Title: **Early Building Design using Computers**

Author(s): Kene Meniru, Claude Bédard and Hugues Rivard
Institution(s): Center for Building Studies, Department of Building, Civil and Environmental Engineering, Concordia University, 1455 Maisonneuve Blvd. West, Montréal, Canada H3G 1M8
E-mail(s): KeMen@illom.net, cbedard@vax2.concordia.ca and rivard@concordia.ca
Abstract: *Computers and software can be fashioned into multipurpose tools with interactive and efficient storage and retrieval processes that are useful to the early building designer. We seek to provide a tool for the early stages of building design, specifically architectural design. Earlier research and protocol analysis have shown the actions that occur on the design workspace concerning how the architect creates and manages early design data. Drawing specifications from this work, this paper presents functions and implementations that can support the early design process using computers.*
Keywords: *Architecture, computer-aided building design, conceptual design, protocol analysis, design research*

1. Introduction

Early design is a complex process characterised by vague issues and shifting goals[1] that is not easily captured by computer support. The ubiquity of computers however leads designers to strive to use them in all stages of the design process including early design[2]. Recently inquiry in the early design process has revealed key characteristics that represent what happens during early design such as transformations and duplications[3,4,5,6].

In addition to these "what" elements in the early design process, the research community is also interested in the "how" part of the early design process and this investigation is accomplished mainly through the use of protocol analysis[7,8]. Combining "what" and "how" knowledge, a set of specifications were derived to guide the conception and implementation of an early design computer tool[8]. This paper is a follow up on the development of specifications. A tool description is given that can help the designer produce and manage data that comes out of the early design process without disrupting their preferred method of design.

Section 2 presents a summary of the specifications we are trying to satisfy. The reader is referred to[8] for more information on the specifications and how they were derived. Section 3 presents our approach for supporting the early design process followed by how this is implemented for the capturing and representing of design items in section 4. Section 5 presents the details of how these captured items can provide interactive assistance and reports on consequences of designer's actions.

2. Specifications

The early design process can be divided into four activities: design problem definition, site planning/analysis, building space configuration and building element configuration. The activities are progressively refined into design solutions using events. An example of an event is the use of drawing sheets for organising design items.

Activities suggest design environments or modules while events suggest tools available in each environment or module. This project concentrates on the building space configuration activity as more time is spent in this activity in comparison to the others and it is also where the designer produces the solutions for the design problem. A subset of the specifications – element interaction, application of requirements, multiple levels of abstraction, version management, automatic feedback, and design overview – is considered, of which only two are discussed in this paper as follows (a detailed discussion can be found in [8]).

Element interaction: Designers interact with design items in abstract and detailed forms. This interaction is considered as completely as possible because early design is a dynamic and non-linear process that depends on the success of interaction to fuel the designer's creativity.

Application of requirements: Requirements and needs of the client, the designer and any laws affecting the site must apply in the design process. This is considered as completely as possible in terms of how it affects interactions in the system, however because requirements repository is not taken into account, the issue of how such requirements are entered into the system is not covered here.

3. Early Design Support: Approach

Design data are represented by means of entities, which are digital objects that can be aggregated to represent parts of a building at different levels of complexity. For detailed information about building entities, the reader is referred to[9].

Building Entity

Entities model buildings at different levels of complexity by aggregating attribute-value pairs to represent functional, design and evaluation units (see Figure 1). Functional units represent the intended purpose, constraints or requirements of the entity (e.g. First floor must provide garage, ample storage, home office and an independent entrance). Design units represent all physical and spatial characteristics that describe a solution (e.g. the building has more than one level, floor to ceiling height in level 01 is 2.4m, office space is in level 01 and is made up of a reception, study and a workshop). Evaluation units represent the behaviour of the entity and include the response to different conditions (e.g. estimated heat loss of 8.2kW at -23° C outdoor temperature). The combination of these units forms an entity.

