A Framework for Modeling and Communicating Abstractions of Constructed Facilities

M. K. Zamanian, SJ. Fenves

EDRC 12-41-91

A Framework for Modeling and Communicating Abstractions of Constructed Facilities[1]

M. Kiumarse Zamanian
Steven J. Fenves

Department of Civil Engineering and
Engineering Design Research Center
Carnegie Mellon University
Pittsburgh, PA 15213-3890 USA

Submitted to CAAD futures '91, ETH Zürich,

This work has been supported by the Engineering Design Research Center, a NSF Engineering Research Center at Carnegie Mellon University.

Abstract

Management of information about constructed facilities in a computer-integrated environment is a challenging task because this information evolves from and is viewed by many different disciplines throughout the facility's lifecycle. We present a general framework for modeling and reasoning about the components of a constructed facility at any desired level of abstraction, and communicating the information across disciplines at any stage in the lifecycle of the facility, as well as across stages. Our research has been motivated by an objective similar to that of STEP, which intends to establish an international protocol for the exchange of CAD data. The descriptive information about a facility is divided into two separate but linked groups: spatial and non-spatial attributes. The primary emphasis of this research is to provide a single, uniform representation and reasoning paradigm for dealing with the various spatial abstractions of the facility components regardless of their geometric dimensionalities.

1 Introduction

The domain of constructed facilities constitutes a large part of the Architecture Engineering and Construction (AEC) profession. This domain involves many different disciplines, e.g., architecture, structural, mechanical and electrical engineering. Each discipline has its own view of the facility and uses different levels of detail and representation schemes for modeling the facility components during the various phases of the facility's lifecycle. As a result, a large number of specialized computer programs have been developed for drafting, design, visualization, analysis, construction management, cost estimation, and facility management, each with its own specialized representation of the facility. Apart from IGES[1] [1], which is widely used for exchange of computer-generated drawing data and the recent international efforts for the development of the IGES's successor STEP[2] [2], very little work has been done on defining a general interchange format for the domain of constructed facilities. While some users have decided to purchase only the products of a specific vendor, such as AutoCAD with its DXF protocol [3], others have been forced to use the low-level drafting primitives of the IGES standard to deal with the issues of data exchange among different CAD/CAE programs. Consequently, the problem of "intelligent** exchange of facility information between different CAD/CAE program still remains a challenge to researchers in industry and academia.

The desire of the CAD/CAE profession to move from "islands of automation" to massive integration, the trend towards a "paperless** office environment, the recent advances in information modeling and management, and the successful use of high-level information exchange protocols in other domains, such as electronics [4,5], are the motivations for our present effort in developing a framework for modeling and communicating facility information in a computer-integrated environment. Because the spatial information of facility components, i.e., their geometric attributes and topological relationships, constitute the most important and often most challenging type of information to be dealt with, this work primarily concentrates on the issues of representing and reasoning about the spatial information of the facility components. The non-spatial information of the facility components and their linkage to the corresponding spatial information are also addressed; however, due to extensive efforts of other researchers in dealing with the non-spatial information, we have limited our efforts in this area to only issues of direct relevance to our overall framework.

2 Proposed Approach

All existing data exchange standards use a file for communicating information from one CAD/CAE system to another. The contents of this file are respectively written and retrieved by special-purpose pre- and post-processing modules of the CAD/CAE system based on a specific format established by its corresponding standard committee. The level of sophistication of this format and the type of data it can describe varies greatly from one standard to another. For example IGES [1] has

initial Graphic Exchange Standard

[2]STandard for Exchange of Product model data

a very primitive flat format with no high-level constructs, while VHDL [5] provides a high-level data definition language suitable for expressing various characteristics of complex integrated circuits. Regardless of how sophisticated and elaborate the format of the data exchange file is, the pre- and post-processing modules of the CAD/CAE programs must still directly deal with the raw information in the exchanged file. This approach exposes detailed data to all users and therefore prevents data encapsulation on the one hand, and severely restricts future modifications and extensions to the format on the other. Furthermore, storing and retrieving certain types of information, such as complex geometric models, into and from a file requires much more knowledge than simply the format of how the data is stored in a file; therefore, this approach is not particularly suitable for exchanging high-level spatial information.

In our approach, we propose to provide a standard functional interface as the protocol for defining and retrieving various types and levels of information about constructed facilities. This approach is similar in nature to standard languages, such as SQL [6], defined for interfacing with relational databases; however, this interface and its underlying data model and programs that manipulate the information are specifically designed and selected for the particular domain of constructed facilities. A similar but somewhat general architecture has been proposed by Howard [7] for interfacing databases and knowledge-based systems in the structural engineering domain. However, to the best of our knowledge, to this date the uniform and consistent modeling and communication of the information pertaining to the domain of constructed facilities, specifically the geometric data and the topological relations, in a computer-integrated environment has not been exclusively addressed by any other research group.

Figure 2 presents the overall structure of our approach. In this figure we define several functional views for a constructed facility (e.g., architectural, structural, electrical and mechanical for the design stage), where each view consists of a set of common and discipline-specific elements, their spatial attributes, and the discipline-specific non-spatial attributes of the elements used in that view. The spatial information is mapped onto a conceptual schema representing the union of all spatial information. Thus the system can, in principle, respond to queries such as: "find all architectural rooms supported by at least one structural beam whose grade of steel is the same as that of the structural column located at the north-west corner of the HVAC zone K3." The spatial attributes of the facility components are handled by a geometric modeling system, while the non-spatial attributes are dealt with in a separate database management system (in our case a relational database). A client program, designed and implemented specifically for storing and retrieving the facility information, interacts with the geometric modeler and the database management server programs and therefore encapsulates the underlying representation schemes and the functional behaviors of the two server programs. This client program consequently provides a functional interface with which users can specify new information and query existing information about a facility to be respectively stored in and retrieved from the server programs. Each of the server programs, as well as the client program, has its own data representation scheme, and it is the client program's responsibility to coordinate the translation between these different representation schemes. Formal specification of the functional interface to this client program in turn establishes the proposed information exchange protocol for constructed facilities.

