Proceedings of DETC'98
1998 ASME Design Engineering Technical Conferences
September 13-16, 1998, Atlanta, Georgia

DTEC98 /EIM-5682

A DATA APPROACH TO TRACKING AND EVALUATING ENGINEERING CHANGES

Tal Cohen
School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, Georgia, 30332-0405
Tal.Cohen@cad.gatech.edu

Robert E. Fulton
School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, Georgia, 30332-0405
robert.fulton@me.gatech.edu

ABSTRACT

Changes are common during any stage of a product life cycle. There are local changes that do not influence other elements of a product. However, there are other changes that can influence different aspects of the product. Consequences of these changes, unless properly anticipated, and accounted for, can be costly. Therefore, it is highly desirable to obtain a mechanism that will be able to anticipate and evaluate product change consequences.

The first task in anticipating and evaluating change consequences is to represent them. The complexity of engineering models makes their representation to be rich and semantic. Information data models like EXPRESS provide tools for modeling products. However, the current EXPRESS and other information models do not have a generic methodology to support contextual change representation and propagation.

In this paper a methodology called Change FAvorable Representation – C-FAR is presented. C-FAR uses an existing product information model to facilitate change representation, propagation, and qualitative evaluation. The EXPRESS schema's main elements are entities, relations among entities, and attributes that describe the entities.

C-FAR facilitates change and change evaluation to the attribute level.

C-FAR has been evaluated using case studies in structural analysis, and bumper design. Results show that C-FAR is capable of representing change and provides a reasonable qualitative evaluation of the change consequences.

INTRODUCTION

Design is a complex and dynamic process [Dym, 1994; Fulton, 1988; Kannapan, 1992, Keller, 1992]. As a result, changes in various design stages influence different aspects of the design. These changes are necessary; they are an integral part of any design. Engineering design, is by nature, an iterative process that evolves until it reaches the optimal point [Cutkosky, 1990]. The optimal design addresses a set of requirements under a certain set of constraints. However, even after reaching this desirable point, the requirements as well as the constraints may change. Our optimal design may no longer be optimal for the new set of requirements. Therefore, the change element should follow the design process from initial conceptual design through maintenance and the entire life cycle of the product [Curtis, 1994; Dagle, 1994; Dym, 1994; Liu, 1993].

Sometimes the change initiator is not aware of the consequences of the changes he or she makes. Changes in requirements may be initiated by an engineering redesign motivated by the customer's ever-evolving needs, by competition, or by the need for internal improvements. The complex design structure includes different data sets that are associated with different parts of the product [Fulton, 1988; Miller, 1996]. Since the design is a complex, elaborate endeavor, the data sets representation emphasizes the design components' interdependency. As a result, a change presented into the system may influence many other aspects of the design [Keller, 1992]. The change initiator has the responsibility to trace the change propagation and evaluate the overall impact that a given change may produce. Eventually, to benchmark the new design, the initiator may compare the new design against the original design.

Concurrent engineering [Curtis, 1994; Cutkosky, 1990; Dym, 1994] provides a framework to make design changes. Concurrent engineering allows feedback from any stage to any other stage in the process design. As opposed to the classical serial design process, the concurrent engineering framework allows for more flexibility in introducing changes along the design. Concurrent engineering is more than an innovative design representation; it is a framework used to capture the evolutionary nature of modern design. This framework facilitates interaction among design stages by identifying, refining, and transforming the requirements and then moving on to structural description and finally to defining a physical plan. However, change investigation and propagation even in concurrent engineering are not emphasized as much as they should be.

The objective of this research is to present a data representation that facilitates change and change propagation in design representation of engineering products. This mechanism will catalyze the redesign process based on the information gathered on the product. The research examines design representation from a data model perspective. There are two parallel outputs in product development: the physical product, which is the traditional design process output, and the information product that describes the physical product. The work suggested in this paper takes a classical data model and converts it to a Change FAvorable Representation (C-FAR), a new and different methodology of representing design information so that redesign changes can be dynamically anticipated and evaluated. In order to do this, C-FAR will use STEP [Bloor, 1991; Gilbert, 1991; STEP Part 1, 1992; Wang, 1991] (Standard for the Exchange of Product) model data, a well-established data model. The method is to develop an approach -- C-FAR -- that will extract the information from the STEP information model to make the design changes more easily traceable. This methodology will take into consideration the interdependencies among design elements, thus facilitating a deeper understanding of a change consequence and of design parts sensitivity. The methodology for C-FAR will utilize modeling case studies to derive the study and validate results. The first case study is a simple model of an automobile structure [Kamal, 1981]. Next, the model is expanded to include other part views, such as an automobile bumper model. The significance of this research is to provide a mechanism that will make design information an active agent in the redesign process. The research approach will add to the design ability by turning the information model into a dynamic part of the design. The first stage of redesign change evaluation is to learn who is affected by the initial change. The second stage is to learn to what degree the initial change indeed influenced a given design artifact.

BACKGROUND

The object-oriented representation facilitates relatively complete design representation and, together with the version management, improves design evolution. However, the change in object-oriented representation as well as in any of the design representations is in the best case implicit. In both the serial design process and in concurrent engineering, there is no change-oriented representation. In fact, the data models including STEP tend to suggest that the design is static. Concurrent engineering strives for the design evolution and the dynamic nature of design. Therefore, it is very important to tighten the integration and the communication among different parts of the product. The STEP technology provides the standardization and the data structure umbrella.

Plausible efforts in building a schema evolution mechanism have been described Bouneffa, 1995; BritsGal, 1995; Monk, 1993; 42; Rumbaugh, 1994; Sciore, 1991; Sjoberg, 1993; Skara, 1996; Uneda, 1990; Ullman, 1993]. However, there is still a gap between the design engineering methodologies and the technologies that are supposed to facilitate the concurrent design. For example, the product data manager capabilities are limited to record controlling, and in fact they can be described as merely upgraded file managers [Curtis, 1994]. On the one hand, the design community builds the design methodology that requires intensive and sophisticated technologies, and on the other hand the information modeling community develops the tools to serve these methodologies. The design methodology is running ahead of the technology, and sometimes the methodology is delayed because of a lack of existing technology. Moreover, if the technology could keep up with the design methodology, it would improve the cross-fertilization between the two. There is a need to reflect change in design and explore its propagation.

PROBLEM SCOPE

In this section, the problem borders are articulated. The first part of this section is dedicated to EXPRESS, since the C-FAR methodology is built on the EXPRESS information model. The next point examined in this section is what the EXPRESS information model provides and what it does not provide.

EXPRESS - DATA DEFINITION LANGUAGE

To understand the C-FAR methodology, it is helpful to first understand STEP and the language that it uses, EXPRESS.

EXPRESS [Schenk, 1991] is the formal information modeling language used to specify the information requirements of other parts of the STEP. EXPRESS defines schemas objects attributes and behavior. For example, a schema “automobile _bumper” will describe an automobile bumper’s objects, attributes, and behavior. EXPRESS is based on the following design goals: the language will be parsable by computers, the language is designed to enable partitioning of the diverse material assessed by this STEP, and the schema is the basis for partitioning and intercommunication. Finally, the language is focused on the definition of entities, which are elements of interests. The definition of entities is in terms of data and behavior. Data represents the properties by which an entity is realized and behavior is represented by constraints. EXPRESS has a graphical subset representation called EXPRESS-G. The EXPRESS-G is a graph-theory type representation method.

