Proc. of the 11th Intl. CAAD Futures Conf. 2005 (CAAD Futures 2005)
eds. B. Martens und A. Brown, Seite 207-216, Springer, Vienna, Austria, 2005

Semantic Roomobjects for Conceptual Design Support

A Knowledge-based Approach

KRAFT Bodo[1], and SCHNEIDER Gerd[2]
[1] *Aachen, University of Technology – Department of Computer Science III*
[2] *Nussbaum GmbH*

Keywords: Conceptual Design, Semantic Modelling, Ontology

Abstract: The conceptual design at the beginning of the building construction process is essential for the success of a building project. Even if some CAD tools allow elaborating conceptual sketches, they rather focus on the shape of the building elements and not on their functionality. We introduce semantic roomobjects and roomlinks, by way of example to the CAD tool ArchiCAD. These extensions provide a basis for specifying the organisation and functionality of a building and free architects being forced to directly produce detailed constructive sketches. Furthermore, we introduce consistency analyses of the conceptual sketch, based on an ontology containing conceptual relevant knowledge, specific to one class of buildings.

1 INTRODUCTION

The building construction process is subdivided into several phases. At the beginning, architects analyse the requirements of the new project and develop a first sketch of the future building. During this early phase, called conceptual design, the functionality and organization of the whole building are more important than exact dimensions and material definitions (Coyne et al. 1990). Usually, architects still use pencil drawings for the conceptual design. After finishing the conceptual design, the sketch has to be manually transferred into a CAD tool. Although the information specified in the conceptual design is essential for all following design phases, current CAD tools are restricted to store only the constructive design information. The semantics, implicitly stored in the pencil drawing, gets lost.

In our project, the research field is to elaborate a new conceptual design support for industrial CAD tools. To give architects more adequate tools for the early design phase, roomobjects and roomlinks with predefined semantics are introduced and implemented by way of example to the CAD tool ArchiCAD (GRAPHISOFT 2005). These new construction elements represent the conceptual relevant entities for conceptual design. Furthermore, new functionality is developed, to support architects working with these entities. Integrated with ArchiCAD, the conceptual sketch is checked against conceptual relevant knowledge, dynamically defined by a knowledge engineer using the ontology editor Protégé. The knowledge can be exported and used by the new developed ArchiCAD ConstraintChecker as the basis for the consistency analyses. Restriction violations are visualized inside the architect's sketch without forcing him to modify his sketch.

In this paper we describe how the conceptual design process is supported using semantic roomobjects and roomlinks. Illustrated by the example of the CarSatellite project (Nussbaum 2005), we demonstrate the gain of semantic-oriented design, abstraction mechanisms, and the explicit definition of functional relationships. We further introduce the knowledge specification process using Protégé and the consistency analyses inside ArchiCAD. We finally close with a conclusion.

In literature there are several approaches to support architects in design. Christopher Alexander describes a way to define architectural design patterns (Alexander 1995). Although design patterns are extensively used in computer sciences, in architectural design this approach has never been formalized, implemented and used. The SEED system (Flemming 1994) provides a support for the early phase in architectural building design. In contrast to our approach, the SEED system mainly focuses on the generation of sketches. The importance of knowledge processing for architectural design is comprehensively discussed in (Coyne et al. 1990). In (Schmitt 1993), different new paradigms for a conceptual design support are proposed. Among other things, he introduces the top-down decomposition and modularisation of sketches and the use of object-orientation for architectural design. Even if the work is neither implemented nor integrated into a CAD tool, the ideas are fundamental for our research. The semantic web approach (Berners-Lee, Hendler, and Lassila 2001) tries to improve the quality of information in the World Wide Web. This approach is based on RDF (Powers 2003), a language developed for modelling knowledge. Even if a lot of ontologies have been developed, none of them is applicable for the conceptual design phase. (Bazjanac 1999) demonstrates the benefits of interoperability between different CAD tools. Starting from a conceptual sketch, developed in Nemetschek's Alberti, the sketch is exchanged using the IFC format between CAD tools for constructive and detailed design. The CAD tool Alberti allows defining a bubble diagram as conceptual sketch which is then transferred into a CAD drawing. We follow a similar idea but additionally provide consistency analyses of the conceptual sketch.

