Off The Page

Object-Oriented Representations

by Michael James Kilkelly

Bachelor of Architecture
Norwich University, 1995
Northfield, VT

Submitted to the Department of Architecture in Partial Fulfillment of the Requirements for the Degree of Master of Science in Architecture Studies at the Massachusetts Institute of Technology.
June 1999

....
Signature of Author

Michael James Kilkelly
Department of Architecture
May 20, 1999

....
Certified by

Takehiko Nagakura
Assistant Professor of Architecture
Thesis Supervisor

....
Accepted by

Roy Strickland
Associate Professor
of Architecture
Chairman, Department
Committee on Graduate
Students

Off The Page

Object-Oriented Representations

by Michael James Kilkelly

Submitted to the Department of Architecture in Partial Fulfillment of the Requirements for the Degree of Master of Science in Architecture Studies at the Massachusetts Institute of Technology.
June 1999

Abstract:

The intention of this thesis is to investigate methods in which deficiencies in construction documents can be addressed through the application of digital technology. These deficiencies take two forms. The first form relates specifically to the limitations of a paper based documentation system with regard to accessing information. This is becoming increasingly prevalent due to the increasing amount of documentation required for complex building projects. The second set of deficiencies is directly related to the time consuming nature of the construction document production process, recognizing that the majority of time is spent reformatting and redrawing previous details and specifications.

The concepts of object-oriented programming and levels of abstraction are used as organizational structures to address these deficiencies. While current methods of documentation utilize more traditionally based organizational system, this thesis explored the use of structures inherent to computational media. Additionally, database structures were explored as a key component to information reuse in the documentation process.

Two prototype systems are developed to propose alternative methods of documentation using computational media. The Building Interface is an interactive system for information access that utilizes varying densities of information and multiple modes of representation. The Drawing Assembler is a graphic search engine for construction details that links a building component database with a construction detail database through the intersection of dissimilar objects.

Thesis Advisor: Takehiko Nagakura
Assistant Professor of Architecture
Department of Architecture

Off The Page

Object-Oriented Representations

Readers

Michael C. Hoffman
Associate Professor of Architecture
Division Head, Division of
Architecture and Art
Norwich University

William J. Mitchell
Professor of Architecture and
Media Arts and Sciences
Dean, School of Architecture
and Planning, MIT

Thanks

Talia Braude
Michael Hoffman
Patrick Kilkelly
Andrea McCarty
William Mitchell
Takehiko Nagakura
Hector Perez
Eric Walter

Table of Contents

1. Introduction

> What can be done with traditional means, what can be represented with traditional models, does not necessarily become more interesting because it has been generated using a computer. What is partially interesting is what was not conceivable before computers, expression that takes advantage of unique new capabilities of the computer.
>
> - Stephen Holtzman
> *Digital Mantras*

Motivation

There is little doubt that the computer is continuing to have a profound effect on the practice of architecture. Simply walk into any architect's office today and you will see that the drafting tables and drawings tools, the long time symbols of architectural production, have been replaced by gray plastic boxes and their various accessories. The drafting pencil has been replaced by the computer mouse, the flat file by the file server. What is increasingly interesting is that despite all of these fundamental changes to the more readily apparent aspects of the profession, the computer has done little to transform the actual working methods of the practicing architect. While a majority of architects now draw with a mouse rather than a pencil or pen, the process used to create and develop the tangible products of the architect, the construction documents, has largely remained unchanged. The use of the computer and CAD software has certainly improved the speed and accuracy of this process, however, it is still a digital version of mechanical drafting. The tools have changed while the process has stayed the same.

Thesis Overview

The intention of this thesis is to investigate methods in which deficiencies in construction documents can be addressed through the application of digital technology. In the scope of this research, I have focused on two primary types of deficiencies. The first type deal specifically with the limitations of a paper based documentation system. This is becoming increasingly prevalent due to the sheer amount of documentation required for complex building projects. The second set of deficiencies are directly related to the time consuming nature of the construction document production process. The production of such documents often amounts for a majority of the allocation of both time and resources for a given project. However, the majority of time is spent reformatting and redrawing previous details and specifications.

The concepts of object-oriented programming and levels of abstraction were used as organizational structures to address these deficiencies. While current methods of documentation utilize more traditionally based organizational system, this thesis explored the use of structures inherent to computational media. Additionally, database structures were explored as a key component to information reuse in the documentation process.

In order to test the applicability of the above concepts, two prototype systems were developed to propose alternative methods of documentation using computational media. The Building Interface is an interactive system for information access that utilizes varying densities of information and multiple modes of representation. The Drawing Assembler is a graphic search engine for construction details that links a building component database with a construction detail database through the intersection of dissimilar objects.

Guiding much of this research was the view that construction documentation is as much a process of designing and organizing the information necessary for the construction of a building as it is a process of designing the building itself. It is largely a problem of information management. These areas have been heavily researched in areas of the discipline of computer science. In order to address deficiencies inherent within architect's own working methods, it is necessary to view construction documents as forms of information and to develop strategies tools that can use this information in the most advantageous manner possible.

2. Background

The purpose of construction documents is not so much to provide the various views of a particular building but rather to assemble and the information required on the construction site. The drawings serve as a mechanism to communicate the intentions of the architect to those who will translate the intentions into physical form. As Edward Robbins states in *Why Architects Draw*, drawings serve as a means "to join the creations of the architectural imagination with the institutions of architecture's material production" (Robbins 1994, 8). Given the scale and complexity of many building projects, the task of properly preparing such documents is crucial in order to maintain within the boundaries of both budget and schedule.

This chapter investigates specifically the development of documentation systems in the architectural profession and how these systems are currently being implemented in practice. While this is not an exhaustive history of construction documents, it is intended to support a more defined analysis into deficiencies associated with traditional documentation systems, which is discussed in the final section of this chapter.

Origins of Documentation

The historical development of the drawing as a tool of the architect and builder is a highly complex issue. Historians continue to debate the actual time period in which the drawing was introduced into construction practices. Many, however, believe that it was during the Gothic period that the drawing was revealed to be a critical instrument for the execution of construction. David Turnbull, a historian of technology, argues that the size and complexity of the Gothic cathedrals necessitated the use of some form of communication device.

> The achievement on the order of complexity and structural innovation involved in the construction of the cathedrals required a high degree of precision in the production of the stones and larger numbers of workers as well as types of workers. These factors create organizational difficulties that turn crucially on a fundamental problem: communication. Knowledge and instructions had to move among many participants (Turnbull 1993, 320).

Turnbull goes on to indicate that from this need for communication came a tool specifically tailored for the task. He states that "the beginnings of the technology of representation that is involved in the modern system of architectural drawings may have come about in conjunction with the development of cathedral building" (Turnbull 1993, 321).

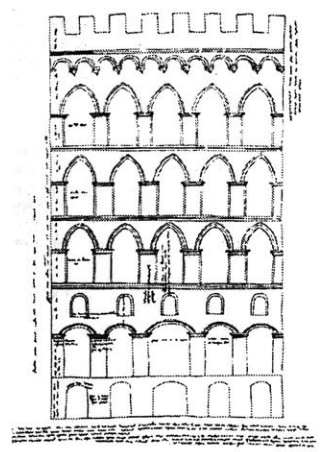


Fig. 1 The Sansedoni Elevation From Why Architects Draw

Concurrent with the evolution of the drawing as a communication tool was the development of architecture as a separate profession from construction. Historically, it was the master builder who oversaw the design and production aspects of a given building and directed construction on the site. The role was of the craftsperson, an individual simultaneously engaged in both the vision and the execution of the built form. In his book, *Why Architects Draw*, Edward Robbins indicates that it was the role of the drawing that brought about this shift. He argues that:

> This last transformation of the architect from craftsperson to artist was accompanied and, arguably made possible by the new centrality and importance of drawing as a critical instrument of architectural creation and production (Robbins 1994, 10).

What was clear was that the use of the drawing as a tool afforded many new possibilities to the process of construction (Fig. 1). Robbins indicates that as the shift from master builder to architect became more evident, the drawing was eventually recognized as a means to free the individual from direct supervision on the site and allowed him to work on several buildings simultaneously (Robbins 1994, 10). Coupled with this was the ability to experiment with more complex construction techniques through use of this system of representation rather than the trial and error techniques previously used (Turnbull 1993, 321). One final advantage to the use of the drawing had more to do with intellectual rather than practical concerns. Among many, there was a desire that the discipline of architecture be defined as the intellectual equal to both writing and mathematics. In order to do so, a formal language of representation was necessary (Robbins 1994, 17)

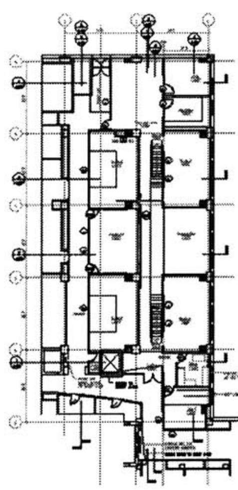


Fig. 2 Traditional Construction Drawing with Notation

Traditional Documentation Systems

Since this time, the construction drawing has evolved into a complex set of documents synthesizing the work of many disparate disciplines and professions. No longer simply a means to convey geometric information, construction documents have evolved to include the extents of both the legal and the contractual obligations of the building project as well. In *The Working Drawing Handbook*, Keith Styles states that construction drawings are used for a multitude of purposes. Initially, the drawings serve as a basis for the bidding process, illustrating the necessary materials and equipment as well as the trades involved. The drawings also indicate the extents of the contractual commitments and also as a statement of intent for the purposes of obtaining permits. Finally, the drawings serve as a record of the variations from the contract and become a base document for determining liability obligations (Styles 1995, 4). With this in mind, it is then the responsibility of the producers of the documentation, the architects and engineers, to ensure that the set of documents function properly within all these specified uses. Deficiencies in any of the above stated functions can cause significant problems on the site or in the courtroom.

From the perspective of the architect, there are several fundamental issues that must be addressed within the scope of construction documentation. As the overall designer of the built form, the architect must display a depth of knowledge in the various materials and processes that are to be employed during the course of construction. Similarly, the architect must be able to direct these processes through use of the prepared drawings. In order to accomplish these tasks in a proficient and suitable manner, the architect must display a mastery of the graphic methods necessary to achieve the intended design intent in a manner that can be readily understood by the end users of the drawings (Talty 1996, 81).

Additionally, in order for a set of documents to act as viable guides for construction, they must support the main activities of the builder or contractor. Through the information provided in the set of documents, the user must be able to procure the necessary materials, components, and equipment required by the extents of the construction. The user must also be able to determine the range and scope of the building trades involved. This includes the quality standards and procedures that are to be employed through the course of construction. Finally, from the documentation, the user must be able to develop a sufficient program of construction and reach confident decisions regarding the proper methods of operations (Styles 1995, 3).

