Computing Ontologies to Support AEC Collaborative Design

Towards a Building Organism delicate concept

Antonio Fioravanti¹, Gianluigi Loffreda², Armando Trento³
¹, ², ³Sapienza University of Rome - Italy
¹, ², ³http://dau.uniroma1.it
¹antonio.fioravanti@uniroma1.it, ²gianluigi.loffreda@uniroma1.it,
³armando.trento@uniroma1.it

Abstract. To help architects in real life work it is needed to clarify what a building is from their point of view. Till now we have seen that building design aid programs are mainly conceived from an "external" perspective, that of computer scientists. Difficulties related to architectural design support programs result from: an insufficient overall model of the building; an inadequate formalization of information; an underestimated complexity inherent in the design process. To overcome these difficulties we introduce a 'systemic' building model that takes into account discipline-specific goals by means of relation structures to relate entities of domains and ontologies to formalize knowledge.

Keywords. Design architecture; building organism; ontologies; collaborative design; situated design.

TO DESIGN ARCHITECTURE MEANS TO COMPUTE AIMS AND TO COLLABORATE

We agree with H. Simon's statement: "... when we use satisficing methods, it often does not matter whether or not the total set of admissible alternatives is 'given' by a formal but impracticable algorithm. It often does not even matter how big that set is. For this reason 'satisficing' methods may be extendable to design problems in that broad range where the set of alternatives is not given..." (Simon, 1996, p. 121).

This is exactly what happens in the Architecture/Building/Construction – ABC – sector, considering that design is also an exploration activity to solve contradictions regarding a solution domain that is not a-priori given, as any building is a singular, prototypical integrated and complex system in a changing context with interleaved problems.

The solution of a problem, in ABC and other sectors, is not simply a functional one expressed by an analytical formula or by generating solutions (Negroponte, 1975) as it refers not only to basic human needs (facilities providing an environment that is comfortable, useful, safe, with infrastructure) but also other needs that are difficult to formalize: the aims of construction. For instance: revitalizing a city – Guggenheim museum in Bilbao by F.O. Gehry; advertising building – Sony palace at Potsdamer Platz in Berlin; linking
two cultures – the English Channel tunnel; “keep the world in one” – Pocket PC.

Like humans, according to Linus Torvalds (Pekka et al., 2001), also computer science will be applied to non-functional programs such as:

“...progress is about going through those very same things as ‘phases’ in a process of evolution, a matter of passing from one category to the next. The categories, in order, are ‘survival’, ‘social life’, and ‘entertainment’.”

The same thing happens in the field of computer aided design:
• initially the concern was to apply the computer’s power of numerical calculation to mathematical formulae (expressions of physical phenomena) which need numerous iterations of calculation cycles and therefore, owing to the problem of rounding off (reduction to significant figures), it is necessary to have a higher numerical precision (e.g., Sperry-Univac, 18 bit (1) hardware and software for the Apollo mission; diffusion of F.E.M. applied to structural problems);
• tools were later developed to model and support interactions with other subjects-actors-agents by means of expert systems and later by the collaborative design paradigm (e.g. computer configuration software, XCON (2), infectious disease analysis, MYCIN (3), and collaboration like those of T. Sasada (2000);
• the time has come to use them to satisfy human aspirations (e.g., possessing the computer-object has aesthetic validity in itself - the iPad), or to be applied to the appeal of the relationship between perceived space-sensations to arouse in the subjects (e.g., a variable shape temporary pavilion in Seoul - Prada Transformers, fig. 1 and 2; Koolhaas, 2009).

We believe that the “boundary” between the cost-effectiveness of applying CAD techniques and the resolution of problems by human beings is currently shifting more and more towards higher semantic levels (Carrara et al., 2009) but this contributes to allowing the human being himself to carry out the action of design conception to a greater extent.

The aim of the present research is to shift this “boundary” forwards by providing a support for the real time verification of design and regulatory constraints in a delocalized working environment in which many specialists are involved.

However, we consider that the global overall ‘satisficing’ solutions of design problems can currently be attained, together with support systems, also by means of trade-offs among actors. We consider that an effective support system should allow actors to modify their own specialist goals and adapt their own specialist design solutions, as collaboration in building design is an inherent necessity (Kvan, 2000).

