Dynamic Generation of Design Plans at the Brief Stage

Mustafa Alshawi

Department of Surveying
Salford University
UK

The traditional approach to design and construction suffers from many limitations. As the technology becomes more available to the average users, the need for an effective and efficient solution has never been greater. This paper introduces an alternative approach to the life cycle of construction projects "application controlled process". Based on this approach, a framework for an Integrated Construction Environment (ICE) has been developed and implemented in a prototype demonstrator "SPACE" (Simultaneous Prototyping for An integrated Construction Environment). This paper is only concerns with those parts of the ICE which are relevant to the dynamic generation of design drawings. The NIRMANI system aims at generating a schematic design by retrieving previous design solutions that matche the problem specification from a multimedia case library. While the Bay Design Systems aims at re-adjusting the produced design solution to minimise construction problems.

Keywords: integrated environments, case-based design, project life cycle, integrated construction environment.

1 Introduction

Design and construction of buildings in general proceed sequentially, coupled through annotated set of architectural and engineering drawings. Firstly, the design process is a cyclical one with a linear progression [1]. It normally starts with a set of clients requirements which are communicated to a design team. Such requirements are cyclically converted into a design brief, with the client consent, which is then transferred into a design solution over several design stages [2]. Secondly, designers do not often anticipate the implications of their design on construction and contractors interpretation of the design solution often does not meet designers intentions. This separation of design and construction processes has not only led to the decay of integration but also to a growing misunderstanding of the role of each profession.

This traditional approach to design and construction suffers from many limitations:

(1) The design process usually takes a considerable time depending on the size, complexity and nature of the project.

(2) There is a tendency for clients and designers to be biased prematurely towards a particular design solution. This may occur due to many factors, such as urgency of design, high cost of alternative consideration, lack of resources, etc. [3].

(3) Design solutions do not usually meet budgetary constraints especially at early design stages [4].

(4) Weak communication between members of the design team due to their different design perspectives [5].

(5) Large percentage of construction problems on site are caused by complex designs [6].

An integrated computer environment can address the above problems where design and construction applications can be integrated under one environment. Such an environment should not be developed to suit a pre-defined set of applications which could limit its usage. It should be designed to integrate "industry standard" software, relatively inexpensive to run, and flexible enough to accommodate users' experience, in-house databases as well as future needs and requirements. If this is developed and implemented effectively, it could significantly shorten the design stage, eliminate data redundancy, maintain up-to-date information, improve accuracy and speed of data transfer, and creating a suitable environment for concurrent engineering. The latter can significantly improve design solutions as users can quickly alter an application's output and examine its impact on other applications.

The complexity and the vast amounts of information involved in any construction project and the lack of standards have made the process of producing an integrated environment very difficult. However, with the advances in knowledge engineering, programming techniques, and technology, the process of integrating various design\construction stages and sharing data by various applications is now possible and are likely to occur within one computer environment. At Salford University, the AIC (Automation and Integration in Construction) research group have developed and implemented a framework for a full Integrated Construction Environment (ICE) within which a project's information is manipulated to serve various design and construction applications. This paper briefly discusses the conceptual representation of the whole environment and explains in detail those parts of the environment that are related to the dynamic production and presentation of the design plans.

2 The proposed approach

The design and construction of projects in the Construction Industry can be described as a "stage controlled process". Traditionally, the life cycle of a project can be divided into six major stages; conceptual, design, tender, construction, occupation and refurbishment, Figure 1. Generally, each stage has a pre-defined output which is considered as the main input for the next stage. For example, at the conceptual stage, it is expected that the client's requirements are converted into a design brief where the latter is the major input for the design stage. The output of the design stage, e.g. a set of design drawings, is the major input to the construction stage. However, each of these major stages can be broken down into several sub-stages. For example, the design stage can be broken into the five sub-stages according to the RIBA plan of work. These sub stages might have cyclical phases among themselves or their input might slightly overlap with the input of the other stages. In the main, the six stages are almost mutually exclusive.

