A THEORETICAL MODEL OF SHARED DISTRIBUTED KNOWLEDGE BASES FOR COLLABORATIVE ARCHITECTURAL DESIGN

GIANFRANCO CARRARA, ANTONIO FIORAVANTI
Dpt. Architettura e Urbanistica per l'Ingegneria
Università degli Studi di Roma "La Sapienza"
Via Eudossiana, 18 – 00184 Roma – Italy

http://www.dau.uniroma1.it *gianfranco.carrara@uniroma1.it*
antonio.fioravanti@uniroma1.it

Abstract. This paper is a report on research in progress on Collaborative Architectural Design.
The proposed model, the resulting system and its implementation are referenced mainly from the point of view of architectural design and related branches, as envisaged in more advanced design studies. The model is not lacking in validity when its rationale is applied to other fields.
The research simultaneously pursues an integrated model:
a) of the structure of the net based architectural design process (consisting of operators, activities, phases and resources);
b) of the required knowledge (distributed and functional to the operators and the phases of the process).
The paper is focused on the second horn of the model: the structure of the distributed KBs of the Entire Building (EB).
The article is divided into three parts. In the first a classification is made of the support tools available to architectural design based on two paradigms: the Conventional Method and Collaborative Design. In part two a description is given of the overall model structure, the stratified structure of the Knowledge Bases (Common, Specialized, Project) and that of the EB.
The latter is represented by means of the Space System and the Building System. Lastly, a description is given of the atomic elements of the Knowledge Bases - the objects-components; their characteristics in a collaborative, distributive context, in the presence of constraints and objectives that vary according to the operators and the context.
In part three a possible implementation of the model is discussed.

1. Introduction

The architectural design business is marked by a progressive increase in operators all cooperating towards the realization of buildings and complex infrastructures (Jenckes, 1997). This type of design implies the simultaneous activity of specialists in different fields, often working a considerable distance apart, on increasingly distributed design studies.

Collaborative Architectural Design comprises a vast field of studies that embraces also these sectors and problems. To mention but a few: communication among operators in the building and design sector; design process system logic architecture; conceptual structure of the EB; building component representation; conflict identification and management; sharing of knowledge; and also, user interface; global evaluation of solutions adopted; IT definition of objects; inter-object communication (in the IT sense).

The point of view of this research is that of the designers of the (architectural) artefact (Simon, 1996); its focus consists of the relations among the various design actors and among the latter and the information exchanged (the building and its components).

Its primary research goal is thus the conceptual structure of the EB for the purpose of combining design activities, sharing knowledge, managing conflicts and developing possible methods of resolving them.

2. Scientific bases

At the beginning of the new millennium, design companies, especially architectural design ones, are characterised by a new advanced working method, Collaborative Design (Howard, 1997; Kvan, 2001), by which is meant a design process characterised by a continuous exchange of design information among all the operators involved, even across the customary interdisciplinary borders, and by asynchronous and/or concurrent working methods.

2.1. TWO PARADIGMS

In this field of investigation, two different schools of thought have emerged over time: the first, defined as the Conventional Method, is to provide a low-level shared data exchange platform enhanced by specific applications for each discipline (Björk, 1992); the second, called Collaborative Design, is based on a semantically rich Common Knowledge Base upon which the various professional profiles may draw (Turner, 1997), Fig. 1.

The following researches may be included in the first paradigm:

STEP - The main aim is stated in the name - STandard for Exchange Product data model. Its first and second implementations in the building sector, although attaining draft status, have never become ISO standards.

The manufacturing industry model was recently adopted in the building sector. Its main aim nevertheless still remains that of mere "product information" and the structure of its semantics is too simple to provide a valid support for building representations. Its generic nature however means that it can be applied to many different sectors.

COMBINE - COmputer Models for the Building Industry in Europe. Although based on the STEP-EXPRESS to transfer the conceptual schema (object, type, definition) and the "neutral files" STEP to transfer object instances, it is a prototype specialized in evaluating energy and HVAC performance (Amor et al., 1995; and 1998).