It should be pointed out that an actor in this paper is used in a more extended meaning than is customary in the literature (Wix, 1997) as it can also represent a human-like actor that has an active role in the process: it passes the Turing test, it can be a partner in a design process – it is a true intelligent assistant. Active means dynamic, as it can modify its definition (ontology) on the fly; and situation sensitive, as it can

Figure 1
Temporary pavilion, R. Koolhaas, Seoul, 2009. The four configurations: 1 - Fashion display, 2 - Movie showing, 3 - Art exhibition and 4 - Special event.
modify its behaviour along with the “context” (in a broad sense, condicio, Carrara e Fioravanti, 2004).

Moreover collaborative work implies that actors “have mutual and joint interests as the overall outcome success/failure is shared” (Carrara and Fioravanti, 2008, p. 1416).

To be effective, this kind of collaboration means that different specialized actors working on a (single) project have to communicate and mutually understand each other in order to be aware of the problems of others. One of the key points is thus the formalization of entities and their interfaces (entity/entity or entity/human of the same specialist domain or of different ones).

DIFFICULTIES INHERENT IN ARCHITECTURAL DESIGN SUPPORT PROGRAMS

In the ABC community a number of efforts have been devoted to overcoming these problems in order to integrate competencies into a single application program and to share knowledge. Among the various initiatives, we mention BIM and IFC, which are mainly devoted to defining and linking the entities of a building and to facilitating data exchange. But design is much more than (less or more accurately) describing a component of a building as it is an activity aimed at helping the actor to conceive artefacts, to record expertise, to implement experience-based rules of design and at “… changing existing situations into a preferred ones” (Simon, 1996, p. 111). Consequently, the effective formalization of information, in a broad sense, remains an unsolved problem.

This kind of difficulty is due to the lack of:

1. An overall model of the building that is representative of its complexity and effective for actors, capable of introjecting aspirations and processing them.

   In any specialist domain an actor involved in a design process manages his/her own entities in order to attain his/her own specific goal in a collaborative work. The formal representation of BIM and IFC does not contemplate this aspect as they consider a building as the sum of entities of a class (Class ≡ hierarchy structured set of entities). In the same way a watch is not just the sum of a crystal, a dial, hands and a movement, given that it is possible to construct other mechanisms with the same elements, but only one assembly forms the system that can tell the correct time.

   A building is actually a system: several classes directed towards a goal (e.g. habitability, energy saving, constructability, etc.). This goal-oriented view is attained through several objectives and sub-objectives, e.g. habitability includes space usability, ergonomics, space brightness, reciprocal disposition of spaces, relationship between spaces and the outside, etc.

2. An adequate formalization of information pertaining to any individual actor and exchanged among the various actors.

   A wide variety of computing and representation software is available on the market that is capable of performing even relatively complex tasks within well-defined disciplinary boundaries although it is designed to enhance the capacity to verify a given design approach rather than to help find a solutions’ conception or choose among hypotheses. These software applications are actually, in that sense, of no help in design collaboration, and indeed even make it more difficult: software specialization...
increases the difficulty of communication and reciprocal understanding among the various actors, as data required by the different specialized programs differ from one actor to the next.

Moreover, each type of software demands the input of data that must generally be inferred from the interpretation of the documents of the design solutions of the other actors involved in the process. In this way different interpretations of the meanings of a “same” entity are a cause of misunderstandings that are all the more detrimental to the overall outcome the greater the degree of complexity involved.

3. Difficulties inherent in the design process.

In building design not only is the ‘transformation’ of meanings and of characteristics of a (quasi-same (4)) entity extremely important, going from one actor (or specialist domain, e.g. architectural composition in which a wall may represent an internal partition of the building, fig. 3a) to another actor (or specialist domain – e.g. Building Science where a wall may be a shear wall, fig. 3b), but also the dynamics of these interactions between actors over time, thus the ‘history’ of the project – it is a data and process driven phenomenon like a narrative novel.

