Mobilizing Information Assets to Work for You:

An Information Management Activities Perspective

The *n*-dim Group
Engineering Design Research Center
Carnegie Mellon University
Pittsburgh, PA 15213, USA
Phone: (412) 268 5221
Fax: (412) 268 5229

1. Introduction

The first question that arises from the title of the document is, “what do we mean by information assets.” In our sense of the term, an information asset is the recorded part of the intellectual asset of the firm (this intellectual asset includes human skill assets). To address the mobilization of information assets, firms have used several organizational means. These include overlapped organizational structures to disperse skills by creating cross functional teams. While information and human skill assets can be thought of independently, this can result in fragmented and ad hoc management of information. Information assets are thus lost or become inaccessible. This can be overcome only if the underlying information management infrastructure is matched to support the daily information management activities of the human participants. These activities include exchanging, translating, transforming, organizing and structuring information.

To elaborate, information assets are a knowledge base composed of records, documents, prototypes, experimentation results, etc. that is created by humans in solving the product design problem. Our view of managing information assets is based on holistically integrating information systems to match the needs of project members. The fundamental premise of our approach is that the information substrate should provide the facilities to perform daily tasks. Further, it should also allow for the participatory creation and maintenance of information assets for use in dissemination, training and innovation.

The purpose of this document is to describe our systematic approach to study information activities as an integral part of the designing, building, testing and transitioning of support technologies. In Section 2 we present a summary of empirical studies in design to understand the nature of information activities. Section 3 presents a classification of information activities derived from these studies. In Section 4 we describe a context of product design in a firm. Section 5 briefly raises the issue of complementing computer support to human needs.

Section 6 is an evaluation of existing technologies and *n*-dim using the classication of information activities. In Section 7 we evaluate features of *n*-dim in an expanded design context and with respect to the needs of information activities. The final section provides a summary and conclusion.

2. Empirical studies of design and information management activities

To understand the nature of information in engineering we have conducted several empirical studies of design. The purpose of these studies were varied and focused on different levels of detail. Our purpose in these studies was to understand product design and manufacturing as an information management problem. More specifically, the objective was to understand the nature of information activities and to develop methods and methodologies for supporting system design. Table 1. provides brief summaries of some of our studies.

Each of these studies highlight the importance of different information activities in different contexts. Even though information activities form the core of performing product design, different tasks often are dependent in different ways on those information activities. For example, terminology reconciliation and acceptance is a pre-cursor to ease of distribution and access in the material information system case. Similarly, the ability to structure the common information for access and retrieval is critical to co-operation in the control systems design case and so on. Within an application context the importance of different activities determines the type of system that will meet the user needs. Further, other factors such as time sensitivity also play an important role in determining the structure of activities. For example, information on co-ordination are sensitive to time delays while information from previous designs will be relatively less sensitive to time delays in their accessibility. Without elaborating on these issues, in the next subsection we provide a classification of information management activities that can serve as a framework for evaluating the needs and methods for the design of information systems.

Table 1: Studies of engineering design conducted by us

Design Project	*Methods Employed*	*Focus*	*Bottleneck information activities*
Process Control system design (Westinghouse)	Direct observation of design meetings; collection of all design documents; recording meetings.	Preliminary design.	Inability to access and integrate tools; Searching and organizing information.
Integration of Material Databases (ALCOA)	Tracking information flows with a survey. Creating concept structures using semi-structured interviews.	Information sharing across divisions to reduce duplicated work.	Sharing information - multiple terminologies. Fragmentation resulting poor access; legacy integration.
CINERG: Multi-University Collaborative Distributed Design	Direct participation and observation. Analysis of documents and messages exchanged. Post hoc review.	Feasibility of electronic collaboration in asynchronous, distributed design with periodic face-to face meetings and conference calls.	Identification of activities and tools. The need for managing information from multiple media. Integration of co-ordination with design object.
Design of and manufacture of electric power devices (multiple studies)	Questionnaire and direct interviews with participants in all phases of the design manufacture and services. Analysis of critical documents.	Information need and flows in the design and manufacturing process (intra-project and inter-project flows).	Identification of points of repository creation. Bottleneck in flows on needed information
Undergraduate project courses in software engineering	Analysis of design information including intermediate and final products and electronic communications among designers.	The effect of communication on outcome.	Identification of the effect of terminological consistency in group design and project outcome.