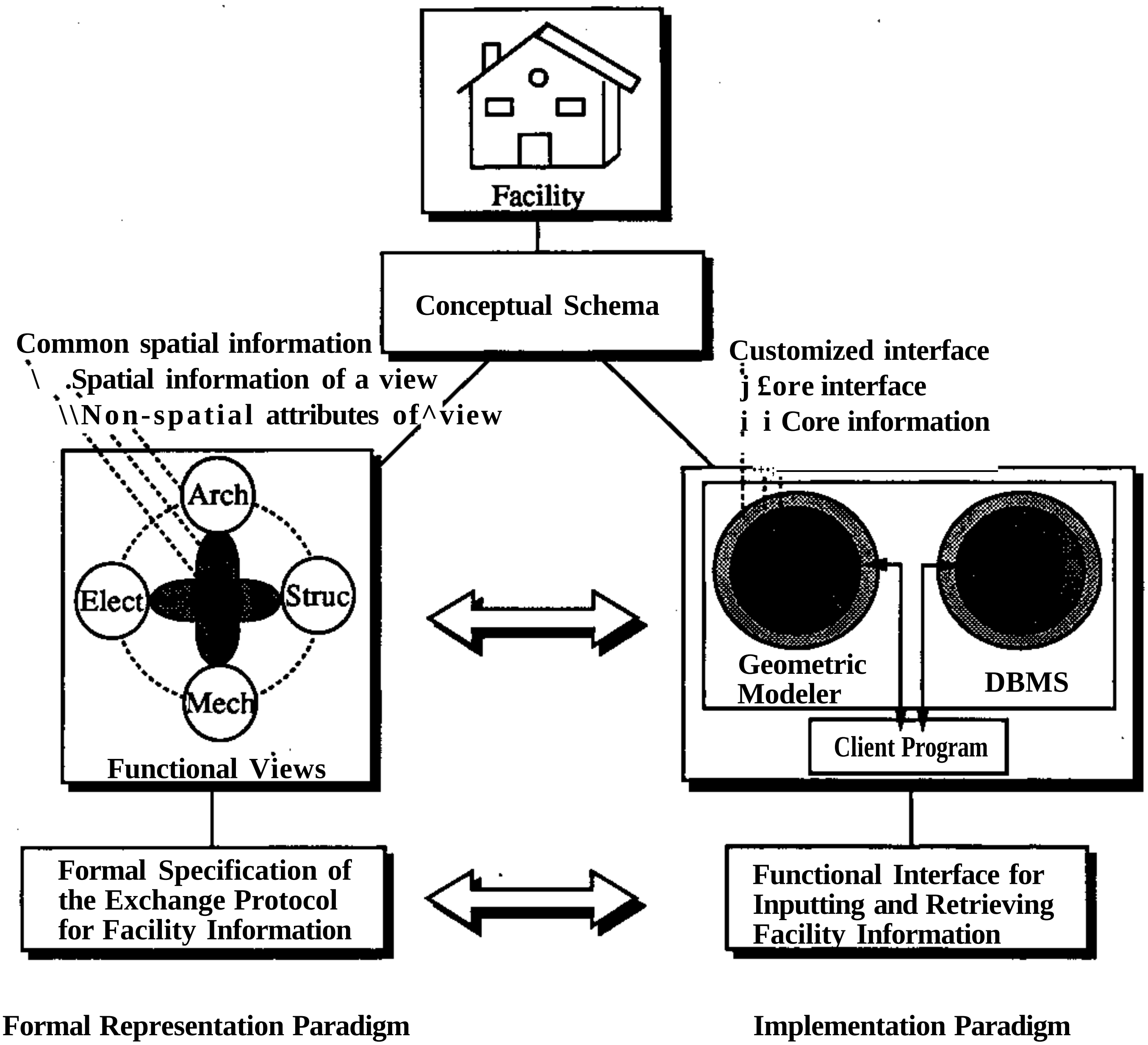


Figure 1: A Framework for Modeling and Communication of Facility Data

3 Organization and Representation of Facility Information

Constructed facilities can be thought of as assemblies of components that are put together to perform specific functions for a variety of disciplines, such as architecture and structural engineering. An important characteristic of these components is that they can be represented as abstractions of data that demonstrate different behaviors to different applications of a facility. These abstractions can in turn be classified, based on their behaviors and specific attributes, into various types. Although these types and abstractions are not as well defined and distinct as those of assemblies used in other domains, such as electrical circuits, they provide a basis for establishing the formal information models of constructed facilities.

A majority of the existing information models use a hierarchical scheme for organizing the components of a facility [8, 9, 10]. Although a hierarchical scheme is suitable for the high-level decomposition of a facility, it fails to model the complex relationships between the low-level components. Therefore, this work proposes to use a more general organization scheme that combines a hierarchical structure for the high-level components of the facility with a network representation that models the decomposition at each level of this hierarchy [11,12].