The exchange of project information and knowledge among the actors (or disciplines) is not given once and for all. The ‘translation’ of the entities among the actors is not fixed for all time. Above all, this exchange, and accompanying modifications, is not occasional but continual: the actors work (design) the (quasi-)same elements-entities. The development of the project is a process whereby, through various phases (brief, preliminary, detailed, constructive and maintenance) its definition is achieved. The activity of actors in the process may be serial or parallel, the work on the entities-components may be disjoined or concurrent, but what is certain is that problems of consistency, versioning, authority, intellectual property, privacy, etc., arise.

These types of difficulty must be set within the framework of the design intended as a system process to realize a project. Design is a means for expressing a project; it is an aptitude of design will: it is an act of faith (Cross, 1984, chap. 9).

The first difficulty is essential from the standpoint of the project as defined above; the two others are ancillary and modal in nature: formalization is important but only in so far as it can serve the function of being computed in order to represent a goal; the process model is required in order to lawfully govern the procedures and effectively to govern the interventions by the actors vis-à-vis the project solution.

The entire building (and the design process underpinning its gradual definition) possesses a number of goals–rules for different actors that are independent of the specific project and often are not consistent among them. Beside these goals–rules there are many ones that are dependent of specific project and context.

Figure 3a (left)
The wall as a diaphragm between spaces – Villa Tugendhat Mies Van Der Rohe, Brno, 1928-30;

Figure 3b (right)
The wall as a shear wall in a stress simulation program.
The “Rule of rules” to be implemented does not exist: the human being (and the support systems) are in need of a progressive refinement of their capacity; for this reason, there are faculties of architecture and engineering on the one hand and knowledge based systems capable of self-modification and updating in the face of concrete problems, on the other.

A typical design aspiration of the actors is the will to construct buildings that are suitable for their times, for which two fundamental requirements must be satisfied: a strong link between interior and exterior and a proper environmental sustainability. In achieving this, two conflicting classes of requirements may emerge, which are a demonstration of the absence of a “Rule of rules”: the requirements of maximum transparency and minimum dispersion (e.g. the Fondation Cartier, Paris); or the aim is to achieve a building-sculpture for promotional purposes in which the articulation and the ‘movement’ of the forms is in contrast with the minimum surface goal required by energy saving (Vitra Museum at Weil-am-Rein); or else when the thinness of a highly insulating wall clashes with the cost of the envelope (Change Phase Material wall).

These simple examples show how architectural design operates on the contradictions. In our view the synthesis of the above is the elective terrain of the designer and the reasoning tools (inferential, deductive, inductive, Bayesian networks, etc.) ‘external’ to the entities-building components in question or in some cases rules (and objectives) ‘internal’ to the same entities, extend and boost the potential of the human mind.

A comparison may be made with the game of chess: there are rules ‘internal’ to the individual pieces-entities which consists of ‘mechanically’ or better said ‘typographically’ (Hofstadter, 1988, Chapter III) applicable deductive rules (e.g., how the various pieces move on the board) and other rules governing their reciprocal movement that require higher level models for the configuration of scenarios (e.g., castling, opening, sacrificing certain pieces, end game); others again are ‘external’ to the chess system and are related to global strategy of the game (fuzzy, predictive, trying to edge adversary’s pieces, etc.), or act at and even higher level of human motivation, that of beliefs (e.g. finishing a chess game quickly in order to go and have a game of tennis).

It is appropriate that the latter rules, the ‘external’ ones, should pertain to the human being also for another reason: unlike chess, in which there is a huge but finite number of possible combinations of patterns of pieces on the board, the ‘pieces’ of the building are increasing and evolving constantly; the rules governing the economic, social and productive context are also changing constantly.

One further aspect that should be considered and which is often underestimated is that reasoning (or the rules) are not applied mechanically or slavishly to all the entities of a class or to lower classes but often actually depend on how these entities are mutually related in the history of the individual project.

A NEW BUILDING MODEL AND A BETTER ENTITY FORMALIZATION

To overcome these difficulties we propose:

1. Using Relation Structures to relate entities to a “systemic” building model.

To make this possible, in a specialist domain, entities of one class and others of another one, are related to each other by means of a specific relationship, a Relation Structure – RS – which an inference engine can use to compute a goal (Carrara and Fioravanti, 2001).