Although it has been specifically developed for the graphical rendition of information models defined in the EXPRESS language, it may be used as a modeling technology in its own right. The EXPRESS main design goal is to be intuitively understandable, and it is also supposed to support levels of abstraction. An information model is considered to consist of definitions of things (entity, type, function, etc.). For example, the following is an example of a circle description in EXPRESS followed by the same circle represented in EXPRESS-G:

```
SCHEMA circle;

ENTITY point_3D;
        x1:     REAL;
        x2:     REAL;
        x3:     REAL;
END_ENTITY;

ENTITY circle;
        center_point:   point_3D;
        radius:         REAL;
END_ENTITY
END_SCHEMA;
```

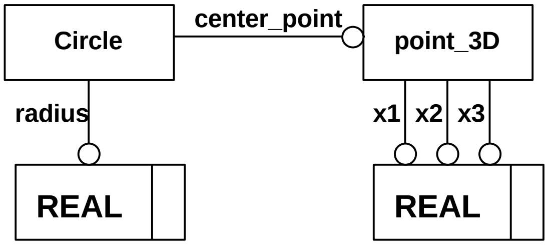

Figure 1 Example of A Circle Description in EXPRESS-G

C-FAR VS. EXPRESS

The Change FAvorable Representation (C-FAR) attempts to use the knowledge domain that exists in the EXPRESS schema for purposes of exploring changes and their influences. The EXPRESS information model captures the domain artifacts with four main elements: a schema that defines the domain frame; entities, which are the main objects in the domain; relations that describe the connectivity between entities; and attributes, which describe the entities.

C-FAR uses the EXPRESS schema as it currently exists and adds domain knowledge to it. This domain knowledge purpose is to facilitate change and change propagation within the existing objects. However, C-FAR does not expand the EXPRESS schema coverage from a contextual perspective. C-FAR does not add entities, attributes, or relations to the EXPRESS schema.

C-FAR adds to EXPRESS in two main ways. The first is to view an entity as a vector and its attribute as the vector's components. This view focuses the changes of a schema to be changes in the entity's attributes. Next, C-FAR creates matrices between entities that are connected with relations. The C-FAR matrices enable change propagation.

THE ENGINEERING CHANGE PARADIGM IN C-FAR

The EXPRESS diagram represents data on two levels, the meta-data level and the instances level. The meta data level underlines the schema, entities, relations and attributes. The actual values are compound of the attributes instances. Engineering changes may translate to changes at either of the two levels. Namely, engineering change -- if it is an introduction of a new set of design variables, replacement of complete functional modules, or a value change in a single variable -- are all reflected in two of the representation levels.

An interesting question is, once we introduce an engineering change, can we know if this change affects the data, meta-data, or both? Since we assume that we have an EXPRESS schema that covers the engineering domain adequately, it is not a difficult task to check all the current attributes and values and examine the proposed change. If the complete proposed change can be described with changing any of the existing attribute or values, then it can be said that the change is meta-data change independent. However, if it is impossible to reflect the engineering change only via the existing attributes, then the engineering change causes a manipulation of the meta data EXPRESS schema. Therefore, given the meta-data schema and the proposed change, it is feasible to classify the engineering changes as meta-data change independent and meta-data change dependent. It is important to emphasize that the schema definition is no less important than the engineering change in determining if a change is meta-data independent or not. The **C-FAR deals with changes that are meta-data independent**. By not considering the meta data dependent changes, the C-FAR research directs its efforts towards engineering changes of semantically parametric nature. Considering meta-data independent changes avoids going into schema evolution theory. Moreover, it creates a disjointed state between the unchanged EXPRESS schema and C-FAR and the proposed engineering changes. Namely, the engineering change initiator should not have the knowledge of EXPRESS or how to build an EXPRESS or C-FAR schema. C-FAR will be a black box for the engineering change initiator and change initiator will be exposed only to the measurable quantities of the schema.

In Figure [2] an example of 2D truss structure model is given. The corresponding EXPRESS schema is given in Figure[3]

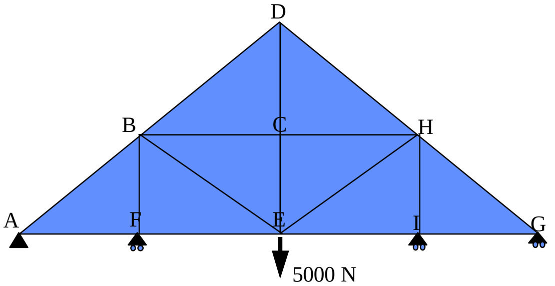

Figure 2 Example 2D Truss Structure Model

The corresponding EXPRESS schema is given in Figure 3

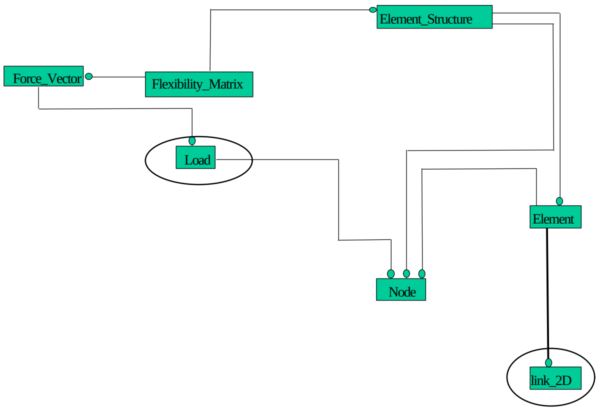

Figure 3 Example EXPRESS-G Model

Following is an example of an engineering change scenario:

How would a change to the Load_x_dir_val attribute of the entity load will influence the entity link_2D ? in Figure 3. Will this change strongly influences it, somewhat influence it, or not influence the link_2D's attributes at all?

C-FAR OBJECTIVES

The overall objective of this work is to devise a methodology which will aid engineering redesign by enabling engineering change representation and change propagation activity. In this section, a description of the research objectives is specified.

RELEVANCY OF COVERAGE

The first objective of the research work concerns the relevance to various engineering disciplines in general and the product-related engineering field in specific. In other words, the research output should be relevant to engineering domains and their corresponding applications and products.

The C-FAR methodology achieves relevancy of coverage by building the C-FAR methodology on top of the STEP standard. The C-FAR methodology is tailored for the EXPRESS information model, which is used to describe the STEP parts. STEP parts are geared to model a wide range of engineering domains, e.g. engineering drawing, finite element methods, automobile applications etc.

ENABLE ENGINEERING CHANGE REPRESENTATION

The methodology should specifically support engineering change representation. Engineering change representation means the ability to express the change in a problem domain from one state to another. The C-FAR methodology should conceptually capture an event, namely engineering change. Engineering redesign activity is compounded of one or more engineering changes. Change representation is one of the two cornerstones that the C-FAR methodology uses to build on top of the EXPRESS model.