2 MOTIVATION FOR SEMANTIC MODELLING

The design support provided by traditional CAD tools is restricted to the simple construction of buildings. The main drawing elements are walls, doors, windows, columns and slabs. Using these construction elements, architects can develop detailed and precise sketches. In Figure 1, a sketch of a CarSatellite garage and the corresponding photo-realistic rendering is depicted. The sketch has already been detailed elaborated; it shows the result of the *constructive design phase*.

The Nussbaum GmbH develops the CarSatellite concept for modular built-up garages. The customer can combine different functional entities to configure a garage

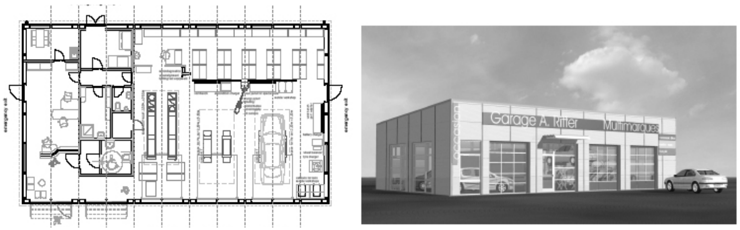

Figure 1 CarSatellite Sketch and Rendering

specific to his needs. He can e. g. specify the number of car diagnostic and car repair places; he can define the size of the storage area, the number of offices and the size of the customer area. The complete garage, composed of different functional entities is then prefabricated including all technical equipment and speedily put up on site.

The executive architect at Nussbaum GmbH uses ArchiCAD for developing the sketches of the CarSatellite projects. Even if ArchiCAD is an adequate tool for the constructive design, as one can see in Figure 1, ArchiCAD does not provide any support while developing the concept of the CarSatellite garages. The development of the conceptual design is a quite challenging task, as design aspects and multiple restrictions from different domains have to be regarded. The architect tries out different combinations of the functional entities to elaborate an optimal sketch; he therefore considers size restrictions, e. g. for a car diagnostic place or workbenches, and needed adjacency relationships between functional entities. The traditional wall structure does not adequately map these functional entities and their relationships. Instead of explicitly expressing the concept of the building, the wall structure just implicitly includes the semantics of the building in the sketch. Without any annotated text, a constructive sketch, like in Figure 1, is not clearly readable.

3 INTEGRATED CONCEPTUAL DESIGN SUPPORT

Looking at Figure 1 again, the garage is composed of a customer area on the left side, including customer toilets, an office room, a dressing room with showers and toilets, and the shop floor area. The shop floor area itself is composed of one car diagnostic and two car repair areas, several workbenches for the wheel balancer, the battery charger, motor diagnostics and mobile workbenches. The storage area is situated in the rear part of the building.

For each of these functional entities, one semantic roomobject is defined; we distinguish between complex and atomic roomobjects. A complex roomobject can be refined into further complex or atomic roomobjets. They allow starting the conceptual sketch in an abstract view which can be stepwise refined. Atomic roomobjects describe simple functional entities that cannot be further detailed; they usually represent a room or a part of a room, as the smallest functional entity occurring in the sketch. In Figure 2, a complex roomobject is depicted as a simple rectangle, an atomic roomobject as a rectangle with rounded edges. The reflexive aggregation

