Journal of Computing and Information Science in Engineering

Guest Editorial

Special Issue on "Engineering Informatics"

Computer-aided design (CAD), intelligent CAD, engineering analysis, collaborative design support, computer-aided engineering, and product lifecycle management are some of the terms that have emerged over the past 50 years of computing in engineering. Codification and automation of engineering knowledge and methods have had major impact on engineering practice. The use of computers by engineers has consistently tracked advancements in computer and information sciences. Computing, algorithms, computational methods, and engineering have increasingly intertwined themselves as developments in theory and practice in both disciplines influence each other. It is now time to begin using the term "engineering informatics" to cover the science of the information that flows through these processes.

Informatics, with origins in the German word *Informatik* referring to automated information processing, has evolved to its current broad definition. The rise of the term informatics can be attributed to the breadth of disciplines that are now accepted and envisioned as contributing to the field of computing and information sciences. A common definition of informatics adopted by many departments/schools of informatics comes from the University of Edinburgh: "the study of the structure, behaviour, and interactions of natural and artificial computational systems that store, process and communicate information." Informatics includes the science of information, the practice of information processing, and the engineering of information systems.

The history of engineering and computers shows a trend of increasing sophistication in the type of engineering problems being solved. Early CAD was primarily geometry driven (using mathematics and computer science). Then came the engineering use of AI, driven by theories of cognitive science and computational models of cognition (logic and pattern based). More recently, models of collaboration and representation and acquisition of collective knowledge have been introduced, driven by fields of social sciences (ethnography, sociology of work) and philosophy. Following this history, the definition of engineering informatics can be taken as "the study of use of information and the design of information structures that facilitate the practice of engineering and of designed artifacts that embody and embed information technology and science to achieve social, economic and environmental goals." Given this perspective, the rest of the introduction identifies different strands of concepts that inform and support the evolution of engineering informatics as a distinct discipline that lives at the interface between engineering and informatics, in the same vein as bioinformatics, medical informatics, and other applied disciplines.

Information is context specific and its engineering is an integral part of any exchange among people and machines. Thus, informatics is the process of (1) creating and codifying the linguistic worlds (representational structures) represented by the object worlds in the relevant domain, and (2) managing the attendant meanings through their contexts of use and accumulation through synthesis and classification. Engineering informatics is a reflective task beyond the software/hardware that supports engineering; it is a cross-disciplinary perspective on the nature of collective intellectual work. It thereby becomes critical that a consciousness of the use of languages[1] and their implications in the storage and retrieval of information in a work community be addressed as part of any information engineering task.

Figure 1 is a comprehensive view of the field of engineering informatics. In this figure, the inner set of circles marked as Informatics covers the fundamental activities associated with informatics in general. These include representation (information modeling), modeling formalisms (mathematical, logical, and rule based), and other aspects including data mining, information organization, and classification.

The next circle, denoted by Product and Process, identifies the multilevel, multiscale modeling activities in terms of components to products in the product realm and atomic to collective and distributed activities in the process realm. The outer circles show the inputs to engineering informatics from a number of disciplines that provide the domain knowledge and methods and tools for modeling and evaluation.

The NAE report, "Making It Better," has identified information technology and sciences to have both created the need for, and play a role in, facilitating the management of complex sociotechnical processes.[2] The opportunities and challenges ahead are the ability to create design methodologies for the creation and management of these complex networks.

Networks have become the mainstay of the role informatics plays in interlinking diverse organizations and institutions. The evolution of technology passes through the stages of creating the raw material (iron and steel) that result in structures (bridges and canals), that in turn lead to machines (steam engines), and that allow the creation of networks (railways). However, this transition is not always smooth due to the independent prior evolution of the technology stages before the networks arise. Networks impose conditions on the further evolution of the technologies developed in the previous stages by requiring them to become compatible. These conditions give rise to standards, without which the compatibility of the different structures and machines to work together to form the network becomes impossible. Network effects and positive externalities derivable from the networks critically depend on the creation and evolution of socially acceptable standards. However, business and corporate strategies and the resistance to the creation of shared networks often pose impediments to the creation of these standards.[3] We can see the same story unfolding again in the information network world as it happened

---

[1]We mean languages to mean not just natural language but also all formal languages including mathematics and computational languages and formalisms.

[2]Making IT Better: Expanding Information Technology Research to Meet Society's Needs (2000) Computer Science and Telecommunications Board (CSTB), National Academies Press.

[3]Product lifecycle management support: a challenge in supporting product design and manufacturing in a networked economy, Eswaran Subrahmanian, Sudarsan Rachuri, Steven J. Fenves, Sebti Foufou, and Ram D. Sriram, International Journal of Product Lifecycle Management 2005, Vol. 1, No. 1, pp. 4–25.

