Computer-Aided Design 45 (2013) 204–228

Contents lists available at SciVerse ScienceDirect

Computer-Aided Design

journal homepage: www.elsevier.com/locate/cad

The evolution, challenges, and future of knowledge representation in product design systems

Senthil K. Chandrasegaran [a], Karthik Ramani [a,b,*], Ram D. Sriram [c], Imré Horváth [d], Alain Bernard [e], Ramy F. Harik [f], Wei Gao [a]

[a] School of Mechanical Engineering, Purdue University, West Lafayette, IN, 47907, USA
[b] School of Electrical Engineering (by courtesy), Purdue University, West Lafayette, IN, 47907, USA
[c] National Institute of Standards and Technology, Gaithersburg, MD 20899, USA
[d] Faculty of Industrial Design Engineering, Delft University of Technology, 2628CE Delft, The Netherlands
[e] Ecole Centrale de Nantes, 44321 Nantes Cedex 03, France
[f] Department of Industrial and Mechanical Engineering, Lebanese American University, Byblos, Lebanon

ARTICLE INFO

Article history:
Received 11 May 2011
Accepted 16 August 2012

Keywords:
Knowledge representation
Knowledge capture
Knowledge management
Product design
Computational tools
Ontology
Systems engineering
Design rationale
Multidisciplinary modeling
Virtual reality
Collaborative engineering
Simulation

ABSTRACT

Product design is a highly involved, often ill-defined, complex and iterative process, and the needs and specifications of the required artifact get more refined only as the design process moves toward its goal. An effective computer support tool that helps the designer make better-informed decisions requires efficient knowledge representation schemes. In today's world, there is a virtual explosion in the amount of raw data available to the designer, and knowledge representation is critical in order to sift through this data and make sense of it. In addition, the need to stay competitive has shrunk product development time through the use of simultaneous and collaborative design processes, which depend on effective transfer of knowledge between teams. Finally, the awareness that decisions made early in the design process have a higher impact in terms of energy, cost, and sustainability, has resulted in the need to project knowledge typically required in the later stages of design to the earlier stages. Research in design rationale systems, product families, systems engineering, and ontology engineering has sought to capture knowledge from earlier product design decisions, from the breakdown of product functions and associated physical features, and from customer requirements and feedback reports. VR (Virtual reality) systems and multidisciplinary modeling have enabled the simulation of scenarios in the manufacture, assembly, and use of the product. This has helped capture vital knowledge from these stages of the product life and use it in design validation and testing. While there have been considerable and significant developments in knowledge capture and representation in product design, it is useful to sometimes review our position in the area, study the evolution of research in product design, and from past and current trends, try and foresee future developments. The goal of this paper is thus to review both our understanding of the field and the support tools that exist for the purpose, and identify the trends and possible directions research can evolve in the future.

1. Introduction

Design is a process that is omnipresent—it arises not only in organizations and projects, but also in day-to-day life. Using existing furniture that seats six people to come up with an arrangement to accommodate ten or using a rubber band to fix a rattling lever on a bicycle can be viewed as a design problem, but a relatively simple problem. On the other hand, designing an electric vehicle or an IC chip is very complex. Design problems vary in complexity, variety, and arise in all disciplines. Tong and Sriram [1] describe design as "a process that constructs a description of an artifact, process, or instrument that satisfies a (possibly informal) functional specification, meets certain performance criteria and resource limitations, is realizable, and satisfies criteria such as simplicity, testability, manufacturability, and reusability". They also differentiate engineering design from this definition: "engineering design involves mapping a specified function onto a (description of a) realizable physical structure—the designed artifact". This mapping between function and structure is often complex: sometimes, a direct correlation cannot be drawn from a specified function to the behavior of a specified structure.

* Corresponding author at: School of Mechanical Engineering, Purdue University, West Lafayette, IN, 47907, USA.
E-mail address: ramani@purdue.edu (K. Ramani).

0010-4485/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cad.2012.08.006

This complexity may even be inherent in the required function, which may require a complex organization of a large number of parts to satisfy it. These parts may interact in many ways, and thus the different ways of designing the parts also interact with each other. These interactions are further complicated by non-functional requirements, or certain inherent behaviors of the structure that are not necessarily required. Thus, in design problems, the mapping of function onto a structure is generally not straightforward.

Tong and Sriram [1] use the term 'ill-structured' in reference to such problems. Rittel and Webber [2] use the term "wicked problems" to characterize the design problem. They compare it to any 'tame' problem, for which "an exhaustive formulation can be stated containing all the information the problem-solver needs for understanding and solving the problem". On the other hand, "to describe a wicked problem in sufficient detail, one has to develop an exhaustive inventory of all conceivable solutions ahead of time". In addition, the design problem representation is often ill-defined to begin with and becomes better defined with increasingly formal specifications as it is gradually solved. Such factors put a substantial demand on the necessity of a wide variety of knowledge sources—heuristic, qualitative, quantitative, and so on. The design problem is then solved in a multi-stage, iterative, and collaborative process, with extensive communication and coordination among teams of experts in various disciplines.

Pahl and Beitz [3] classify the activities of designers into: (1) conceptualizing, (2) embodying, (3) detailing, and (4) computing, drawing, and information collecting. While the first three categories are divided in time in the design process, meaning that they occur in specific stages in the design process, the fourth category of activity takes place throughout the design process. By the end of the design process, there is substantial information accumulated that would potentially be useful for future designs if this information were sent back to the designer at earlier stages in the design process. Today, there is enormous pressure on the designer in terms of demands for a faster turnaround time, lower margin for error, efficiency in managing resources not just for the product but also for the design process, a greater need to collaborate in multidisciplinary teams. All this is further complicated by the virtual explosion in the volume of data that needs to be processed in order to make better-informed decisions. The process of converting data to information, and subsequently to knowledge, and to represent, store, and use that knowledge has become increasingly crucial.

While knowledge is viewed as structured information, it can also be considered as information in context. This context depends on a number of variables—the product being designed, the organization, the design philosophy followed, the particular stage of design at which the knowledge is being used, and, most important of all, the mind of the designer. While it is important to structure and organize data for easy retrieval and reuse, it is also important to understand that neither the mind of the designer, nor the process of design ideation, follows a specific structure or sequence. Earlier computer-aided design systems were driven more by the computer than by design, thus forcing the designer to learn procedures to use different computational tools. This often impedes with the ideation process and creativity of the designer. The computer-aided design tool of the future needs to be driven more by the design rather than by the computer [4]. Research in product design needs to bridge the gap between the unstructured, disruptive ideation process that the designer is comfortable with, and the structured way of storing and indexing knowledge for retrieval through a computer. When managing knowledge, it is essential to keep in mind the points of view of different users to the same knowledge, the redundancies or gaps in knowledge caused by different software, and the fact that knowledge is often transformed by enterprize processes [5]. Goel et al. [6] propose four characteristics of the next generation computer-aided design (CAD) systems:

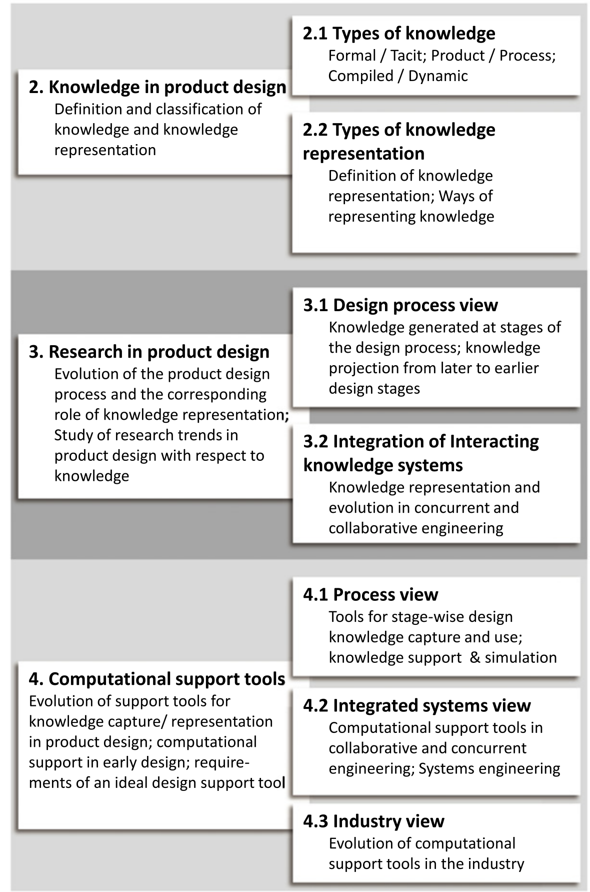


Fig. 1. Paper overview.

- Focus on conceptual design, especially creativity,
- Emphasis on creative design including analogical design,
- Support for collaborative design, and
- Grounding in design cognition.

They describe the DANE (Design by Analogy to Nature Engine) system for supporting cross-domain analogies in biologically-inspired design as a prototype. Understanding design-cognitive perspective is rapidly becoming an important area which may provide answers that would drive the next-generation computational support tools.

In this paper, we attempt a review of the product design process from a knowledge capture and representation perspective, keeping in mind the designer, the design process, and the current demands of the market on the product design industry. Most of the potential challenges such as the vastness and diversity of research, were reduced by the diverse fields of expertise of the authors involved in writing this review, as well as their geographical locations. We focused on a time frame of the last 20 years in general and the last decade more specifically, in order to balance the breadth and depth of the review, relaxing the timeframe where it was necessary to look further in the past to establish relevance. Publications in major journals like Computer-Aided Design, Journal of Computing and Information Science in Engineering, Artificial Intelligence in Engineering, Design and Manufacturing, and Journal of Mechanical Design were referred to seed our search, which spread out to include literature from other diverse areas. However, it still is possible that we have overlooked research worth mentioning here, and hope that adequate reader feedback will help us in future reviews that we may undertake.

The paper structure, indicated in Fig. 1, is as follows: Section 2 discusses the perception of knowledge, its types and its representations with respect to product design. Section 3 looks at knowledge processing in design from two perspectives: (1) individual applications (e.g. conceptual design, modeling and analysis, etc.), and (2) integrated system view, especially collaborative design. Section 4 discusses the computational tools that are being developed for design and how these tools handle knowledge representation. Finally Section 5 discusses future trends in research in this area. Individual sub-sections will be accompanied with similar figures to provide a roadmap to the reader.

2. What is knowledge in product design?

The question of "What is knowledge?" can have a variety of answers, as there are various meanings of the term even in the context of engineering and design. Knowledge is not directly available but is obtained by interpretation of information deduced from analysis of data. Data is available to an organization in the form of observations, computational results and factual quantities. Interpretation, abstraction or association of this data leads to generation of information. Finally, knowledge is obtained by experiencing and learning from this information and putting it into action [7]. In fact, looking at engineering design from a teleological point of view, it can be said that the primary function of engineering design research should be to transform empirical or rational knowledge into a form that can be used for practical deployment [8].

Sainter et al. [9] describe knowledge as the "experience, concepts, values, beliefs and ways of working that can be shared and communicated". Sriram [10] describes knowledge in the context of intelligent engineering systems as "something that an intelligent being possesses and utilizes for problem solving". Sunnersjö [11] uses the term 'knowledge' in terms of design as an understanding of given information—its content, its origins, and its applicability. He argues that "the knowledge should include not only the rules that the designer should adhere to, but also the background knowledge that makes the design rules possible to review and understand".

The definitions and understanding of knowledge within the realm of product design are varied, depending on various contexts. The question "what is knowledge?" is likely to produce different answers when posed to different design teams for the same product, or to analogous design teams of different products, or even to similar design teams in different organizations. While it is important to understand and define what knowledge is, it is equally important to understand that the definition of knowledge depends upon the context. While all the above definitions are valid, it would be unwise to apply any one of the above definitions to all aspects of product design.

2.1. Classification of knowledge

Classifying knowledge, just like understanding knowledge, is crucial in order to determine ways to represent it [12]. In the field of design and engineering, knowledge can be classified along several dimensions. Each classification has its own basis, as shown in Fig. 2.

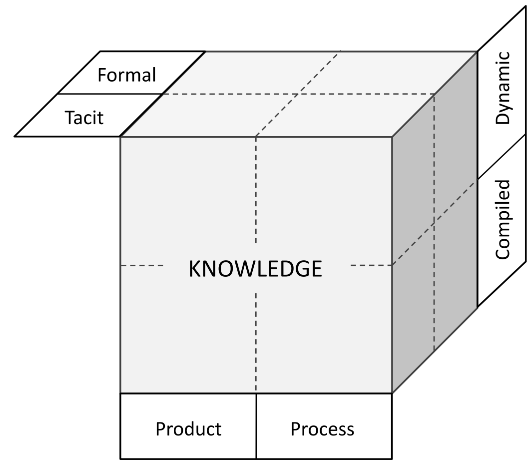


Fig. 2. Classifying knowledge—the different dimensions.

Formal Vs. Tacit. Formal knowledge is embedded in product documents, repositories, product function and structure description, problem solving routines, technical and management systems, computer algorithms, expert knowledge systems, etc. [7]. These create the intellectual platform necessary to build and manufacture a product. On the other hand, knowledge tied to experiences, intuition, unarticulated models or implicit rules of thumb is termed tacit. Nonaka [13] popularized the concept of tacit knowledge, by highlighting the problems of the previously narrow approach to knowledge, and suggesting a more holistic approach capturing tacit and often subjective insights, experiences, and intuitions of personnel. Tacit knowledge is necessary to create new value in a product. It exists as the intellectual property of designers or a particular design team directly involved in the product development effort. It is generally gained over a long period of time with learning and experience, is difficult to express, and can only be transferred by the willingness of people to share their experiences. Unfortunately, this knowledge is also lost with the loss of the person or team from the organization.

Product Vs. Process. Product knowledge includes various pieces of information and knowledge associated with the evolution of a product throughout its lifecycle. This includes requirements, various kinds of relationships between parts and assemblies, geometry, functions, behavior, various constraints associated with products, and design rationale. Process knowledge can be classified into design process knowledge, manufacturing process knowledge, and business process knowledge. Design process knowledge, which can be encoded as methods in a product representation, provides mechanisms for realizing design details at various stages of the product lifecycle. Manufacturing process knowledge is mainly concerned with activities associated with the manufacturing floor [14]. Business process knowledge includes all processes associated with marketing, strategic planning, supply chain management, financial, and other associated functions. While product and process knowledge are not independent of each other, they are distinct aspects of the dimension, and hence merit separate consideration.

Compiled Vs. Dynamic. Compiled knowledge is essentially knowledge gained from experience that can be compiled into rules, plans or scripts, cases of previously solved problems, etc. In compiled knowledge the solutions are explicit. Dynamic knowledge encodes knowledge that can be used to generate additional knowledge structures, not covered by compiled knowledge. In dynamic knowledge the solutions are implicit. Dynamic knowledge can be classified into qualitative knowledge and quantitative knowledge. At the qualitative level, the knowledge may consist of: commonsense reasoning, approximate theories, causal models of processes, general problem solving knowledge, etc. The quantitative level could consist use of: constitutive, compatibility, equilibrium equations (physical laws), numerical techniques, closed form equations, etc. [10].

2.2. Classification of knowledge representation

Bringing knowledge forward and making it explicit is one of the key roles of knowledge representation. Davis et al. [15] describe knowledge representation in terms of five roles, as

Pictorial	Symbolic	Linguistic	Virtual	Algorithmic
Sketches	Decision tables	Customer Requirements	CAD Models	Mathematical Equations
Detailed drawings	Production rules	Design Rules, constraints	CAE Simulations	Parametrizations
Charts	Flow charts	Analogies	Virtual Reality simulations	Constraint Solvers
Photographs	FMEA diagram	Customer feedback	Virtual prototypes	Computer Algorithms
CAD model views	Assembly tree	Verbal communication	Animations	Design/ operational procedures
	Fishbone diagrams		Multimedia	
	Ontologies			

Fig. 3. Classification of knowledge representations: Based on Owen and Horváth's [7] classification, with representative examples from product design.

1. A substitute for an entity such that one can determine effects by thinking rather than acting,
2. A set of ontological commitments or ways of thinking about an entity,
3. A part of intelligent reasoning expressed in terms of sanctioned and recommended inferences,
4. A computational environment for thinking, and
5. A medium of human expression.

Sowa [16] describes knowledge representation as a multidisciplinary subject that combines techniques from logic, ontology, and computation. The origins of knowledge representation lie in research in artificial intelligence, but its influence has extended to many fields, including the design process. Bernard and Xu [17] propose a reference framework for the representation and the characterization of knowledge that is applicable to all types of knowledge. Like knowledge, knowledge representation can be classified as well. Owen and Horváth [7] classify knowledge representations into five categories: pictorial, symbolic, linguistic, virtual, and algorithmic. Fig. 3 shows some of the various representation forms with respect to product design for both product and process knowledge, with some examples.

From Fig. 3, it immediately becomes apparent that there are various representations that can fit under more than one category. For example, fishbone diagrams can fall under both pictorial as well as symbolic representations. Assembly charts can fall under both algorithmic and pictorial representations. Thus while the classification of knowledge representation is helpful, it does not mean that there is a clear line of demarcation between different representations.

There have been well-established forms of representing mathematically and geometrically related knowledge. Experimental knowledge is also well represented by simulation analysis and virtual reality prototypes. Researchers have been working on developing means of representing tacit heuristic knowledge. Mapping heuristic knowledge to a physical form is a very hard problem, these mappings are not unique. Heuristic methods based on geometry attributes, composition, and inheritance for determining mapping in engineering ontologies are still developing areas. Developments in this area will lead to the enhanced ability to create, share, and exchange knowledge for solving design evaluation problems [18]. Since knowledge can be seen as information in context, it follows that the representation of knowledge would depend on both the content and the context of the information. A good product design support tool should therefore have the ability to not only capture knowledge through the design process, but also represent it in ways that reflect the relevant context. A significant part of research in product design is thus concerned with this capture, representation, and reuse of knowledge. The next section takes a knowledge-oriented look at previous and ongoing research in product design.

3. Research in product design: the knowledge perspective

Engineering design is the activity of finding solutions to technical problems by applying insights from natural and engineering sciences, at the same time taking into account the conditions and constraints of a given task [3]. Engineering design research is the instrument of exploration, description, arrangement, rationalization, and utilization of design knowledge [19]. Thus, simply put, research in engineering design is a way to explore, understand, and use design knowledge, so that it may be used to find solutions to certain problems.

Research in engineering design has developed rapidly in the recent past, and as a result, the development is often disjoint in nature. A framework for reasoning has been proposed by Horváth [8], based on the teleology of research in engineering design. As mentioned earlier in Section 2, Horváth argues that one goal of design research is to transform knowledge from various fields into a form that can be used for practical applications. On this basis, engineering design knowledge and research have been broadly classified into *source categories, channel categories, and sink categories*. Source categories include forms of knowledge that provide the fundamental reasoning mechanisms for engineering design. Channel categories provide knowledge that help couple theoretical knowledge with practical (heuristic) knowledge. Finally, sink categories concern knowledge required for the deployment/ application of engineering design knowledge.

For more than three decades, researchers have been working towards better ways of representing design as a synthesis procedure, which was significantly influenced by Alexander [20] and Simon [21]. Early research on computer support for engineering design concentrated on problem-solving techniques [22–24]. Blackboard systems in artificial intelligence, with their flexible organizational principles, their ability to decompose problems into loosely coupled subproblems, and the provision for knowledge abstraction, were suggested as appropriate to handle the ill-structured and knowledge-intensive nature of design problems [25]. Establishment of CAD/CAE (Computer-Aided Engineering) systems led to widespread research in the area of parametric model generation, geometry modeling and constraint solving [26].

On the product design and manufacturing front, it was becoming clear to organizations that there were pitfalls to vertical integration, where all activities in product development were done in-house: while it made coordination easier, the assets needed to support vertical integration were inordinately expensive [27]. In addition, managing the wide range of skills and activities that are required to develop products of increasing complexity also becomes uneconomical. Outsourcing parts of the product design and manufacturing came to be seen as more economical. In the realm of product design, this led to an increasing need for collaborative design processes and means to communicate information pertaining to the requirements, constraints, specifications, features, and associated decisions and processes. On the other hand, technical breakthroughs by an organization higher up on the supply chain like Intel and Shimano provide an incentive for profitability of the proprietary systems developed, thus pushing the market toward a more vertical integration. Fine [28] calls this cycle of alternating vertical integration and disintegration to modular products the "Double Helix" of the industry, and argues that an organization's key competence lies in designing its value chain network—deciding what to and what not to outsource.