3. Information Management Activities (IMA): A Classification

The classification of information activities presented in this section is one of many ways these activities can be organized The purpose of this classification is to identify the information activities at different level of needs, purposes, support functions and measures to evaluate the effect of providing support function to achieve a purpose. There are three primary classes of information activities. They are: Information Manipulation, Knowledge Building and Collaboration. The classification presented serves to focus on particular compositions of activities even if they overlap in their membership in different higher level categories.

In the following paragraphs, we present the activities in terms of purposes they address; the support functions required for achieving the purpose and an initial set of measures for effectiveness.

Information manipulation:

This class of activities includes retrieving, translating and transforming data and analyzing and evaluating data with analytical tools. These activities are critical to any product design team member, as was noted in Table 1, one of the major obstacles is the time consuming nature of the translation and transformation tasks. Further, non-standard information needs and formats also leads to continuous re-adjustment on the part of the user. This makes the task even more tedious.

1. *Purpose:* To retrieve, classify and evaluate information

   Support Functions:

   - Structuring information
   - Finding information
   - Use of Standards
   - Analysis (numerical and other)

   Measures: *Time measures to access and volume of information handled in retrieval of data. Relevancy measures for retrieval of data. Availability and access to analysis and other numerical tools.*

2. *Purpose:* To transform and translate information

   Support Functions:

   - Information Visualization
   - Sharing Methods and Tools
   - Use of Standards.
   - Integrating legacy and external methods and tools
   - Evolution of information structures.

   Measures: *Time taken to transform; for example, to move from one form (excel) to another (word). More complex situations include data transfer and translation across multiple representational structures (computer or otherwise)- drawing to calculation charts, etc.*

3. *Purpose:* To store, access and protect information

   Support Functions:

   - Sharing information
   - Protecting information

   Measures: *Process time for storage and retrieval with access controls. Flexibility and enforceability of access controls.*

Knowledge building

Knowledge building is a specific and deliberate activity that is based on the need to take data from the experiences of product design and consolidate and synthesize that data into coherent and well structured pieces of knowledge. It is this synthesis and distribution of these structured pieces of knowledge that is critical to leveraging this experience in the future. The importance of knowledge building is often underestimated as it is less visible and lacks immediate payoffs.

1. *Purpose:* To capture and re-use the design process and design rationale.

   Support Functions:

   - Capturing history
   - Capturing rationale
   - Structuring information
   - Knowledge visualization

   Measures: *The measure of re-use and extent of completeness of documentation for re-use.*

2. *Purpose:* To capture, consolidate and re-use knowledge from multiple perspectives generated from the previous purpose.

   Support Functions:

   - Learning by induction
   - Enabling end user customizing
   - Sharing information

   Measures: *Time to research and find relevant knowledge in the corporate memory. Failures to re-use knowledge from prior efforts.*

Collaboration

Collaboration is critical to any group activity. Collaboration takes place best when the symbolic representations that are used in the co-ordination are sharable, visualizable and accessible at the time of and location of need. For example, these symbolic representations are used for negotiating and maintaining terminological compatibility and for identifying the current state of the designed object and the process used. In our definition of collaboration, it is the sharing of context and process information so the participants in the product design team can react to changes in the information state. Their reaction results in coordinating and performing actions to change the state of information so that it reflects the new desired state of the process and design.