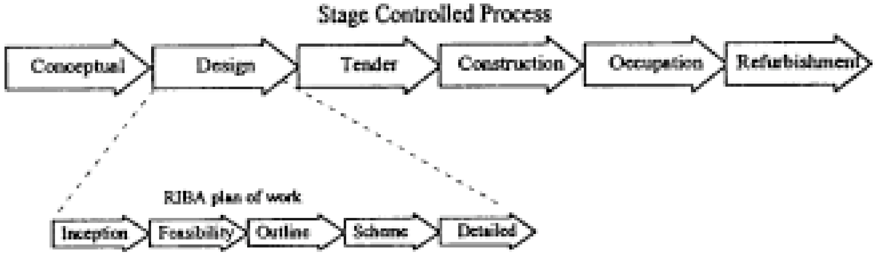


Figure 1: Traditional Life Cycle of Construction Projects

The proposed approach takes an alternative view to the traditional staged controlled approach i.e. an "Application Controlled Process", Figure 2. This approach emphasises the construction applications and their associated tools rather than the life cycle stages. Construction applications such as design, estimating, construction planning, site layout, facility management, etc. which are normally carried out at various stages of the life cycle of a project, can be run and their data accessed at any time and by any professional party. As the design progresses, the project's data is continuously updated and made available to all other construction applications. Centred around an object oriented technology, construction applications can be triggered off at any time, therefore enabling concurrent engineering to be carried out with relative ease. All construction professionals i.e. clients, design team, contractors, users, etc. can simultaneously share the project's data while having the opportunity to access the data which is relevant to their perspectives. For example, clients can visualise the project in 3D and/or find out whether or not the current design meets their budgets, while designers can examine the impact of their design on construction thus giving them opportunity to improve on the constructability of their design.

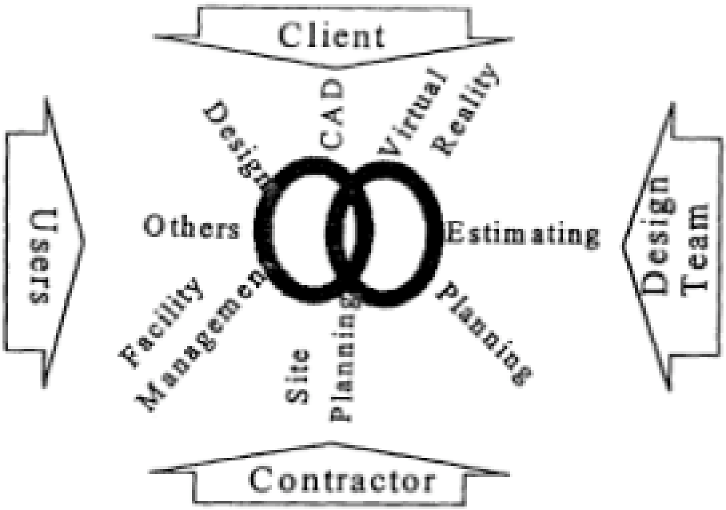


Figure 2: An Application Controlled Approach

3 Framework for the integrated construction environment (ICE)

Based on the above approach, a framework for an integrated environment is proposed which consists of three main parts. As shown in Figure 3, these parts are; brief and design generation, central data models, and external construction applications. The first part transfers clients early needs and requirements into a complete design brief. This is then fed into an AI tool which retrieves the best matching design plan based on previous design cases where users can adapt the retrieved drawings to their own requirements. This process embraced multi-media tools in order to enhance the quality of the communicated information. The output of first part is materialised in a set of design drawings. The grid layout dimensions, type of materials, superstructure, etc. are then fed into the Bay Design System" where the constructability of the design is checked. The system re-adjust the grid layout and other elements of the building and produces a vertically and horizontally co-ordinated design to minimise construction problems. High level representation objects of the drawings are then transferred into the central core of the ICE, part 2.