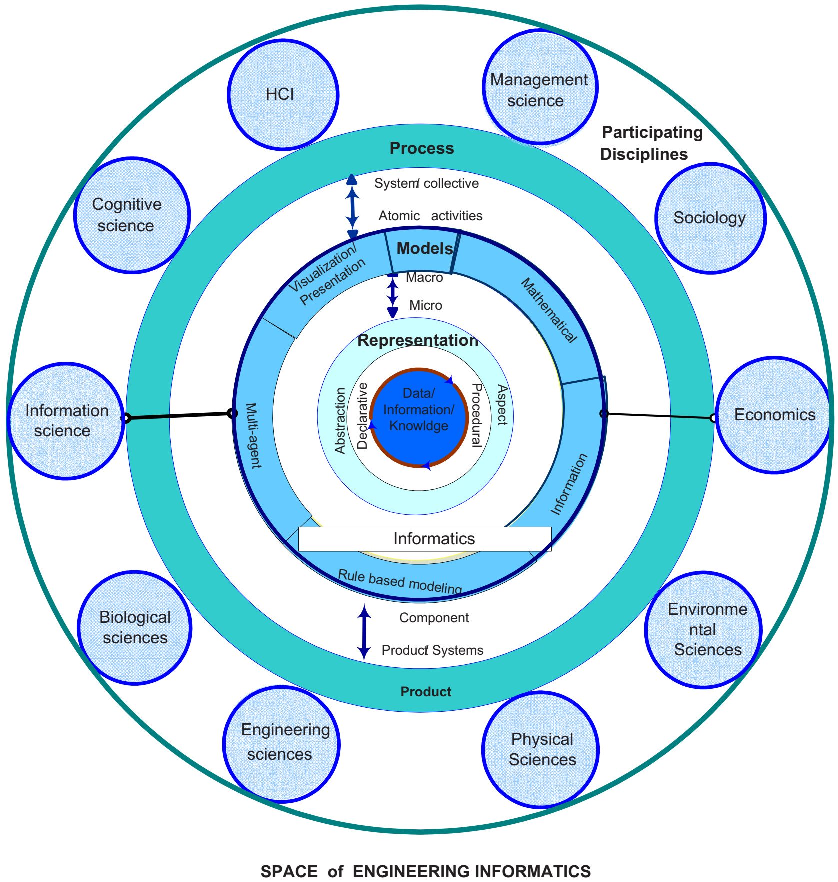

Fig. 1 A framework to view engineering informatics

in canal networks, rail networks, and airline networks in terms of the needs for standards. This is especially so in the complex product organization and development world.[4]

The role informatics plays in engineering products and services has become significant in the past 10 to 15 years. Most of the development has happened in an ad hoc manner, as can be expected. Techniques appeared in computer science and in programming practice; these techniques get used in engineering as is. Early computing in engineering was limited due to the capacities of computers. Computational power and telecommunications systems have started to converge, resulting in the possibilities of untethered connections and exchange of information that was just a distant dream in the early computing days. These developments have made the problems of distance less onerous and allow for global design, manufacturing, and supply chains. However, the problem of managing a global supply chain still is a daunting task with numerous incompatibilities in information exchange and coordination. The recent delays in the Airbus 380 and Boeing 787 are examples of problems of this nature.

The problem of integrating entire sets of industries in a flexible and ad hoc manner is still a dream especially for small-scale industries within the larger global environment. For this dream to become a reality, standards become critical. With technology evolving continuously, the task of creating information standards for varieties of exchanges from the syntactic to the semantic is a challenge yet to be resolved.

[4]Information sharing and exchange in the context of product lifecycle management: Role of standards, Sudarsan Rachuri, Eswaran Subrahmanian, Abdelaziz Bouras, Steven J. Fenves, Sebti Foufou, and Ram D. Sriram, Computer Aided Design, in press, available online, July 1, 2007.

The complexity of engineered systems arises along the dimensions of the (1) product and/or the process axis and (2) the organization required to create, maintain, retrofit, and dispose of the engineered system. Product complexity is commonly understood as arising from scale, variety of components, technology, distributed design and production, maintenance, and/or disposal. Process complexity is the complexity of coordination, communication, and information and knowledge management across disciplines, functions, time, space, and cultures. This complex global environment is inherent to a modern day engineering organization, large or small. The management of complexity requires the use of simulation, modeling, information management, management of language, and creation of adaptive processes. Without an understanding of the role of the tools of informatics, one cannot deploy a complex engineered system for a given purpose.

There are two types of engineering information: (1) the information from creation to disposal of the products and (2) the information the product itself uses and processes to achieve its intended behavior. The first type is embedded in the organization and society and the second embedded in the product. The information embedded in the product is embedded in multiple technologies (electrical, chemical, computer hardware and software, mechanical, biological) that interact to create a coherent behavior. In the case of the organizational and social embedding, the behavioral goals become information formalization and accessibility, support of cooperative work, creation and maintenance of archives, training, decision support, coordination practices, knowledge awareness, and a symbiotic human-machine system that is adaptable and flexible in its process. The two types of information interact with each other and provide a rich potential for cross fertilization of methods and ideas.

