Early Building Design: Capturing Decisions for Better Interoperability

K. Meniru
Concordia University, Montréal, Canada

C. Bédard & H. Rivard
ETS – École de Technologie Supérieure, Montréal, Canada

ABSTRACT: Interoperability depends largely on accurate interpretation of data from one form to another. Current efforts (e.g., the IFCs) provide ways to do this, however such solutions are not available when work is accomplished manually. This is unfortunately how work is done today in the most important stage of building design – early building design. The translations that occur when transferring concepts to later stages or when referring to work at this stage entails a high potential for errors that increase cost and delay work. This paper investigates the work process in this early stage of building design and presents requirements for supporting it, a design that satisfies these requirements and a prototype to illustrate an implementation of the design.

1 INTRODUCTION

The building delivery process consists of feasibility, design and construction stages. The scope of work in the design stage is further divided into conceptual (schematic), preliminary (development) and detailed (construction) stages.

Computers have become commonplace in the building design and construction process due to their ability for interaction, recording and recalling data (Rivard et al. 2000b). A recent study of activities and systems associated with managing and exchange of electronic and paper-based data in the building industry reveal inefficient interoperability issues that costs the industry $15.8 billion annually (Gallaher et al. 2004). Most interoperable costs for architects and engineers occur during the design phase, especially the manual re-entry of data that costs 17 times more in the design phase than in the construction phase.

A major part of this excess cost is due to the lack of adequate computer support for the conceptual design stage. The decisions made in this stage are manual-based and work has only recently begun towards understanding it and finding ways to capture it electronically (Cross et al. 1996, Meniru et al. 2003).

This paper discusses how decisions in the early design stage can be captured and shared between all building design teams using a neutral file format like the IFC. The following section presents a summary of issues in the early design process. A section follows with requirements needed for adequate support. The rest of the paper presents a design for supporting these requirements and an implementation for illustration.

2 EARLY BUILDING DESIGN

Building design is complex and its success is most likely when different professionals can collaborate, especially in the early stages (Howard et al. 1989, Bedard et al. 1991). During the resolution of problems encountered in the early design process, the designer does not necessarily complete each issue before moving onto the next. Often, the designer pursues opportunities in the design and may leave incomplete efforts to return at a later time (Goldschmidt 1991, Purcell & Gero 1998, Heylighen et al. 1999).

There are intermittent periods of assessment and reference to the collected requirements when the designer checks the design process against initial established needs (Fricke 1999). Labels are used to substantiate the drawing as reminders or for capturing the intended use of a space. Designers may use stencils or predetermined spatial dimensions but the dependence on precise measurements is typically avoided. Transparent sheets are used in manipulating drawings as these reduce the effort necessary by making it possible to copy from already drawn items. Transparent sheets also support the creation of design alternatives which assist the designer in ultimately producing the perceived best possible solution for the design problem (Atman et al. 1999, Casakin & Goldschmidt 1999, Dörner 1999).

Designer's sketches undergo duplications and transformations (Goel 1995, Mcgown et al. 1998, Verstijnen et al. 1998, Rodgers et al. 2000). During duplication efforts the designer reproduces ideas or sketches already existing, such as a staircase. There are two types of transformations – vertical and lateral. During vertical transformations, the designer successively details an idea, like adding different spaces to a design such as a kitchen then a bedroom. During lateral transformations, the designer investigates alternate ideas of the same space (e.g. two or more kitchen designs for the same building problem) (see section 3.4 Figure 1).

The next section presents requirements that must be considered in supporting this activity.

3 REQUIREMENTS FOR SUPPORT

3.1 Introduction

Support for early design must cover the essential parts of the process that lead to successful solutions such as the use of vague drawings that stimulate imagination for the creation of alternate solutions. There are five main requirements: appropriate interface between designer and design; proper recognition of items created; collection of alternate solutions; organization of items created and making engineering knowledge transparently available.

3.2 Interface Issues

Drawing is a main part of the early design process. The use of simple graphical forms simplifies the drawing process and leaves the designer's mind free to concentrate on design. This simplicity makes it possible to discard drawings when necessary to explore more than one solution. Support must be provided in a way that encourages the use of simple forms and also minimize the effort in manipulating them.

3.3 Design Item Recognition

The result of using simple forms in drawings is vagueness due to very little definite information. This leads to inconsistent interpretations that may conflict and jeopardize the success of the solution (Fazio 1990). Designers use labels and other text to add more information, however there are design issues that cannot be captured by text and labels are not provided all the time. Some means of consistently recording or capturing design intent is required.