The central data model (the central core) represents a repository for the ICE whereby all information related to a project type, e.g. concrete office buildings, is maintained [7]. It consists of a number of data models which describes the life cycle of the project type under consideration. These models have been classified as building data model and other data models. The building data model mainly describes the physical features and behaviour of the project type and support different levels of data abstraction. The extent and structure of the building data model depend on the scope and the main objectives of the ICE. Different ICEs may have different structures of building data models, level of abstractions, different coverage of the project's life cycle, etc. Other data models represent data required by the various applications within the ICE such as Construction Planning Data Model, Site Planning Data Model, Specification Data Model, Estimating Data Mode, etc. Each data model should be developed to achieve a well defined set of objectives and to fulfil the need of a particular application. For example, a construction planning application requires a data model to support the information required to produce effective construction plan by the package e.g. generic construction activities, resources available, construction methods, etc.

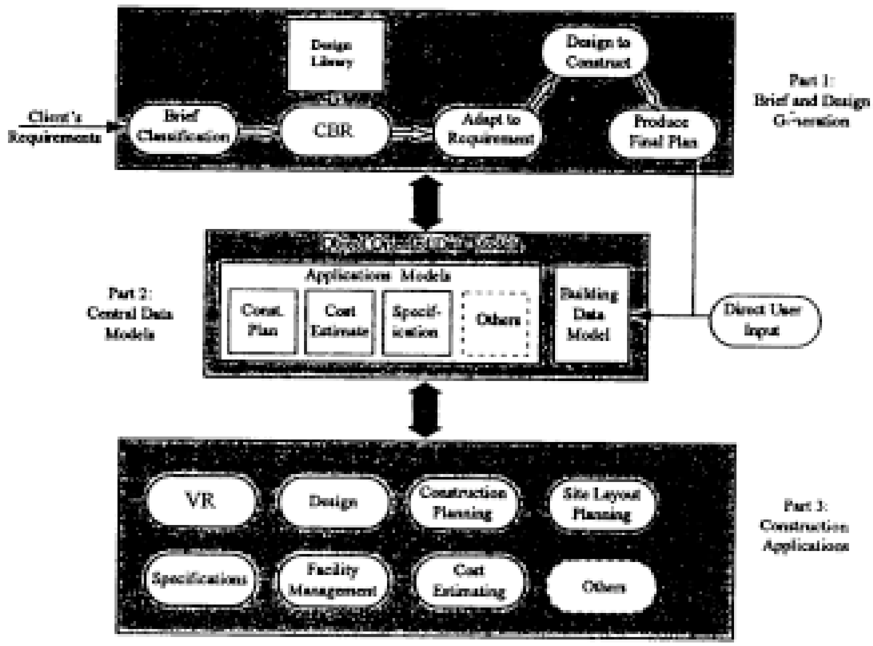


Figure 3: A Conceptual Representation of the ICE

The third part of the integrated environment represents the down stream construction applications such as design, construction planning, estimating, site planning, virtual reality modelling, etc. Such applications could either be external, i.e. existing stand alone applications, or internal i.e. developed within the ICE software tools. In either case each application has

(1) a built-in user interface where users can manipulate the output information, and

(2) a specially developed two way communication channel to transfer information between the application and the central core at real time.

Such an environment is normally triggered off by feeding in project specific information through a design package e.g. a CAD system whereby large amount of projects information can be extracted [8]. As the design progresses, design information is dynamically transferred to the central core, at a high level format i.e. design elements [9]. Once the central core is populated with the project specific information, users can run any other application at any stage of the design. For example, the cost of the so far developed design can be dynamically determined by running the estimating application. The central core, in this instance, transfers the design elements along with their specifications and Quantities to the estimating package to produce either the total cost or cost break down of e current design product. Users should be able to alter the generated cost Figures, add cost constraints on certain design elements, etc. using the estimating package interface. This altered information is then transferred to the central core where actions are triggered-off if the altered information is non-feasible or does not comply with regulations, standards, inhouse database, etc.

Users can switch between various applications at any stage at any time. Applications respond to requests from the central core and their behaviours are limited to the supplied amount of information. This paper is only concerned with the first part of the ICE where both the dynamic generation of design drawings and the design to construct principles are briefly explained.

