14. Use of Data Modeling in the Conceptual structuring of Design Problems

Charles M. Eastman

Graduate School of Architecture and Urban Planning
University of California,
Los Angeles, California, U.S.A.

An approach is presented for defining the information needed or used in a design task, based on data modeling techniques. Called EDM, it allows representation of the information complexity imposed both from the performances or technologies involved as well as imposed criteria, such as aesthetic intentions. Here, EDM is applied to the design of chairs, a design domain with highly diverse technologies and information structures. The relation is shown between the information considered and the class of designs possible. Also shown is the complexity of different design structures and the implication of information structures for conventional and creative design.

INTRODUCTION

In all fields, design involves the *management of complexity*. The complexity may be *internal*, imposed by performance requirements or the construction technology, as in aerospace vehicle and space station design. Alternatively, the complexity may be imposed *externally*, as a set of design intentions and aesthetic rules, as in certain avant garde architecture. Complexity of a design, as meant here, is different from visual complexity; a visually simple design may have a significant underlying structural complexity. Today, designers can only consider complexity intuitively, in the experience of a particular problem. There has been no way to represent it or measure it.

This recognition suggests a serious shortcoming in existing capabilities for representing design problems. There is a long history of development of representations of *design solutions*, through drawings and specifications, and more recently, through the use of computer tools such as surface and solids modeling. But these representations only depict the resulting form, not the contextual conditions, the expected performances or the intentions imposed upon the design by its designers. Thus they do not depict the design's structure.

There have been some previous attempts to depict the structure of design problems, for example:

- Alexander's pattern language in architecture and urban planning (Alexander, 1977), and
- Karnopp's development of Bond Graphs in mechanical and electrical system design (Karnopp and Rosenberg, 1975).

Neither of these tools has had an impact on professional design. Both have been influential, however, in the education of designers, as early (and recently resurrected) means to explicate the nature of design problems and ways of solving them.

The need for better ways to represent the full structure of design problems has been only a theoretical and pedagogical concern until recently. The development of computer databases for design has led to renewed interest in defining the structure of information about a design. The motivating need is to specify a database structure for use in a particular design context, so that the database can respond to the problem with the same generality and flexibility as paper and pencil.

Ad hoc efforts to develop integrated design databases that cover a variety of functional criteria, support data needed for fabrication (and possibly testing and assembly) and that provide some level of knowledge-base or expert system advice, have been developed in different product domains. These generally have been very expensive to develop, do not respond well to the actual design problems, impose severe restrictions on the scope of design allowed and require constant re-implementation when technological or functional knowledge changes (Encarnacao and Krause, 1982). These problems suggest that better tools, possibly based on formal methods to define the structure of design information, could be extremely helpful in developing more powerful CAD systems.

DATA MODELING

The complexity encountered in the early days of general database research led to the development of data modeling techniques (Jardin, 1974). The intention was to develop very high level languages, close to the application, that could capture the nature of the enterprise being modeled. Several data models have been developed, some out of the database research field such as the entity-relationship model (Chen, 1976) and NIAM (Van Griethuysen 1983), others later out of the expert system, knowledge base and artificial intelligence fields (Peckham and Maryanski, 1988).

Recently, data modeling has been applied to engineering and architecture, as a tool to help address the complexity of data organization for design databases. Most of these efforts have applied data models to capture certain complex semantic issues, such as certain relations between form and function (Batory and Kim, 1985), (Hartzband and Maryanski, 1985). Others have compared the use of different data models in engineering (Shaw et al, 1989). International standards efforts, PDES and STEP, have started defining product models using data modeling, in order to define standards for translation between CAD systems (Smith, 1986). Here, architecture is one product area and some draft product models have been defined (Gielingh, 1988), (Turner, 1988).

Recently, work has been undertaken by the author and his colleagues to develop data modeling techniques specifically tailored to deal with design knowledge. This work is based on the recognition that design knowledge has a different logic and structure from traditional databases. For example, design data is incomplete for most of design, lacks integrity, may carry multiple alternative descriptions and includes procedural as well as declarative knowledge (Bond, Eastman and Chase, 1991). Responding to these distinctions, the author's group developed a new data model, called EDM. The development of EDM is based upon the following goals (Eastman, Bond and Chase, 1991a):

1. representation of function as well as the form and physical properties of the product;

2. support for multiple levels of abstraction and for the various phases of the product life cycle;
3. representation of semantics sufficient for all uses;
4. ability to determine the integrity of information, as needed by a particular application;
5. extensible semantics, supporting the addition of new functional evaluations, new technologies and new conceptual generalizations.

The intention has been to define a small set of structures that capture the semantics of design and engineering information. For defining these structures, we chose as a neutral but powerful meta-language set theory and predicate logic.

The lower level primitives are based on the premises that all design data can be represented as sets of data and relations over the sets. The relations are defined by and come from many sources, including: (i) definitions of measurements, i.e., mass/volume = density; (ii) physical laws, e.g., single occupancy of regions in space-time; (iii) issues of representation, e.g., well-formedness rules for solids modeling or the consistency of indices with the list they refer to; (iv) functional criteria, i.e., rules of composition for pipes, structures or walls; (v) performance measures, such as fluid flow or sound transmission; (vi) dimensional fit, e.g., sizing conditions for pipes and other components that join; (vii) trade practices, such as tolerances for poured concrete, grout spacing, and (viii) design intent, such as dimensional spacing, form relationships and aesthetic criteria (Bond, Eastman and Chase, 1991). We use *constraints* to define all such relations and dependencies between data. Here, a constraint is an expression over a set of data that evaluates to true or false. Our premise is that constraint relations in design vary from being simple to highly complex. It is their aggregate organization, however, that makes design problems especially difficult.

Based on these primitives, EDM provides a small, well-defined set of forms which is used to define a product model. These forms provide various scopes for constraints. Thus they organize the relations between data. We consider the management of these relations to be a central issue in design information management. Our effort has been to package constraints into structures in a form that makes their behavior both apparent and tractable. The forms allow definition of both abstract, conceptual design information as well as low level production based information.