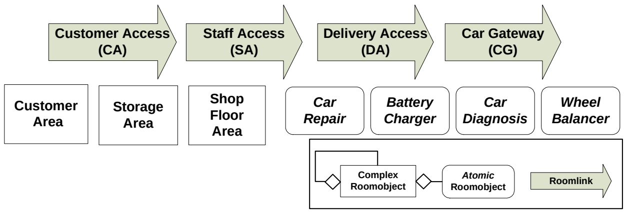

Figure 2 Roomobjects and Roomlinks, specific for Garages

relation between complex roomobjects and the simple aggregation relation between complex and atomic roomobjects is also depicted in Figure 2. To enable architects describing relationships between two roomobjects, we introduce semantic roomlinks. Using roomlinks, the architect can specify functional connections (e. g. access, view, vicinity) between roomobjects that provide a semantic representation containing more information than simple constructive drawing elements (e. g. window, door). The specification of roomlinks helps the architect to explicitly identify the needed relationships between functional entities. As a result, one can clearly recognize the organization of the garage, the semantics of the sketches is explicitly shown. Once defined, the set of roomobjects and roomlinks can be used for each building project of the corresponding class of buildings. In Figure 2 a subset of the roomobjects and roomlinks for the CarSatellite project is depicted. One can identify four different roomlinks to distinguish between the different access relations inside a garage. For a more complete discussion of the dynamic knowledge model definition see (Kraft and Wilhelms 2004).

In Figure 3 three conceptual sketches of the previously introduced CarSatellite garage are depicted, as example of a conceptual design process. The level of abstraction of the conceptual sketch starts with the simple description of the main functional area and its dimension, and ends with a completely elaborated conceptual sketch that can be transferred into a traditional wall structure.

In the first step of the conceptual design process (① in Figure 3), the architect starts with the definition of the building-project's size and shape using the most abstract complex roomobject *CarSatellite-Garage*. The size usually depends on the building site, calculated costs and the wishes of the investor. To formally define how the garage can be accessed by different groups of people, the architect introduces semantic roomlinks to the sketch. In the example in Figure 3, the size of the CarSatellite garage is set to 264 sqm, a customer access (CA) and car gateway (CG) are planned to be situated in the front part, a staff access (SA) and delivery access (DA) in the rear part of the building.

In the next step (② in Figure 3) the architect continues identifying the most important functional areas inside the CarSatellite garage. Here, the architect decides to subdivide the garage into a staff area, a storage area, a costumer area and the garage area. Again, the dimension and size of these areas are continuously calculated and displayed inside the roomobjects to give the architect helpful information. Furthermore, the architect now refines the relationships between the semantic roomobjects. In the example sketch, a customer should only be allowed to access the customer area, while the staff has full access anywhere inside the garage. To formalize these requirements, the architect specifies only the staff access (SA) between the main areas of the garage and restricts in this way the traffic flow inside the building.

In the last step (③ in Figure 3), the architect further refines the previously defined sketch, up to the level of atomic roomobjects. The garage area is now composed of one car diagnostic place and two car repair places, a wheel balancer, a battery charger, and a generic workbench. Again, the access is restricted to the person group *staff* to avoid customers accessing these areas.

For an integrated conceptual design support, we extend the ArchiCAD product model with new semantic roomobjects and roomlinks, based on the GRAPHISOFT GDL technology and an ArchiCAD programming interface. Each roomobject represents one functional entity inside the conceptual sketch. The roomobjects can easily be imported to ArchiCAD and used like any other drawing object, so that architects are familiar with the usage.

The architect can continuously choose between a detailed and abstract representation of the conceptual sketch. While e. g. elaborating the detailed conceptual design of the staff area using atomic roomobjects, the other areas in the sketch can be displayed in their abstract representation, hiding currently less important details. Using the ArchiCAD rendering features a 3D representation of the conceptual sketch can be generated to help the architect estimating the proportion of volumes. The 3D representation depicts the sketch in the chosen level of abstraction and is available during the whole conceptual design process. The top-down decomposition of the