In addition to the organization of a facility, formal representation of various attributes of each component of the facility is an important information modeling issue. Two major types of information must be dealt with in modeling the components of a facility: descriptive that consists of spatial and non-spatial attributes, and functional [11]. This paper deals solely with the descriptive type of information. The spatial attributes pertain to the component's geometry and topology (e.g, a room is a cuboid of certain dimensions adjacent to a set of walls, floors, and ceilings), while the non-spatial attributes are all the other properties of the component (e.g., the thermal conductivity of a wall, the color of a face of the wall, etc.). Furthermore, different disciplines may use different abstractions, aggregations, and spatial subdivisions of the components of interest. Every component of a facility, physical or abstract, is in turn a collection of these types of information and is represented by an "object" [11, 12] in the information model. An object provides an abstraction for a component and can be viewed differently by various disciplines. For example, a wall may be viewed as a partitioning element by the architect, as a load bearing component by the structural engineer, or as a routing agent by the mechanical or electrical designer. The geometric representations associated with each object also depend on the level detail and type of information required by a specific discipline at a particular phase of the facility lifecycle. The geometric representations therefore may vary from simple wireframe to complex solid models. Dealing with the mixed-dimensional geometric abstractions and the topological relationships of the facility components are of particular interest in this work and are discussed extensively in the next section.

3.1 Facility Spatial Information

While the majority of the non-spatial attributes of facility components can be represented using relational databases or other traditional data modeling techniques, other, more specialized representation schemes are needed for dealing with the spatial attributes. Presently, many CAD/CAE systems provide a geometric modeling system that offers a variety of representation schemes, such as wire-frame, surface, and/or solid modeling; however, these schemes are generally disjoint and have entirely different internal data structures and algorithms. This limitation prevents users from modeling and reasoning about various spatial abstractions of a facility in a single, uniform framework. For example, a wall may be represented as a line during the preliminary design phase and by a planar surface or a cuboid for the detailed design or analysis phase. Representing these different abstractions in a single geometric modeling paradigm is presently not supported by any AEC CAD/CAE system, primarily due to the lack of support for the more recently developed non-manifold geometric modeling techniques [13]. Our research has heavily utilized this new representation paradigm, specifically the vertex-based, non-manifold geometric modeling system, NOODLES, developed by Gursoz [14].

Several major issues have been identified in our research for dealing with the spatial information of constructed facilities; these include:

- overall spatial organization,
- spatial decomposition and aggregation,
- compositional polymorphism,
- topological polymorphism, and
- reasoning about implicit topological relationships.

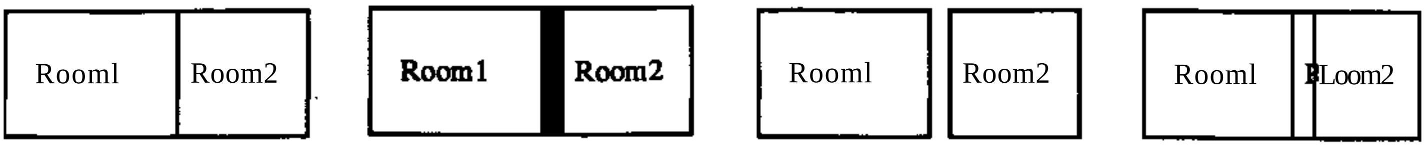
These items are briefly described in the following paragraphs:

Overall spatial organization. The organization of components (or objects in a general data modeling sense) in a constructed facility can be conceptually viewed as an acyclic graph structure whose nodes correspond to the objects and whose links describe the spatial relationship between related objects [15]. As an extreme example, a fire zone can be linked to a HVAC duct via the "contains" relationship. This graph structure can potentially contain all the spatial information about all the facility components; however, it is too general and contains too much data (objects and relationships) for any one particular discipline to deal with. Therefore, it is necessary to provide each discipline with its own, more specialized organizational scheme of the components while maintaining consistency between these discipline-specific models and the primary model. Consistency is achieved by making each specific spatial model a discipline-specific (or functional) "view" of the primary (or conceptual) model, in the same sense as used in database management systems. For example, architects may wish to deal with spaces such as floors which contain suites which in turn consist of rooms that are separated by walls. This organization of spaces is simply a specialized view of the primary model containing the spatial information about all the spaces in the facility, which may include rooms, pipes, walls, beams, etc.

Spatial decomposition and aggregation. Although different disciplines have their own special arrangements of objects in a constructed facility, in most cases they deal with the spatial decomposition of the facility in a hierarchical fashion, i.e., given a particular object, they subdivide that object into smaller subobjects by defining either partitions or individual subobjects, or both. This process is in fact initiated by modeling the constructed facility itself, either by defining its bounding envelope or directly via its topology and geometry, where the three-dimensional physical universe is partitioned into objects corresponding to the "inside" and the "outside" of the facility. Once this "inside" object is defined, it can be recursively decomposed into disjoint subobjects until the desired level of granularity has been reached for any particular discipline. During this process, it is also possible to decompose a union of several objects instead of limiting the decomposition technique to the leaf nodes of the hierarchical tree. It is important to note that although the decomposition methodology described here is the same for all disciplines, it does *not* imply that the level of decomposition or the arrangement of objects must also the same. In other words, each discipline is free to arrange its own spatial organization; meanwhile, it can also use the objects generated by other disciplines for defining its own objects.

Compositional polymorphism. Based on the above discussion, one of the major objectives of this work is to allow any discipline to define and retrieve the spatial information of the components of a constructed facility at any desired level of abstraction or detail. By using a non-manifold geometric modeling paradigm, representations of different dimensionality can harmoniously coexist in the same model. Furthermore, an application can exploit any level of spatial composition of the objects, via the proposed functional interface of the client program, regardless of the level of dimensionality used for representing those objects. One of the main advantages of this approach is that objects of lower-level dimensionality can generally be extended to higher-levels of dimensionality by producing their approximate enclosing envelopes from the available geometric attributes. For example, the line representation of a girder, often used in simple structural analysis programs, can be extruded to a two-dimensional rectangle by using the specified height of the girder and the direction of extrusion. The rectangular representation of the girder can in turn be used for interference checking with distribution elements, such as pipes or ducts. Furthermore, the girder stiffeners can be represented as nodes on the one-dimensional, or line segments on the two-dimensional, representation of the girder, respectively. Finally, the girder flanges can be represented as additional two-dimensional rectangles, or the entire girder can be represented as a three-dimensional solid. Thus, depending on the level of approximation at which the spatial information is available or needed, various abstractions with different dimensionalities of the facility components can be composed.