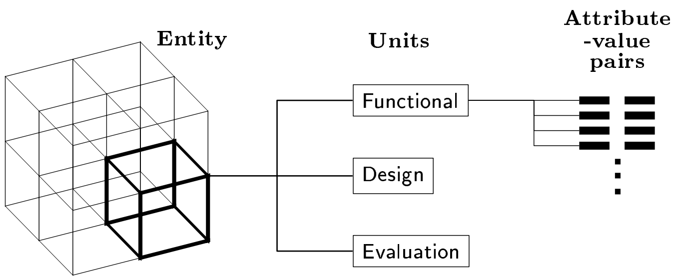

Figure 1: Attribute-value pairs in an entity

Hierarchical Decomposition

Groups of entities can be aggregated to represent a complete building hierarchy (see Figure 2). Note that parts of the building higher in the hierarchy are more abstract than those in the lower positions. The higher elements usually refer to spaces while the lower elements refer to components. Such hierarchies represent the normal working process of designers where abstract spaces get successively refined in iterations towards more concrete design objects.

Solution Path

A Solution Path is a collection of all the decisions that lead to a solution. A typical design session can contain many solution paths. Critical events that guide the formation of solutions paths are

Transformations (Vertical and Lateral) and Duplications. A *vertical transformation* indicates the elaboration of a design idea. A *lateral transformation* indicates the introduction of a different approach or version of design idea. *Duplication* indicates reuse of a ready-made idea or an object from an external project/design or a manufacturer's brochure.

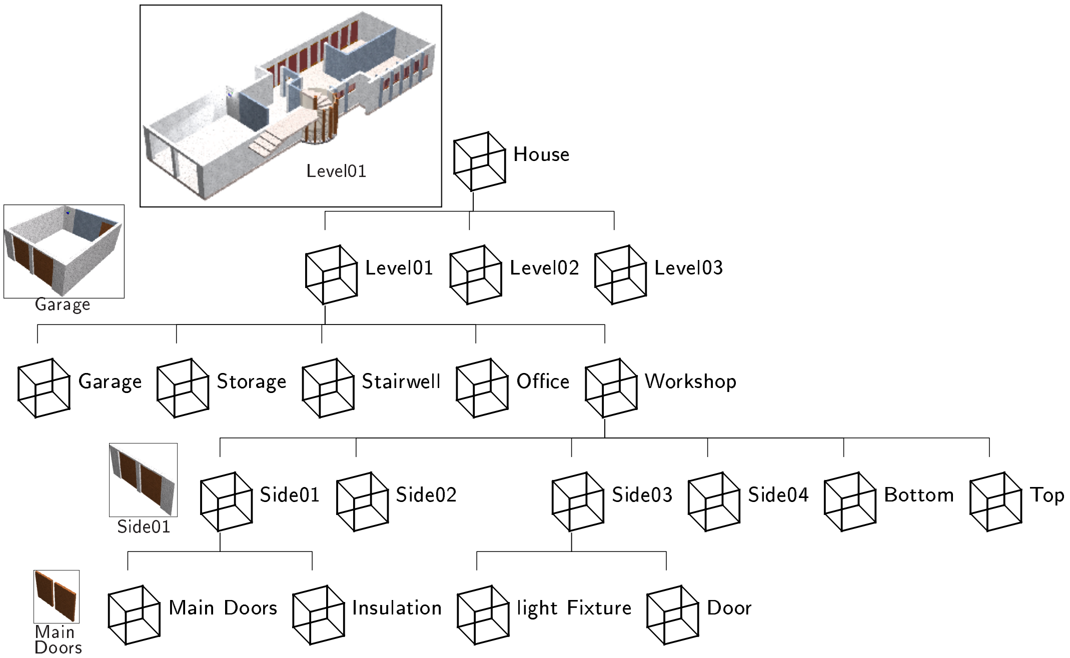

Figure 2: Hierarchical decomposition of a house design

Figure 3 illustrates two solution paths. The first path (Path01) begins at some point in the configuration of a house.