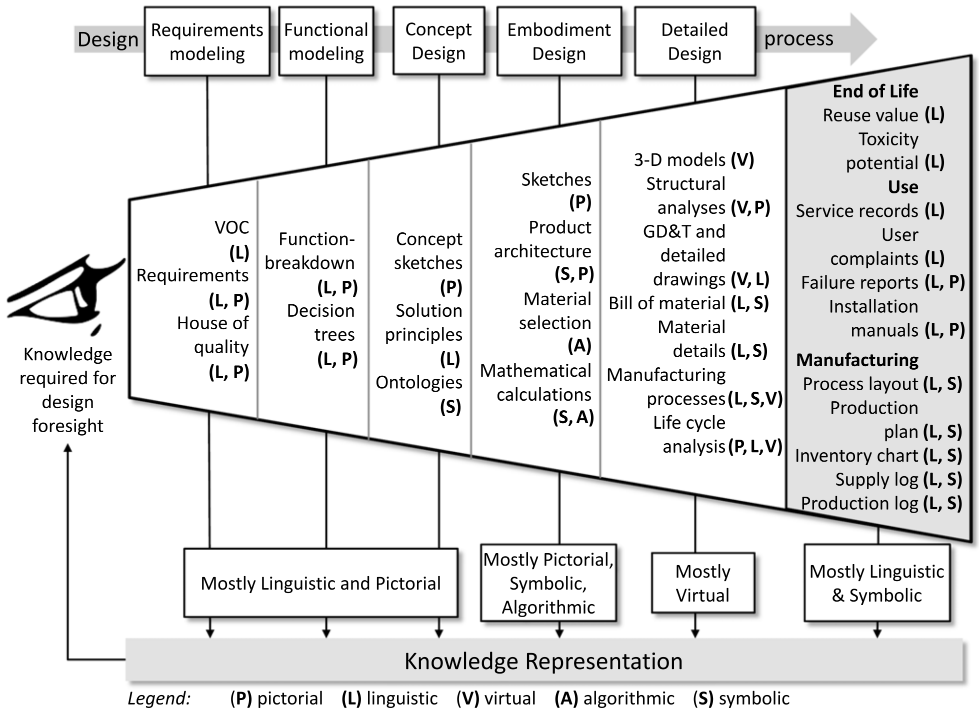


Fig. 4. Knowledge representations in product design. The design stages shown at the top are based on Pahl and Beitz [3]. The columns below each stage of design show examples of representations of knowledge used at each stage. These are then tagged as (P)ictorial, (S)ymbolic, (L)inguistic, (V)irtual, and (A)lgorithmic, using Owen and Horváth's [7] classification of knowledge representations.

The current challenges faced by the engineering design industry are the need to attract and retain customers, the need to maintain and increase market share and profitability, and the need to meet the requirements of diverse communities [29]. Tools, techniques, and methods are being developed that can support engineering design with an emphasis on the customer, the designer, and the community. The development of *decision support tools* and *methods for achieving integration* can be seen as two prongs in design research. It has been speculated that 75% of the manufacturing cost is committed early in the design process [30], when the knowledge of the product is unclear, incomplete, and difficult to represent. It is therefore essential to equip the designer with effective tools that help make better-informed decisions, and better explore the design space in the early stages of design.

The explosion in the volume of data that could be shared and the need for collaborative engineering to increase competitive advantage in this era of globalization requires companies to adequately store, transfer and make maximum use of the available knowledge. Lack of a unified protocol for knowledge representation is one primary reason for lack of interoperability of design support tools, in particular in the early design stages [31]. Developments in information technology led to better digitization of information, which started the development of systems like Product Data Management (PDM) and later Product Lifecycle Management (PLM) that captured information about a product throughout its lifecycle. Looking at the product and the product design process from an integrated point of view is crucial to collaborative engineering, especially for inter-disciplinary, complex systems [32]. Managing the effective communication of product knowledge and appropriately representing this knowledge among different groups then becomes the challenge.

The challenges of knowledge modeling and representation can thus be seen from two perspectives:

1. The encoding of design product and process knowledge at different design stages in a way that will lead to better quality design, and
2. The capture, use, and communication of knowledge between different individuals, teams, and organizations.

The first challenge is more apparent when considering design processes or events that are largely sequential, while the second challenge is predominant when considering design processes that occur concurrently or simultaneously. Product design involves both kinds of processes, so we will consider both views of knowledge.

3.1. Design processes view of knowledge

The design process view of knowledge is concerned with knowledge that is generated and used at various stages of the design process. Fig. 4, based on Pahl and Beitz's [3] idea of process flow in design, shows the design process and, based on the authors' experiences, lists various forms of knowledge that are used or produced at each stage. Please note that this is a representative figure, indicative of the predominance of certain forms of knowledge in certain stages of design, and other forms of knowledge often spill over from one stage to another. It can be seen that in the early stages of design, knowledge representation is predominantly linguistic and pictorial in nature. The other representations such as symbolic, virtual, and algorithmic appear in the embodiment design onwards, when much of the design is already committed. More and more information is accumulated as design reaches the embodiment and detailed design phases, and the challenge is to reuse or re-inject this knowledge into the earlier phases of design using an appropriate representation. This section thus discusses the knowledge-intensive aspects of early design, such as ideation, concept generation and its interaction with the designer's mind, and the different types of modeling used.

Fig. 5 outlines the structure of this section, and gives an overall view of the topics discussed in the section.

3.1.1. Conceptual design

Ullman defines a concept as "an idea that is sufficiently developed to evaluate the physical principles that govern its

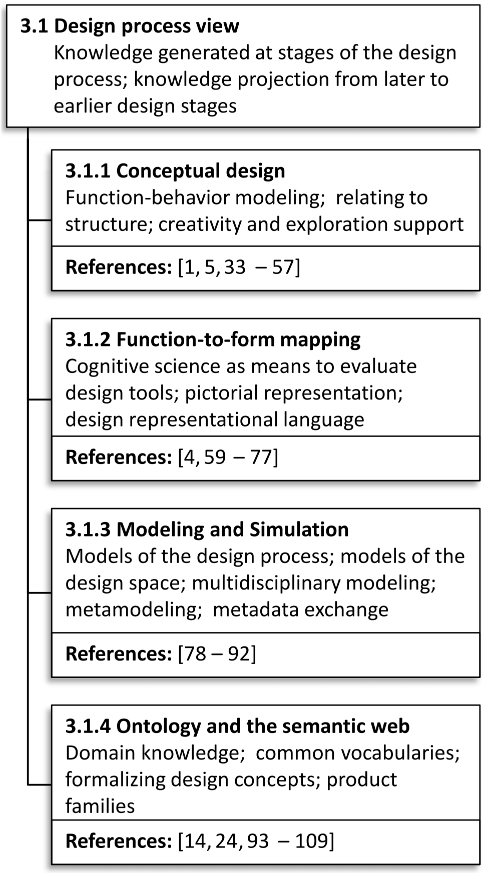


Fig. 5. A design process overview of knowledge.

behavior" [30]. Thus it is best to generate concepts whose behavior(s) are modeled based on the function(s) of the product being designed. The function-behavior based modeling approach was the focus of research in the 1990s. Research on conceptual design evolved from using top–down refinement and constraint satisfaction strategies in the 1980s [1] to using case-based/analogy-based [33] and Function-Behavior-Structure (FBS) modeling in the 1990s [34] and its extension into an ontology in the 2000s [35]. The FBS Ontology categorizes the properties of an object into into three levels: function, or "what the object is for", the behavior, or "what the object does", and the structure, or "what the object consists of" [36].

Developing different techniques in functional modeling provides an abstract and direct method for understanding and representing an overall product or artifact function without reliance on physical structure [37]. An FBS modeler was developed by Umeda [38] and extended by others [39,40] to support functional design in the analytic as well as synthetic phase according to the designer's intention. Stone and Wood [41] developed a design language called a functional basis to characterize product function using a verb-object format.

The process of synthesis involves the transformation of the intended behavior into structure [35]. Welch and Dixon [42] propose a transformation of functional requirements to a new representation of behavior called behavior graphs by mapping the desired parameter relationships to relationships based on physical principles and phenomena. Neville and Joskowicz [43] present a language for describing the behavior of fixed-axes mechanisms, which uses predicates and algebraic relations to describe the configurations and motions of each part of the mechanism and the relationships between them. Mervi [44] presents a mapping from the functional model to a physical structure involving specific shape, dimension and relations between the components for creating an assembly family. Qian and Gero [45] argue a topological structure for design by analyzing function and structure representation in different design domains.

Gorti and Sriram [46] develop an approach to mapping an evolving symbolic description of design into a geometric description, which derives spatial relationships between objects as a consequence of the functional relationship and presents the evolving descriptions of geometry. Damski and Gero [47] demonstrate a framework to represent graphical shapes as predicates in logic in two dimensions, which is applied to demonstrate shape concepts associated with topology and emergence. Sheu and Lin [48] present a representation scheme with five basic constituents: B-rep solid components, measure entities, size, location and constraints; here, dimensions are used to determine the size and locations of form features. However, the scheme requires a large shape feature database and constraint library to be established in advance. Dani and Gadh [49] describe a virtual reality environment–covirds (COnceptual VIRtual Design System) for creating concept shape designs, which uses a bi-modal voice and hand-tracking based user interface in determining the shape and dimensions of the product rather than specifying shape and dimensions in traditional CAD models. Taura et al. [50] propose the shape feature generating process model (SFGP model) representing free form shape features with the aim of making the system capable of holding and manipulating the shape features after synthesis to support early design process. Labrousse and Bernard [5] proposed an extended conceptual and operational approach based on a FBS-PPRE (Process-Product-Resource-External effect) generic model for the modeling and the life-cycle management of all enterprise objects and in particular adapted to design issues.

Additionally, research has focused on supporting design creativity and exploration of conceptual solutions, managing uncertainty in conceptual design, and methods that support the flow of cognitive processes of designers during conceptual design. Gero and Maher [51] use analogical reasoning and (mathematical) mutation as computational processes, and use an explicit representation of design knowledge to support design creativity. Horváth and van der Vegte [52] represent design concepts using a new product modeling methodology called the "nucleus theory" on the basis that all engineering products can ultimately be broken down into physically coupled pairs, and this lowest level entity or nucleus carries morphological and functional information and can be used as a generic modeling pattern. Ölvander et al. [53] use equation-based models to quantify, and then optimize, the morphological matrix. Malak et al. [54] propose a set-based approach to concept design that enables systematic arrival at a solution, in spite of imprecise characterizations of design concepts. Shai et al. [55] revise and use the infused design approach [56,57], and propose a method for supporting creative concept design.

3.1.2. *Function to form mapping and representation*

From 1990 to 2000, research in product representation focused chiefly on interactions between designers, between designers and computers, and between designers and users, especially in the early stages of design. In the last few decades, the importance of cognitive science has emerged through the increasing focus on creativity in design thinking, and in the ergonomic and psychological evaluation of design tools like CAD and CAE. Since social and cognitive requirements drive the design process throughout, computers needed to be designed to facilitate human activity and experience [58].

Lawson [59] points out that "of all the questions we can ask about design, the matter of what goes on inside the designer's head is by far the most difficult and yet the most interesting and vital". Design involves a highly organized mental process: manipulating many kinds of information, blending them all into a coherent set of ideas and finally generating some realization of those ideas. The challenge of psychological studies on the novice designers' mental process during creative design is to know whether they effectively represent the problem when a set of constraints needs to be satisfied. Expert designers, in comparison, have sufficient knowledge and ability to stand back from the specifics of the accumulated examples, so more abstract conceptualizations and problem-solving strategies are pertinently established [60] and variable preference constraints are simultaneously satisfied [61] in the process of design.

The design process, especially creative design, involves the use of visual imagery using pencil-paper or digital freehand tools [59]. Research in cognitive science has suggested a strong connection between semantic attributes and visual attributes [62], which explains the frequent use of pictorial representation, especially sketches and drawing, among designers. Designers are visual and not verbal people [63], and engineers often find the need to sketch their rough ideas in order to think [64]. Schooler et al. [65] argue that verbalization interfered with solving insight problems that require non-verbal information including visual reinterpretation. Goel [66] demonstrates that paper based systems support creative thinking by facilitating lateral transformations because of the qualities of denseness and ambiguity within the mark itself. Stones and Cassidy [67] report from their study on novice graphic designers that pencil-paper based sketching supports creative thinking in terms of fluency and flexibility in design synthesis, while the computer seemed to hinder them. While they caution against attributing this solely to the computer and not to the designer's skill in using the software, they also draw attention to Schenk's study [68] that highlight designers' preference of generating sketches over using digital tools.

Ullman et al. [64] classify the "marks on paper" engineers make into two broad categories: support notation and graphic representations. Support notation includes explanatory notes, dimensions, calculations etc., while graphic representations include drawings of objects and their functions, plots, and charts. They conduct a detailed study and confirm three of their hypotheses:

1. Drawing is the preferred method of external representation,
2. Sketching is an important form of representation in mechanical design, and
3. Drawing is a necessary extension of visual imagery.

Their fourth hypothesis, that drawings require transformations dependent on the medium, was difficult to support, partly because of the single medium (paper) used in their study, but also because understanding how humans store and manipulate visual information is not very well known. Goldschmidt [69] suggests that the interactive imagery offered by the sketching process introduces a special kind of dialectics in design reasoning, and contrasts this with visual thinking processes that do not involve sketching. Self-generated sketches allow the designer the flexibility to sidetrack, manipulate, and edit or overwrite. Furthermore, the lack of constraints allows the designer to try out anything that comes to mind [70]. Ferguson [71] divides sketches into three categories: the thinking sketch which the designer uses to aid his/ her thinking, the talking sketch which is used for communication in a discussion, and the prescriptive sketch which specifies the design for people not involved in the design. Understanding the cognitive aspects of the first two categories would provide substantial knowledge that could help develop computational support tools for the early stages of design.

Design representational language [72–74] in cognitive and innovative idea generation is another area of research that may provide some clues to appropriate knowledge representation in the conceptual stage. Graphical idea representation, especially sketching, may be better suited for design representation in terms of its straightforwardness, conciseness, and easy flaw-detection. Several cognitive models and relative experiments for novel designs are developed based on cognitive psychology, mental imagery, and visual thinking. The future trends will be the construction and development of specific and precise models to offer more insight into the cognitive processes [75]. There are probably other preliminary design tasks where the computer and symbol systems excel for novice designers. Designers use a range of tools such as words, sketches, computer and sketch modeling in their ideation process in different crucial moments of discovery. As Jonson [76] suggests, "CAD may foster new patterns, relationships, or aesthetics expanding, rather than reducing designers creative options". To do this, however, future systems need to help the visualization of function in the early stages when the geometry is not fully defined [4,77].

3.1.3. Modeling and simulation

The use of models in engineering design have been widespread early in the 20th century, beginning with iconic models or physical representations of a design. This was followed by analog models, or the use of one thing to represent another, leading to analog computers in the 50s [78]. The most commonly used models by engineers are symbolic models, which are predominantly mathematical models. Analog and symbolic models represent increasing levels of abstraction from iconic models, and are less expensive to create. Other models like models of the design process, models that represent design information and knowledge, and models of design sets have also been developed to aid the designer in his/ her work [79]. The design problem itself has been modeled and represented in different ways. Feldkamp et al. [80] model the design problem using a structural model, a taxonomy-based library of solutions, and constraint-propagation techniques. Chen et al. [81] introduce design capability indices, which are based on process capability indices used in statistical process control. These design capability indices are used to determine the capability of a family of designs represented by 'ranged sets' of design solutions in satisfying a ranged set of design requirements. This helps maintain flexibility in the design requirements and parameters early in the design process. Devanathan and Ramani [82] model the embodiment design as a generalized configuration problem using a combination of constraint-satisfaction and optimization techniques. The design can also be modeled as a set of variables that constitute a 'design space' represented by a polytope, in which case a parametric design problem is modeled as a geometry problem, enabling design exploration to be carried out using computational geometry algorithms [83].

Current product development practices require modeling and simulation of many aspects of a given product: appearance, shape, behavior etc. [84]. The heterogeneous, or multidisciplinary nature of today's products has necessitated developments in modeling of multi-physics systems. Michopoulos et al. [84], in their review of computational methods in the modeling and simulation of multi-physics systems acknowledge the importance of experiment-based data gathering from multi-physics systems to drive automated modeling and of managing the representational complexity of these models. Knowledge representations that remain common to all domains, or are easily transformable into the relevant domains, have also been identified as a future area of development in multi-physics models.

Computation-intensive design problems have been common challenges in the design industry today. To reduce the computation burden, approximation techniques such as metamodeling are used. Wang and Shan [85] list the areas where metamodeling can play a role. These areas include model approximation, design space exploration, problem formulation, and optimization support. In their review of metamodeling, the lack of research on large-scale problems and the modeling, sampling, and visualizing challenges posed by these problems are presented. Representing prior knowledge of computationally intensive processes in meta-modeling and intelligent sampling to minimize the number of sampling points to represent a function in a meta-model are stated as some of the challenges.

Designers are no longer merely exchanging geometric and mathematical data, but more general knowledge about design and the product development process, including specifications, design rules, constraints, and rationale. Product Data Technology (PDT) was developed in the late 1970s to support the product information capture throughout its lifecycle in a computer-interpretable form. For PDT to function effectively, there needs to be an adequate exchange standard for product data. The development of the Standard for the Exchange of Product model data (STEP), the Unified Modeling Language (UML), Parts Library (PLIB), the Process Specification Language (PSL), Manufacturing Management Data Exchange (MANDATE) are all exchange specifications for various components, aspects, or processes in product design [86].

Cross-domain analogies have always been a creative way of thinking for designers. Recent approaches in this direction have resulted in more and more biologically inspired concepts and approaches being brought to develop solutions of engineering problems. A deeper understanding of properties of biological organisms such as adaptivity, robustness, versatility, and agility [87] also guide the construction of new representational and functional models [88] during the design process and improve the quantity and quality of the generated design ideas in the conceptual design phase. Recently, there have been some attempts at understanding the processes of biologically inspired design as an early design activity. Vincent and Mann [89] propose the extension of the TRIZ database including biological information and principles. Mak and Shu [90] develop a taxonomy of verbs that relate biological and engineering designs. Helms et al. [91] state that a detailed information-processing model needs to be provided in biologically inspired design, which focuses on the cognitive processes or 'mechanisms'that facilitate and constrain the design practices and products. Mak and Shu [92] find that functional descriptions of biological systems in the form of flow of substances improve generated design ideas. The use and application of biological knowledge models to support design remain to be developed to meet the requirements and challenges in the early stage of a product's lifecycle.

3.1.4. Ontology and the semantic web

While these approaches provide the opportunity for progressive automation of processes in the industry, the use of ontologies can help integrate and migrate valuable, unstructured information and knowledge, and provide a richer conceptualization of a complex domain such as construction [93]. The diversity and complexity of engineering design requires the knowledge representation forms to be both flexible and robust in nature [24]. The interoperability of knowledge based systems also requires a formal protocol for knowledge representation. These systems take past knowledge from the design and interact with the designer in the form of questions and answers [94]. Hence, there is a great interest in developing ontologies for managing knowledge. An ontology is a system of fundamental concepts that allows the designer to model and represent a particular domain of the world around him/her in terms of axiomatic definitions and taxonomic structures [95]. For a particular domain, an ontology is a highly structured system of concepts covering the processes, objects, and attributes of that domain as well as all their pertinent complex relations [96]. It also provides formal definitions and axioms that constrain the interpretation of these terms [97,98]. An ontology is often captured in some form of semantic network—a graph with nodes representing concepts or individual objects and arcs representing relationships or associations among the concepts [99].