Topological polymorphism. The representation of a spatial abstraction of an object is merely a mathematical approximation of the actual geometry of that object. The degree of approximation used for representing a particular object is generally determined by the level of detail needed for dealing with that object. Furthermore, in order to be able to retrieve the desired topological relations between spatial abstractions of different dimensionality, it is necessary to make these



a) touching b) touching a common wall c) not intersecting d) overlapping

Figure 2: Topological Relationships of two Rooms

relations invariant with respect to the dimensionality of geometric entities used to represent those abstractions. For example, one must be able to retrieve topological relations, such as intersection or part-of, between an HVAC piping network that is represented by either a connected set of one-dimensional lines or a set of solid objects and the structural floor systems that are modeled by either planar rectangles or three-dimensional cuboids. The use of a non-manifold geometric modeling scheme addresses the topological polymorphism issue when one deals only with the explicit topological relations between the entities of the geometric model. Handling the implicit topological relationships that are perceived by various disciplines at higher levels of abstractions than those represented in the geometric model is a much more complex task and is somewhat domain dependent. This issue is discussed next.

Reasoning about implicit topological relationships. The explicit topological relationships of the geometric representations of the facility abstractions are readily available in the non-manifold geometric modeling paradigm used. However, most disciplines often perceive topological relationships, e.g., "connected-to", "next-to" or "adjacent-to", that differ from the explicit topological relationships represented in the geometric model of the facility. For example, consider the topological relationships of two rooms illustrated in Figure 2. For simplicity, the two rooms are represented by two rectangles in all four cases, since the dimensionality and type of geometric entities used are not important. The two rooms in all cases of this example are perceived to be adjacent or next to each other by the architect or the HVAC designer; however, it is only in case (a) that the two corresponding rectangles explicitly share one edge and thus are said to be explicitly adjacent. For the remaining three cases, there is simply not enough information in the geometric model to directly provide this implicit topological relationship. To resolve such implicit topological relationships, we first proposed to augment the spatial operators of the geometric modeler to construct a tolerance envelope around the desired geometric entities based on some user defined tolerance value [15]. After further studying the problem, we concluded that the tolerance envelope approach is only appropriate for very simple cases and becomes extremely complicated in more general cases. Therefore, we decided to deal with this problem at a higher level of abstraction than that provided in the non-manifold paradigm, and subsequently devised a new representation scheme on top of a non-manifold paradigm for representing and reasoning about various abstractions of constructed facilities regardless of the dimensionality and type of geometric entities used. This representation scheme is described next.

3.2 A General Spatial Representation Scheme

The need for reasoning about implicit topological relations and for a high-level symbolic language to define and query the spatial information of constructed facilities motivated the development of this general spatial representation scheme. Some existing geometric representation techniques used in the architectural and the structural engineering domains [16,17,18] have been studied throughout this work, but none of these techniques provides a sufficiently general scheme for dealing with the mixed-dimensional spatial abstractions of facilities in a single, uniform fashion. This section provides a formal description of this representation scheme and discusses its relationship to the non-manifold representation scheme used as its implementation basis.

Two types of spatial representation schemes are identified in this work: a minimal representation scheme corresponding to that of the the vertex-based, non-manifold geometric modeler adopted, and a maximal representation scheme developed in this work. The minimal scheme is based on a boundary representation scheme that uses four disjoint, atomic geometric elements, i.e., vertex, edge, face, and solid, and a number of internal topological elements, e.g., shell, loop, etc., for modeling geometric models with possible "dangling" elements while guaranteeing closure under all Boolean operations. The maximal scheme uses special geometric entities referred to as "superior elements," such as planes, lines, or points, to define half-spaces of appropriate dimensionality with which any spatial subset of a spatial configuration is represented. The two representation schemes are consistent [19]; however, there is only a unique mapping from the maximal representation to the minimal representation, and not *vice versa*[3]. Although the superior-element representation scheme can theoretically handle non-linear configurations, due to limitations of the existing geometric modeling techniques [13] our research is limited to linear spatial configurations. This limitation is not severe, because the majority of constructed facilities have linear geometry and their curved segments can be accurately approximated as piece-wise linear segments.

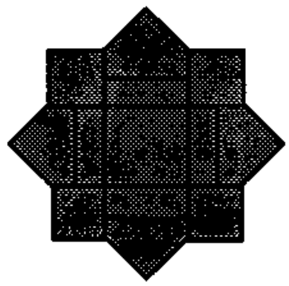
An Ai-dimensional ($1 \leq n \leq 3$) spatial configuration S is a subset of E^n and can consist of various m-dimensional ($0 \leq m \leq n$) spatial subsets. In the 2-dimensional spatial configuration illustrated in Figure 3a, the spatial subsets correspond to all possible two-dimensional spaces (spaces are n-dimensional) and the one- and zero-dimensional partitions separating those spaces. The set A, consisting of disjoint, atomic elements corresponding to the minimal representation of this configuration, is shown in Figure 3b, and set the J5, shown in Figure 3c, contains the superior lines (superior-elements are $n - 1$-dimensional). Figure 3d illustrates two spatial subsets of this spatial configuration and their corresponding minimal and maximal representations. It is important to note that the concave[4] subsets of a spatial configuration can not be uniquely represented by the superior-element scheme but are representable by applying Boolean operations on convex subsets, which in turn are uniquely represented by the maximal scheme.