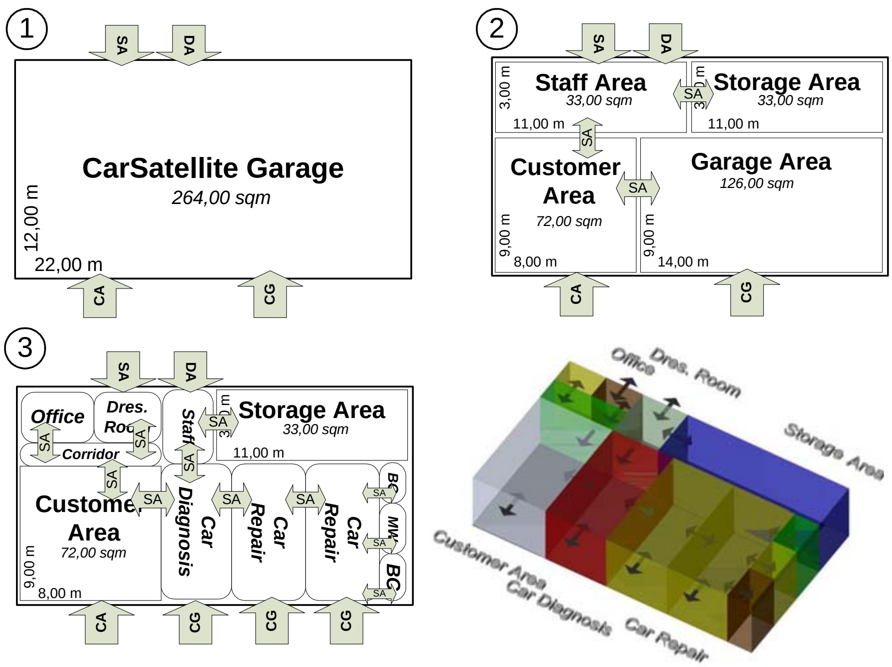

Figure 3 Conceptual Design Sketch with Top-Down Decomposition

sketch helps the architect to reduce the complexity of the sketch, to hide temporarily less important design details and to concentrate on the relevant design tasks. The described functionality is fully integrated in ArchiCAD so that the settling-in period to learn using these tools is minimal. Roomobjects and roomlinks can easily be choosen from a new developed user interface. All functionality to define roomlinks, and to refine or to coarse the level of details can be easily executed. Different editing modes allow a read-only or read-write decomposition of complex areas; the focus is automatically adopted to provide an optimised view on the conceptual design (Schmitt 1993).

4 KNOWLEDGE REPRESENTATION

The complete scenario of the project also consists of a knowledge specification part. In this part, conceptual relevant knowledge, specific to one class of buildings, is being formalized. For knowledge specification and consistency analyses we follow two different approaches. In the graph-based approach, we try to identify the necessary expressive power and the needed structuring elements to develop a visual language for knowledge specification. The consistency between the graph-based knowledge and the conceptual design is calculated using complex analyses based on graph transformations (Kraft and Nagl 2004; Kraft and Wilhelms 2004). In the second approach, we use formal ontologies to classify and define domain specific knowledge. The consistency analyses here are manually implemented in C-code and directly integrated in ArchiCAD. In this paper we concentrate on the second approach.

Using the ontology editor Protégé we initially start with developing an ontology for mapping rules. The ontology is subdivided into a domain unspecific part, which defines the syntax and structure of the rules, and a domain specific part, here specific to CarSatellite garages. Protégé allows a visual definition of the ontology (Figure 4) and the base knowledge (Figure 5).

The domain unspecific part of the ontology consists of four main classes: semantic object, relation, attribute and rule. The class rule is further specialized into attribute, cardinality and relation rules. While the domain unspecific part of the ontology is used as a basis for knowledge specification about any class of buildings, the domain specific part describes the relevant functional entities of one particular class of buildings. The effort, defining knowledge only pays off, if it is reused for several projects of the corresponding class. Therefore, the defined knowledge is on the type level.

Based on the type definition in the ontology, the base knowledge is formalized. Using attribute rules, properties for one roomobject can be demanded or forbidden. A relation rule allows defining demanded or forbidden interrelations between two roomobjects. Attribute rules allow e. g. prescribing a minimal and maximal size for a room-type or define needed equipment to be installed. Relation rules can e. g. demand two rooms to be neighboured or accessible from each other.