Computer scientists or engineers by themselves cannot solve engineering informatics problems or the processes required to manage information in the context of engineered systems—it has to be a collaborative effort. The lack of skills among computer scientists in engineering and engineers in computing has led to problems bridging the disciplines. What pedagogical stance can help prepare students to deal with the complexities that are inherent in the task of engineering informatics? The culture of learning has to encourage the appreciation of diversity at the same time looking for the core essence and canonical nature of the experiences. While the products of today are increasingly designed for variety, we still have not mastered this process conceptually, let alone are we preparing our students. The fundamental characteristic of engineering informatics is that it is applicable at local levels of decision making in a design process as well as at the holistic level of product management and organizational design. We believe that knowledge is generated at the interfaces of several disciplines that participate in the design; a group solves complex problems by tapping their expertise in creating a new reality. This has to be the cornerstone of any education in informatics. Introducing this way of thinking early will create innovative solutions and make the students look for opportunities at the interfaces of disciplines and cultures.

Training engineers for the 21st century—sure to be dominated by the use of engineering informatics—presents a challenge that all universities will have to face in the coming years. Students will need education not just in their disciplines and the resulting products but also in the process of creating those products, working both individually and in groups within and across nations, and about the context of the world in which these products will be used. The world of product development is at the interfaces of different disciplines, and the ability to manage and exchange knowledge and information will be critical to any endeavor. The ubiquity of informatics will be the future, and it will be technically and socially embedded in a manner that is more complex than we have managed to date.

We will now relate the contribution of the papers in this special issue to the framework laid out in the foregoing remarks.

The paper titled "Engineering Complex, Information-based, Networked Industrial Systems: A Research Roadmap" by Albert Jones et al. argues for a new and different approach to design and understand what they termed as complex information-based networked industrial (CINI) systems. The authors borrow concepts and principles from physical systems and living systems to model and simulate CINI systems. They describe the four major model-building activities of engineers that relate to CINI systems, namely, topology, behavior, decisions, and information. The paper summarizes existing approaches, discusses a few representative applications of those approaches, and provides a list of new research areas. The contribution of the paper is at the system level design of processes for networked system and the role of other disciplines, namely, life sciences, physical sciences, HCI, cognitive and engineering science.

The paper by Vijayakumar and Chakrabarti, "Understanding the Knowledge Needs of the Designers During Design Process in Industry," addresses issue of identification of the types of knowledge required by a designer. Using the taxonomy of knowledge types, they perform empirical case studies[5] using a combination of analyses of observations, questionnaires, video recordings, and design documents and data to arrive at the information needs. This paper is an example of the use of methods from social science in understanding the information needs of designers while using a model of knowledge types; an information classification modeling approach to identify the needs for information retrieval and reuse.

Li et al., "Developing Engineering Ontology for Information Retrieval," describe an ontology based engineering information retrieval system from unstructured engineering documents. Information retrieval (IR) deals with retrieval of unstructured data, in response to a query. Many effective methods of automated IR have been proposed and implemented and it is one of the critical tools needed for corporate knowledge base. The authors also describe a process for developing an engineering ontology. The paper demonstrates how ontology based IR for engineering information performs better than traditional keyword-based search for a sample set of engineering documents. Though the paper deals with engineering informatics in general, its main contribution is domain information models and IR.

The paper by Ameri and Dutta, "A Matchmaking Methodology for Supply Chain Deployment in Distributed Manufacturing Environments," addresses the problem of designing a supply chain network from a set of suppliers based on supply and demand criteria and the advertisement of manufacturing services by the suppliers. The authors have developed a language similar to web services description language (WSDL) and manufacturing services description language (MSDL) using description logic. They have created a match making algorithm that uses MSDL constructs and graph matching algorithms to create an optimal supply chain. The paper also includes an initial evaluation of the performance of the algorithm. The paper falls under the topics of multiagent systems, information models, and algorithmic system level composition of services. The paper is an example of a method for design of particular class of networks.

The paper titled "Ontology Based Trajectory Simulation Framework," by Durak et al. presents a framework for trajectory simulation of missiles. In this paper, the authors use information modeling techniques (description logic) from ontology research in creating a hierarchy of simulation software modules and domain object classes. Using this framework, they provide a platform for reuse of software modules and domain concepts in creating a simulation workbench. The authors have brought together concepts from software engineering and information modeling, model driven approaches, and ontology based modeling for domain concepts used in the trajectory prediction of missiles. The paper addresses the support for composition and execution of simulation models.