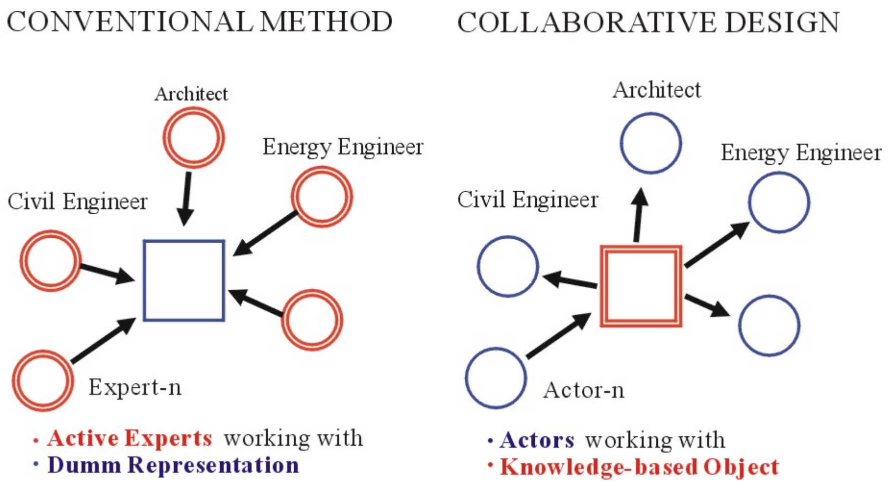

Figure 1. The two paradigms of Collaborative Design.

OXSYS - Oxford System. Software commissioned by the Oxford regional health authorities; it was conceived as an aid in the integrated design of hospitals using a predefined "kit of parts".

ICADS - A system based on a very small number of architectural objects (seven) which develops a set of controls according to the blackboard paradigm and a set of rule-based expert systems in the design of a building. The system is conceived for concurrent and distributed design; any cases of conflict are signalled to the operator, although no knowledge or strategies are provided to overcome them (Pohl and Myers, 1994).

The following systems may be considered as belonging to the second paradigm:

KAAD – Knowledge-based Aided Architectural Design. Strongly design aid oriented (Galle, 1995). Its strengths: semantic richness, the large number of building components used, multiple inheritance, being open and scalable. Composed entirely of objects, also its graphic representation is additive. Allows constraints to be activated and de-activated, and new ones created. It does not determine a single top-down or bottom-up design flow. However, it is restricted to a single knowledge base (Carrara, Kalay, and Novembri, 1994).

EDM and EDM-2 Engineering Data Model. Developed to represent a building having different levels of abstraction, building technologies and projected uses. These aspects are represented by the objects BOUNDED_SPACE, CONSTRUCTED_FORM and ACTIVITY, respectively. It too is open and scalable. One of its drawbacks is the separation of the representation of the building from its geometry. This prevents any control being exerted over the consistency of the geometric data with the space attributes of BOUNDED_SPACE except by means of a calculation activated each time by each application (Eastman, Bond and Chase, 1991; Eastman and Siabiris, 1995; Eastman, Jeng, and Chowdbury, 1997).

BDA - Building Design Advisor. Comprises different integrated performance evaluation applications (such as DOE-2, Superlite, etc.). It has a unified and known interface, supplies a representation of shared data and envisages format exchange tools. It acts as a repository for the values produced by the different applications. The control and maintenance of its semantic integrity is performed by external intelligent agents (Papamichael et al., 1996).

The first paradigm encounters the serious problem of the impossibility of modifying or increasing the applications, which are fixed like the shared data platforms at a low semantic level. They are often too generic. On the other hand, they are more practical to use and easier to implement.

The second runs up against the complexity implicit in the large number of complex object relations, and is characterized by a rough and ready implementation of objects and semantically rich relations.