In order act as a truly communicative device, it is critical that the prepared documentation facilitates rather than hinders the user's activities. As such, a user has the right to expect to expect that the information depicted within the set of documents is an accurate record of the design intentions and is presented in a manner that is clearly expressed and easily understood. This information must also be comprehensive and sufficiently detailed for the purpose of construction. Given the mass of information required for construction, it is also expected that the information will be readily available and capable of easy retrieval. While these criteria may seem fairly straightforward, deficiencies in the construction documentation can result in serious difficulties on the construction site (Crawshaw and Daltry 1973, 1-2)

Deficiencies of Construction Documents

Within the scope of this research, two types of construction document deficiencies were identified. The first set of deficiencies, which are illustrated in the following section, are related to the medium used. These deficiencies are due largely to the limitations of a paper based system of documentation. The second set of deficiencies, as indicated in the next chapter, are related to the process used in the production of construction documents. This addresses specifically the use of computer-aided drafting software as the main tool of document production.

Despite the obvious importance of clear and consistent communication between the designer and the builder, problems often arise through the course of construction. In their paper, "Working Drawings in Use", D. Crawshaw and C. Daltry investigate the effectiveness of construction drawings in providing the information needed on the site. Through a study of fifteen sets of documents produced by a variety of architecture firms, they conclude that in the majority of cases, the given documents are inadequate at providing the necessary information needed by the builders. Specifically, the report indicates five areas of deficiency within the drawing sets. These areas are uncoordinated drawings, incorrect information, failures in transmission, missing information, and confusing information (Crawshaw and Daltry 1973, 1-2).

If a set of construction drawings is viewed as an assembled collection of information, then locating specific information within the set is of crucial importance. Given the nature of the medium, linking various drawings or schedules is often done through the use of a variety of symbols. These are typically either implicit or explicit in nature. Explicit symbols, such as section or elevation markers provide direct access to the relevant pieces of information (Fig. 3). Provided that the notation on the symbol is correct, this method often requires little effort on the part of the users aside from shuffling through the sheet of drawings to locate the information they seek. Information that is not explicitly marked however, must be searched for through indirect means (Fig. 4). This method draws on the expertise and judgement of the users and introduces a great deal of uncertainty to the process. Indirect searches usually being by using three methods. One method is to search by drawing title, looking through each individual title for an indication of the information that it contains. Another method is to search by building element. Typically, similar elements are grouped closely to one another, such as casework or stairs. The final method is to search by drawing type. As with building elements, similar drawing types, such as wall sections, are often organized in a series of drawing sheets. Through the use of indirect search methods, there is no guarantee that the information that is retrieved will be correct of complete. This could lead to costly errors and delays on the construction site (Crawshaw and Daltry 1973, 9, 12)

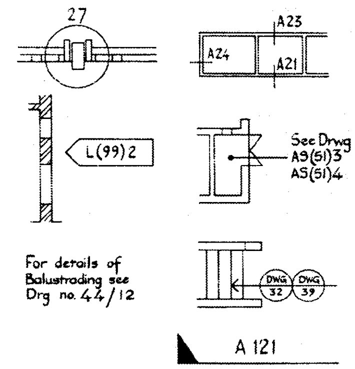


Figure 3. Direct search symbols (from Working Drawings in Use*)*

Understandably, the structure of a set of drawings greatly adds to its searchability. Systematically organized drawings, such as ConDoc in the United States and CI/SfB in Great Britain, typically specify particular arrangements for different types of information (Fig. 5). Additionally, location drawings, such as plans, elevations, and building sections are often used solely for purposes of referencing other, more detailed and specific drawings. In contrast, traditionally arranged set can have various drawings located on any one sheet. This is usually done on the basis of available space rather than any preconceived design. These drawings typically contain as much information as possible in an effort to conserve space. As a

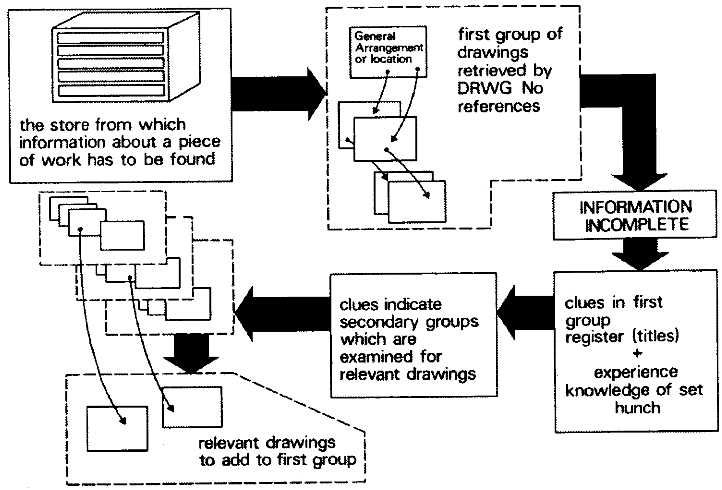


Figure 4. Indirect search pattern (from Working Drawings in Use*)*

result, traditional sets often contain denser levels of information making indirect search procedures more difficult (Crawshaw and Daltry 1973, 10, 16).

Crawshaw and Daltry also indicate that adequacy of information is another major problem with the current state of construction drawings. The information constrained within a set of documents is derived from a variety of sources including the architectural and engineering drawings as well as the bill of quantities and the specification. With specific regard to the architectural drawings, information can be provided in several forms depending on what is to be conveyed. Plan and elevation drawings are generally used to illustrate position and dimension in the horizontal and vertical plane respectively. Sectional drawings can show either horizontal or vertical extents are often used to coordinate the plan and elevation drawings as well as indicate the specific material requirements of the building (Talty 1996, 82). Three-dimensional drawings can also be employed to indicate both horizontal and vertical dimension (Fig. 6). Finally, schedules can be used as organizing and referencing devices for specific elements of the building such as doors or windows.

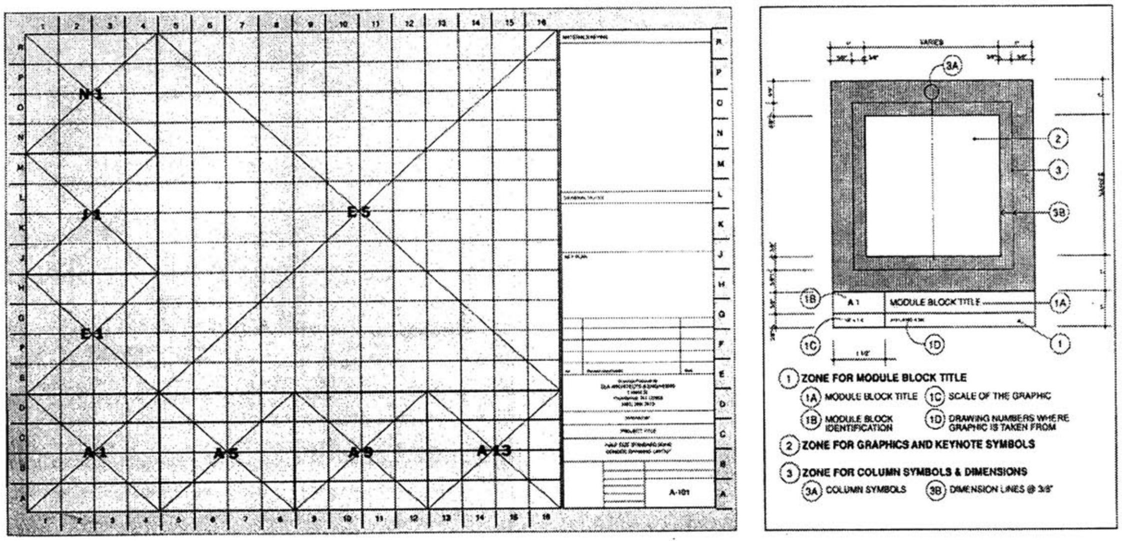


Figure 5. Example of a construction drawing using the ConDoc documentation system developed by the American Institute of Architects

Within these forms of drawings, there are generally three methods in which information can be communicated. First, the information can be explicitly given through the form of dimensions or written notes. This is certainly the most accurate method of transmission. Second, the information can be calculable, such as surface area or volume. This method relies on the abilities of the user to translate the given information into a usable form. However, provided that the indicated dimensions or notations are correct, a reasonable level of confidence can be assumed. Finally, information can be given implicitly, requiring an assumption on the part of the user. This is the least secure method of transmission, relying solely on the abilities and knowledge of the user (Crawshaw and Daltry 1973, 7).

Through the course of their research, Crawshaw and Daltry determine that both the accessibility and the adequacy of the given information causes the major deficiencies in the current system of construction documentation. This is true even in sets using a systematic organization system as well as those that were prepared with a great deal of care toward addressing these issues. While some of the fault can be extended to those prepar-

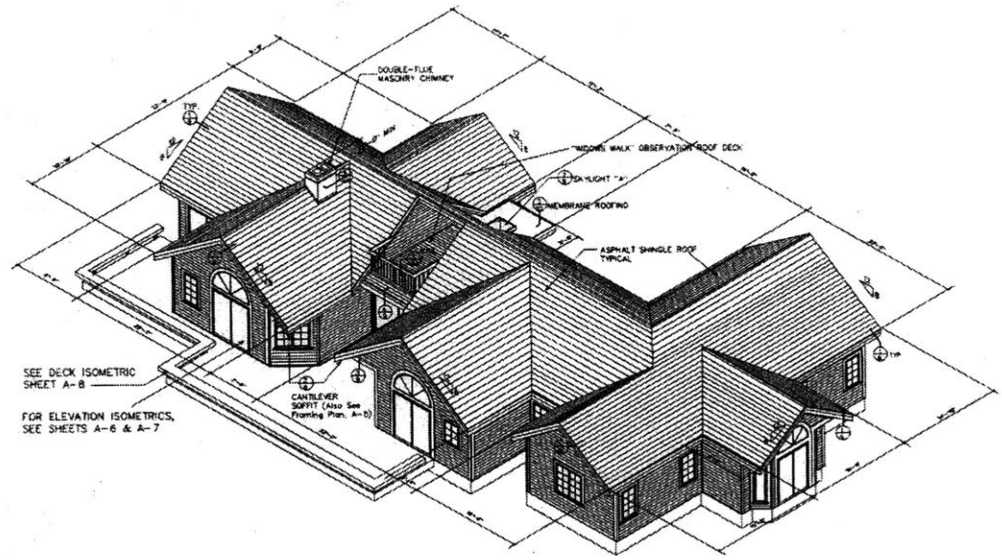


Figure 6. Example of three-dimensional construction drawing (from Architecture, *January 1994)*

ing the documents, Crawshaw and Daltry state that some of the deficiencies "seem insolubly linked with traditional methods of preparing and arranging working drawings." The limitations of a paper based documentation system are especially clear. Additionally, they go on to state that many of the underlying causes of the deficiencies are due to the "lack of guidance generally about what is to be communicated". What is evident is that in order for documentation to improve, new approaches and mediums need to be employed in order to overcome the inherent deficiencies of the current system. Crawshaw and Dalton make several recommendations to improve documentation standards. First, they state that the set should have a systematic structure with separate groups of location, schedule, assembly, and component drawings. The location drawings should be made large enough to minimize the fragmentation of the plans and elevations and direct references should be used wherever possible. Each view should also be fixed relative to each other through use of a grid or controlling line structure. Finally, Crawshaw and Dalton recommend that a brief guide should be included with the set to illustrate the arrangement and organization of the drawings (Crawshaw and Daltry 1973, 27, 28)

3. Digital Media in Practice

Computer Aided Drafting

Since the early beginnings of graphic software, the computer has been touted as a means to achieve an efficient and productive practice. This has been largely through the use of Computer Aided Drafting (CAD) software in the production of construction drawings. It is interesting to note that despite its widespread use in the majority of architectural offices, the computer has had little influence on the nature of construction documents. While the tools to produce construction drawings have changed, from pencil and paper to mouse and monitor, the content and structure has essentially remained the same. Computer-aided drafting software is used largely as a means to replicate known methods of work. As Malcolm McCullough indicates in *Abstracting Craft,* CAD is essentially a process of task automation where the computer is used to perform known processes more efficiently as opposed to replacing them with different, yet higher level modes of work. It is a view of the computer as a tool rather than a medium unto itself (McCullough 1996, 79).