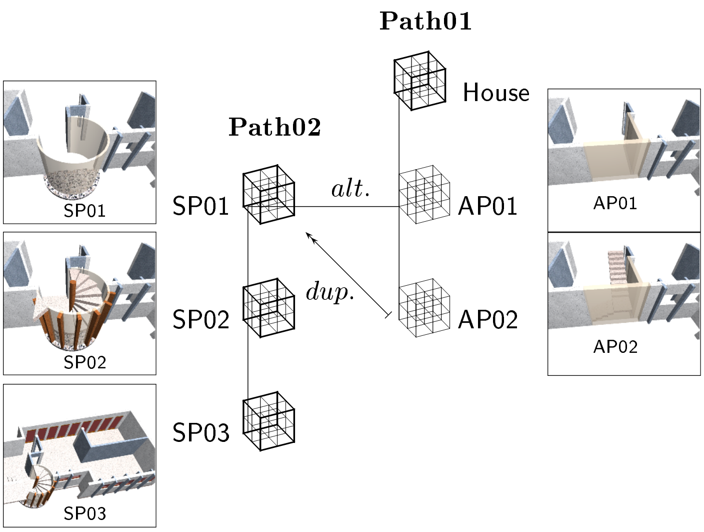

Figure 3: Solution paths

A vertical transformation is illustrated as the designer develops the stairwell and staircase using straight flights at AP01 and AP02. The designer changes and reconfigures the stairwell using a circular form, which is captured in a lateral transformation that creates a design alternative SP01 from AP01. Designer duplicates parts of AP02 to complete alternative solution at SP01. The branch leading to AP02 is then abandoned while a new set of vertical transformations is used to develop alternative solution at SP01 into the solution at SP03. The steps from “House” to “SP03” show the solution path that leads to the design solution accepted by the designer.

Using entities as digital building blocks, the designer's work is captured and organised in hierarchies and solution paths. Hierarchies provide access to each part of a solution while solution paths capture the creation of multiple solutions that can be merged later to create the best possible solution.

4. Early Design Support: Implementation

Every item drawn is accepted as an entity. Entities, when initially created, have a 3D size and location. Observing architects[8], it is noted that marks and shapes are made on paper that does not have exact meaning. However when they mean something, a label is often added. This style is adopted to provide knowledge for interaction in the computer environment.

Draw Support

The designer sets the base point and the height of every item drawn by using a slider that is always available on the screen. The designer is also allowed to create virtual drawing surfaces or sheets for grouping design items, as they would normally do using physical drawing sheets. All drawings and manipulations take place in 2D views while data recorded and visualisations are carried out in 3D.

While drawing, a continuous arm movement is expected for each entity. The drawn entity is captured as shown in Figure 4. The solid outline indicates the designer’s actions while the shaded forms with dotted outline represent the interpretation of these actions by the system (note that the exaggerated distance between the solid line and the boundaries of the shaded figure is for the benefit of the illustration). The interpretation is necessary for the system to capture line drawings as entities with interior and exterior areas.

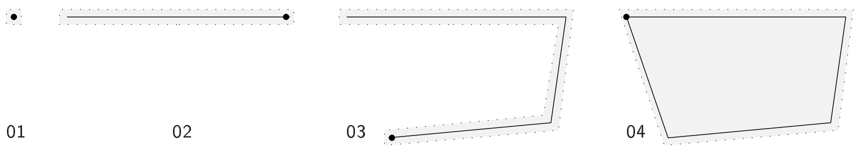

Figure 4: Typical entity drawing steps

The rest of this section illustrates a sample design session showing how the system captures and represents early design data.

Capture and Represent Support

The designer interacts with two windows. The Draw window is used for drawing entities and the Classify window for the designer to identify each drawn entity. This section shows how the system captures and recognises drawn entities. Section 5 will describe how the system enables each entity to be “aware” so as to provide needed interactivity and assistance.

Capturing Data - Step 01: Assume the design of a residential house. At the beginning (see Draw window in Figure 5), the system presents spaces from the requirement repository in the Classify window.

Representing Data - Step 01: The system depicts the state of the design as a 3D box “House” in Figure 5. In the following illustrations, data representations are shown as hierarchically arranged 3D boxes.