ENABLE CHANGE PROPAGATION

The methodology should provide a mechanism to facilitate an initial change propagation throughout the problem domain description. A change propagation mechanism is an essential part of the C-FAR methodology. Change propagation is the second cornerstone that the C-FAR methodology uses to build on top of the EXPRESS model.

CORRECTNESS

The methodology should adequately reflect the proposed change of an EXPRESS attribute linkage to another EXPRESS attribute. Therefore the correctness of the C-FAR approach is dependent on the goodness of change representation, matrix construction and change propagation.

C-FAR APPROACH FOR CAPTURING CHANGE

The Change FAvorable Representation methodology -- C-FAR -- represents the notion of change and facilitates change propagation in the engineering information model framework. While existing information models provide a comprehensive product description, they are not able to represent change or indicate change consequences.

C-FAR's purpose is to transform the information model to a contributing active participant in exploration of the product's engineering characteristics. The C-FAR methodology achieves this by estimating change consequences. For example, given an information model about automobile bumper components, C-FAR is able to indicate and qualitatively estimate whether a change in the automobile height would change the choice of a bumper component.

This section provides an overview of the C-FAR methodology, components and assumptions.

The Change FAvorable Representation methodology is based on EXPRESS. The EXPRESS information model was created to define engineering products and support management of key engineering data. Specifically, EXPRESS provides linkages between engineering elements. EXPRESS defines its main artifacts as objects or entities. In turn, these entities are described via their attributes. For example, the entity “Circle” is described by its radius and center point. At the very heart of EXPRESS is the notion of a *relation*. EXPRESS ties relevant entities into a relation. The entities that are in this relation have a certain contextual importance to one another.

EXPRESS serves its purpose of modeling engineering products, well but it falls short in another way. EXPRESS only links entities **--** it has no mechanism to describe the linkages between the attributes of the entities. C-FAR represents entities as vectors and their attributes as vector components. A matrix called a C-FAR matrix provides links between the attributes of one entity and the attributes of another entity. The components used to construct the C-FAR matrix are called **linkage values.** A linkage value represents the relation between two attributes, one from each entity. Since C-FAR is geared towards the notion of change, a linkage value between two attributes is assigned to answer the following question. How would a change to the attribute in one entity affect the attribute of the second entity? This question is answered by a domain expert and the answer is ‘high’ if the entity is strongly affected, ‘medium’ if it is affected somewhat, and ‘low’ if it is not affected.

By creating the notion of a C-FAR matrix, more information about the product is being represented, and this information is geared towards change. However, a C-FAR matrix only provides linkages between the attributes of two entities that are connected by a relation. A new question may arise: How would a change in one entity influence a third entity which is not directly connected to a changed entity? One answer to this question is to ask a domain expert to build linkages between all the possible entities. Even though this would be feasible, doing this poses some problems. First, consider a schema with n entities and 3n relations. If a domain expert were to build linkages between all possible entities, this would increase the number of matrices describing linkages from O(n) matrices to O(n$^{\wedge 2}$). This would clearly increase the complexity of the job. There is another problem with having experts build linkages between entities. Considering that an EXPRESS schema can be across several engineering domains, many knowledgeable domain experts may be required to make linkages between entities.

There is an alternative to having domain experts make these linkages. Instead, a change propagation mechanism could be used. C-FAR provides a change propagation mechanism, which is explained in detail in the following sections. Fortified with a combination of improved data dependency description and a mechanism to propagate its linkages, C-FAR’s aim is to qualitatively evaluate the affect of engineering change that is made from one attribute’s entity to another.

C-FAR VECTOR REPRESENTATION

To facilitate engineering change representation, C-FAR uses the EXPRESS schema, which has entities, attributes and relations. The following is a description of the C-FAR interpretation of a given EXPRESS schema. Within C-FAR, an entity is a **vector**, and the vector’s dimension is the number of attributes of the entity. For example, Figure 4 describes the EXPRESS entity and attributes for a bottle.

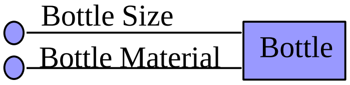

Figure 4 Example Bottle Entity in EXPRESS

The C-FAR meta-data representation for the entity load is the following:
[Bottle Size, Bottle Material]

The following is an example of data representation of this entity:
Bottle[3, Glass]

The bottle vector is defined by the value of its attributes. In this case, the bottle size is 3 liters and the bottle material is glass.

C-FAR VECTOR CHANGE REPRESENTATION

A change in any of the bottle vector attributes values will create a new bottle vector.

For example, assume a change to the bottle size:
Bottle[Δ , 0]

This is a change vector for the bottle vector where the changed attribute is the bottle size. The notion of change in this research describes whether the attribute is subjected to a change or not. It does not speculate on the type of change, e.g. large, small, increase or decrease.

C-FAR RELATION REPRESENTATION

C-FAR matrix relates each component of one vector to the components of the other vector. In the EXPRESS information model, a relation connects two entities. Therefore, a two dimensional matrix is sufficient to represent an EXPRESS relation. The matrix components are called linkage values and their role is to qualitatively illustrate how a change in one attribute will influence the other. The matrix dimensions are n*m where n is the dimension of one vector and m is the dimension of the second.

C-FAR LINKAGE VALUE

Linkage values can be, H, representing high linkage between the attributes; M, representing medium linkage; and L, representing low linkage between the attributes. Clearly the optimal situation would be to have an absolute knowledge source provide a normalized number in the interval 0 to 1 to symbolize linkage between two attributes. However, since in the engineering domain mere humans have to estimate linkage values, providing a limited choice of linkage values options is a better approach. Many sociological and psychological discussions have been held on how to refine the number of choices given to a respondent [Guinta, 93], [Bicknell, 96]. In this research, arguments for various types of ranking systems were considered, and the house of quality [Clausing, 88], [Bahrami, 1992] using three classifications -- high, medium and low – was selected. The house of quality relates engineering attributes to each other using high, medium and low linkage values. A low linkage value between Element A and Element B means that a change to Element A does not influence Element B. A medium linkage value between Element A and Element B means that a change to Element A somewhat influences Element B. A high linkage value between Element A and Element B means that a change to Element A strongly influences Element B.

The following is an example illustrating this terminology. Figure 5 illustrates a bottle containing a liquid. The relation to be examined is between the bottle itself and the liquid.

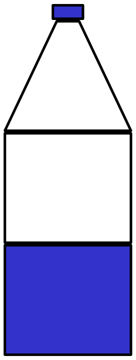

Figure 5 Bottle and Liquid

The bottle attributes are:
1. Bottle Size
2. Bottle Material (glass, plastic, etc.)

The liquid attributes are:
1. Liquid Type (wine, beer, water, oil, milk, etc.)
2. Liquid Quantity

This example illustrates how to determine the linkage values between all the liquid attributes and the bottle attributes. Then the question can be asked how, for example, a change in the liquid type would influence the bottle attributes.