This situation is not unique to architecture or even the computer. There is instead a long established pattern of new forms of media emulating the old. Early television programs, for example, often used a format based on that of radio. Even the beginnings of printed were tied to the processes of the past. Muriel Cooper, one of the founders of the Visible Languages Workshop at the Media Lab at MIT makes the observation that in its infancy, print

> . . .emulated the conventions of calligraphic writing on vellum; typography was modeled on the penmanship of the scriptorium; images and color establishment continued to be added to the printed page by hand, emulating the methods of the monastery (Cooper 1989, 4).

Many software programs, CAD included, have followed this route. Early paint programs for example, were generally modeled after the use of physical brushes. The behaviors of various painting mediums were overlayed on to a digital system (Cooper 1989, 4). The evolution of the operating system for the personal computer is another example. The DOS operating system made its debut in the early part of the 1980's. Using a strictly text based interface, DOS required the user to memorize obscure commands names in order to perform tasks such as copying and moving files. Searching for a more intuitive means to manipulate data and file structures, Apple offered its line of Macintosh computers, which made use of the now familiar desktop system. Finally, the explosion of the World Wide Web is providing a further developed conception of the compute as a global connection of computer systems.

In thinking about the role of the computer in the construction documentation process, it is interesting to consider some of the early research conducted in this area. One of the first graphic applications was demonstrated in 1963 at that year's Spring Joint Computer Conference. Ivan Sutherland, a graduate student at MIT's Lincoln Laboratory presented a documentary film on his recently developed "Sketchpad" system (Fig. 7). The film showed how an operator could move a light pen across a computer screen to create lines. Through use of keyboard commands, the operator could create an object. More commands entered on the keyboard could move the object, enlarge it, reduce it, or rotate it to show opposing side. This was the first public demonstration of interactive computer graphics. Those watching the demonstration regarded it as miraculous (Baker 1993, 14). Commenting on this new application, Sutherland stated:

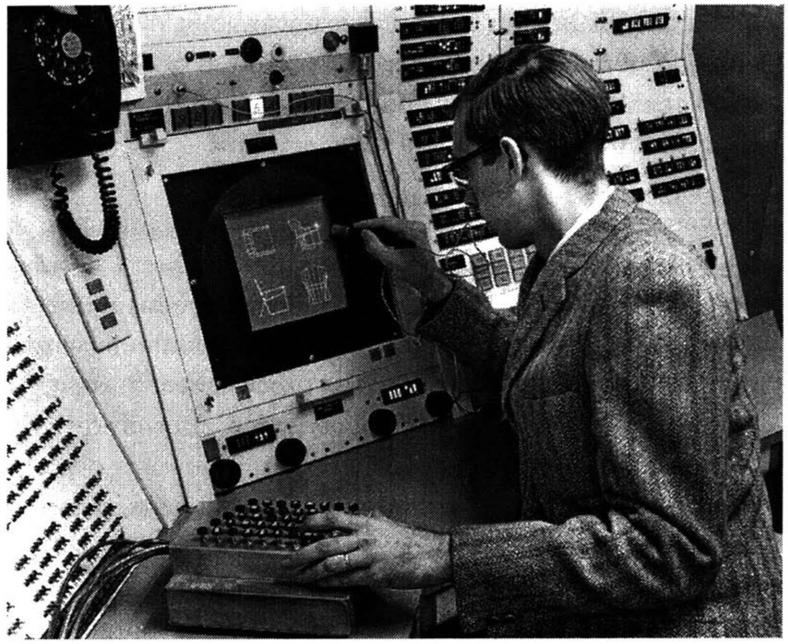


Figure 7. Ivan Sutherland and the SketchPad at MIT, 1963 (from Designing the Future*)*

> To a large extent it has turned out that the usefulness of computer drawings is precisely their structured nature . . . An ordinary [designer] is unconcerned with the structure of his drawing material. Pen and ink or pencil and paper have no inherent structure. They only make dirty marks on the paper. The [designer] is concerned principally with the drawings as a representation of the evolving design. The behavior of the computer-produced drawings, on the other hand, is critically dependent upon the topological and geometric structure built up in the computer memory as a result of drawing operations. The drawing itself has properties quite independent of the object it is describing (Baker 1993, 55).

It is interesting to note the state of computer equipment development at this time. Sutherland's display of graphical manipulations required an enormous amount of hardware and resources. The Whirlwind computer that was used weighed approximately 250 tons and contained over 12,500 vacuum tube. The size of such a machine was roughly equivalent to that of a large house (Baker 1993, 13). Today, exponentially more complex operations are accomplished on computers that inhabit more than one half of a desktop. However, regardless of the equipment used, Sutherland's comments toward the nature of computer drawings are still quite relevant.

As was stated earlier, CAD is essentially the automation of a known process. It allows an operator to enter information into the computer that would have once been drawn onto paper. The screen becomes an abstraction or simulation of the drawing surface. This process becomes particularly efficient in the area of revisions. However, this does not utilize the full potential of the computer drawing as was recognized by Sutherland. Paul Richens, a member of the Martin Center for Architectural and Urban Studies at the University of Cambridge, is actively involved in the development of architecturally based computer systems. Commenting on the use of computers in the professional practice of architecture, Richens states:

> Nowadays all large offices and a sizable minority of small ones have some CAD capacity. Nine times out of ten this means AutoCAD. It is used primarily for production drawings, in much the same way as a word processor is used for letters and reports. Its impact on the quality of architecture coming out of an office is about the same as that of a secretary's word processor — which is very little (Richens 1994, 307).

In order to become an effective component in construction documentation, the use of the computer must move beyond the conception of Computer Aided Drafting software. While CAD may speed up the drawing process, current implementations do nothing to address the previously indicated shortcomings of the standard documentation system. What is necessary is a rethinking of what constitutes a set of construction drawings and of how the inherent capabilities of the computer can be best utilized to achieve a more accurate and consistent transfer of information from designer to builder. In order to do this, it is important to look beyond the use of CAD and a strictly paper-based representation system.

Object-oriented Programming

One area that has received a great deal of attention regarding computer-aided design research is that of object-oriented programming. Object-oriented programming or OOP is a set of concepts that have been used extensively in software design and engineering in order to effectively construct and manage large-scale computational systems. While computer programs are certainly not building, at an abstract level, the concepts of OOP provide a means to organize information in an effective and meaningful manner. Additionally, the application of these concepts to architecture and building documentation is appropriate in that it follows a methodology inherent to the nature of the computational medium.

It seems safe to conclude that the computer will increasingly play a more important role in the profession. As was stated in the previous section, in order to maximize the use of the computer and address deficiencies in current methods of working, it is necessary to adopt practices that best use the inherent capabilities of the medium rather than translate current and past methods of work. Object-oriented programming concepts provide one means to approach this issue. This section provides an overview of the concepts of object-oriented programming. This is intended not as an exhaustive description of OOP but rather an introduction to some of the major underlying principles.

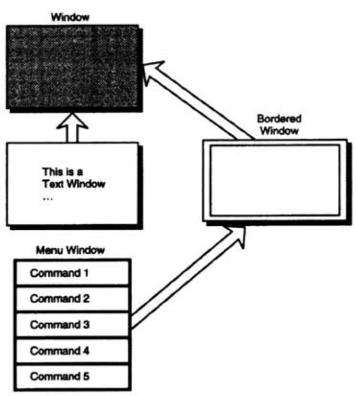


Figure 8. Class hierarchy diagram (from Object Orientation*)*

Object-oriented programming or OOP is an approach to software design that can comfortably manage complexity while providing a framework for later addition and reuse. The overriding concept of OOP is the division of complex tasks into small, easily managed pieces, called objects. These objects are computer abstractions that model the physical or abstract pieces of the system that is being simulated. In addition to greatly reducing the complexity of a given system, these objects can also be reused in other projects or combined to create more complex software modules. Object-oriented programming also allows

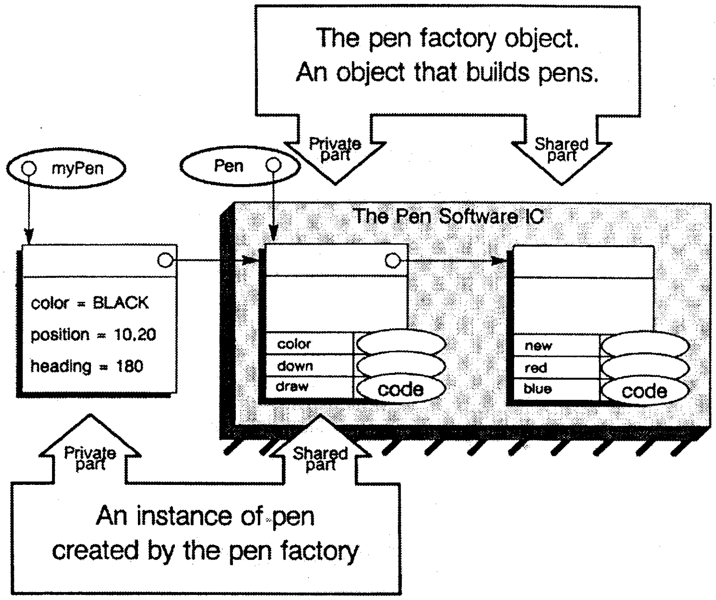


Figure 9. Instance creation diagram
(from Object-Oriented Programming: An Evolutionary Approach*)*

for easy modification and extension of individual components without requiring the programmer to re-code the entire component from scratch. As Brad Cox states in *Object-Oriented Programming: An Evolutionary Approach*, "Programmers no longer build entire programs from raw materials, the bare statements and expressions of a programming language. Instead they produce reusable software components by assembling components of other programmers" (Cox and Novobilski 1991, 2)

Object-oriented programming has evolved into a complex and intricate set of methods and operations. There exist many variations of object-oriented languages, each with their own specific characteristics and syntax. However, despite the differences in operation, all object-oriented languages share a fundamental set of principles. These defining principles are identified as abstract data typing, inheritance, and object identity (Khoshafian 1995, 7). It is these underlying principles that give object oriented programming its strength and separate it from the rigidity of procedurally based programming.