These two paradigms are closely linked with the algorithmic and IT problems involved in Collaborative Design. These include building constraint management, the automated or semi-automated solution of inconsistencies, as well as the complex organization of the system to allow the exchange and merging of contributions from all Collaborative Design participants.

All these aspects are only partially resolved by commercial packages and applications deriving from industrial initiatives, such as that of the Bentley System for the IFC (Architectural Desktop, Project Bank, Allplan, Project

Wise, ReviewIT AEC2, Active Project, Volo View, TurboCAD, Expedition Express, etc.).

Unfortunately, even when highly sophisticated, these tools lack the general applicability and flexibility needed to manage all the constraints and knowledge involved in a complex project. Furthermore, commercial products have the drawback of being difficult to apply to new building components.

3. Ongoing research

3.1. THE OVERALL STRUCTURE OF THE SYSTEM

With regard to that previously explained, the research is placed in the second paradigm of Collaborative Design.

The Aim of the research is to develop instruments for a Collaborative Design through the management of knowledge. The knowledge which has been taken into consideration is that pertaining to the preliminary phase of building and architectural design, when the simultaneous collaboration of various professionals is most obvious and crucial.

We think that developing the components, based on the designer's knowledge, in such way that they are able to take into account the constraints and the consequences of each design decision will show up the inconsistencies and possibly suggest alternative solutions. These aspects are absent, or only marginally present, in the current commercial and non commercial environments.

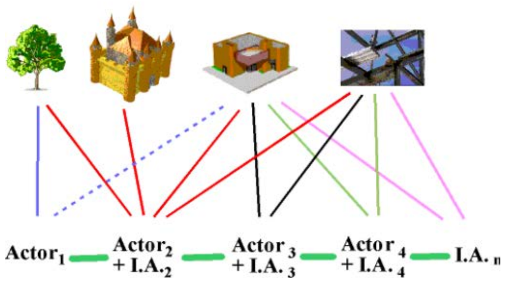

Figure 2. The relations among operators in a (Architectural) Collaborative Design context.

The structure of the software system model compliant with the customary effective design processes, will consist of a set of systems themselves consisting of: User Interface (UI), Data Bases DB, Graphics Primitives (GP), Network Programs (NP), Knowledge Bases (KBs), an Inferential Engine (IE), and Relation Structures (RS), Fig. 3.

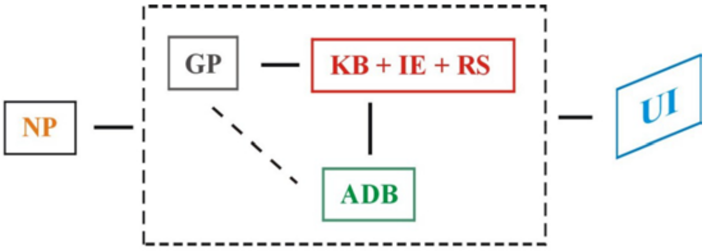

Figure 3. The structure of each system model representation.

The latters consist of semantic and/or operating relations among the elements comprising an EB, with regard to a given physical phenomenon and/or a design object, i.e. the case of energy saving. This goal takes into account the EB and configures the its Relation Structures (RS) as a software System in its own right.

The RS are not necessarily systems external to the KBs. For example, the goal of achieving a correct coupling of objects is contained in the objects themselves which know which surroundings are desirable or denied or acceptable.

The RS – Perspective (Fig. 4), allows each operator to see only the relevant information in a building being designed (≡ the instance created by means of Knowledge Bases).

Another application of the Perspective is when we wish to regroup several components on the basis of “conceptual clustering” methods (Carrara et al., 1995).

The RS – Filter, has the task of correlating the representation of an element used in an application with another representation of the same element used in a different application. This is done by transforming, in IT representation, the structure of the object. So, the Filter allows the interoperability of the objects and thus of the applications as well as other subsystems (Fig. 4).

In short, in order to achieve this, what are called net-objects will be introduced (with perspectives, plug-ins and filters), as well as the (bi-lateral and multi-lateral) net-constraints and lastly the net-context (of the state and stage of the project).