Recent research in ontologies is helping to establish common vocabularies and capture domain knowledge, and they have proven to be an advantageous paradigm over recent years [100]. The Core Product Model (CPM) is a generic, abstract model with generic semantics, with meaningful semantics about a particular domain to be embedded within an implementation model and the policy of use of that model. It was built on earlier product representations described in [24]. In addition to traditional geometric representations, CPM supported the notions of form, function, and behavior. The Open Assembly Model (OAM) extended CPM to provide a standard representation and exchange protocol for assembly, which includes for tolerance representation and propagation and kinematics representation [101]. A formal semantic model, based on CPM/OAM, and consistent with OMG's (Object Management Group) Model-driven Architecture, is described in [102]. Witherell et al. [103] showcase the potential value of ontology in representing application-specific knowledge while facilitating both the sharing and exchanging of this knowledge in engineering design. Horváth et al. [104] apply the ontology paradigm to formalize design concepts, using the advantage of the formal specifications to capture domain knowledge at the conceptual level. Bohm et al. [105] develop a method to transform heterogeneous product knowledge into a coherent design repository. They point out seven main groups of information types that comprise the design information captured by the design repository data schema, and organize the information using formal concept analysis (FCA). The FCA is semantically enriched using Web Ontology Language (OWL) to transform stored information into reusable knowledge.

Knowledge Management (KM), developed to capture and reuse knowledge in an organization, can be used to integrate ontological structures into design process management. Holsapple and Joshi [106] identify the lack of a well-integrated framework that would unify KM, and develop a KM ontology (H and J ontology) that can be used and further developed by KM practitioners. A case study by Sicilia et al. [107] on integrating the H and J ontology inside the OpenCyc knowledge base allowed the concept and predicate definitions of OpenCyc to be extended consistently to produce a well-equipped knowledge representation for KM applications. The MOKA European project proposed a KM methodology adopted by different European companies, based on the ICARE ontology used to model and structure knowledge in a full web knowledge-base. Candlot et al. [14] extended this ontology to ICARREF, which is based on the FBS-PPRE model.

Nanda et al. [108] contribute an integration of three steps of knowledge representation, management and application. These steps are to visualize relationships captured in new product data information management and to navigate through the extensive information for designers. Several approaches such as network bill of material (NBOM), product vector matrix (PVM) and function component matrix (FCM) are used to represent and map different design information structures and to facilitate design information management. After storing knowledge through the use of formal knowledge representations, indexing techniques, and the issue of the role of standardized language and terminology, desired knowledge from the repository is retrieved. Once the information is stored in the appropriate format, a product family can be redesigned using either the component-based approach or the product-based approach [109].

A view of the various models used in engineering design shows the variation in the type of knowledge ranging from the primarily symbolic and linguistic ontology-based models that represent abstracted functions of concepts, to mathematical models for exploring and optimizing the design space, to multidisciplinary models, which are used to represent the behavior of designs mathematically or virtually. All these models try to capture and represent different aspects of the product design at different stages of the design process in different design teams. There may not be a single unified model that captures all aspects of a design. In such a situation, the answer may lie in the direction of more flexible systems, or even multiple systems that are best fits for required applications, with means to exchange information between these systems.

3.2. Integration of interacting knowledge systems

The term Concurrent Engineering was first used in the US in 1989 to describe the intent or the method of reducing lead time, thus increasing competitiveness without any detrimental effects on quality or cost [110,111]. This is achieved broadly by (1) integrating product development with design, process planning, and production processes, and (2) within these individual processes, distributing a process into several simultaneous processes. Concurrent engineering today is evident in almost all products, from the automobile to the electric kettle. Thus, product designers have the need to exchange ideas, daily progress reports, and details of actual designs, both within and in between teams, which in turn can be inter-disciplinary, and placed in different locations. This need cannot be effectively met without systems that could allow the seamless transfer of information and knowledge between different teams, locations, systems, and disciplines.

In addition, the product itself can be considered to be an integration of various systems. Such systems thinking is already prevalent in the design of software systems. Product systems, however, have the added requirement of engineering analysis of reliability, robustness, performance, and other parameters [112], which are even more critical at the systems level, when there are unforeseen effects of system-to-system interactions.

Thus the integrated systems view of knowledge refers to both: the knowledge that is generated by interaction between designers, teams, or organizations during the course of a product design, as well as the knowledge concerned with the objectives, processes, and results of disintegrating a product into two or more complex systems during the design processes, and re-integrating these systems to form the product.

In this section, we will discuss the evolution of research in design with such a 'systems' view. The structure of this section is illustrated in Fig. 6.

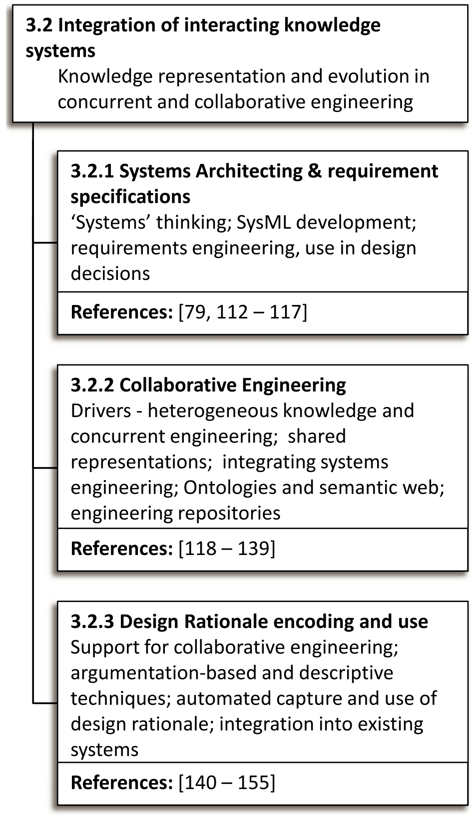


Fig. 6. An overview of the integration of interacting knowledge systems.

3.2.1. Systems architecting and requirements specifications

The complexity of most products and their design processes led to the idea of 'systems' thinking [113], where each product is seen as an agglomeration of systems, each system in turn an integration of subsystems, and so on. Mark Maier [114] define a system as "a set of different components so connected or related as to perform a unique function not performable by the elements alone". The International Council of Systems Engineering (INCOSE) definition of a system is "a collection of components organized to accomplish a specific function or set of functions". The product breakdown structure (PBS) was seen as an intuitive way of looking at the product architecture [115], but the rigid and largely static system could not effectively work in the inherently dynamic product design process.

Chung et al. [116] develop a Product-Node (PN) architecture which is based on the PBS system, but is more flexible and is thus more suited for the design process. A graphical modeling language, Systems Modeling Language (SysML) was developed specifically for specifying, analyzing, designing, and verifying complex systems [112]. SysML is based on the Unified Modeling Language (UML) 2.0. It extends UML to model systems knowledge, namely requirement and parametric diagrams [117,79]. SysML quickly became known as a powerful tool to express the result of a design activity. However, it has some shortcomings, most importantly capturing interconnections within diagrams.

3.2.2. Collaborative engineering

The complexity of modern products poses several challenges to the designer. This increasing complexity necessitates a distributed and heterogeneous collaborative engineering design environment [118,119]. Concurrent design of products and processes has thus become essential, and successful firms today are those that are able to manage concurrent activities in an intra- or inter-organizational network and deliver value to the marketplace.

Cross functional teams, which are the essence of such networks, need a more intensive knowledge exchange process. The knowledge exchanged includes geometry, design rules and constraints, requirements, rationale, etc. Collaborative engineering thus requires the support of computational frameworks. The research for computational support for collaborative engineering has focused on areas like architectural frameworks, shared representations of product information, engineering repositories, constraint management, coordination, conflict mitigation, organizational issues, collaborative design, and decision-based design issues [120].

The product design process builds an information model of the product through connection to the information and expertise that resides in the designing community [121]. This includes personal experience and informal networks of contacts in design [122]. Computational support for the generation and use of shared representations in collaborative engineering needs to include functional abstraction, geometric representation, constraints, and generation of multiple functional views through the product lifecycle. Ontologies aid in this sharing by providing a common vocabulary with shared semantics. Sure et al. [123] introduce OntoEdit—an ontology editor that aids collaborative development of ontologies for the Semantic Web. CAD systems, meanwhile, are moving beyond the representation of purely geometric entities, to integrating knowledge from the design and manufacturing domains into the CAD models as well. Staley and Anderson [124] suggest a 20-point functional specification based on then-current research on CAD databases. The specification includes, but is not restricted to requirements like support for multiple representations, support for iterative design, support for multiple levels of detail, support for engineering transaction processing, and maintenance of data consistency. Turner and Anderson [125] and Anderson and Chang [126] take the idea forward and suggest a feature-based model as a means of capturing and transferring knowledge from CAD to process planning. They propose and implement a generic object-oriented model of a feature, which incorporated a hierarchical vector-based tolerance structure, that in turn is used by their geometric reasoning model to classify and refine features into machinable cavities for automatic process planning. Shah et al. [127], in their review of feature recognition techniques from CAD models, identify the need for a domain-independent representation of feature information, and the need for an embodiment of non-geometric data relevant to a domain or discipline in the feature data. They suggest a construct called the *exemplar*, a pattern of topological, geometric, algebraic, and semantic relationships, as a means of representing data richer than features. The concept of associative feature was introduced by Ma and Tong [128] to bridge the gap between the interfacing of knowledge-oriented tools and CAD applications, to aid intelligent product development. Chen et al. [129] extend this concept and develop a unified feature-based modeling scheme that includes the functional requirements and the concept design into the feature modeling process, and supports geometrical and non-geometrical feature associations, which include facts in the higher-level knowledge model.

Szykman et al. [130,131] developed the NIST design repository—an intelligent knowledge-based design artifact modeling system for capturing, representing, and reuse of knowledge in an organization. Xue and Yang [132] extended the NIST design repository model and introduced a concurrent engineering-oriented design database representation model (CE-DDRM) for supporting aspects of the product lifecycle in concurrent design. The CE-DDRM model describes both geometric and non-geometric information, and also integrates function, behavior, and form into the same computing environment. Kim et al. present a paradigm of ontology-based assembly design in [133]. In this paper they describe a collaborative assembly design framework, supported by an assembly design (AsD) ontology.

In their review of product family design and platform-based product development, Jiao et al. [134] discuss the need to incorporate more front-end issues like customer modeling and integration, marketing and economic issues, as well as back-end issues involving marketing and supply chain. Thevenot et al. [135] suggest a metric called the design for commonality and diversity method, based on functional attributes to help designers decide where to use common components and where to use unique ones. Ouertani et al. [136] propose a standardized approach to trace and share product knowledge and identify key constructs to support traceability during the product development process. They also propose a definition for product knowledge and its traceability. Hoffman et al. [137] argue that geometric modeling of mechanical components and systems is no longer a mere supporting activity, and that interoperability of virtual models is critical to digital, model-based engineering. They propose a query-based approach that supports the transfer of a model from one system to another and addresses the issues of geometric interoperability, and illustrate their approach in a CAD—analysis interoperability example, and state that the approach would overcome barriers in realizing the potential of today's model-based engineering and platform-based engineering systems.

The need brought about by heterogeneous and multidisciplinary engineering design knowledge for higher semantic modeling in engineering design is actively being addressed [138]. In a collaborative design environment, assembly design sharing, consistent interpretation of constraints and requirements, and the evolution of a web of relations form a knowledge base. In addition, Design Theory and Methodology (DTM) research, as Tomiyama et al. summarize in their review of the subject [139], also needs to expand into considerations of product complexity, multi-disciplinarity, integration of domains, and consideration about globalization trends, to name a few. Engineering repositories and product families have come a long way from standard component-sharing to decision support based on function breakdown. The semantic web has a key role to play in product design—existing knowledge models can be connected via the internet and design collaborators can access product models, reuse existing design knowledge, and promote interchange of knowledge across multiple projects and disciplines.

3.2.3. Design rationale encoding and use

In the 1980s the concept of 'design rationale'started gaining importance. Products and their components or features were defined in terms of the way they worked, but not why they were designed in a certain way [140]. This resulted in the collaborating teams needing inordinate effort and communication to understand each other's designs. Design rationale systems were introduced as a basis of reasoning and communication among such teams. Knowledge representations used to capture design rationale fall under two categories—argumentation-based techniques and descriptive techniques. Argumentation-based systems have a structured, graphical format of nodes and edges for connecting design issues and relationships [141], and descriptive approaches record the sequence and the history of activities in the design process [142]. Questions, Options, and Criteria (QOC) [143] is an example of an argumentation-based technique while Issue-Based Information Systems (IBIS) [144,145] is an example of a descriptive technique.

Lee [146] reviews design rationale systems with regards to the services they provide and how they create, manage, and access rationales. He also lists the advantages of such systems as enabling better support for redesign and reuse, collaboration, dependency or constraint management, design maintenance, learning, and documentation. Design rationale systems have also been used in conflict resolution in a collaborative design environment [147]. Capturing design rationale can help extract existing design knowledge from past designs, capture it through systematic dissection and manage the product information in an online design knowledge base. Using this knowledge base, design rules are formulated that can then be reused to build new design concepts [54,148]. Szykman et al. [142] mention design rationale as an important issue in supporting an evolving product knowledge base. They observe that recording knowledge and constructing design rationale has been implemented in design rationale systems both through automatic capture and through user intervention. The latter approach has not been very successful since designers are reluctant to spend time on annotating their designs with rationale.

While automated design rationale capture tools have been implemented [149–152], completely automating design rationale was noted as a significant challenge. Shum et al. [153] acknowledge the 'intrusiveness'of human intervention-based design rationale capture. Subsequent systems like Questmap and Compendium [153], and DRed [154] evolved based on the IBIS system.

Shum et al. [153] acknowledge that the capture of design rationale cannot be completely automated, and training the designer to capture rationale becomes necessary. Bracewell et al. [154] note that a general-purpose open source modeling tool that can be easily customized can be an alternative to an application developed from scratch.

The design rationale system has had a long history in the movement to provide computational support for design, but its inherent structured approach, which is often intrusive to the designer [153] and also not conducive to capturing tacit knowledge, has limited its adoption in the industry. There has been some success, notably the incorporation of a design rationale editor (DRed) into the Rolls–Royce PLM toolset [154]. Design rationale systems can capture knowledge either automatically or by user intervention. However, user intervention-based design rationale capture has met with only limited success, as designers are typically very reluctant to spend time on annotating their designs with rationale [142]. A promising approach is suggested by Sung et al. [155] where individual designer behavior in a CAD environment is logged and interpreted to interpret patterns in sequences of user actions, and use these to offer context-sensitive help to individual users. While structuring of knowledge and capturing of rationale for design is essential, it may require less structured approaches, especially from the point of view of the product designer.

4. Computational support tools in design

The design process is very dynamic—there is continuous interaction between problem definition and solution generation. A design solution goes through multiple iterations before it is optimally or robustly determined. The use of computational support tools in design enable the capture of knowledge generated at each design stage and the representation of this knowledge to provide design decision support. There are a number of support tools for the later stages of design, especially the detailed design stage, while there are relatively fewer support tools in the earlier conceptual stages of the design process. This reduces the efficacy of the designer in manipulating, organizing, representing, and using design data [156]. The lack of support during the conceptualization stage limits the potential of design space exploration as designers are limited by past knowledge, experience and constraints on time and resources.

There is an increasing need to make computer support tools more designer-centric. This requires an understanding of the cognitive working of designers so that the transformation from thought to representation could be made and generated knowledge is effectively captured [4]. Knowledge representation techniques range from computer-centric techniques, which ensure that the computational implementation is carried out efficiently, to human-centric techniques, which aid the creativity of the designer. These representations could range from formal specification methods like language or geometric models to visual methods like images [157].

The evolution of support tools for engineering design is supported by collaboration, artificial intelligence and developments in information technology [119]. Work in the 80s on computational support included ways to integrate codified design knowledge from different systems, like Chalfan's symbolic computing approach [158] to codify the domain knowledge related to the product's design that provided a means to loosely integrate different programs and create a generic design analysis tool. A lot of work was done in the 1990s on parametric modeling using CAD systems and automatic uncoupling and storage of geometric constraints [159,160]. Geometric modelers, sketchers and constraint solvers became the key components of a parametric computer modeling system [161]. Though artificial intelligence was commercialized in early 1980s, it found an application in computer aided conceptual design only by mid-1990, when "Conceptual design became a process based on search, logics, grammars, planning, learning, case-based reasoning, qualitative reasoning, first principles, constraint propagation, collaboration, emergence, analogy, evolutionary systems, and neural networks" [162].

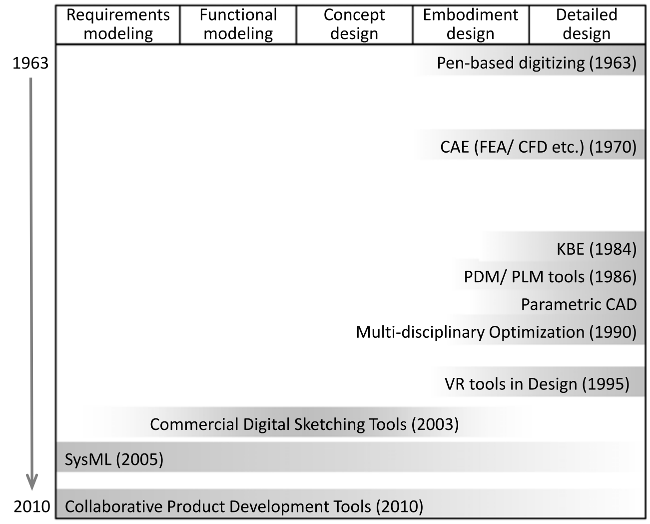


Fig. 7. First occurrences of computer-based support for product design. The degree of shading indicates the tool's prevalence at the corresponding design stage. Examples are representative of tools first developed in their category, like Sutherland's Sketchpad for CAD [163], the NASTRAN solver for CAE [164], Pro/E feature-based modeling application for CAD [165], the LS-CLASS Multi-disciplinary optimization program [166], the ICAD programming tool for Knowledge-Based Engineering [167], PDM and PLM specifications and software [168], Virtual Reality tools [169,170], and more recently, Digital sketching tools [171], SysML for systems engineering [172], and Vuuch for collaborative product development [173]. The list is not comprehensive, but indicate the development trend in computer support tools.

Similar to the preceding section, this section on computational tools discusses computational tools for product design with both a process view as well as a systems view. The process view of computational tools will try and establish the ways in which the designer is supported in his/her process of ideation, concept generation and knowledge capture and retrieval, while the systems view will look at tools that aid the designer in a collaborative environment, and in the design of a product as a system.

4.1. Process view of computer support tools

The application of computer support tools in early design had been quite difficult as the knowledge about design requirements and constraints available during conceptual design phase is often imprecise, approximate or incomplete [119]. Fig. 7 shows the development of various computational support tools over the years and their application in the design process.

We can see in the figure a gradual spread of computational support tools towards the earlier stages in the design process. There have been early and subsequently significant development in tools like CAE, parametric CAD modeling, optimization—tools that are used in the later stages of design. Most of these tools have found industrial application. Some tools like PLM, languages like SysML, and knowledge management tools, although relatively recent, have also found industrial application. Tools that are applied to the early and unstructured design problems, like gesture-based sketching, are still in their nascent stages in the industry. In recent times, research is actively being done to overcome these problems and provide support upstream in the design process. CAD software

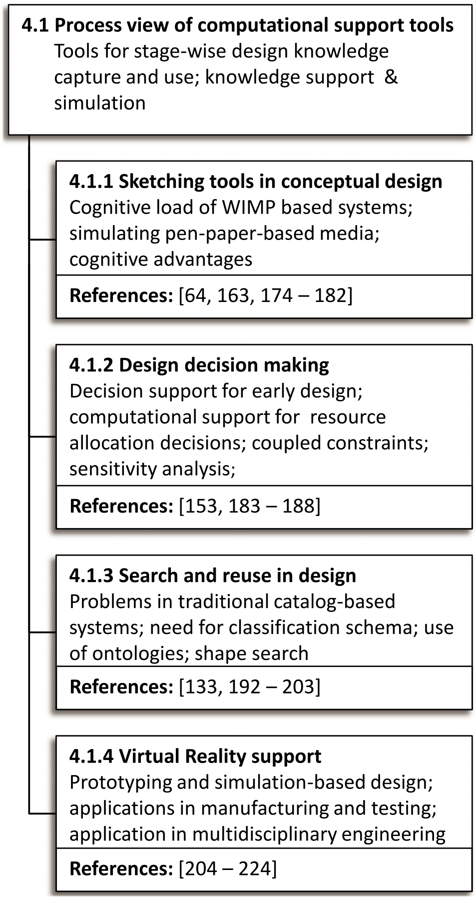


Fig. 8. A process-oriented overview of computer support tools.

is being continuously modified to incorporate support for concept development and configuration activities. A wide span of issues are being explored, like sketching of shapes, conceptual design, requirement engineering, decision making, design search and reuse, all of which cover some computational and design issues of interest in conceptual design [54]. Fig. 8 gives an outline of this section, and an overview of the topics discussed.

