Proceedings of DETC'03
ASME 2003 Design Engineering Technical Conferences and
23rd Computers and Information in Engineering Conference
Chicago, Illinois USA, September 2-6, 2003

DETC2003/CIE-48241

SYSTEMATIC DESIGN METHOD FOR INFORMATION MODELING IN CAD/CAE

Sai Zeng[1], Edward J. Kim[1], Gregory M. Mocko[1], Angran Xiao[2], Russell Peak[3*], Farrokh Mistree[2]
[1]Engineering Information System Laboratory
[2]Systems Realization Laboratory
[3]Manufacturing Research Center
G. W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332-0405

ABSTRACT

As engineering systems are increasingly becoming more complex, the need for information models is growing accordingly. Extensive research is currently underway to develop engineering data management capabilities and to understand the role of information as a systems integrator. In order to develop information models more effectively, a systematic methodology is needed to better manage data and develop information models.

In the area of CAD/CAE/CAM applications, an information gap exists between design models and analysis models. To this end, a multi-representational architecture (MRA) is presented to facilitate the transformation of information from design models to various support analysis models. In this paper, our primary focus is on ABBs (Analysis Building Blocks) for solid mechanics and thermal systems that generate FEA (Finite Element Analysis) SMMs (Solution Method Models) to obtain their results.

Our focus in this paper is to investigate the effectiveness of the Pahl and Beitz methodology in developing the ABB information model. The Pahl and Beitz design methodology is intended for physical product design applications. Three of the four phases of the Pahl and Beitz methodology are examined and modified to facilitate development of the ABB information model. The augmentations of these phases are presented in this paper. The results of the development of concepts of ABB information model using the Pahl and Beitz methodology support the use of systematic design methodologies for the development of information models. The emphasis of this work is on the methodology used to develop the ABB information model rather than the technical result of the ABB model.

Keywords: Information Models, Engineering Design Methodologies, Systematic Design, Pahl and Beitz Methodology, Multi-representation Architecture, Analysis Building Block, Solution Method Model, Decision Support Problem.

1 FRAME OF REFERENCE

The development of new information communication technologies, especially Information Technology (IT), is forcing engineers to change their roles in product development, from the so called knowledge generators who collect information and make decisions, into information managers who receive, manipulate and organize vast amounts of information. This situation is more obvious in the domain of CAD/CAE/CAM. In this domain engineers have different usage views and requirements toward product information using discipline-oriented software. At the same time, each software application typically has a unique data format, as well as information processing approaches. Product information modeling is one of the key technologies facilitating information management and

** **Corresponding Author** Phone: 404-894-7572; Fax: 404-894-9342
Email: russell.peak@marc.gatech.edu

CAD/CAE/CAM software integration, which unfortunately, is accomplished by engineers using traditional, *ad hoc* approaches. For a complex product, engineers have to modify or even re-construct the product information model at each major step in the product development process; this soon becomes into a 'messy' work rendering errors, long product lead time, and high cost. For instance in our research area, product design and analysis integration, which is just a part of a product development process, a well known gap exists between design and analysis because of the different usage views towards designs and analysis models. We intend to bridge this gap by using a multi-representation architecture (MRA), introduced in detail in the next section, which realize the transformation of information between design and analysis models through four stepping-stone information representations, including analyzable product models (APM), context-based analysis models (CBAM), analysis building blocks (ABBs), and solution method models (SMMs). Each of these four models has to be constructed by engineers whose experiences decide the efficiency and quality of the information modeling activities. The situation in an entire product development process is much more complex than our case. Therefore, in order to relieve the burden of information management from engineers, as well as realize seamless integration of CAD/CAE/CAM activities, it is necessary to present a systematic methodology guiding information modeling activities.

The methods for developing information models are not widely known or well documented. However, systematic design methodologies are widely used in the physical product design, such as Pahl and Beitz methodology [1], Decision Based Design [2], axiomatic design approach [3], etc. We believe at least some of these methodologies can be used guiding information modeling, which generally is also a design process at the information level. In that, an attempt to explore this idea, we select the Pahl and Beitz methodology, a typical systematic design methodology has been used effectively in the area of mechanical design for decades. Hence, the question to be answered in this paper is.

> *Can a systematic design methodology, more specifically the Pahl and Beitz methodology, be used to facilitate the information modeling activities such as those in MRA?*

This research question is answered in the context of ME6101, an introductory graduate course in engineering design offered each fall at the Georgia Institute of Technology. This paper is a result of serendipity based on a semester-long design project. The Pahl and Beitz method is chosen as the method for exploring information modeling because it is presented as the baseline method for exploring and understanding systematic design. In turn, students are challenged to internalize, augment, personalize, and extend the Pahl and Beitz method for their domain of interest. These domains range from traditional machine design to, in this case, developing an information model. In no way do we assert this is the best or only method for developing information models. However, we present this research to demonstrate how engineering design methods can be systematically modified. Further details are provided in Chamberlain and coauthors [4]. Our course project is in part the motivation for the development of the ABB information model concept using Pahl and Beitz methodology. As one of the information models in the MRA framework, ABB model represents the analytical usage view for analysis engineers, such as concepts of solid mechanics and thermal system. The ABB model characterizes semantically rich, reusable, modular, and tool-independent entities. In using the Pahl and Beitz methodology to develop the ABB information model, modifications and augmentations to the methodology are used to fulfill the design needs for information modeling.

In this paper, we provide answers to the proceeding research question, in the context of developing the ABB information model. In Section 2, background domain knowledge of information modeling is given. In Section 3, the original structure of Pahl and Beitz and the necessary augmentations needed for developing an information model are presented. In Section 4, the results and observations of this study are addressed. Finally closing comments and potential future research are addressed in Section 5.

2 INFORMATION MODELING

A significant gap typically remains between computer-aided design (CAD) and computer-aided engineering (CAE) due to the fact that engineers have different usage views towards product information models. A recent survey of design-analysis integration practice and research highlights the following needs [5]:

- *General methodologies for automating routine analysis to support product design*: Methodologies are lacking for creating CAE systems that provide designers with product-specific tools while taking advantage of general-purpose analysis tools.
- *Representation of design-analysis associativity.* Design-analysis integration requires capturing how a CAE model is related to a CAD model, both for creating the analysis model and for associating analysis results back with the design model.
- *Support for numerous diverse analysis models* for each product type. The same kind of product often has analysis models from a variety of engineering disciplines that involve different solution techniques. Even within the same discipline, analysis models of varying resolution and complexity can exist for the same analysis problem. The unifying factor among these numerous analysis models is the product itself. Hence, the product information used by these analysis models should ideally come from a common source to maintain consistency and support analysis automation.