	Influences Bottle Size	**Influences** Bottle Material (Glass, Plastic, etc.)
How change in Liquid Type	Medium	High

Table 1 Example Linkage Value

Table 1 contents explanations:

A change in the liquid type is somewhat related to the bottle size. There are different liquid types that have the same bottle size and there are liquid types with unique bottle sizes. It cannot be said that there is no linkage at all between liquid type and bottle size. However, it cannot be said that say there is a strong linkage between them, the label for this linkage value is **Medium.**

A change in the liquid type is strongly related to the choice of bottle material. For example, milk comes in a plastic bottle, alcoholic beverages come in glass, etc. Therefore, it can be said that there is a strong linkage between them, so the label for this linkage value is **High.**

	Influences bottle size	**Influences** bottle material (Glass, Plastic, etc.)
How change in liquid quantity	High	Low

Table 2 Example Linkage Value

The following explains the contents of Table 2:

Since a change in the liquid quantity is strongly related to the bottle size, we can say there is a strong linkage between them. Therefore, the label for this linkage value is **High.**

A change in the liquid quantity is not related to the choice of bottle material. For example, drinking water comes in plastic containers of all sizes. We can say there is no linkage between them. Therefore, the label for this linkage value is **Low.**

Generally, the medium linkage value will be assigned to a relation when in some instances there will be linkages between attributes, but in other instances the relation will be low or irrelevant. It is possible to reverse the roles of the attributes in Tables 1 and 2. Table 3 illustrates how a change in the bottle's attributes influences the liquid's attributes. The reverse matrix does not necessarily contains the same linkage values.

Liquid **Bottle**	**Influences** liquid type	**Influences** liquid quantity
How change in bottle size	Medium	High
How change in bottle material	High	Low

Table 3 Example Linkage Value

Tables 1, 2 3 can be combined to Table 4:

Bottle **Liquid**	Bottle Size	Bottle Material
Liquid Type	M/M	H/H
Liquid Quantity	H/H	L/L

Table 4 Example Double Linkage Value

As illustrated in Table 4, each linkage value table slot contains two linkage values. The left linkage value in each slot indicates how a change in the liquid's attributes influences the bottle's matching attributes and the right linkage in each slot value indicates how a change in the bottle's attributes influences the liquid's attributes.

C-FAR MATRIX

The C-FAR matrix represents a relation between two entities and defines the influence of a change of one on the other in either directions. The number of rows in the C-FAR matrix is the number of attributes in one entity and the number of columns is the number of attributes in the second entity. The C-FAR matrix elements are linkage values. Each element in the C-FAR matrix is a compound of two linkage values. One linkage value represents how a change in one attribute of Entity A influences the attributes in Entity B. The second linkage value represents how a change in one attribute of Entity B influences an attribute in Entity A. For example, Figure 6 illustrates a C-FAR matrix with two linkage value for each slot.

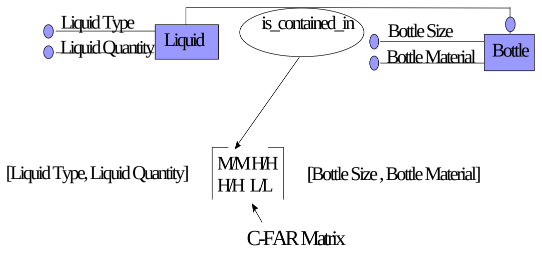

Figure 6 Example C-FAR Matrix

The left linkage value in each slot indicates how a change in the liquid's attributes influences the bottle's matching attribute and the right linkage value in each slot indicates how a change in the bottle's attributes influences the liquids' attributes. This C-FAR matrix was derived from Table 4

SEMI C-FAR MATRIX

The semi C-FAR matrix between Entity A and Entity B is denoted as C(A,B). Unlike the C-FAR matrix where each matrix element includes two linkage values, C(A,B) has only one linkage value per element. The linkage value represents how a change in one attribute of Entity A influences the attributes in Entity B. C(B,A) is a corresponding semi C-FAR matrix in the opposite direction between Entity B and Entity A. Each element value in C(B,A)represents how a change in one attribute of Entity "B" influences the attributes in Entity "A".

C-FAR CHANGE PROPAGATION

A source entity is an entity where one of its attributes has been changed by the user. In Figure 7 entity A is the source entity and the goal is to measure the change originated at this entity.

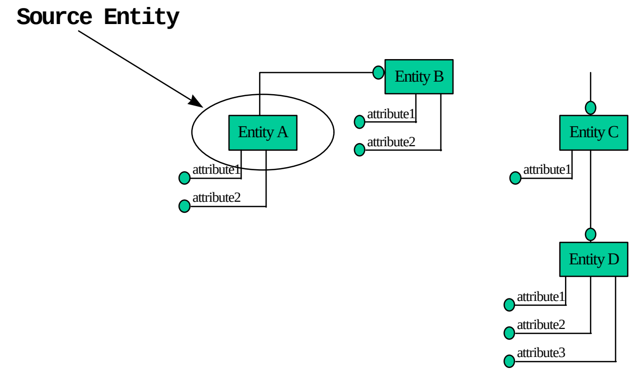

Figure 7 Source Entity Example

The target entity is defined as an entity that is influenced by the source entity and has been selected for examination. In Figure 8, a change in Attribute1 in Entity C is measured.

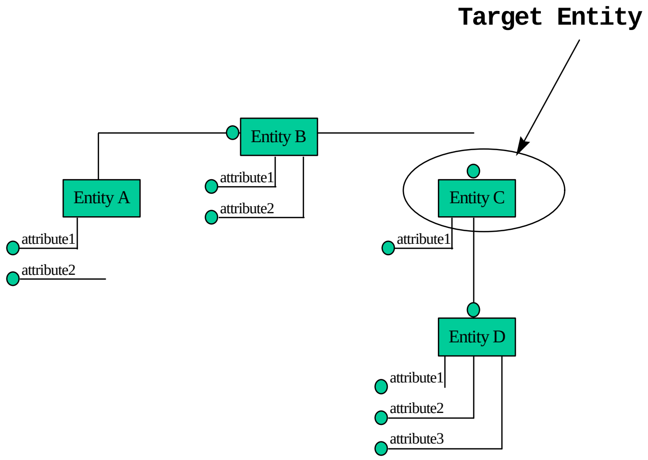

Figure 8 Target Entity Example

Change propagation is an important part of the C-FAR methodology. This mechanism is used for calculating the consequences of a change from a source entity to a target entity. A simple path that leads from the source entity to the target entity is called a simple influence path. An example of a simple influence path is given in Figure 9. The simple influence path starts at the source entity, which is the load, and ends at the target entity, which is the element. An influence path is composed of a series of relations that can be represented as a series of semi C-FAR matrices. Consider an influence path with n entities, n-1 semi C-FAR matrices and an initial change vector of Δ vector(1).

A change propagation is defined as the following set of multiplication:

Δ vector(2) = Δ vector(1) * C(Entity_1, Entity_2)
Δ vector(3) = Δ vector(2) * C(Entity_2, Entity_3)
.
.
Δ vector(n) = Δ vector(n-1) * C(Entity_n-1, Entity_n)

Δ vector(n) represents the change consequences of Δ vector(1) along a single influence path.