1. *Purpose*: Support negotiation, Co-ordination

   Support Functions:

   - Sharing information
   - Change management
   - Workflow & Process tracking
   - Knowledge visualization

   Measures: *Extent of terminological consistency, Scope of task co-ordination. Efficiency in exchange of information co-relation between communication patterns (measures) and outcomes(measures).*

The information activities are currently supported through a variety of systems, methods and procedures. However, based on our observations of design practice, much of the difficulty in designing lies in dealing with the issues of acquiring, manipulating, transforming, and using information and storing it in a manner suitable for exchange and subsequent reuse -- thereby developing and sharing knowledge. The situation is further compounded by the fact that design takes place in a rich and quite diverse set of contexts requiring engineers to deal with a great deal of multiplicity and change. The combination of these two result in a situation that can be described only along several dimensions of the engineering context; the context exhibits great deal of complexity and variety. And as Ashby (1958) points out, a "control *(support)* system" for such a situation, if it is to be adequate to the task, must exhibit at least as much complexity and variety.

In the next section, we use an exemplar product development organization to illustrate the interactions between the different organizational departments, projects and products. This example serves as a context in which these information management activities are undertaken.

4. Information Management Activities: A Context

These information based activities are directed towards achieving certain goals within a project and across projects in firm. These goals include the management of creation and dissemination of information and knowledge about a project, across projects and product lines depending on the structure of the market segment of the product. Essentially, the problem of information management boils down to methods for evolving orderly repositories of product and process descriptions, application of methods and tools, their inter--relationships in a given context . Figure 1. illustrates this idea for a company with many product lines that depend on common services and functions. P1...P4 represent projects, they may belong to different product lines, the rows represent the service and functional departments commonly utilized by the projects (e.g. in consumer electronics, motor is one such function). For a different organization such as Power plant design we might have a different structure and interactions, while essentially retaining the distinction of intra-projects (to denote immediate) and across project (to denote longitudinal) in the management of information.

The goal of the information support infrastructure is to act as a facility to build an inter-linked set of repositories of information. The structure of the repositories consist of a static and a dynamic part. The dynamic part is critical to accommodate the changing needs of information structuring and sharing. In the case where information structures are static or mostly static (such as in some types of applications like parts databases, product data management and document management), commercial off the shelf components such as relational databases, PDM and document management's systems can be used. The dynamic part is often dealt with as separate and as a secondary issue when designing support systems. In our approach we have made a strong hypothesis on the nature of critical knowledge in product development. We contend that the identification and recognition of interrelationships or lack there of, between the relatively static and evolving information structures, is where the essence of good and bad practice is embedded. Given our hypothesis, without supporting and capturing the dynamic part of information structuring we would be losing critical experiential information. Any infrastructure that supports engineering work has to accommodate both forms of interaction with information. The infrastructure should serve as an integrator and provide the ability to evolve these systems to satisfy continual needs (the dynamic part) that arise in the exercise of information based activities in the process of doing engineering work. It is this type of infrastructure that our support system (*n*-dim) aims to create. The primary objective of the project is the ability to capture and disseminate good and bad practice as an integral part of supporting information management activities of an engineering organization in an evolutionary and participatory manner.