Targeting the needs of design and analysis integration a general methodology for automating ubiquitous analysis to support product design is developed by Peak and coauthors [5, 6, 7]. In this methodology, the multi-representation architecture (MRA) is presented to facilitate heterogeneous transformations by explicitly representing design-analysis associativity and supporting numerous diverse analysis models for each product type. The MRA consists of four stepping-stone information representations, specifically the analyzable product models, context-based analysis models, analysis building blocks (ABBs), and solution method models (SMMs). ABBs and SMMs are product-independent models that facilitate generalized mappings between a single product model and diverse analysis models. ABBs describe the theoretic physical systems, such as continuum mechanics systems, while SMMs represent ABBs in relatively low-level solution technique form, such as finite element analysis models.

The MRA is illustrated using a solder joint thermo-mechanical analysis example in Figure 1. On the right side is a solution method module (SMM), marked with ①, which represents an analysis model in relatively low level and solution method specific form. An SMM combines solution tool inputs, outputs, and control into a single information entity to facilitate automated solution tool access and results retrieval. Analysis building blocks (ABBs), marked with ②, represent analytical engineering concepts in a manner that is largely independent of product application and solution method. ABBs obtain results by generating SMMs through transformations, ${}_{ABB}\Psi_{SMM}$, that are based on solution method considerations. Analyzable Product Models (APMs), marked with ③, represent detailed design-oriented product information. An APM is considered the master description of a product that supplies information to other product life cycle tasks, including engineering analysis and manufacturing. To enable its usage by potentially many analysis applications, an APM in the MRA goes beyond its traditional design role by supporting idealizations that relate detailed design-oriented attributes with simplified analysis-oriented attributes. Finally, a context-based analysis model (CBAM), marked with ④, contains linkages that represent design-analysis associativity between an APM and an ABB model, ${}_{APM}\Phi_{ABB}$. These associativity linkages indicate the usage of idealizations for a particular analysis application (e.g. solder joint deformation). Thus, CBAMs show how product independent ABBs are supplied with design-related information to help solve product-specific analysis problems.

From the MRA viewpoint, providing solutions to the design-analysis integration problem involves defining these four representations (SMMs, ABBs, APMs, and CBAMs) and two inter-representation mappings (${}_{ABB}\Psi_{SMM}$ and ${}_{APM}\Phi_{ABB}$).

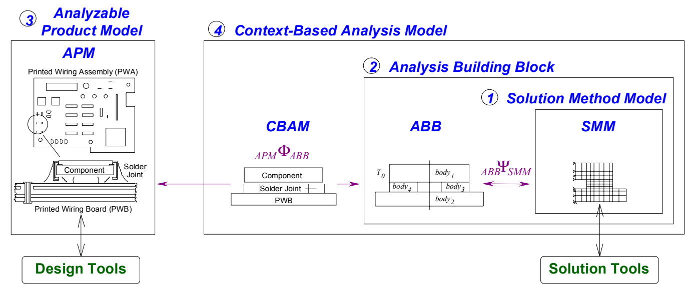

Figure 1. Multi-Representation Architecture (MRA) [5, 6,7]

For the purposes of developing an information model using a systematic design methodology, the focus is on the development of the ABB information model concept. ABBs represent the analytical usage view for analysis engineers. ABBs represent product-independent analysis concepts such as continuum mechanics bodies and idealized interconnections as semantically rich, reusable, modular, and tool-independent objects.

3 THE PAHL AND BEITZ METHODOLOGY – A BASELINE METHOD

In developing concepts for the ABB information model using systematic design, the Pahl and Beitz methodology is chosen to be used as the baseline method. The Pahl and Beitz method is selected for two primary reasons. First, the method is presented as the baseline method and introduction to systematic design in ME6101. Second, the method is well-known and commonly accepted systematic design approach. However, the methodology as it stands in its original form requires augmentations that make it suitable for developing information models. The Pahl and Beitz methodology is composed of four primary phases. Within each phase, various steps are to be followed to properly complete the phase. A brief summary of the phases follows:

1. *Planning and Clarification of Task* – During this phase, the product planning, analysis of the market/company, and product proposal are developed. In addition, the requirements and corresponding requirements list are generated from various constraints and design desires.
2. *Conceptual Design* - This design phase determines the principle solution. In this phase we abstract the essential problems, establish function structures, search for suitable working principles and combine these into a working structure. This leads to "specification of principle" or concept.
3. *Embodiment Design* – During this phase, a concept (working structure, principle, solution) is elaborated into the construction structure of a technical system in line with technical and economic criteria.
4. *Detail Design* – In this phase the arrangement, forms, dimensions, and surface properties of the individual parts are finally laid down.

Despite its many strengths, the Pahl and Beitz methodology still has limitations. Our efforts are concentrated on the limitations that would inhibit the information modeling design process. The Pahl and Beitz methodology is primarily intended for traditional, sequential mechanical engineering applications in design. Many of the steps within the various phases do not apply directly to a software design type of application. The requirements list headings, function structures, working principles, concept selection process, and preliminary layout design are among the limitations we choose to address in our augmentations of the Pahl and Beitz methodology as shown in Figure 2.