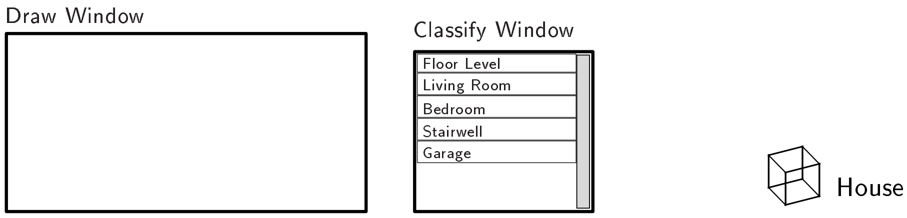

Figure 5: Capture and represent - 01

Capturing Data - Step 02: The designer draws a shape in the Draw window as shown in Figure 6, then selects from the Classify window to tell the system that it is a “Living Room”. Once a choice is made in the Classify window, the options change to reflect entities that could be located within the newly classified space. Thus the options in Classify Window change to reflect entities available in a standard living room as shown in Figure 7.

Representing Data - Step 02: Step 02 is captured as shown in Figure 6. It now shows that “House” contains a space called “Living Room”. Although manipulations are allowed in 2D views only, it is important to note that data captured by the system includes 3D information.

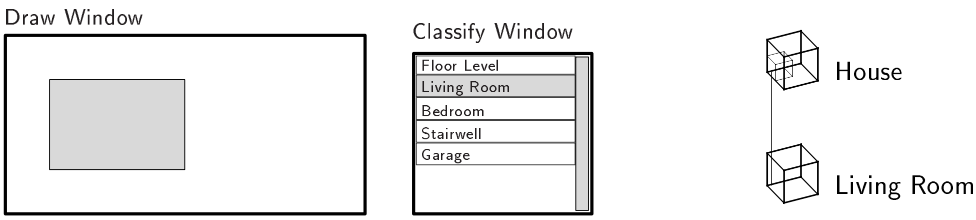

Figure 6: Capture and represent - 02

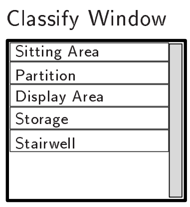

Figure 7: Classification changes

Capturing Data - Step 03: The designer draws a second shape (see dark shade in Figure 8). As this drawing takes place on an area not occupied by any previous space, the Classify window makes available entities similar to those in the Classify window in Figure 5.

Representing Data - Step 03: Data captured in step three is represented as shown in Figure 8 hierarchy.

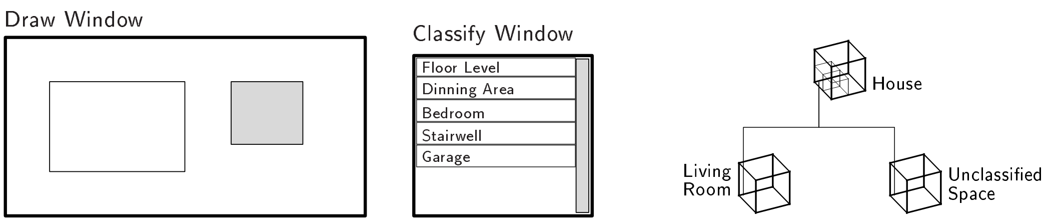

Figure 8: Capture and represent - 03

Capturing Data - Step 04: The designer does not classify the new space. Instead another space is drawn within the living room space as shown in Figure 9 prompting the Classify window to offer new options from where the designer selects to classify the new space as “Sitting Area”.

Representing Data - Step 04: The Classify window updates its contents to that shown in Figure 10 and the design is captured as shown in the Figure 9 hierarchy.