In previous papers, the concepts of EDM have been formally defined (Eastman, Bond and Chase, 1991c), (Bond, Eastman and Chase, 1991) and have been applied to various areas of design knowledge (Eastman, Bond and Chase, 1991b), (Eastman, 1991). These previous papers have laid out the formal foundations for the approach. They also tested in a limited way the range of knowledge EDM is able to depict. In the previous papers, the emphasis was on using EDM for defining design databases.

In this paper, we extend the range of design knowledge being investigated and apply these tools more directly to the issue of design knowledge. The intent is to gain a better understanding of the structure of design knowledge. Making the structure of design information explicit is a powerful step in addressing the intuitive issues of complexity.

In this paper, the EDM data model is applied to a highly unstructured design area, where many diverse creative designs have been developed and where re-formulations have led to highly diverse results. The product area considered here is the design of *chairs*. Chairs respond to a single primary function and have been produced by almost all civilizations.

Given available materials and technology, a great variety of chairs have been designed, some of them ingenious and creative. Today, new designs for chairs evolve primarily from externally imposed structure. Even in such a simple product, the complexity of relations can be quite high.

Chairs are particularly interesting because of the wide range of structure they may incorporate. Chairs may have articulated parts or be of a single molded form. They may rest on a floor or hang from some support. This range challenges most work on design and information structures because of its diversity.

By making the information structures explicit in the design of different classes of chair, the effects of design assumptions can be seen and the limitations of concepts exposed. That is, we can see how different knowledge structures lead to different designs. Also, by investigating data modeling issues in such an unstructured and creative product area, insights into the use of data modeling in more internally complex areas can also be gained. By considering the information re-structuring issues used in previous innovations, clues are offered regarding how future innovations might be developed.

An underlying assumption of this work, not investigated here, is that the information needed to structure and explore designs in an external storage medium is not different from the information used by humans in the intuitive solving of design problems. The knowledge structures developed in the following sections may be relevant for understanding design, whether by man or machine. The processes used to design with, using this information, may be very different, however. Arguments supporting this assumption can be built from work in cognitive science and the psychology of design (Akin, 1986), (Lawson, 1990).

THE EDM STRUCTURES

EDM Elements

Below are the primitive structures from which the higher level forms are constructed.
We start with:

domain(Dname,{Values})

a name and a set of possible values. The value set may be explicitly enumerated (for example, color names) or denoted implicitly by lower and upper bounds over some rational scale (for example, weights). A domain roughly corresponds to the computer language concept of type.
Domains are grouped into aggregations, where:

aggregation(Aname1,{Variable:Dname1 | Variable2 = Aname2})

a named aggregation and a set of variable - domain name pairs (bars (|) denote alternative subsets). A variable may denote a domain or another aggregation. Each variable identifies a particular use of a domain in some context. In traditional parlance, an aggregation is a record or relation type. Relations between variables (in aggregations) are denoted using constraints:

(i) a *constraint definition* has the form:

constraint(Cname,Exp,[Evariables])

where the expression *Exp* is a predicate definition evaluating to true or false. *Evariables* are simply the list of variables occurring in the expression Exp. The *Evariables* occurring in the *Exp* must be uniquely identified.

(ii) a *constraint use* includes the path variable definitions and correspondences to Evariables, in the context of its use.

Ccall ::= Cname[Pvariables]
where Pvariable ::= Fename.Pvariable | Aggname.Pvariable | Variable

A constraint definition is similar to a procedure definition and a constraint call is similar to a call of the procedure.

These primitives, domain, aggregation and constraint, allow us to construct high level forms that package data and constraints, eg. design knowledge, in ways that are easy to interpret and manage.

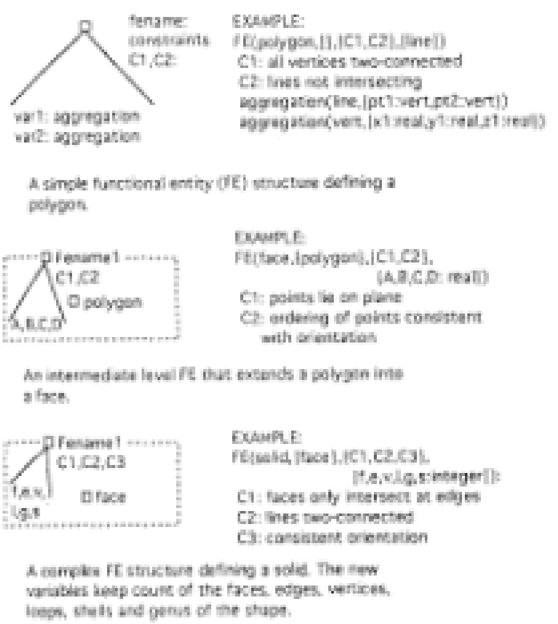


Figure 1. The example FEs, a simple one with only an aggregation and two others that inherit FEs to define a boundary representation solids model.

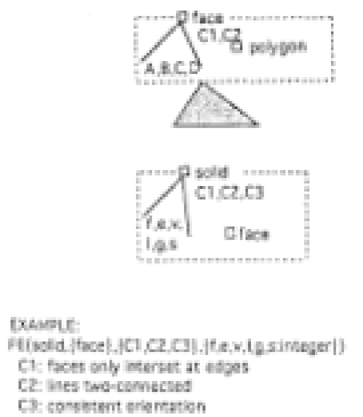


Figure 2. A generalization structure allows one FE to inherit the semantics of another. The filled inverted cone designates that the FE is specialized by another.

EDM Forms

Given the above elements, the four EDM structures are defined below.

Functional Entity

The singular data object within EDM is the Functional Entity (FE). It defines an aggregation and a set of embedded FEs. Constraints apply to variables in both the embedded FEs and aggregation. It has the form:

FE(Fename1,{Fename2},{Cname[Pvariable1]},Aname1)

a composite structure with the name *Fename1* and is made up of a set of FEs, *{Fename2}*, an aggregation *Aname1*, and a set of constraint calls *{Cname[Pvariable1]}*. The *Pvariables* of the FE constraint must be a subset of the aggregation variables of the component FEs or the aggregation.