As was stated earlier, one of the key benefits of OOP is that it computationally models operations based on their real world counterparts. At the heart of the concept of OOP is the use of objects or abstract data types. Objects can be defined as abstractions that contain the properties (data) and behaviors (operations) of the entities within a system (Schroeder, Martin, and Lorensen 1998, 19). Commonly, these abstract data types are implemented as classes in most OOP languages. A class is essentially a type or a set of similar objects (Fig. 8). These objects have similar structure or representation and exhibit similar behaviors. In other words, classes provide constructs that directly define that data structures of an object as well as the operations that are used to manipulate instances of the data structure (Khoshafian 1995, 9).

Once a class is defined, it is possible to create a collection of objects from that class. These objects are referred to as instances of the class and maintain all of the internal representations of the class (Fig. 9). As Setrag Khoshafian states in *Object Orientation*, "A class is like a factory that produces instances, each with the same structure and behavior" (Khoshafian 1995, 77). Taken as a whole, all of the instances of a particular class form a collection or extension of that class. Within the collection of instances, the nature of the abstract data type allows for variations and differences among the instances. While the names and types of attributes and methods may be the same, the specific values within an instance may be different.

There are many obvious benefits associated with the use of classes. Most importantly, classes allow a better conceptualization and modeling of the real world. While classes and objects are not physical entities, they share many characteristics of everyday objects such as defining characteristics and functions. The use of classes also enhances the performance and robustness of the system. This is largely due to the separation of implementation from specification. This makes it possible to make adjustments or modifications to a particular class and greatly facilitates extensibility and reuse within other systems (Khoshafian 1995, 44).

Another powerful concept regarding the use of classes is the principle of inheritance. As was stated earlier, the use of OOP facilitates extensibility and reuse. This is largely accomplished through the use of inheritance, which allows new classes to be built on top of existing classes. These new classes will then inherit both the behavior and the representation of the existing class without requiring the inclusion of those definitions within the class itself. This allows slight variations of a class to be developed and also taxonomizes the classes into well-defined hierarchies with explicit parent/child or super/sub class relationships (Khoshafian 1995, 78). Since the use of inheritance creates a hierarchical class structure within the system, it is possible to create what are known as abstract classes, or classes that exist only to serve as a super-class. Such abstract classes are very useful for gathering together all of the attributes and methods that a set of sub-classes will use (Schroeder, Martin, and Lorensen 1998, 24)

In certain instances, it is desirable to have a class inherit properties from more than one class. Most object-oriented languages allow the creation of classes with multiple inheritances. Such a sub-class would combine the characteristics of each super-class and any subsequent changes made to either super-class would directly effect the sub-class. Through the use of multiple inheritance, it is possible to rapidly develop a complex network of class variations (Khoshafian 1995, 9).

Object-oriented programming offers a powerful tools for managing and implementing computational systems. It provides an open structure that accommodates modularity and reuse. While these principles are not directly related to construction documentation, the main principles can be applied to the organization of building related information, as will be discussed in the next chapter.

Digital Documentation

Many of the deficiencies of current construction drawings can be attributed to the medium on which these documents are delivered. While paper is certainly a highly portable and flexible medium, its inherent disadvantages limit its overall effectiveness as a systematic means of construction documentation. Similarly, the use of CAD software as a tool to automate the task the producing paper based construction drawings only supports this system of distribution and does nothing to address the inherent shortcomings. In order to overcome the deficiencies indicated by the work of Crawshaw and Dalton, it is necessary to reconsider the nature of the construction document. The digital environment offers one means to do so.

The *American Heritage Dictionary* defines a document as "A written or printed paper bearing the original, official, or legal form of something, and which can be used to furnish decisive evidence or information." The emphasis here is on the physical representation of the document as an official "paper". However, over the course of several years, digital media has challenged this traditional view toward what constitutes a document. Files created using spreadsheet or word processing software exist as digital versions of their physical counterparts. Is it possible to say that one is more of a "document" than the other? In his book, *Digital Documents*, Bruce Duyshart refers to a document as any container of information (Duyshart 1997, 1). This definition is independent of the medium used, giving equal relevance to both the physical and the digital. The emphasis is instead on the information. While the medium used certainly matters, it is the ability of a document to convey the meaning and the message of the author that is most critical.

Yet one cannot simply discount the nature of the medium that is used. Each form has advantages and disadvantages unique to its own characteristics. The appropriateness of a particular medium is dependent on the nature of the information contained within. In the instance of paper based documents, the advantages are fairly obvious. As a medium, paper is highly portable and easy to reproduce. It is relatively inexpensive to distribute has high contrast display characteristics. Paper is familiar to every user. However, these advantages also have their drawbacks. Paper documents often require large amounts of storage space and require the use of physical delivery and distribution services. Also, information within a paper document can be difficult to access and locate as well as reuse in other documents. While these disadvantages may not be considerable in every instance, for certain applications, such drawbacks can become highly problematic (Duyshart 1997, 2,3)

In contrast, digital documents provide a great deal of control and flexibility. Due to their very nature, digital documents require almost no physical storage space. Tens of thousands of text pages can be contained within a single CD-ROM. Also, using established hierarchies and structures, it is possible to control and manage various groups of documents as well as distribute them via the Internet or e-mail. The proliferation of database software as well as Internet search engines makes it considerably easier to search for relevant content within a series of digital documents.

The construction drawing occupies a rather interesting position between the paper and the digital document. The majority of construction documentation is produced through the use of digital tools, be it CAD or word processing software. These documents are then output in a paper format. Through the course of this process, the body of information that has been assembled through the design process is fractured into a collection of graphic tokens that are then related to the building entity through a series of non-hierarchical grouping symbols. This disassembly requires the user to sort through the documentation, the sheets of drawings and specifications, in order to locate the particular information they seek. As was stated earlier, CAD software does nothing to address the deficiencies indicated by the research of Crawshaw and Dalton. There exist, however, several widely used digital technologies that could be utilized to explicitly address these issues. It is first necessary to move beyond the established conception of the document as a strictly paper-based object.

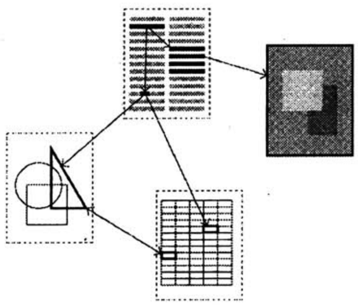


Figure 10. Hyperlinks between documents (from Digital Design Media)

The widespread popularity of the Internet has clearly shown the value of linking various sources of information in a systematic manner. These connections from one source to another allow the user to navigate their own path through the information in a manner that is appropriate to the task at hand. While this is often used to excess on the Internet, the use of hyperlinks and hypermedia provides a means to move beyond the page based document model and the conception of information as a strictly two dimensional linear flow (Fig. 10). As Kathryn Henderson states in *Online and On Paper*, "Information flow is a myth. Information does not flow but rather must be constructed interactively by the human and nonhuman actors involved" (Henersen 1999, 59) Hypermedia treats the document as a collection of nodes through which the user can navigate. While physical pages can hinder access to information, hypermedia makes the organizational structure of a particular document explicitly available to the user and shifts the emphasis from reading fixed sequences to developing a series of associations (Mitchell and McCullough 1995, 317)

The direct access symbols, such as the section of elevation markers, used in a set of construction documents acts as a primitive form of hypermedia links. However, instead of bringing the user directly to the information they seek, it is necessary to manually locate the information by following the notation included with the symbol. As was indicated by Crawshaw and Dalton, this can sometimes lead to misinformation when the symbol notation is incorrect or the user misinterprets it. Hypermedia is one method to eliminate this uncertainty by bringing the information directly to the user, rather than bringing the user to the information.

While hypermedia provides a means to directly access information, database technology naturally lends itself to the overall organization and structure of a document. Through viewing the construction document set as not simply a collection of drawings but rather as an assemblage of various items of information relating to a single entity, the idea of a central repository for all instances of data becomes quite clear. The database has several advantageous characteristics. First, it provides an efficient means to store data that eliminates the possibility of multiple versions of a single item of information. Construction drawings contain thousands of discrete elements of information. Often times this information is used in several locations within a particular set of drawings. By simply creating a single instance of this item that is stored in the database and then linking it to various locations eliminates the necessity for correcting or updating multiple versions. This data can also be used in a multitude of ways, all of which support the various uses of the construction documents. The database can also be constantly added to or modified over the course of construction. Given the ever-changing nature of the industry, such a structure provides a needed level of flexibility (Mitchell and McCullough 1995, 317, 355)

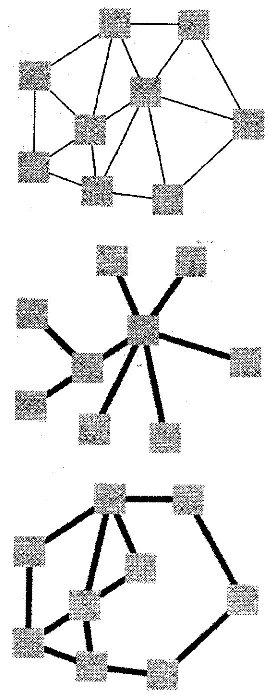


Figure 11. Node relationship diagram (from Digital Design Media*)*

Database structures provides an efficient means to indirectly search the content of a document for relevant information. As Crawshaw and Daltry indicated, one of the major deficiencies of traditional construction documentation is the difficulty in locating information that has not been directly referenced. Through use of a database search engine, it is possible to locate items within the overall structure based on a variety of criteria, such as title, material or location. Much like hypermedia, a database structure can eliminate much of the uncertainty inherent within traditional forms of documentation (Schilling and Schilling 1987, 133).