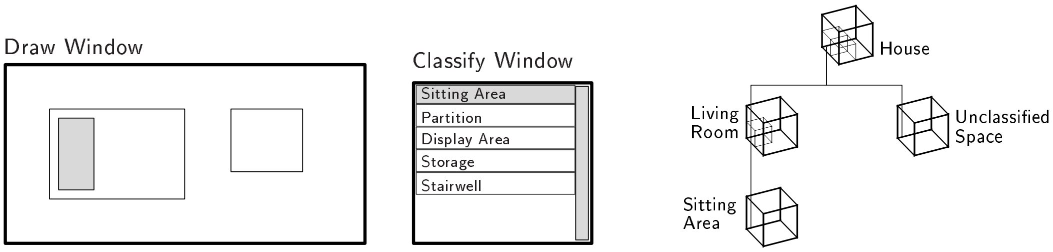

Figure 9: Capture and represent - 04

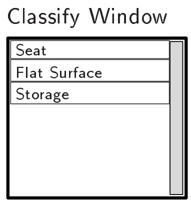

Figure 10: Classification changes

Capturing Data - Step 05: The designer draws a space that encompasses all spaces in the design (see Figure 11). The Classify window updates its options and the designer selects to classify it as “Floor Level”. The system represents step 05 as shown in Figure 11 hierarchy. If the designer modifies step 05 (see Figure 12), the system captures this in a lateral solution path (see Figure 12 solution hierarchy). Modifications or changes to decisions are represented using solution paths so designer can backtrack or select different parts of different solutions to merge into one.

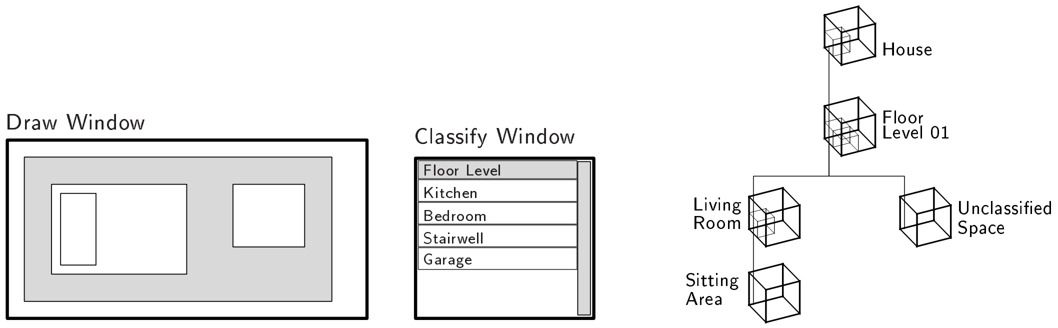

Figure 11: Capture and represent - 05

Each item in Figure 12 solution hierarchy embodies one step of the design process as illustrated so far. Note that access to individual components of a solution is possible in a hierarchical decomposition as shown in Figures 5, 6, 8, 9 and 11 hierarchies while ability to go back to a prior solution state is possible in solution paths as shown in Figure 12 solution hierarchy.