Figure 1 diagrammatically depicts both a simple FE, consisting only of an aggregation, and its sequential embedding in more complex cases, where the FE is made up of both FEs and variables in an aggregation. Aggregation variables are graphically depicted by the enclosing funnel lines. In the first case, a set of lines satisfying two constraints define a polygon. The constraints require that the polygon is well-formed: that its vertices are all two-connected and its lines non-intersecting. In the second case, the polygons are extended with a surface definition, to define a bounded surface. Constraints here make the polygon lie on the plane and order the polygon points. The bounded surfaces are then arranged into a boundary representation solids model (Mantyla, 1988). The arrangement of faces is restricted by three constraints: allowing faces to only touch at edges, making all edges two-connected, and making faces consistently oriented. In Figure 1, the set *{Fename2}* consists of only one subFE; in other cases there may be several.

Most often, an FE is a set of properties: a shape model or a set of material properties. These are composed into other FEs to define composite definitions corresponding to physical objects. Thus some FEs correspond to physical objects.

Generalization and Specialization

Generalization is broadly recognized as an important form of data abstraction (Shaw et al,1989). In (Bond, Eastman and Chase, 1991), we introduced a new definition of generalization which is a partial order relation between two EDM forms. In general, one form is a generalization of another if it has fewer variables or its constraints allow a greater space of alternatives. The generalization relation is denoted A<<B.

Specialization is the inverse relation to generalization. It is not defined as a separate relation, but is embedded in the FE definition:

FE(Fename1,{Fename2},{Cname[Pvariable1]},Aname1) => Fename1 << Fename2 for all Fename2.

Thus the previously defined FE form is also sufficient for carrying the generalization-specialization relation. Through specialization, the variables and constraints within a FE may come from many different FEs, which have been specialized into a lattice of FE definitions. We graphically depict a generalization as shown in Figure 2. Parent FEs are identified by an inverted filled funnel. The specialized FE carried within another FE, is identified by name within a box denoting the other FE.

Generalization is the means to organize conceptual classes of objects and instances of those objects. Instead of making a strict distinction regarding class and instance, as found in most programming languages, we recognize that in design specifications can be made at any level of specificity and shared across multiple sub-classes or across instances of physical objects.

An intuitive example is shown in Figure 3. Here, a wall opening is defined at the top, possibly with height, width and reference wall as variables. There are three specializations of wall opening: door, window and arch, each with different variables and possibly constraints that specialize them. Windows, for example, have a light transmission coefficient and an area. Doors have a type: pocket, sliding, hinged. These definitions are then specialized further. Glass doors is the merging of both the properties of window and door. Also, no specification is an instance; it can always be specialized by adding new variables or constraints. Thus the window size is partially specified, possibly for one side of the building, then members created from this partially determined size. The intent that all windows on one side of the building are to have the same height is captured by the partial specification. New aggregations or FEs could be added even to base FEs, which have all variables constrained to single values (not shown here). These capabilities are all supported by our definition of generalization-specialization.

Accumulation

Another information structure in design involves the relation between detailed functional information and its more aggregated forms. This relation covers those between disjoint FEs. This structure is:

Accumulation(Accname,Fename1,{Cname1[Pvariable1]},{Fename2},{Cname2},[Pvariable2]})

a relation named *Accname*, defined by a set of constraints *{Cname1}* over *Fename1* and a the set of FEs, *{Fename2}*. The accumulation defines a one-to-many relation between the variables within the FEs defined. In addition, an accumulation includes a constraint set *{Cname2}* that defines the preconditions that the FEs must satisfy. That is, these constraints define the necessary relations between the members of *{Fename2}* if they are to possess the function or intent characterized in the expression in *Cname1*.

Figure 4 diagrams an accumulation. It graphically shows a relation (in other cases the algorithm could be a complex algorithm) over FEs, possibly nested within other FEs. In this example, the relation defines the relation of forces in a set of members incident to a joint and the forces within them.

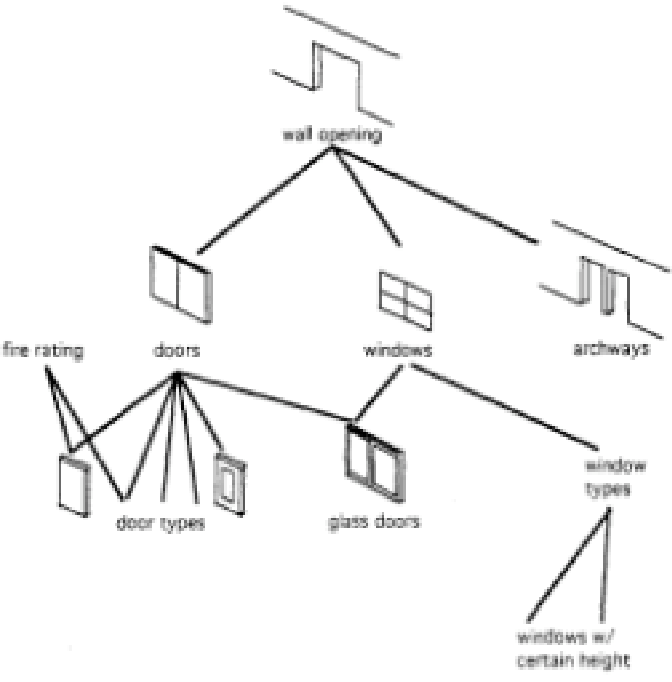


Figure 3. An example generalization-specialization lattice.

This relation must sum to zero for a static truss. The general interpretation of an accumulation is that it defines a one-to-many relationship between disjoint FEs. The relation may involve any type of relation: property accumulation for some particular functioning or performance, for construction, for aesthetics, etc. Analyses are typically defined as accumulations and determine the performance of the set *{Fename2}*; the result is assigned to *Fename1*. Design decisions typically operate top-down, from a single aggregate definition to components.

An accumulation also includes restrictions on allowed FE values, in the form of constraints, that must be satisfied if the functions associated with the accumulation are to be realized. The constraints {Cname2} in the figure define the well-formedness of the structural system, i.e., that they are connected as a graph, with loads only imposed at joints. In other cases, they would define the well-formedness of walls or the size constraints between the parts of an assembly, for example.

