Regulatory Knowledge Encoding Guidelines for Automated Compliance Audit of Building Engineering Design

Johannes Dimyadi[1], Charles Clifton[2], Michael Spearpoint[3] and Robert Amor[4]

[1]Department of Computer Science, University of Auckland, Auckland, New Zealand; email: jdim006@aucklanduni.ac.nz
[2]Department of Civil Engineering, University of Auckland, Auckland, New Zealand; email: c.clifton@auckland.ac.nz
[3]Department of Civil and Natural Resources Engineering, University of Canterbury, Christchurch, New Zealand; email: michael.spearpoint@canterbury.ac.nz
[4]Department of Computer Science, University of Auckland, Auckland, New Zealand; email: trebor@cs.auckland.ac.nz

ABSTRACT

The main challenges in automating the regulatory compliance checking of building engineering designs are the availability of computable representations of the building and the regulatory knowledge, as well as a system that can process and manage these representations effectively. The emergence of Building Information Modelling (BIM) and Industry Foundation Classes (IFC) at the start of the millennium has sparked useful research in the area of sharing building information effectively, but challenges remain with producing a practical and manageable regulatory knowledge representation that can be processed effectively by a compliance checking system.

Research is being conducted to develop a two-part regulatory knowledge representation, which can be maintained independently by designers and regulators. One part is a set of compliant design procedures modelled as Business Process Diagrams (BPD) using an open standard Business Process Model and Notation (BPMN), and the other is the associated regulatory constraints and rules encoded in a computable format suitable for execution with the BPMN.

This paper reports on a set of guidelines developed for the purposes of encoding regulatory knowledge into the proposed computable representation. A verification method (C/VM2) prescribed by the New Zealand Building Code (NZBC) for the performance-based design of buildings related to fire safety has been selected as a case study to illustrate the encoding process. These guidelines are adaptable for encoding the entire NZBC.

BACKGROUND

Regulatory Knowledge Representation. Building codes and design standards are typically written in natural language for human interpretation. There have been numerous attempts to replicate regulatory texts as digital representations for computer processing. The most common approach to date has been rule-based systems (Eastman et al., 2009). Other approaches reported in the literature include the application of artificial intelligence and semantic modelling techniques to allow computers to interpret complex legal texts (Salama and El-Gohary, 2011; Zhang and El-Gohary, 2012). There has also been research on utilising document mark-up techniques to assist with knowledge extraction (Hjelseth and Nisbet, 2011). Although there have been some promising research outputs in this area, there is a common practice to integrate regulatory knowledge representation into the compliance checking system. Consequently, it has become tedious and costly to keep the representation up to date with frequently revised regulatory texts. The need to keep the regulatory knowledge representation independent of the checking system has been identified as a key factor for a successful implementation (Greenwood et al., 2010).

Generally, there are two types of knowledge inherent in regulatory texts, namely a set of compliant design logic or procedures and the associated constraints and rules. Different aspects of a design call for different compliant design procedures which are effectively the same set of procedures often followed intuitively by designers during the design process (Dimyadi and Amor, 2013). Instead of focusing on automatically extracting knowledge from regulatory texts, the current research proposes developing a digital library of compliant design procedures in Business Process Model and Notation (BPMN) version 2.0 (Object Management Group, 2011), which can be maintained independently by designers and used as an input component to the compliance checker. The associated regulatory data, constraints and rules can be encoded into a computable format such as Drools Rule Language (DRL) (De Ley and Jacobs, 2011), which is suitable for BPMN execution in a computing environment.

New Zealand Performance-based Building Code. The NZBC is part of Building Regulations made under the Building Act 2004, which is the New Zealand's official legislation. It is a performance-based code covering all aspects of building work including fire safety, structural stability, health and safety, accessibility, moisture control, durability, energy efficiency, etc.

A performance-based code does not prescribe how a design and construction process should be carried out, but provides technical clauses that specify functional requirements, and the qualitative or quantitative performance criteria to which the completed building and its components must meet throughout its intended life. This allows for innovation and uniqueness in designs proven by established scientific and engineering principles. Performance-based codes are usually accompanied by a set of prescriptive requirements, which are deemed to satisfy the performance criteria, to facilitate the compliance of common building designs.