The superior-element representation scheme can be formally described by the following syntax:

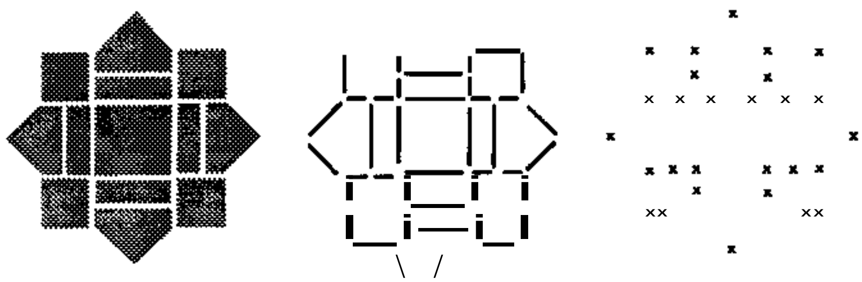
$$S \;\; \_\!=\! A$$

[3]This is similar to the relation between CSG and B-Rep solid models[19]

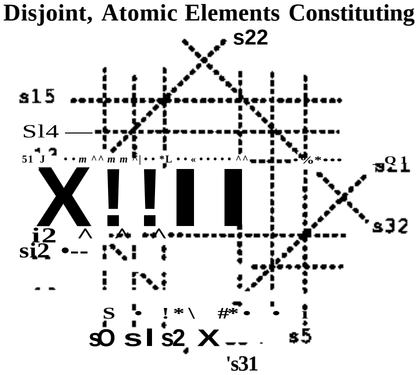
[4]A concave geometric entity has internal angles greater than 180 degrees and possibly cavities or dangling parts of same or lower dimensions.



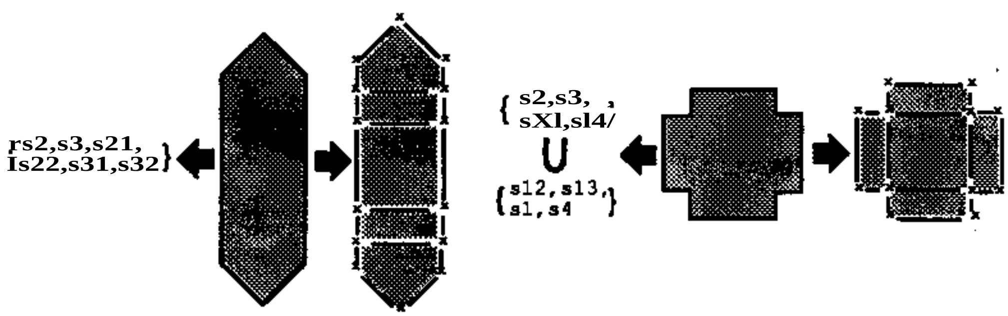
a) Spatial Configuration S



b) Disjoint, Atomic Elements Constituting *A*



c) Superior Elements Constituting *B*



d) Representation of Convex and Concave Spaces

Figure 3: An Example of a Spatial Configuration

$$
\begin{aligned}
S &::= (e_1, \ldots, e_n) \\
e_i &::= (C(b_i, \ldots, b_m)) \mid (BOOL(e_j e_k)) \\
C &::= NULL \mid E \mid b_i \mid (b_i.b_j) \\
BOOL &::= \text{UNION} \mid \text{INTERSECTION} \mid \text{DIFFERENCE} \\
b_i &\in B
\end{aligned}
$$

Here«Sisart-dimensional(l $\leq n \leq$ 3) spatial configuration that is comprised of various possible m-dimensional (0 $\leq m \leq n$) spatial subsets ft and its minimal representation is the set *A*. Each convex e_i is defined by a m-dimensional "carrier"[5] model *C* that spatially contains e_i and a list of superior elements that uniquely contain all the boundaries of e_i. C, depending on the dimensionality and type of ft, can be *NULL* (when m = 0), the Euclidean space £7", a superior element, or the intersection of two superior elements. Each concave spatial subset is in turn defined by a Boolean operation and two other spatial subsets (concave or convex). Finally, the n — 1-dimensional superior elements b_i corresponding to *S* are members of a set *B* specified by the user.

3.3 Spatial Decomposition, Selection, Composition, and View Creation

As discussed earlier, constructed facilities can be modeled as assemblies of various objects that represent different abstractions of the facility components (e.g., rooms, zones, girders, and pipes). In the top-down definition of these objects, objects corresponding to higher abstractions are recursively decomposed to create lower-level objects of desired abstractions. Similarly, for the bottom-up definition of the objects, lower-level objects are grouped to create the high-level abstractions needed for other objects. Furthermore, it is often desirable to create views by collecting information from one or more facility objects. These issues are extensively addressed in some of the recent data models of constructed facilities [20, 12, 2, 11, 9, 21] mostly for dealing with the non-spatial attributes of the facility components. In this work, we use our proposed superior-element representation scheme as the basis for specifically addressing the management of the facility spatial information. The spatial operations proposed in this work are closely related to similar operations used in database management systems for developing appropriate taxonomies and groupings of information. Our initial findings suggest that there exist parallel hierarchical structures in the geometric model and the database organization of a constructed facility.

Decomposition. Based on the above description of the proposed maximal representation scheme, a recursive spatial decomposition technique is developed. This technique is formally described as follows using the above nomenclature:

$$
\begin{aligned}
S_i &\subset d \\
d &= E \mid a(i-_m)k \mid \{a(i-_m)k \cdots \text{fl}(i-_n)\}(1 \leq m,n \leq i) \\
S_i &= A_i
\end{aligned}
$$

[5]The notion of "carrier" is borrowed from [13]

fly € **A,-**

dij ::= vertex | edge | face | solid

bij € ***Bi***

Ci®Bi =* ***Ai***

This formalism basically states that a spatial configuration *Si* (with *So* representing the starting configuration) is created by partitioning its carrier model C*;* with a set of superior elements *bij* in *Bi* via the © operator. The carrier model is equivalent to either the Euclidean space (often used to initiate the decomposition process), an atomic element, or a composition of the atomic elements created in the previous decompositions, i.e., from Aj_„(i ≤«≤*)• This recursive process is continued with a new set of superior elements, which may have a different dimensionality than the dimensionalities of sets used in the previous steps, until the desired level of spatial granularity is reached.