Composition

The last product information structure deals with the compositional aspects of products:

composition(Comname,Fename1,{Fename2},{Accname})

a named relation between one FE and a set of FEs that compose it. The composition has the functions or performances specified by the set of accumulations, *{Accname}*. See

Figure 5. A composition is equivalent to a set of accumulations that all apply to the same set of FEs. The set *{Fename2}* can be considered the replacement to *Fename1*, defining it in more detail. The information gain of the composition is that the constraint sets may be reduced to a much smaller set than those defined by the accumulations separately.

In Figure 5, the joint accumulations for all joints are combined with another for shape to define the composition for a whole truss. The shape relation defines the relation between the aggregate shape of the truss and its component parts.

Compositions group a set of accumulations that apply to a common one-to-many structure of FEs. In Figure 5, the parts of the composition are members and joints and the whole is a truss. A composition typically corresponds to what is intuitively thought of as a technology, an assembly with a range of specified performances. Thus a box truss is an alternative composition having similar functions and

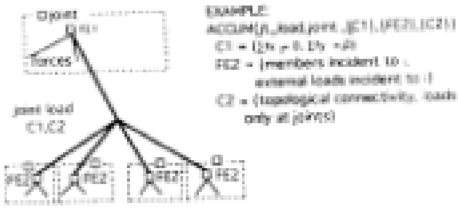


Figure 4. The depiction of an accumulation, which relates disjoint FEs in a one-to-many relation. In the example, the sum of the forces about a joint are derived.

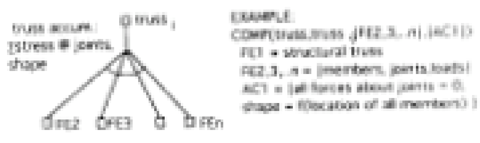


Figure 5. The depiction of a composition, which integrates multiple accumulations.

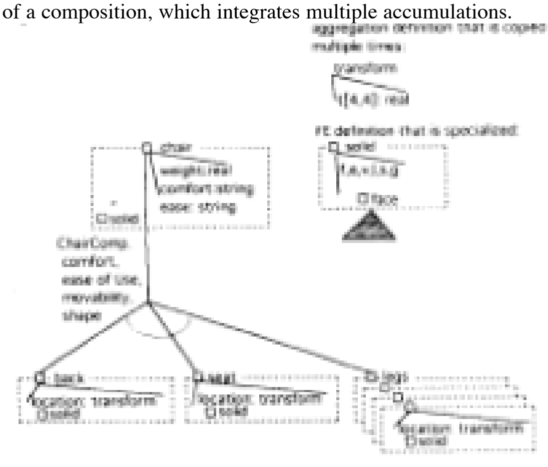


Figure 6. The simple conceptual definition of a chair, based on the dictionary definition.

Thus a box truss is an alternative composition having similar functions and capabilities. A composition can be considered as the efficient encoding of the set of accumulations over a set of FEs. The inverse of composition is part-of (but without accumulations or constraints):

part-of(Ptname,Fename2,Fename1) =>
composition(Comname,Fename1,{Fename2},{Accname})
(for all Poname2)

Composition and part-of have been proposed structures in most other product models (Shaw et al, 1989). Part-of relations, defined separately and without a composition, lose much of their semantic meaning. Thus we always derive Part-of from Composition.

When these forms are used in design, it is assumed that organized arrangements of forms will be pre-defined as knowledge structures that can be retrieved and applied within a general design environment. These then become available as knowledge-bases to be incorporated into particular design projects.

There are at least three different ways in which EDM structures might be used to define a CAD database. Designers may retrieve high level EDM structures that define typical structures used for products and design with them. This use would result in *conventional design*, the most common case. Alternatively, designers may select pre-defined low level structures and arrange them together in a possibly unique higher level structure in which to design. Here, the result may be an *innovative design*, resulting in new arrangements of standard components and functions. For example, existing low level FEs and constraints may be arranged to define new high level FEs. As a third alternative, designers may define new semantic structures using the EDM forms, that is new knowledge not currently within an available knowledge-base, and use this to design with. In this case, the design reflects new knowledge. It is only this case, that I would call *creative design*. These definitions are based upon the information structures used in design and the space or set of designs they imply.

Unique knowledge structures may be defined within EDM for a single design. This is accomplished by modifying existing or adding new EDM forms to define new semantic relations. Thus EDM supports creative design. The decisions regarding how the new information forms link to accumulations and satisfy constraints corresponds to the design issues of determining how functions will be supported and performances satisfied. Thus the definition of an arrangement of EDM forms corresponds to the structuring and formulation of a design task. At the same time, it delimits the set of forms possible and has associated ranges of performance capabilities. It is in this manner that EDM provides insight into the structure of design tasks. An emphasis of EDM is the normalized structure of design knowledge, allowing it to be encoded and modularized for re-use.

THE DESIGN OF CHAIRS

Probably the best way to evaluate EDM's ability to encode design knowledge is by considering an example. For the example, I approach product knowledge informally, close to the way it is traditionally understood by designers. Then I will show how this knowledge can be encoded in the EDM structures. I will rely on EDM's graphic syntax for depicting the information, rather than the textual one.

Simple Chairs

A *chair* is defined as (Oxford English Dictionary, 1971):

> "*a seat for one person, implying comfort and ease, that is an article of furniture, four-legged, with rest for the back.*"

A *seat* is defined to be:

> "*for sitting, which is a posture in which the weight of the body rests upon its posterior.*"

Furniture is defined as:

> "*movable articles in a dwelling house.*"

The dictionary definition of chair provides some basic knowledge needed to conceptually design examples. The definition, with the supporting definitions of seating and furniture, includes its function (for sitting) and parts of its form (four-legged, back). Qualities are implied (comfort and ease). Taken at this level, the knowledge regarding the concept chair can be diagrammed as shown in Figure 6. It consists of a high level FE called chair, which has some basic attributes defined in its aggregation. This FE is decomposed into at least six component FEs: seat, back and four legs. Here, each subFE is assumed to have a shape and location. The solid FE defined earlier is used to define shape (see Figure 1) and is inherited through the specialization relation. The transform is assumed to be defined as an aggregation and is copied into the forms that need it.

