OVERVIEW OF CONCEPTS FOR MODEL CHECKING

Eilif Hjelseth, Ph.D. student, eilif.hjelseth@umb.no
Norwegian University of Life Sciences (UMB), Dept. of Mathematical Sciences and Technology, Norway
Nick Nisbet, Director, nn@aec3.com
AEC3 UK Ltd, United Kingdom

ABSTRACT

This paper presents an overview of concepts for model checking. This have resulted in a description of four different concepts based on the intention for checking. The four concepts are: a) Validating systems, b) Guiding systems, c) Adaptive systems and d) Content based checking. By use of an ontological approach we propose a four level taxonomy of model checking 1) Intention, 2) Result, 3) Rule set and 4) Type of products. Model checking should be regarded as a knowledge system for support of the design process.

Keywords: checking, ontology, BIM

1. INTRODUCTION

This paper develops an overview of concepts for model checking in BIM (building information models). The complexity in the design, building and facility management process is increasing. According to Ingvaldsen, (1994, 2001) can as much as 40% of the defects can be related to blunders in the design process. Model checking software has achieved increased interest and is often regarded as one of the big benefits by using BIM / IFC based software in the design process (SmartMarket Report, 2009). Model checking is normally done by use of stand-alone applications as Solibri Model Checker, SMARTcodes, ePlanCheck, AEC3 Compliance or EDM Model Server (Bell, Bjørkhaug and Hjelseth, 2009).

An often used example of model checking is clash detection to validate if for example different types of pipes intersect each other. Another example can be to check if the width of the doors is according to codes of accessibility. However, there is a common perception that model checking is about validation and yes/no answers.

The full range of model checking concepts can be regarded as a framework for utilization of existing “knowledge systems”. The use of this must support the focus on doing the right things – and doing the thing right. To do this, a number of approaches have to used. Different concepts of model checking (not only validating) can be contribution in this effort. There is reason to assume that too much validation and code compliance can be contra-productive with respect to development of new ideas and solutions of the built environment. Supplementary concepts with other intention can be introduced.

2. THE FRAMEWORK OF CONCEPTS FOR MODEL CHECKING

Model checking used wisely is a way to share and utilize knowledge. But model checking can easily be misused by thinking that passing a validation check proves a good design of the building. At its best, one can avoid bad design solutions. This is not a bad target itself. Model checking can be regarded as a part of a knowledge system. With an ontological foundation, is should be possible to build a scalable system with trustworthy result. The challenge is to reduce the "knowledge soup" made of; a) Over-generalizations, b) Incomplete definitions, c) Conflicting defaults and d) Unanticipated applications. (Sowa, 2006).

The AEC industry can also learn from other industries when implementing these expert systems. Experiences from the oil and gas industry, model checkers were one of the main levers for utilizing product models. The increased interest for BIM and interoperability through open standards as IFC will be a driving force enabling model checking for a large number of design software based on IFC-export / import. BIM based model checking is based on processing of information supported by a semantic system for automatic generation of applicable code. This will according to Hjelseth and Nisbet (2010) enabling a cost and time effective solution.

There are several terms in use such as rule checking, validating, code compliance checking for what we have defined as "Model checking". Model checking is used as the general term for several reasons. The checking is performed on a "model" and the information it contains[1)]. This will often be a file in IFC or ifcXML format from design software. It also gives a reference to utilization of BIM. A look-up in the ISO Concept Database (ISO, 2010) do not contain any of the terms above. The term Model checking is well described on Wikipedia, while rule checking and code checking is not. (Wiki, 2010). This is an indication for use of model checking as the preferred term for this function. In general, ontology is the study or concern about what kinds of things exist. In information technology, ontology is the working model of entities and interactions in some particular domain of knowledge or practices, such as electronic commerce or "the activity of planning." In artificial intelligence (AI), ontology is, according to Tom Gruber, an AI specialist at Stanford University, "An ontology is an explicit specification of conceptualizations, used to help programs and humans share knowledge (Gruber, 1993). The knowledge perspective is also supported by Hendler et. al. (2000) who almost equates ontology with knowledge base. Shakeri et.al. (2001) present in Figure 1. the relations from meta-model and ontology to knowledge model.