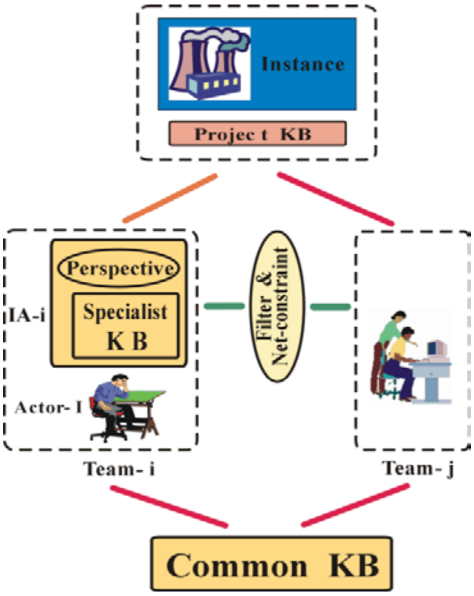

Figure 4. The Knowledge Bases, the actors and the instance of EB.

The running Knowledge Base for designing an architectural artefact (the EB) is constituted by:

- Common Knowledge Base of all actors (CKB);
- Specialist Knowledge Base, peculiar to each participant (SKB);
- Project Knowledge Base of the single project (PKB).

The three levels of the Knowledge Base are additive, Fig. 5.

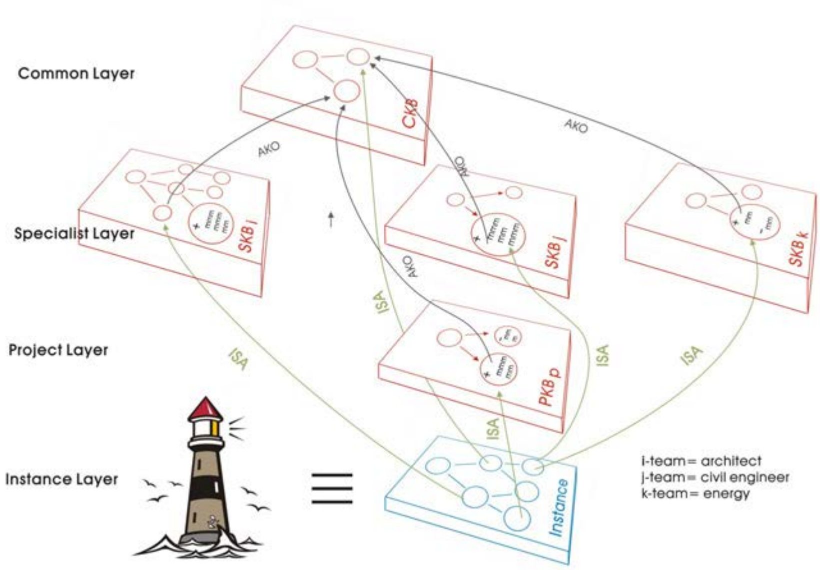

Figure 5. The Knowledge Bases and the Entire Building.

PKB is set up in addition to the conventional representation of objects in an EB, as it is necessary for design purposes. In fact, the need to create a new building component may arise. For example, in certain climatic conditions, there may be a need to define a new type of external wall, which cannot be derived from those of the CKB or from the SKB. It is useful to create this new wall because it will be in much demand during that particular project, although it does not necessarily go so far as to increase the other Knowledge Bases.

This new component will be part of the KBs in order to take advantage of inheritance, geometric representation, constraint checks, spatial congruence relations, and interrelations with the other building components in the project as well as verifying their performances.

The same may occur when introducing new constraints. For example, in health institutions for the aged, it is better if the bedroom is at least twenty metres from the lift; or in an art gallery, it is better if the lighting is higher than a given lux level.