What is also interesting with regard to the issue of digital documentation is the opportunity to exploit forms of representation not possible within the confines of the paper document. Interactive three-dimensional models as well as sequence or time-based animation could prove to be quite useful in addressing the issues of information adequacy. Dace Campbell, an architect and researcher with NBBJ Architects provides one such example with his application of VRML (Virtual Reality Modeling Language) to a set of web based construction documents. In this instance, a region of a building currently in the design process was selected as a test case. A three-dimensional model, including the associated dimensions and notations, was created using the VRML language. Hyperlinks were created to connect aspects of the model with text-based specifications and other associated information. The completed model was then made available on a dedicated web site for the project. Using a VRML browser, users could navigate through the model interactively. Successive levels of detail were employed to illustrate aspects of the project in a range of complexities. From a user's point of view, the VRML model made it possible to experience and understand the spatial intentions of the designer in a manner not previously possible. The documentation becomes more than simply lines depicted on the page but rather a dynamic representation of information (Campbell 1998)

It is quite clear that the concept of the digital document affords many advantages over its physical counterpart. In addition, by looking at digital tools beyond the scope of CAD software, it becomes possible to address the inherent deficiencies within traditional construction documentation systems. As Muriel Cooper states, "Visualization and graphic manipulation of information, interface, and interactive design will be valued not as cosmetics, but as vital necessities in an information society" (Cooper 1989, 30) Given the increasingly complex nature of construction and its documentation, this sentiment is becoming more and more evident. In order for the architect to retain a leading role in the construction process, new tools

4. Implementation

Based on the research documented in the preceding sections of this paper, a series of prototypes were developed to address deficiencies within current modes of construction documentation the use of digital technology. As the move is made toward more computational modes of representation, it seems clear that instead of continuing with existing paper-based practices, more appropriate organizational structures can be developed and utilized to better interface with the digital medium. The concepts of object-oriented programming are used here as a primary organizational structure for construction information. The first prototype, the Building Interface, was designed to provide interactive access to building information through use of multiple levels of abstraction. The second prototype, the Drawing Assembler, addressed information reuse in the documentation process through use of objects as the base level of representation.

As Crawshaw and Daltry indicate, many of the deficiencies inherent within current modes of construction documentation can be attributed to the nature of the paper based document delivery system. Additionally, as an increasing amount of construction information is being prepared through use of the computer, there is the opportunity to reformulate the construction document in a manner more appropriate to the medium used in its development. If construction documentation can be viewed as purely a collection of information rather than specific drawings or specifications, there is the opportunity utilize research and thinking in the fields of interface design and information visualization. In this sense, construction documentation can become an interactive, multimedia interface rather than simply sheets of paper.

One of the key benefits to the use of object-oriented programming is that it facilitates the reuse of computational components. In this sense, a program can be assembled from pre-built elements rather than constructed from scratch every time. Similarly, most architectural firms develop a set of standard construction details. However, current documentation tools, namely CAD software, do not actively support the integrated reuse of this information in a systematic manner. The preparer of the documentation must either have knowledge of the existence of a particular detail or must search through several directories to locate the detail that is needed. As a result, details are often generated from scratch as the time required to locate and manipulate an existing detail is often equivalent to actually drawing it outright. A prototype of the Drawing Assembler was developed to computationally facilitate information reuse in the documentation process. The Drawing Assembler provides access to the extents of a firm's accumulated detail knowledge by making such information readily available to the user in a contextual manner.

Information Organization

One of the fundamental issues regarding this pursuit is the organization of information. This is a prevalent concern regardless of the form of the document. Through the evolution of two-dimensional documentation techniques, a consistent language developed to provide a means to access the information. As was stated earlier, these methods provide a reasonable means of access but are deficient on several levels. With the predominant use of digital technology in the profession, it seems clear that rather than use traditional techniques in an electronic manner, it may be advantageous to investigate how similar issues have been addressed by those in the computer science discipline. While the abstractions of computers and computer programming are quite removed from the physical nature of architecture, at a more symbolic level, both deal with the organization and representation of information.

An area that has received a great deal of attention in computer-aided design research is that of object-oriented design and analysis. Driving this line of thought is the recognition, or perhaps, re-recognition, of buildings as assemblies of discrete components, similar to the classes and objects used by computer programmers to develop software. Again, this is not a new concept or understanding (see Habraken 1979). However, current methods of documentation do not directly address this notion and rely instead on the "views" of the building, i.e.. plan, section, and elevation. While these drawing types are efficient in their ability to abstractly provide an , they do not make available other, more significant, relationships that are inherent in the built form. This is not to suggest that such modes of representation are outdated and no longer valid. This would be a rather presumptuous statement given both the history and versatility of two dimensional graphics conventions. Rather, it is a suggestion that additional methods of representation can be utilized to provide "views" that are not strictly based on geometric or visual relationships.

As was indicated earlier, object-orientation is not a new concept within the realm of CAD research. Yehuda Kalay, a professor of architecture at the University of California at Berkeley, has written extensively on the need for what he calls "semantically rich" building representations (Kalay, 1997). Additionally, Anton Harfmann, a professor at the University of Cincinnati, proposes a component-based paradigm as an appropriate model for building representation. There are many other examples of similar research into the application of object-oriented principles to building representation. However, most of this research focuses primarily on early stages of design (Kalay) or on constructing three-dimensional models (Harfmann). There is the opportunity to extend these concepts into the stages of construction documentation and utilize the advantages of the system throughout the entire design/ documentation process

Classes can be used to describe building components utilizing inheritance as a means to easily develop a library of objects. Similarly, the principles of object-oriented analysis can be used to describe the relationships between components and their associated assemblies. Object model diagrams can be used to illustrate the extents of components within a given building. In a similar manner, functional model diagrams can illustrate how these class instances or components are organized to form distinct assemblies within the building. While these diagrams do not indicate the geometric properties of the objects, they do make clear inherent relationships between the objects as well as how they are applied to construct the major and minor assemblies within the building.

Building Interface

The intention of the Building Interface is to provide a structure and a means for visualizing building information in a more dynamic and interactive manner. This is accomplished by providing visual access to several levels of information relationships. Through these means, the interface provides a more comprehensive and consistent structure for accessing building information. Where as traditional document sets are generally comprised of several sets of drawings for each specific discipline, the Building Interface acts as a singular point of reference for each individual discipline as well as a means to dynamically interact with a variety of representations.

This research made use of many concepts developed by researchers in the discipline of computer-human interface design. Again, the approach is one of visualizing information, in the general sense, rather than by focusing purely on the developed methods of construction documentation. It was the intention that by looking beyond the strict conventions of the architectural profession, insight into the more fundamental issues of information design and interaction could be gained. There were four main areas that the Building Interface looked to address. These areas

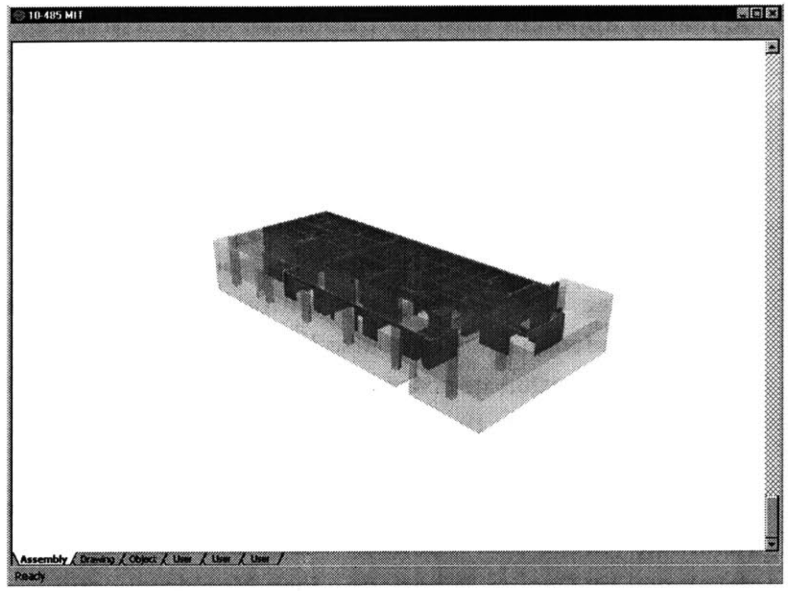


Figure 12. Main window of Building Interface

consisted primarily of levels of abstraction, information density, scalar zoom, and information filter. In all of these instances, research was conducted into the fields of interface design and information visualization for strategies and techniques that could be effectively applied to construction documentation. These concepts were then synthesized into a single prototype.

While there certainly a great deal of unresolved issues regarding the distribution and legal consequences of such a form of document, the intention of this prototype was to explore the possibilities of displaying rather than address the issues involved in the practical application of such a proposal. As such, the Building Interface exists as an attempt to apply several concepts of interface design and information visualization to formulation of a digital set of construction documents.

While traditional two dimensional drawings provide an abstracted view of the geometric properties of a building, the concepts of object-oriented programming provide an additional means with which to "views" building information. As was stated earlier, by viewing information in such a manner, it is possible to recognized inherent relationships between assemblies and components that may not be readily apparent through traditional documentation techniques. Much research in this area has come from the need to visualize abstract objects and relationships in computer science. Peter Young's paper, Three Dimensional Information Visualization" provides an overview of current visualization techniques (Young 1996). Where as these methods are essential in such disciplines as computer science, which operates at a fundamentally abstract level, these techniques may also be applied to construction documentation in order to make apparent inherent relationships within the building.

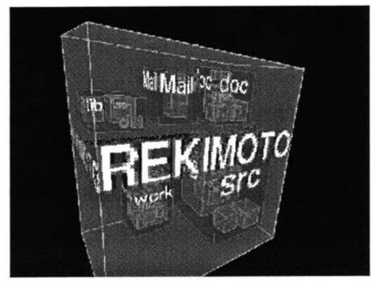


Figure 13. View of Info Cube

The Information Cube, developed by Mark Green and Jun Rekimoto (Green and Rekimoto 1993) and Cone Trees, developed by George Robertson, Jock Mackinlay, and Stuart Card at Xerox PARC (Robertson, Mackinlay, and Card 1991) are two such methods that were utilized in the Building Interface. The Information Cube (Fig. 13) provides a three dimensional view of hierarchial information based on the nested box metaphor. The Cone Tree (fig. 14) is another three dimensional visualization technique that provides access to the hierarchical structure of a collection of information. Both of these visualization methods were employed in providing access to multiple levels of abstraction within the Building Interface.

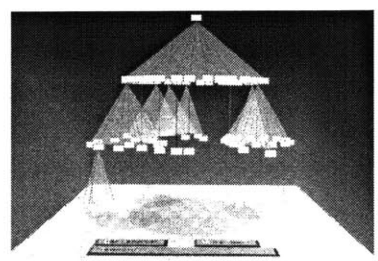


Figure 14. Cone Tree hierarchy

Another area of interest was that of information density. This is an area that has received quite a bit of interest in the fields of information visualization and interface design. As construction projects and the resultant information required in their construction are growing increasingly complex, what is needed is a means to efficiently manage and visualize the information in a clear manner so as not to overwhelm the user. Providing variable levels of information density is one means to achieve this goal. In this sense, the user can select how "deep" within the hierarchy they wish to view. The density of a particular display is then directly related to how much of the hierarchy is displayed at one time. It is largely a process of managing the display of complex relationships.