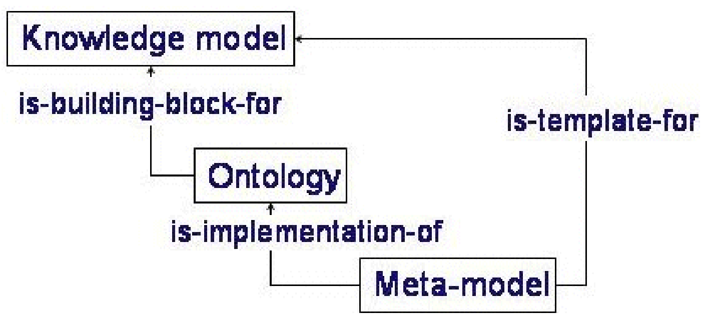

Figure 1: Knowledge model – Ontology – Meta-models (Shakeri et.al. 2001).

We propose a framework based on four different "intentions": a) Validating, b) Guiding, c) Adaptive design and d) Content based checking. The presented concepts are described by general terms that are a part of a class (classification) who is defined by taxonomy as part of an ontology. Based on this we want to present and ontology for classification model checking system. These four concepts are described in the following chapters.

---

1 This makes a distinction to database and text based search systems or similar based AI (artificial intelligence) systems. The information in a model is also ordered in a pre-defined system, as with IFC where slabs and walls are related to a defined floor

3. VALIDATING MODEL CHECKING

The purpose with a validating checking is to determine if the content in the model is in accordance to a code, standard, regulative etc. Validating model checking is performed on predefined criteria and result in an Yes / No answer. The rule will have constraints with which set the limitations of values or existence / not existence of an object. To pass a validity test, the rules must not be activated. This indicates that missing rules or invalid constraints can make a model pass the test. For trustworthy results, the user must be aware of what is being checked and what is not. If not, it becomes too easy to pass an automatic model check which will discredit automatic model checking systems. An example is automatic accessibility checking. This cannot be checked automatically without a definition of accessibility which involves a large number of rules. However, if rules for accessibility are based on a defined standard, checking within this domain of computable rules can be done trustworthy (Hjelseth, 2009b).

- Geometry based checking – Clash detection

Clash detection and checking of component pairs is maybe the most known concept of checking. When merging models in interdisciplinary collaboration models, this kind of checking is very useful and often included as part of the quality assessment (QA) system. The algorithm (rule) for checking the turning circle of a wheelchair, is the same rule that can be used to check minimum / maximum distance from any object to any other object. By enabling parametric selection of object (in addition to tolerance) it is possible for the designer to check and correct other distances for example between cabinets for fire extinguisher. The checking is based on topological relationships and Boolean algebra. These rules can also be implemented parametrically, allowing the user to adjust the “rule” by changing the min / max tolerances the components are checked against. (Borrmann, 2008). Because of on its geometrical nature, automatic checking can also be easily re-examined by just looking at the 3D model from different angles and sections, and then mark if this is a failure to be solved or just an over reporting. The checking is in practice a semi-automatic process with high degree of transparency in the checking process.

Clash detection checking has a tendency to report too many "issues". An example of this is when a pipe goes through six beams, is this one issue or six? Or if a beam goes through six pipes, is this one issue or six? Normally one will get six reports of errors in both cases, but the answer should depend on rules and the information about the pipe. In an office building, the structural beams may have priority and the expected solution is re-positioning the pipes. In an oil platform the pipes may have priority and the structural solution must adapt. This example also illustrates the relation between saying what is wrong and what must be done to correct. In software systems one will combine more of the concepts presented here. But without a clear understanding of what the system does (and does not), reporting of errors, suggestion to solutions, the information used in the rules and the rule itself.