Four accumulations are implied for this composition, based on the dictionary definitions: for evaluation of ease of use, comfort, movability and for accumulating the total shape from the shape of the component parts. This last accumulation is used for visual review, planning of packing and shipping, etc. An intuitively obvious constraint that can be used to accumulate properties for movability is summation of the chair's weight. The constraint for deriving the complete shape of the chair from its parts is intuitively obvious as the union of all the individual part shapes. Ease of use and comfort could be intuitively dealt with by building a prototype and trying it out. Each of these accumulations has an evaluation value in the top-level chair FE. These values are what the accumulations define in relation to the component properties.

Many people consider design to be easy, because common cultural knowledge allows one to partially structure some design problems in a way that allows simplistic solutions. The given cultural definition of chair has been outlined above. The importance of such definitions should not be underestimated, because they are what most users apply. But like most culturally defined design definitions, this one is incomplete. Necessary performances are omitted, particularly structural rigidity, and the accumulations for comfort and ease of use are limited. Many chairs feel acceptable upon initially sitting in them, but become uncomfortable after a few minutes. Thus more technical definitions are needed for serious design. Such knowledge is part of the technical and craft knowledge in the various design domains.

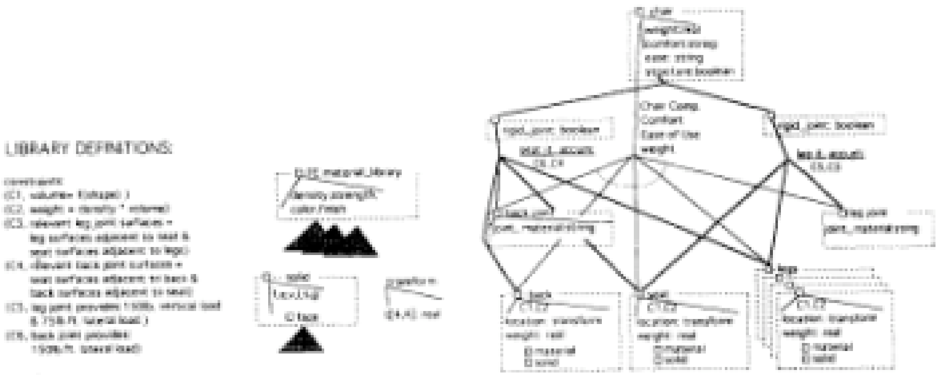


Figure 7. Definition of a more complex chair, based on a library of properties and constraints.

To be more complete, the knowledge structure for designing chairs requires means to define the joining between the leg and seat and between the seat and the back, allowing the structural relations between these parts to be considered. These are shown integrated into the original structure in Figure 7. Pre-defined library knowledge is shown on the left. The joint conditions are defined as accumulations over the multiple parts involved, with rigid structural joining being the overall property. A joint FE is added to the overall chair composition, defining the joint material, such as glue, weld or bolt. The structural criteria have been defined very explicitly, in constraint C5, with specific loads the joint must tolerate. The joint uses a precondition constraint, (C3), to identify the surfaces making up the joint. The identified surfaces, together with the joint FE, allow a variety of tests to be defined, including empirical testing of a prototype or finite element analysis.

The joint accumulation for the back similarly identifies a performance criterion, C6, and a pre-condition to identify the relevant surfaces. It may apply to the seat and/or the legs, depending upon the nature of the joint. Notice also that compositions and accumulations may overlap, as happens in Figure 7. This is because any accumulation may relate subsets of variables describing features that may be overlapping. The rigid joints are first shown separately for legs and back. Combining the rigid joints is another accumulation that checks for the structural soundness of the overall design.

Material has been added as an inherited FE for each part of the chair. Here, a material library is pre-defined with the desired properties specialized into the different chair parts. The accumulation for deriving the weight of the chair has been included also. The derivation takes several steps. First, the volume of the shape must exist. This is checked in constraint C1, which checks that the volume value is consistent with the solid shape, for each chair component. The volume is then multiplied by the material density to derive the weight. The density is assumed to be a property of the material of each part. The weights are then summed for each part of the chair (joint materials are assumed to be too small to affect the computation).

This EDM data model is schematic, without support for detail design. Only the weight and structure accumulations have been defined. Yet given this structure, defining to some level of detail the conceptual structure of a class of chairs (a fairly limited set, I should add), one could begin to design within the class defined, that is, to create specializations of this structure.

Figure 8 presents example chairs within this class. All the chairs have the same major compositional structure: four legs, a seat and a back. They vary in dimensions, curvature, angles, and also in whether they have arms or bracing. Technically, this range of designs can be generated by varying the parameters of a single family of designs. Such a range of designs seems similar to that allowed by prototypes, as used in expert systems (Maher, 1988).

Figure 8. Some sample chairs allowed by the simple design concept.

The difference between Figure 6 and Figure 7 is the difference between the cultural knowledge of chairs and the addition of a minimal amount of more specialized design knowledge. Comfort has still not been considered, nor has any externally imposed stylistic or aesthetic relations.

A More General Definition of Chair

The dictionary definition of chair is unnecessarily restrictive. It eliminates pedestal chairs, as designed by Charles Eames and the bent tube Breuer chairs, among others. In addition to other forms of legs, there are chairs without legs at all, such as basket and other forms of hanging chairs. A unique variation of a chair is the "bean bag chair", filled with any of a variety of irregular, small grain filling.

How can this range of chair designs be embedded into a design knowledge structure? One elaboration involves eliminating the FE form called "leg" in our definition and replacing it with a more abstract functional definition that the seat must be supported off the ground.

Then this functional requirement can be responded to by several alternative compositional structures. For example, in Figure 9, three are shown: constructed frames, single element structures and hanging structures. Each of these would have a different set of components and different rules of composition. Disjoint alternatives are graphically denoted by a line with diagonal dashes. Chairs thus have different structures and no single structure can carry the detail information needed for designing. Figure 9 assumes the availability of the same library as Figure 7.