3.4 Collection of Alternatives

Creation of alternatives enhances the possibility for more informed decisions and a successful design. Figure 1 shows two alternatives, solution paths 01 and 02. A solution path is the sequence of recognized design decisions taken in the resolution of a design problem such as shown in “AP01” and “AP02” in Figure 1 (addition of vertical circulation). The second solution path is initiated by the creation of a new version of “AP01” in “SP01” (creation of alternate vertical circulation). In addition, information such as number of steps and riser height is also duplicated from “AP02”. The two solution paths record two solutions for the design problem, “House”.

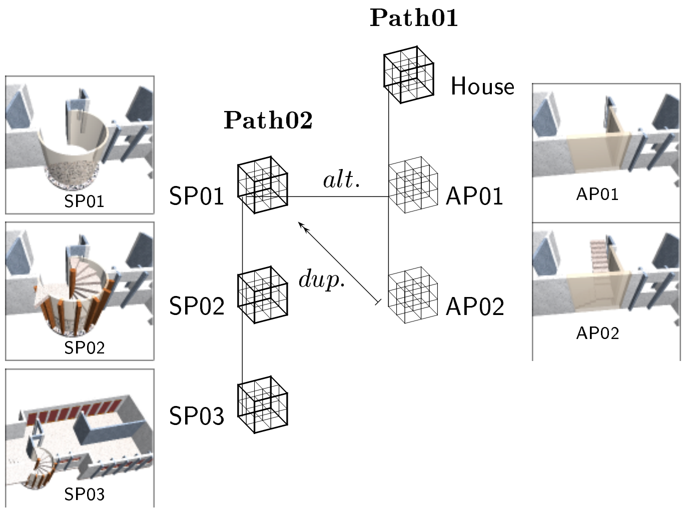

Figure 1. Solution paths

3.5 Design Decomposition

Building parts have a hierarchical relationship in which spaces can be decomposed into their corresponding parts. This forms a natural representation of the connections between them, which provide an intuitive means by which building data can be organized and presented (Rivard & Fenves 2000a). This can be done in many levels of detail. Figure 2 shows the relationship between building levels and the building. Figure 3 shows relationship between spaces and a level.

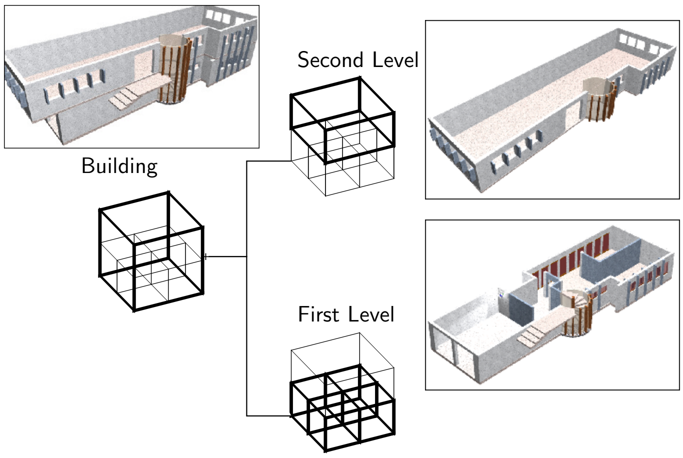

Figure 2. Decomposition of a building

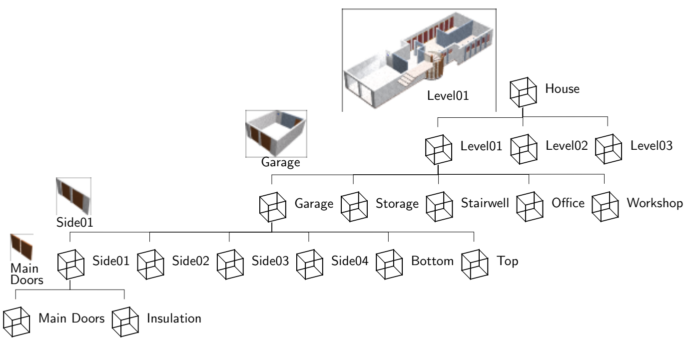

Figure 3. Decomposition of a floor

This provides the designer an efficient overview of relationships between parts of the building. It also provides direct access with minimum search effort.