- Compliance checking / rule checking

The purpose is to check if the solutions in the model are in accordance with codes, regulations, standards and so on. Automatic management of building permit application has long been a beacon for model checking. One reason is that permitting is a critical point that all facilities have to pass. The first large scale implementation was in Singapore in the CoreNet project for assessment of building applications. Developments at the US International Code Council may lead to another example of large scale implementation.

The rules are often clustered into rule-sets to determining the domain to be checked. By combining different rule sets, it is possible to check multiple demands automatically. We do not go into depth on procedures for development of rules. This is a large area with connection to KBE, knowledge based engineering and to AI, artificial intelligence (Hjelseth, 2009b). On the other hand, we are now beginning to get standards based on BIM and with AEC industry based approach in the series of IDM standards. The IDM standard,ISO 29481-1:2010, contains following components: Process maps (PM), Exchange requirement (ER), Functional Parts (FP) and Validation Rules (VR), and Business Rules (BR).

4. GUIDING MODEL CHECKING

The intention with this concept is to guide the designer to consider a larger range of realistic solutions than is practical without this support. It is particularly relevant in areas where the designer is not the expert or experienced. The checking is based on two elements: Identify rules for the situations where "problems occur", and second, present a list of possible actions. The rules are activated on defined situations in the model, which provides a range of possible, but relevant questions / solutions. A situation can be identified by a large number of elements. This can be visualized as a decision tree where you somewhere in the middle, getting suggestions for next branches. As an example of complete model guide:

In Norway ->
Dwelling house ->
The area is above X m^2 _>
A chimney and open fireplace is modeled.
Next options to be considered:

1) Add a stove for heating.
2) Increase the volume of concrete so it act as a heating reservoir.
3) supplement with a technical room for wood pellet heating in lowest floor.

There can be links from the suggestions to more information, details and calculations, experiences and examples.

This concept for model checking can also be regarded as a “learning system” or “experience utilization”. Manual version of this exists in check–list or collection of product documentation and old projects. The problem is that this is within your own domains, and often as “tacit knowledge”. The challenge is how to let other use this “experience” and to “remember” to use this (identify the problems and find solutions) in practice in an on-going project. Another aspect is that it motivated to go one step further before one turn down a solution being to expensive, complicated, not possible here – and so on. This checking can be done in all stages and all actors in the design process.

- Pre defined solution checking

This checking identifies a situation and suggests a predefined solution. This solution can be related to product or to work (assembly) process. This concept will normally be based on automatic lookup in the knowledge library as e.g. the Byggforsk knowledge system (BKS) from Sintef Byggforsk (2010). The list of “accepted” solutions must then be further be analyzed.

- Search based solution checking – IFD

This concept of checking is based on search based on concept. IFD is classification concept based on relevant products is mapped against a concept, e.g. a door. This concept, generic door, has attributes which make it possible to match this against real products. A search in the IFD library compliant product databases will list the possible products. The search can be developed as a rule and run automatic. An example is searching for a “Door” this is in IFD defined as what in UK is defined as a Door set , and in Norway as a door leaf + door-casing. The criteria is geometry: to fit into an opening in the wall that is 900 mm wide and 210 mm high, opening direction, Fire rating class, Outdoor with lock, environmental labels / ECO labels and so on. General policies as lowest price, local vendor or vendor from a predefined list can also be included (Hjelseth 2009c).

5. ADAPTIVE MODEL CHECKING

The intention here is to let the object itself register its environment and adapt automatically to this by following embedded pre-defined rules... The design process is dynamic and it is therefore a big challenge to update all consequences of an update. This concept of model checking will often be very useful on areas / domains where you are not the expert. There are two levels of adaptive model checking; object and system level.

- Adaptive object / parametric objects/ intelligent objects (object level)
An example of this concept is automatic adaptation of the diameter of column related to number of floors, and definition of Zone – Load per area. The column change diameter automatically. The adaptation can also be a concrete with compressive strength, or more reinforcement. This adaptation can be based on calculations, heuristics or preferred solutions.

Another example is based on fire rating where doors and window specifications change dependent on the fire rating of the walls in which they occur, or the spaces they bound. If you try another layout or change area, then you change fire rating values – and for consistency this changes the specification of the doors, windows and walls etc.