Selection. Specific atomic elements created during the decomposition process can be selected based on some given criteria. Here we define a binary selection operator © and its two operands as sets *Ai* and *Dut* corresponding to the specified sets of atomic elements and selection criteria, respectively. The result of a selection operation, A*, is consequently a subset of A/, i.e.,

$$A_{ik} = A_i \ominus D_{ik}$$

Note that the second subscript in the above equation indicates the possibility of multiple selections from the same set of atomic elements. The © function is similar to the "select" operation used in relational databases, with A*;* being analogous to a relational table and *Dut* representing the constraints imposed on the attributes of selected tuples.

The elements in *Dut* can be of three types:

1. geometric class of elements in A,-,
2. superior elements in *B&* defining the spatial extent of A*, and
3. a list of labels of labeled elements in A,-.

The selection criteria in *D&* are combined through the logical connective "AND" in order to satisfy each criteria and return only those selected elements that satisfy all the given criteria.

Composition. Selected atomic elements generated via the decomposition process can be put together to compose new subsets of their corresponding spatial configuration. The composition operation, indicated by the binary operator ®, therefore creates a new set of atomic elements from two other sets that are generated via the decomposition, selection, or composition operations. Because the atomic elements of any spatial set must be disjoint, overlapping elements from the two sets in a composition operation are split into their corresponding disjoint atomic elements. As shown in Figure 4, the composition operation distinctly preserves the overlap of its operands by creating new elements that can also be represented and identified by the superior elements associated with the composed set.

a) 2 overlapping elements b) composition of 2 elements

Figure 4: Composition of two Overlapping Elements

View creation. Discipline-specific views (or functional views) of a spatial configuration are represented by spatial subsets of that configuration such that those subsets contain only the information needed by their corresponding disciplines. Views are created using the selection and the composition operations described above.

Labeling of spatial elements. Although the superior-element representation scheme can be used for identifying the elements of a spatial configuration and its subsets, this scheme may appear cumbersome and not very efficient. Therefore, a mechanism is provided for attaching user-defined labels to spatial configurations and their elements or subsets when they are created or when later identified by the superior-element representation scheme.

An example usage of the spatial operations. Using the spatial configuration from our previous example in Figure 3, the spatial operations and the labeling mechanism discussed above are used to create this configuration and subsequently several of its spatial subsets as illustrated in Figure 5. A graph structure representing the sequence of operations used to create these configurations is shown in Figure 6.

4 Linkage between Non-spatial and Spatial Information

As briefly mentioned earlier, management of the non-spatial attributes of the facility components in a computer-integrated environment is as important as dealing with the spatial attributes. Much work has been done in the area of data modeling and information management for the domain of constructed facilities, primarily for dealing with the general organization of the facility representation and of the non-spatial attributes [20, 22, 12, 9]. The contribution of the present work is to provide a linkage between the spatial and non-spatial attributes of the facility components while maintaining separate data models and representation schemes for each of the information types. The major advantage of separating the management of the spatial and non-spatial information is that data representation schemes and algorithms that are most appropriate for the particular type of data being considered are used. On the other hand, there is a performance price to pay when data is partitioned across different server programs and has different representation schemes. The proposed system, illustrated in Figure 2, involving separate server programs for the spatial and