3.6 Knowledge Integration

The decisions made in the early design process provide a framework upon which all other decisions are built (Cross et al. 1996). The complexity and the personal nature of early design decisions make appropriate collaboration difficult. Designers often revisit decisions made because adequate information was not easily available at the time or was outside the scope of their expertise. Rule-of-thumb knowledge should be available in early design to reduce inappropriate decisions and/or costly amendments.

The next section proposes means to satisfy these requirements.

4 OVERVIEW OF SUPPORT

The concept for supporting early building design in computers is based on the creation and manipulation of two digital objects called Corporeal and Incorporeal design items. Corporeal items denote space usage or space occupied such as a bedroom or a structural column. Incorporeal items denote all other design items such as orientation or direction. When created these items are captured in a way that shows their relationship to all other design items, but to be a part of the solution, first they must be recognized. This recognition process is provided by the Classify function. Once recognized, a design item becomes a part of the design solution and is then collected as a step in the progress towards a solution. Further customization that may make a space more suitable for a particular client, site or budget can then be done through the application of specific knowledge to control behavior or state.

Figure 4 shows an overview of the objects and functions that will provide the required functionality. The draw function makes it possible to create Corporeal and Incorporeal objects. The Classify function makes their classification possible and the designer can also apply knowledge to further configure them. The objects are then captured in a hierarchical view, which occurs automatically, when an object is created. However, they are added to the solution path only when they are classified and can play a specific role in the design.

The next section presents the design of these functions in an attempt to illustrate how to adequately support the process of early design.

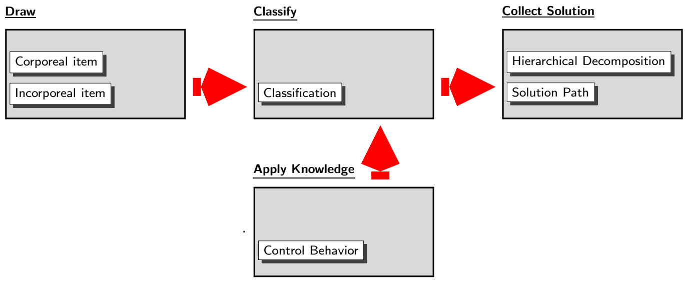

Figure 4. Overview of support functions

5 DESIGNING FOR SUPPORT

5.1 Introduction

The illustrations in this section present a visual design session being conducted in a digital environment. The diagrams showing Draw [DW] and Classify [CW] represent windows in which the designer interacts with the support system. The diagrams showing Hierarchy (decomposition) [HW] and (Solution) Path [PW] represent windows that automatically update to reflect the support tools' reactions to the designer's actions.

5.2 Step 01

(Figure 5). At the beginning of the session, a blank screen is presented in DW with a list of names for spaces that can be created in CW.

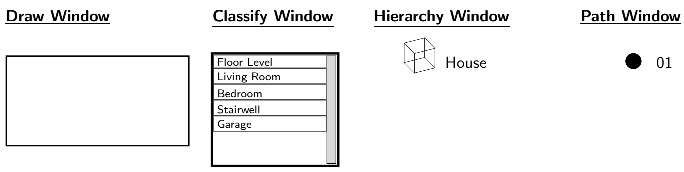

Figure 5. Step 01

The HW depicts the current state of the design session, which is an empty space for solving a "House" design problem. The state of the current solution is captured as a progression of steps in PW, shown here as a first dot.

5.3 Step 02

(Figure 6). The designer draws a shape in the DW at (A) and selects the item "Living room" for its classification at (B). The system automatically adds this classified item to the hierarchy tree as shown in the HW at (C) and this progress in the solution of the design problem is shown by the addition of a new step in the PW at (D).

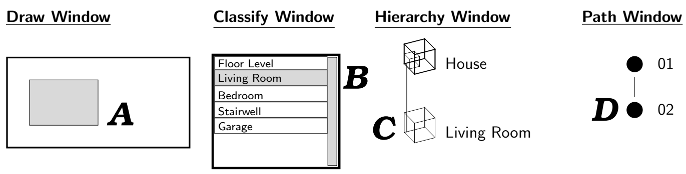

Figure 6. Step 02

5.4 Step 03

(Figure 7). The designer draws a second shape in DW at (A). Since this new shape is not contained in any other space, the space names offered in CW are similar to those in Figure 6. This new space is added to the hierarchy window as shown in (C) but there is no change in the PW because this new item is not classified and does not provide any obvious improvement or additional feature to the current solution. Designers often draw shapes that may not be used in the final design solution.