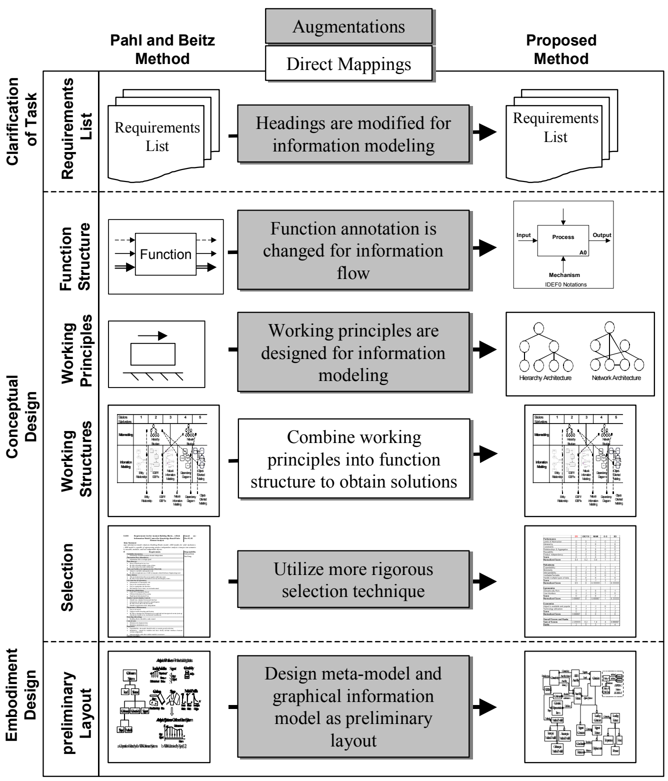

Figure 2. Augmentations of the Pahl and Beitz Methodology

In the section that follows, a more detailed summary of our proposed augmentations is presented.

4 AUGMENTATIONS

4.1 Augmentation in Clarification of Tasks

In the Clarification of Tasks phase, the original requirements list headings are augmented to support the information modeling-specific requirements identification (Figure 2).

The requirements list, including its qualitative and quantitative information, serves as the basis for formulating general and specific design guidelines and standards for product design. It is useful to draw up the requirements list based on a headings of functional and general areas of operation, such as the requirement list headings proposed by Pahl and Beitz.

On the basis of system and available technology analysis, functional and operational differences between the 'soft' information model design and 'hard' machine design arises the concern. Rigorously adopting the Pahl and Beitz headings is not appropriate to aid in the ABB development. It is necessary to personalize and augment the requirements list headings, targeting the final design products as computable information models. The designed requirements list headings are classified as personalized items that are directly mapped from the headings from Pahl and Beitz, and additional items, which are absent from Pahl and Beitz headings. By identifying the requirements list headings, it is possible to understand particular characteristics of the intended solution. Throughout the stages of the development of the ABB, the requirements list remains a living document. As progress is made during the other phases of the Pahl and Beitz methodology, amendments and corrections are made to the requirements list.

Due to the nature of the requirements of an information model, the components of the requirements list are primarily qualitative in nature. Beyond the scope of this particular design project, additional work will be done on developing and enhancing ABB models. For this reason, a detailed description of the reasoning behind the components of the final requirements list headings is given below. The headings for the augmented requirements list are those for the intended use for information modeling development. The parenthesized items are the corresponding original requirements list headings of the Pahl and Beitz methodology.

Scalability (Geometry) -The information should be product domain independent.
Information Flow (Kinematics) - The information flow of the data stream should be of a high quality. This requirement will be dependent on the intended customer's use.
Data (Energy) - The information model must be able to handle multiple types of data. Also, the information model should be able to identify data relationships and aggregates.
Tools and Facilities for Implementation (Materials) - The information model must be able to be implemented as a Computer-Aided Software Engineering tool. An additional requirement would be the ability to adjust to commercially available and popular information tools.
Safety - The integrity and information security of the data transferred and process must be maintained at a high level of security. The information model must also protect against malicious users.
User Interface (Ergonomics) - The information model must be semantically rich and easily customizable by the end users.
Quality Control - The information model must be testable and the information definition should be unambiguous.
Recycling - The information model must be able to be reused an infinite number of iterations.

Costs - The development time for the ABB should be minimized. The cost for implementation by the end users must not exceed unreasonable limits, deemed by the end user.
Robustness - The modularity of the ABB model is necessary. The information model should have clearly defined interfaces between various components.
System (Assembly) - The information model must be able to describe different levels of abstraction. Also, the model must be able to break down hierarchically into more detailed descriptions.

Based on the augmented requirements list headings, we develop a requirements list appropriate for the ABB information model. This requirements list serves as a key document to guide subsequent design.

4.2 Augmentations in Conceptual Design

Augmentations in Conceptual Design phase include function annotation design, working principle design and the inclusion of an alternative selection technique (Figure 2). The explanations of each augmentation are illustrated in following sub sections.

4.2.1 Function Decomposition

Function decomposition reduces the complexity of the problem. Pahl and Beitz produced the one of the most influential works on functional decomposition, in which they proposed a method of specifying functions and their relations to one another in terms of a transfer of materials, energy, and signal. For the purposes of developing an information model, the functionality is more than just the transformation of materials, energy, and signal. Our functional establishment is based on flow of information. The input and output of the system is identified as analysis concept of solid mechanics and computable ABB model respectively, according to the crux of the problem identified. In order to link the input and output, the overall function is defined as an information modeling process, which transforms the analysis concept into computable format – the ABB model. The key constraint for this function is time. Additionally, the tools that support the functional realization are the computer-aided software engineering (CASE) tools. Ideally, all these elements are present in the preliminary function structure. But the function structure notations defined in Pahl and Beitz methodology are not appropriate to represent what is needed to incorporate into the function structure due to the fact that they are specifically designed for physical product and represent the flow of energy, materials and signals. Therefore, we propose an alternative well-known function representation technique, which serves for the same purpose as for function structures, namely Integrated DEFinition 0 (IDEF0).

IDEF0 is used to specify the functions, which show the high-level activities of a process. IDEF0 modeling starts with the most abstract level of activities. The activities will be decomposed further down into various levels of details until the desirable resolution obtained. The notation of IDEF0 (Figure 3) composed activity box and associated ‘data’ arrows, which represent input, output, control and mechanism information. Input is information such as resources consumed or transformed by a process. Output is the transformed result from the process input. Controls are the standards, guidelines and policies that govern the process. Mechanisms are the agents that accomplish the action; it may be the person, hardware tools, software tools, etc.