3.2. THE STRUCTURE OF REPRESENTATION OF THE ENTIRE BUILDING

The structure of an EB can be viewed as being composed of some KBs and an ADB. Each project has its own ADB_p (Assembled geometry Data Base of the p-project) that goes alongside the $\sum KB_i$ (given by CKB + $\sum SKB_i$ +

PKB_p {i э Actors | i = 1, n}), since the two components are different (Fig. 6): the KBs allow for a dynamic verification of the performance and constraints activated at the moment data is inserted or of the creation of a new instance (Fig. 7) (both as a building component and as a constraint check); whereas the ADB allows for a rapid gathering of stored data, as the net surface of a floor to instant data of the project process (without utilizing the inheritance as its control) including the geometric data of the EB, allowing the spatial congruence of the objects comprising the EB to be verified.

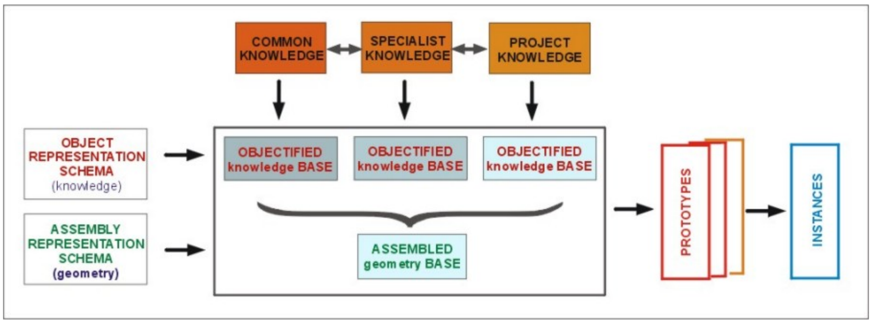

Figure 6. The structure of representation.

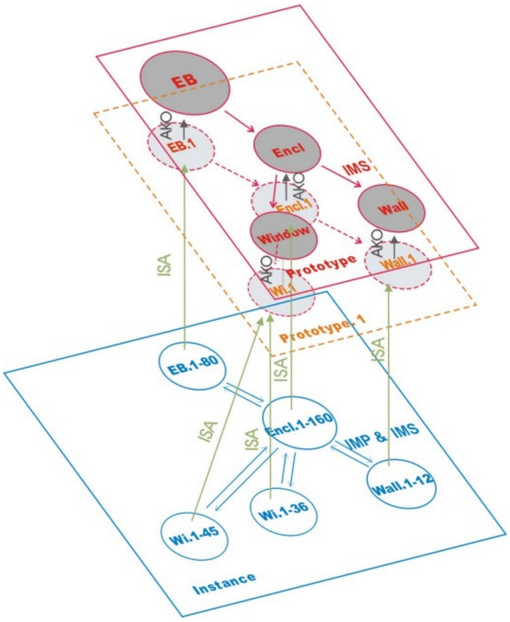

Figure 7. Relations among prototypes and instances for a typical wall.

The CKB, which both the SKB_i and the PKB_p conform to, has the following structure: two independent systems, the Space System and the Building System bound by “Relation Structures” (Carrara, Fioravanti and Novembri, 2000).

The Space System is the structured set of the equivalency classes of the space elements needed to carry out a specific activity throughout defined sets of performances. It is made up of two domains: Environmental Domain and Elementary Space Domain. The former consists of spaces contained by physical elements, such as BU and SU (Building Unit and Space Unit) components; the second is made up of space elements that are not directly surrounded by physical elements, such as ESs and PEs (Elementary Spaces and Partition Elements).

In a dual and symmetrical way, the Building System is the structured set of the equivalency classes of the physical elements needed to shape the Space System, fulfilling defined sets of performances. The B.S. is sub-divided into two domains: Constructive Domain and Raw Material Domain. The former is made up of constructive building components such as FSSs and FEs (Functional Sub-Systems and Functional Elements) that have complex performances; the latter is made up of materials that have only simple properties: the BEs and RMs (Base Elements and Raw Materials).