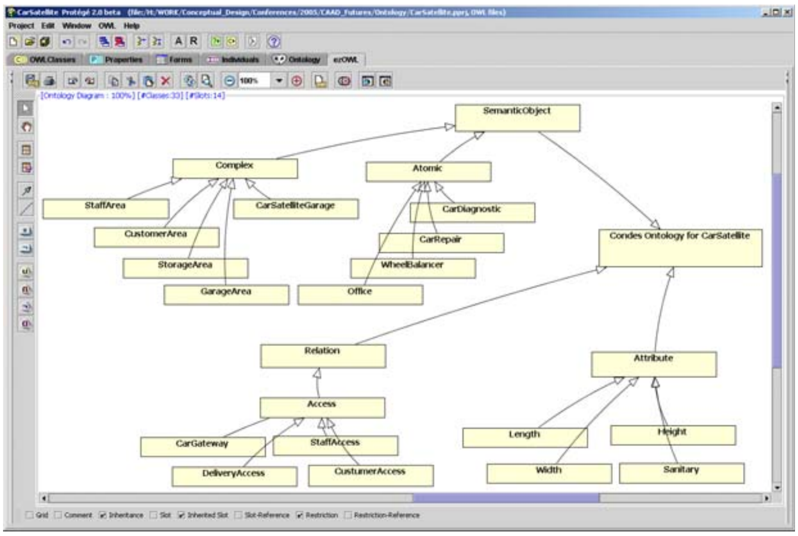

Figure 4 Visual Ontology Specification with Protégé

In the CarSatellite example, we define the relevant functional entities corresponding to the semantics of the previously defined roomobjects and room relations to simplify the consistency analyses, described in section 5. The domain knowledge is currently being developed in cooperation with the Nussbaum GmbH. The merit of focusing on this class of buildings is the well identified set of rules for an extensively elaborated design concept. By restricting ourselves to a highly specialized building project, the development of a closed knowledge base is facilitated.

In Figure 4 a screenshot of the visual ontology definition with Protégé and the ezOWL add-on is displayed. Derived from the class semantic object, the needed complex and atomic functional entities, as well as a set of relations and attributes are created. The classes to represent relation, attribute, and cardinality rules are defined in the same way; the syntax of the rules is specified using associations between the classes. An attribute rule e. g. is composed of three classes: rule, semantic object, and attribute. Using the class hierarchy depicted in Figure 4, all available combinations of semantic objects and attributes can be used for defining rules.

The knowledge definition for CarSatellite comprises e. g. size restrictions for each functional entity. The dimensions of car repair places must be between 9m and 10m depth and 4m width. The depth of the whole CarSatellite garage must be between 10m and 24m depending on the configuration of the garage. It can be composed of just one car repair place, a car repair place and a storage area, two car repair places with or without a storage area in the middle. Furthermore, the knowledge specification prescribes demanded or forbidden relations. The customer lounge e. g. should be situated neighboured to the car diagnostic place. A possibility to view the car should be provided, so that the foreman can show the customer the work to do. To avoid noise pollution, these functional entities should be separated by a wall with a window. The costumer should not have access to the garage area. Finally, the