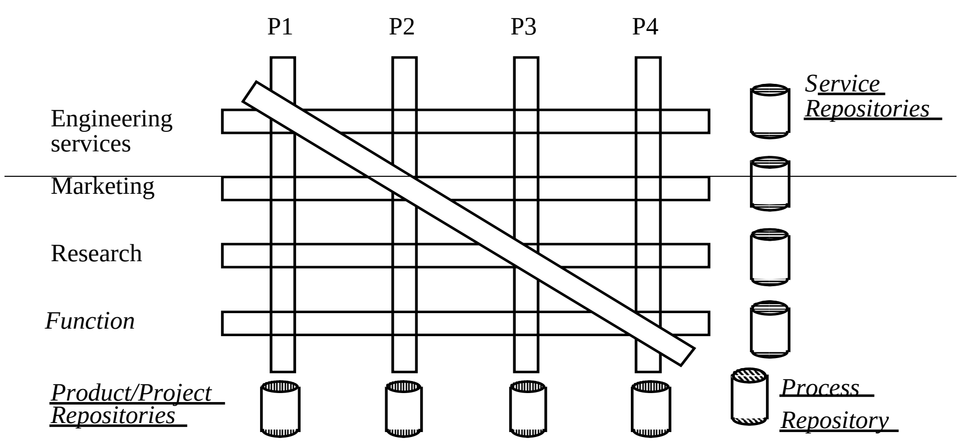

Figure 1: Information management context for product design

5. Information Management Activities: Complementing computer support to human needs

In order to deal with the complexity of engineering design information management, organizations have developed, adapted, and adopted a very wide variety of specific methods and tools so as to have the requisite variety necessary for effectively supporting design. By and large these are "point" tools; i.e., tools which solve well defined and circumscribed problems. Unfortunately, of course, such an agglomeration of point tools further compounds the complexity faced by the engineer since each such point tool requires its own sub-language and other idiosyncrasies. This suggests that we develop an integrated support environment. However, a sufficiently rich integrated environment, unless carefully designed, could end up being as complicated (if not more so) to the engineer as the original problem. In order to deal with this dilemma we build a support system on a foundation of a few well designed features which, when appropriately composed (in light of the existing information management problem in its context) can generate the desired variety in behavior. The strategy, then, is to carefully select features that are both simple to grasp (for the design engineer -- the user and the system designers -- the developers) and yet can be easily put together to exhibit a very wide range of behavior. From a different perspective, and generally because of the attendant complexity, it is almost impossible for any of us as support system builders to know enough of a given specific design context to get the larger integrated system right -- or even approximately right -- the first time.

In order to incrementally and iteratively arrive at a core set of features, we have developed a methodological approach. This approach has steps for understanding the overall information flow problems and activities and for building, testing and transitioning specific implementation for a variety of tasks. Tables 5a and 5b in Appendix B summarize the objectives, methods, tools and outcomes of the approach.

6. Information Management Activities: A brief comparison of existing technologies and *n*-dim

Having identified the information activities and the current approaches to solving the problem we will now evaluate the systems that are currently available on the market. To provide a contrast we have also included the *n*-dim system. The process of creating the table was based on the extent to which a particular tool supports an information management function. We use **F** (full coverage) to indicate that the system allows for complete and transparent coverage of that activity. **P** (partial coverage) is used when the system supports that activity but provides either only pre-structured, inflexible support or provides support for only some of the functions of the activity.

Table 2: Comparison of IMA support systems and *n*-dim

Technologies / *Activities*	*Notes*	*WWW*	*PDM*	*DBMS*	*Video*	*Shared Work Spaces*	*Caelum*	*n-dim*
Sharing information	P	P	F	F	P	P	P	F
Sharing tools	N	P	F	N	N	F	P	F
Capturing history	P	N	P	P	N	N	N	F
Capturing rationale	P	P	N	N	F	F	N	P
Negotiating	P	P	N	N	P	P	P	P
Coordinating	F	N	P	P	P	P	N	F
Integrating Tools	P	P	F	P	N	N	P	F
Structuring information	P	P	P	P	N	P	N	F
Finding information	P	P	F	P	N	P	P	F
Using standards	N	N	F	F	N	N	P	P
Information Visualization	P	P	N	N	P	P	P	P
Evolving system	P	P	N	N	N	P	N	F
Learning	N	N	N	N	N	N	N	F
Customizing	P	P	N	P	N	N	P	P

Legend: F - Full coverage; P - Partial coverage; N - No coverage

To illustrate, how the comparison was made, we will take four activities: sharing information, structuring information, finding information and using standards.

a) Sharing Information: Lotus Notes is designated partial coverage, “P”, because it allows for only pre-structured modes of exchange primarily through document templates and e-mail. Newer versions allow creation of interfaces for retrieving data from part and financial DBMS. PDM systems are designated full coverage as they carry several information structures in the same context (e.g. document and CAD data can co-exist within a PDM/Data base system). *n*-dim is closer to PDM systems in the sense of sharing information.