An example of the change propagation is given in Figure 9. A change in the influence of the load magnitude on the element length is examined. The example illustrates two semi C-FAR matrices along one influence path.

The change consequences of Δ vector(1) on the target entity is Δ [H*H+L*L].

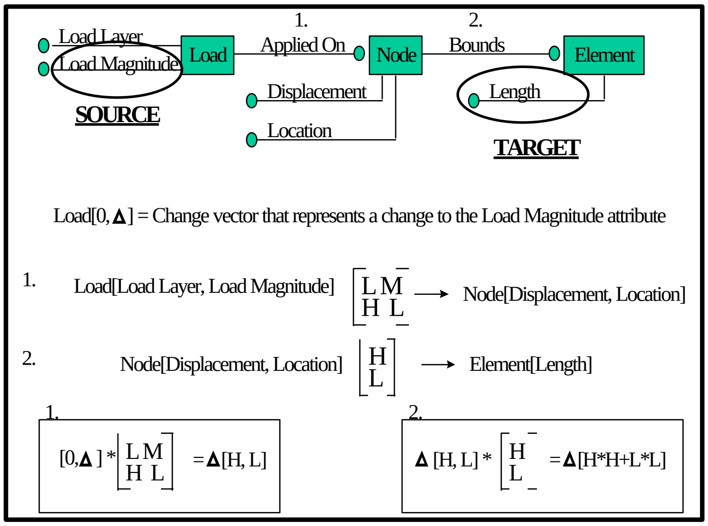
1.
2.
Load Layer
Load Magnitude
Load
Applied On
Node
Bounds
Element
Displacement
Location
Length
SOURCE
TARGET
Load[0,Δ] = Change vector that represents a change to the Load Magnitude attribute
1.
Load[Load Layer, Load Magnitude]
L M
H L
Node[Displacement, Location]
2.
Node[Displacement, Location]
H
L
Element[Length]
1.
[0,Δ] * L M H L = Δ[H, L]
2.
Δ [H, L] * H L = Δ[H*H+L*L]

Figure 9 Example, Simple Change Propagation

NUMERIC VALUES FOR LINKAGE VALUES

A sender entity is an entity along the influence path. A sender entity propagates the change vector through its attributes. All the entities along the influence path, except the target entity, are at one time sender entities. A receiver entity is also an entity along the influence path. A receiver entity receives the change through its attributes. A low linkage value ("L") means that a change in an attribute of a sender entity does not influence an attribute of the receiving entity. Considering this, the following are two propagation assumptions. A change vector element that is multiplied by the linkage value "L" is equal to "L". A change vector element that is added to the linkage value "L" is equal to itself.

The numeric value of a low linkage value is zero. Therefore, a minimum linkage value is low. The high ("H") linkage value means that a change in an attribute to a sender entity strongly influences an attribute of the receiving entity. The numeric value for the high linkage value has been chosen to be 0.9. The "M" linkage value means that a change in an attribute to a sender entity somewhat influences an attribute of the receiving entity. The numeric value for the "M" linkage value has been selected to be 0.3. The numeric choices for the "M" and "H" rankings were influenced by the recommendations in the House of Quality method [Clausing, 88], [Bahrami, 1992]. Choosing a value of 0.9 for "H" means that for each element with each propagation, the accumulated change effect is going down by a factor of at least 0.9. EXPRESS relates relevant entities with relations. Therefore, it is assumed that the longer the influence path, the less likely it is that there is a strong linkage between the source and the target, which explains the choice of 0.9 to represent "H." The value of "M" is 0.3 and is equal approximately to ten consecutive propagations of the "H" linkage value.

C-FAR IMPLEMENTATION

As seen in Figure 10, C-FAR implementation has two main stages: C-FAR construction and C-FAR usage. C-FAR construction is an activity which enriches the EXPRESS schema with knowledge from an expert domain. This knowledge is translated to a C-FAR matrix, which provides a qualitative linkage measure to relations in the EXPRESS schema. The second main stage of the C-FAR approach is usage. After the C-FAR schema is built, a user can query the schema and ask what the consequences are of a given change. In this section the C-FAR implementation is explained. First the C-FAR Construction is illustrated followed by a detailed explanation of C-FAR usage. This section also includes discussion about C-FAR algorithmic implantation.

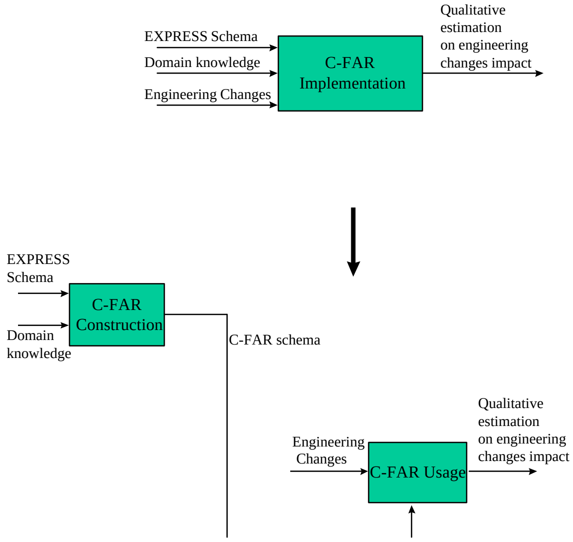

Figure 10 C-FAR Implementation

C-FAR CONSTRUCTION

C-FAR construction can be divided into two main stages. The first is the scoping stage, where the EXPRESS data schema is manipulated to facilitate incorporation of the C-FAR matrices. The second stage involves building a C-FAR matrix for any two entities that have a relation between them. Figure11

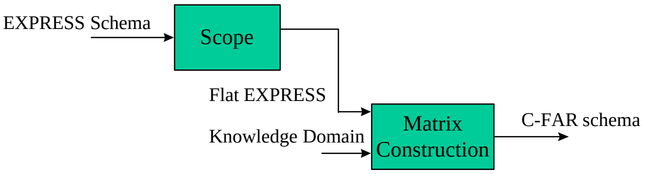

Figure 11 C-FAR Construction

EXPRESS SCHEMA SCOPING

The purpose of the scoping stage is to prepare a schema for C-FAR matrices. The preparation is made up of two main steps. The first step is to isolate the entities' attributes and the relations in the EXPRESS schema. An EXPRESS schema may also incorporate constraints and functions. In this stage, C-FAR does not use these components of EXPRESS.

A second main action in the scoping stage is 'folding' a supertype subtype relationship.

C-FAR MATRIX CONSTRUCTION

The C-FAR matrix is the main element that is added to the EXPRESS schema. Domain experts are responsible for building the matrices. Each C-FAR matrix encapsulates two semi-C-FAR matrices. A semi-C-FAR matrix represents how a change in any of the attributes of the sender entities influences the attributes of any of the attributes of the receiver entity. The second semi C-FAR matrix switches between the sender and the receiver matrices. The domain expert should evaluate each relation between two entities. For each attribute, the expert should ask himself how a change in this attribute influences any of the attributes of the reclining attributes. Specifically, the expert should first ask him or herself if a change in the attribute does not influence the receiving entity attribute. If the answer is positive, then the linkage value of "L" is attached to the relevant slot in the semi C-FAR matrix. If the answer is negative, the next question should be the following: Does a change in this attribute strongly influence the receiver entity attribute? If the answer is positive, then the linkage value of "H" is attached to the relevant slot in the semi C-FAR matrix. However, in case the answer is negative again, then the change of the attribute only somewhat influences the receiving entity attribute.