Displaying information at a variety of scales in another effective means for managing the amount of information at a single instance. Construction documents already make effective use of this technique. However, the fractured nature of traditional documents does not provide a consistent sense of context with regard to changes in scale. Often times, related drawings of differing scales are located on separate sheets, forcing the user to flip back and forth in order to understand the area in question. Digital media is not necessarily bound to this page-based system. Instead, it is possible to use real-world conventions with regard to changes in scale, notably moving zooming in and out of an object as a means to view more detailed information.

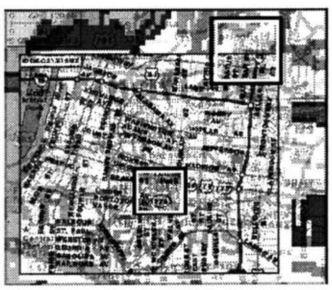


Figure 15. View of macroscope

Henry Liberman, a research scientist at the MIT Media Lab developed the macroscope (Fig. 15) as a means to interactively browse a very large display space (Liberman, 1994). Using Charles and Ray Eames' film, *The Powers of Ten*, as a metaphor for interacting with digital media, Liberman's macroscope using zooming and panning in multiple translucent layers as a means to provide a sense of context at multiple scales. While Liberman's research was applied in a more geographic setting (a map of the United States), the principle of dynamic zooming could very well be applied to construction documentation, allowing the move from general to specific information in a consistent and contextual manner.

Finally, traditional construction documentation often contains several sets of drawings, each related to a specific discipline (such as architecture, structural engineering, mechanical engineering, etc...) In order to coordinate various systems, it is often necessary to check multiple sets of drawings for the necessary information. However, through the use of digital documentation, it is possible to create various filters that "look" onto a consistent body of information. It is a again a matter of providing contextual information from which to make associations between information specific to individual disciplines. This is essentially how layers work in most CAD drafting packages. However, once the drawings are printed, this separation of information is lost. Through use of a digital documentation system, such a separation of information is possible. Matt Belge, a researcher at SunSoft, Inc. developed a graphic user interface system that made use of transparent layering to organize information (Belge, Lokuge, and Rivers 1993). Through use of such a system, it is possible to view various forms of information from a consistent representation.

The Building Interface serves as an interactive interface for construction information. It provides a multitude of levels of abstraction from which to view the construction information, ranging from three dimensional models to two dimensional hierarchical diagrams of assembly and component relationships. The idea is to provide a multitude of "views", both abstract and representation, from which to visualize the information.

The Building Interface is operated from the main interactive window (Fig. 12). The main abstraction view is presented and can be rotated by clicking either left or right on the window. The current level of abstraction (Fig. 16) can be changed by clicking either up or down. The default view is the three dimensional model or the lowest level of abstraction. Moving the cursor to the left causes a tilebar to appear (fig. 17). This bar indicates the current level of the hierarchy as well as the next levels higher and lower. The location within the three dimensional representation are highlighted by moving the mouse over the appropriate level on the bar. Additionally, the user can navigate to the next level of the hierarchy by clicking on the bar. This indicates a change from general to specific and the main window zooms into the selected level of the model. At this point, it is also possible to change the density of information by moving the scroll bar located to the left of the main window. This shifts the view to display information at deeper levels in the hierarchy.

In order view more direct information about a particular level, the user selects that region from the main window, causing the information panel to appear (Fig. 18). The

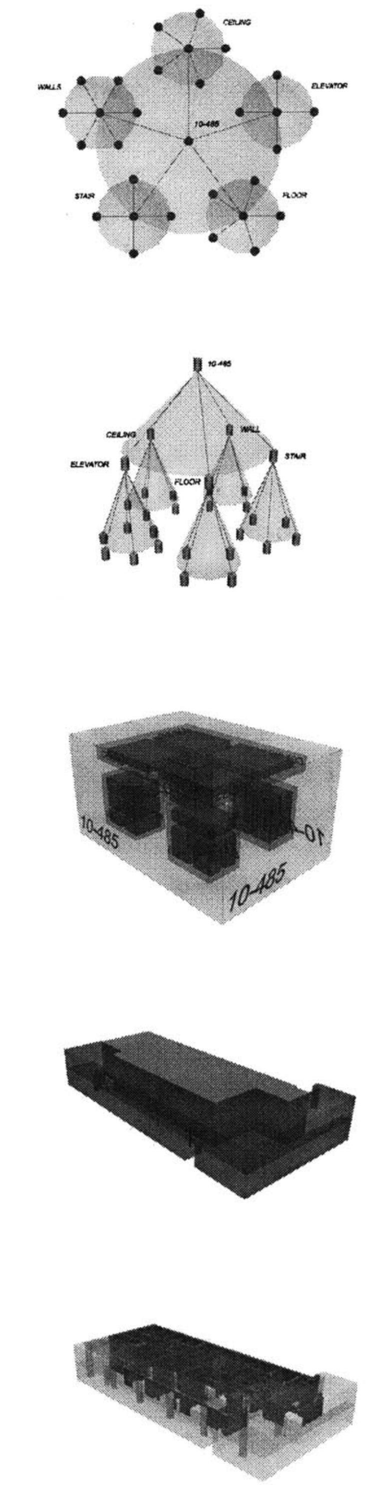


Figure 16. Levels of abstraction

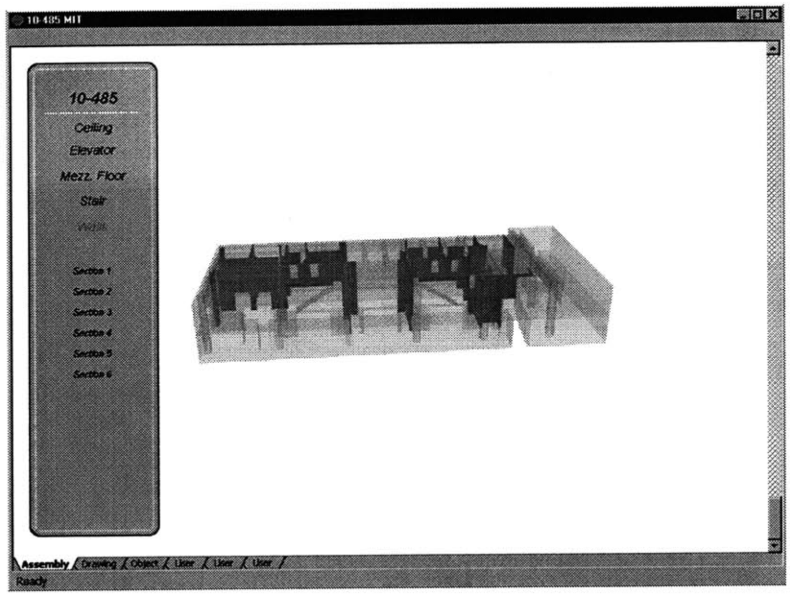


Figure 17. Level selection menu of Building Interface

information panel contains all of the information appropriate at that level of the hierarchy. This information can take many forms, including drawings, specification, or photographs as well as video or interactive media. The Building Interface provides a comprehensive and singular access structure for displaying and navigating construction information in a dynamic and interactive manner.

Drawing Assembler

While the Building Interface addresses the issue of information visualization, one of the more prevalent issues regarding construction documentation is directly related to its process of production. As was stated earlier, current CAD programs provide automation only on the level of the task performed (McCullough 1996). Drafting a set of documents on a computer may only marginally speed up the process. Additionally, the information that is contained in a set of documents is often highly fragmented and localized to the particular project. While many firms create a set of standard details, often times these details require modification to fit specific instances. Since these variations are generally not documented or indexed from a

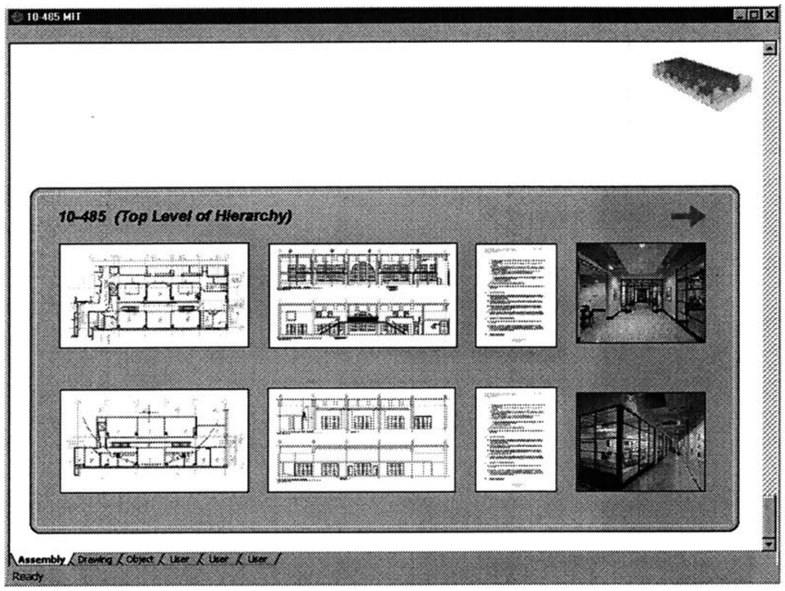


Figure 18. Main information panel

central source, similar details developed for other projects often go unnoticed. This results in a great deal of redundant details and time spent solving previously resolved problems.

One of the key benefits to object-oriented programming is the ability to reuse software components. In this sense, programming becomes a process of assembling the various functional components rather than creating everything from scratch. A similar approach has also been used successfully in the development of multimedia applications (Kahn, Nanard, and Nanard 1998) through the use of design templates. The emphasis in both cases is on the strategic reuse of information. This approach provides many obvious advantages with regard to development time and quality control. A component can be evolved over time and the initial investment in developing the component is recouped in its continuous use in other situations.

In order to directly address the deficiencies related to the process of construction documentation, what is needed instead is a system that recognizes the component nature of construction and provides an immediate means to access the depth of a firm's construction knowledge. An automation of the process rather than the tasks of construction documentation. The Drawing Assembler is an approach to construction documentation that functions at the level of the component. Other research has been conducted in this area, specifically that of Harfmann (Harfmann 1993) and Gross (Gross 1996). Both, however, focus primarily on early stages of design and on developing precise representations regarding the arrangement of elements in a planimetric manner. The emphasis is on resolving the relationships between components are they are assembled in either two dimensions (Gross 1996) or three (Harfmann 1993). In contrast, the Drawing Assembler operates on the level of the detail. Rather than resolving interfacing components at the level of the plan, the Drawing Assembler serves as a means to search and generate details that are specific to the interfacing components. It is more along the lines of a graphical search engine than drafting or modeling software.

The Drawing Assembler (Fig. 19) is a means to interactively bring together two separate databases in a graphical manner. The first database is the Component Database. It contains a collection or library of assemblies and components for use in assembling a building. These objects are defined by type, typically using the CSI specification for classification. The second database contains a series of parametrically defined details. These details are similarly defined through use of a systematic classification system. The Drawing Assembler then provides a means to collect details particular to a specific building through the interaction of various objects. The emphasis is on the intersection of the objects rather than their precise geometric location.