b) Stucturing Information: *n*-dim is distinguished from the others by its full support, “F”, for the information structuring activity. PDM and Lotus Notes provide support at the developers level and hence are designated as “P”. The combination of structuring and sharing information is embodied in *n*-dim to support the evolutionary aspects of information structures.

c) Finding information: Most systems offer one or two forms of searching and they often operate independently. In *n*-dim the features are such that multiple indexing schemes work over the same information space and can be used in combination, thereby making the search more efficient. In other systems, incorporating additional search mechanism is far more difficult. Hence we have designated “F” for *n*-dim and “P” for most of the other tools.

d) Using standards : Tools such as PDMs are based on engineering data standards and hence are designated “F.” They are directly built on standards and so are inflexible in evolving standards. Lotus notes and WWW use software standards to make use of tools and to achieve data transfer. We allocated “P” for them since they do deal with engineering standards. *n*-dim has been allocated “P” as well because *n*-dim uses or intends to incorporate software standards. On the other hand *n*-dim allows for the modeling of engineering standards and hence is able to communicate with PDMS systems. Again full support for structuring information in the form of generic modeling provides the ability for *n*-dim to play the role of the integrator of disparate information structures.

Each of these rows can be similarly elaborated. The purpose of the examples is to provide a guide in using the table and to illustrate the interlinkages between the activities.

Summary of Comparison with other systems

Product Data Management and Manufacturing Planning Tools . There are two sets of computational systems that are currently being used in product design and manufacture. There are systems like Baan, Sherpa and SAPOnes that originate from the information and process management requirements of manufacturing. Then there are PDM systems such as Metaphase and STEP Tools that are available through a number of vendors. The manufacturing systems such SAP are also starting to include PDM systems as integral to their packages. These integrator systems have also been termed “middle ware” in the industrial jargon.

These middle ware systems are well structured and are complex and are customized to each individual customer. However, evolving these systems to allow for changes in the modeling of processes is done outside the loop of use. This makes these tools difficult to evolve in place. However, these systems are suited for well structured tasks that can be mapped directly from detailed design to manufacturing operations. These tools are not suited for design of a product where the need for information is fluid and is not always possible to structure *a priori* all the models that are required in doing design.

For example, even in the systems mentioned above, the configuration management of the product information is predetermined. However, studies in software design has shown the need for multiple models of configuration management, each of which is useful in different phases of the project. This leads to the problem of a mismatch of tools to tasks. Our objective in the *n*-dim project is not to replace these systems but to integrate them by modeling their representation in the *n*-dim environment, thus providing the user an interlink between this information and any other relevant information that is modeled and maintained in *n*-dim.

Asynchronous Co-operative work Tools: Lotus notes, Inter. Lotus notes is the most advanced of these asynchronous co-operative work tools. It provides the facilities to create applications whereby a group of people can exchange information directed towards particular projects. Several companies use Lotus notes as the mechanism for e-mail, to maintain project information and for gathering information from maintenance, service and other departments.

The newer version of Notes allows users to create small applications in their workspace through end-user programming. Lotus Notes is primarily a document database that is replicated across the network. It serves as an ideal application for global e-mail and reporting. Notes offers limited search facilities over the documents through a keyword search.

Other Collaborative systems . There are collaborative systems in the market such as Toyota's Celeum and Xerox's shared White Board and video conferencing systems. Toyota's Celeum brings together access to files, e-mail and video conferencing. Xerox's shared white board allows for sharing of the same workspace on PC based systems allowing joint evaluation of documents, drawings and other materials. Video-conferencing systems are one end of the spectrum where there is only audio-visual contact with each other. These systems support several aspects of collaboration but are limited in the history and process capturing facility as their primary motive is to facilitate immediate communication and is directed towards the accumulation of knowledge through capture and organization. In effect they do not deal with the longitudinal aspect of collaboration that is "management of knowledge."

7. Addressing of the requisite variety of information activities by *n*-dim

The *n*-dim infrastructure is based on a small set of features in addition to the hyper-graph-based representation of information objects (see Appendix A). In this section, we evaluate the features of the *n*-dim infrastructure against: a) an elaborated context of design and b) the set of information activities defined in Section 2. The tables presented in this section illustrate a design problem where the impact of several interacting factors are unknown in specifying the correct design. In this case the design problem is that of designing an information management system.