By using the IDEF0 modeling technique, the overall function structure is shown in Figure 3. In order to reduce the complexity of the function and increase the relative transparency of the relationships between inputs and outputs, the overall function is divided into sub-functions. To accomplish this, analysis is carried out to determine how deep the sub-function will be, what sub-functions involved for each level will be, and how to divide the information flow for different sub-functions at different levels. By working on search and establishment of an optimum function structure, the identification of the sub-function for the first level is shown in Figure 4

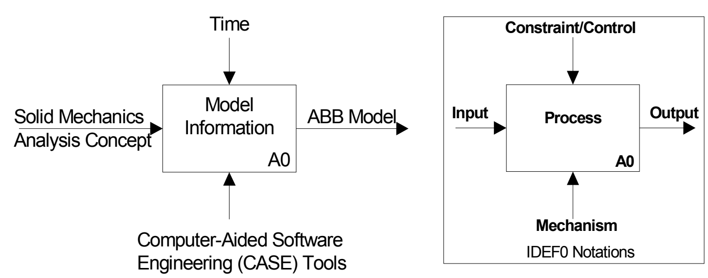

Figure 3. Overall IDEF0 Model

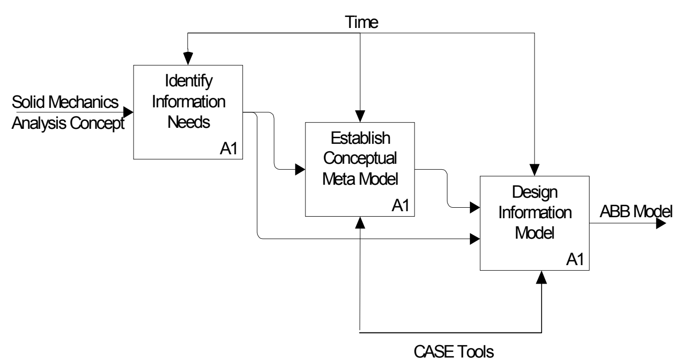

Figure 4. First-level IDEF0 Sub Model

4.2.2 Working Principles

Working principles defined in the context of Pahl and Beitz include the physical process along with the necessary geometric and material characteristics. Again, these working principles are dependent on the flow of energy, materials and signal. The working principles feasible in this context should support information flow and information relationships.

The information transformation from solid mechanics concepts into computable information involves step-by-step transformation, namely from abstract to concrete. The sub functions, as seen in Figure 3, represent such processes. In order to transform information, the search for working principles to fulfill sub functions must be performed and be combined eventually to concretize the working structures.

The conceptual meta-model is established based on working principles established from intuition and a comprehensive literature search. To search for the working principles to establish the conceptual meta-model, a literature search is implemented, and intuitive-based methods are applied. The basic capability of the conceptual meta-model is that it can represent the information as well as the relationships of the information. The architectures to support this level of modeling are mainly the hierarchic architecture and network architecture as shown in Figure 5. These two architectures are the working principles to establish the meta-models.

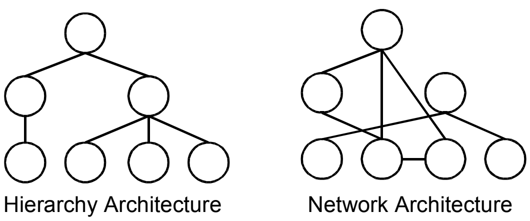

Figure 5. Hierarchic Architecture and Network Architecture

The working principles of information modeling are reviewed in Section 4.2.4. Those are representative working principles, which are the most appropriate for this context. They are Entity Relationship Diagram (ER), Integrated DEFinition 1 (IDEF1/IDEF1x), Nijssen's Information Analysis Modeling (NIAM), Dependency Diagram, and Object-Oriented Model.

By identifying the possible working principles for each sub function, the combination of the working principles together to form the solutions are presented in the following section.

4.2.3 Working Structures

To fulfill the overall function, the working principles of sub functions need to be combined together to elaborate the overall solutions. The combinations are not as simple as adding working principles together. In order to ensure the smooth information flow, the compatibility of the combinations is demanded. The combination of both sub functional principles involves information mapping. Therefore interest of compatibility moves to compatibility of information mapping which can be evaluated by two criterions: 1) Allow direct information mapping as much as possible between sub functions. 2) Information model should represent the meta architecture as well as the meta information.

The morphological matrix (Figure 6) is used to systematically combine working principles to develop working structures, or the solution variants.

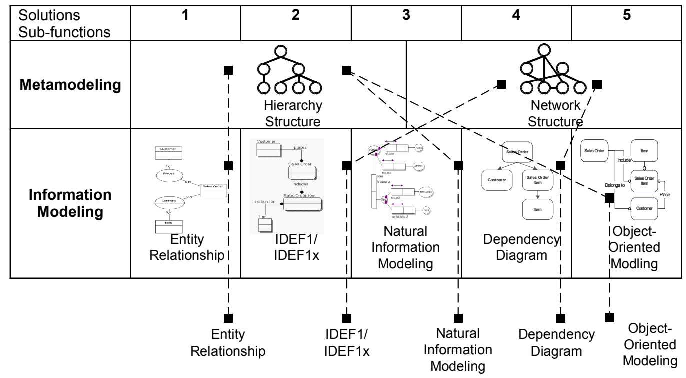

Figure 6. Morphological Matrix

Generally speaking, the hierarchy structure is a simplified network structure, whose architecture is easily represented. On the other hand, simple architecture means provisions of simple relationships, which may not be enough to capture the complicate information system in the real world. But, the specialty of such circumstance is that if an information model supports mapping from a meta-model with a network structure, it will also support mapping from meta-model with a hierarchy structure. The inverse convention is not true. So if it is assumed that the information system of solid mechanics can be represented enough by the hierarchic meta-model, the compatibility will be satisfied by any combination, if not, the compatibility will be confined to specific combinations. Based on our knowledge, we realize the former assumption stays true. Consequently, the five solution concepts emerge for this step. They are Entity Relationship Diagram, IDEF1/IDEF1x, Natural Information Modeling, Dependency Diagram and Object-oriented Modeling. The simple review of solution concepts and selection of the solutions concepts will be introduced in next session.

4.2.4 Selecting Suitable Working Structures

Development of working structures involves selecting and indicating preference activities in the Pahl and Beitz methodology. The tool designed for selection in the Pahl and Beitz methodology lacks of addressing the following needs:

- Quantitative concept-to-concept comparison.
- Ranking the concepts
- Interactions between difference concepts

Consequently, the selected concept may not be the most-likely-succeed concept. To address those needs, a technique for making selections in a complex, multi-faceted design environment is proposed by Mistree and colleagues [8]. As an augmentation to the Pahl and Beitz systematic approach, the Preliminary Selection Decision Support Problem (DSP) Technique is used as the selection technique for our design. The representative iteration steps to select the most-likely-to-succeed concept are overviewed below. The technique involves comparing multiple datums, where each concept is set as the datum and compared to dispel any prejudice. Additionally, multiple weighting schemes are employed to address the interaction of the selection criteria. We present a brief discussion of the Preliminary Selection DSP Technique in this paper. A detailed explanation of the algorithm is presented in [8].