4 The integrated case-based design & estimating system (NIRMANI))

This section describes the concept behind the dynamic generation of the design drawings, whereas a detailed description of the approach is explained in Perera et al, 1995. Research is currently limited to Light Industrial Buildings LIB (e.g. warehouses) where a working prototype is being developed to demonstrates the validity of the concept. The research is implementing a Case Based Design (CBD) technique.

The NIRMANI system (semantically described in Figure 4) aims at generating a schematic design for LIB by retrieving previous design solutions that matches the problem specification from a multimedia case library. The retrieved design can be adapted if required architecturally, structurally, or for services requirements, which ultimately can provide a cost plan for the building to form as the budgetary guide for further design developments. Re entire CBD process is interactive giving the designer sufficient authority to guide the design process and achieve creativity as much as possible. The various stages of the case-based design and estimating process are briefly explained below.

(1) The NIRMANI system starts a session using an initial brief where each criterion, e.g. number of occupants, total cost, shape etc., in this initial brief is weighted with the aim of matching them with those in the cases library. Cases which match the initial criteria are retrieved and ranked according to their matching criteria.

(2) The multimedia case store allows designers and clients alike to visualise any preferred case from the retrieved cases using 2D and/or 3D images, video clips, scanned images (photographs), text etc.

(3) Designers can evaluate and confirm the selection of the most preferred case.

(4) The system maps all the current design information to the new case, thus creating a design brief, i.e. building specifications, for the new situation which is based on the original case.

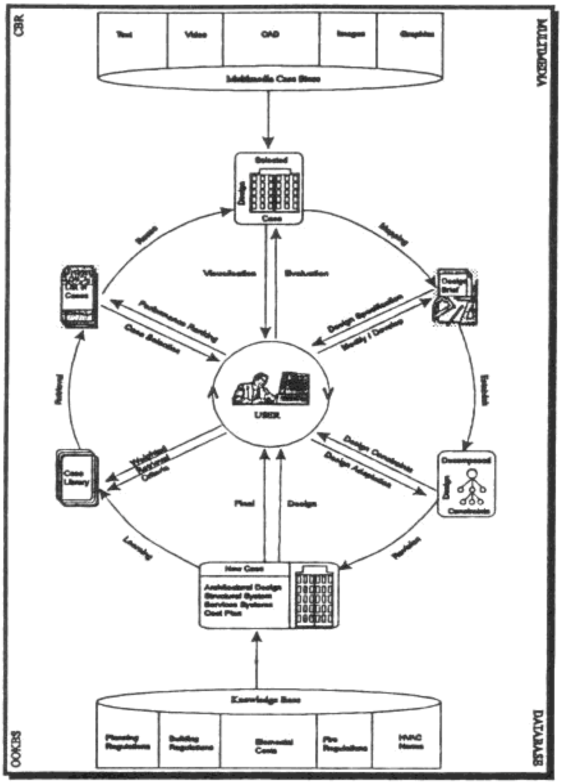
(5) Designers can modify and develop the brief to suit their requirements.

(6) The system compares the developed brief and requirements with the original case and establishes design constraints for each decomposed design perspective/element.

(7) The system adapts an interactive adaptation process to satisfy design constraints using knowledge obtained from the knowledge base in the form of methods and rules. At this stage, designers can revise the design in order to meet the new requirements.

(8) Once constraints are satisfied, i.e. to a level acceptable to the designers, and designers accept the new design, the design is stored in the case library as a new case for the system, thus completing the CBD cycle and learning from the new experience, i.e. the system conforms to the dynamic memory model (11, 12).

The NIRMANI system maintains the many perspectives of the design professionals involved in the design process but avoids information storage on the perspective of the client in order to rationalise information storage and minimise redundancy. Further, at a given time the client or user may take any of the perspectives of Architects, Structural Engineers, Services Engineers & Quantity Surveyor or all of it.



5 Critical comparison of NIRMANI

The CBD systems developed for the construction industry and found in the literature can be classified as

(1) Design Aid Systems e.g.: ARCHIE, ARCHIE II, MEMORABILIA

(2) Design Generative Systems eg. CADSYN, ACABAS, CADRE.