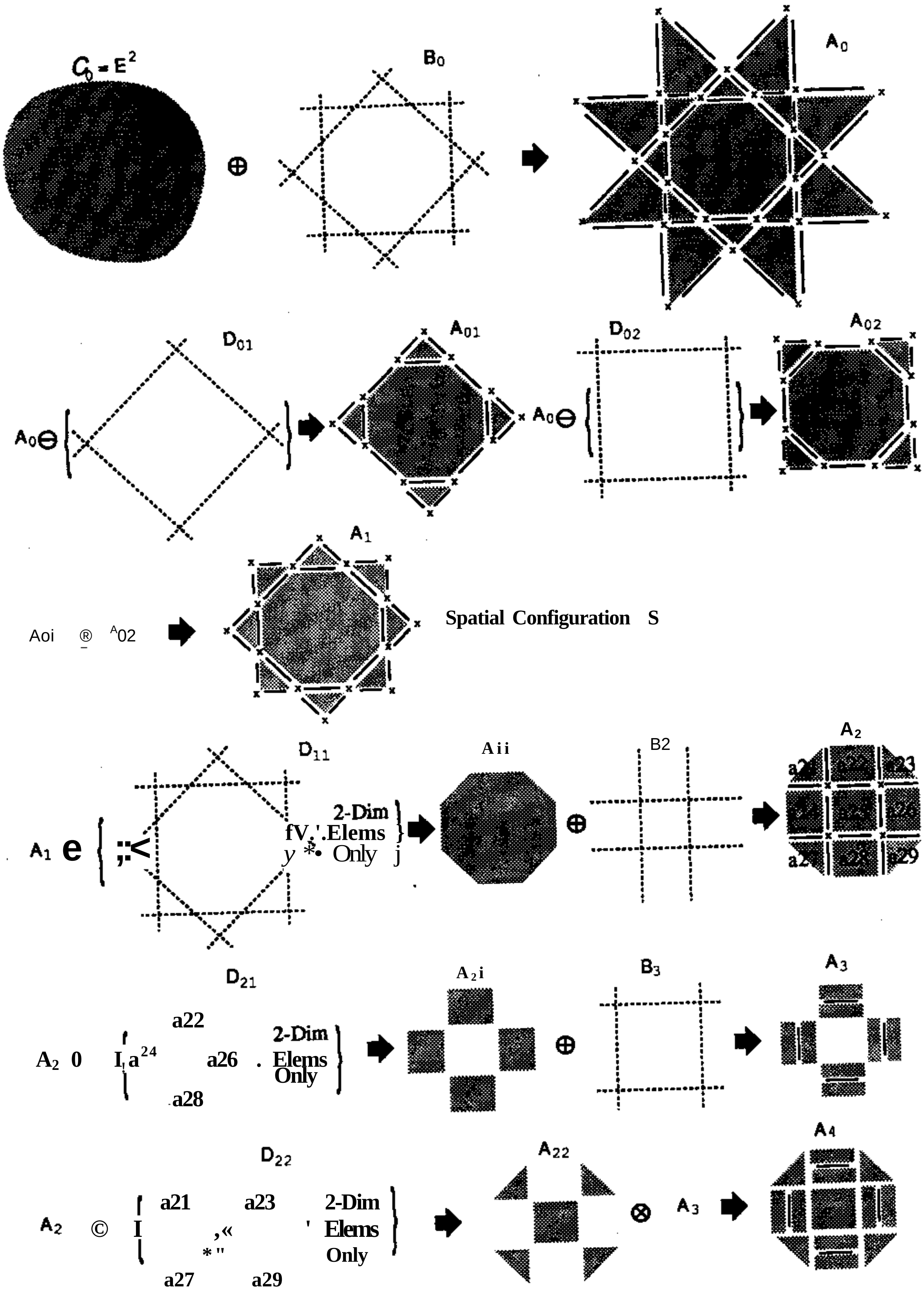


Figure 5: An Example Usage of the Proposed Spatial Operations

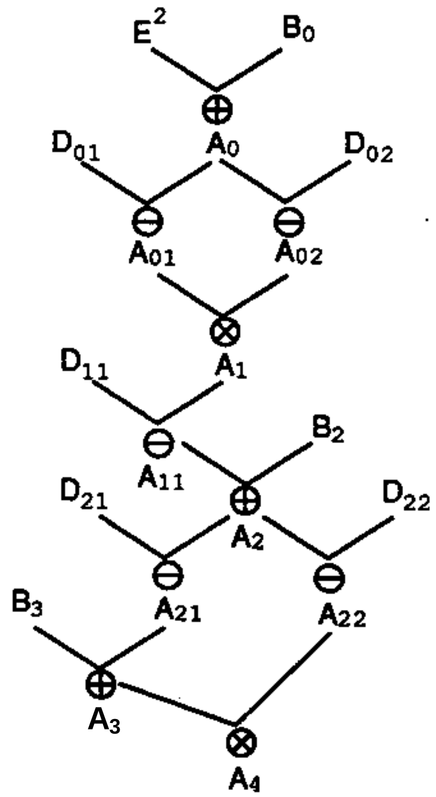


Figure 6: Graph Illustrating the Operations used in the Example

non-spatial information is intended to minimize excessive performance costs by capitalizing on three assumptions:

1. The hierarchical operations of decomposition, selection, composition, and view creation pertain equally to non-spatial and spatial attributes of facility components[6]. In other words, the discipline specific (or functional) organization of non-spatial information follows the same hierarchical decomposition as that of the spatial information.

2. All non-spatial attributes of interest in a view are keyed or indexed to some spatial element in the decomposition hierarchy. The unique identifier attached to each spatial element (user-defined label or system-generated internal identifier) also serves as the key to the non-spatial attributes of that element in every functional view.

3. Unlike the spatial information, which is likely to exhibit commonality across views (at least at the higher levels of abstraction), it is anticipated that most non-spatial attributes will be specific to one functional view only. Therefore, separation of management actually imposes beneficial segregation of the non-spatial attributes by function.

The third point above does not preclude retrieval of non-spatial attributes across functional views; it simply capitalizes on the common spatial information that unites the multiple functional-

[6]For non-spatial attributes, composition is limited to assemblies of disjoint (non-overlapping) elements.

ities of facility components. To illustrate this point, let us consider the hypothetical query used in Section 2, i.e., "find all architectural rooms supported by at least one structural beam whose grade of steel is the same as that of the structural column located at the north-west corner of the HVAC zone K3." This query can be broken into several subqueries issued to the geometric modeler and the database manager as follows:

1. **locate north-west corner of zone K3 in the HVAC view (spatial);**
2. **find a column in the structural view at the location found in 1 (spatial);**
3. **retrieve the grade of steel for the column found in 2 (non-spatial);**
4. **retrieve all beams in the structural view with grade equal to the value found in 3 (non-spatial); and**
5. **find all rooms in the architectural view such that each room is supported at least by one of the beams found in 4 (spatial).**

Implementation. **The current partial implementation of the proposed framework shown in Figure 2 links the SYBASE relational database [23] with the NOODLES non-manifold geometric modeler [14]. A menu-driven prototype client program with interactive graphic viewing capability has been developed in order to define the facility components by mixed-dimensional geometry of any desired abstraction, and optionally to attach non-spatial attributes to these components. The linkage to the relational database enables users to automatically transfer data from or to the database via the client program without the detailed knowledge about the underlying table organizations or the SQL commands issued. Every component is uniquely identified by its user-defined name and functional view. Furthermore, a unique internal identification is attached to every component once it is defined, and the spatial attributes (represented in terms of the geometric entities of NOODLES) and the non-spatial attributes (represented in terms of the tuples in relational tables) of a component are linked together via this internal identification. The client program encapsulates the underlying data representations and functional interfaces provided by the geometric modeler and the relational database, thus providing a higher level of abstraction for dealing with the facility information.**

Two of the future facilities to be incorporated in the prototype client program are best illustrated in connection with Figure 5.