In a manner similar to *support_structure*, the back and seat can also defined as a single functional unit, allowing them to be defined by one or more components (not shown). Various seat-back compositions can be defined to depict different families of solution: wicker on a frame, leather stretched between the seat and the back frame, molded fiberglass seat-back, etc. Each of these define disjoint families of seat-backs. Together, these revisions allow a wider range of chair forms to be defined.

Some chairs do not even have the same high level structure defined in the dictionary definition. But if one tries to define a single general product model for this range of chairs, it would be too general to hold the specific information needed to design with. Some alternative combination of structures seems mandatory. In most areas of design, it is necessary to consider different classes of composition, each having different model. For example bean bag chairs have a different data model from most chairs, with different components and attributes. Such a technology can be defined to include all types of fillers, such as beans, sand, water (both very heavy) or air (inflated chairs). In Figure 9, on the left is shown this alternative structure.

Each composition defines a different set of components that can make the same assembly, with different forms responding to the fixed, given set of functions. A system of components that, when composed, provide a set of functions is here called a *technology*. Thus the filled bag chair is a technology. Alternative technologies exist for seat-backs and also for support_structures.

One kind of basic knowledge involved in design, as demonstrated here, is the knowledge of alternative technologies/compositions. Here, each composition refers to a parameterized family of forms that satisfy a known set of functions. Knowledge of a different technology adds a new family of designs to the possible set. The larger the set of alternative technologies for some part of a design, the larger the space of alternative designs.

Evaluation of Chair Designs in Terms of Comfort

A large amount of ergonomic research has studied the stress on the back resulting from various seating postures and also subjective assessments of extended sitting in chairs with different seat and back contours. Some tests have assessed posture with different tasks, such as reading, writing and relaxing. Many of these studies are summarized in Grandjean (1973).

The dictionary definition and our elaboration distinguished back from seat. If these are taken to be physical elements, then we have seen that the definition is too restrictive. They need to be considered *features* that are sets of surfaces within the form of the chair. The comfort and ease of use accumulations apply to the relevant surfaces, even though there is not a seat and back part per se.

The angle of the chair seat and its height from the floor interact. Too shallow or horizontal seats result in sliding forward. Too steep angles result in difficulty in getting up. In some studies, angles of 5° to 7° from horizontal appear to be preferred for most activities (Keegan, 1962), (Akerblom, 1948). Other studies show larger angles are more relaxing, up to 26° (Yamaguchi, 1970). The seat height found most

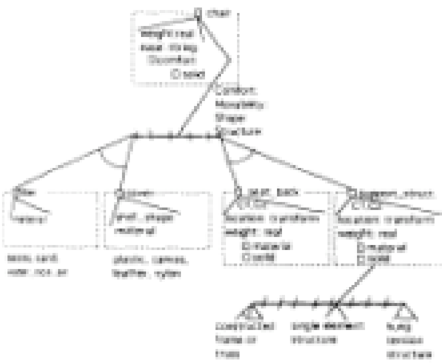
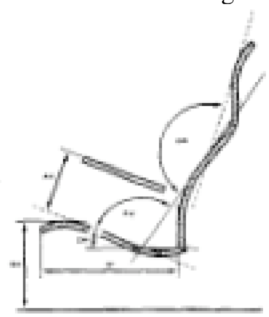


Figure 9. Elaboration of the chair concepts.



Construction Detail:

	Healthy person	Persons w/ back ailments		Compromise
	Reading	Resting	Rest	Read and Rest
Angle of seat	23	26	20	23
Slope at backrest	103	107	106.5	107
Seat Height(cm.)	40	39	41.5	40
Depth of seat(cm.)	47	47	48	48
Height of lumbar pad above seat(cm.)	14	14	9	8-14
Height of arms(cm.)	26	26	26	26

Figure 10. Ideal seating for people without and with back ailments, from (Grandjean et al. 1967).

comfortable in various studies ranged from 32 cm. to 42 cm., with most centering around 39 cm. These studies have been made with different cultural groups, whose body stature varies. Lower seats require a greater seating angle. Seat depths of 43-44 cm. seem preferable for most people (Keegan, 1962),(Akerblom, 1948). More angled seats lead to deeper preferred seating, up to 48 cm. for relaxation at 26^{o} (Grandjean,1973, pp. 120-122).

The preferred seat-to-back angle also varies according to the angle of the seat. Larger seat angles lead to smaller back angles. Different studies suggest angles between 100 and 120^{o}, with larger angles for resting and smaller angles for reading and writing (Grandjean, 1973, pp.106-114).

These results suggest a set of constraints, with regard to the seat height, the seat angle from horizontal and the angle of the back to the seat. The constraints can set limits within which a reasonable level of comfort will be achieved.

But seating typically involves not flat surfaces but curves. The comfort of a chair responds subtly to gradations in contour. Figure 10 reproduces an ideal contour developed by Grandjean et al, (1967) from extensive empirical studies. The grey area indicates stiff latex above a hard shell, indicated by the black line. The latex allows even distribution of pressure over the seating surface (as does all padding, on the seat or the person's posterior). Lumbar support in the lower back should be 8-10 cm. above the point where the back and seat join.

None of the work on seating comfort has been defined quantitatively, in a manner easily processed on computer. Most existing studies have attempted to define a single fixed contour that is comfortable to the largest segment of the population. Existing data appears to be sufficient to provide ideal contours from which deviation could be measured. I have found no work that would calibrate the degree of discomfort resulting from such deviations.

Figure 11 presents a later study's ideal contour (Grandjean, 1973, pp.118-119). This contour could be defined within the comfort accumulation. Deviations from this ideal could be computed at constant intervals with the equivalent cross section of a designed chair. The differences between the two contours, stored in a vector is one measure of the comfort of the chair. Such an accumulation is shown in Figure 12. The seat and back contour are extracted using the accumulation pre-condition constraints. These are then compared to the ideal contour and differences identified.