The proposed NIRMANI system mostly falls into the second category with a component of knowledge guiding the adaptation process. Most of the other systems deal with either architectural design or structural design accept ACABAS which considers them both. The NIRMANI system deals with architectural schematic design, structural grid optimisation, services system selection, and cost planning. Thus, it attempts to give a comprehensive design. In terms of design re-use, CADRE [13] and ACABAS [14] [15] use a manual case selection procedure whereas ARCHIE [16] & MEMORABILIA [17], [18] selects matching cases to provide guidance to the designer but not for direct re-use. Only SEED [19], [20], a system for the conceptual stage, proposes a complete re-use of design for design generation.

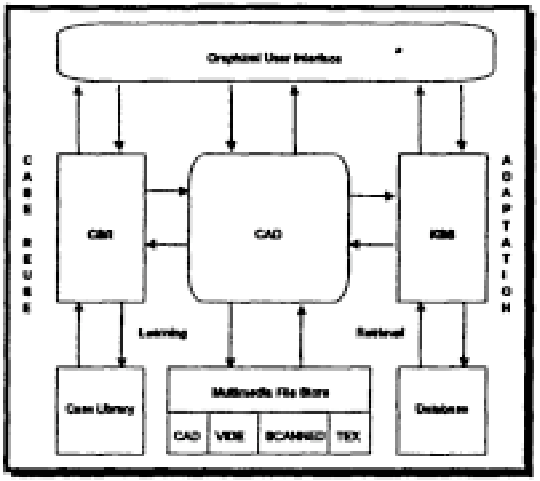


Figure 4: An overview of the NIRMANT prototype

But it is more of a design generative system using first principles of design with a component of case-based reasoning to assist the design generation task. The NIRMANI system, on the other hand, provides automatic retrieval of the most appropriate case directly for adaptation and re-use.

Unlike most CBD systems developed, NIRMANI is envisaged to learn by itself, acquiring new cases as it generates them. Furthermore, case indexing will be hidden from the user thus providing greater enhancement in useability. This puts the system in line with SEED but clearly distinguished from other approaches that rely on considerable efforts to build a memory of cases [21], [22] ,[14] and [15]. But SEED uses a combination of space generation expert systems LOOS, ABLOOS & GENESIS [23], [24], [25] to develop the design whereby case-based support is provided where needed.

It is intended that case adaptation to be semi-automatic allowing a certain degree of user interaction. The adaptation process will prompt the user with alternatives and suggestions from which the user can select or confirm. This is, therefore, partially in line with CADRE [13] but more similar to that proposed by SEED [19], [20].

6 The Bay Design System (BDS)

The adapted design plans are then fed into the Bay Design System (BDS) for constructability improvements. This section briefly explains the concepts behind the BDS.

The DRS implements the design-to-construct principle [6]. The approach used to develop the system aims at breaking down the design project information into areas of intense construction problems such as superstructure, substructure, and finishes as defined by previous research in this field . Each group of information is analysed separately therefore scoping down the amount of information involved in the analysis process.

Focusing on the superstructure information group for mid-rise reinforced concrete office developments, a number of problems were identified and grouped. One group related to the ill co-ordination between the building s envelope elements and the frame. In another

group, the problems related to tolerance incompatibility between the building envelope elements and the frame. A final group related to the need for enhanced standardisation /repetition of elements within a project.

The information analysis focused on tracing information related to problematic construction areas back to their relevant design stage. A complete analysis of the design function was carried out where main processes and their information flows were highlighted. This then led to the establishment of three major activities for the improvement of the design related problems.

(1) Horizontal co-ordination. This activity includes the co-ordination between the grid dimension and columns cross-sections with the cladding horizontal dimension.

This is necessary to ensue that the cladding horizontal dimension, between two columns, together with the column's cross-section reflects the grid layout.

(2) Vertical co-ordination. This activity includes the co-ordination between the storey height and the beam depth with the cladding vertical dimension.

(3) Element Information Consistency. This activity covers the compatibility between factory tolerances of the selected types of cladding and the recommended tolerances of the selected frame type.