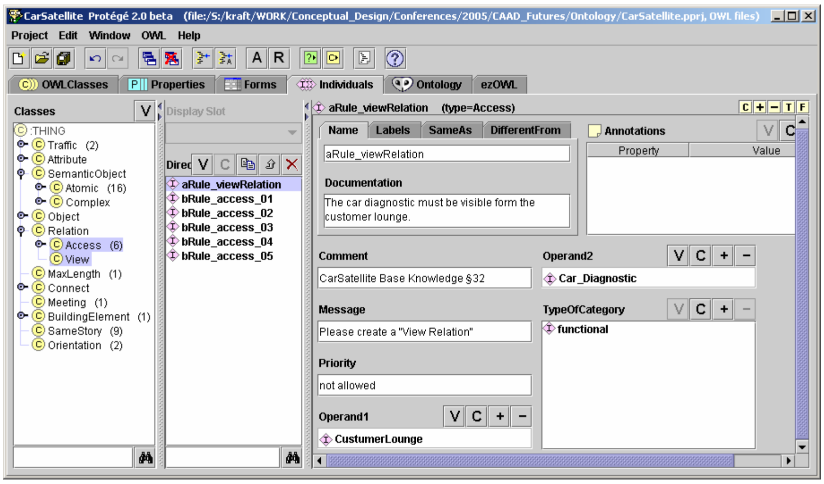

Figure 5 Defining the conceptual knowledge, based on the Ontology

knowledge specification comprises cardinality rules to define the number of allowed occurrences of a room type. It is e. g. a legal restriction to plan a dressing and pause room. Depending on the number of employees separated toilets for women and men have to be installed.

In Figure 5 a screenshot depicts the knowledge specification using Protégé. Based on the previously defined ontology, the so called base knowledge is inserted. Looking at the example, the *view relation* between the customer lounge and the car diagnostic place is depicted. The knowledge specification in Protégé comprises a set of rules. Each rule is composed of a unique identifier, the corresponding semantic objects, attributes or rather relations. Furthermore, two comment fields allow introducing a non formal description and a source of the rule. Finally, an error message containing a hint how to fix the error is stored for each rule. The so defined knowledge is exported using Protégé into an OWL formatted file. As OWL is an adequate data model for ontology storage and exchange, the CarSatellite ontology as well as the corresponding base knowledge can be completely stored and processed as a basis for the consistency analyses in ArchiCAD.

5 CONSISTENCY ANALYSES

To combine the conceptual design using roomobjects and roomlinks with the formally specified knowledge in Protégé, we extend ArchiCAD with new functionality to import and interpret an OWL file to check it against the sketch. We call this extension: ConstraintChecker. The ConstraintChecker allows choosing an OWL file at runtime, depending on the currently used class of rooms (Figure 6).

The ConstraintChecker starts by analysing the attribute rules of all roomobjects, with reference to the length, depth, height and size. The analyses of further attributes, like demanded equipment (sanitary installation, compressed air) is part of

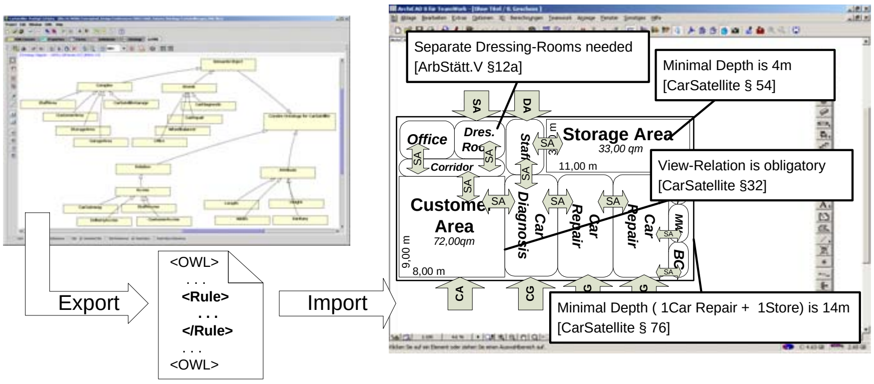

Figure 6 Integrated Consistency Analyses in ArchiCAD

the future work. In case of a restriction violation, the ConstraintChecker displays an error message assigned to the affected roomobject. Analogously, the ConstraintChecker continues analysing all relation and cardinality rules. Obligatory relationships (e. g. staff access anywhere inside the garage) have to be sketched using appropriate roomlinks, forbidden relations must not be sketched. Again, the ConstraintChecker notifies the architect about restriction violations.