With reference to buildings, there are two fundamental ontology classes: that of the spaces and their aggregations, which in a project go to make up the so-called ‘Spatial Class’ domain, and that of the physical elements (components) and their aggregations which in a project make up the constructive apparatus, defined by UNI (Italian Standard Organization) as a ‘Technological Class’ domain (fig. 4).

For instance, an architect can conceive a building like a system made up of entities of these two classes (spaces and components) plus his/her
own specific Relation Structure that applies its inferential engine habitability rules to linked entities of the two domains: there are many possible assemblies of building components but only a few of them are space aggregations where people can ‘satisficingly’ live or enjoy better lives (fig. 5).

2. Using ontologies to formalize knowledge

Another fundamental requirement for overcoming the above problems is to understand knowledge, namely technical knowledge. Technical knowledge concepts can be formalized and structured by means of the technology of ontology, in order to define entities and by means of explicit semantics to define their meanings.

In the present context, knowledge refers to the fields of architecture, energy saving, sustainability, building stability, etc. and its entity definition includes both the formal structure of the entities considered in a project (and the related aspects, i.e. meanings, geometry, properties, relations, etc.) and the formal models (generally mathematical) that allow simulations, verifications and reasoning to be performed (Carrara et al., 2009).

3. Using Filters to deal with the actor’s perspective

This model of a design process is similar to Multiverse theory where as many universes exist as there are measurement operation results: many specialist domains are seen and interpreted by actors. This view reveals that actors work together on the same footing, and that each of them from his/her own point of view acts in a situated context (Gero and Reffat, 2001).

As ABC design is a multi-disciplinary sector it needs entities (concepts) of an ontology of a specialist domain to be linked to those of another one.

![Building Object Diagram](image)

Figure 4
The Building Object made by Space System – Ω (right) and Technology System – Ω-1 (left): correspondingly subdivided into Spatial Domain and Elementary Space Domain, and Technological Domain and Material Domain.
Likewise, using Relation Structure within a single disciplinary domain, we can map entities of several different specialist domains (ontologies) to link meanings by means of a Filter mechanism (Fioravanti, 2008). This way actor can become aware of the situated design solutions and problems of others.

ONTOGY REPRESENTATIONS AS A MEANS FOR KNOWLEDGE ATOM REPRESENTATION

The chosen way to find an answer to these questions is the development of a Collaborative Architectural Design system based on Knowledge formalized by several Ontologies that can significantly improve collaboration between different specialists (Ugwu, 2005; Fioravanti, 2008).

One of the greatest difficulties in this field is how to rapidly formalize the prototype entities, which we may term a knowledge atom, making up the ontology of a specialist actor.

So far we initially took into consideration three types of formalization to model Systemic Knowledge of Building and its entities (components, building parts, characteristics, constraints, relationships, etc.) as explained in Fioravanti and Loffreda (2009), subsequently we concentrate our efforts in using Lisp and Protégé tools.

The first implementation in Lisp allows to manipulate the instantiation and the inference engine ‘on the fly’ and to modify the characteristics of the entities relatively freely and precisely, indeed ad hoc, but at the cost of a artisanal implementation.

The implementation of entities is very slow as programmers would have to be skilled in architecture and in computer science with the aggravating circumstance of no visual editor program aid.

The main characteristic of entities is related to the ‘type’ of entity: the membership ‘class’. This one is formalized by means of a custom made frame structure, similar to the one investigated by McCarthy (1960), by means of an AKO slot (A Kind Of). Our frame has a four-tier structure: frame, slot, facet, value).

The advantage of being able to manipulate also the type of an entity’s structure allows not only to change the inheritance of an entity but also to mix entity assemblies. The freedom we are given by this formal logic enables us to compose an entity of a class (whole-of) also from entities of different classes belonging from heterogeneous domains, for ex. a room of a ‘Spatial Class’ domain with a pillar of the ‘Technological Class’ domain.