4.1.1. Sketching tools in conceptual design

Sketching is one of the most important activities in the design and development of new products [174]. It is the main design tool used for archiving the geometric form of design and communicating ideas between people. Traditional CAD systems use WIMP (Windows, Icon, Menu, Pointing) style-based Graphical User Interfaces (GUIs), which require the designer to be trained in a particular application's menu-based system. The added cognitive load often is a detriment to the designer. Designers thus continue to use the convenient, portable, and cheap pen–paper medium for rapid construction and evaluation of the new ideas. Hence, to be a useful aid for conceptual design, sketching on CAD platforms will need to be developed [64].

Since Sutherland's Sketchpad [163], numerous works have been investigated in drawing diagrams on 2D and 3D free-hand sketch. Stahovich et al. [175] demonstrated a system that could interpret the causal functionality of a two-dimensional mechanism depicted in a sketch, and generate alternative designs. A gesture- based system for interactively constructing 3D rectilinear models is developed by Zeleznik et al. [176], who develop a prototype system "SKETCH-N-MAKE" for the design and manufacturing of parts; Cohen et al. [177] proposed a constructive approach to modeling free-form shapes using Skin algorithm. Qin et al. [178] established an on-line sketching system to 2D and 3D geometry based on fuzzy knowledge. Cherlin et al. [179] presented a sketch-based system for the interactive modeling of a variety of free-form 3D objects using just a few strokes. Yang et al. [180] presented a sketch-based modeling of parameterized objects. For the systems which also explore the physical analysis of 2D and 3D shapes, Murugappan and Ramani [181] develop a tool (FEAsy) to help users quickly transform, simulate and analyze the finite models through 2D freehand sketch. An iterative, Tablet-PC-based design system is presented by Tian et al. [174] for a fast kinematic simulation and finite-element-based static analysis.

The importance of pictorial representation is evident in Fig. 4, especially for the earlier stages of design. Sketching is one of the most intuitive ways designers communicate, especially in the early stages of design. Designers use informal sketching as a principal medium of external thinking—they provide an external memory for images in the designer's mind, they allow for easier manipulation of ideas, and they allow the information to be represented in various forms [182,64].

4.1.2. Design decision making

As mentioned earlier, research focus has shifted upstream in the design process to support early design decisions and to handle design rationale. During the design process, the designer repeatedly seeks guidance [4]. The designer needs to decide whether he or she should develop more evaluation information, or generate new potential solutions, refine feature criterion and targets, negotiate changes in criterion features and targets, decompose the issue into sub issues, or reach conclusions and document results. Recent research concentrates on support for design decisions, for traditional decision processes involving materials and resource allocations, as well as more recent developments in the form of system durability—for instance, a methodology for decision support in the selection of product design projects [183]; using RQFD (Resource QFD) to support resource allocation decisions and risk management in product development [184]; a decision support system that uses utility theory to provide a mathematically rigorous means for capturing designer preferences, along with a decision support problem construct to aid engineering judgement for decision making [185], etc. Design for manufacturability (DFM) is another area where the decision support has found application. Xiao et al. [186] introduce a Collaborative Multidsciplinary Decision-making Methodology (CMDM) with a game-theoretic approach to aid decision making, using a leader/follower protocol to separate design from manufacturing teams and resolve interactions between the teams' decisions. Molcho et al. [187] introduce a Computer-Aided Manufacturability Analysis (CAMA) tool that provides necessary information and insight to the designer about the manufacturability of a proposed design.

Currently, not many commercial tools exist to support design decisions at the conceptual design phase. Designers often use a familiar solution that has worked well in the past, instead of exploring new alternatives [188]. Also at times, while designing a complex system with coupled constraints, the designer might not be aware of all the relationships between the various variables. Hence, he/she may fail to estimate the effect of the changes in one part of the design on the other. An estimation of the sensitivities of all variables in the conceptual phase is hence very helpful. This reiterates the need for development of early design support systems so that designer can make robust design decisions during conceptual stage and avoid time/monetary wastage in making poor decisions. Interactions between the designer and the computational synthesis tool aid the designer's decision-making during concept evaluation so that it can be analyzed, modeled, and later used for faster search of larger design spaces [154].

4.1.3. Computational support for search and reuse in design

Engineers spend a significant amount of time searching for the right product information during the design process. This is compounded by the fact that product information exists in varying forms at different stages of the design process. More than 75% of the design activity comprises reuse of previously existing knowledge [189]. Considering that engineers spend about 60% of their time searching for the right information [30], this translates to considerable inefficiencies for an organization. There has been considerable research in search and retrieval as well as in the indexing and storage of information in a relevant manner.

There exists a gap between the high-level, conceptual mental model of a product needed by a customer and the low-level, physical query that searches and retrieves the needed information from a library. The key issues of old catalog services such as IndustryNet® [190] and Alta Vista™ [191] include recognizability, usability, efficiency, and manageability of various design information. The emergence of Active Catalog system establishes a set of integrated ontologies and a semantic network which improve the correctness (hit ratio) and the completeness (coverage) of both the search and application [133] in the context of heterogeneous, internet-based distributed computing environments.

Li et al. [192], in their review of product information retrieval, classify indexing and retrieval systems as shape-based, knowledge-based, and ontology-based systems. McMachon et al. [193] identify engineers' needs for support in the management of text documents, especially in the early stages of design, the need for a unified system for the storage, classification, and retrieval of these documents. They develop an information system with a flexible architecture called Waypoint that employs a user-predefined classification schema, to address these issues.

Engineers have always depended on referring documents for information to execute specific tasks in product design. While the digitization of data has made this access easier, engineers rely mostly on keyword searches to obtain this information [193]. Through electronic document management (EDM) and product data management (PDM) systems, there is some support for the knowledge retrieval in organizations. However, design information is largely contextual and such systems provide limited support, and often retrieve information that is not relevant [194]. Ontology-based systems offer effective indexing mechanisms that provide ways to structure documents which are typically maintained unstructured in organizations [195,196,96]. Kim and Kim [197] introduce a "causal knowledge" model, which represents design processes in the form of a network of vertices and edges, with the vertices representing design/ process alternatives, with context-sensitive probabilities assigned to the vertices. They compare it to procedural knowledge from the perspectives of knowledge expression ability, decision alternative representation ability, reasoning capability, and knowledge cultivation ability, and argue that causal knowledge is superior for knowledge representation and reuse.

There is an increasing interest for the use of 3D content due to the continuous development of multimedia technologies, virtual worlds and augmented reality [198]. The proliferation of 3D models on the Internet and in-house databases has led to the development of technology for effective content-based search and retrieval of 3D models [199]. This has led to significant research and development in shape similarity detection, multi-level representation for shape matching and retrieval, feature extraction, model decomposition and segmentation [200]. However, a problem with shape search techniques is that there is not any particular search technique that fits all applications, as each distinct problem requires some customization according to its domain. An engineering shape benchmark (ESB) is developed by Jayanti et al. [201] with the motivation of determining whether various shape representations have enough shape content in them to discriminate between different forms that exist in the engineering domain. Alizon et al. [202] develop a 'Reuse Existing Unit for Shape and Efficiency'(R.E.U.S.E) method for knowledge reuse of manufacturing information in design through three stages: similarity study, efficiency assessment, and configuration. Some of this research has resulted in commercial tools for search in the manufacturing supply chain, such as Vizseek™ [203], which is also moving towards creating a "social network" of members with specific manufacturing capabilities in order to foster better collaboration.

4.1.4. Virtual reality support in design

Virtual reality (VR) tools are primarily applied downstream in the design process. The simulation of environments or processes that would occur in the manufacture, assembly, or use of a product is critical in providing information that would help support decisions earlier in the design process. One of the strengths of virtual reality lies in applications that have complex relationships among product data that need human interpretation and decision-making applications that enable exploration of patterns or trends in abstract high-dimensional data [204].

In product design, Virtual prototyping and simulation-based design are two important applications of VR [205]. In simulation-based design, Jayaram et al. [206,207] and Wang et al. [208] developed a virtual assembly design environment (VADE) to aid designers in assembly planning and verification, maintenance verification, and design for assembly. Gill and Ruddle [209] use Jack, an ergonomics and human factors product by Siemens PLM, to evaluate manufacturing ergonomics. Ryken and Vance [210] propose the implementation of VR in stress analysis, and use a surround screen virtual environment to display and interact with the geometry. Jayaram et al. [211] use VADE with Jack to suggest ways to implement a quantitative ergonomic analysis in virtual environments. Bernard and Hasan [212] introduced the working situation model for safety evaluation during the design process of systems. This model has been the base of an information system used for configuring a VR environment including a Knowledge-Based Engineering (KBE) system. This environment is dedicated to risk evaluation during the design phase of systems, by considering different configurations of the system [213,214]. Dukic et al. [215] perform a verification of visual demands in car assembly work using virtual tools, while Dorozhkin et al. [216] use VR as an immersive environment to perform concurrent simulation of a tractor assembly line with the capability to modify simulation parameters while immersed in the virtual environment. A virtual environment for disassembly to produce necessary information for the identification of the disassembly path is proposed by Cappelli et al. [217].

VR has been developed as an internet tool, as well. Kan et al. [205] develop an internet-based VR-based collaborative environment called VRCE to aid collaborative design for low-cost products developed by small to medium size organizations. Ottosson and Holdmadahl [218] combine a content management system and transforming VR files, so that the VR application could be used as an ordinary web application.

Virtual prototyping is the process of evaluating the product performance, and simulating the product, its user, and their interaction [219]. Several conceptual design virtual prototype systems and environments such as JCAD-VR [220], Active World [221] and Design World [222] have been developed. VR environments have also been used in design synthesis, especially in difficult-to-visualize scenarios like spherical four-bar mechanism synthesis [223].

A real-time collaborative 3D virtual environment for multidisciplinary design is another growing research area where the scope of a problem is determined by exploring a range of alternative solutions to a brief or set of requirements in the conceptual design phase. Relevant challenges in this area are different decomposition schema of the model among the collaborators; relationships within and across the different schema; multiple representations and versioning of elements; ownership and access to elements and properties of elements and shared visual representation in a 3D virtual world [224].

4.2. *Integrated view of computational support for design*

Developments in information technology and the web created fundamental shifts in the way engineering was done by enabling the use of many new tools for human interaction and collaboration [225]. Dynamic business needs coupled with globalization required the product development teams (like design, manufacturing, sales and management) to do real-time sharing product data in a concurrent manner [226,227]. The existence of a product data management (PDM) tool became pivotal to the rapid launch of new products in the market. Network Centric CAD was developed, where the design process was driven by the availability of design information, such as component databases, CAD models and simulation results, all accessible through a computer network [228]. However, the implementation of industrial product data exchange systems has been slow owing to the fact that there are diverse data formats and standards, giving rise to conversion problems between different CAD/CAE applications. This section discusses, as indicated in Fig. 9, the challenges faced in collaborative and multidisciplinary design environments and the development of computational support tools to address these challenges.

4.2.1. *Collaborative design*

With the increase in globalization, the amount of information about the product that needs to be managed and shared across various development platforms is very large. Research is actively carried out for developing methodologies and technologies of collaborative computer-aided design systems to support design teams that are geographically dispersed [229,230]. Collaborative design can heavily reduce introduction time to market and redesign cost for the new product development. Rapid advances in information technology have provided the platform for distributed CAD systems to support collaborative design, which allows the users to remotely view and augment a product model. Design and product history, such as previous modification processes and feature information of the product is also available to all teams [231]. Updates to the product model are displayed instantly to the users after any design modifications have been made to the CAD model. Li et al. [230] review the methods and technologies for collaborative computer-aided designs, and state the importance of advancements in 3D streaming technology, which will enable collaborative, dynamic review of components over the web.

The importance of collaborative design is steadily increasing, and with it computer technology must not only increase the capabilities of the individual specialists, but must also enhance the ability of collaborators to interact with each other and with computational resources [119]. A number of emerging technologies including distributed agents and Web technology have been proposed to implement collaborative design systems. The Web is used as a medium to share data, information and knowledge for product data management [232] and supervision of design process and working system by the design team members [233]. Huang et al. [234] proposed the framework of Web-based collaborative design system and developed Web-based DFX tools. Rodgers et al. [235] developed a WEbCADET system for distributed design support.

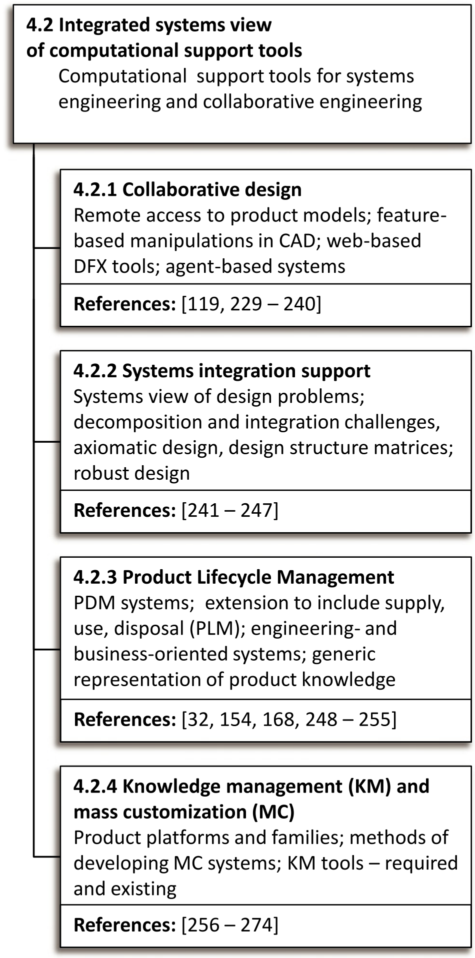


Fig. 9. Integrated view of computational support for design.

The Distributed and Integrated Collaborative Engineering Design (DICE) project [118], started in 1986 at MIT, made several novel contributions to collaborative design, with the introduction of synchronous and asynchronous communication tools, integration of qualitative and quantitative geometric reasoning, knowledge-based design, use of asynchronous teams for solving both symbolic and numeric constraints, design rationale capture, and collaborative negotiation. Agent-based design systems use a communicative and intelligent coupled network of problem solvers. Similarly, research systems such as PACT [236], DIDE [237], A-Design [238] and workflow approaches [239,240] were developed. A key requirement of these tools will be the representation of knowledge in an appropriate form, based on the designers' needs, the context in which it is being used, and the nature of the information being conveyed.

4.2.2. *Systems integration support*

It is a common practice in product design to break a problem into chunks of smaller problems, in order to simplify the overall problem, and to address these chunks of problems concurrently, reducing the lead time. The systems engineering view of the product is an example of such an approach, and so is problem decomposition, frequently used in breaking down a large optimization problem into smaller, more manageable pieces. The main challenges in this practice are the decomposition itself, as it is often difficult to identify suitable sub-problems, and the integration of the solutions of the individual problems into the overall system [241]. Eppinger et al. [242] suggest the use of the Design Structure Matrix (DSM) for managing complex task-based design processes as well as complex parameter-based component designs. Browning [243] reviews the application of DSMs in four areas: Component-based system architectures, people or team-based organizations, activity-based processes, and parameter-based design relationships, and concludes that DSMs "facilitate intelligent system decomposition and integration analysis".

Suh [244] introduces the concept of axiomatic design as a way to reduce complexity, and introduces two axioms—the 'independence axiom' which requires that "the functional independence be satisfied through the development of an uncoupled or decoupled design" , and the 'information axiom' which states that "the design that has the least information content is the best design". He uses these axioms, along with the Functional Requirements (FR) and Design Parameters (DP), to create a case for reusability of FR/DP relationships to form a knowledge base that can be used to develop a controllable and robust system.

Recent research on system integration has been in the area of improving the integrated system robustness, by exploring integration from a system configuration viewpoint to minimize system-to-system interactions and overall system sensitivity to noise factors [245]. Research in Robust Design also aims to solve integration issues as it involves the development of methods intended to make a product's function more consistent in the face of variations in downstream processes, environments, and customer use patterns [246], like a Taguchi method-based model to minimize the impact of design parameter changes on implementation processes by considering possible design parameter changes [247].

4.2.3. Product Lifecycle Management (PLM)

PDM systems were developed to manage heterogeneous product-related data and provide access of relevant files to designers, engineers, manufacturing personnel, management, and other groups that need to work together in a manufacturing enterprise [168]. They were used to keep track of various product data and the processes related to product development. PDM systems managed documents such as reports, CAD data, drawings, analysis data through the design process. However, there were several shortcomings of PDM systems, such as the lack of formal product representation in the form of function, behavior and structure, lack of reuse of design knowledge, lack of impact analysis, etc., [248]. The concept of PLM appeared in the 1990s as an extension of PDM. PLM is defined as the process of managing a company's products from their conception, design, manufacturing, all the way to its use and disposal [249]. It is an integrated approach including a consistent set of methods, models and IT tools for managing production information, engineering processes and applications along the different phases of the product lifecycle [250]. Anderson [251] defines the need to examine the difference in strategic variables—in our case product, processes, and resources—between stages of the product lifecycle, as well as differences among the determinants of high performance across stages of the product lifecycle. PLM has been used by organizations as a means to streamline their processes and to manage their product knowledge. While PDM focused on the management of product data chiefly with respect to design and manufacturing, PLM extended the process to cover other phases in the product lifecycle, like supply, use, and disposal. Another main difference between PDM and PLM is that while PDM concentrated on data management, PLM supports the management of knowledge through the product lifecycle [252].

PLM software solutions can be separated into two main categories: Engineering oriented (EO) and Business oriented (BO) [253]. Usually, the EO PLM solutions are more concerned with the technical issues related to the product; they deal solely with a physical product and cannot manage a service. The original intent here is to integrate CAD, CAE, and computer-aided manufacturing (CAM). To support collaboration needs, it was necessary to provide solutions for document generation, drawing sharing, and more recently, knowledge management.

Recent research is focusing on developing a generic representation of product knowledge that includes other kinds of product knowledge beyond form, function, and behavior, in order to support a broader level of information exchange and interoperability [32,254]. Product knowledge is represented in terms of requirements, specifications, artifacts, form, functions, behaviors, design rationale, constraints and relationships. Such a representation supports multiple levels of abstraction, which provides computational support for early design activities [255]. Other tools are slowly being integrated into the PLM system. For example, Bracewell et al. [154] develop a design rationale editor (DRed) and successfully integrate it into the Rolls–Royce PLM toolset. We are likely to see a similar merging in areas of shared interests between supply chain and PLM software tools in the future.

4.2.4. Knowledge management and mass customization

Mass customization (MC) is the process of designing and manufacturing products customized to meet the user's needs, but at speeds and costs equivalent to mass production. Product families or the concept of mass customization have been recognized as necessary to cater to a variety in customer needs, while keeping manufacturing costs low [256]. In achieving mass customization, product platforms have been widely considered a critical factor. Meyer and Utterback discuss the idea of product families as early as 1992 [257]. Meyer and Lehnerd [258] describe the product platform as a set of subsystems and interfaces that form a common structure from which a stream of derivative products can be efficiently developed and produced. Robertson and Ulrich [259] define it as a collection of assets like components, processes, knowledge, people, and relationships, shared by a set of products.