In the author's experience with chairs, the lateral contour of the seat, from side to side and the contour of the back, also affect seating comfort. No information has been found, however, with regard to these issues.

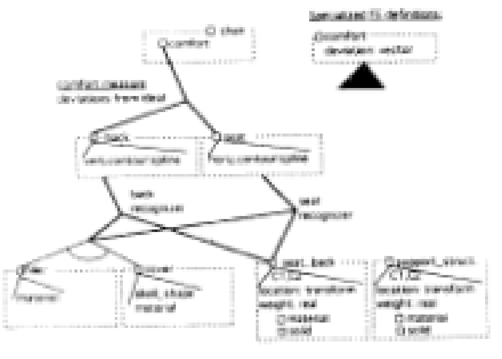


Figure 11. The contour derived by Grandjean in a later study of ideal seating (Grandjean,1973).

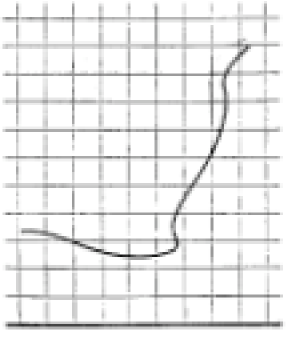


Figure 12. The comfort accumulations applied to the chair concepts. Seat and back are recognized features, then evaluated.

Thus any measurement based evaluation of alternative chair designs, in terms of comfort, is currently limited. Because of the lack of data, it is still mandatory that construction of a prototype and test of its comfort be undertaken. This, of course, only allows determination of its acceptability, comfort wise. It tells us very little about its relative comfort.

Figure 13 updates the joint and structural evaluation. The rigid joint conditions needed for checking the structural soundness must address different conditions in the different compositional structures. Supports made by frames have different joint conditions than those in hanging chairs. Part of the special knowledge about a composition is how to evaluate different functions within it. This becomes more apparent when we consider the definition of a composition as a set of elements related by accumulations.

Aesthetic Considerations in the Design of Chairs

I turn now to an issue of external complexity. There are many ways to develop an aesthetic approach to the design of furniture and it may be based on quite varied philosophical bases. Here, we consider aesthetic considerations to be those that are outside of the function of the product and that influence its cultural interpretation, particularly stylistically.

Historically, one important influence on the aesthetic aspects of chairs has been the materials used. Different styles have associated materials: steam bent woods for the modernist style of Alvar Aalto, fiberglass for the chairs of Charles and Ray Eames, the tubular steel of the various European architectural modernists: Corbu, Breuer, and Mies. Recently, avant garde Italian design is producing chairs made from flat steel stock, which have their own distinctive style. Each material has ways for bending and joining it, which defines a vocabulary of forms. Many of these styles evolved from the exploration of production ways of working a material. Thus in many cases, the aesthetic issues of a chair design are determined by the choice of materials and the set of forms the material defines, using current production methods.

Associated with a particular material may a range of methods for working it. In these cases, other issues may be applied. In Figure 14, I show a simple exploration of the change of scale of the whole chair design and also its component members. In the example, it is shown that oak may be used to define a mission style chair, by using flat wide elements and wide overall proportions. By narrowing the overall proportions, the same material and design may be used to create other chairs, such as a scandinavian style chair. In these cases smaller rounder elements are selected. In the last case, one slat is removed from the back.

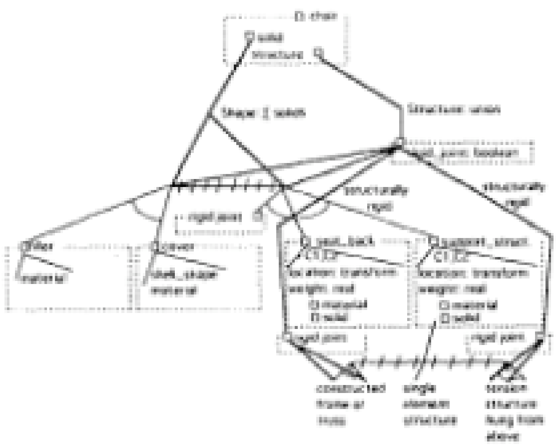


Figure 13. The structures and shape accumulations applied to the more general chair concept. Alternative structures are connected with a line crossed with dashes.

These issues are captured within EDM in different ways. The aesthetic issues associated with a material are captured by restricting the set of materials, geometry and joining of materials to a restricted set or range of alternatives. The overall proportional issues could be captured in the shape accumulation, which could have constraints added that eliminated certain overall proportions. The component material proportions could be restricted by imposing constraints on the set of shapes allowed for defining the component designs. These issues only scratch the issues associated with aesthetic concerns.

Use of an EDM Structure for Design

The EDM structures laid out here define a conceptual model of the product being designed, in this case a chair. We have discussed how the definition of this conceptual structure affects the class of designs allowed. We assume that a designer both generates chair conceptual structures, as defined here, and further constrains the structures by selecting materials, colors and other properties that determine the final design of a chair.

The analysis of design information used in chair design has not proceeded to the level of detail needed to actually construction. In particular, the information needed to develop a detail design within a particular technology has nor been defined, though we have given some hints about its contents. Such information would be needed to support detail design, to the fabrication level.

Discussion

Most of the efforts to structure design knowledge have relied on simple, well-defined forms. Until now, design knowledge in intuitive and open-ended design domains has not been attempted. Part of this avoidance has been the paucity of adequate tools.

I have attempted to show here how an open-ended and ill-structured design domain may be recorded. In particular, I have generalized various classes and sub-classes of knowledge in a way that they can be used by others. These knowledge structures are not the only ones possible; there may be more elegant means to represent the same or an expanded domain of chair design knowledge. Elegance will only come after mastery of the concepts.

Most of the names of things used in design, both whole products and components, have both an associated function and a form. The information structures defined here imply both. Probably the most common method of design creativity is to redefine an existing component as a function and find new ways to support that function. An example is the re-definition of seat and back to be seat and knees in the new posture chairs recently introduced to the market. This change of definition was based on original studies of comfort and support.