The NZBC allows three different means of compliance. The first is the "Acceptable Solution" (AS), which must demonstrate full compliance with a set of prescriptive requirements. The second is the "Verification Method" (VM), which must demonstrate compliance by means of verification with a set of prescribed computational or design methods. Beyond AS and VM, there is the "Alternative Solution" by means of a proven and peer-reviewed engineering design, which may involve specific numerical computations, simulations, and/or laboratory tests and certifications.

Verification Method C/VM2. New Zealand has recently pioneered the incorporation of a new Verification Method (C/VM2) (Ministry of Business Innovation and Employment, 2013) into NZBC for the fire safety design of buildings. This is based on a set of industry accepted engineering design guidelines and computational procedures for use to evaluate ten different design scenarios, which are prescribed fire events that will govern the design and fire safety systems in a building. For example, “Blocked Exit” (BE) scenario where an escape route could be blocked by a fire, “Unknown Threat” (UT) where a fire starts in a normally unoccupied space and can potentially endanger a large number of occupants in another space, “Concealed Space” (CS) where a fire starts in a concealed space and can potentially endanger a large number of people in another space, “Smouldering Fire” (SF) where a fire is smouldering near a sleeping area, “Fire Fighting Operation” (FO) scenario where the safe operation of firefighters in a building is to be verified, etc.

C/VM2 Document Structure and Naming Convention. C/VM2 document consists of four parts containing regulatory texts that are arranged into numbered paragraphs. Part 1 contains a list of vocabulary and definitions, as well as stating the objectives and performance criteria. Parts 2 and 3 provide common rules and parameters for the analysis of design scenarios and movement of people. The rules and parameters are mainly provided in the form of tabulated data and mathematical equations, which are embedded in the texts together with various constraints and conditions. Part 4 contains ten paragraphs, numbered 4.1 to 4.10, which specifies the compliance method relevant to each design scenario.

ENCODING REGULATORY KNOWLEDGE

Modelling Design Procedures in BPMN. **A subset of BPMN 2.0 components considered most essential for describing the compliant design logic in C/VM2 is given in**
Table 1.

Case Study. The “Unknown Threat” (UT) design scenario is selected as a case study to illustrate the modelling process. This is a scenario where a fire could start and grow to a significant size in a normally unoccupied space such as a store room, cleaner’s cupboard, etc. (*Fire Space* or the space of fire origin), and can potentially spread and endanger occupants in a target occupied space (*Target Space*), which is defined as follows:

1. It is connected or located adjacent to the *Fire Space*.
2. It has an occupant load (OL) > 50 persons.

For encoding purposes, the term *Fire Space* is used to describe the space of fire origin, which is applicable to many design scenarios. The expected outcome of the UT scenario assessment is a verification that either occupants in the *Target Space* can escape safely, or the fire can be confined to the *Fire Space*. These are the two criteria inherent in the text of Paragraph 4.2 of C/VM2 (Table 2) to be encoded.

Table 1: Proposed subset of BPMN objects

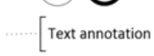
Object	Description
	User Task, an atomic activity where human action is required, e.g. enter certain information to continue. The description is a brief comment about the task.
	Script Task, an atomic activity that executes an external script written in a language such as Javascript. The description is a brief comment about the task.
	Business Rule Task, an atomic activity that executes an external rule written in compatible languages such as the DRL. The description is a comment about the task.
	Call Activity, a task that calls external sub-procedure. This allows for reusable standard sub-procedures. The description is a brief comment about the task.
	Sequence flow, which defines the normal execution order of activities
	Default sequence flow, which is the default branch of activity to be processed if all other conditions evaluate to FALSE
	Conditional flow, has a condition assigned to the branch that determines whether or not it should be executed. If not, the default flow will be executed instead.
	Parallel (AND) Gateway. Create a fork where two activities can run in parallel; create a join that synchronises or combines parallel activities.
	Data/event based exclusive (XOR) Gateway (or Decision). Diverge to a particular path depending on which option is selected option; converge to a single output.
	Inclusive (OR) gateway. Diverge to one or more paths based on a condition. A default path must be specified and is only followed when no other conditions are met; converge to a single output using only the first input signal that arrives.
	Start event; Intermediate event; and End event, respectively.
	Text Annotation. Additional human readable non-executable remarks.