The division of each system into two domains is to some extent arbitrary, but it is linked to the design context. The domains of the ESs and RMs have a lattice structure. The Environmental and Constructive Domains have a semi-lattice structure since in the KB and in their elements only “whole” (also called IMS, IMmediate Successor) unidirectional relations are present (Fig. 7).

This is how each element of the KBs is able to have more than one “father element” and thus allows the multiple inheritance to be obtained. Thus with only a few simple elements many complex components can be obtained.

The challenge of the research is thus focused on the capacity of transferring knowledge to the objects, in order that they acquire “intelligent” behaviour and become “active” design elements.

The “intelligence” of the objects comprising the Spatial System, the Building System and the Relation Structure, lies in the possibility of them:

- having a semantics, knowing what they are through a definition thereof, through the hierarchical structures to which they belong;
- providing explanations, not only on their state but also on their behaviour;
- possessing rules of behaviour towards other objects, constraints, goals;

- having a coherent and accurate representation of the context, meaning by this the design phase and the applicational environment in which it is instanced;
- knowing their own positioning versus a reference system and also other objects (at any moment it is possible to verify the dimension constraints);
- being multi-hierarchical so that it can be ordered according to different criteria using Conceptual Clustering techniques (not only through properties explicitly defined in the object) (Carrara, Fioravanti, and Novembri, 1995);
- varying their own IT structure to relate to objects having a different structure in other applications (polymorphism).

Above all, however, all these characteristics must be dynamic, and linked to the specific phase of the design process.

This is due to the fact that the IT objects of the software system by nature practically never have characteristics with fixed values. These values are assigned and defined dynamically during the design work. Moreover, these values practically never reside in the objects themselves (except for the instances) but are computed by means of the inheritance mechanism from methods and procedures residing in the higher-level prototype objects of which they represent sub-types. In our case there will also be the peculiarity of having to change the structure of the prototype-object in order to correspond to the specialist KBs in the various disciplines and to the specific applications used by the various professionals.

For example, in a Collaborative Architectural Design the EB (and consequently the EB instance) must have the capacity to accept characteristics and procedures of the object deriving from several SKB_i of the i-actor. This possibility confirms the advisability of having a PKB_p (Project Knowledge Base of the p-project) in which a specific component prototype for the Project can be created dynamically and temporally.

3.3. CONFLICTS AND DESIGN OBJECTIVES

The capacity to provide explanations concerning several project or physical constraints is essential in order to resolve conflicts rapidly.

The resolution of conflicts as quickly as possible is a considerable advantage in improving the design process both temporally and cost wise. Whenever a conflict is not identified in time and the design activities are continued in the meantime, the construction of the EB may be impaired by unsatisfactory solutions. Indeed, when the designer modifies an instance (or creates new one from a prototype) that does not match the required constraints in that context, the object-instance signals not only that the design constraint has not been respected, but also explains why. In this way the designer is able to receive adequate explanations and thus redirect his design decisions towards respecting the constraints.

This situation often occurs in real-world professional studios because designers do not know all the implications of their design decisions.

Moreover, the capacity to provide explanations about a constraint not respected by another professional (or disciplinary sector) is a powerful aid in asynchronous design.

4. Implementation of the system model for Collaborative Design and conclusions

The focus of the research is the module of KB+IE+RS as drawn in fig. 3. In fact, it is beyond the scope of the present research to develop graphic primitives or new interfaces. For this purpose, applications will be used that allow industrial standards like IFC or international or de facto standards to be exploited.

Our project is aimed at developing the components, based on the designer’s knowledge, so that it is possible take into account the constraints and the consequences of each design decision, show up the inconsistencies and possibly suggest alternative solutions.

In the past the research group developed a support system for architectural design within only one working group, and thus with a single KB (Carrara, Kalay, Novembri, 1994), by using a frame language in LISP.

This was used even recently by Kalay (1998), thanks to its effective four-tier hierarchical structure - frame - slot - facet - value.