C-FAR USAGE

C-FAR construction is the initial stage of the implementation. The C-FAR usage articulates the capabilities and the scope of the methodology. Figure 6.7 illustrates the main parts of the C-FAR usage. The first two boxes represent pre-processing stages where the engineer interacts with the C-FAR schema to choose the relevant changeable objects according to desired engineering changes he or she wants to deploy on the current design that is reflected in the EXPRESS schema. The following two boxes in Figure 12 Find "Simple Paths" and "Calculate Linkage Value" represent a user transparent algorithmic part of the C-FAR usage. The last box, "Interpret Results", answers the user with an estimation on how a change in a given object may influence another object.

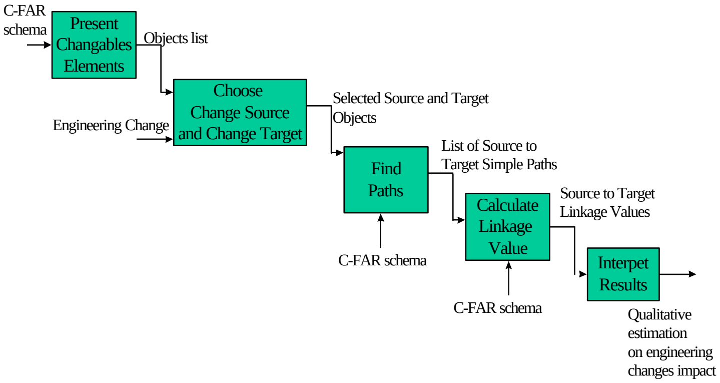
C-FAR schema
Present Changables Elements
Objects list
Choose Change Source and Change Target
Engineering Change
Selected Source and Target Objects
Find Paths
List of Source to Target Simple Paths
C-FAR schema
Calculate Linkage Value
Source to Target Linkage Values
C-FAR schema
Interpet Results
Qualitative estimation on engineering changes impact

Figure 12 C-FAR Usage Components

BUMPER MODEL CASE STUDY

This bumper case study concentrates on the bumper components and their relation to the bumper requirements.

Several factors are important in designing bumpers. Some of these factors are styling, weight reduction, corrosion resistance, reparability, engine cooling and cost. The bumper core is a beam, which can be steel or plastic laminates or reinforced thermoplastic beams with long glass fibers. Attached to the beam is an energy absorbent element. Its role is to take most of the energy from the impact. Energy absorbent material can be a foamed plastic or plastic honeycomb. The mounting brackets also take some of the impact load energy. Finally, the decorative bumper requirements are fulfilled by the bumper covers, or facia. The material must be able to flex without breaking or cracking during impact. The facia material can be thermoplastic olefin (TPO) or a material from the thermoplastic polyester elastomer family, or reaction injection molding.

More details about the bumper components can be found in [K. C. Rusch. "An Overview of Automotive Plastic Bumpers" ,1990]

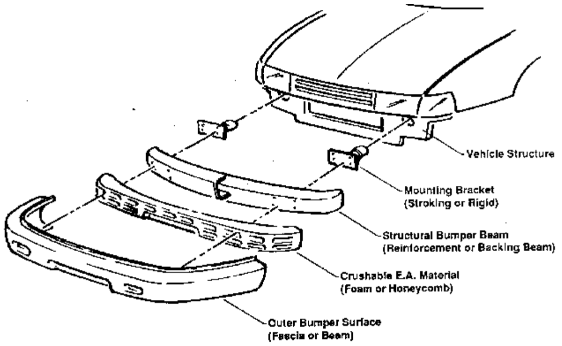

Figure 13 Bumper Components
(Credit K. C. Rusch. "An Overview of Automotive Plastic Bumpers" ,1990)

INFORMATION MODEL

In this case study, a short description of the model is given first, followed by a schematic information model. The EXPRESS model is then translated to a flat EXPRESS model. Its role is to capture the relations, entities and their attributes. Next, an explicit layout of the C-FAR schema is given, including C-FAR matrices and change scenarios. Finally, two examples of change scenarios are provided, followed by a case study summary.

In this case study, the EXPRESS model captures the main components of the bumper on the one hand and the bumper tests on the other hand. The bumper entity has a "is_compound_of" relationship to the "Bumper_Component" entity. The "Bumper_Component" entity is a supertype entity for the four main bumper components: "Energy_Absorbent", "Bumper_Beam", "Bumper_Facia" and "Bumper_Brackets". Also described in the schema are the two test type that are deployed on the bumper. Figure 14 is an EXPRESS-G diagram of the entities. A complete data and case study description is given in appendix C.   The EXPRESS schema is shown here:

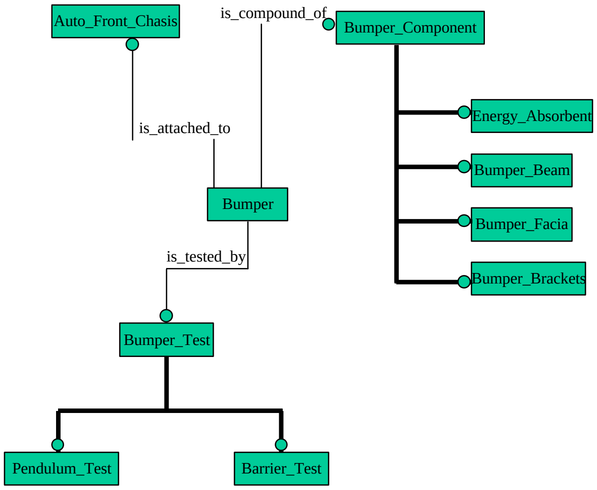

Figure 14 Bumper EXPRESS Model

C-FAR MATRICES

The following is a representative set of C-FAR matrices that are used in the scenarios.

En_Absorbent Bumper	weight	length	width	depth	abs density	absorber pattern
part_assembly number	L/L	L/L	L/L	L/L	L/L	L/L
weight	H/H	M/M	M/M	M/M	L/H	L/M
length	M/M	H/H	L/L	L/L	M/L	L/L
width	M/M	L/L	H/H	L/L	M/L	L/L
depth	M/M	L/L	L/L	H/H	M/L	L/L
height	L/L	L/L	M/M	L/L	L/L	L/L
color	L/L	L/L	L/L	L/L	L/L	L/L
offset	L/L	L/L	L/L	H/H	L/L	L/L
styling_req.	L/L	M/L	H/L	H/L	L/L	M/L
corrosion. resistance_req.	L/L	L/L	L/L	L/L	L/L	L/L
weight_reduction req.	H/L	M/L	M/L	M/L	H/L	M/L
damage_protection req.	H/L	M/L	H/L	H/L	H/L	H/L
engine_cooling req.	L/L	L/L	M/L	M/L	M/L	H/L
cost	L/H	L/H	L/H	L/H	L/H	L/H

Table 4 Bumper vs. Energy Abs. C-FAR Matrix

EVALUATION BUMPER MODEL SCENARIO

How would a change in the energy absorbent length influences the bumper beam attributes? In this case, there is only one simple path, and this path passes from the "Energy_Absorbent" entity through the "bumper" entity to the "bumper beam" entity.