Table 2: Regulatory text on the method of evaluating UT design scenario

Paragraph 4.2 of C/VM2	Either: a) Carry out ASET/RSET(*) analysis to show that the occupants within target spaces are not exposed to untenable conditions, or b) Include *separating elements* or fire suppression to confine the fire to the room of origin. If *separating elements* are used the FRR(**) shall be based on the following design criteria: i. If no automatic fire detection is installed in the space of fire origin, separating elements shall have fire resistance to withstand a full burnout fire (Paragraph 2.4) ii. If automatic fire detection is installed in the space of fire origin, separating elements shall either: 1. Have a fire resistance rating of not less than 60 minutes (-/60/60), or 2. Demonstrate the separating elements will be effective for the period from ignition to the time when the occupied space (target space) is evacuated.

(*) *ASET=Available Safe Escape Time, RSET=Required Safe Escape Time*
(**) *FRR=Fire Resistance Rating*

Fire Space and *Target Space* objects together with their physical and geometric properties such as the floor area, occupant load, usage classification, etc. are the input parameters in this design scenario, which could be obtained using a spatial query from a BIM model. The task for getting the input parameters is an atomic activity represented by a Business Rule Task, which would execute external rules written in a compatible language such as DRL for the task. A general procedure representing the methods prescribed by Paragraph 4.2 of C/VM2 can be expressed graphically in BPMN as shown in Figure 1. The two evaluation methods of the design scenario are represented by the conditional and default flows in the main procedure, which is identified as Procedure 4.2C/VM2 (Figure 1). Each space object that has been checked is marked as such and is no longer available for selection. For automation, the entire procedure can be set to repeat until all Fire Space and Target Space objects on each level of a building have been checked.

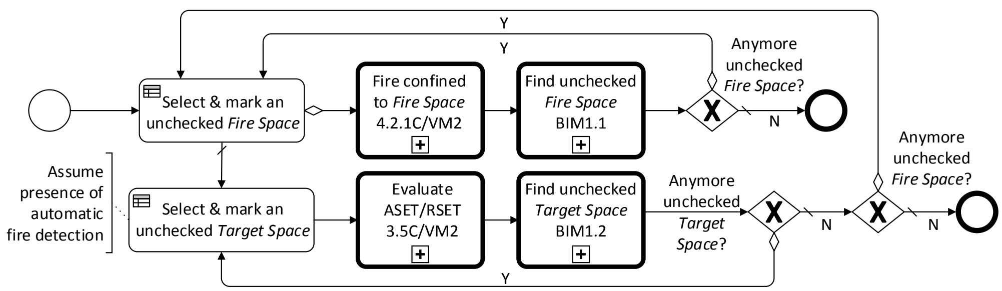


Figure 1. Procedure 4.2C/VM2 – to evaluate UT design scenario

The method to assess if the fire can be confined to a *Fire Space* is a sub-procedure, identified as Procedure 4.2.1C/VM2 (Figure 2), which is called by the main procedure, Procedure 4.2C/VM2. "Separating element" (or *Fire Barrier*, as used in this paper) is a barrier such as a wall having fire resistance rating (FRR) that provides design resistance to the spread of fire. The method to check for the effectiveness of *Fire Barrier* is also a sub-procedure, identified as Procedure 4.2.2C/VM2 (Figure 3).

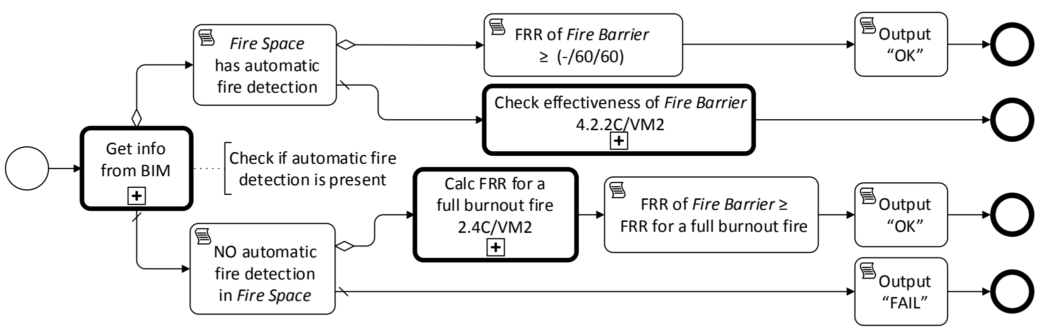


Figure 2. Procedure 4.2.1C/VM2 – to check fire confinement in *Fire Space*

The sub-procedure to calculate FRR for a full burnout fire (Procedure 2.4C/VM2) is not covered in this paper, but it consists of three different evaluation options that may involve interfacing with external computational tools, which will be covered in the future work. Similarly, the task to calculate *Fire Barrier* burnout time is also not covered in this paper. These sub-procedures may look up tabulated data or call certain regulatory rules to evaluate. Some examples of formalised rules are given in Table 4. The output activity is typically a Script Task, which is most suited for executing simple actions, or evaluating mathematical equations or functions.