Commonality or standardization and sharing of components, modularity and application in configurations, scalability or sorting changeable product parameters, and postponement of variation as far as possible in the manufacturing system are the strategies that are key to a product platform application [260]. In addition, an effective product platform should lend itself to future product expansions. Different methods have been proposed to design product platforms that allow for such expansions, based on utility-based compromise decision support problems [261], or software implementation using mechanical buses [262]. The use of mathematical models for optimal configurations has also been explored, for instance Huang et al. [263] utilize mathematical models for tradeoffs between robustness and customization; Siddique and Rosen [264] suggest a design space modeling methodology to ease the exploration of large combinatorial design spaces that are inevitable in designing configurations in product families.

From a product platform variant, products can be derived and then formed into a product family. A challenge while designing an appropriate product family has been in mapping product variety to the right market segment. A state of the art on this mapping, i.e., product family positioning and on product family design is presented by Jiao et al. [134]. Hernandez et al. [265] propose a six-step method for mass customized products, combining multiple approaches. The steps are as follows: (1) define the space of customization, i.e., the set of all combinations of values of product specifications that are to be satisfied; (2) formulate an objective function; (3) identify modes for managing product variety like modular combinations, dimensional customization, customization in configurations etc. in order to customize the product; (4) determine the number of hierarchy levels and decide the ways to handle product variety in these levels; (5) formulate a multi-stage optimization problem; and (6) solve the problem.

The method has several advantages, like cost-effectiveness and versatility, but it requires the formulation of an objective function that captures the various costs involved in a design. Thus, different mass customization strategies need different information systems to support them, and the integration of these information systems within the entire supply chain plays an essential role in the successful implementation of mass customization [266]. Frutos and Borenstein [267] list necessary functionalities for such an information system as

1. Facilitation of collaborative product development,
2. Eliciting knowledge about clients,
3. Virtual enterprise environment
4. Providing enrichments to clients, and
5. Providing an open system architecture.

For the success of an MC system, a main factor is Knowledge Management, since it allows collaboration between customers and vendors [267,268]. There are various definitions of Knowledge Management (KM) [269]; a simple way of looking at it is as a business process that identifies knowledge from previous experiences and selectively applies it to current decision-making processes. Effective application of KM in an organization leads to cost reduction in design, production, and distribution [12]. KM tools were defined by Ruggles [270] as technologies for knowledge generation, codification, and transfer.

The context of KM implementation is often complex and highly dynamic, and various types of tools have been developed for various applications. They are listed as follows:

- General tools and methods serving for formal and informal knowledge sharing, like yellow pages, knowledge maps, wikis, etc.
- KM tools for knowledge capitalizations, like ACQUIRE® and XpertRule®, which help capture expertise for use in expert systems
- KM tools for knowledge sharing, like Centric Insight®, which allows people to find and share knowledge in a secure environment, and Open Text Corporation, which provides a KM tool to enhance web-based knowledge sharing capabilities.
- KM tools for knowledge retrieval, like Open Text Discovery Server, a search engine for enterprise information retrieval, and a knowledge-based repository developed by Wong et al. (2006) that can perform search and analysis, and then provide users with summarized results
- KM tools for query, which enable quick, efficient responses to each potential query, can provide exact knowledge to user requirements. The Open Text Federated Query Server, and AQUA, and automated real-time question answering system, are examples of such tools.

There are several effective KM tools available, and while selecting the most appropriate tool is not easy, there exist criteria like the ones provided by Tiwana and Ramesh [271] which help the selection process. There are, however, several barriers that lead to non-successful implementation of KM, like knowledge confidentiality, lack of adequate training in KM use, language, cost, technology levels and so on [272]. In developing effective MC systems, it is thus crucial to also implement methods and tools for effective knowledge management for the product lifecycle [273].

Support tools for design thus encompass a wide range of requirements, disciplines, and applications. Ullman [4] and Rezayat [274] give many requirements for an ideal engineering design support system. Some key aspects of such a system include:

- Intelligent design and engineering using knowledge-based systems, software agents, web-based standards, key characteristics and features, and practical human/electronic protocols
- Integration of requirements and constraints into the development of parts and assemblies
- Integration of material, manufacturing and cost to management of product lifecycle
- Design knowledge capture and dissemination through web-based and non-web based applications
- Use of multimedia/virtual-reality/net-conferencing for collaborative design and engineering
- Product integration and management through network-centric CAD and global access to information
- Innovative methods for performing engineering analysis and simulation for design optimization as well as for product definition in the conceptual design stage
- Freehand sketching for CAD
- Guidance to the designer for decision making during the conceptual design phase.

There may not be a single design support tools that addresses, or perhaps even needs to address all these requirements. However, it would be useful for anyone developing support tools for design, to keep these requirements in mind.

4.3. Computational support tools—an industry view

While it is educational to study the directions research has taken in developing computational support tools for design, it is useful to also take a look at the commercial side of design. Early developments in computational support tools in the industry involved the development of detailing and solid modeling applications. This work was mostly supported by the aviation and the automotive industry, and in some cases the U.S. government. These were initially available for internal use. For instance, Lockheed supported research and development of the Computer Augmented Design And Manufacturing (CADAM) system [275]; GE and Calma developed a Design, Drafting, Manufacturing (DDM) system [276]; Structural Dynamics Research Corporation (SDRC) developed the Integrated Design and Engineering Analysis Software (I-DEAS) [277]; McDonnell Douglas developed the Unigraphics application; and Dassault Systemes developed the Computer Aided Three-dimensional Interactive Application (CATIA)[275]. The automobile industry followed suit —Volkswagen developed their VWSURF system [278], Ford developed the Product Design Graphics System (PDGS)[279], and Renault integrated EUCLID to develop its CAD system [278]. CAD systems of today have evolved to an extent that they no longer are restricted to a geometric rendering of the design, but are capable of supporting richer and more diverse data associated with the product. These systems also come with advanced simulation and lifecycle management applications and have integrated many aspects of product design into more monolithic tools.

A timeline of significant acquisitions and the resulting product launches or integrations are shown for four organizations that provide CAD, CAE, and PLM solutions: Autodesk, Dassault Systèmes, Parametric Technology Corporation (PTC), and Siemens PLM Software in Fig. 10. These four organizations garner the highest market share in the CAD industry [280]. It is interesting to note the parallels in their timelines: the gradual expansion from solid modeling and surfacing software, to the integration of other design modules like CAE and PLM/PDM have occurred in all four organizations. Most of these are through various acquisitions and partnerships with other organizations that specialize in those specific design modules. During the last two decades, two critical milestones that influenced the CAD industry are:

- Samuel Geisberg's concept of feature-based parametric modeling, which uses parameters, dimensions, features, and relationships to capture intended product behavior and create a recipe which enables design automation and the optimization of design and product development processes, and
- The development and extension of PLM solutions.

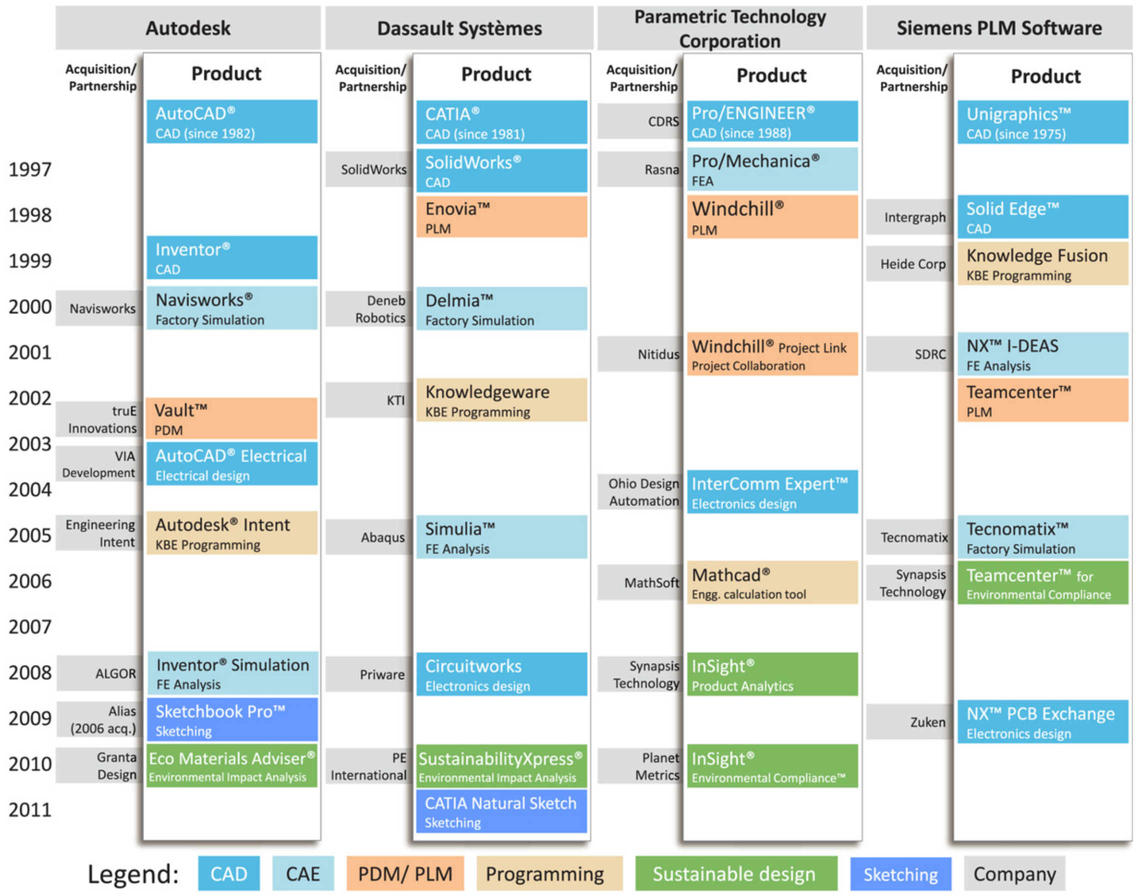


Fig. 10. Integration of support tools for design in the industry—a view of acquisitions and integration of products among four major organizations that provide software support for the product lifecycle. References: Autodesk archives [171,281], PTC History and acquisitions: [165]; Siemens PLM Timeline: [282,277]; Dassault Systèmes history: [283,275]. The list of acquisitions is not comprehensive, but represents the industry trend.

A broad range of catalogs in PLM such as visualization, collaboration, document management, process planning, factory layout and quality management are now gradually employed by SIEMENS PLM Software's Teamcenter®, Dassault's Enovia™, PTC's Windchill®, and a PDM application in the case of Autodesk® Vault. KBE was also incorporated into CAD tools in the early 2000s. For example, the Intent!™ KBE language from Heide Corporation (later known as Engineering Intent) was integrated into NX™ as the Knowledge Fusion tool, and into Autodesk® in the form of the Intent KBE application. KnowledgeWare and Component Application Architecture (CAA) were equivalent applications in CATIA®, and Mathcad® was the corresponding PTC application that incorporated programming and knowledge modeling tools in CAD, chiefly to support generative design, configuration design, customization, and optimization from within the CAD application. Such tools have found applications in Computer-Aided Process Planning (CAPP), like the USIQUICK project [14] developed in CATIA CAA by Harik et al. [284], and an integrated CAD/CAPP system implemented on Solid Edge® by Zhou et al. [285]. In contrast, some organizations (not indicated in the figure) have focused on the need to create a design or modeling framework that integrates the data and analyses performed between different modeling and simulation applications. Modeling framework software like Adaptive Modeling Language (AML) developed by Technosoft, ModelCenter developed by Phoenix integration, and iSight™ developed by Engineous Software (now part of Simulia) have concentrated on different ways of integrating the modeling and analysis capabilities available commercially [286]. The acquisition of Engineous by Dassault serves to underline the need for and importance of such modeling frameworks.

With recent environmental regulations, modules like Solid Works® SustainabilityXpress and Autodesk® Eco Materials Adviser give users access to measure, document, and report carbon footprint, greenhouse gas, hazardous materials, and other environmental concerns. PTC's acquisition of Synapsis Technology and Planet Metric has resulted in the incorporation environmental compliance in their InSight® product analytics software. This enables users, manufacturers and retailers to model, analyze, and optimize environmental performance, cost, and reliability throughout the entire value chain, from concept to end-of-life, in the near future. An earlier partnership between Siemens PLM and Synapsis Technology initiated the incorporation of environmental compliance into Teamcenter.

Sketching support as a tool for conceptual design is a recent development and is seen only in two organizations: Autodesk®, with its Sketchbook® Pro, and Dassault, with its CATIA Natural Sketch, integrate gesture-based 2D and 3D sketching into the design process, more specifically for Industrial designers.

The current industry trend, as can be seen from the figure, is tending towards integration of various applications into one monolithic system. While this may ostensibly indicate better interoperability between various applications and thus various stages of the design process, the fact remains that there are multiple design vendors who provide services to multiple OEMs or Tier-I suppliers, all of whom may not use the same monolithic systems. Thus the problem of collaborative design and knowledge reuse is not solved, but merely percolates lower down the supplier layers.

5. Discussion and future trends

In the previous sections we identified the role of knowledge in various design stages and across design disciplines. We also reviewed a number of existing systems that encode a variety of knowledge representation schemes. With the advent of the Internet we will continue developing software frameworks for distributed design that will improve the ability to represent, capture and reuse design knowledge, and will enable design integration across time and space, and disciplinary or corporate boundaries in a seamless manner. To achieve such an environment the following areas of research in knowledge-based systems were envisioned in a previous review paper [142].

1. Development of a comprehensive representation for product development knowledge
2. Integration of traditional engineering software with knowledge-based applications
3. Mechanisms for indexing, searching, and retrieving design cases
4. Design rationale capture and conflict mitigation
5. Need for commercial CAD/CAM/CAE vendors to support knowledge design knowledge capture and reuse.

It has been a decade since that paper was published. Although much progress has been made as outlined in the current paper, we are yet to see widespread use of knowledge-aided systems in the industry. Nearly two decades ago Alan Mullaly, Chief Engineer for the Boeing 777 aircraft, had this to say about AI's role in mechanical design: "Computers don't design airplanes. We have not put the knowledge that is in the airplane designer's head into AI (*artificial intelligence*) that balances all these objectives. But, someday we will continue to probably move to that. Right now the knowledge to design airplanes is in the designer's head". Although AI has made some inroads into design automation, primarily in VLSI design, Mullaly's observations hold even today. There are several reasons for this:

1. *Knowledge management problems.* Many organizations have problems with knowledge management as described in [287–289]
2. *The knowledge-acquisition bottleneck.* This is due to inadequate tools for knowledge-acquisition in an intuitive manner. There is also a need for development of machine learning tools for automated knowledge acquisition.
3. *Vendor resistance.* There is reluctance among CAD tool vendors to fully support KAD (knowledge-aided design) for fear of losing market share for traditional CAD tools. As noted in the previous section some commercial vendors are making progress in providing knowledge encoding facilities.
4. *Lack of design knowledge-bases.* We do not have appropriate standards nor do we have widely accessible design knowledge repositories.

Trends for the future

The current convergence of technologies in computing and communication systems is reducing the technical barrier to innovation by increasing the ease of access to information and providing frameworks for real and virtual product development. We are poised on the brink of a fundamental shift in the way products are realized and the way knowledge is assimilated and disseminated. Some of the driving factors that are already bringing about this change, or are likely to do so in the near future, are listed below:

1. *The rise of the wikis.* The development of the wiki in 1995 as a system for writing documents collectively, and its emergence in the form of the Wikipedia project in 2001 introduced a way of collaborative sharing that was fine-grained and complex [290]. A wiki is a database of interactive web pages that allows members of a user group to collectively edit the same material from any computer with an Internet connection. While several mediums for communication during design have been discussed in earlier sections, the wiki is unique in that "it closely emulates a real verbal discussion with the added feature of being persistent" [291]. Studies of wikis in the light of collaborative design show that wikis can support the activity, although the state of use and technology inhibit more efficient use [292]. The use of wikis as a tool in project-based design courses has yielded positive results [293–296]. Wikis provide a flexible and self-organizing platform that is especially useful from the point of view of early design, when the information and knowledge is unstructured, and from the point of view of collaborative design, where all communication is persistently recorded and loosely organized through user-defined tags.
2. *Problem solvers.* We are likely to see development of intelligent systems that encode design knowledge and aid students in answering text book questions, in a similar manner to Project Halo [297] for biology. This may involve the development of large engineering knowledge-bases, first proposed in [298]. Computational search engines such as Wolfram-Alpha [299] and enterprise search frameworks like Autonomy IDOL™ [300] can be integrated into such knowledge-base frameworks. The MEml (Mechanical Engineering modeling language) effort to capture mechanical engineering knowledge at various levels of detail may be a step in achieving this goal. Such knowledge-bases will also aid in the advancement of machine learning systems for design.
3. *Bioinspired knowledge for design.* The abstraction of biological concepts for inspiration and use by engineers is a challenge that is being addressed by researchers in biologically-inspired design. Bio-inspired designs can be classified under the heads of 'conceptual', where the result of the inspiration is an artifact, or 'computational', where the result is a process [301]. Both areas face the challenge of the identification of relevant biological phenomena, the abstraction of concepts to a level that can be understood by engineers without a background in biology, enabling non-obvious applications of the phenomena, and avoiding misinterpretations of the underlying biological phenomena [90]. Current research approaches to meet these challenges are from the areas of Natural Language Processing (NLP), Functional Basis [41], and Ontologies.
4. *Ontologies and semantic interoperability.* Section 3.1.4 discussed the role of ontologies for design support systems. These ontologies are required for both encoding design knowledge and for facilitating semantic interoperability. Development of engineering ontologies in the large scale can evolve in a similar manner to the compilation of the Oxford Dictionary. Researchers (across the globe) could undertake ontology development in selected areas and then contribute to a global repository. This would require the establishment of appropriate standards for encoding ontologies. Systematic methods for evaluating these ontologies should be developed. We recommend the establishment of a Global Center for Engineering Ontologies, similar to the National Center for Biomedical Ontologies [302] in the U.S.
5. *Mass collaborative product development.* The paradigm of mass-collaborative product development (MCPD) gained popularity in the software domain with the creation of software like Linux and Apache. This model is now gaining popularity in the physical product and services domain in two forms:

Crowdsourcing and mass collaboration. Crowdsourcing [303] is where design and development is carried out in response to an open challenge with a reward, like InnoCentive [304], Quirky [305], Local Motors [306] and Darpa's Vehicleforge program [307]. In mass collaboration, a product emerges as a result of people with similar interests working together on an idea. Examples include the Arduino™ Controller [308] and the Open Source Car (OScar) [309]. Le and Panchal [310] suggest that products with a modular architecture are more suitable for the largely decentralized development and decision-making that is prevalent in the MCPD model. One can foresee a merging of the MCPD model into product families, discussed earlier in Section 3.2.2, and representation of physical modules with a sufficient degree of abstraction analogous to the way classes and modules are represented in open-source architecture could gain importance. Business models of the near future would do well to design towards sustainable platforms like shapeways [311]

6. *Natural user interfaces.* Reality-based systems facilitate intuitive human–computer interaction with little user training or instruction [312]. This is evident in the recent upsurge in touch-based personal computing devices like smartphones and tablet computers, and in gesture-based controls in gaming, like the Nintendo Wii™ and the Microsoft Kinect™. Gesture-based interactions in VR systems have already been discussed in Section 4.1.4, and studies of natural gesticulation in the description of 3D objects [313] and the creation of free-form shapes using augmented reality interfaces [314] give an indication of how these interfaces can be used as an alternative for designers to give form to and communicate their ideas in 3D space. The portable and ubiquitous nature of tablet computers make them ideal for collaborative design processes like the recording and progressive documentation of design discussions. It is thus likely that NUIs may prove an important factor towards mass collaboration and the democratizing of the design process.
7. *Sustainable systems.* Sustainability and globalization are two forces that will shape the future of product engineering and manufacturing. The needs of sustainability and the global distribution of design and production require going beyond traditional geometry-based CAD to semantics-based KAD. Information/knowledge-based models for products and manufacturing processes that contain key attributes necessary for sustainable and lifecycle information-based manufacturing (such as energy and environmental costs of manufacturing equipment or material recovery costs of product components) will need to be developed. Further, a computational framework that supports sustainability evaluations will require a move from product data exchange to product information and knowledge exchange across different disciplines and domains in a networked information infrastructure, servicing all phases of the lifecycle. This necessitates seamless exchange of vast quantities of information across the design and manufacturing network. Hence, sustainability-based lifecycle support systems will need both syntactic and semantic interoperability through well-defined standards. The integration of SustainabilityExpress into Solidworks and PTC's acquisition of PlanetMetric are indications of the rising significance of sustainability in the industry. These acquisitions increase the need for better knowledge management in the industry.

