TOWARDS A TOPOLOGICAL REASONING SERVICE FOR IFC-BASED BUILDING INFORMATION MODELS IN A SEMANTIC WEB CONTEXT

Jakob Beetz[1], **Jos van Leeuwen**[2], **and Bauke de Vries**[3]

ABSTRACT

One of the classic problems identified in the interdisciplinary use of Building Information Models (BIM) is the different representation requirements regarding topology (Eastman 1999). Although this problem has been addressed in several modeling efforts (Augenbroe 1995) the most widely spread BIM to date (IFC) does not bridge one of the essential gaps, namely that between geometry-centered (used by most generic CA(A)D applications) versus topological space-centered models (used by energy performance tools).

Based on a previously developed Description Logic representation of the IFC model notated in OWL (Beetz, de Vries, van Leeuwen 2005), we describe the ongoing development of an online reasoning service to demonstrate the practical use of the Semantic Web tool chain in the AEC domain context. This reasoning service is part of a demonstration scenario in which energy performance estimates based on the ESP-r package are integrated into the architectural design process mediated by a multi agent system.

In this paper we demonstrate how to infer spatial relations of a geometry-centered building model instance exported from standard packages. The approach we propose can be used in various scenarios where translation between different representations and mappings of other information are necessary. While these transformations and mappings are well understood and have been proven feasible in other research and developments (Bazjanac 2004, Treeck 2004), the uniqueness of our approach lies in the computational modularization of some aspects of this process and its integration into emerging technologies of the Semantic Web. Chaining these kind of semantically enhanced reasoning service modules together, future domain application developers might be relieved of implementing some of the cumbersome aspects when integrating their tools into the design and decision making process.

KEY WORDS

building information modeling, semantic web, knowledge engineering, agent systems, interoperability.

---

[1] Phd Student, B, Dept Architecture, Building and Planning, Vertigo Building, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands, j.beetz@bwk.tue.nl

[2] Assistant Professor, Dept Architecture, Building and Planning, Vertigo Building, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands, j.p.v.leeuwen@bwk.tue.nl

[3] Professor, Dept Architecture, Building and Planning, Vertigo Building, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands, b.d.vries@tue.nl

INTRODUCTION AND RELATED WORK

The reasons for the growing need of interoperability for ICT in the building and construction industry are manifold: The growing amount of available software tools, the increasing adoption of digital over hardcopy-based information exchange, the internationalization of the business process, and the growing demand for sustainable information coverage over the whole lifecycle of a building are just a few to be named here. The standardization efforts of product data exchange formats that have a long tradition in research have led to the creation of the Industry Foundation Classes (IFC) in recent years. Many of the market-leading software vendors have stepped (back) into the process of implementing this exchange format in their packages, enabling the exchange of information in heterogeneous project teams. However, many problems remain to make this process efficient. Not only have various aspects related to the general nature of information modeling, storage and processing still to be addressed and solved: The introduction of a storage model that is constantly evolving in size and complexity has also created some new problems. Especially small software vendors that have specialized in niche domains and markets as well as research and other public institutions face severe problems of adopting and maintaining interfaces between their internal computational models and the IFCs. Several efforts within the IFC development community, such as the introduction of the Part 28 XML notations for instance data, partial model view definitions, Property Sets and the advent of mechanisms for external reference of information try to overcome some of these problems. At the same time, some of these features impose the risk of an even further growing diversification of instance models e.g. by sanctioning the use of weakly typed information encoding.

The Semantic Web initiative was prompted by the problems related to the heterogeneous data formats in distributed collaboration setups, (even though XML was embraced as 'the end of interoperability problems' in the beginning), a lacking standard of exchange policies, and the high level of manual human engineering requirements resulting from this. Although many varying views regarding the nature and use of the Semantic Web are in existence ('information annotation', 'classification', 'web of services', 'one giant database', 'information indexing' etc.) a set of goals is common to all of them: To enhance the machine readability and interpretability of distributed and possibly incomplete information and to standardize the way this information is exchanged among pieces of software. To achieve this, several new methods and technologies are being actively developed, existing ones adapted and almost forgotten ones resurrected. This paper discusses the underlying principles, their current implementation status and most importantly their applicability to problems in the Building Information Modeling domain. As a proof of concept, we introduce a small, simplified use case scenario and its implementation. In the concluding part of this paper we give an outlook to future research.