Step 1: Describe alternatives and providing acronyms.

A total of five concepts are generated. Below are the concepts along with a brief description and accompanying acronym.

Entity Relationship Diagram (ER) [9] – Entity Relationship diagramming provides a viewpoint of data entities and their associated relationships. The beauty of this method is its simplicity, and its representation of information entities in a “real world” way. Also, it is relatively straightforward to map ER diagrams into a relational database design.

Object Oriented Model (O-O) [10] – The Object Oriented modeling method is one in which the data are grouped into packages which occur “naturally”; this makes it compatible with the way people think about the real world. It allows for data abstraction, inheritance, information hiding, and dynamic binding.

IDEF1/IDEF1X – The IDEF1/IDEF1 [11] method is similar to ER conceptually, but is different graphically, and more complicated semantically.

Nijssen’s Information Analysis Modeling (NIAM) [12] – The NIAM methodology is another data modeling method, showing objects and their relationships; constraints, however, can be modeled in this method.

Dependency Diagrams (DD) [10] – The Dependency Diagram is a combination of a dependency list and diagram, in which the dependencies of data elements to each other are described.

Step 2: Provide generalized criterion with acronyms and weighting constants for the specific criteria.

The criteria selected needs to be independent and thorough. The process must be based on the requirements list developed during the Clarification of Task phase. Also, our preferences and past design experience influenced our selection of the criteria. The development of our selection criteria is an iterative process. The criteria, explanation, and preferences are shown below in Table 1 through Table 3.

Table 1. Performance Criteria, Explanations and Preference

Performance	Explanations	Preference
Levels of Abstraction	This is the ability of the model to describe levels of detail.	Multiple levels of abstraction is preferred
Hierarchy	This is the ability of the model to break down hierarchically into a more detailed description.	A high level of Hierarchy structure is desired
Constraints	This is the ability to describe and incorporate any process and information constraints that exist.	No limitations of kind of constraints or number of constraints.
Relationships and Aggregates	This is the ability to describe groupings of information and materials and relationships between information and materials.	The model should support both relationships and aggregates
Reusability	This is the ability for the model to be used multiple times in different applications	Must be reusable for a high number of cycles
Product independency	This is the ability of the model to be independent from any particular product and be suitable for generic use.	The model must be completely independent of all products

Table 2. Robustness Criteria, Explanations, and Preference

Robustness	Explanations	Preference
Expandability	This is the ability for the model to be further developed as changes are needed	The model must be able to expand over the course of time
Modularity	This is the ability for the model to be modularized into subcomponents and parts for easy of use and development.	The model must be modular to minimize development time and effort
Interoperability	This is the ability for the model to be used with multiple types of software	The model must be compatible with multiple types of software
Verifiable/Testable	This is the ability to verify and test that the model is essentially a correct representation of reality	The model must be able to verify results
Handle multiple types of data	This is the ability to handle a wide range of data types	All common types of data must be handle by the model

Table 3. Ergonomics and Economic Criteria, Explanations, and Preference

Ergonomics	Explanations	Preference
Semantically Rich	This is the ability for the model to be full of semantic information	The model should be very semantically rich
User Interface	This is the ability of the model to have a easy to follow user interface	The model should have a easy to use interface
Economics	**Explanations**	**Preference**
Adjust to available and popular tools	This is the ability for the model to adjust to available and popular tools	The model should be able to adjust to available and popular tools
Technology utilization	This is the ability for the model to use existing technologies such as CASE tools	The model should be compatible with existing technologies

Step 3: Choose a datum with which all other concepts will be compared.

The Entity Relationship Diagram is chosen as the initial datum for comparing all concepts. There is not a particular reason for choosing this concept over the remaining concepts. The ER diagram is by our engineering judgment a strong candidate for development. However, the multiple iterations of the DSP process will make the initial choice frivolous.

Step 4: Compare the concepts.

A comparison is performed between each of the remaining concepts and the datum concept, the ER diagram. Several iterations are made with the "most-likely-to-succeed" concept of each iteration serving as the next datum. The concepts are evaluated against the datum with a –1, 0, +1 scoring scale. If the concepts are equal, a value of 0 is given. If a concept is superior to the datum, then a score of + 1 is given. If the concept is inferior to the datum then a score of –1 is given. The scores are summed and normalized. From the sum of the scores, a rank is given. A sample comparison matrix is shown in Table 4.

Table 4. Comparison Matrix for Entity Relationship Datum

	ER	IDEF1X	NIAM	O-O	DD
Performance					
Levels of Abstraction	0	0	-1	0	-1
Hierarachy	0	-1	0	1	0
Constraints	0	1	1	1	0
Relationships & Aggregates	0	0	0	0	0
Reusability	0	-1	0	1	-1
Product independency	0	0	0	0	0
Score	0	-1	0	3	-2
Normalized Score	0.4	0.2	0.4	1	0
Robustness					
Expandabliity	0	-1	0	1	-1
Modularity	0	-1	0	1	-1
Interoperability	0	0	0	0	0
Verifiable/Testable	0	-1	-1	0	1
Handle multiple types of data	0	0	0	1	0
Score	0	-3	-1	3	-1
Normalized Score	0.5	0	0.33333	1	0.33333
Egronomics					
Semantically Rich	0	-1	0	0	-1
User Interface	0	-1	0	1	0
Score	0	-2	0	1	-1
Normalized Score	0.66667	0	0.66667	1	0.33333
Economics					
Adjust to available and popular	0	-1	-1	0	-1
Technology utilization	0	-1	-1	1	-1
Score	0	-2	-2	1	-2
Normalized Score	0.66667	0	0	1	0
Overall Scores and Ranks					
Sum of Scores	2.23333	0.2	1.4	4	0.66667
Ranks	2	5	3	1	4

Step 5: Evaluate the merit function for each concept with each generalized criterion.

The summed score and resulting normalized score served as the merit function for each generalized concept. Ranks are assigned based on the summed score of all of the normalized scores for each criterion. The results are presented in Table 1.

Step 6: Include interactions between generalized criteria.