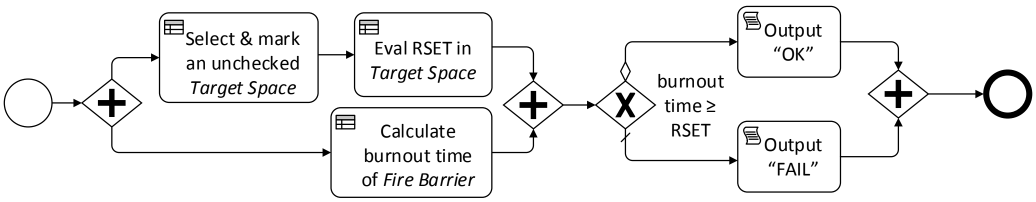


Figure 3. Procedure 4.2.2C/VM2 – to check *Fire Barrier* effectiveness

ASET/RSET analysis is the other method of verification for this design scenario. The objective of the analysis is to establish that the required safe egress time ($RSET$) or the evacuation time from each target space is less than the available safe egress time ($ASET$) or the time it takes for the escape route from that space to become untenable due to the effects of the fire, i.e. $ASET > RSET$. This is expressed by a sub-procedure as shown in Figure 4. The task to evaluate *ASET* or *RSET* is a call to external sub-procedure, which may involve interfacing with external simulation tools. This will be covered in the future work.

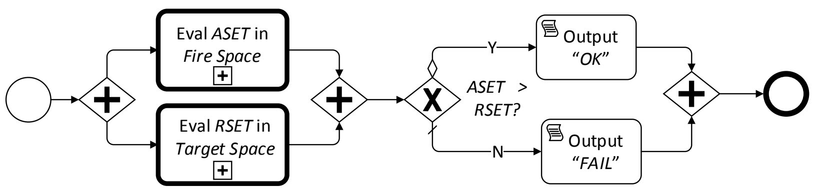


Figure 4. Procedure 3.5C/VM2 – to evaluate ASET/RSET

NZBC Clauses C4.3 and C4.4 provide the performance criteria for tenability conditions that must be satisfied. There are actually three separate criteria in C4.3, which can split as C4.3a, C4.3b, and C4.3c accordingly (Table 3). FED is the fraction of the dose (of CO or thermal effects) that would render a person of average susceptibility incapable of escape. This is typically obtained from numerical computations or simulations.

Table 3: Performance criteria to be satisfied

NZBC Clause	Performance criteria
C4.3	The evacuation time must allow occupants of a building to move to a place of safety in the event of fire so that occupants are not exposed to any of the conditions specified in C4.3a, C4.3b, and C4.3c
C4.3a	A fractional effective dose (FED) of carbon monoxide (CO) > 0.3
C4.3b	FED of thermal effects > 0.3
C4.3c	Conditions where, due to smoke obscuration, visibility < 10 m, except in spaces < 100 m^2 where visibility < 5 m
C4.4	If *OL* < 1000 and the space is protected by an automatic fire sprinklers system, then criteria C4.3b and C4.3c do not apply.

Formalised Regulatory Rules. Regulatory rules must be formalised to enable encoding into computable formats. For example, paragraph 3.2.6 of C/VM2 states "*Doors on escape routes shall be hung to open in the direction of escape and, where escape may be in either direction, doors shall swing both ways. These requirements need not apply where the number of occupants of spaces with egress using the door is no greater than 50. Manual sliding doors are permitted where the relevant number of occupants is no more than 20*". This paragraph actually contains two separate rules, formalised as Rules 3.2.6aC/VM2 and 3.2.6bC/VM2 (Table 4). However, some rules are more difficult to encode and may require interpretations on their intent.