- Complete building concept adaptation (system level)
A commercial example of this is form the company Selvaag Bluethink who has developed the KBE (Lisp) based application “House Designer”. This is an example of how comprehensive this concept can be developed. Based on a draft layout, House Designer can suggest a very detailed design of the entire building. The profile of solutions is related to which rule sets that are activated. The rule sets can be different owner-defined requirements (standards / product types), different public standards (from different countries) and other requirements. (Opdahl and Olsen, 2009).

6. MODEL CONTENT CHECKING

The intention here is to examine the professional content in a BIM model for a specified use. There is no need for programming rules, just "filters" for reporting relevant information. This can be further analyzed in software as spreadsheets, word processors or databases. This gives this concept a large flexibility.

Today this can be done very easily and manually by model viewers, or with more complication using model servers. User friendly solutions between these solutions are needed.

- Content of information compliance / IDM compliance
The focus on this check is the content of information in the model compared to a requirement. This requirement can be defined in an IDM (ISO 29481-1:2010), in a BIM guide or specified as a separate project delivery. A general example is that all walls, windows, roofs and slab types shall contain information about energy performance (U-value), noise reduction and fire rating.

- Client demand / space program / design program
This is a variant of is based on comparing demands (constraints) from user specification with information in the model. Statsbygg - Public Construction and Property, Norway (2010) offered this as a service in phase 2 of the international architecture competition in the new National Museum in Oslo.

- Not done checker / Next domain checker
This is a version of the two examples above. Most checkers focus on what is in the model. One situation is that missing or forgotten technical equipments can give an impression that there is enough space). This can be done by replacing the demand list with a list of "default content" related to defined solutions. The default list can be based on experiences or just generally solutions (objects / attributes for defined solutions such as specified spaces; a surgery theatre (of specified type), occupy Xm^2 of equipment. This can be used for correctives of the space program.

- Model comparison checking / Deviation analysis

This model checking solution read the information in one model, and compares this with information in another model. The rule is based on compare demands. The rules is defining what one want to compare, such as; area or space, building or defined spaces, area of external wall, area of window, content of defined objects (HVAC). The results from this checking can be further analyzed.

- Design process checking – Status

This is a check of metadata related to the object. So far this information is not present in design software today, but a system for this was presented by Hjelseth (2009a) in the paper "Exchange of relevant information in BIM-objects defined by the Life cycle Information Model (LIM)". A design process, and its objects, is passing through different stages (ISO 22263:2008) from proposal to demolishing . The change of status can be checked for keep track of the process. This will also give improved control of clients demand and how these are satisfied (Kiviniemi, 2005).

- Work progress checking – Schedule

This metadata based concept for model checking can also be applied on the building site for checking the information in the real objects with the information in the schedule model. The challenge is to get real time data in a cost effective way. One solution is to use RFID (radio frequency identification) for automatic identification and data capture enables a real time logistics and makes is possible for having an “as-building” model (Hjelseth, 2010).

7. DISCUSSION

This paper use semantic arguments to propose the term model checking as identification for this domain of functionality. Many of the used terms as e.g. code checking has a limited scope, (here, checking only codes, not standards.) but are use for all types of checking. Whether the term model checking will be taken into the common language is dependent on a large number of circumstances outside this paper. However, some kind of systematic overview will be useful as counterweight to the mess in terms and brandings used today. In this respect we have introduced model checking as the term for this domain of function. The lack of formal definition in ISO etc. in this new area makes it easy to introduce own terms and brandings.

This paper do also propose four top level concepts of model checking. The question is whether the ontological structure is correct used itself, and if the number of four is complete as top level concepts. The ontology is based on the relation between four criteria; Intention, Result, Rule set and Type of product. We have been able to use this a taxonomy for classification within of all concepts (top level class) and in this respect find it useful for identification of meaning regardless of a term, brand or abbreviation.