In order to see the interaction between the generalized criteria, different weightings scheme are used. The first four scenarios are based on a single dominant generalized criteria. The fifth scenario is an educated guess on the true weights of the criteria. The scenarios are shown in Table 5. The normalization scheme presented in Mistree and colleagues [8] is used.

Table 5. The Weighting Scheme Scenarios

Generalized Criteria	One	Two	Three	Four	Five
Performance	0.4	0.2	0.2	0.2	0.3
Robustness	0.2	0.4	0.2	0.2	0.3
Egronomics	0.2	0.2	0.4	0.2	0.2
Economics	0.2	0.2	0.2	0.4	0.2

Normalized scores for each concept are computed based on the merit function values in Table 4 and the weighting scenarios in Table 5. The normalized scores for each concept are computed by multiplying the score for each concept by the weighting values. The results from the ER datum are shown in Table 6.

Table 6. Merit Function Values of Concepts for ER Datum

	Scenario Number				
Concept	One	Two	Three	Four	Five
ER	0.527	0.040	0.413	0.900	0.167
IDEF1X	0.080	0.040	0.040	0.040	0.060
NIAM	0.360	0.347	0.413	0.280	0.353
O-O	0.900	0.900	0.900	0.900	0.900
DD	0.533	0.400	0.400	0.333	0.467

From the merit function values, a ranking for each of the scenarios is completed. Based on the numeric values in Table 6, a rank ordering is completed. The concepts are ranked based on higher merit function values. The results of the rank ordering for the ER as the datum are presented in Table 7. The results are presented graphically in Figure 7.

Table 7. Ranking of Concepts for The ER Datum

	Scenario Number				
Concept	One	Two	Three	Four	Five
ER	3	4	2	1	4
IDEF1X	5	4	5	5	5
NIAM	4	3	2	4	3
O-O	1	1	1	1	1
DD	2	2	4	3	2

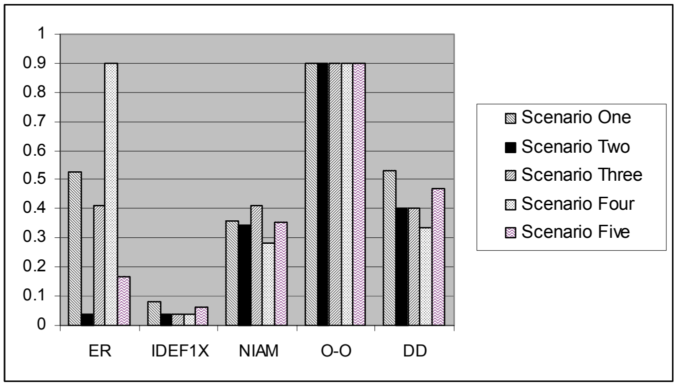

Figure 7. Graphical representation of preliminary selection for ER datum

The results presented in Table 6 and Figure 7 indicate that O-O model is the most likely to succeed model for all weighing scenarios when ER is set as the datum. Since the O-O concept consistently is ranked the highest for all weighting scenarios. It is chosen as the next datum. Step 6 is iterated in a similar manner using multiple datums for all weighting scenarios.

Step 7: Post-solution analysis: determine the most likely to succeed concept(s).

In Table 6, it is shown that the O-O concept is the most likely to succeed concept for all scenarios. However, it is too soon to declare that the O-O modeling is the most likely to succeed concept. To determine the most likely to succeed concept a comparison must be based on multiple datums. .selecting a concept to further develop is based on comparison to all datums. The average overall merit functions for each of the concepts is computed for all weighting scenarios are shown in Table 8.

Table 8. The Overall Merit Function for Preliminary Selection

	Scenario Number				
Concept	One	Two	Three	Four	Five
ER	0.503	0.180	0.393	0.867	0.128
IDEF1X	0.193	0.180	0.147	0.213	0.187
NIAM	0.331	0.360	0.393	0.271	0.346
O-O	0.900	0.892	0.892	0.867	0.892
DD	0.500	0.356	0.344	0.300	0.428

The values in Table 8 are computed using the simple mathematical average for all datums computed. These values are then rank ordered. The rankings based on the highest average merit functions. The results are presented in Table 9 for the most likely to succeed concepts.

Table 9. Overall Rankings for the Most Likely to Succeed Concepts

	Scenario Number				
Concept	One	Two	Three	Four	Five
ER	2	4	2	1	5
IDEF1X	5	4	5	5	4
NIAM	4	2	2	4	3
O-O	1	1	1	1	1
DD	3	3	4	3	2

In Table 9, it is shown that O-O modeling is the most likely to succeed model consistently across weighting scenarios and across the multiple datum. The object-oriented information model, is firmed up into a more detailed information model in the following section.

4.3 Embodiment Design - Model Development

After the modeling concept has been selected, the embodiment design is carried out to develop model. We implemented conceptual meta-modeling and information modeling to develop ABB model step by step (Figure 2). Meta-modeling is close to human thinking process relative to information modeling. It is in a relative simple and freeform format. We use meta-modeling to help us abstract and organize the solid mechanics concepts and turn these concepts into a more concrete information model.

From meta-model to object-oriented information model, the mapping processes are necessary to transform the concept into the objects. The obtained information model should capture the architecture of the meta-model rigorously. Also, it should provide the complete information content to describe the solid mechanics systems. The information model establishment is explained in Section 4.3.1 and Section 4.3.2.

4.3.1 Establish Conceptual Meta-model

An ABB model represents engineering analysis concepts as a set of computable information entities, which are independent from specific solution techniques [5]. In the context of solid mechanics, ABB concepts are presented in Figure 7. To facilitate representing a variety of solid mechanics systems, ABB information content is categorized by composition in Figure 7a. A solid mechanics system consists of two key components: idealized structure and idealized loads. Structure represents any assembly of objects that supports or transmits loads, *e.g.* idealized building structure, aircraft, vehicles, etc. Loads represent active forces that are applied onto the structure because of external causes, *e.g.* pressure, vibration, temperature, and etc. Both these aspects in the composition hierarchy are necessary to completely represent a solid mechanics system.