Dimensions of design context and n-dim

To explore the first set of relationships, we extend the design context of Section 3. In this extension we impose the dimensions of time, place, culture, languages, tool expertise, perspectives, interactions and usability (work enabling) over the information flow structures. Table 4 provides a summary of our current understanding on the influences and impacts between the dimensions of design context and the features and applications of *n*-dim.

For an illustration of how to interpret Table 3, let us take the Flat space feature which affects the need to accommodate multiple perspectives and languages. To create multiple perspectives, one requires the facility to allow for multiple ways of indexing and labeling the same object in different collections. To allow multiple formal languages to interact over the same set of information objects we should be able to access them from any of the languages. Both needs have the same requirement - an accessible flat space of objects. They are different in that they conform to differing degrees of formality in their description of the collection of information objects. Similarly, each column and row can be explained based on constraints imposed by the context of the information management activities and the collection of features that address these constraints.

Table 3

Features and Applications of n-dim / *Dimensions of design context*	*General Graph*	*Flat Space*	*Modeling Languages*	*Publication*	*Tool Encapsula tion*	*Workspaces*	*Scripting Language*	*Search*	*Repositories*	*Issue-Based Discussion*	*NLP Tools*
Time				+				+		+	
Place				+				+		+	
Culture	+		+			+					+
Languages	+	+	+			+	+				+
Tools			+		+		+		+		
Expertise	+		+		+					+	+
Perspectives	+	+	+			+				+	+
Interaction			+		+		+		+	+	
Usability	+		+		+		+		+		

I should be noted that the features of flat space, modeling languages and general graph modeling have to be a single bundle, to address language and perspective requirements, as each individually cannot solve the problem. We add or subtract features from this table based on the needs of the information activity supported.

n-dim features and information management activities

Table 4 summarizes the relationships between IMA and their support with respect to *n*-dim features and applications. The purpose of the table is to provide a check list to evaluate the scope of the impact of the features and applications on individual information management activities.

To illustrate this process consider, for example, that it was possible to generate an hypothesis regarding the efficacy of NLP (Natural Language Processing) tools in assisting negotiation and anticipating other communication problems. Several software development projects were studied wherein project communications and documents were collected and subject to NLP and subsequently to a statistical analysis using the outputs of the NLP to generate the data[1]. It was found, for example, that delayed negotiation of terms between design teams was indicative of future problems at the integration phase. More generally, we found that communication metrics can be used as indicators of problem areas and potential downstream risks to the design project. Based on this study, we are currently deriving a basic set of analysis and diagnostic tools that can become part of the support environment and, if desired,

---

1. This is also an example of the use of social science approaches shown in Table 3.

used by designers to forewarn them. It is from this experience that the “+” sign in Table 5, NLP-negotiation cell, was obtained.

Thus, as our studies of design progresses, these tables will continue to evolve: some rows or columns may be broken down, some consolidated, some deleted, and new ones added, and some cells changed (from a “+” to a blank or vice versa as our understanding and knowledge grows). Additionally, this table can be used to indicate areas for research and development (as suggested by blank cells or even entire rows). We schedule these studies based on their impact on practical projects.

Table 4

Information Manipulation	*Knowledge Building*	*Collaboration*	*Features and Applications of n-dim* / *Information Management Activities*	*General Graph*	*Flat Space*	*Modeling Languages*	*Publication*	*Tool Encapsulation*	*Workspaces*	*Scripting Language*	*Search*	*Repositories*	*Issue Based Discussion*	*NLP Tools*
+			Creating Information	+	+	+			+			+		+
+	+		Structuring Information	+	+	+			+			+		+
+			Finding Information								+	+		+
	+	+	Sharing information				+				+	+	+	
+	+	+	Information Visualization			+		+			+			
+		+	Using Standards	+				+		+				
+		+	Sharing Tools					+		+		+		
+			Integrating Legacy Tools					+		+				
	+	+	Capturing History			+	+					+		
	+	+	Capturing Rationale			+	+					+		
	+		Learning by Induction	+	+	+		+			+	+		