We see a strong trend towards the true democratization of product conception and realization, influenced by two main factors: (1) the development of design and programming frameworks that are becoming more accessible and intuitive to use and learn, and (2) the developments in manufacturing and supply chain practices towards a more scale-free framework. This has already become visible in the "mobile app store" in both Android [315] and iOS [316] operating systems that provide frameworks for users to develop and market applications without the need to be embedded in an organizational framework, as well as within crowd funding networks like Kickstarter [317] that provide a network for individual or small groups of product developers to showcase their prototypes to potential investors. The rising popularity in the "Makers" subculture [318] — a more engineering and design-oriented aspect of the do-it-yourself culture—is encouraging individuals to participate in open value creation systems like open design and open production [319]. Free CAD applications like Trimble® sketchup [320] and the Autodesk® 123D® family [321] are beginning to focus on this trend, which in turn aids in creating a repository of designs and models for the open source community.

West's analysis of open source platform strategies [322] indicates two strategies that are hybrids of proprietary and open platforms: (1) "opening parts", or retaining control of core layers and opening up commodity layers, and (2) "partly open", or restricted disclosure of technology such that it provides value to customers, while still protecting it from competitors. On the other hand, there exist open-source hardware platforms like the Arduino™ where the hardware design and the software are both open. The value to the inventor is brought in the form of consulting services and new knowledge from the open community of users [323]. Fine's notion of 'industry clockspeed' [324] emphasizes the importance of a firm's capability to design and assemble assets, organizations, and competencies for a competitive advantage that is at best temporary. It is thus vital for new innovative business model designs to take this trend into account and develop sustainable platforms that aid knowledge sharing and reuse on a global level. With the current focus on innovation as the critical factor in boosting the economy, an open architecture framework for products, processes, and services could very well shape the future.

Acknowledgments

We are grateful to Richa Bansal for her contributions to the literature research and development of the initial drafts of the paper. Barbara Guttman, Paul Witherell, Ashok Goel, and Dave Anderson provided many constructive criticisms which greatly enhanced the readability of the paper. Sundar Murugappan, William Bernstein, Devarajan Ramanujan, and Vinayak helped proof-read the paper and provided useful feedback. We would also like to acknowledge the feedback given by the anonymous reviewers. This work is partly supported by the National Science Foundation (NSF) CMMI EAGER Grant # 1153538, A Prototype Network Architecture for Advanced Manufacturing.

Disclaimer

Certain commercial software systems are identified in this paper. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology (NIST); nor does it imply that the products identified are necessarily the best available for the purpose. Further, any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NIST, NSF or any other supporting US government or corporate organizations.

References

[1] Tong C, Sriram R. Artificial intelligence in engineering design, vol. 1. Academic Press; 1992 (Chap. Introduction).
[2] Rittel H, Webber M. Dilemmas in a general theory of planning. Policy Sciences 1973;4(2):155–69.
[3] Pahl G, Beitz W. Engineering design: a systematic approach. Springer; 1996.
[4] Ullman D. Towards the ideal mechanical engineering design support tool. Research in Engineering Design 2002;13(2):55–64.
[5] Labrousse M, Bernard A. FBS-PPRE, an enterprise knowledge lifecycle model. In: Bernard A, Tichkiewitch S, editors. Methods and tools for effective knowledge life-cycle-management. Springer; 2008. p. 285–305.

[6] Goel AK, Vattam S, Wiltgen B, Helms M. Cognitive, collaborative, conceptual and creative—four characteristics of the next generation of knowledge-based cad systems: a study in biologically inspired design. Computer-Aided Design 2012;44(10):879–900.
[7] Owen R, Horváth I. Towards product-related knowledge asset warehousing in enterprises. In: Proceedings of the 4th international symposium on tools and methods of competitive engineering, TMCE 2002; 2002. p. 155–70.
[8] Horváth I. A treatise on order in engineering design research. Research in Engineering Design 2004;15:155–81.
[9] Sainter P, Oldham K, Larkin A, Murton A, Brimble R. Product knowledge management within knowledge-based engineering systems. In: Proceedings of the ASME 2000 design engineering technical conference and computers and information in engineering conference; 2000.
[10] Sriram R. Intelligent systems for engineering: a knowledge-based approach. Springer Verlag; 1997.
[11] Sunnersjö S. A taxonomy of engineering knowledge for design automation. In: Proceedings of TMCE 2010 symposium; 2010.
[12] Ammar-Khodja S, Bernard A. An overview on knowledge management. In: Bernard A, Tichkiewitch S, editors. Methods and tools for effective knowledge life-cycle-management. Springer; 2008. p. 3–21.
[13] Nonaka I. The knowledge creating company. Harvard Business Review 1991; 69:96–104.
[14] Candlot A, Perry N, Bernard A, Ammar-Khodja S. Case study, usiquick project: Methods to capitalise and reuse knowledge in process planning. In: Bernard A, Tichkiewitch S, editors. Methods and tools for effective knowledge life-cycle-management. 2008.
[15] Davis R, Schrobe H, Szolovits P. What is a knowledge representation? AI Magazine 1993;14(1):17–33.
[16] Sowa J. Knowledge representation and reasoning: logical, philosophical, and computational foundations. Brooks/Cole; 2000.
[17] Bernard A, Xu Y. An integrated knowledge reference system for product development. CIRP Annals–Manufacturing Technology 2009;58(1):119–22.
[18] Zhan P, Jayaram U, Kim O, Zhu L. Knowledge representation and ontology mapping methods for product data in engineering applications. Journal of Computing and Information Science in Engineering 2010;10(2).
[19] Pugh S. Engineering design—unscrambling the research issues. Research in Engineering Design 1990;1(1):65–72.
[20] Alexander C. Notes on the synthesis of form. 1st ed. Harvard University Press; 1969.
[21] Simon H. The sciences of the artificial. 2nd ed. MIT Press; 1981. 1st ed. in 1969.
[22] Brown D, Chandrasekaran B. Design problem solving: knowledge structures and control strategies. San Mateo, CA: Morgan Kaufman; 1989.
[23] Goel AK, Chandrasekaran B. Artificial intelligence in engineering design (vol. II). San Diego, CA, USA: Academic Press Professional, Inc.; 1992. p. 165–83 (Chap. Case-based design: a task analysis).
[24] Gorti S, Gupta A, Kim G, Sriram R. An object-oriented representation for product and design processes. Computer-Aided Design 1998;30(7):489–501.
[25] Nii HP. Blackboard application systems, blackboard systems and a knowledge engineering perspective. AI Magazine 1986;7(3):82–106.
[26] Mullins S, Anderson D. Automatic identification of geometric constraints in mechanical assemblies. Computer-Aided Design 1998;30(9):715–26.
[27] Fine C, Whitney DE. Is the make-buy decision process a core competence? In: Proceedings of the 4th international symposium on logistics (ISL99)—logistics in the information age; 1999. p. 31–63.
[28] Fine CH. Clockspeed: winning industry control in the age of temporary advantage. Perseus Books; 1998. p. 43–67 (Chap. 4: the secret of life).
[29] Liu S, Boyle IM. Engineering design: perspectives, challenges, and recent advances. Journal of Engineering Design 2009;20(1):7–19.
[30] Ullman D. The mechanical design process. 3rd ed. McGraw-Hill; 2003.
[31] Allen R, Sriram R. Workshop on knowledge-based systems interoperability. Journal of Research of the National Institute of Standards and Technology 1995;103(5):535–7.
[32] Subrahmanian E, Rachuri S, Fenves SJ, Foufou S, Sriram RD. Product lifecycle management support: a challenge in supporting product design and manufacturing in a networked economy. International Journal of Product Lifecycle Management 2005;1(1):4–25.
[33] Maher D, Balachandran B, Zhang D. Case-based reasoning in design. 1st ed. Psychology Press; 1995.
[34] Gero J. Design prototypes: a knowledge representation schema for design. AI Magazine 1990;11(4):26–36.
[35] Gero JS, Kannengiesser U. The situated function-behaviour-structure framework. Design Studies 2004;25(4):373–91.
[36] Gero J, Kannengiesser U. A function-behavior-structure ontology of processes. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 2007;21:379–91.
[37] Hirtz J, Stone R, McAdams D, Szykman S, Wood K. A functional basis for engineering design: reconciling and evolving previous efforts. Research in Engineering Design 2002;13(2):65–82.
[38] Umeda Y. Supporting conceptual design based on fbs modeler. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 1996;10(4): 275–88.
[39] Shimomura Y, Yoshioka M, Takeda H, Umeda Y, Tomiyama T. Representation of design object based on the functional evolution process model. Journal of Mechanical Design 1998;120(2):221–9.
[40] Christophe F, Bernard A, Coatanéa E. Rfbs: a model for knowledge representation of conceptual design. CIRP Annals–Manufacturing Technology 2010; 59(1):155–8.
[41] Stone R, Wood K. Development of a functional basis for design. Journal of Mechanical Design 2000;122(4):359–70.
[42] Welch R, Dixon J. Representing function, behavior and structure during conceptual design. In: 4th international conference on design theory and methodology; 1992. p. 11–8.
[43] Neville D, Joskowicz L. Representation language for mechanical behavior. ASME Design Engineering Division (Publication) DE 1993;53:1–6.
[44] Mervi R. Integration of functional and feature-based product modelingms/gnosis experience. Computer-Aided Design 1996;28(5):371–81.
[45] Qian L, Gero J. Function-behavior-structure paths and their role in analogy-based design. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 1996;10(4):289–312.
[46] Gorti S, Sriram R. From symbol to form: a framework for conceptual design. Computer-Aided Design 1996;28(11):853–90.
[47] Damski J, Gero J. A logic-based framework for shape representation. Computer-Aided Design 1996;28(3):169–81.
[48] Sheu L, Lin J. Representation scheme for defining and operating form features. Computer-Aided Design 1993;25(6):333–47.
[49] Dani T, Gadh R. Creation of concept shape designs via a virtual reality interface. Computer-Aided Design 1997;29(8):555–63.
[50] Taura T, Nagasaka I, Yamadishi A. Application of evolutionary programming to shape design. Computer-Aided Design 1998;30(1):29–35.
[51] Gero JS, Maher ML. Mutation and analogy to support creativity in computer-aided design. In: Schmitt G, Editor. CAAD Futures '91; 1991. p. 241–9.
[52] Horváth I, van Der Vegte WF. Nucleus-based product conceptualization-part 1: principles and formalization. In: International Conference on Engineering Design; 2003.
[53] Ölvander J, Lundn B, Gavel H. A computerized optimization framework for the morphological matrix applied to aircraft conceptual design. Computer-Aided Design 2009;41(3):187–96.
[54] Malak Jr RJ, Aughenbaugh JM, Paredis C. Multi-attribute utility analysis in set-based conceptual design. Computer-Aided Design 2009;41(3):214–27.
[55] Shai O, Reich Y, Rubin D. Creative conceptual design: extending the scope by infused design. Computer-Aided Design 2007;41(3):117–35.
[56] Shai O, Reich Y. Infused design: I theory. Research in Engineering Design 2004;15(2):93–107.
[57] Shai O, Reich Y. Infused design: I practice. Research in Engineering Design 2004;15(2):108–21.
[58] Carroll J. Human-computer interaction: psychology as a science of design. Annual Review of Psychology 1997;48:61–83.
[59] Lawson B. How designers think: the design process demystified. Kent: Butterworth Architecture; 1990.
[60] Cross N. Expertise in design: an overview. Design Study 2004;25(5):427–41.
[61] Darses F. The constraint satisfaction approach to design: a psychological investigation. Acta Psychologica 1991;78:307–25.
[62] Paivio A. Images in mind: the evolution of a theory. New York: Harvester Wheatsheaf; 1991.
[63] Shah P, Miyake A. The separability of working memory resources for spatial thinking and language processing: an individual differences approach. Journal of Experimental Psychology: General 1996;125(1):4–27.
[64] Ullman D, Wood S, Craig D. The importance of drawing in the mechanical design process. Computers and Graphics 1990;14(2):263–74.
[65] Schooler J, Ohlsson S, Brooks K. Thoughts beyond words: when language overshadows insight. Journal of Experimental Psychology: General 1993; 122(2):166–83.
[66] Goel V. Sketches of thought. MIT Press; 1995.
[67] Stones C, Cassidy T. Comparing synthesis strategies of novice graphic designers using digital and traditional design tools. Design Studies 2007; 28(1):59–72.
[68] Schenk P. Before and after the computer: The role of drawing in graphic design. Visual:Design:Scholarship 2005;1(2):11–20.
[69] Goldschmidt G. The dialectics of sketching. Creativity Research Journal 1991; 4(2):123–43.
[70] Goldschmidt G. Read-write acts of drawing. (TRACEY) Internet Journal Dedicated to Contemporary Drawing Issues 2002; Issue on Syntax of mark and gesture.
[71] Ferguson E. Engineering and the mind's eye. MIT Press; 1992.
[72] McKoy F, Hernandez N, Summers J, Shah J. Influence of design representation on effectiveness of idea generation. In: ASME Design Theory and Methodology conference, Paper No.DTM-21685; 2001. p. 10–3.
[73] Shah JJ, Smith SM, Vargas-Hernandez N. Metrics for measuring ideation effectiveness. Design Studies 2003;24(2):111–34.
[74] Shah JJ, Smith SM, Vargas-Hernandez N. Empirical studies of design ideation. In: ASME Design Theory Conference; 2006.
[75] Vargas-Hernandez N, Shah J, Smith S. Cognitive models of design ideation. In: ASME DTM conference; 2007.
[76] Jonson B. Design ideation: the conceptual sketch in the digital age. Design Studies 2005;26(6):613–24.
[77] Taborda E, Chandrasegaran S, Kisselburgh L, Reid T, Ramani K. Enhancing visual thinking in a toy design course using freehand sketching. In: ASME international design engineering technical conferences and computers and information in engineering conference; 2012.
[78] Hazelrigg G. On the role and use of mathematical models in engineering design. Journal of Mechanical Design 1999;121(3):336–41.

[79] Devanathan S. Towards design representations for product space exploration in early design. Ph.D. Thesis, Purdue University; 2009.
[80] Feldkamp F, Heinrich M, Meyer-Gramann K. Syder—system design for reusability. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 1998;12(4):373–82.
[81] Chen W, Simpson TW, Allenc JK, Mistree F. Satisfying ranged sets of design requirements using design capability indices as metrics. Engineering Optimization 1999;31(5):615–9.
[82] Devanathan S, Ramani K. Combining constraint satisfaction and non-linear optimization to enable configuration driven design. In: International conference on engineering design, ICED '07; 2007.
[83] Devanathan S, Ramani K. Creating polytope representations of design spaces for visual exploration using consistency techniques. Journal of Mechanical Design 2010;132(8):081011 (10 pages).
[84] Michopoulos J, Farhat C, Fish J. Modeling and simulation of multiphysics systems. Journal of Computing and Information Science in Engineering 2005; 5(3):198–213.
[85] Wang G, Shan S. Review of metamodeling techniques in support of engineering design optimization. Journal of Mechanical Design 2007;129(4): 370–80.
[86] Gerritsen B, Gielingh W, Nowacki H, Anderl R, Dankwort W. Editorial: current state and future of product data technologies (pdt). Computer-Aided Design 2008;40(7):735–7.
[87] Pfeifer R, Lungarella M, Iida F. Self-organization, embodiment, and biologically inspired robotics. Science 2007;318(5853):1088–93.
[88] Vakili V, Chiu I, Shu L, McAdams D, Stone R. Including functional models of biological phenomena as design stimuli. In: Proceedings of the ASME 2007 international design engineering technical conferences and computers in engineering conference; 2007.
[89] Vincent J, Mann D. Systematic technology transfer from biology to engineering. Philosophical Transactions Of The Royal Society: Physical Sciences 2002;360(1791):159–73.
[90] Mak T, Shu L. Abstraction of biological analogies for design. CIRP Annals - Manufacturing Technology 2004;53(1):117–20.
[91] Helms M, Vattam SS, Goel AK. Biologically inspired design: process and products. Design Studies 2009;30(5):606–22.
[92] Mak T, Shu L. Using descriptions of biological phenomena for idea generation. Research in Engineering Design 2008;19(1):21–8.
[93] Rezgui Y, Boddy S, Wetherill M, Cooper G. Past, present and future of information and knowledge sharing in the construction industry: towards semantic service-based e-construction. Computer-Aided Design 2011;43(5): 502–15.
[94] Gruber TR. Toward principles for the design of ontologies used for knowledge sharing. International Journal of Human and Computer Studies 1995;43(5-6): 907–28.
[95] Guarino N, Giaretta P. Towards very large knowledge bases—knowledge building and knowledge sharing. IOS Press; 1995. p. 25–32 (Chap. Ontologies and knowledge bases—towards a terminological clarification).
[96] Li Z, Raskin V, Ramani K. Developing engineering ontology for information retrieval. Journal of Computing and Information Science in Engineering 2008; 8(1):011003 (13 pages).
[97] Uschold M, Jasper R. A framework for understanding and classifying ontology applications. In. Benjamins, V, editor. IJCAI99 workshop on ontology and problem solving methods: lessons learned and future trends; 1999.
[98] Gómez-Pérez A, Corcho O, Fernández-López M. Ontological engineering: with examples from the areas of knowledge management, e-commerce and semantic web. In: Advanced information and knowledge processing. Berlin: Springer-Verlag; 2002.
[99] Huhns M, Singh M. Ontologies for agents. Internet Computing, IEEE 1997; 1(6):81–3.
[100] van der Vegte W, Kitamura Y, Mizoguchi R, Horváth I. Ontology-based modeling of product functionality and use—part 2: considering use and unintended behavior. In: Proceedings of the EdiPROD conference; 2002. p. 115–24.
[101] Rachuri S, Baysal M, Roy U, Foufou S, Bock C, Fenves S, et al. Information models for product representation: core and assembly models. International Journal of Product Development 2005;2(3):207–35.
[102] Lee J, Huh S, Fenves S, Sudarsan R, Fiorentini X, Sriram D, et al. A semantic product modeling framework and language for behavior evaluation. In: NIST IR 7681. National Institute of Standards and Technology; 2010.
[103] Witherell P, Krishnamurty S, Grosse IR. Ontologies for supporting engineering design optimization. Journal of Computing and Information Science in Engineering 2007;7(2):141–50.
[104] Horváth I, Kuczogi G, Vergeest J. Development and application of design concept, ontologies for contextual conceptualization. In: Proceedings of the 1998 ASME design engineering technical conferences DETC, DETC98/ CIE-5701; 1998.
[105] Bohm M, Stone R, Szykman S. Enhancing virtual product representations for advanced design repository systems. Journal of Computing and Information Science in Engineering 2005;5(4):360–72.
[106] Holsapple C, Joshi K. A formal knowledge management ontology: conduct, activities, resources and influences. Journal of the American Society for Information Science and Technology 2004;55(7):593–612.
[107] Sicilia M, Lytras M, Rodrguez E, Garca-Barriocanal E. Integrating descriptions of knowledge management learning activities into large ontological structures: a case study. Data and Knowledge Engineering 2006;57(2):111–21.
[108] Nanda J, Thevenot H, Simpson T, Stone R, Bohm M, Shooter S. Product family design knowledge representation, aggregation, reuse, and analysis. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 2007;21(02):173–92.
[109] Nanda J, Simpson T, Kumara S, Shooter S. A methodology for product family ontology development using formal concept analysis and web ontology language. Journal of Computing and Information Science in Engineering 2006;6(2):103–13.
[110] Sohlenius G. Concurrent engineering. CIRP Annals - Manufacturing Technology 1992;41(2):645–55.
[111] Prasad B. Concurrent engineering fundamentals (Two volume series). 1st ed. Prentice Hall; 1995.
[112] Peak R, Burkhart R, Friedenthal S, Wilson M, Bajaj M, Kim I. Simulation-based design using sysml-part1: a parametrics primer. In: INCOSE International Symposium; 2007.
[113] Sage AP, Cuppan CD. On the systems engineering and management of systems of systems and federations of systems. Information-Knowledge-Systems Management 2001;2:325–45.
[114] Mark Maier ER. The art of systems architecting. 2nd ed. CRC Press; 2000 (Chap. one: Extending the architecture paradigm).
[115] Prasad B. Concurrent engineering fundamentals: integrated product and process organization, vol. I. New Jersey: Prentice Hall; 1999.
[116] Chung CCW, Choi JK, Ramani K, Patwardhan H. Product node architecture: a systematic approach to provide structured flexibility in distributed product development. Concurrent Engineering 2005;13(3):219–32.
[117] Friedenthal S, Moore A, Steiner R. A practical guide to SysML: the systems modeling language. 2nd ed. The MK/OMG Press; 2011.
[118] Sriram RD. Distributed and integrated collaborative engineering design. Sarvan Publishers; 2002.
[119] Wang L, Shen W, Xie H, Neelamkavil J, Pardasani A. Collaborative conceptual design — state of art and future trends. Computer-Aided Design 2002;34(13): 981–96.
[120] Sriram R, Szykman S, Durham D. Guest editorial: special issue on collaborative engineering. Journal of Computing and Information Science in Engineering 2006;6(2):93–5.
[121] McMahon C, Lowe A, Culley S. An information connection model of design. In: Proceedings of the international conference of engineering design; 1999. p. 1651–56.
[122] Marsh J. The capture and structure of design experience. Ph.D. Thesis; 1997.
[123] Sure Y, Erdmann M, Angele J, Staab S, Stueder R, Wenke D. Ontoedit: collaborative ontology development for the semantic web. Lecture notes in computer science 2002;2342:221–35.
[124] Staley SM, Anderson DC. A functional specification for cad databases. Computer-Aided Design 1986;18(3):132–8.
[125] Turner GP, Anderson DC. An object-oriented approach to interactive, feature-based design for quick turnaround manufacturing. In: Proceedings of the 1988 ASME international computers in engineering conference, vol. 1; 1988. p. 551–6.
[126] Anderson D, Chang T. Geometric reasoning in feature based design and process planning. Computers & Graphics An International Journal 1990; 14(2):225–35.
[127] Shah JJ, Anderson D, Kim YS, Joshi S. A discourse on geometric feature recognition from cad models. Journal of Computing and Information Science in Engineering 2001;1(1):41–51.
[128] Ma YS, Tong T. Associative feature modeling for concurrent engineering integration. Computers in Industry 2003;51(1):51–71.
[129] Chen G, Ma YS, Thimm G, Tang SH. Associations in a unified feature modeling scheme. Journal of Information Science in Engineering 2006;6(2):114–26.
[130] Szykman S, Sriram R, Smith Se. Proceedings of the nist design repository workshop, nistir 6159, 1998.
[131] Szykman S, Bochenek C, Racz J, Senfaute J, Sriram R. Design repositories: next generation engineering design databases. IEEE Intelligent Systems 2000; 15(3):48–55.
[132] Xue D, Yang H. A concurrent engineering-oriented design database representation model. Computer-Aided Design 2004;36(10):947–65.
[133] Kim J, Ling S, Will P. Technical report. The University of Southern California. Information Sciences Institute; 1997 (Chap. Ontology engineering for active catalog).
[134] Jiao J, Simpson T, Siddique Z. Product family design and platform-based product development: a state-of-the-art review. Journal of Intelligent Manufacturing 2007;18(1):5–29.
[135] Thevenot H, Alizon F, Simpson T, Shooter S. An index-based method to manage the tradeoff between diversity and commonality during product family design. Concurrent Engineering: Research and Applications 2007; 15(2):127–39.
[136] Ouertani M, Baina S, Gzara L, Morel G. Traceability and management of dispersed product knowledge during design and manufacturing. Computer-Aided Design 2011;43(5):546–62.
[137] Hoffmann CM, Shapiro V, Srinivasan V. Geometric interoperability for resilient manufacturing. Tech. Rep. TR-11-015. Purdue University; 2011.
[138] Zhang W, Yin J. Exploring semantic web technologies for ontology-based modeling in collaborative engineering design. The International Journal of Advanced Manufacturing Technology 2008;36:833–43.
[139] Tomiyama T, Gu P, Jin Y, Lutters D, Kind C, Kimura F. Design methodologies: industrial and educational applications. CIRP Annals - Manufacturing Technology 2009;58(2):543–65.