Such activities have to be incorporated into the design function in order to minimise the above mentioned site problems. This meant that a new design process has to be established i.e., re-engineering the design function . As a result, the above activities have been incorporated into a new process i.e. the Dimensional Bay Design System. This has then been mapped onto an object oriented analysis model which in turn has been implemented onto an object oriented environment.

Through its graphical interface, users can use the system with relative ease. Users, following the introduction screen which gives a brief account of the background behind the system and the objectives and scope of the system, enter information (specification) via four icons displayed on a menu screen. Each of the four icons; building dimensions", frame elements', "glazing requirements" and "aesthetics", reveal a submenu with a number of options (icons). The user selects an icon for entering the necessary data i.e. the building's horizontal and vertical dimensions, the frame element's construction type, the aesthetics of the building's envelope etc.

Once the user has entered the necessary information, they continue to select a "commence design" icon to trigger the system's main operation is conducted in the following sequence:

(1) Importing the appropriate cladding and lining from a database of standard elements. During such a task, each cladding and lining combination is set-up as a potential solution for which the remaining operations are to act upon.

(2) The horizontal co-ordination of each solution, adjusting the grid layout and sizing the columns to correspond to the cladding and lining horizontal arrangements. The window(s) are positioned and sized to fit between whole cladding and lining elements.

(3) The vertical co-ordination of each solution, during which, the floor to ceiling height is adjusted and beams are sized to correspond to the vertical arrangement of the cladding /lining elements. Again the size and position of the window(s) are determined ensuring the window is positioned between whole cladding and lining elements.

(4) Depending upon the selected cladding and lining types, working through each solution, a tolerance compatibility check is conducted between the frame and the cladding and lining elements. Any incompatibility found amongst the solutions, results in the deletion of that solution.

(5) Establishing the most optimum lining element for each cladding element, based on the least volume of concrete of the frame elements. Other options are deleted. The most optimum solutions are presented by a labelled icon in a graphical profile, again based on the volume of concrete frame elements. The user selects the icon for the solution they wish to view which is then presented in a textual and two-dimensional graphics. Finally the user has the option to change the current specification or to enter a totally new specification.

7 Conclusions

This separation of design and construction processes has not only led to the decay of integration but also to a growing misunderstanding of the role of each profession. Each stage of the traditional "stage controlled" life cycle of construction projects has a predefined output which is considered as the main input for the next stage. This process enforces the barriers between the various professionals involved in a project. This paper has introduced an alternative approach to the life cycle of construction project i.e. an "application controlled process . The new approach emphasises the construction applications rather than the life cycle stages. Centred around an object oriented technology, construction applications can be triggered off at any time, therefore enabling concurrent engineering to be carried out with relative ease. All users i.e. clients, design team, contractors, users, etc. are then able to access the relevant project information at any stage.

Based on this new approach a structured integrated environment has been described where all construction applications are integrated. The first part of this environment automatically generates design drawings based on previous stored cases and then improves the constructability of the design solution. This paper has only discussed those systems which are relevant to part one of the integrated environment. The NIRMANI system aims at generating a schematic design by retrieving previous design solutions that matches the problem specification from a multimedia case library. While the Bay Design Systems aims at re-adjusting the produced design solution with the aim of minimising construction problems.

The above approach has been successfully implemented, by the AIC research group Salford University, on a PC based integrated environment. Information is automatically transferred between the various applications through one central object oriented knowledge based system. Currently, the prototype demonstrator SPACE (Simultaneous Prototyping for An integrated Construction Environment) integrates design, construction planning, site layout planning, estimating, bill of quantity, elements specifications, and virtual reality in one environment.

8 References

[1] Hillebrandt, P. M., 1984, Analysis of the British Construction Industry, Macmillan, 1984.

[2] Mackinder, M., & Marvin, H. E., 1982, Design Decision Making in Architectural Practice, Research paper no. 19, York Institute of Advanced Architectural Studies.

[3] Harper, D. R., 1978, Building: the process and the product, Construction Press, London.

[4] Ferry, D. J., & Brandon, P. S., 1991, Cost Planning of Buildings, 6th edition, BSP.