The change vector is: Δ Energy_Absorbent change vector = [0 Δ 0 0 0 0]

The path: Energy_Absorbent - Bumper - Bumper_beam. C(*Energy_Absoebent*, *Bumper*) is extracted from Table4.

[0 Δ 0 0 0 0]* C(*Energy_Absoebent*, *Bumper*) =
Δ * [L M H L L L L L L L L L H] = Δ 1
Next, this vector is multiplied with
C(*Bumper*, *Bumper_Beam*) which extracted from Table 5.

Bumper_Beam **Bumper**	part number	weight	length	width	depth
part_assembly number	H/H	L/L	L/L	L/L	L/L
weight	L/L	H/H	M/H	M/H	M/H
length	L/L	H/M	H/H	L/L	L/L
width	L/L	M/M	L/L	H/H	L/L
depth	L/L	M/M	L/L	L/L	H/H
height	L/L	L/L	L/L	M/M	L/L
color	L/L	L/L	L/L	L/L	L/L
offset	L/L	L/L	L/L	L/L	H/H
styling_req.	L/L	L/L	M/L	H/L	H/L
corrosion. resistance_req.	L/L	L/L	L/L	L/L	L/L
weight_reduction req.	L/L	H/L	H/L	H/L	H/L
damage_protection req.	L/L	H/L	H/L	H/L	H/L
engine_cooling req.	L/L	L/L	L/L	M/L	H/L
cost	L/L	L/H	L/H	L/H	L/H

Table 5 Bumper vs. Bumper_Beam

Bumper_Beam(cont) **Bumper**	material code	beam profile	wall thickness	elas modul e	beam type
part_assembly number	L/L	L/L	L/L	L/L	L/L
weight	L/L	M/H	M/H	M/H	M/H
length	L/L	M/M	M/M	M/M	M/M
width	L/L	H/H	M/H	M/M	M/M
depth	L/L	H/H	M/H	M/M	M/M
height	L/L	L/H	L/L	L/L	L/L
color	L/L	L/L	L/L	L/L	L/L
offset	L/L	L/L	L/L	L/L	L/L
styling_req.	L/L	M/L	L/L	L/L	M/L
corrosion. resistance_req.	L/L	L/L	M/L	M/L	H/L
weight_reduction req.	L/L	H/L	H/L	H/L	H/L
damage_protection req.	L/L	H/L	H/L	H/L	H/L
engine_cooling req.	L/L	M/L	M/L	M/L	H/L
cost	L/L	L/H	L/M	L/H	L/H

Table 5 Bumper vs. Bumper_Beam (cont)

$$\Delta 1 * C(\textit{Bumper, Bumper\_Beam}) =$$
$$= \Delta * [L\ H\ H\ 0.09\ 0.09\ L\ 0.09\ 0.09\ 0.09\ 0.09]$$

RESULTS ANALYSIS

The results repress the influence of a change on the length attribute of the energy absorbent entity on the attributes of the bumper beam entity. Three classes of influence can be observed. The first class is the attributes that are strongly influenced by a change in the energy absorbent length. It is expected that as the bumper beam length will change, so will its weight .The second group of attributes have calculated linkage of 0.09 and the third attribute group have a linkage value of L. It is interesting to point out that the attributes that belong to the second group are describing physical characteristics of the bumper beam that are not directly related to the given change in the energy absorbent. The 0.09 linkage value hints that those attributes, like the bumper beam width or depth, are not as closely linked to the given change as the bumper beam length. However those attributes are more linked to the given change than to the part number attribute.

SUMMARY

The purpose of this paper is to introduce and demonstrate C-FAR's capability to model in the real engineering domain and to facilitate change representation and propagation mechanisms. The starting point of a case study was an existing EXPRESS model. The model was converted to flat EXPRESS format and was enhanced with C-FAR matrices. The C-FAR matrices were constructed by a person who is an expert domain. The domain expert also suggested a set of scenarios for each case study. The scenario role is to suggest a meaningful engineering change to the case study and test the change representation and propagation mechanism of the C-FAR methodology. Four case study were explored by the C-FAR methodology. A simple 2D truss, an automobile vehicle, PWB design and injection molding process. This paper focus was on the C-FAR basic methodology.

CONCLUDING REMARKS

This research attempted to aid the redesign process by introducing data driven change representation and a propagation mechanism. The main innovation element in this research was devising, and implementing a methodology to utilize existing information for representing change and its consequences. In increasingly complex engineering domain problems, data modeling has become an important aid to understanding and conveying the domain nature.

Information models are now being developed to support management of key engineering data. The results of this study show that such information models provide a global representation of the linkages among the various engineering components and as such, can be used to assess the propagation of engineering changes.

C-FAR's coverage and change representation is very dependent on the information model scope. There are two important points that may degrade C-FAR performance. First, an inaccurate EXPRESS model will consequently damage C-FAR capability to represent or reflect changes in a reasonably correct manner. Secondly, since C-FAR evaluates change by attribute values, the evaluation will only reach as far as the EXPRESS attributes allow.

ACKNOWLEDGMENTS

The Authors wish to thank Georgia Tech Dr. Sham Navathe and Dr. Leo Mark Dr. Suresh Sitaraman and Dr. David Rosen for their help in this research. Special thanks to Dr. Dennis Li and Dr. Marios Tesangaritis from General Motors for their help to this research. Considerable help in developing the case studies was done by Diego Tamburini, M.C Ramesh and Dr. Russell Peak, all from the Engineering Information System lab in Georgia Tech.

REFERENCES

A. Bahrami. "Computer Assisted Conceptual Design by Utilizing Fuzzy Logic and Quality Function Deployment," *Journal of Design Manufacturing*, vol. 2, 1992, pp. 155-165.

R. Banares-Alcantara. "Representing the Engineering Design Process: Two Hypotheses," *Computer Aided-Design*, vol. 23, no. 9, November 1991, pp. 595-603.

J. Banerjee, W. Kim, H. J. Kim and H. F. Korth. "Semantic and Implemantation of Schema Evolution in Object Oriented Database." In *Proc. ACM SIGMOD Conf*., 1987.

M.S. Bloor and J. Owen. "Product Data Exchange, The Next Step," *Computer Aided Design*, vol. 3, May 1991, pp.237-243.

M. Bouneffa and N. Boudjlida. "Managing Schema Changes in Object-Relationship Databases," In *OOER '95*, 1995, pp. 113-132.