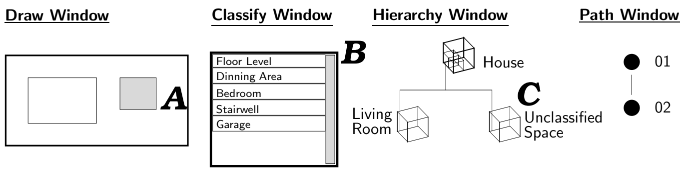

Figure 7. Step 03

5.5 Step 04

(Figure 8). A new space is drawn inside the living room space in DW at (A). The system presents space names in CW. These different choices of names are provided based on the context of the Living room space. The designer makes a choice at (B) causing the system to add the newly classified space in the HW at (C) within the living room space. Finally this progress in the design session is captured in PW at (D).

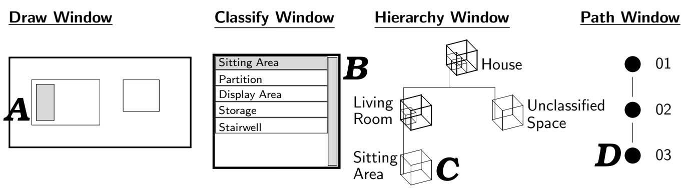

Figure 8. Step 04

5.6 Step 05

(Figure 9). The designer draws a space that encompasses some spaces in DW at (A) and classifies this new space in CW at (B). The system accepts this classification in HW at a higher level (C) than the existing rooms because it is a floor and it encompasses them. The solution path information is then updated in PW at (D).

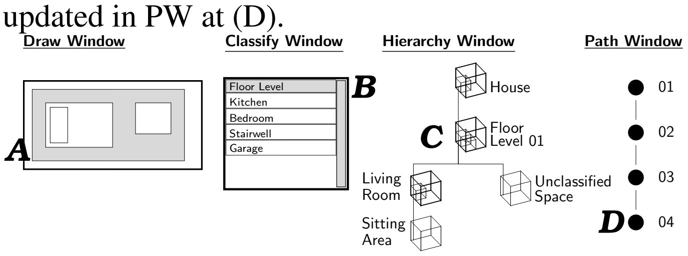

Figure 9. Step 05

5.7 Step 06

(Figure 10). The designer modifies the sitting area in DW at (A) by introducing an alternate version of the solution created in Step 04. No item is classified in this step and the data in HW does not change, however the alternate solution is automatically recorded by the system in PW at (B). The first solution explored is recorded in PW by the path 01–04 while the path 01–03 and 05 records the alternate solution.

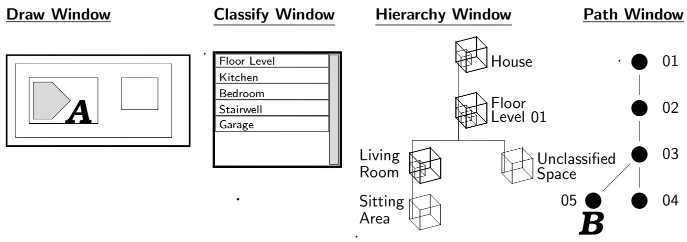

Figure 10. Step 06

6 PROTOTYPE

6.1 Main Window

The prototype developed is called CoBL DT. It presents a main window that is divided into areas in which different activities take place as shown in Figure 11 and as explained below.

The top area presents pull-down menus and toolbars of commands that the designer can use to control the creation and manipulation of design items as shown at A. It provides a location for most of the commands used in CoBL DT.

The second area presents a drawing window shown at B where the designer creates and manipulates the design items. These items are captured automatically by the system and presented in the organization area where the designer can either review or rearrange them with regard to their relationship to each other.