[140] Regli W, Hu X, Atwood M, Sun W. A survey of design rationale systems: approaches, representation, capture and retrieval. Engineering with Computers 2000;16(3-4):209–35.
[141] Shum SJB, Hammond N. Argumentation-based design rationale: what use at what cost? International Journal of Human Studies 1994;40(4):603–52.
[142] Szykman S, Sriram R, Regli W. The role of knowledge in next-generation product development systems. Journal of Computing and Information Science in Engineering 2001;1(1):3–11.
[143] MacLean A, Young R, Bellotti V, Moran T. Questions, options, and criteria: Elements of design space analysis. Human-Computer Interaction 1991; 6(3–4):201–50.
[144] Kunz W, Rittel H. Issues as elements of information systems. Working Paper No. 131. Institute of Urban and Regional Development, University of California, Berkeley, California; 1970.
[145] Conklin E, Burgess-Yakemovic K. A process-oriented approach to design rationale. Human-Computer Interaction 1991;6(3):357–91.
[146] Lee J. Design rationale systems: understanding the issues. IEEE Expert 1997; 12(3):78–85.
[147] Peña Mora F, Sriram R, Logcher R. Design rationale for computer supported conflict mitigation. ASCE Journal of Computing in Civil Engineering 1995; 9(1):57–72.
[148] Takai S. A case-based reasoning approach toward developing a belief about the cost of concept. Research in Engineering Design 2009;20(4):255–64.
[149] Garcia A, Howard H. Acquiring design knowledge through design decision justification. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 1992;6(1):59–71.
[150] Ganeshan R, Garrett J, Finger S. A framework for representing design intent. Design Studies 1994;15(1):59–84.
[151] Quereshi S, Shah J, Urban SD, Harter E, Parazzoli C. et al. Integration model to support archival of design history in databases. In: Proceedings of the 1997 ASME design engineering technical conferences and computers in engineering conference; 1997.
[152] Myers K, Zumel N, Garcia P. Acquiring design rationale automatically. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 2000;14(02):115–35.
[153] Shum SJB, Selvin AM, Sierhuis M, Conklin J, Haley CB, Nuseibeh B. Hypermedia support for argumentation-based rationale: 15 years on from gibis and qoc. In: Dutoit A, McCall R, Mistrik I, Peach B, editors. Rationale management in software engineering. Springer-Verlag; 2006. p. 111–32.
[154] Bracewell R, Wallace K, Moss M, Knott D. Capturing design rationale. Computer-Aided Design 2009;41(3):173–86.
[155] Sung R, Ritchie JM, Rea HJ, Corney J. Automated design knowledge capture and representation in single-user cad environments. Journal of Engineering Design 2011;22(7):487–503.
[156] Meniru K, Rivard H, Bedard C. Specifications for computer-aided conceptual building design. Design Studies 2003;24(1):51–71.
[157] Piegl L. Editorial: Cad'04 special issue: modeling and geometry representations for cad. Computer-Aided Design 2005;37(8):781.
[158] Chalfan KM. A knowledge system that integrates heterogeneous software for a design application. The AI Magazine 1986;7(3):80–4.
[159] Roller D. An approach to computer-aided parametric design. Computer-Aided Design 1991;23(5):385–91.
[160] Verroust A, Schonek F, Roller D. Rule-oriented method for parameterized computer-aided design. Computer-Aided Design 1992;24(10):531–41.
[161] Anderl R, Mendgen R. Modelling with constraints: theoretical foundation and application. Computer-Aided Design 1996;28(3):155–68.
[162] Gero J. Design prototypes: a knowledge representation schema for design. AI Magazine 1990;11(4):26–36.
[163] Sutherland I. Sketchpad: a man-machine graphical communication system. In: Proceedings of the SHARE design automation workshop, DAC '64; 1964. p. 6.329–6.346.
[164] MacNeal RH. Some organizational aspects of nastran. Nuclear Engineering and Design 1974;29(2):254–65.
[165] Ptc history and acquisitions, http://www.ptc.com/company/history-and-acquisitions.htm; 2011.
[166] Sobieszczanski-Sobieski J, Haftka RT. Multidisciplinary aerospace design optimization: survey of recent developments. Structural and Multidisciplinary Optimization 1997;14(1):1–23.
[167] Rocca GL. Knowledge based engineering: between ai and cad. review of a language based technology to support engineering design. Advanced Engineering Informatics 2012;26(2):159–79.
[168] Johnson RH. Product data management—with solid modeling. Computer-Aided Engineering Journal 1986;3(4):129–32.
[169] Pratt MJ. Virtual prototyping: virtual environments and the product design process. Springer; 1995. p. 113–28 (Chap. 10: Virtual prototypes and product models in mechanical engineering).
[170] de Sa AG, Zachmann G. Virtual reality as a tool for verification of assembly and maintenance processes. Computer and Graphics 1999;23(3):389–403.
[171] Autodesk press release archive. URL http://usa.autodesk.com/adsk/servlet/index?siteID=123112\&id=14271573 ; 2012.
[172] Team SM. Systems modeling language (sysml) specification: Version 1.0 alpha. Tech. Rep. ad/2006-03-01. Object Management Group; 2005.
[173] Bertoni M, Chirumalla K. Leveraging web 2.0 in new product development: lessons learned from a cross-company study. Journal of Universal Computer Science 2011;17(4):548–64.
[174] Tian C, Masry M, Lipson L. Physical sketching: reconstruction and analysis of 3d objects from freehand sketches. Computer Aided Design 2009;41(3): 147–58.
[175] Stahovich T, Davis R, Shrobe H. Generating multiple new designs from a sketch. In: Proceedings of the thirteenth national conference on artificial intelligence and the eighth innovative applications of artificial intelligence conference; 1996. p. 1022–30.
[176] Zeleznik R, Herndon K, Hughes J. Sketch: an interface for sketching 3d scenes. In: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques; 1996. p. 163–70.
[177] Cohen J, Markosian L, Zeleznik R, Hughes J, Barzel R. An interface for sketching 3d curves. In: Proceedings of the 1999 symposium on interactive 3D graphics; 1999. p. 17–21.
[178] Qin S, Wright D, Jordanov I. A conceptual design tool: a sketch and fuzzy logic based system. Proceedings of the Institution of Mechanical Engineering Part B, Journal of Engineering Manufacture 2001;215(1):111–6.
[179] Cherlin J, Samavati F, Sousa M, Jorge J. Sketch-based modeling with few strokes. In: Proceedings of the 21st spring conference on computer graphics; 2005. p. 137–45.
[180] Yang C, Sharon D, Panne M. Sketch-based modeling of parameterized objects. In: Eurographics workshop on sketch-based interfaces and modeling; 2005.
[181] Murugappan S, Ramani K. Feasy: a sketch-based interface integrating structural analysis in early design. In: ASME design engineering technical conferences, computers and information science in engineering; 2009.
[182] Herbert D. Study drawings in architectural design: applications for cad systems. In: Proceedings of the 1987 workshop of the association for computer-aided design in architecture, ACADIA; 1987.
[183] Wei W, Chang W. Analytic network process-based model for selecting an optimal product design solution with zero-one goal programming. Journal of Engineering Design 2008;19(1):15–44.
[184] Reich Y, Paz A. Managing product quality, risk, and resources through resource quality function deployment. Journal of Engineering Design 2008; 19(3):249–67.
[185] Fernandez MG, Seepersad CC, Rosen DW, Allen JK. Decision support in concurrent engineering the utility-based selection decision support problem. Concurrent Engineering 2005;13(1):13–27.
[186] Xiao A, Seepersad CC, Allen JK, Rosen DW, Mistree F. Design for manufacturing: application of collaborative multidisciplinary decision-making methodology. Engineering Optimization 2007;39(4):429–51.
[187] Molcho G, Schneor R, Zipori Y, Kowalsi P, Denkena B, Shpitalni M. Computer aided manufacturability analysis closing the cad-cam knowledge gap. In: ASME conference proceedings. 48357; 2008. p. 309–15.
[188] Guindon R. Knowledge exploited by experts during software system design. International Journal of Man-Machine Studies 1990;33(3):279–304.
[189] Hou S, Ramani K. Dynamic query interface for 3d shape search. In: Proceedings of DETC '04: ASME 2004 design engineering technical conferences and computers and information in engineering conference; 2004.
[190] Industrynet®. URL http://www.industrynet.com/ ; 2012.
[191] Altavista™. URL http://www.altavista.com/ ; 2012.
[192] Li Z, Liu M, Ramani K. Review of product information retrieval: representation and indexing. In: Proceedings of ASME 2004 design engineering technical conference and computers and information in engineering conference, vol. 4; 2004. p. 971–9.
[193] McMahon C, Lowe A, Culley S, Corderoy M, Crossland R, Shah T, et al. Waypoint: an integrated search and retrieval system for engineering documents. Journal of Computing and Information Science in Engineering 2004;4(4):329–38.
[194] Iyer N, Lou K, Jayanti S, Kalyanaraman Y, Ramani K. Shape-based searching for product lifecycle applications. Computer-Aided Design 2005;37(13): 1435–46.
[195] Uschold M, Grüninger M. Ontologies and semantics for seamless connectivity. SIGMOD Record 2004;33(4):58–64.
[196] Li Z, Liu M, Anderson DC, Ramani K. Semantics-based design knowledge annotation and retrieval. In: Proceedings of IDETC/CIE 2005: ASME 2005 international design engineering technical conferences and computers in engineering conference; 2005.
[197] Kim KY, Kim YS. Causal design knowledge: alternative representation method for product development knowledge management. Computer-Aided Design 2011;43(9):1137–53.
[198] Zaharia T, Preteux F. Shape-based retrieval of 3d mesh models. In: IEEE international conference on multimedia and expo, ICME 2002; 2002.
[199] Ohbuchi R, Minamitani T, Takei T. Shape-similarity search of 3d models by using enhanced shape functions. International Journal of Computer Applications in Technology (IJCAT) 2005;23(2-4):70–85.
[200] Iyer N, Jayanti S, Lou K, Kalyanaraman Y, Ramani K. Three-dimensional shape searching: state-of-the-art review and future trends. Computer-Aided Design 2005;37(5):509–30.
[201] Jayanti S, Kalyanaraman Y, Iyer N, Ramani K. Developing an engineering shape benchmark for cad models. Computer-Aided Design 2006;38(9): 939–53.
[202] Alizon F, Shooter S, Simpson T. Reuse of manufacturing knowledge to facilitate platform-based product realization. Journal of Information Science in Engineering 2006;6(2):170–8.
[203] Vizseek®. URL http://www.vizseek.com/Info/vizseek.aspx ; 2012.
[204] Oliver J. Guest editorial: Special issue on virtual reality application in product development. Journal of Computing and Information Science in Engineering 2004;4(2):81–2.