The majority of commercially available CAD systems focus primarily on the geometric properties of objects and entities. Rather than continue with this line of reasoning, the intention of the project was to instead focus on the details. As such, the primary means of object assembly was kept purposefully primitive, using only two-dimensional representation and allowing imprecise intersections. Again, the emphasis was on the occurrence of intersection rather than the precise geometric location of

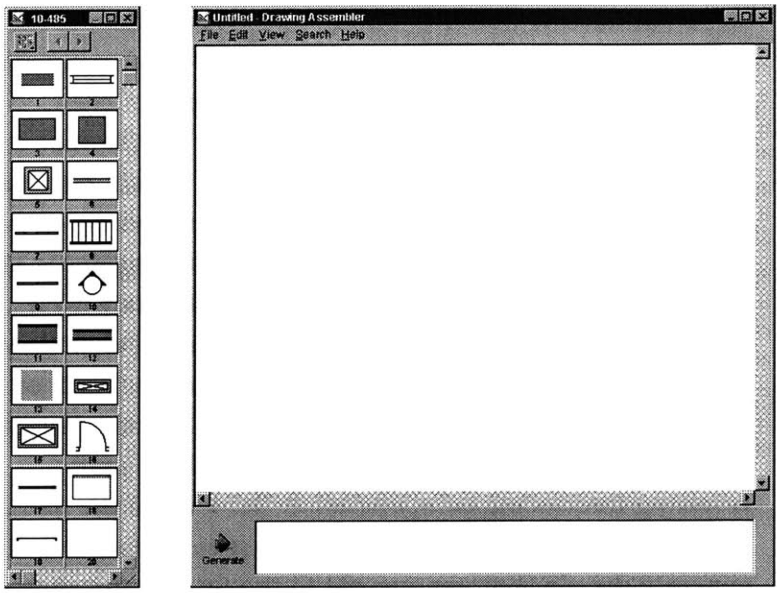
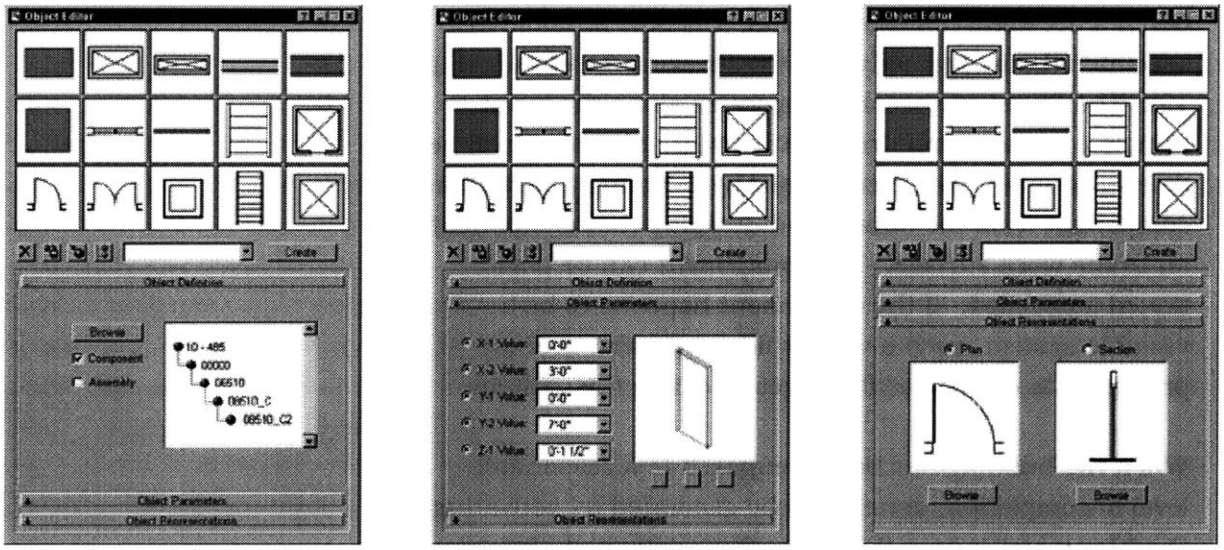


Figure 19. Main window of Drawing Assembler

the intersection. The generated details would be used to specifically address this information while, as recommend by Keith Styles in The Working Drawing Handbook (Styles, 1995), plan and section would be used primarily as a means to locate this more specific information.

Objects in the Component Database are created and modified using the Object Editor (Fig. 20). The Object Editor acts as a means to establish the various representations of an object (both plan and section) as well as its particular definition and parameters. The Object Editor consists of three main sections. In the first section, Object Definition, the search parameters that will be used in the Drawing Assembler are defined. The level of specificity for a particular object is entirely variable. Using the organizational structure discussed previously, objects can be defined at any level of the hierarchy. The level of specificity for a particular object can either broaden or narrow the detail search in the Drawing Assembler. This flexibility makes it possible to use the system as either a pure documentation tool in the sense that the desired detail condition is well defined or in a more suggestive manner, using more broadly defined objects to examine a range of potential solutions.



Object Definition *Object Parameters* *Object Representations*

Figure 20. Object Editor menus.

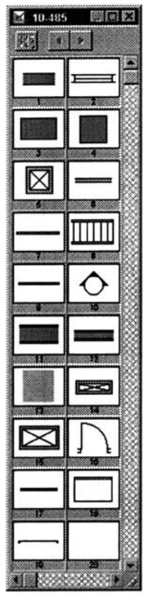
After defining an objects description, the user can enter values for an objects particular parameters. The variable parameters are directly determined by the objects definition. A more specific definition may require more parameters. These parameters define the geometric dimensions of the object and are used to generate the resultant detail once a proper match has been made. The definition of such parametric detail templates is such that they can directly interface with the values of the specific objects. The final step of the Object Editor is to select the particular symbols that will be used to represent an object in the Drawing Assembler. As will be discussed later, these representations include both plan and section. The symbols are the primary means of manipulation in the Drawing Assembler but are not necessarily directly associated to the objects parameters. As in traditional drawing techniques, a simplified symbol can be used to abstractly represent a more complex geometric component. It is at the level of the detail that the particular geometric properties of a component will represent in depth. Once a series of objects have been defined and made part of the Object Database, the user can use the Object Editor to organize a collection of objects into a project specific library.

Given a set library of components, it is now possible to assemble them together using the Drawing Assembler (Fig. 19). The Drawing Assembler operates along the input principle of "drag and drop". A plan (or section, as will be discussed later) can be composed by selecting a particular component from the library and dragging that component on to the work surface. This component can be positioned as needed. If the component happens to be a variable system, such a wall or floor type, the user is prompted to enter specific values to define this system.

When the building has been composed to a reasonable level of completion, details can be generated by activating the "Generate" button (Fig. 21). Once activated, the system determines all of the intersections between various components. The system then systematically isolates each individual intersection and, using the definitions of each component, performs a boolean search (if the definition of A and B equals TRUE) of the Detail Database to determine a matching detail. If there are details that match the definitions of the intersecting object, a detail is generated using the parameters defined by each object.

In instances where there is more than one detail that matches the descriptions, the user is presented with a selection of details from which to choose. This operates on two levels in terms of document production. Objects can either be specifically defined in order to perform a narrow search to generate a particular detail or objects can be loosely defined in order to select from a broader range of detail options. This can vary from object to object as they are specified in the Object Editor.

Once a detail is generated, it is placed in the detail window and a marker is generated at the intersection of the components to indicate the existence of such a detail. Similarly, if the search yields no details matching the objects definitions, a marker in generated to indicate that there is no detail for this specific condition. At this point, the user can either change the definition of the objects and search again or create a specific detail to address that particular condition and enter it into the Detail Database.



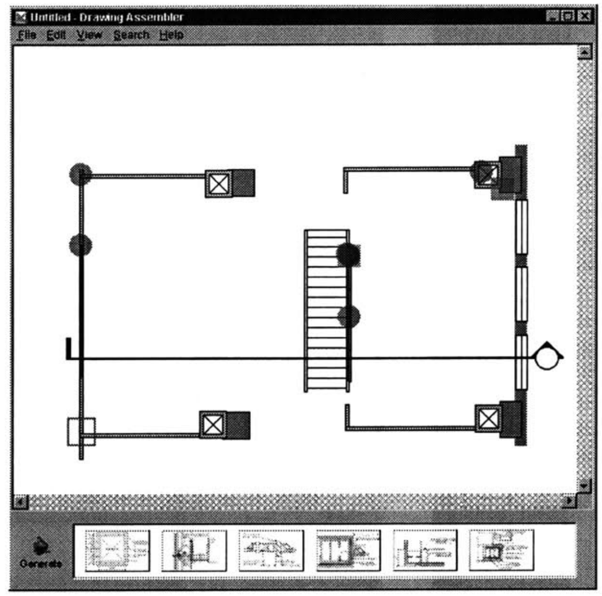


Figure 21. Generation Sequence - Details are generated based on the intersection of dissimilar objects

Throughout the course of this search and generate process, the Drawing Assembler maintains a list of the details generated and checks to see if instances of a particular detail have already been created in order to prevent a redundancy of information. If a detail has previously been generated, the subsequent intersections of a similar type are indicated with markers that reference the initial generation of the detail.

Similar to component objects, the Drawing Assembler also allows the use of more abstract or representational objects. One such example is the Section Object, which takes into account the assembly of components in the Z axis. The section object is used in the same manner as other component or assembly objects and simply dragged on to the work surface and positioned accordingly. However, by clicking on the section object, the plan view is replaced by a section at that particular location. Objects created in the Object Editor are defined with both plan and section views as well as parameters for each The section object simply recognizes what objects it intersects and creates a view placing the section representations relative to their location in plan. However, when objects are entered into plan view, the Drawing Assembler does

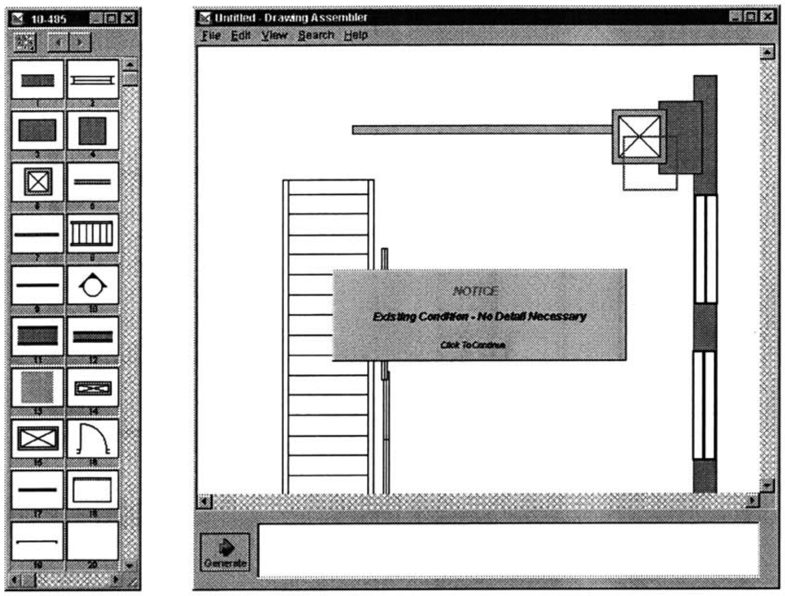


Figure 22. Generation Sequence - Notice window indicating that no match was found in the detail database

not request input regarding their position in the Z-axis. When the section object is first used, it is necessary to position the objects relative to the Z-axis. Again, the intention of the project was to focus on the intersection of objects, rather than their precise location in three-dimensional space. As such, no effort was extended to solve this particular problem.