In the paper by Bohm et al., "Using a Design Repository to Drive Concept Generation," the authors address the issue of creating design repositories that use functional basis and component naming conventions in the characterization of artifacts. Furthermore, the method for building and searching the design repository uses a mapping between the morphological characterization of the functions and function solution concepts. The goal of the design repository is to provide support for creating design concepts based on the functional specifications provided by the user of the system. In this paper, the authors identify a representational scheme for indexing functions and solution concepts and search a method to support individual designer during conceptual design.

The paper by Wong, et al., "Knowledge Transfer: From Maintenance to Engine Design," recognizes the fact that knowledge created during the service of the product is critical for products that are sold with service contracts. The paper describes a system that uses documents generated during maintenance operations to create a document summary and an ontology based encoding and indexing of the information using resource description format (RDF) triples. Using this base set of information in RDF, their system uses service-oriented architecture (SOA) to make this information available to a number of services that include information retrieval, analysis, and other services that can use the maintenance information in RDF format. The paper is an integration of knowledge extraction from documents, modeling of the information using ontology, and standards based information network provision of the service for distributed needs and enterprise. The

[5]For a more comprehensive discussion on use of empirical studies in designing of support systems see "The Role of Empirical Studies in Understanding and Supporting Engineering Design," edited by Subrahmanian et al., Delft University Press, 2004.

system was tested for validity using industrial case studies. This paper is an exemplar of design and validation of an information system to support knowledge transfer.

In the paper "On Architecting and Implementing a Product Information Sharing Service," Srinivasan et al. describe an open standards based framework for product information composition and service orientation. The framework uses OMG PLM Services 1.0 specifications, SOA, and PROSTEP's OPENPDM software. The authors also point out that this product information sharing service is one of the first industrial examples of successful application of service-oriented architecture to product lifecycle management. This paper, in its use of information modeling, identifies two different classes of objects: engineering objects and business objects. The authors argue for this division as a means to be able to integrate services required for ERP, bill of materials, and other business oriented processes with engineering information generated using authoring tools such as CAD and PDM systems. The paper brings together ideas from information modeling, standards, and distributed system processes.

The paper by Fenves et al., "CPM: A Core Model for Product Data," describes a set of core components (core product model (CPM)) that defines an abstract model for a product representation. CPM includes three aspects of a product or artifact: its function, form, and behavior as first class objects. It is argued that CPM can support purely functional reasoning about a product in the conceptual stages of design as well as the recording and modeling of its behavior in the postdesign stage. This paper covers product modeling using information modeling paradigms.

We are entering an era of networks where different infrastructural networks can be connected through information networks. The information network can connect the manufacturing network to the design and supply chain network in almost real time using information systems that include sensors and ID tags. One's imagination is the limit in this integrative power of information networks. It is this new complex world that we need to teach students, among other things, the ability to reflect on the information they use and how to handle this information, what it means to use (or not) computational tools, the need to create tools at different scales of inquiry and across disciplines, and how to view one's own discipline from an engineering informatics point of view.

The need for such an understanding of engineering informatics is expressed by the following quotes from the NAE report. This report identifies the challenges in use of information technology in social and technical contexts including the need for development of new methodologies for design of two kinds of complex systems.

> "Large-scale systems are IT systems that contain many (thousands, millions, billions, or trillions or more) interacting hardware and software components. They tend to be heterogeneous…. tend to span multiple organizations (or elements of organizations) and have changing configurations. Over time, the largest IT systems have become ever larger and more complex, and, at any given point in time, systems of a certain scale and complexity are not feasible or economical to design with existing methodologies.
>
> Social applications of IT serve groups of people in shared activities. The most straightforward of these applications improve the effectiveness of geographically dispersed groups of people who are collaborating on some task in a shared context. …Characteristic of social applications of IT is the embedding of IT into a large organizational or social system to form a "socio technical" system in which people and technology interact to achieve a common purpose—even if that purpose is not obviously social, such as efficient operation of a manufacturing line (which is a conjunction of technological automation and human workers) or …Social applications of IT—especially those supporting organizational and societal missions—tend to be large-scale and complex, mixing technical and non technical design and operational elements and involving often-difficult social and policy issues such as those related to privacy and access."

While the papers in this issue do not span the entire domain of engineering informatics, they do provide a sampling of efforts in the field. There are several other areas such as archival science for engineering information, standards development for product, process and infrastructure modeling, collaborative management of information, performance measurement, and testing of engineering informatics systems and information metrology. The goal of this issue was to create a dialogue in the definition and consolidation of engineering informatics as a discipline.

The editors would like to acknowledge the comments and suggestions of Professor Steven Fenves in improving the editorial but all the mistakes remain ours.

Eswaran Subrahmanian
Sudarsan Rachuri
Guest Editors

Disclaimer: Commercial equipment and software, many of which are either registered or trademarked, are identified in order to illustrate certain concepts. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.