The application of methods and technologies stemming from the artificial intelligence community has a long tradition in the ICT in AEC/FM domain. However, many of the promising applications have not succeeded in industry practice and have on the contrary created a significant amount of disappointment and frustration. In the meantime, the AI community itself has not been able to fulfill some of the high goals that have been promised to achieve in the past and has re-calibrated many of its short-term aims.

METHODOLOGY

THE NOTIONS OF THE TERM 'SEMANTIC WEB'

One of the most elemental and agreed upon widely accepted visions of the Semantic Web initiative is to create a network of interconnected knowledge resources rather than loosely coupled documents that are merely compatible on a syntactical level. The goal is to make these resources discoverable, retrievable, interpretable and processable for pieces of software that act on behalf of a user. In order be able to interweave different knowledge resources describing different domains, members of the W3C have come up with several cascading standards and recommendations that enable their interoperability. While the lower layers of this overall architecture (referred to as the 'semantic web stack'), such as XML, namespaces and XML Schema, have reached the level of maturity and are widely accepted and adopted as a means of data exchange in many industry domains and day-to-day applications, the upper layers are partly still in its infancy and are constantly evolving.

Graphs, Syntax and Structure: RDF and RDF/Schema

At the lowest level of the evolving semantic web stack, the Resource Description Framework (RDF) has been developed and specified by Lassila et al (Lassila 2000, Lassila and Swick 1999). Its main concept is to express information as a 3-tuple (subject, predicate, object) in which each element is (preferably) identified by a Uniform Resource Identifier (URI).forming a directed graph. In principal, every relational database can be decomposed into subject-predicate-object triplets. This characteristic, along with some additional features of RDF, containers and reification (making statements about statements) form a very flexible method to model information distributed among all possible locations that can be identified by a URI (though in practice most resources are identified by URLs, a subset of URIs).

Although there are several possible notation formats of RDF, its most well known flavor is the XML concrete syntax. Among its most popular uses on the World Wide Web are the RDF Site Summary variants to aggregate updates of web pages and the Friends-of-a-Friend (FOAF) vocabulary to create typed links between information about people, thus to construct social networks. As a general purpose description language RDF is heavily used by the Mozilla foundation to describe user interfaces (XUL). A rich set of tools for performant query and persistent storage (Sesame, RDFDB, RDF Gateway, a.o.) of RDF graphs are available as mature and robust open source implementations. Although no definite standard has been officially announced by the W3C yet, some of the query languages like RDQL, SPARQL and SeRQL, have reached a level of quasi-standardization. Many of these languages exhibit some resemblance to the popular database-query language SQL, which facilitates the shift from traditional technologies (or "Webizing existing systems" as Berners-Lee refers to it). An interesting feature of some of these languages and implementations is the blurring border between mere traditional query and logic-based inference capabilities. In the demonstration scenario we give an example how to retrieve a hashtable from an in-memory RDF-based notation of an IFC instance file using SPARQL (Prud'hommeaux and Seaborne 2005).

With the addition of RDF Schema a rich and extendable object-oriented vocabulary to RDF resources has been recommended by the W3C (Brickley and Guha, 2004) as a powerful modeling vocabulary for information, including multiple inheritance, domain and range restrictions. Although these modeling extensions already constitute a semantical enrichment to some degree, the advanced features of modeling information according to concepts from the artificial intelligence community are even an additional layer higher up the stack. A good introduction that discusses the differences of pure RDFS and higher-level descriptions can be found in (Lassila and McGuiness 2001)