The result of this approach based on its intention is that we need only four top level classes of model checking concepts. The limited number can be discussed, but should be kept as low as possible to reduce interference between the concepts. The number of four is the lowest number we have been able to define as concepts. In practical use will often lower level terms, or commercial branding, be used as description. We have for each concept described some sub classes for illustrating practical utilization. The proposed terms is only an description, not a definition, and multiple terms are therefore used. However, by use of the criteria for model checking, it should be possible determine which kind of model checking who is mentioned.

8. CONCLUSION

This paper gives an overview of model checking concepts based on its intentions. All concepts are based on an ontological foundation with a four level taxonomy consisting of *Intention – Result – Rule set – Type of products*, illustrated in figure 2 below.

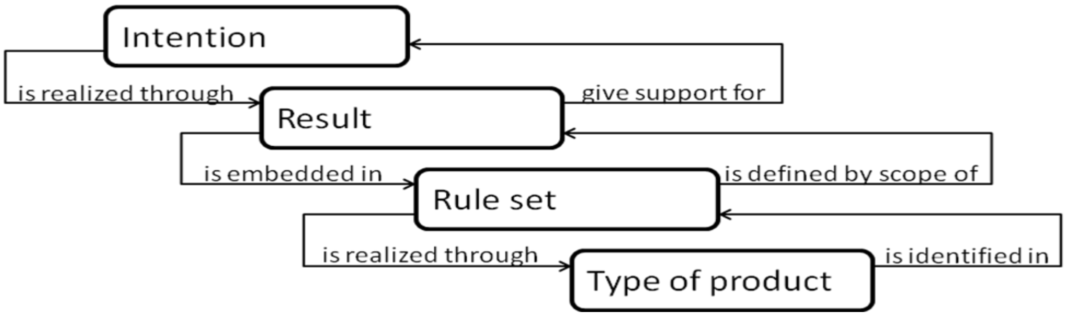

Figure 2: Ontology of model checking concepts

Based on "Intention" as the top level we propose following four concepts of model checking summarized in table 1 below.

Table 1: Concepts of model checking

Intention	**Result**	**Rule set**	**Type of product *)**	**Generic example**
Validating:	pass/fail	is related to used standard / code / regulation etc. or parts above	is related to criteria in rule set	Clause = R or not(A) or not(S) or E
Guiding:	options and advice			set of possible R given A, S, E
Adaptive:	a modified model			one R given A, S, E
Content:	a filtered norm			set of relevant R given A,S,E

*) can also be a materials, functions or other limitable subjects

R = Requirement, A = Applies, S = Select, E = Exception[2]

The proposed ontology for model checking can seen as a contribution to more precise use of model checking. This approach makes model checking as a part of a knowledge system (cf. figure 1.). Mixed systems of e.g. validating and guiding concepts can be developed for practical use, however the criteria to each concept remains unchanged.

9. REFERENCENS

Bell, Bjørkhaug and Hjelseth. (2009) "Standardized Computable Rules", Computable rule project, http://www.standard.no/en/Sectors/Bygg-og-anlegg/Digital-byggeprosess/ISO-BIM-standards/Computable-rule-project/ (accessed: 2010-09-29).

Borrmann, A. and Rank, E. (2008) "Topological Operators in a 3D Spatial Query Language for Building Information Models Computation in Engineering", Technische Universitat Munchen, Germany

CIB-IDDS. (2010) "Introduction to Integrated Design and Delivery Solutions", http://www.cibworld.nl/site/programme/priority_themes/integrated_design_solutions.html (accessed: 2010-09-29).

Gruber, T.R. (1993) "Towards Principles for the Design of Ontologies Used For Knowledge Sharing", Technical report KSL 93-04, Knowledge Systems laboratory, Stanford University. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.89.5775&rep=rep1&type=pdf (accessed: 2010-09-29).

[2] The operators are based on Hjelseth and Nisbet (2010).

Hendler, J. and McGuinness, D. L. (2000) “The darpa agent mark-up language. In IEEE intelligent systems trends and controversies”, November/December 2000. http://www.ksl.stanford.edu/people/dlm/papers/ieee-daml01-abstract.html (accessed: 2010-09-29).