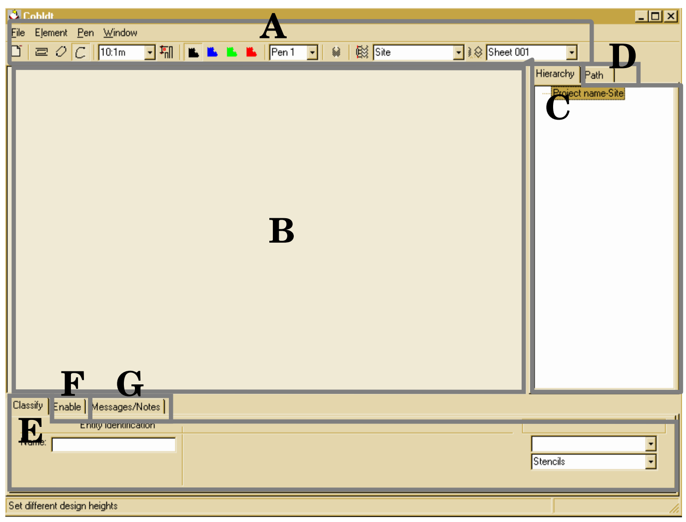

Figure 11. CoBL DT main window.

The third area presents two windows for capturing design items in hierarchies at (C) or capturing design solution paths at (D).

The fourth area presents three windows. The first two provide the designer with an interface for classifying and configuring design items at (E) and (F). The third window at (G) is used by CoBL DT to provide text-based feedback to the designer about internal operations or consequences of certain interactions. The second region allows the designer to make personal notes.

6.2 *Drawing*

The prototype allows the designer to create generic spaces as Corporeal design items. Generic items are not assigned any role in the design and are drawn with dashed outlines as shown in Figure 12 at (B1). At this time, only the size and orientation is known as shown at (E1). Additional basic function or characteristics can be provided by selecting an option at (E2) to classify the generic Corporeal item.

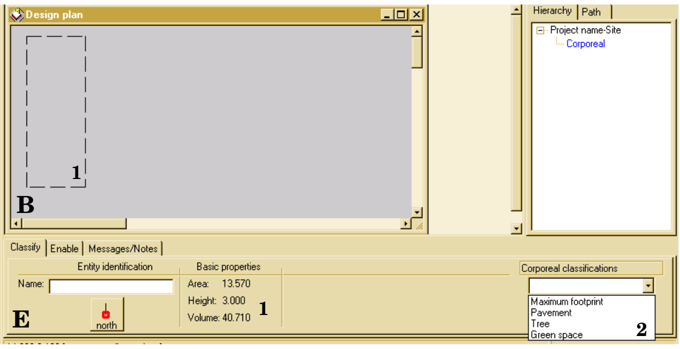

Figure 12. Creating a corporeal design item

6.3 *Classifying*

Classification is akin to labeling sketches in the traditional design process. When classified, the Corporeal item inherits all the basic characteristics and knowledge for the classification as shown in Figure 13 at (B1). The item is then rendered in solid lines and becomes aware of its basic function in the design. Additional options are provided for further configuration and customization at (E1)–(E5) providing one way of applying knowledge in the design.

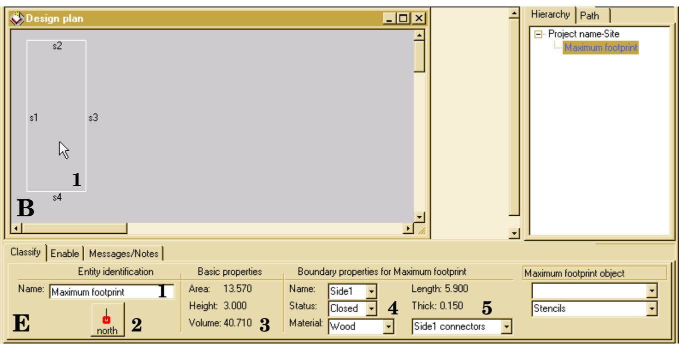

Figure 13. Classified corporeal design item

6.4 *Organizing*

The prototype automatically assists the designer by organizing design items using hierarchical decomposition as shown in Figure 14 at (C). The relationship between all design items is shown and provides an easy access for the designer. Compare with view (B) where items in the current sheet only can be accessed.

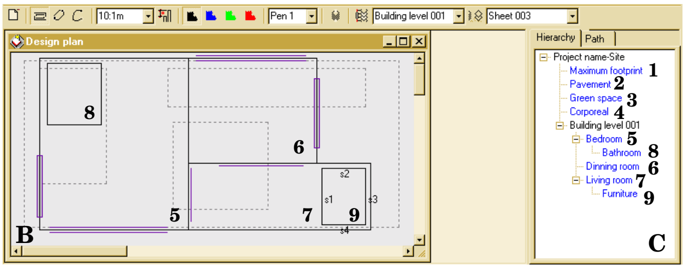

Figure 14. Hierarchical decomposition

CoBL DT can also provide automatic organization by collecting each recognized design item as a step towards a solution (solution path). Figure 15 and Figure 16 show how a new path in the solution is recorded. Solution 01 (Figure 15, Step[1a]–Step[4a]) is automatically captured by CoBL DT. The designer alters this solution at (Step[3a]) to create (Step[3c]). CoBL DT creates solution 02 (Figure 16, Step[1c]–Step[3c]) by duplicating Step[1a] in Step[1c] and Step[2a] in Step[2c]. This is crucial if the designer decides to alter Step[1c], for example. Then it is still possible to go back to solution 01 if desired.