Knowledge Representation, OWL and rule description languages

Knowledge representation systems to capture structured information about a domain of expertise have been around since the early days of information technology. They have a long tradition in Epistemology, a discipline in philosophy, and have been applied with varying success in many different domains. The two most important families of knowledge representation systems that are relevant in the current semantic web developments are the frame-based systems family (Minsky 1975) including semantic networks (Woods 1975) and the Description Logic (DL) (Baader et al 2002). More recent developments, that apply these methods such as the OIL and DAML have introduced a blended strategy of these methods, lending the frame-slot-filler concepts from one and the rigid axiomatic logic including entailment and theorem proving from the other. This leads to the possibility to encode ontologies, "formal specification of a shared conceptualization" (Gruber 1993)

OIL and DAML have been merged into the joint OWL standardization effort, which uses the concepts introduced in RDF(S) and extends them with entailment operators, multiple range and domain restrictions and cardinality constraints. In order to make use of the reasoning engines, which enable consistency checking and inference of implicit knowledge a hierarchical set of different flavors of OWL has been proposed, ranging from generally provable, but expressively poor OWL Lite to undecidable OWL Full.

To enable reasoning and logic programming systems to make more deeply 'hidden' implicit knowledge explicit, various efforts to standardize encodings of rule languages are currently struggling for acceptance. The most important ones among them are SWRL and RuleML that allow the encodings of rules and their variable bindings in way that integrates into the rest of the semantic web stack. The most important aspect of these efforts are their crucial position in the whole concept of an enhanced interoperability: Only if a successful separation of knowledge and application code is achieved, will the Semantic Web succeed in this regard (which, again, is only one of the possible, and maybe not its most important one).

Agent Systems and Semantic Web Services

According to Weiss, an agent is "a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives." While historically preceding the semantic web, the notion of an agent has been around in the artificial community domain for a while. However, only the aspect of its autonomy is entailed by the broad range of scientific definitions. The term Semantic Web Services is agreed upon as the addition of semantic information to the well known technology of web services by not only referencing an ontology that describes their underlying knowledge representation and thus enable clients to share it with them, but by also describing *how* it should be accessed with regard to sequence and conditions. The unique opportunity in bringing the two together lies in the ability to harmonize the internal representations of an agent accessing a service with its counterpart. If no such direct connection can be achieved, a special ontology mapping service can be consulted by the agent to translate between the two.

APPLICATION

In the preceding part, we have illustrated the various methods and technologies related to the semantic web. In this part of the paper, we will show an example how some of these technologies can be used to assist in solving interoperability problems in the building and construction industry domain. It is important to understand that the example we have chosen to illustrate this is very simplified and that the solution we propose is only one of many possible.

DEMONSTRATION SCENARIO

In order to demonstrate the feasibility of the semantic web tool chain to common problems in the AEC/FM interoperability the following scenario is considered: An architect has created a small building design an would like to estimate the consequences of the spatial layout of different rooms with regard to energy consumption. In order to estimate the energy performance of a building design using the ESP-r package a topological representation of spaces forming different zones are required. Each room in a design is bounded by a connected set of faces. Each of the bounding faces has a number of material properties such as conductivity, density and thickness assigned to them. These form the necessary information to calculate properties like heat transmission and others[4].

In a traditional workflow, the designer would have to perform the following tasks:

1. Draw a wireframe representation of each wall
2. export the wireframe as DXF
3. import the DXF model into ESP-r
4. assign material properties to the wall
5. create zones
6. assign properties to the zones
7. run the simulation
8. analyze the simulation results (with help of a domain expert)
9. adapt the design accordingly
10. GOTO 1

---

[4] Additional values that are required for the simulation are left out in this scenario and will be filled in with default values.

Although some of the issues involved in this workflow could be addressed or even solved by adding an IFC I/O integration layer to the simulation software (which is desirable but lies outside the scope of our research interests), some problems remain unsolved. Depending on the model-generating software (in our case an architectural CAAD package) a range of different building models with varying richness and usefulness of information are possible. Building elements, such as walls, doors etc. can be represented by different geometry types (2D-lines, polygonal representations, complex nested CSGs etc.), IfcSpaces and IfcZones can either be defined or omitted and in none of the currently available IFC-generating packages that we evaluated higher-level relations such as IfcRelSpaceBoundaries are defined.