W. Clancey "Acquiring, representing, and evaluating competence model." *In The nature of Expertise*. 1988

P. Coad and E. Yourdon. *Object Oriented Analysis*. Prentice-Hall, Englewood Cliffs. NJ, 1990.

P. Coad and E. Yourdon. *Object Oriented Design*. Prentice-Hall, Englewood Cliffs. NJ, 1991.

T. Cohen. "A Data Approach to Tracking and Evaluating Engineering Changes" A Ph.D. Thesis 1997.

T. Corman and C. E Leiserson, and R. L. Rivest "*Introduction to Algorithms*",
MIT Press and McGraw-Hill 1994.

Costin & Phipps "Racing and Sports Car Chassis Design" Cambridge 1971

M. R. Cutkosky and J. M. Tenenbaum. "Toward a Computational Framework for Concurrent Engineering," *IEEE Spectrum*, April, 1990.

C. Dagle and A. Kusiak eds. *Intelligent System in Design and Manufacturing.* ASME Press, New York , 1994. pp. 233-235.

C. L. Dym. *Engineering Design a Synthesis of Views*. Cambridge, 1994.

C. M. Eastman and N. Fereshetian. "Information Models for Use in Product Data Design: a Comparison," *Computer-Aided Design*, vol. 26, no. 7, Jul. 1994.

R. Elmasri and S. B. Navathe. *Fundamentals of Database Systems*. The Benjamin/Cummingd Publishing Company, 1994.

R. E. Fulton and C. P. Yeh. "Managing Engineering Design Information." In *AIAA'88 Aircraft Design, System & Operations Conf. AIAA*. USA,1988.

A. Gal and O. Etzion. "Handling Change Management Using Temporal Active Repositories." In *OOER '95*, 1995, pp. 378-387.

A. Ghali, and A Neville "Structural Analysis a Classical and Matrix Approach" 1985

T. Govindarej. "Qualitative approximation methodology for modeling and simulation" *IEEE Transactions on systems, Man and Cybernetics* 1987

L. J. Grigley and S. B. Billatos. "Case-Based Reasoning Approach to Concurrent Engineering." In *Intelligent Concurrent Design: Fundamentals, Methodologies, Modeling and Practice, ASME*, 1993, pp. 83-95.

L. Hunter. "Knowledge acquisition planning: Results and prospects. In. *Proceedings of the sixth international machine learning workshop*. San Mateo, CA: Morgan Kaufman

ISO TC184/SC4/WG12_ N022. Parametrics. Jun. 1997.

V. R. Jamalabad and N. A. Langrana. "Extraction and Re-Use of Design Information." In *Design Theory and Methodology ASME*, 1993, pp. 129-137.

S. M. Kannapan and K. M. Marchek. "A Schema for Negotiation Between Intelligent Design Agents in Concurrent Engineering," *Intelligent Computer Aided Design*, 1992.

R. M. Keller, C. Baudin, Y. Iwasiki, P. Nayak and K. Tanaka. "Compiling Re-Design Plans and Diagnosis Rules." In *Structure/Behavior Device Model in Knowledge Aided Design*, edited by M. Green. Academic Press Ltd., 1992, pp. 74-116.

A. Kemper and G. Moerkotte. *Object-Oriented Database Management*. Prentice-Hall, Englewood Cliffs, NJ, 1994.

W. Kim and F. Lochovsky, eds. *Object Oriented Concepts Database and Application*. Addison -Wesley, USA, 1989.

J. S. Landis. "Design Evolution and History in an Objected-Oriented CAD/CAM Database," *IEEE*, CH2285-5/86, 1986.

C. T. Liu, P. K. Chrysanthis and S. K. Chang. "Database Schema Evolution through the Specification and Maintenance of Change on Entities and their Relationship," *Entity Relationship Approach - ER*, 1994, pp. 132-151.

H. Liu, C. D. Row and W. Wen. "Design, Evaluation and Re-Design - Handling the Ad-Hockness of a Knowledge-Based Design System," *IEEE*, 0-8186-2690-9/92, 1992, pp. 267-271.

L. Liu. "Adaptive Schema Design and Evaluation in an Object-Oriented Information System." In *OOER '95*, 1995, pp. 21-31.

T. H. Liu and G. W. Fisher. 'An Approach for PDES/STEP Compatible Concurrent Engineering Application," *Concurrent Engineering Tools and Technologies for Mechanical System Design*, vol. 108, 1993, pp. 433-464.

S. Marche. "Measuring the Stability of Data Models," *The European Journal of Information Systems*, vol. 2, no. 1, 1993, pp. 37-47.

E. Miller. *PDM Today - The Current State of the Industry*. SIMdata, 1996.

H. Okumura and K. Okura. "H-SMC Bumper Beam Endured 5 mph Impact" In *Automobile Bumper System and Exterior Panels,*1990

P. Salvia. "Sensitivity of Physical Database Design to Changes in Underlying Factors," *Information and Management*, vol. 15, 1988, pp. 151-161.

J. Rumbaugh. "The Life of an Object Model, How the Object Model Changes During Development," *Journal of Object-Oriented Programming*, March/April 1994, pp. 25-32

K. C. Rusch. "An Overview of Automotive Plastic Bumpers" In *Automobile Bumper System and Exterior Panels*,1990

G. Samaras, D. Spooner and M. Hardwick. "Query Classification in Object-Oriented Engineering Design Systems," *Computer-Aided Design*, vol. 26, no. 2, Feb. 1994, pp. 127-136.

M. J. Saylor and S. Bansal. "Open Systems Decision Support," *Data Management Review*, Jan. 1995.

D. Schenk. *EXPRESS Language Reference Manual: External Representation of Product Definition*. ISO TC184/SC4/WG5 Document N14 International Standards Organization, 29 Apr. 1991.

A. J. Schwab, S. Bruno and K. Zinser. "Concurrent Engineering," *IEEE Spectrum*, 1993.

E. Sciore. "Multidimensional Versioning for Object-Oriented Databases,*"* *Deductive and Object Oriented Databases*, Dec. 1991, pp. 355-370.

J. Shah and P. Wilson. "Analysis of Design Abstraction, Representation and Interfacing Requirements for Computer Aided Design," *Des. Studies*, vol. 10, no. 3, 1989, pp. 169-179.

A. H. Skara, and S. B. Zdonik. "The Management of Changing Types in Object-Oriented Database." In *Proc. First OOPSLA*, 1996, pp. 483-494.

STEP Part 44, ISO-10202-44. Product Data Representation and Exchange. 1992

STEP Part 11, ISO-10303-11. EXPRESS Language. 1992.

STEP Part 1, ISO 10303-1. Overview. 1992.

J. B. Thompson and S. C. Y. Lu. "Design Evolution Management: a Methodology for Representing and Utilizing Design Rational." In *DTM '90 DE*, vol. 27, 1990, pp.185-191.

Y. H. Umeda, T. Takeda, H. Tomiyama and H. Yoshikawa. "Function, Behavior, and Structure Application of Artificial Intelligence in Engineering." In *Proceeding of the Fifth International Conference*, (V. J. Gero, ed.), Boston, July, 1990, Computational Mechanics Publications, pp. 177-193

D. G. Ullman. "The Evolution of Function and Behavior During Mechanical Design." In *Design Theory and Methodology, ASME*, 1993, pp. 91-103.

S. D. Urban, J. J. Shah and M. T. Rogers. "Engineering Data Management: Achieving Integration through Database Technology," *Computing and Control Engineering Journal*, June, 1993.