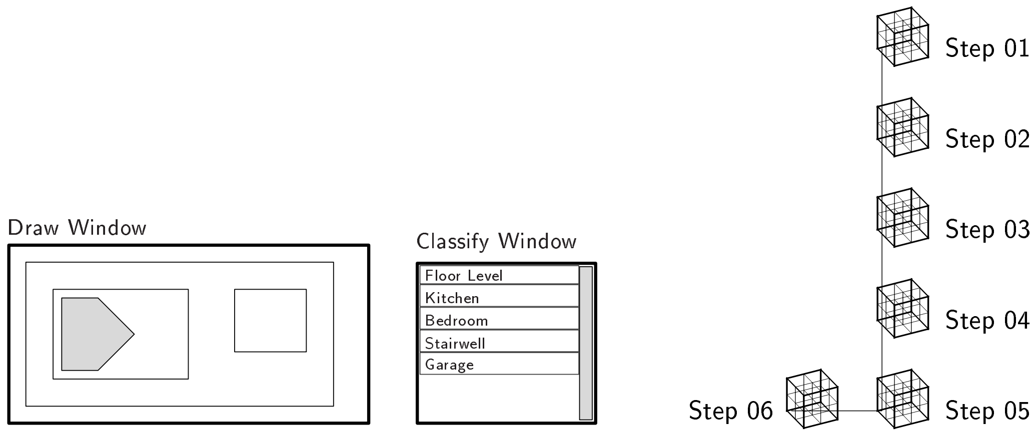

Figure 12: Capture - 06 and solution hierarchy

5. Enabling Captured Entities

Captured entities are enabled with knowledge in the form of rules that make it possible for them to behave and interact appropriately with the designer as well as other entities in the design. Rules represent common-sense knowledge that assist in solving problems by specifying a set of actions to be performed for a given situation. Rules concerning standard behaviour in an entity are provided but can be modified by the designer if desired.

Other rules (usually unique to the project) such as managing the line of sight into the bathroom or maintaining a certain distance between two spaces can be defined before or during the design process. In the case of the bathroom, the entity ‘to remain unseen’ will monitor its position in relation to the door.

The setting of rules is accomplished with the use of keywords. Assuming a designer creating a bedroom decides to maintain a minimum distance with the workroom to limit noise. The entity “workroom” is selected and the rule “reject” is applied (by selecting it). It is possible to include a distance but without one the system would provide a reminder at a later time. Entities being created or modified in the design always provide access to their rules for possible editing.

6. Conclusions

Computers perform a vital role in the design process but most of the usefulness is currently available in the later stages only. This is a major issue because vital decisions are made in the early stages when computers could potentially make a major impact if they were available for use. We have adopted a number of specifications formulated from observations and interviews made with designers at work during the early stages of the architectural design process[8].

First, this project considered an information model that can capture design information and support the type of interaction and visualisation required by the process. Such a model is based on building entities. By aggregating digital information called attribute-value pairs, units of data are assembled that completely describe every part of the building. Using the entity as a digital building block the design is captured, organised and represented in hierarchies and solution paths. Hierarchical decompositions provide access to any level in a design solution. Solution paths provide access to any previous point in the evolution of a design idea and also allow the designer to save multiple solutions of a design problem.

Finally, interactions between entities and the designer are enabled by knowledge about their role in the design. This does not only stimulate the designer's imagination but it encourages experimentation using ‘what if...’ scenarios and allows the designer to set limitations and dependencies which are crucial as design projects increase in the number of participants and complexity.

References

1. **Purcell A T and Gero J S.** Drawings and the design process. *Design Studies,* Vol. 19. No. 4 (1998), pp. 389—430.

2. **Rivard H.** A Survey on the Impact of Information Technology on the Canadian Architecture, Engineering, and Construction Industry. *Electronic Journal of Information Technology in Construction,* Vol. 5 (May 2000), pp. 37--56. (http://itcon.org/2000/3/).

3. **McGown A, Green G and Rodgers P A.** Visible Ideas: Information patterns of conceptual sketch activity. *Design Studies,* Vol. 19. No. 4 (1998), pp. 431—453.

4. **Goel V.** Sketches of thought. *The MIT Press, Cambridge, MA* (1995).

5. **Verstijnen I M, Hennessey J M, van Leeuwen C and Hamel R.** Sketching and creative discovery. *Design Studies,* Vol. 19. No. 4 (1998), pp. 519—546

6. **Rodgers P A, Green G and McGown A.** Using Concept Sketches to Track Design Progress. *Design Studies*, Vol. 21 No. 5 (2000), pp. 451—464.

7. **Cross N, Christiaans H and Dorst K.** Introduction: The Delft Protocols Workshop. *Analysing Design Activity,* John Wiley & Sons. Nigel Cross, Henri Christiaans and Kees Dorst (eds). (1996), pp. 1—16.

8. **Meniru K, Rivard H and Bédard C.** Specifications for Computer-Aided Conceptual Building Design. *Design Studies,* (2002) in press.

9. **Rivard H and Fenves S J.** A Representation for Conceptual Design of Buildings. *Journal of Computing in Civil Engineering,* Vol. 14 No. 3 (2000), pp. 151—159.