Since this kind of additional information is not only required in our scenario, where we need to be able to relate walls and their associated material properties to boundaries of spaces, it seems desirable to offer a generic mechanism that infers the additional relations from the existing data.

In our approach, we use an expert system enhanced with additional vector math capabilities that draws conclusions by applying forward-chained reasoning methods. In order to demonstrate this inference of new facts whilst keeping the example describable, we will assume the following boundary conditions for the input geometry:

1. All spaces are non-overlapping
2. All walls are connected to each other at endpoints only, not in the middle.

SYSTEM ARCHITECTURE

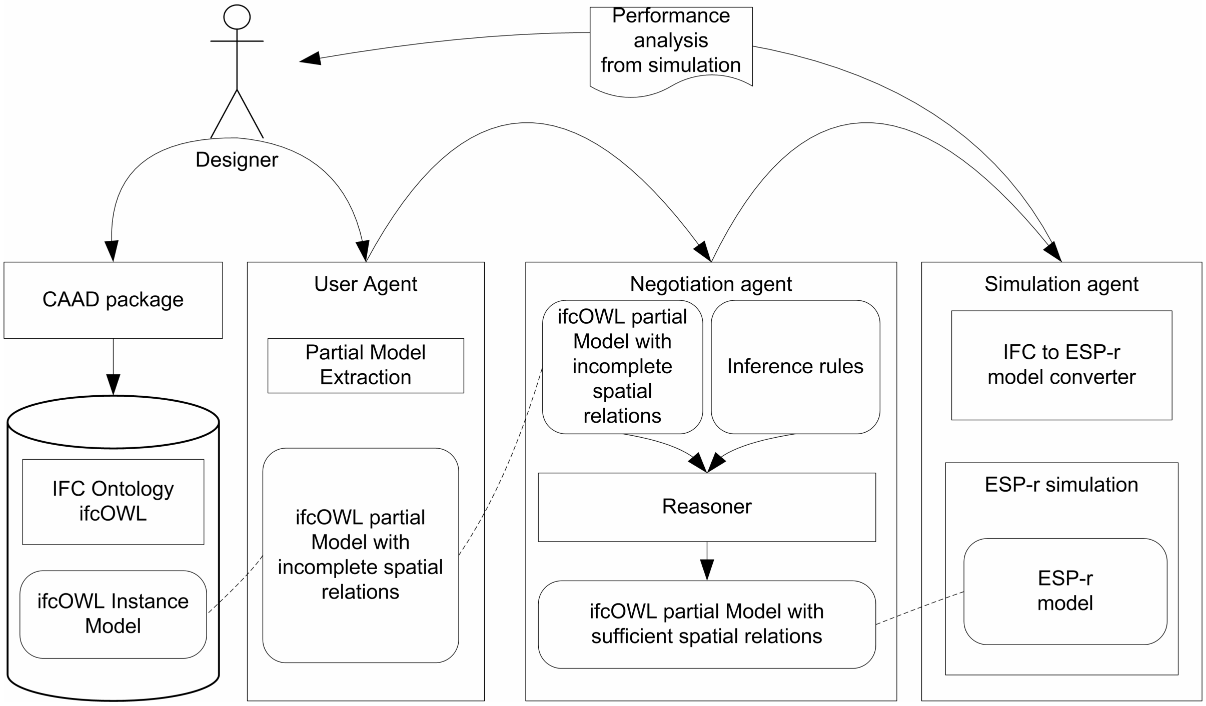

Figure 1: Schematic diagram of the overall scenario

Model checker

Before a (partial) model is extracted and transferred to the topological inference service, some simple tests are being conducted to assert the validity of the first boundary condition mentioned above. To assert, that all walls are only connected at their ends we query the model for the “ConnectedFrom“ and “ConncetedTo” slot fillers of every wall for their respective values and ensure that only IfcConnectionTypeEnums with “ATSTART” and “ATEND” instance values exist. This is done in a similarly efficient and fast fashion as demonstrated by the query in the following section.