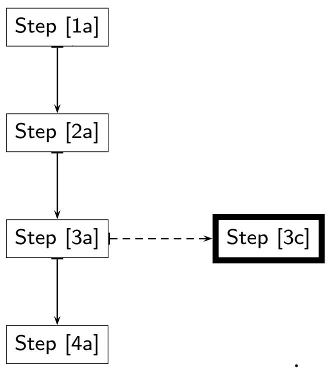

Figure 15. Branch-off point.

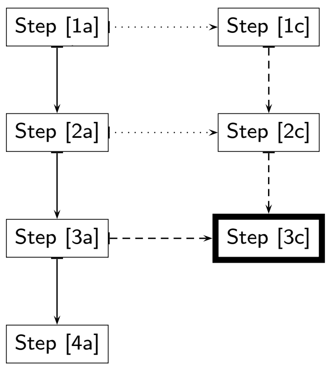

Figure 16. Replication

6.5 *Application of Knowledge*

Doors and windows are called connectors and are used to establish relationships between spaces as shown in Figure 17 on sides s1 and s2. Connectors are classified in the same way as spaces. They can also be adjusted or removed.

Certain checks can be made as design items are created. For example a minimum size for a space can be provided in which case CoBL DT watches to make sure that this is satisfied. It assists the designer to adhere to requirements. When a violation is detected, a feedback is provided showing approximate acceptable size and the creation or manipulation of the item is rejected.

Design aids are also provided such as the use of stencils. Stencils are predetermined sizes for use in design such as automobile sizes for configuring garages or parking spaces.

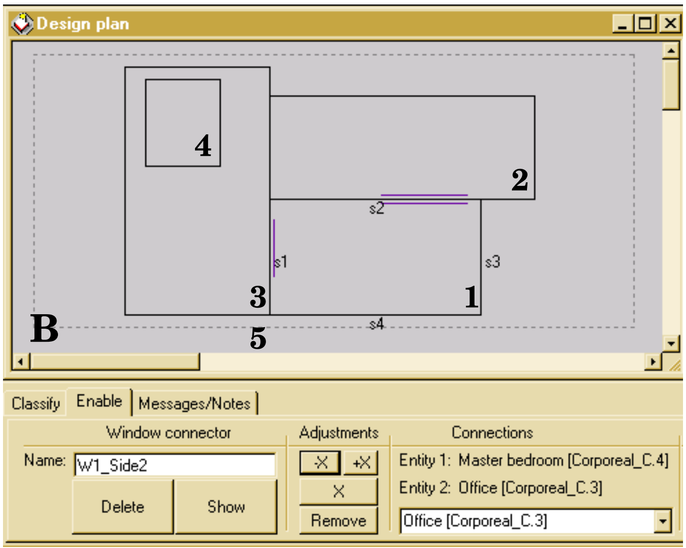

Figure 17. Using connectors

Complex calculations, especially those required to satisfy engineering requirements, can be provided. Figure 18 shows the result of automatic calculations for savings factor in space heating expenses.