Understandably, a section often contains components and, more generally, information that is not present in plan. However, the section object works in a similar fashion to the plan view, allowing the user to drag and drop objects into place and then search and generate details. The operation is unchanged from that of entering object in plan, objects are simply dragged to the work space and dropped into place. In order to relate the section to the plan, a plan object is created in the section, creating a linkage between the two representational views. In either case, it is possible to create multiple representations by creating either plan or section objects accordingly. This provides a consistent referencing structure among the various location drawings. Unlike the notion of the comprehensive three-dimensional building model, this approach makes use of the efficiency afforded by abstract

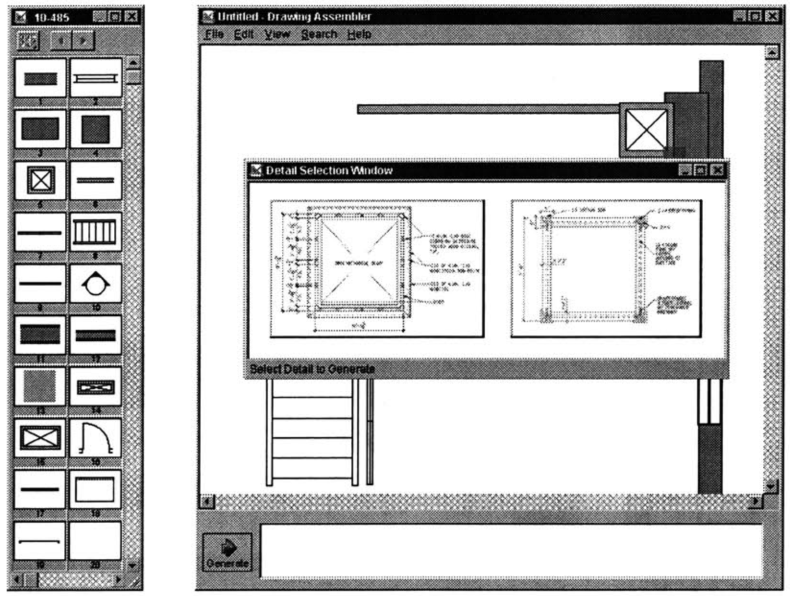


Figure 23. Generation Sequence - Detail selection window prompts user to select appropriate detail

two-dimensional representations while providing an integrated framework for building description and documentation.

Once a building has been assembled and the details generated, the Drawing Assembler provides a means to organize this information on drawing sheets of a specified size. This process allows the user to specify a particular organizational structure (by drawing type, element, scale, etc. . .) and then places the detail on the sheets according to this structure. At the same time, location drawings (plan and section, depending on what was used to compose the building) are generated and appropriately keyed to the location of the location of the details on the sheets. This addresses on of the main deficiencies indicated by the research of Crawshaw and Dalton, that being the location of information within a set construction documents. Rather have the user keep track of each individual element of information, this approach makes use of the computer's inherent capabilities for managing complex yet defined associations. As a final step, the drawings would be printed as a set and a file encapsulating the drawing information generated and saved.

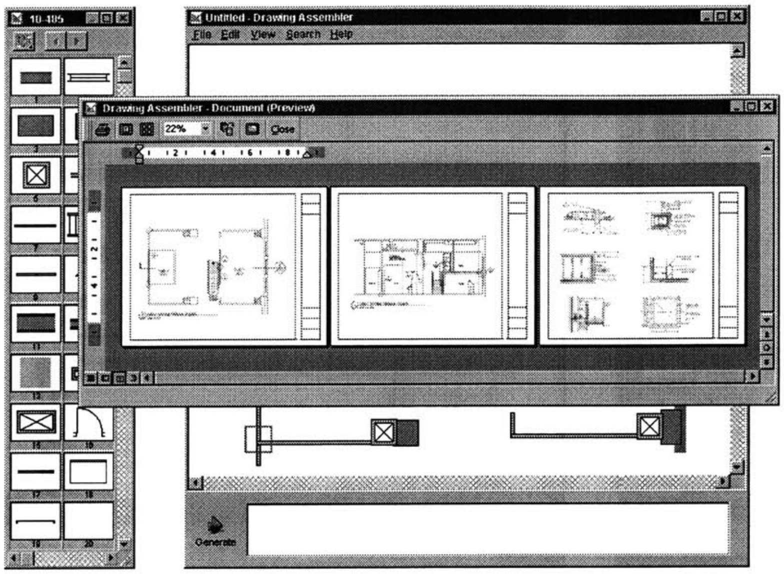


Figure 24. Print Sequence - Details and location drawings are generated and keyed appropriately

The Drawing Assembler is an approach to construction documentation that focuses on an automation of the process through the systematic reuse of information, in the form of construction details. In addition, the Drawing Assembler operates at the level of the building component. In this sense, construction documents can be assembled from objects rather than drawn by lines. It functions as a form of graphical search engine, more closely related to Altavista and Yahoo than to either AutoCAD or MicroStation. The intention, however, is not to remove the architect from this process. Rather, the intention of the Drawing Assembler is to make a base of knowledge, one that already exists within every practicing firm, readily available to the user. This is not a knowledge-based system in the sense it determines the appropriateness and feasibility of a particular construction technique. It is instead a means to access the range of solutions developed over time by a particular firm in a comprehensive and systematic manner. The intelligence of the system still rests, as it has since the very beginning of the profession, in the mind of the architect.

5. Conclusion

This research has illustrated that the deficiencies within traditional methods of construction documentation can be addressed through the appropriate application of digital technology. In order to do so, it is necessary to distinguish construction documents not as drawings and specifications but rather as simply information, in the broad sense of the word. Reformulating the definition of this documentation allows the introduction of concepts and strategies that have been successfully employed in disciplines other than architecture.

The prototypes that were developed through the course of this research served as a means to test ideas regarding the process and visualization of construction information. They represent a demonstration of the potential for further application and development. While there are certainly issues within each that need further resolution, the intention was to illustrate the potential application of the illustrated concepts.

The overriding objective of this thesis has been to develop tools and strategies that allow architects to work more effectively through the appropriate application of digital technology. As was indicated earlier in this paper, current methods of working do not take advantage of the technology in the most suitable manner. What is needed is a rethinking of the processes used and an investigation of alternative methods of working. Looking outside the discipline of architecture for solutions to similar problems instead of simply adopting current methods is necessary in order to realize the benefits that digital technology can bring.

6. References

Belge M, Lokuge I, and Rivers D. "Back to the future: a graphical layering system inspired by transparent paper". in Proceedings of the ACM SIGCHI '93 Conference

Campbell, Dace A. "Architectural Construction Documents on the Web: VRML as a Case Study" in *Digital Design Studios: Do Computers Make a Difference?*, ed. T. Seebohm and S. Van Wyk, ACADIA, 1998, pp. 266-275

Cooper, Muriel. *Design Quarterly* (142)

Cox, Brad J. and Novobilski, Andrew J. Object-Oriented Programming: An Evolutionary Approach. (Reading, MA: Addison-Wesley, 1991)

Crawshaw, D. T. and Daltry, C. D. *Working Drawings in Use.* (London: Building Research Establishment, 18, 1973)

Duyshart, Bruce. *Digital Document* (Architectural Press, 1997)

Evans, Robin. *Translations from Drawing to Building.* (Cambridge, MA: MIT Press, 1997)

Henderson, Kathyrn. *On Line and On Paper: Visual Representations, Visual Culture, and Computer Graphics in Design Engineering.* (Cambridge, MA: MIT Press, 1999)

Johnson, Steven. *Interface Culture: How Technology Transforms the Way We Create and Communicate.* (San Francisco: HaperEdge, 1997)

Khoshafian, Setrag and Abnous, Razmik. *Object Orientation.* (New York: John Wiley and Sons, Inc., 1995)

Lieberman, Henry. "Powers of Ten Thousand: Navigating in Large Information Spaces". from ACM Symposium on User Interface Software Technology [UIST-94], Marina del Rey, CA, November 1994.

McCullough, Malcolm. *Abstracting Craft.* (Cambridge, MA: The MIT Press, 1996)

Mitchell, William and McCullough, Malcolm. *Digital Design Media.* (New York: Van Nostrand Reinhold, 1995)

Nanard M, Nanard J, and Kuhn P. "Pushing Reuse in Hypermedia Design: Golden Rules, Design Patterns and Constructive Templates". from the Ninth ACM Conference on Hypertext and Hypermedia (HT'98); June, 1998, Pittsburgh, PA.

Norman, Donald. *The Design of Everyday Things.* (New York: DoubleDay/Currency, 1988)

Rawlins, Gregory J. E. *Slaves of the Machine: The Quickening of Computer Technology.* (Cambridge, MA: MIT Press, 1997)

Rekimoto, J and Green, M. "The Information Cube: Using Transparencty in 3D Information Visualization". in Proceedings of the Third Annual Workshop on Information Technologies & Systems (WITS'93), pp. 125-132, 1993

Robbins, Edward. *Why Architects Draw.* (Cambridge, MA: MIT Press, 1994)

Robertson G.G., Mackinlay J. and Card S. "Cone Trees: Animated 3D Visualizations of Hierarchical Information". in Proceedings of the ACM SIGCHI '91 Conference on Human Factors in Computing Systems, pp. 189 - 194, New Orleans, LA, USA, April 1991

Schilling, Terrance G. and Patricia M. *Intelligent Drawings.* (New York: McGraw Hill, 1987)

Schroeder, Will, Martin, Ken, and Lorensen, Bill. *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics.* (Upper Saddle River, NJ: Prentice Hall PTR, 1998)

Styles, Keith. *Working Drawings Handbook.* (Butterworth Architecture, 1995)

Talty, Peter. "Working Drawings: Clarity for Construction". in *Classicist* 1995-1996, n. 2, p. 81-86

Tufte, Edward R. *Envisioning Information.* (Chesire, CT: Graphics Press, 1990)

Turnbull, David. "The Ad-Hoc Collective Work of Building Gothic Cathederals with Templates, String, and Geometry." in *Science, Technology, and Human Values* 18(3), p. 315-340.