Everything described in the preceding chapters our choice of the multi-Representative mode Vs the multi-Modal representation (Rosenman and Gero, 1996), the fact that the operators of the design process vary in time and that all the skills, hence operators also, could be not known a priori) means that we must use computerized representation structures that are as flexible as possible.

From another viewpoint, with the experience acquired during the development of ad hoc software produced in-house at great cost in terms of resources, also therefore for basic software (e.g. for multiple inheritance, the creation of instances, to change the direct superclasses), it is preferable to adopt the standard ANSI structures of IT representation.

The choice thus fell on CLOS and MOP as, in agreement with the findings of authoritative researchers (Zang and Norman, 1994; and several researchers at PARC), its use in the case of distributed KBs was considered more appropriate.

However, CLOS with its fixed three-tier structure – frame – slot – facet, seems to be more constraining than the pure four-tier frame structure.

To get round this it was decided to transform also the slots and facets into objects - Intelligent Links (ILK) which, depending on the validation provided in a given disciplinary area, project phase, SW application context, project manager and object creator, allow the object structure to be varied.

In this way we free ourselves from the "frame problem" due to a purely encodigism approach (Bickhard and Terween, 1995). The frame problem arises when too many objects are needed to fully describe a complex set or "word".

The implementation is in progress and the development of this approach seems to promise well. The structure of the proposed representation appears adequate for satisfying the needs that effective design processes deal with, but a new problem arises: the bottleneck affecting the manual transfer of knowledge into KBs owing to the huge number of relations among objects – components – constraints - goals (due to the complexity of the building design process).

Acknowledgements

The research was funded by MURST, Ministry of the University, and Technological and Scientific Research; National Research 2000.

Acronyms and abbreviations

IFC Industrial Foundation Classes
CLOS Common Lisp Object System
MOP Meta Object Protocol

EB Entire Building

SS Space System
- BU Building Unit
- ES Elementary Space
- PE Partition Element
- SU Space Unit

BS Building System
- BE Base Element
- RM Raw Material
- FS Functional System
- FSS Functional SubSystem

KB Knowledge Base

ADB	Assembly Data Base
AKO	A Kind Of
CKB	Common Knowledge Base
IA_i	Intelligent Assistant of actor-i {i э Actor | i = 1,n}
ILK	Intelligent LinK
IMP	IMmediate Predecessor
IMS	IMmediate Successor
ISA	IS A
PKB_p	Project Knowledge Base of project-j {j э Project | j = 1,m}
RS	Relation Structure
SKB_i	Specialist Knowledge Base of actor-i {i э Actor | i = 1,n}

References

Amor, R., Augenbroe G., Hosking F., Rombouts W. and Grundy J.: 1995, Directions in modelling Environments, *Automation in Construction*, **4**(3), 173-187.

Amor, R.: 1998, A Survey and Analysis of Integrated Project Databases, *CIB-W78*, Sweden.

Bickhard, M.H., Terween, L.: 1995, *Foundational Issues in Artificial Intelligence and Cognitive Science*, Elsevier Science Publishers B.V., Amsterdam, NL.

Brusasco, P.L., Caneparo L., Carrara G., Fioravanti A., Novembri G. and Zorgno A.M.: 2000, Computer Supported Design Studio, *Automation in Construction*, **9**(4), 393-408.

Björk, B.C.: 1992, A unified approach for modelling construction information, *Building and Environments*, **27( )**, 173-194.

Carrara, G., Confessore, G., Fioravanti, A., Novembri, G.: 1995, Multimedia and knowledge-based Computer-aided Architectural Design, *in* B. Colajanni and G. Pellitteri (eds.), *Multimedia and Architectural Disciplines*, Proceedings of eCAADe 1995, pp. 323-330, ISBN/ISSN: 0-9623687-1-4.

Carrara, G., Fioravanti, A., and Novembri, G.: 2000, A framework for an Architectural Collaborative Design, *Promise and Reality: State of the Art versus State of Practice in Computing for the Design and Planning Process*, Proceedings of eCAADe 2000 18th Conference, Dessau, pp. 57-60.