Concluding Remarks and Summary

We began this document by asking what we meant by “information assets” as a part of the intellectual asset of a firm. We contended that to manage the information asset we need to understand the context in which this information asset is created, disseminated and managed. we identified a classification of information activities from our studies of engineering work. In light of our concentration on information assets, we make a case for a systematic and holistic approach to the information system problem due its attendant complexity. With this in mind we have provided a classification of information activities, a comparison of existing systems and our own n-dim system against these activities.

The purpose of this document is to describe our systematic approach to study information activities as an integral part of the designing, building, testing and transitioning of support technologies. We have rooted our investigations in the hypothesis that critical knowledge resides in the inter-relationships of well-structured information and as such it is often not captured. To test such a hypothesis, we build systems that allow us to capture precisely that knowledge and demonstrate its subsequent utility. We have emphasized the importance of the information asset and its consolidation. Further, we contend that consolidation of the knowledge will be fragmented without extensive methods for and a substrate for information capture. If supported systematically, the information asset becomes a critical resource in the dissemination of experience, in training and in engendering new innovations.

In conclusion, in order to emphasize and provide an understanding of the n-dim approach, we have presented evaluations in the form of tables. These tables provide us a check list to evaluate the completeness of the minimal set of features to support all information activities. Two distinguishing features of our approach are its rootedness in an empirical approach to developing design systems that match human needs and our focus on a minimal set of features as the basis for accommodating complex information tasks.

APPENDIX A:

n-dim - An infrastructure for information modeling and applications in engineering Design

The basic premise of the *n*-dim system is that every member in the product design team operates in an information space that is characterized by the domain of experience and skill of the participant. The information space of the product is characterized by the union of the information spaces of the individual participants. This union of information, the product information space, is not a straight forward union as there are terminological inconsistencies across the information spaces and well understood and not so well understood relations between the elements of the information space. Further, in each information space of the participants and in the product information space, the organization of information itself evolves as processes and product understanding increases. The objective is to support the individual evolution of knowledge and collective evolution of knowledge in the form of information structures that are constructed by the participants in the course of product development process. The history of both process and product is critical to ensuring that evolution takes place in an effective manner. This is important to both the short run evolution of a project and to a long run evolution of policies of operation. To address this, we have taken as our hypothesis that a generalized graph modeling environment that operates over the elements (other information structures - graphs and atomic information elements) in the information spaces is necessary to capture the structure and evolution of information and knowledge, both formal and informal and individual and group.

Concepts in *n*-dim

Information Objects: Information objects are of two types: atomic objects and structured objects. Atomic objects are strings, numbers, images, audio fragments, etc. They are not decomposable. Structured objects are graphs whose nodes are atomic objects or other structured objects. The graph includes named links that can exist between any two nodes.

Models: For convenience we use the term “model” to denote both atomic and structured objects. Objects are referenced in a model rather than being embedded in a model. Models imply object association by having their pointers collected together and named links are used to describe the relationships between the object pairs.

Flat space: Flat space is a term we have given to the conceptualization of an information space where any model is directly referable. This allows for the creation of a user defined set of relationships across information objects of any granularity. Users have the ability to create any arbitrary model over a subset of the entire collection of information objects in the information space.

Modeling language

A model can be abstracted to create a set of building blocks that correspond to the type of information objects in the graph and the types of named links in the graph. These abstractions can be made to create a vocabulary that to create other model instances. For example, one can create an object and abstract the features of that object in creating another object of different dimensions, scale, etc. Here one has developed a language for describing that particular artifact. Languages restrict the type of objects and named links users may use to construct further instances of the model type. Modeling languages are models therefore you can use any model to define the grammar for other models.