At the next level, structure is composed of individual continua, and the interrelations between those continua are described using connectivity concepts (idealized interconnections). For instance, slip bonding between two continuum entities indicates the condition that the two continuum entities are in contact, while only relative displacement along the contact interface is allowed. Relative displacement interrelations between a structures and its environment are identified as support constraints such as rigid support, pin support, etc. ABBs are categorized by types into several levels, including analysis primitives that are used in building intermediate ABBs and analysis systems (Figure 7b). For instance, shape and material models are primitive ABBs; they are combined together with other ABB concepts to represent a continuum ABB – an intermediate type of ABB primitive. An ABB system as shown in Figure 7b for a cantilever beam analysis system is formed by assembling ABB primitives such as a loading force, a continuum beam, and a rigid support.

In the composition hierarchy given in Figure 7a, the leaf nodes denote analysis primitive categories at the levels where they can be easily changed or reused in a plug-and-play manner, and the root node represents an ABB analysis system. Hence, an ABB model is composed of fundamental building blocks to represent all the necessary information in an analytical sense.

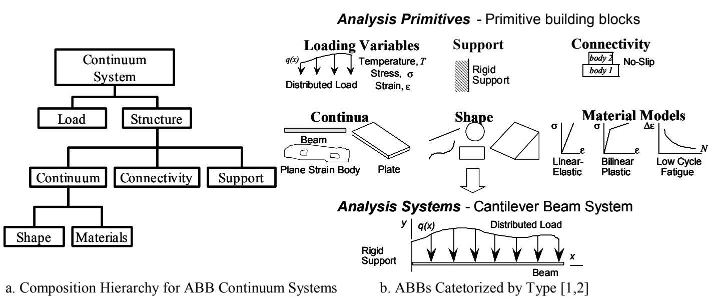

Figure 8. Information Content for Example ABB Concepts

4.3.2 Establish Graphical Information Model

The ABB model of a continuum system can be represented using constraint objects (COBs) [13] [14]. The COB representation is a knowledge representation that employs object technology and constraint graph concepts. This modeling language provides modularity, reusability and multi-directionality and is closely matched with the way engineers interpret their interaction with an idealized environment. One of the COB representation components is the graphical Object Relationship diagram presented using Express-G notation from the ISO 1030-11 information modeling standard [15]. Diagram in Figure 9 illustrates various types of ABBs, their attributes, and their object hierarchy at the template level, i.e., not at the instance level.

The partial ABB information model shown in Figure 9 is influenced by the work of STEP Part 42 and AP210 [15]. In the ABB composition hierarchy of Figure 7a, the root node, continuum system, is mapped to the object type *ABB Assembly* in Figure 9. The main attributes of this object are *Connectivity*, *Loads* and *Supports*. Continua can also be defined, and they are joined in an ABB system by *Connectivity* objects. By mapping each item in the ABB concept hierarchy of Figure 7a into a corresponding object type in Figure 8, the information content for a class of solid mechanics systems is fully described. This Express-G model can be formulated as a lexical Express model, which can then be turned into computable data structures.

The *Assembly Feature* object type represents the region on/in the specified continuum where an interconnect, load, or support is applied. Its attribute *Definition* indicates the shape of the applied region, which can be a solid, face, edge, or vertex. This attribute is an object of type *Geometric Representation Item* (from STEP Part 42) that helps to identify the location of an interconnect between two continua, or the location where a load or support is applied. Attribute *Associated_continuum* indicates the specified continuum on/in which the interconnect, load, or support is applied. Besides knowing the location of interconnects, loads, and supports and their associated continua, other properties of these instances need to be specified. For example, is an interconnect of type glue bond (i.e. no slip) or slip bond? Which degrees of freedom does the support constrain? For an analysis system, what are the idealized environment effects, such as force, temperature, and pressure? These conditions are specified as objects of type *STRING, Support Feature*, and *Loading Feature* respectively.

The major object relationships and attributes of the ABB information model have been described in Figure 7, above. The resulting ABB information model is equivalent to the preliminary layout proposed in the Pahl and Beitz methodology during the embodiment phase. For a definitive layout, further development is still in ongoing.

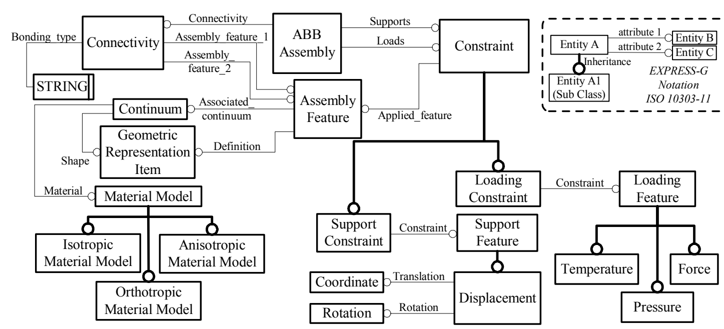

Figure 9. Partial ABB Information Model

After embodying the initial concept of the ABB information model, observations about using the Pahl and Beitz methodology for information modeling are discussed in the next section.

5 OBSERVATIONS AND DISCUSSION

The globalization and distribution of product design increases the interest and complexity of the data management. Until now, the design of application-specific information models has primarily been conducted in an *ad hoc* manner. The previous design practice we had was based either on *ad hoc* expertise and experience of design engineers or on detailed information model development for design analysis integration. However, the current research climate does not allow for the long amount of time needed to develop such information models. There is an urgent need for a rigorous, science-based methodology for information model design that requires less data and model design time than *ad hoc* approaches. After using the Pahl and Beitz methodology to develop the ABB information model, the potential benefits of systematic design for information modeling become quite evident.

It is not our intent to validate or verify the completeness or correctness of the ABB information models. Rather, our focus is on validating and verifying the augmentation to the Pahl and Beitz systematic design method. The development of the ABB model serves as one example to test and validate several of the augmentations to the proposed method. However, it is impossible to validate or verify the recommended augmentations with one example test application. Future work is needed in two areas. The first is in developing additional cases for further validation and verification of the augmented method. The second lies in the technical validation and verification of the ABB information model.

The potential benefits of using the systematic approach to developing an information model are identical to those for in physical product design. The systematic method reduces the dependency of relying on chance and circumstance to find a feasible design solution. *Ad hoc* design resorts to 'aimless' and intuitive solution searching. The ABB information model is developed using a structured, sequential approach in a similar fashion to that of mechanical design.