Table 4: Exemplar formalised regulatory data and rules from C/VM2

Rule ID	Formalised rules
2.3.2 C/VM2	If average upper layer temperature first reaches 500°C, then "flashover" occurs.
3.2 C/VM2	$RSET = (t_d + t_n + t_{pre}) + (t_{trav} \text{ or } t_{flow})$
3.2.3 C/VM2	Given the space usage, look up Table 3.3 C/VM2 for pre-travel activity time (t_{pre})
3.2.4 C/VM2	Travel time is the greater of t_{trav} or t_{flow}, where t_{trav} = time taken (s) to travel to the doorway t_{flow} = time taken (s) for all occupants to flow through a doorway, if queuing occurs Equation 3.2C/VM2: $S = (k - a \times k \times OD)$ Equation 3.3C/VM2: $t_{trav} = L_{trav}/S$ where: S = horizontal travel speed (m/s), OD = Occupant Density (persons/m^2) Equation 3.3.1C/VM2 (implied): $L_{trav} = (Length_{room} + Width_{room})$ For horizontal travel: $S = 1.2$ m/s maximum, $k = 1.4$, $a = 0.266$ For vertical travel, use stairs geometry (tread and riser) and Table 3.4 to find k
3.2.6a C/VM2	If $OL > 50$, doors on escape route must open in the direction of escape
3.2.6b C/VM2	If $OL \leq 20$, sliding door is permitted to be used on escape route.
3.2.7 C/VM2	Design any primary entrance to egress 50% of total OL of the space, remaining occupants to be evenly distributed in proportion to the number of exits

For naming convention, every rule is given the same number as the containing text paragraph. Mathematical equations are given the same number as they appear in the document, except for those derived from the texts where they are given a number related to either the parent equation or paragraph.

CONCLUSION

A set of guidelines for encoding regulatory knowledge for automated compliance checking has been given in this paper. A subset of BPMN 2.0 objects is proposed for use to represent compliant design procedures graphically. To illustrate the modelling process, a performance-based design scenario prescribed by the C/VM2 of NZBC has been selected due to its unique approach to compliance, which allows for interfacing with external computational or simulation tools. The use of an open standard graphical approach such as BPMN to describe these procedures would make them relatively easy to construct and maintain with the guidelines provided.

Rules that are embedded in the regulatory texts can be formalised as illustrated and encoded accordingly for use by some of the Business Rule Tasks together with lookup tables that are encoded from tabulated regulatory data. The proposed method of encoding regulatory knowledge illustrated here is adaptable for other performance-based criteria in different design scenarios. Subjective criteria that require human intervention such as those involve laboratory testing, cost-benefit analysis or expert judgement may also be incorporated using the User Task component. The proposed method described here will be subject to a reliability and performance assessment.

ACKNOWLEDGEMENT

This research receives funding assistance from the New Zealand Building Research Levy.

REFERENCES

De Ley, E., and Jacobs, D. (2011). Rules-based analysis with JBoss Drools: adding intelligence to automation. In *Proceedings of ICALEPCS 2011* (pp. 790–793). Grenoble, France: Software Technology Evolution.

Dimyadi, J., and Amor, R. (2013). Regulatory Knowledge Representation for Automated Compliance Audit of BIM-based models. In *Proceedings of the 30th CIB W78 International Conference* (pp. 68–78). Beijing, China.

Eastman, C., Lee, J., Jeong, Y., and Lee, J. (2009). Automatic rule-based checking of building designs. *Automation in Construction*, *18*(8), 1011–1033. doi:10.1016/j.autcon.2009.07.002

Greenwood, D., Lockley, S. S. S., Malsane, S., and Matthews, J. (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, France. Retrieved from http://nrl.northumbria.ac.uk/6955/

Hjelseth, E., and Nisbet, N. (2011). Capturing normative constraints by use of the semantic mark-up (RASE) methodology. In *Proceedings of CIB W78-W102 Conference* (pp. 1–10). Sophia Antipolis, France.

Ministry of Business Innovation and Employment. (2013). *C/VM2 Verification Method: Framework for Fire Safety Design For New Zealand Building Code Clauses C1-C6 Protection from Fire* (Errata2 ed., pp. 1–65). Wellington, New Zealand: The Ministry of Business, Innovation and Employment.

Object Management Group. (2011). *Business Process Model and Notation (BPMN)* (pp. 1–538). Object Management Group.

Salama, D. M., and El-Gohary, N. (2011). Semantic modeling for automated compliance checking. *Journal of Computing in Civil Engineering*, 641–648.

Zhang, J., and El-Gohary, N. (2012). Extraction of Construction Regulatory Requirements from Textual Documents Using Natural Language Processing Techniques. *Journal of Computing in Civil Engineering*, 453–460.