Hjelseth, E. (2009a) “Exchange of relevant information in BIM-objects defined by the Life cycle Information Model (LIM)”, Presented at CIB-IDS, Finland, Espoo, 10th -12th July 2009.

Hjelseth, E. (2009b) “Foundation for development of computable rules”. Presented at CIB-W78 conference in Istanbul 1st – 3rd October 2009.

Hjelseth, E. (2009c) “ Introduction to the International Framework for Dictionaries (IFD)”, The BIM, Building Information Modelling, (p. 38-43), Journal of buildingSMART Korea, www.buildingsmart.or.kr (accessed: 2010-09-29).

Hjelseth, E. (2010) “RFID for improved logistics in the construction industry”, Presentation at Workshop ”RFID and the Internet of Things - Are you ready?” 11. May 2010, Oslo. http://www.rfid-rnet.com/Presentations_11_May_2010/E_Hjelseth_Standard_Norge_Workshop_IoT_Oslo_11_May_2010.pdf (accessed: 2010-09-29).

Hjelseth, E. and Nisbet, N. (2010) “Exploring semantic based model checking”. Presented at the CIB-W078 conference in Cairo 2010.

Ingvaldsen, T. (1994) ” Byggskadeomfanget i Norge. Utbedrings kostnader i norsk bygge-/ eiendomsbransje- og erfaringer fra andre land”, NBI Rapport 163, Oslo.

Ingvaldsen, T. (2001) ” Skader på bygg: Grunnlag for systematisk måling”. NBI Rapport 308, Oslo.

ISO. (2010) “ISO Concept Database” http://cdb.iso.org/cdb/search.action (accessed 2010-09-29).

ISO 10303-202:1996. Industrial automation systems and integration -- Product data representation and exchange -- Part 202: Application protocol: Associative draughting. http://www.iso.org/iso/search.htm?qt=10303-202&sort=rel&type=simple&published=on (accessed: 2010-09-29).

ISO 22263:2008. Organization of information about construction works -- Framework for management of project information. http://www.iso.org/iso/search.htm?qt=22263&sort=rel&type=simple&published=on (accessed: 2010-09-29).

ISO 29481-1:2010 Building information modeling - Information delivery manual - Part 1: Methodology and format. http://www.iso.org/iso/search.htm?qt=29481&sort=rel&type=simple&published=on (accessed 2010-09-29).

Kiviniemi, A. (2005) “ Requirements Management Interface to Building Product Models”. Ph.D. thesis CIFE Technical Report # 161. Center for Integrated Facility Engineering, Stanford University.

McGuinness. D.L. (2002) “Ontologies come of age”. In Dieter Fensel, Jim Hendler, Henry Lieberman, and Wolfgang Wahlster, editors, Spinning the semantic web: bringing the world wide web to its full potential. MIT press, 2002. http://www.ksl.stanford.edu/people/dlm/papers/ontologies-come-of-age-mit-press-(with-citation).htm (accessed 2010-09-29)

Opdahl, P.O. and Olsen, Y.(2009) “Knowledge based design Integrations based on Bluethink Applications”. CIB-IDS conference, Suomi, 10-12 June 2009.

SmartMarket Report (2009) “Building Information Modelling Trends”. SmartMarket Report. McGraw-Hill Companies, ISBN: 978-1-934926-25-3. http://construction.ecnext.com/coms2/analytics (accessed 2010-09-29).

Sowa, J. F. (2006) “The Challenge of Knowledge Soup”. Published at Research Trends in Science, Technology and Mathematics Education, Mumbai. http://www.jfsowa.com/pubs/challenge.pdf (accessed 2010-09-29).

Statsbygg. (2010) “Public Construction and Property”, Norway. http://www.statsbygg.no/Utviklingsprosjekter/NationalMuseum/ (accessed 2010-09-29).

Wiki. (2010) “ Model checking”. Wikipedia, the free encyclopedia, http://en.wikipedia.org/wiki/Model_checking (accessed 2010-09-29).