Query and partial model extraction

Since only a limited subset of the information stored in the building model is relevant for the calculations, the reasoning service first extracts a partial model of the overall building design. Using an existing implementation of the SPARQL query language for RDF we retrieve the relevant instances of the model by performing queries and constructing partial intermediate graphs of the ifcOWL ontology. In these partial graphs, only the relevant concepts, their slots and roles are replicated into an intermediate subontology for which relevant facts exist. In order to be able to later merge the newly inferred facts into the original model, the URIs of the instance resources are being reflected into newly created subontology[5].

Table 1: SPARQL query statement that performs the task “retrieve all coordinates of only those IfcPolylines that are IfcRepresentationItems for IfcWallStandardCases and map all variables into a hashtable.”

```
SELECT *
WHERE {     ?wall a ifc:IfcWallStandardCase .
            ?wall ifc:Representation ?productShape .
            ?productShape ifc:Representations ?representation.
            ?representation ifc:Items ?polyline.
            ?polyline ifc:Points ?point .
            ?point ifc:Coordinates ?coordinate
}
```

Although we have no indicative performance data in comparison to existing conventional tools as of yet, early performance and scalability tests have been developed by Guo et al (2004) and some implementations show very promising results for large-scale scenarios However, these early results of prototype systems have to be treated with much care.

Topological inference

The topological inference itself is performed using a rule-based programming system. In a first step, the facts that have been extracted from the model in the first part are being

---

[5] Persistence related problems that might occur when the original model is changed while the inference process is going on are not being considered here.

converted into the appropriate representation of the expert system. Currently, a number of tools and reasoning engines are under development or ready for use, which support the automatic conversions between OWL augmented with rules notated in the Semantic Web Rule Language SWRL and internal representation of expert systems. Besides Racer(Pro), Hoolet and Bossam, the SweetRules suite of tools is an excellent candidate for the task at hand. Using SweetRules, a number of different expert systems can be interfaced, namely the Prolog implementation XSB and the production rule programming systems Jess and CLIPS. As an additional level of flexibility, the SweetRules development team aims at a standardized way of procedural attachments to those system, referred to as "Situated Corteous Logic Programs" SCLP (Grosof, 2001).With this extension mechanism generic external functions (e.g. written in Java) can be used for both *effector* statements (if a rule in the systems fires, an external procedure is invoked) and more importantly in our case *sensor* statements (a antecedent condition is evaluated by a external function). We will point out the use of these extension mechanisms in the discussion section of this paper. For the actual topological inference, we construct a number of cascading rules that are fired when their conditions hold using an implementation of the RETE algorithms by Forgy (Forgy 1982).

After collecting the results, we introduce them as new relations back into the model, effectively facilitating the further use in larger context of the overall scenario.

DISCUSSION

The demonstration scenario we have chosen to illustrate the application of some of the methods discussed has been kept simple on purpose. Although there might be other tasks that are more suitable to demonstrate inference using expert systems in a building model, e.g., escape route planning, building law conformance checking, constraint modeling, etc., we have chosen this well understood problem to illustrate the differences in the approach compared to conventional purely sequential procedural methods.

It might be argued that the proposed use of procedural extensions mechanisms to logic programming contradicts the effort to keep code separated from logic. However recent developments show, that at least a mathematical subset of these kinds of extensions will be transferred to the common domain and adopted by the implementers of reasoning engines. We see the main advantage of using such procedural extensions in scenarios, where they might constitute modular sensor augmentation, e.g., to model servers, subscribing to a certain subset of the general model on behalf of a domain specialist and notifying him on certain events by triggering the users respective agent.

Overall we think that one of the great opportunities in the approach demonstrated lie in the perspective proliferation of small specialized tools that can be chained together much like in the UNIX world.