1. **The introduction of a "level" attribute as part of the component identification. Thus if both *A2* and *A3* are part of the same functional view, the higher-level elements in A2 can be differentiated from the lower-level elements in A3, and different non-spatial attributes associated with elements of each level.**
2. **The automatic generation of hierarchical, topological, and geometric attributes in the non-spatial database in response to user requests, to facilitate processing by application programs. Examples of these types of attributes are:**

- spaces in *A** that are subdivisions of spaces in A2 (hierarchical);
- end vertices of edges or edges of faces (topological); and
- coordinates of vertices, lengths of edges, areas of faces, or volumes of solids (geometric).

5 Conclusions and Future Work

This paper discusses issues with regards to the representation of mixed-dimensional spatial abstractions of the components of constructed facilities. These issues are of great importance for the development of the proposed framework for modeling and communicating all facility information in a computer-integrated environment. A general representation and identification scheme is developed to deal uniformly with the topological relationships between mixed-dimensional geometric abstractions of components used by different functional views throughout the lifecycle of a constructed facility. This scheme is based on a new maximal boundary representation technique that is provided on top of the minimal boundary representation of the underlying non-manifold geometric modeler. This maximal (or superior-element) representation scheme provides a symbolic way of representing and identifying the implicit topological relationships in a spatial configuration and its subparts consistently and uniformly regardless of the dimensionality of the configuration's geometric entities.

Several issues will be studied closely and possibly implemented in the remaining time frame of this research. These issues include: associating multiple, mixed-dimensional geometric representations to a single component while providing consistent topological relationships between components; expansion of lower-level geometric entities to higher-level ones to deal with the evolution of the facility throughout its lifecycle; development of an object-based client program for managing the interactions between the two server programs used for handling the spatial and non-spatial attributes of the facility; and, formal specification of a data exchange protocol for specifying and retrieving facility information via the functional interface of the object-based client program.

Acknowledgments. This research has been supported under the cooperative agreement number ECD-8943164 by the Engineering Design Research Center, an NSF Engineering Research Center at Carnegie Mellon University.

References

[1] National Bureau of Standards, Gaithersburg, MD. *Initial Graphic Exchange Specification (IGES) Version 3.0*, 1987.

[2] W. Gielingh. General AEC reference model (GARM). Technical report, AEC committee of ISO/STEP, P.O. Box 49, 2600 AA Delft, The Netherlands, 1988.

[3] Autodesk Inc., Sausalito, CA. *AutoCAD Reference Manual,* 1987.

[4] Electronic Industries Association, 2001 Eye Street, N. W., Washington D.C. 20006. *Introduction to EDIF,* September 1988.

[5] R. Lipsett, E. Marschner, and M. Shahdad. VHDL - the language. *IEEE Design and Test,* pages 28-40, April 1986.

[6] C. J. Date. *A Guide to SQL Standard.* Addison-Wesley, Reading, MA, 1988.

[7] H. C. Howard. *Interfacing Databases and Knowledge-based systems for Structural Engineering Applications.* PhD thesis, Carnegine Mellon University, Pittsburgh, PA, 1986.

[8] J. Turner. *AEC Building System Model, ISO TC184/SC4/WG1 doc. 322A.* University of Michigan, Ann Arbor, MI, July 1988.

[9] G. H. Powell and R. Bhateja. Data base design for computer-integrated structural engineering. *Engineering with Computers,* 4:135-143,1988.

[10] A. Lavakare and H. C. Howard. Structural steel framing data model. Technical report, Center for Integrated Facility Engineering - Stanford University, Stanford, CA, June 1989.

[11] W. T. Keirouz. *Domain Modeling of Constructed Facilities for Robot Applications.* PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 1988.

[12] B. Bjork and H. Penttila. A scenario for the development and implementation of a building product model standard. In *Current Research and Development in Integrated Design, Construction and Facility Management,* Stanford, CA, March 1989. CIFE, Stanford University.

[13] C. M. Hoffmann. *Geometric and Solid Modeling: An Introduction.* Morgan Kaufmann, 1989.

[14] E. L. Gursoz, Y. Choi, and R B. Prinz. Vertex-based representation of non-manifold boundaries. In M. J. Wozny, J. U. Turner, and K. Preiss, editors, *Geometric Modeling for Product Engineering,* pages 107-130, New York, September 1990. IFIP WG 5.2/NSF, North-Holland.

[15] M. K. Zamanian and S. J. Fenves. An information exchange protocol for constructed facilities. Technical Report 12-37-90, Engineering Design Research Center (EDRC), Carnegie Mellon University, Pittsburgh, PA, 1990.

[16] S. J. Fenves and S. A. Liadis. A data structure for computer-aided design of buildings. *APEC,* pages 14-18, Fall 1976.

[17] G. Stiny. Introduction to shape and shape grammars. *Environment and Planning B,* 7:343-351, 1980.

[18] R. F. Coyne and U. Flemming. Planning in design synthesis: abstraction-based LOOS. Boston, MA, July 1990. AIENG 90.

[19] A. A. G. Requicha. Representations for rigid solids: Theory, methods, and systems. *Computing Surveys,* 12(4):437-464, December 1980.

[20] S. J. Fenves, U. Hemming, C. Hendrickson, M. L. Maher, and G. Schmitt. Integrated software environment for building design and construction. *CAD,* 22(l):27-36,1990.

[21] K. H. Law, T. Barsalou, and G. Wiederhold. Management of complex structural engineering objects in a relational framework. *Engineering with Computers,* 6:81-92,1990.

[22] C. M. Eastman, A. Bond, and S. Chase. A formal approach for product model information. *Research in Engineering Design,* 1990. (in process).

[23] Sybase, Inc., Emeryville, CA. *Sybase SQL Server Reference Manual,* 1989.