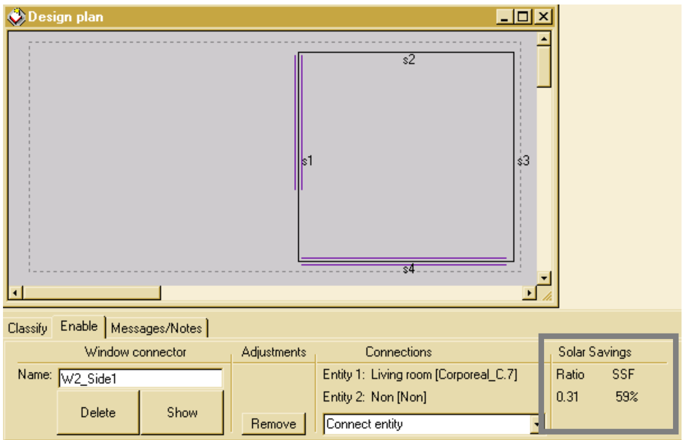

Figure 18. Calculation of solar savings factor

7 SUMMARY

The evidence for more efficient interoperability is compelling and there are new and ongoing efforts to create better formats that existing tools can utilize to transfer data and decisions. These are important efforts however the conceptual stage of design, that provides the fundamental issues which are used by every member of the building team and which relies on the interpretation of decisions made, cannot be supported by these efforts. The reason is that it is not, currently, a digital process.

This paper begins with a description of the early building design process and the requirements for its support. The concept for support in a digital environment is then presented followed by the design for supporting interactions in a digital environment. This includes the development of an appropriate interface for capturing the necessary parts of such a vague process. Finally the main components of a prototype that demonstrates appropriate support based on the concept and design as discussed, is presented. This prototype fulfills the need for interoperability with the ability to capture and transfer early design decisions using the IFCs.

REFERENCES

Atman C.J., Chimka J.R., Bursic K.M. & Nachtmann H.L. 1999, A Comparison of Freshman and Senior Engineering Processes. In *Design Studies, vol. 20* (2): 131–152.

Bédard C., Gowri K. & Fazio P. 1991, Integration in the building industry. In *Journal of Computing in Civil Engineering, vol. 5* (5): 333–335.

Casakin H. & Goldschmidt G. 1999, Expertise and Use of Visual Anology: Implications for design education. In *Design Studies, vol. 20* (2): 153–175.

Cross N., Christiaans H. & Dorst K. 1996, Introduction: The Delft Protocols Workshop. In N. Cross, H. Christiaans and K. Dorst (eds.), *Analysing Design Activity*: 1–16. John Wiley & Sons

Dörner D. 1999, Approaching Design Thinking Research. In *Design Studies, vol. 20* (5): 407–415a.

Fazio P. 1990, A case study in integrated building design. In *Canadian Journal* for *Civil Engineering, vol. 17*: 636–642.

Fricke G. 1999, Successful Approaches in Dealing with Different Precise Design Problems. In *Design Studies, vol. 20 (5)*: 417–429.

Gallaher M.P. & O'Connor A.C. & Dettbarn J.L. & Gilday L.T. 2004, *Cost Analysis of Inadequate Interoperability in the U.S. Capital Facilities Industry*, chap. Six. NIST GCR 04-867 (U.S. Department of Commerce, Technology Administration, National Institute of Standards and Technology). 6:1–6:19.

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

Goldschmidt G. 1991, The Dialectics of Sketching. In *Design Studies, vol. 4*: 123–143.

Heylighen A., Neuckermans H. & Bouwen J.E. 1999, Walking a thin line — between passive knowledge and active Knowing of components and concepts in architectural design. In *Design Studies, vol. 20* (2): 211–235.

Howard H.C., Levitt R.E., Paulson B.C., Pohl J.G. and Tatum C.B. 1989, Computer Integration: Reducing Fragmentation in AEC Industry. In *Journal of Computing in Civil Engineering, vol. 3* (1): 18–21.

McGown A., Green G. & Rodgers P.A. 1998, Visible Ideas: Information patterns of conceptual sketch activity. In *Design Studies, vol. 19* (4): 431–453.

Meniru K., Rivard H. & Bédard C. 2003, Specifications for Computer-Aided Conceptual Building Design. In *Design Studies, vol. 24* (1): 51–71.

Purcell A.T. & Gero J.S. 1998, Drawings and the design process. In *Design* Studies, *vol. 19* (4): 389–430.

Rivard H. & Fenves S.J. 2000a, A Representation for Conceptual Design of Buildings. In *Journal of Computing in Civil Engineering, vol. 14* (3): 151–159.

Rivard H. & Fenves S.J. 2000b, SEED-Config: A Case-Based Reasoning System for Conceptual Building Design. In *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, vol. 14 (5): 415–430*

Rodgers P.A., Green G. & McGown A. 2000, Using Concept Sketches to Track Design Progress. In *Design Studies, vol. 21* (5): 451–464.

Verstijnen I.M., Hennessey J.M., van Leeuwen C. & Hamel R. 1998, Sketching and creative discovery. In *Design Studies, vol. 19* (4): 519–546.