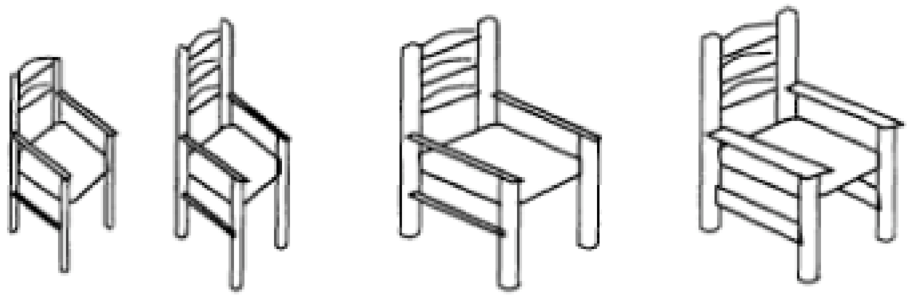


Figure 14. Stylistic transformation of chairs dealing with both overall proportions and proportions of parts. The left is a scandanavian style, which is re-scaled into the ones to the right, becoming one with mission style. The number of back slats was also reduced.

Innovation may come from identifying an alternative technology for any one of the structures defined. For example, the support provided by a hammock, strung between two trees, might be the basis for generating a new support technology for a chair design. Alternatively, one might attempt to develop and use a hovercraft air-cushion. Each technology has a range of geometry and other properties that affect the total valuation of a chair design. We usually do not worry about the noise a chair might make, but this would become an issue if an air cushion is used.

New forms of chair easily arise out of this type of analysis: inflatable chairs, as a modification of a bean-bag chair; hovering and hammock chairs, defined by identifying new alternatives for the support function, as defined in Figure 9, or new forms of seat-backs, such as a potato chip chair.

As these examples are presented, it should be apparent to the reader that the focus of EDM is not to define specific designs, though that is possible, but rather to define design knowledge.

The application of data modeling has clear implications for the definition of design tools. As CAD technology moves toward systems that include more intelligence about a product domain, data modeling suggests the limits of a particular data model. The chair definitions here could be used as an initial specification for the development of a chair design system.

Acknowledgements

The author acknowledges the contribution of his associates, especially Alan Bond and Scott Chase, to the ideas developed here. This work has been supported by the National Science Foundation, grant number DDM-8915665.

REFERENCES:

Akerblom, B., 1948, *Standing and Sitting Posture*, Nordiska Bokhandeln, Stockholm.

Akin, Omer, 1986, *The Psychology of Architectural Design*, Pion Press, London.

Alexander, C., S. Ishikawa, M. Silverstein 1977, *A Pattern Language*, Oxford University Press, New York.

Batory, D. S. and W. Kim, 1985, "Modeling concepts for VLSI CAD Objects", *ACM Trans. on Database Sys.* 10, no.3, pp. 322-346

Bond, Alan H., C. Eastman and S. Chase, 1991, "Theoretical Foundations of EDM Product Models", Design and Computation Report, GSAUP, UCLA, 1991.

Chen, P. 1976, "The entity-relationship model: towards a unified view of data", *ACM Trans. on Database Sys.*, 1, no.1, pp. 9-36.

Eastman, C.M., 1991, "A data model analysis of modularity and extensibility in building databases", *Environ. and Building,* (to appear)

Eastman, C., A. Bond and S. Chase, 1991a, "A formal approach for product model information", *Research in Engineering Design*, 2, no.2, pp.65-80.

Eastman, C. M., A. Bond and S. Chase 1991b, "Application and evaluation of an engineering data model", *Research in Engineering Design*, 2:4, (forthcoming).

Eastman, C. M., A. Bond and S. Chase, 1991c, "A data model for engineering design databases, *Proc. Artificial Intelligence in Design*, Edinburgh, Scotland, June.

Encarnacao, J. and F. L. Krause (eds.) 1982, *File Structures and Databases for CAD*, North Holland Press, N.Y.

Gielingh, Wim 1988, General AEC Reference Model (GARM), ISO TC184/SC4, Document 3.2.2.1 (Draft), (October), TNO-IBBC.

Grandjean E., A. Boni and H. Kretszchmar, 1967, Entwicklung eines Ruhesesselprofils fur gesunde und rückenkranke Menschen, *Wohnungsmedizin*, 5, pp. 51-56.,

Grandjean, E. 1973, *Ergonomics of the Home*, Taylor and Francis, London.

Jardine, Donald, (ed.,), 1974, *Data Base Management Systems*, 1973 SHARE Working Conference, North Holland Press, N.Y..

Karnopp, Dean and R. Rosenberg 1975, *System Dynamics: A Unified Approach*, John Wiley and Sons, N.Y.

Keegan, J.J., 1962, "Evaluation and improvement of seats", *Industr. Med. Surg*. 31, pp. 137-148.

Lawson, Bryan, 1990, *How Designers Think,* 2nd Ed. Butterworth Architecture, London.

Maher, M.L., 1987, "Engineering design synthesis: a domain independent representation" *A.I. in Engr. Design and Manag*., 1, no.3, pp.207-213.

Mantyla, M., 1988, *Introduction to Solid Modeling*, Computer Science Press, Rockville MD.

Oxford English Dictionary, 1971, Oxford University Press.

Peckham, J. and F. Maryanski, 1988, "Semantic data models", *ACM Comp. Surv*. 20, no.3 pp. 153-190.

Shaw, N. K., M. S. Bloor, and A. de Pennington, 1989, "Product data models", *Research in Engineering Design*, 1, no1, pp. 43-50.

Smith, B., 1986, "A Report on the PDES initiation effort", National Bureau of Standards, (May 20), Gaithersburg, MD.

Turner, J. A. 1988, "A systems approach to the conceptual modeling of buildings" *CIB Conference on Building Data Models*, TC 5.2 and 7.8, Lund, Sweden, Dec. 1988, Christensen, P. (ed.)

Van Griethuysen (ed.) 1983, *On Conceptual Schema*, ISO Report TC97/SC5/WG3.

Yamaguchi, Y. and F. Umezawa, 1970 "Development of a chair to minimize disc distortion in the sitting posture", unpublished paper reported in (Grandjean, 1973).