As models and modeling languages are graphs that can be modified and extended and operated upon, operations can be defined on the graphs represented by the model. These operations are inherited by the model instances created by using the model as the modeling language. Thus the system allows for standardization of modeling languages and their use and for the evolution of new graph types from the model instances. A deductive and inductive approach is supported in the modeling process.

Evolution: Published, Public and Private

History is critical to effective evolution and ordered evolution is essential to building history. We have developed an ordered evolution of the system with the following three facilities. These facilities deal with different levels of granularity: published, public and private.

Published: The published mode of operation is an archival facility, i.e., a library. Any information object that is entered into the library cannot be withdrawn (it is persistent). Changes are published by copying, modifying and then re-publishing a model. The system automatically records the act of copying and re-publishing, thereby keeping a branched (time and owner) history of the model. The model that allows for the tracing of the origin of the document is itself a graph within the system.

Public: The public mode of operation is a public forum area. Here the primary objective is to provide the ability to all participants to share and add to the model synchronously and synchronously. As with any forum the language of the forum is restricted to the purpose and domain of discourse as determined by the participants or the existing body of knowledge. History can be recovered by viewing a model's state in time.

Private: Private, as the name denotes, is the private information space of the individual. There are no restrictions on how a private space is managed. The users can add, delete, and restructure their information objects as they please.

The above characterization of the system is necessarily abstract, as the details of the system cannot be described in this short note.

Strength and weaknesses of our system

The primary strength of the system is its approach to dealing with software development and knowledge development in an evolutionary manner. The system combines evolution, history and modeling within the same framework - the framework of graph based modeling. The other main strength of the system is the flexibility of integration of legacy tools, they can be invoked from within the system in their native form or can be integrated fully into the system. Further, the system also allows for the creation of new tools by the user as needed. We are also expanding our research efforts in creating graphically based end-user programming capability to make the above tasks easier.

Another strength of our system is at the level of the infrastructure. The flexibility of the object tool kit (**Table 1**) allows for extensions to the system incrementally without damaging the underlying system. This problem is acute in many commercial systems, where moving from one version to another version often requires a transition time which may last from hours to weeks.

The *n*-dim system itself is an infrastructure that is customized to particular applications. It is not a system that can just be bought and installed. This can be viewed as a weakness from a commercial point of view and as we are keeping that much in mind as we plan for commercialization. But a flexible infrastructure with the strong capabilities of *n*-dim is potentially a great strength for any organization that chooses to make the investment.

APPENDIX B

Table 5a Process steps, Methods and Outcomes

	1	*2*	*3*	*4*	*5*
Methods *Process Steps*	*Information flow-study*	*User participation*	*Prototyping*	*Testing by users (uncontrolled study) Industry/Classroom*	*Code maintenance and "hardening"*
Understanding of the current state of information management	An information map of the Business division studied	Identification of a specific target area for support			
Development of support systems		Use and system specification document	A series of working prototypes	Areas of improvement of use and performance before testing	Improving scope. quality, performance, and usability
Assessment of support system effectiveness	An information map after system installation	Continuous feedback.			
Evolution of system		Continuous identification of new needs	Continuous evolution	Identification of needs (research and improvement) to reduce effort and time	

.

Table 5b: Methods, Tools and Actions

	1	*2*	*3*	*4*
Tools *Methods*	*Questionnaire and interviews*	*Infrastructure for evolving information systems*	*Layered modular architecture*	*Social Science methods (regression/multiple regression/ natural language analysis)*
Information flow-study	Identifying communication gaps			
User participation				Source of action research methodology
Prototyping		Support for quick prototyping, customization, legacy tool integration and evolving the infrastructure	Potential re-use of existing legacy layers (e.g., DB)	
Testing by users (not controlled study) Industry/Classroom		High usability to support early testing		Identification of needs (research and improvement.) to reduce effort and time
Code maintenance and “hardening”		Support for improving performance of validated code	Support for improving layers with new technologies	
Basic research (e.g., study the role of Communication in design projects)				Identification of needs (research and improvement.) to reduce effort and time