[5] Aouad, G, Cooper, G., Brandon, P. 5., Brown, F., Child, T., Ford, S., Kirkham, J., Oxman, R., & Young, B., 1993, Integrated Databases for Design and Construction.

[6] Alshawi, M. & Underwood, J., 1995, Information Analysis Approach For Integrating Design And Construction, in print, The journal of Engineering Construction and Architectural Management.

[7] Faraj, I. & Alshawi, M., 1995, A Generic Approach For An Integrated Construction Environment, Send for publication, ASCE, Computer in Civil Engineering, Special issue on product and process modelling.

[8] Dym, C. & Levitt, R., 1991, Knowledge-based Systems In Engineering, McGraw-Hill Inc., pp 182-183.

[9] Alshawi, M., 1994, A Run Time Exchange Of Elemental Information Between CAD And Object Models: A Standard Interface, The International Journal of Construction Information Technology, 2 (2), pp 37-52.

[10] Perera, S., Watson, I., & Alshawi, M., 1995, Case-Based Design Approach For Integration of Design and Estimating, Civil-Comp 95, August, London.

[11] Schank, R. C. & Abelson, R. P., 1977, Scripts, Plans, Goals And Understanding, Erlbaum, Hillsdale, New Jersey, US.

[12] Schank, R. C., 1982, Dynamic Memory: A Theory Of Learning In Computers And People, New York: Cambridge Univ. Press.

[13] Hua, K., & Faltings, B., 1993, Exploring Case-Based Building Design-Cadre, AT EDAM, 7(2), Academic Press Limited, ppl3S-l43.

[14] Schmitt, G., 1993, Case-Based Design And Creativity, Automation in construction, 2, pp. 11-19

[15] Schmitt, G. N., 1993, Case-Based Reasoning In An Integrated Design And Construction System, In: The International Journal of Construction Information Technology, 1(3), pp. 39-51.

[16] Goel, A., Kolodner, J. L., Pearce, M., Billington, R., & Zimring, C., 1991, Towards A Case-Based Tool For Aiding Conceptual Design Problem Solving, E. R. Bareiss (ed) Proceedings: Workshop on case-based reasoning (DARPA), Washington, D.C. San Mateo, Morgan Kaufmann.

[17] Oxman, R. E., 1993a, INDEX: C Case-Based Reasoning Approach Of ContentBased Indexing For Design, In: The 8th international conference on application of artificial intelligence, 1, pp. 203-218.

[18] Oxman, R. & Oxman, R., 1993b, PRECEDENTS: Memory Structure In Design Case Libraries, CAAD Futures 93, Flemming U., and Wyk S. V. (ed.), pp. 273-287.

[19] Flemming, U., 1994, Case-Based Design In The Seed System, Automation in Construction, 3.

[20] Flemming, U., Coyne, R., & Snyder, J., 1994, Case-Based Design In The SEED System, ASCE, pp. 446-453,1994.

[21] Rosenman, A., Gero, J.S., & Oxman, R.E., 1991, What's In A Case: The Use Of Case Bases, Knowledge Bases And Databases In Design, Schmitt, G.N. (ed.), CAAD Futures 91, pp. 285-300.

[22] Hua, K., Faltings, B., Smith, I., Schmitt, G. & Shih, S. G., 1992, Adaptation of Special Design Cases, 2nd international conference on Al in design, CMU, Pittsburgh, pp 559-575.

[23] Flemming, U., 1989, More On The Representation And Generation Of Loosely Packed Arrangements Of Rectangles, Environment Planning, Building, Planing and Design, 16, pp. 327-359.

[24] Coyne, R. F. & Flemming, U., 1990, Planning In Design Synthesis: AbstractionBased LOOS, in J. Gero (ed.), Artificial Intelligence in Engineering V, 1: Design, Springer, pp. 91-111.

[25] Heisserman, J. & Woodbury, R., 1993, Generating Languages Of Solid Models, Proc. Second ACM / IEEE Conference on Solid Modelling and Applications, Montreal, Canada.