In our case we implemented the System of Spaces which, together with the System of Technologies, contribute to fully defining a building so that the two systems (normally separate) can be interfaced directly through a shift of the inheritance relationship (AKO slot in the frame structure) with the assembly one (fig. 6).

At the time of instantiation this peculiarity makes it possible to simultaneously verify the constraints that are normally found on ‘orthogonal’ logical planes: classes and assemblies.
Using a plain and widespread ontology editor - Protégé - is the second approach. Each entity of the building was formalized according to the triplet ‘Meaning-Properties-Rules’ (Carrara et al., 2009; Fioravanti and Loffreda, 2009).

In this case, the distinction among entity ‘meaning’ (name and description), ‘properties’ (slots and associated attributes) and ‘rules’ is sharp and well defined. The rules, in particular, are formalized by means of SDK of Protégé, the PAL – Protégé Axiom Language; - they operate on the instances of the ontology and establish relations, constraints and specifications associated with the entities to which they are applied.

The constraints checking and verification, as it is separated from the definition of the entity, is not contemporaneous with the instantiation of the object and so the processes of verification and control of consistency, coherence and congruence are necessarily subsequent to the completion of the instantiation of the entities involved in the design solution. Another limitation is that each relationship of an entity should be specific: as its coherence check examines constraints one to one. For instance it is not possible to have a general relationship like “space_room_has_a” (wall, or door, or window, or etc. with some cardinality on each of its elements) but several specific relationships like “space_room_has_a_wall” plus “space_room_has_a_door”, “space_room_has_a_...”.

CONCLUSIONS AND FUTURE DEVELOPMENTS

On these premises a formal model of the structure of knowledge used in the design process as well as of its management has been developed, based on:

• a formalization of knowledge (by means of pure Lisp and Protégé,...);
• an inferential engine (how to infer and to compute knowledge);
• some goal-oriented Relation Structures (f.i., evacuation safety path).

Figure 6
An example of ‘R’ layer: Swapping between two domains: a possible ‘AKO shift’ between entities of a Technology Class domain – Ω-1 – with entities of a Spatial Class domain - Ω.
The paper innovates and clarifies two concepts: a systemic vision of buildings and a satisficing solution for goals.

The first one is not limited to the ABC sector but is generally true of all industrial sectors. Every sector, every disciplinary domain has a specific goal that can be formalized by means of a Relation Structure and an Inference Engine that jointly direct a space-component dipole towards a goal. For instance: void and particle towards material structure in physics, room and wall towards habitability in architecture, space and duct towards comfort in plant engineering, etc.

The second concept is related to the perceived quality of building usability (and before, constructability), but is not an aseptic and pernickety check list of requirements: building quality means men and women have to live in it in comfort and with aesthetic enjoyment.

Research has so far underestimated the role of men living in a “design space”: their concrete action interacts with the model of Space/Technology classes and with other men.

It can be seen as the spirit in the Zen conception that divides Man into three parts: body-mind-spirit. Likewise, the building can be seen as a merging of these “aspects”: the Space/Technology dipole (body), the Relation Structure (mind) and Man (spirit). A building is a living thing: the building organism (fig. 7).

REFERENCES AND NOTES


Hofstadter D 1988, Gödel, Escher, Bach: un’Eterna Gir-


Negroponte N 1975, Soft Architecture Machines, MIT, Cambridge, MA, US.


(1) That was more precise compared to its contemporary computers where a “word” was coded as 16 bits.

(2) In the eighties the Digital Equipment Corporation – DEC – developed XCON – eXper CONfigurer – an expert system to configure its computers based on “Rete” algorithm, a program by C. Forgy written in OPS5.

(3) In the early seventies at Stanford University a software was developed (“mycin” is the suffix many antibiotics have) to identify bacteria causing severe infections, such as bacteremia and meningitis, and to recommend antibiotics with the dosage adjusted for each patient’s body weight and other characteristics. It was written in Lisp.

(4) The same in the sense of referring to a “same portion” of the building; the example given refers to a wall between two slabs, and certainly not to the wall from the standpoint of the architect or the entire shear wall incorporating the wall as for the structural engineer.