Looking at the example sketch in Figure 6, the CarSatellite ontology in Protégé and the conceptual sketch in ArchiCAD are depicted. After importing and interpreting the knowledge specification, the ConstraintChecker identified four restriction violations. In the sketch, e. g. the depth of the storage area remains under the minimal allowed size and the view relation between the customer area and car diagnostic place has not been defined.

The consistency analyses integrated in ArchiCAD works interactively, so that the architect can run the ConstraintChecker anytime during the design process. Thus, the ConstraintChecker is not only used for a final consistency analyses, but also to get continuously more information about the used roomobjects and parts of the sketch.

6 CONCLUSION

Although the conceptual design phase is the basis for the whole design process, none of the traditional CAD tools provide an adequate tool support. We tried to resolve the difficulty supporting the creative elaborating of conceptual sketches by introducing hierarchically processable semantic roomobjects. The abstraction from constructive details to a semantic representation and the possibility to hide temporarily less important cut-outs is conducive to the creativity of the architect. Furthermore, the provided knowledge support aims to help architects avoiding conceptual design errors and to release them keeping plenty of rules and restrictions in mind.

The concepts introduced in this paper are universally applicable for conceptual architectural design. The main idea, to identify important functional entities, relevant for a specific class of buildings, and relations between them, can be introduced to any CAD tool. Our basic implementation in ArchiCAD demonstrates the feasibility of such an approach. The formal knowledge definition, based on a domain specific ontology, is reusable as well. Nevertheless, supporting creativity keeps an ambiguous goal and the success mainly depends on its acceptance of the architects.

7 REFERENCES

Coyne, R. D., M. A. Rosenman, A. D. Radford, M. Balachandran, and J. S. Gero. 1990. *Knowledge Based Design Systems*. Boston: Addison-Wesley.

GRAPHISOFT. 2005. *GRAPHISOFT Homepage*. Internet. Available from http://www.graphisoft.com; accessed 5 January 2005.

Stanford Medical Informatics. 2005. *The Protégé Ontology Editor and Knowledge Acquisition System*. Internet. Available from http://protege.stanford.edu; accessed 30 January 2005.

Nussbaum. 2005. *Carsatellite, die modulare mobile KFZ Werkstatt von Nussbaum.* Internet. Available from http://www.car-satellite.de; accessed 12 January 2005.

Alexander, C. 1977. *A Pattern Language: Towns, Buildings, Construction*. Oxford: Oxford University Press.

Flemming, U. 1994. *Case-Based Design in the SEED System*. In *Knowledge Based Computer Aided Architectural Design*, 69-91. New York: Elsevier.

Schmitt, G. 1993. *Architectura et Machina - Architectural Design und virtuelle Architektur*. Wiesbaden: Vieweg.

Berners-Lee, T., J. Hendler, and O. Lassila. 2001. *The Semantic Web*. In *Scientific American*. Internet. Available from http://www.sciam.com/article.cfm?ArticleID=00048144-10D2-1C70-84A9809EC588EF21; accessed 5 January 2005.

Powers, S. 2003. *Practical RDF*. Cambridge: O'Reilly.

Bazjanac, V. 1999. *Industry Foundation Classes and Interoperable Commercial Software in Support of Design of Energy-Efficient Buildings.* In *Proc. of Building Simulation '99*, Volume 2: 661-667. Boston: Addison-Wesley.

Kraft, B. and N. Wilhelms. 2004. *Interactive Distributed Knowledge Support for Conceptual Building Design*. In *Proc. of the 10th Intl. Conf. on Computing in Civil and Building Engineering*, eds. K. Beucke and B. Firmenich, 1-15 (CD-ROM). Weimar: Bauhaus-Universität Weimar.

Kraft, B. and M. Nagl. 2004. *Parameterized Specification of Conceptual Design Tools in Civil Engineering.* In *Proc. of the Intl. Workshop on Applications of Graph Transformation with Industrial Relevance*, eds. M. Nagl and J. Pfalz, 95-105. Berlin: Springer