The development process of the ABB information model confirmed many of the advantages of using systematic design. The use of the Pahl and Beitz methodology results in being beneficial for developing the ABB information model. By using the Pahl and Beitz methodology, future variant design will be easier to perform. As design requirements change due to external causes, the structure of the design may need to change. The development of the other parts of the MRA architecture (APM, CBAM, and SMM) may be spawned from the initial design work from the development of the ABB information model. The step-by-step nature of the systematic approach allows for changes for variant design without revisiting many of the issues in the initial stages of design.

A secondary benefit of using the Pahl and Beitz methodology is that the design process is well documented and traceable from the inception of the design. This helps reduce the potential amount of iterative work for a design. Iteration for a design can start from a particular step in the phases versus reengineering an entire process.

The Pahl and Beitz methodology is effective in facilitating the development of the ABB information model. Based on the results of the development of the ABB model, the systematic approach may be used to develop other information models. However, it would be prudent to develop other information models to further investigate the effectiveness of the systematic approach to information modeling.

6 CLOSURE

Systematic design methodologies are used widely for the purposes of physical product design. We want to investigate the use of systematic design for the purposes of information modeling. Because systematic design is effective for physical product design, the question of whether it could be applied for a software-type application arose. In attempt to answer this question, the Pahl and Beitz methodology is used to develop an information model. The information model chosen for development is the Analysis Building Block (ABB). The ABB is part of a multi-representation architecture (MRA) that bridges the gap of information between design and analysis models. The design activities completed in the development of the ABB model represent a typical process for developing information models. Similarly, the Pahl and Beitz design methodology represents a typical systematic design methodology. Together, the ABB information model and the Pahl and Beitz design process provide a good example for exploring the efficacy of existing systematic design methodologies or developing methodologies for information modeling. This study is performed in the context of ME6101, a graduate level engineering design course.

In the development of the ABB information model, we are able to take the existing structure of Pahl and Beitz and augment it to better suit the needs of information modeling. Traditionally, Pahl and Beitz has been used for physical product design and manufacturing. For the purposes of software design, the methodology needed be to be extended beyond its normal scope. In doing so, we are able to modify and add to the steps of the various phases of Pahl and Beitz. The requirements list headings, function structures, working principles, and conception selection process are geared toward the use of information modeling. This augmented form of the Pahl and Beitz methodology allowed for the development of the ABB information model.

The use of the Pahl and Beitz methodology to develop the ABB information model strengthens the idea that a systematic approach could be used to guide the development of information models. The systematic approach reduced the dependency of relying on chance and intuition to find feasible design solutions. It also resulted in a design process that could be used for variant design and a reduction in iteration design time.

The Pahl and Beitz methodology is effective in developing the ABB information model. However, this study does not conclusively indicate that existing systematic design methodologies should be used to develop all information models. For the purposes of this study, systematic design is effective but this may not always be the case. Further research needs to be conducted with development of other information models using systematic design for a more conclusive result.

ACKNOWLEDGEMENTS

We are particularly grateful for the support of the following people: Yukiharu Takeuchi, Kuniyuki Tanaka, and Shinichi Wakabayashi of Shinko Electric Ltd.; Greg Bettencourt of Shinko Electric America, Inc.; Rod Dreisbach of The Boeing Company; Mike Dickerson of the NASA Jet Propulsion Lab (JPL) and Manas Bajaj and Injoong Kim at the Engineering Information Systems Lab, Georgia Tech. We gratefully acknowledge NSF Grant DMII#0085136.

REFERENCES

[1] Pahl, G. and Beitz, W., (1996), *Engineering Design - A Systematic Approach*, Springer, Berlin, Heidelberg, New York.

[2] Mistree, F., Smith, W. F., and Bras, B. A., (1993), "A Decision-Based Approach to Concurrent Engineering" in *Handbook of Concurrent Engineering*, (H. R. Paresai and W. Sullivan, Eds.), pp. 127-158,New York: Chapman & Hall.

[3] Suh, N.P., (1990), *Principles of Design*, Oxford University Press, Oxford, U.K.

[4] Chamberlain, M. K., Williams, C. B., Cowan, F. S., Mistree, F., 2001, "Orchestrating Learning in a Graduate Engineering Design Course," accepted by the 13th International Conference on Design Theory and Methodology, ASME, September 9-12, Pittsburgh, Pennsylvania.

[5] Peak, R.S., Fulton, R.E., Nishigaki, I., and Okamoto, N., (1998), "Integrating engineering design and analysis using a multi-representation approach," *Engineering with Computers*, 14, 2, pp. 93-114.

[6] Peak, R.S., Scholand, A.J., Tamburini, D.R., and Fulton, R.E., (1999), "Towards the Routinization of Engineering Analysis to Support Product Design, Invited Paper for Special Issue: Advanced Product Data Management Supporting Product Life-Cycle Activities," *Intl. J. Computer Applications in Technology*, Vol. 12, No. 1, pp. 1-15.

[7] Peak, R.S. and Wilson, M.W., (2001), "Enhancing Engineering Design and Analysis Interoperability-Part 2: A High Diversity Example," *First MIT Conference on Computational Fluid & Structural Mechanics (CFSM)*, pp. 704-709, Cambridge MA.

[8] Mistree, F., Lewis, K., and Stonis, L., (1994), "Selection in the Conceptual Design of Aircraft," *5th AIAA/USAF/NASA/ ISSMO Symposium on Multidisciplinary Analysis and Optimization*, pp. 1153-1166, Panama City, FL.

[9] Chen, P., (1976), "The Entity Relationship Model - Toward a Unified View of Data," *ACM Transactions on Database Systems*, 1, 1, pp. 9-36.

[10] Fulton, R.E., Chadha, Bipin, (1999), "Engineering Data Management Course Notes,"

[11] Bruce, T.A., (1995), *Designing Quality Databases with IDEF1X Information Models*, Dorset House.

[12] Halpin, T.A. (1997), "Object-Role Modeling: an Overview," www.orm.net.

[13] Wilson, M. (2000). "Constrained Object Representation for Engineering Analysis," Master's, Mechanical Engineering, Georgia Institute of Technology, Atlanta.

[14] Wilson, M.W. and Peak, R.S., (2001), "Enhancing Engineering Design and Analysis Interoperability-Part 1: Constrained Objects," *First MIT Conference on Computational Fluid & Structural Mechanics (CFSM)*, pp. 750-754, Cambridge MA.

[15] 10303-x, I. and Sys, I.A., "Standard for the Exchange of Product Model Data (STEP),"