CONCLUSION AND FUTURE WORKS

In this paper we have demonstrated a use-case for the application of methods and technologies related to the emerging Semantic Web in the AEC/FM domain. We have given an overview of the underlying technologies and their practical implications in distributed collaboration scenarios. We have shown that some aspects of common problems in interoperability of building information modeling can be successfully addressed using existing tools and tailoring them to domain specific needs.

In general, due to the novelty of the notions of the Semantic Web, many of the underlying technologies are not very well studied as of yet. Future research is necessary to investigate the implications of these methods, especially with regard to performance, scalability and trustworthiness of these systems.

For the building information modeling community we consider a number of aspects worth further investigation, among them partial model distribution over networks, tool chain orchestration of specialized services and maintainability enhancements for the Industry Foundation Classes model.

Our future work in this area will be focused on the field of communication protocol development for multi agent systems addressing the specific needs of the heterogeneous building and construction industry.

REFERENCES

Augenbroe, Godfried L.M. (1995), “COMBINE, 2 Final Report” EU DG XII JOULE Report, Brussels

Baader, (2002) “The Description Logic Handbook: Theory, Implementation, Applications” Cambridge University Press, Cambridge, UK, 2003.

Bazjanac, V. and Maile, T. (2004). “IFC HVAC Interface to EnergyPlus - a Case of Expanded Interoperability for Energy”, SimBuild 2004, IBPSA-USA National Conference Boulder

Brickley,D and Guha, RV (2004) “RDF Vocabulary Description Language 1.0: RDF Schema”, W3C Recommendation, 2004

Eastman, Charles M. (1999) “Building Product Models” CRC Press, Boca Raton, Florida

Forgy, Charles (2004) "Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem", Artificial Intelligence, 19, pp 17-37, 1982

Guo, Yuanbo, Pan, Zhengxiang and Hefflin, Jeff (2004) "LUBM: A benchmark for OWL knowledge base systems” Web Semantics: Science, Services and Agents on the World Wide Web

Volume 3, Issues 2-3 , October 2005, Pages 158-182

Grosof, Benjamin (2001)“Representing E-Business Rules for the Semantic Web: Situated Courteous Logic Programs in RuleML”, Proc. Workshop on Information Technologies and Systems (WITS '01), New Orleans, December, 2001.

Gruber, T. 1993. “A translation approach to portable ontology specifications”. Knowledge Acquistion 5(2): 199-220

Lima, C., Fies, B., El-Diraby, T. A., & Lefrancois, G. (2003). The challenge of using a domain Ontology in KM solutions: the e-COGNOS experience". In Proceedings of 10th ISPE International Conference on Concurrent Engineering: Research and Applications 2003.

Lassila, Ora: "The Resource Description Framework", IEEE Intelligent Systems 15(6): 67-69 (November/December 2000)

Lassila, Ora and Swick, Ralph R, “Resource Description Framework, (RDF) Model and Syntax Specification”, http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/

Lassila, Ora and McGuiness, Deborah (2001) “The Role of Frame-Based Representation on the Semantic Web”, Linköping Electronic Articles in Computer and Information Science, Issue: Vol. 6 (2001), No. 005

Minsky, Marvin (1975) "A Framework for Representing Knowledge", in Patrick Henry Winston (ed.), The Psychology of Computer Vision, McGraw-Hill, New York, 1975.

Prud'hommeaux, Eric and Seaborne, Andy (2005) “SPARQL Query Language for RDF” W3C Working Draft 23 November 2005, http://www.w3.org/TR/rdf-sparql-query/

Treeck, Christoph van and Rank, Ernst (2004) „Analysis of Building Structure and Topology Based on Graph Theory”, Proceedings of ICCCBE, Xth Int. Conf. on Comp. in Civil and Buildg. Eng., Weimar.

Weiss, G. (ed.) 2000. “Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence” MIT Press Cambridge, MA, USA

Woods, William A (1975) "What's in a Link: Foundations for Semantic Networks", in D.G.Bobrow & A.M.Collins (eds.), Representation and Understanding: Studies in Cognitive Science, 35-82, Academic Press, New York, 1975.