[205] Kan H, Duffy V, Su C. An internet virtual reality collaborative environment for effective product design. Computers in Industry 2001;45(2):197–213.
[206] Jayaram S, Jayaram U, Wang Y, Tirumali H, Lyons K, Hart P. Vade: a virtual assembly design environment. IEEE Computer Graphics and Applications 1999;19(6):44–50.
[207] Jayaram S, Vance J, Gadh R, Jayaram U, Srinivasan H. Assessment of vr technology and its applications to engineering problems. Journal of Computing and Information Sciences in Engineering 2001;1(1):72–83.
[208] Wang Y, Jayaram U, Jayaram S, Shaikh I. Methods and algorithms for constraint based virtual assembly. Virtual Reality 2003;6:229–43.
[209] Gill S, Ruddle R. Using virtual humans to solve real ergonomic design problems. In: Proceedings of the 1998 international conference on simulation; 1998. p. 223–9.
[210] Ryken MJ, Vance JM. Applying virtual reality techniques to the interactive stress analysis of a tractor lift arm. Finite Elements in Analysis and Design 2000;35(2):141–55.
[211] Jayaram U, Jayaram S, Shaikh I, Kim Y, Palmer C. Introducing quantitative analysis methods into virtual environments for real-time and continuous ergonomic evaluations. Computers in Industry 2006;57(3):283–96.
[212] Bernard A, Hasan R. Working situation model for safety integration during design phase. CIRP Annals - Manufacturing Technology 2002;51(1):119–22.
[213] Houssin R, Bernard A, Martin P, Ris G, Cherrier F. Information system based on the working situation model for an original concurrent engineering design approach. Journal of engineering design 2006;17(1):35–54.
[214] Hasan R, Bernard A, Ciccotelli J, Martin P. Integrating safety into the design process: elements and concepts relative to the working situation. Safety Science 2003;41(2-3):155–79.
[215] Dukic T, Ronnang M, Christmansson M. Evaluation of ergonomics in a virtual manufacturing process. Journal of Engineering Design 2007;18(2):125–37.
[216] Dorozhkin D, Vance J, Rehn G, Lemessi M. Coupling of interactive manufacturing operations simulation and immersive virtual reality. Virtual Reality 2012;16:15–23.
[217] Cappelli F, Delogu M, Pierini M, Schiavone F. Design for disassembly: a methodology for identifying the optimal disassembly sequence. Journal of Engineering Design 2007;18(6):563–75.
[218] Ottosson S, Holmadahl L. Web-based virtual reality. Journal of Engineering Design 2007;18(2):103–11.
[219] Song P, Krovi V, Kumar V, Mahoney R. Design and virtual prototyping of humanworn manipulation devices. In: Proceedings of the 1999 ASME design technical conference and computers in engineering conference, DETC99/CIE-9029; 1999.
[220] Conti G, Ucelli G, De Amicis R. Jcad-vr—a multi-user virtual reality design system for conceptual design. In: TOPICS. Reports of the INI-GraphicsNet 2003;15:7–9.
[221] Rosenman M, Merrick K, Maher ML, Marchant D. Designworld: a multidisciplinary, collaborative design environment using agents in a virtual world. In: Proceedings of the second international conference on design computing and cognition; 2006.
[222] Maher M, Rosenman M, Merrick K. Agents for multidisciplinary design in virtual worlds. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 2007;21(03):267–77.
[223] Furlong TJ, Vance JM, Larochelle PM. Spherical mechanism synthesis in virtual reality. In: Proceedings of the ASME DETC98: design automation conference; 1998.
[224] Rosenman M, Smith G, Maher ML, Ding L, Marchant D. Multidisciplinary collaborative design in virtual environments. Automation in Construction 2007;16(1):37–44.
[225] Regli WC. Research issues in network-centric computer aided design. In: Proc. TeamCAD: CVU/NIST workshop on collaborative design; 1997. p. 3–10.
[226] Hsu W, Woon I. Current research in the conceptual design of mechanical products. Computer-Aided Design 1998;30(5):377–89.
[227] Dong A, Agogino A. Managing design information in enterprise-wide cad using 'smart drawings'. Computer-Aided Design 1998;30(6):425–35.
[228] Menon J, Regli W. Editorial, special issue: network centric cad. Computer-Aided Design 1998;30(6):409.
[229] Ding L, Davies D, McMahon C. The integration of lightweight representation and annotation for collaborative design representation. Research in Engineering Design 2009;19(4):223–38.
[230] Li W, Lu W, Fuh J, Wong Y. Collaborative computer-aided design—research and development status. Computer-Aided Design 2005;37(9):931–40.
[231] Shen Y, Ong S, Nee A. Product information visualization and augmentation in collaborative design. Computer-Aided Design 2008;40(9):963–74.
[232] Numata J. Knowledge amplification: an information system for engineering management. Sony's Innovation in Management Series 1996;17.
[233] Shen W, Barthès J. An experimental environment for exchanging engineering design knowledge by cognitive agent. In: Mantyla M, Finger S, Tomiyama T, editors. Knowledge intensive CAD-2. London: Chapman & Hall; 1996. p. 19–38.
[234] Huang G, Lee S, Mak K. Web-based product and process data modeling in concurrent 'design for x'. Robotics and Computer-Integrated Manufacturing 1999;15(1):53–63.
[235] Rodgers P, Huxor A, Caldwell N. Design support using distributed web-based ai tools. Research in Engineering Design 1999;11(1):31–44.
[236] Cutkosky M, Engelmore R, Fikes R, Genesereth M, Gruber T, Mark W, et al. Pact: An experiment in integrating concurrent engineering systems. IEEE Computer 1993;26(1):28–37.
[237] Shen W, Barthès J. Dide: a multi-agent environment for engineering design. In: Proceedings of the first international conference on multi-agent systems; 1995. p. 344–51.
[238] Campbell M, J C, Kotovsky K. A-design: an agent-based approach to conceptual design in a dynamic environment. Research in Engineering Design 1999;11(3):172–92.
[239] O'Brien P, Wiegand M. Agent-based process management: applying agents to workflow. Knowledge Engineering Review 1998;13(2):61–174.
[240] Brazier F, Jonker C, Treur J, Wijngaards N. Compositional design of a generic design agent. Design Studies Journal 2001;22(5):439–71.
[241] Pimmler T, Eppinger S. Integration analysis of product decompositions. In: Proceedings of the ASME 6th international conference on design theory and methodology; 1994.
[242] Eppinger SD, Whitney DE, Smith RP, Gebala DA. A model-based method for organizing tasks in product development. Research in Engineering Design 1994;6:1–13.
[243] Browning T. Applying the design structure matrix to system decomposition and integration problems: a review and new directions. IEEE Transactions on Engineering Management 2001;48(3):292–306.
[244] Suh NP. Axiomatic design theory for systems. Research in Engineering Design 1998;10:189–209.
[245] Zakarian A, Knight J, Baghdasaryan L. Modelling and analysis of system robustness. Journal of Engineering Design 2007;18(3):243–63.
[246] Jugulum R, Frey D. Toward a taxonomy of concept designs for improved robustness. Journal of Engineering Design 2007;19(1):75–95.
[247] Xue D, Cheung S, Gu P. Parameter design considering the impact of design changes on downstream processes based upon the Taguchi method. Journal of Engineering Design 2008;19(4):299–319.
[248] Bilgic T, Rock D. Product data management systems: State-of-the-art and the future. In: Proceedings of the 1997 ASME design engineering technical conferences and computers in engineering conference; 1997.
[249] Stark J. Product lifecycle management: 21st century paradigm for product realization. Springer; 2004.
[250] Abramovici M. The future of product development. Berlin Heidelberg: Springer; 2007. p. 665–74 (Chap. Future trends in PLM).
[251] Anderson C. Stage of the product life cycle, business strategy, and business performance. Academy of Management Journal 1984;27(1):5–24.
[252] Ameri F, Dutta D. Product lifecycle management: closing the knowledge loops. Computer-Aided Design and Applications 2005;2(5):577–90.
[253] Lee S, Maa YS, Thimm G, Verstraeten J. Product lifecycle management in aviation maintenance, repair and overhaul. Computers in Industry 2008; 59(2–3):296–303.
[254] Le Duigou J, Bernard A, Perry N, Delplace J. Inductive approach for the specification of a plm system dedicated to smes. In: Proceedings of the CIRP design conference; 2009. p. 109–15.
[255] Szykman S, Fenves S, Shooter S, Keirouz W. A foundation for interoperability in next-generation product development systems. In: Proceedings of the 1999 ASME design engineering technical conferences and computers and information in engineering conference; 2000.
[256] Ramani K, Cunningham R, Devanathan S, Subramaniam J, Patwardhan H. Technology review of mass customization. In: Conference proceedings, international conference on economic, technical and organizational aspects of product configuration systems; 2004. p. 5–11.
[257] Meyer M, Utterback J. The product family and the dynamics of core capability. Sloan Management Review 1992;34(3):29–47.
[258] Meyer M, Lehnerd A. The power of product platforms. Free Press; 1997.
[259] Robertson D, Ulrich K. Planning for product platforms. Sloan Management Review 1998;39(4):19–31.
[260] Huang G, Simpson T, Pine B. The power of products platforms in mass customization. International Journal of Mass Customization 2005;1(1):1–13.
[261] Seepersad C, Mistree F, Allen J. A quantitative approach for designing multiple product platforms for an evolving portfolio of products. In: ASME conference proceedings, vol. 2002; 2002. p. 579–92.
[262] Gu P, Slevinsky M. Mechanical bus for modular product design. CIRP Annals - Manufacturing Technology 2003;52(1):113–6.
[263] Huang G, Zhang X, Lo V. Optimal supply chain configuration for platform products: impacts of commonality, demand variability and quality discount. International Journal of Mass Customization 2005;1(1):107–33.
[264] Siddique Z, Rosen DW. On combinatorial design spaces for the configuration design of product families. Artificial Intelligence in Engineering Design, Analysis and Manufacturing 2001;15:91–108.
[265] Hernandez G, Allen JK, Mistree F. A theory and method for combining multiple approaches for product customisation. International Journal of Mass Customization 2006;1(2–3):315–39.
[266] Ruohonen M, Riihimaa J, Mäkipää M. Knowledge based mass customisation strategies: cases from finnish metal and electronics industries. International Journal of Mass Customisation 2006;1(2–3):340–59.
[267] Frutos J, Borenstein D. A framework to support customer-company interaction in mass customization environments. Computers in Industry 2004;54(2):115–35.
[268] Helms MM, Ahmadi M, Jih WJK, Ettkin LP. Technologies in support of mass customization strategy: exploring the linkages between e-commerce and knowledge management. Computers in Industry 2008;59(4):351–63.
[269] Raman M. Editorial: applied knowledge management in an institutional context. Knowledge Management & E-Learning: An International Journal 2009;1(2):81–9.

[270] Ruggles R. Knowledge management tools. Oxford: Butterworth-Heinemann; 1997.
[271] Tiwana A, Ramesh B. A design knowledge management system to support collaborative information product evolution. Decision Support Systems 2001;31(2):241–62.
[272] Du Plessis M. What bars organisations from managing knowledge successfully? International Journal of Information Management 2008;28:285–92.
[273] Bernard A, Tichkiewitch SE. Methods and tools for effective knowledge lifecycle-management. Springer; 2008.
[274] Rezayat M. Editorial: some aspects of product and process development in the 21st century. Computer-Aided Design 2000;32(1, 5-6):83.
[275] A short history of Catia & Dassault Systèmes. URL http://www.edstechnologies.com/download/history-catia.pdf ; 2003.
[276] Weisberg DE. The engineering design revolution. CADHistory.net. Chap. 11: CALMA; 2008. URL http://www.cadhistory.net/ (accessed 2012).
[277] Carlson W. A critical history of computer graphics and animation, Section 10: Cad/cam/cadd/cae. URL http://design.osu.edu/carlson/history/lesson10.html (accessed 2012).
[278] Tovey M. Computer-aided vehicle styling. Computer-Aided Design 1989; 21(3):172–9.
[279] Bliss FW. Interactive computer graphics at Ford motor company. SIGGRAPH Comput Graph 1980;14(3):218–24.
[280] A midterm test for the cad industry. URL http://gfxspeak.com/2012/04/18/a-midterm-test-for-the-cad-industry/ ; 2012.
[281] Autodesk milestones. URL http://en.autodesk.ca/adsk/servlet/pc/index?siteID=9719649&id=15622058 ; 2012.
[282] Siemens plm timeline. URL http://www.plm.automation.siemens.com/en_us/about_us/facts_philosophy/timeline.shtml ; 2011.
[283] Dassault systèmes: History of the company. URL http://www.3ds.com/fileadmin/COMPANY/FINANCE/PDF/History%20of%20the%20Company_2010.pdf ; 2010.
[284] Harik R, Derigent W, Ris G. Computer aided process planning in aircraft manufacturing. Computer-Aided Design and Applications 2008;5(1-4): 953–62.
[285] Zhou X, Qiu Y, Hua G, Wang H, Ruan X. A feasible approach to the integration of cad and capp. Computer-Aided Design 2007;39(4):324–38.
[286] Scott A. An evaluation of three commercially available integrated design framework packages for use in the space systems design lab. Georgia Institute of Technology, 2001. URL http://www.ssdl.gatech.edu/papers/mastersProjects/ScottA-8900.pdf.
[287] Conner K, Prahalad C. A resource-based theory of the firm: Knowledge versus opportunism. Organization Science 1996;7(5):477–501.
[288] Choi B, Lee H. An empirical investigation of km styles and their effect on corporate performance. Information & Management 2003;40(5): 403–17.
[289] Bergman JP, Jantunen A, Saksa JM. Managing knowledge creation and sharing scenarios and dynamic capabilities in inter-industrial knowledge networks. Journal of Knowledge Management 2004;8(6):63–76.
[290] Weiss A. The power of collective intelligence. Networker 2005;9(3):16–25.
[291] Cunningham W, Leuf B. The wiki way : quick collaboration on the web. Boston, EUA: Addison-Wesley; 2001.
[292] Werasinghe J, Salustri F. Use of wikis as an engineering collaborative tool. In: International conference on engineering design ICED 2007; 2007.
[293] Wodehouse A, Grierson H, Ion W, Juster N, Lynn A, Stone A. Tikiwiki: a tool to support engineering design students in concept generation. In: International engineering and product design education conference IEPDE; 2004. p. 449–56.
[294] Chen HL, Cannon D, Gabrio J, Leifer L, Bailey T. American society for engineering education annual conference and exposition; 2005 (Chap. Using wikis and weblogs to support reflective learning in an introductory engineering design course).
[295] Raitman R, Augar N, Zhou W. Employing wikis for online collaboration in the e-learning environment: case study. In: Third international conference on information technologies and applications ICITA05; 2005. p. 142–56.
[296] Walthall C, Devanathan S, Ramani K, Hirleman E, Kisselburgh L, Yang M. Evaluating wikis as a communicative medium for collaboration within co-located and distributed design teams. Journal of Mechanical Design 2011; 133(7):071001 (11 pages).
[297] Project halo. URL http://www.projecthalo.com/ ; 2010.
[298] Sriram D. Toward a comprehensive knowledge-base for engineering, Tech. Rep. FS-92-01, AAAI; 1992.
[299] Wolfram-alpha: computational knowledge engine. URL http://www.wolframalpha.com/ ; 2012.
[300] Autonomy. URL http://www.autonomy.com/ ; 2012.
[301] Arciszewski T, Cornell J. Bio-inspiration: learning creative design principia. In: Smith I, editor. Intelligent computing in engineering and architecture. Lecture notes in computer science, vol. 4200. Berlin, Heidelberg: Springer; 2006. p. 32–53.
[302] The national center for biomedical ontology. URL http://www.bioontology.org ; 2012.
[303] Howe J. Crowdsourcing: why the power of the crowd is driving the future of business. New York: Crown Business; 2008.
[304] Innocentive. URL http://www.innocentive.com ; 2012.
[305] Quirky: Social product development. URL http://www.quirky.com/ ; 2012.
[306] Local motors. URL http://www.localmotors.com/ ; 2012.
[307] Avm collaboration capability (vehicleforge.mil). URL http://www.darpa.mil/Our_Work/TTO/Programs/AVM/AVM_Collaboration_Capability_(Vehicleforge_mil).aspx ; 2012.
[308] Arduino. URL http://arduino.cc/ ; 2012.
[309] Open source car. URL http://www.theoscarproject.org/ ; 2012.
[310] Le Q, Panchal JH. Modeling the effect of product architecture on mass-collaborative processes. Journal of Computing and Information Science in Engineering 2011;11(1):011003 (12 pages).
[311] Shapeways. URL http://www.shapeways.com/ ; 2012.
[312] Wigdor D, Wixon D. Brave NUI world: designing natural user interfaces for touch and gesture. Morgan Kaufman; 2011.
[313] Holz C, Wilson A. Data miming: inferring spatial object descriptions from human gesture. In: Proceedings of the 2011 annual conference on human factors in computing systems. CHI '11; 2011. p. 811–820.
[314] Fuge M, Yumer ME, Orbay G, Kara LB. Conceptual design and modification of freeform surfaces using dual shape representations in augmented reality environments. Computer-Aided Design 2012;44(10):1020–32.
[315] Android. URL http://www.android.com/ ; 2012.
[316] Apple ios. URL http://www.apple.com/ios/ ; 2012.
[317] Kickstarter. URL http://www.kickstarter.com/ ; 2012.
[318] More than just digital quilting. URL http://sketchup.google.com/ ; 2011.
[319] Wulfsberg J, Redlich T, Bruhns FL. Open production: scientific foundation for co-creative product realization. Production Engineering 2011;5:127–39.
[320] Google sketchup. URL http://sketchup.google.com/ ; 2012.
[321] Autodesk 123d. URL http://www.123dapp.com/create ; 2012.
[322] West J. How open is open enough?: melding proprietary and open source platform strategies. Research Policy 2003;32(7):1259–85.
[323] Torres P. Why the arduino won and why it's here to stay. URL http://blog.makezine.com/2011/02/10/why-the-arduino-won-and-why-its-here-to-stay/ ; 2011.
[324] Fine C. Clockspeed: winning industry control in the age of temporary advantage. Perseus; 1998.

Senthil K. Chandrasegaran is a Ph.D. student in the School of Mechanical Engineering at Purdue. He obtained his bachelor's degree in mechanical engineering in 2000 from the Regional Engineering College (now National Institute of Technology), Trichy, India. He has nine years' industry experience in areas like automotive interior design, structural analysis in automotive and heavy engineering sectors, and knowledge-based engineering in the heavy engineering sector. His research interests include design pedagogy, knowledge engineering, and specifically the integration of computer support tools to aid design learning in the classroom.

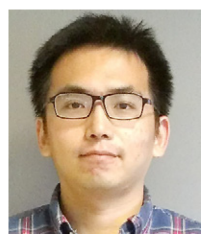
Karthik Ramani is the Donald W. Feddersen Professor in the School of Mechanical Engineering at Purdue University. He earned his B.Tech from the Indian Institute of Technology, Madras, in 1985, an MS from The Ohio State University, in 1987, and a Ph.D. from Stanford University, in 1991, all in Mechanical Engineering. He serves on the editorial board of the Elsevier Journal of Computer-Aided Design and the ASME Journal of Mechanical Design. His research lies at the intersection of mechanical engineering, and information science and technology. His research areas encompass design and manufacturing, new kernels for shape understanding using machine learning, geometric computing and human-computer natural user interaction and interfaces with shapes and sketches. Major areas of emphasis in his group are computer support for early design, shape searching, sketch-based design, cyber and design learning, sustainable design, and natural user interfaces for shape modeling. He is also currently serving on the NSF Advisory Committee for the SBIR/STTR program of the Industrial Innovation and Partnerships program. In 2006 and 2007, he won the Most Cited Journal Paper award from Computer-Aided Design and the Research Excellence award in the College of Engineering at Purdue University. In 2009, he won the Outstanding Commercialization award from Purdue University and the ASME Best Paper Award from technical committees twice at the IDETC. In 2012 his lab's paper won the all-conference best paper award from ASME-CIE. He has published over 200 peer-reviewed papers, over 90 Journal publications; over 70 invited presentations, and granted 10 patents.

Ram D. Sriram is currently the chief of the Software and Systems Division, Information Technology Laboratory, at the National Institute of Standards and Technology. Before joining the Software and Systems Division, he was the leader of the Design and Process group in the Manufacturing Systems Integration Division, Manufacturing Engineering Laboratory, where he conducted research on standards for interoperability of computer-aided design systems. He was also the manager of the Sustainable Manufacturing Program. Prior to joining NIST, he was on the engineering faculty (1986–1994) at the Massachusetts Institute of Technology (MIT) and was instrumental in setting up the Intelligent Engineering Systems Laboratory. At MIT, he initiated the MIT-DICE project, which was one of the pioneering projects in collaborative engineering. Dr. Sriram has extensive experience in developing knowledge-based expert systems, natural language interfaces, object-oriented software development, life-cycle product and process models, geometrical modelers, object-oriented databases for industrial applications, health care informatics, bioinformatics, and bio-imaging. He has consulted for several leading corporations all over the world. Dr. Sriram has co-authored or authored nearly 250 papers, books, and reports, including several books. He was a founding co-editor of the International Journal for AI in Engineering. In 1989, he was awarded a Presidential Young Investigators Award from the National Science Foundation, U.S.A. Dr. Sriram is a fellow of the American Society of Mechanical Engineers, a fellow of the American Association for the Advancement of Science, a senior member of the Institute of Electrical and Electronics Engineers, a member (life) of the Association for Computing Machinery, a member of the Association for the Advancement of Artificial Intelligence, and a member of the American Medical Informatics Association.

Imré Horváth is a Professor of Computer Aided Design Engineering in the Faculty of Industrial Design Engineering, Delft University of Technology, the Netherlands. He completed his university studies at the Technical University of Budapest, Hungary. He earned M.Sc. titles in mechanical engineering and engineering education. He initially worked with the Hungarian Shipyards and Crane Factory for more than six years, after which he specialized in computer-aided design and engineering. Prof. Horváth has held various faculty positions at the Technical University of Budapest, and earned doctoral titles, including that from the Hungarian Academy of Sciences. His research has focused on issues concerning geometric and structural modeling, knowledge-intensive software tools, advanced design support of conceptual design, and virtual reality technologies and applications. Prof. Horváth has published more than 30 journal articles and more than 150 conference papers, has won 4 best-paper awards, and is serving 3 journals as permanent editor and many more as a guest editor. He initiated the International Symposia on Tools and Methods of Competitive Engineering (TMCE) and has been its general chairman for 12 years. He has also served the Executive Committee of the CIE Division of the American Society of Mechanical Engineers for 7 years, also as Chair of Division, and is a fellow of ASME. I presented several invited and keynote talks at international conferences. As an educator he is interested in advanced support of product design, in particular that of conceptual design, integrating research into design education, and teleconferencing-based active learning.

Alain Bernard graduated in 1982, received his Ph.D. in 1989, and was associate-Professor, from 1990 to 1996 in Centrale Paris. From Sept. 1996 to Oct. 2001, he was Professor in CRAN, Nancy I, in the "Integrated Design and Manufacturing" team. Since 0ct. 2001, he is Professor at Centrale Nantes and Dean for Research. He is in IRCCyN in the "System Engineering – Product-Performance-Perception" team. His research topics are KM, system modeling, interoperability, performance evaluation, virtual engineering, and rapid product development. He supervised more than 20 Ph.D. students, recently on extended enterprise modeling, simulation and performance evaluation. He published more than 250 papers in refereed international journals, books and conferences. He is chairman of WG5.1 of IFIP (Global Product Development) and vice-chairman of CIRP STC Design.

Ramy F. Harik is an Assistant Professor at the Lebanese American University (LAU) and holder of a Ph.D. specialized in CAD/CAM in Industrial Engineering from Henri Poincaré University, France (2007). He holds a joint appointment in Mechanical and Industrial engineering and teaches courses in the areas of design and manufacturing. In 2011, he was a Fulbright visiting scholar at the Computational Design and Innovation Lab at Purdue University. His research is in the areas of CAD geometric modeling, feature recognition, manufacturing identification and ergonomics. He is a member of ASME, IIE & CAD'xx. Since 2008, he has been a member of the program advisory board of the annual CAD conference and exhibition and was appointed as the conference co-chair of the CAD'10 edition.



Wei Gao is currently a Ph.D. student in the School of Mechanical Engineering at Purdue. Mr. Gao received his bachelor's in Mechanical engineering in 2009 from the University of Shanghai for Science & Technology. His graduate research involves origami-inspired design and optimization on morphable and self-assembly engineering system, robotic analysis, simulation and actuation, and developing tools that can help synthesize foldable and reconfigurable mechanisms/robotic forms early in the design process.