building Information Modelling Initiative. Vol.1, Strategy: A strategy for the focussed adoption of building information modelling and related digital technologies and processes for the Australian built environment sector, report to the Department of Industry, Innovation, Science, Research and Tertiary Education*, Sydney.

Ding, L. et al., 2004. Automated code checking. In *CRC CI International Conference 2004*.

Ding, Lan et al., 2006. Automating code checking for building designs-DesignCheck. *Clients Driving Innovation: Moving Ideas into Practice*, pp.1–16.

Eastman, C., Lee, J., et al., 2009. Automatic rule-based checking of building designs. *Automation in Construction*, 18(8), pp.1011–1033.

Greenwood, D. et al., 2010. Automated compliance checking using building information models. In *The Construction, Building and Real Estate Research Conference of the Royal Institution of Chartered Surveyors, Paris 2nd-3rd September*.

Grilo, A. & Jardim-Goncalves, R., 2010. Value proposition on interoperability of BIM and collaborative working environments. *Automation in Construction*, 19(5), pp.522–530.

Gu, N., Williams, A. & Gül, L.F., 2007. Designing And Learning In 3d Virtual Worlds. In *Proceedings Of The Iadis International Conference On Cognition And Exploratory Learning In Digital Age (Celda 2007)*. Algarve, Portugal, pp. 227–234.

Hjelseth, E. & Nisbet, N., 2010. Exploring semantic based model checking. In *Proceedings of the 2010 27th CIB W78 International Conference*.

ICC, I., 2006. International building code. *International Code Council, Inc.(formerly BOCA, ICBO and SBCCI)*, 4051, pp.60478–5795.

Jeong, J. & Lee, G., 2010. Requirements for automated code checking for fire resistance and egress rule using BIM. In *ICCEMICCPM 2009*. pp. pp.316–322.

Khemlani, L., 2011. *CORENET e-PlanCheck: Singapore's Automated Code Checking System. AECbytes*,

Martins, J.P. & Monteiro, A., 2013. LicA: A BIM based automated code-checking application for water distribution systems. *Automation in Construction*, 29, pp.12–23.

Noh, S.-Y. & Gadia, S.K., 2006. A comparison of two approaches to utilizing XML in parametric databases for temporal data. *Information and Software Technology*, 48(9), pp.807–819.

Omari, A. & Roy, G.G., 1993. A representational scheme for design code information in an expert systems approach to building design. *Computing Systems in Engineering*, 4(2), pp.253–269.

Rogers, W., 2012. BIM Implemetation Stategies. In *Building information modeling: a strategic implementation guide for architects, engineers, constructors, and real estate asset managers*. Wiley, pp. 27–55.

Shih, S.-Y. & Sher, W., 2012. Development of Building Information Modelling enabled Code Checking Systems for Australia. In *Advancement of Construction Management and Real Estate*. CRIOCM 2012. Shenzhen, China.

Sjøgren, J., 2007. *buildingSMART–a smart way for implementation of standards*,

Succar, B., 2009. Building information modelling framework: A research and delivery foundation for industry stakeholders. *Automation in Construction*, 18(3), pp.357–375.

Tan, X., Hammad, A. & Fazio, P., 2010. Automated code compliance checking for building envelope design. *Journal of Computing in Civil Engineering*, 24(2), pp.203–211.

Yang, Q.Z. & Li, X., 2001. Representation and execution of building codes for automated code checking. In *Proceedings of CAAD Futures*. pp. 315–329.