Carrara G., Kalay, Y.E., and Novembri, G.: 1994, Knowledge-Based Computational Support for Architectural Design, *in* G. Carrara and Y.E. Kalay (eds.), *Knowledge-Based Computer-Aided Architectural Design*, Elsevier Science Publishers B.V., Amsterdam, pp. 147-201.

De Martino T, Falcidieno B. and Hassinger S.: 1998, Design and engineering process integration through a multiple view intermediate modeller in a distributed object-oriented system environment, *CAD* **30**(6), pp. 437-452.

Eastman, C.M., Bond, A., and Chase, S.: 1991, A data model for engineering design database, *in* J.S. Gero (ed.), *Artificial Intelligence in Design '91*, Butterworth-Heinemann, London, pp. 339-365.

Eastman C.M., and Siabiris, A.: 1995, A Generic Building Product Model incorporating Building Type Information, *Automation in Construction*, **3**(4), 283-304.

Eastman C.M., Jeng, T.S., and Chowdbury, R.: 1997, Integration of Design Application with Building Models, *in* R. Junge (ed.), *CAAD Futures 1997*, pp. 45-59.

Galle P.: 1995, Towards Integrated, 'Intelligent', and Compliant Computer Modeling of Buildings, *Automation in Construction* **4**(3), 189-211.

Jenckes, C.: 1997, Nonlinear Architecture. New-science = new architecture?, *Architectural Design* **67**(9/10), 6-9, 14-31.

Jeng, T.-S., Eastman, C.M.: 1998, A database architecture for design collaboration, *Automation in Construction*, **7**(6), 475-483.

Howard, R. (guest ed.): 1997, *Automation in Construction*, Special Issue Building IT 2005, **6**(1) 1-68.

Kalay, Y.E, Séquim, C.F.: 1998, A suite of prototype CAD tools to support early phases of architectural design, *Automation in Construction*, **7**, 449-464.

Kvan T. (guest ed.): 2001, Special Issue CAADRIA, *Automation in Construction*, **10**(3), 293-382.

Maher, M.L., Balachandran, B. M., and Zhang Dong, M.: 1995, *Case-Based Reasoning in Design*, Lawrence Erlbaum Associates Inc., NJ, USA.

Maver, T., Kim I., Liebich T.: 1997, Managing design data in an integrated CAAD environment: a product model approach, *Automation in Construction*, **7**, 35-53.

Papamichael K., La Porta, J., Chauvet, H., Collins, D., Trzcinski, T., Thorpe, J., and Selkowitz, S.: 1996, The Building Design Advisor, *Design Computation: Collaboration, Reasoning, Pedagogy*, Proceedings of ACADIA '96 Conference, Tucson, Arizona, pp. 85-97.

Pohl, J., Myers, L.: 1994, A Distributed Cooperative Model for Architectural Design, *in* G. Carrara & Y.E. Kalay, (eds.), *Knowledge-Based Computer Aided Architectural Design*, Elsevier Science Publishers, Amsterdam, pp. 205-242.

Rosenman, M A., and Gero, J S.: 1996, Modelling multiple views of design objects in a collaborative CAD environment, *Computer Aided Design*, **28**(3), 193-205.

Simon, H.A.: 1996, *The Sciences of the Artificial*, MIT Press, Cambridge, MA, USA.

Turner, J.A.: 1997, Some Thoughts on the Existence of a Generic Building Object, *in* R. Junge (ed.), *CAAD Futures 1997*, pp. 533-552.

Wasserfuhr, R. and Scherer, R.: 1999, Towards Role Based Management of Cooperative Design Process, *Concurrent Engineering in Construction: challenges for the new millennium,* Proceedings of CEC99, Espoo, Finland, pp.176-184.

Zhang, F., Norman D.A.: 1994, Representations in Distributed Cognitive Tasks, *Cognitive Science*, **18**, 87-122.