Western University
Scholarship@Western

Digitized Theses | Digitized Special Collections

2011

COMPUTER-SUPPORTED COLLABORATIVE KNOWLEDGE BUILDING IN ENGINEERING DESIGN

Hongmei Lu

Follow this and additional works at: https://ir.lib.uwo.ca/digitizedtheses

Recommended Citation

Lu, Hongmei, "COMPUTER-SUPPORTED COLLABORATIVE KNOWLEDGE BUILDING IN ENGINEERING DESIGN" (2011). *Digitized Theses*. 3296.
https://ir.lib.uwo.ca/digitizedtheses/3296

This Thesis is brought to you for free and open access by the Digitized Special Collections at Scholarship@Western. It has been accepted for inclusion in Digitized Theses by an authorized administrator of Scholarship@Western. For more information, please contact wlswadmin@uwo.ca.

COMPUTER-SUPPORTED COLLABORATIVE KNOWLEDGE BUILDING IN ENGINEERING DESIGN

(Spine title: Computer-supported CKB in Engineering Design)

(Thesis format: Monograph)

by

Hongmei Lu

Graduate Program in Mechanical and Material Engineering
Faculty of Engineering

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Engineering Science

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

© Hongmei Lu 2011

THE UNIVERSITY OF WESTERN ONTARIO
School of Graduate and Postdoctoral Studies

CERTIFICATE OF EXAMINATION

Supervisor	Examiners
Dr. Ralph O. Buchal	Dr. Michael D. Naish
	Dr. O. Remus Tutunea-Fatan
	Dr. Lu Xiao

The thesis by

Hongmei Lu

entitled:

Computer-supported Collaborative Knowledge Building in Engineering Design

is accepted in partial fulfillment of the
requirements for the degree of
Master of Engineering Science

Date

Chair of the Thesis Examination Board

Abstract

Engineering design is defined as a process of devising a technical system, component, or process to satisfy desired needs. Collaborative engineering design (CED) is a knowledge-intensive process that involves multidisciplinary people working jointly, sharing resources and outcomes, and building new knowledge while solving problems. People need to collaborate synchronously or asynchronously, either in the same place or distributed geographically. This thesis proposes that engineering design can be modeled not only as a process of knowledge transformation, but as a process of collaborative knowledge building (CKB). CKB is a goal-driven collaborative process of generating and refining ideas and concepts of value to the community. Properly applied and supported, CKB has the potential to improve both learning and design outcomes resulting from collaborative design projects. Existing collaboration tools have evolved without a clear understanding of designers' needs, even though a portion of the required functionalities has been achieved separately. This thesis proposes an integrated CKB-orientated model for collaborative engineering design, incorporating the key elements of Stahl's CKB model, Lu's ECN-based collaborative engineering model, Nonaka's knowledge creation theory, and Sim and Duffy's model of a design activity. Based on the model, a set of specific requirements for collaboration tools are presented and some functionalities not existing currently are identified.

Keywords

Collaborative Engineering, Collaborative Design, Collaborative Engineering Design, CED, CSCW, CSCD, Computer Support, Collaborative Knowledge Building, CKB, SECI

Acknowledgments

I would like to express my gratitude to those who provided me with great help and encouragement during the course of this thesis.

I am deeply indebted to my supervisor Dr. Ralph O. Buchal for his continuous and patient guidance, encouragement, expertise, and valuable contribution to this thesis. While revising the thesis time after time, without his numbers of comments and suggestions this thesis could not be finalized.

I would also like to express my sincere gratitude to my husband, Qiyue, who took care of almost all the housework in the last two months while I worked intensively on the thesis. Words cannot just describe how valuable his unselfish support for completion of this work.

Also, I would like to take this opportunity to thank two of my classmates, Jianming Ma and Xiangjing Li, who gave me generous help during my graduate courses studying.

Table of Contents

List of Tables

List of Figures

Nomenclature

CED	Collaborative Engineering Design
CSCD	Computer Supported Collaborative Design
CSCW	Computer Supported Cooperative Work
CKB	Collaborative Knowledge Building
CIRP	College International pour la Recherche en Productique
ECN	Engineering Collaboration via Negotiation
PDM	Product Data Management
PLM	Product Lifecycle Management
SECI	Nonaka's knowledge conversion modes: Socialization, Externalization, Combination and Internalization

1 Introduction

Engineering design is a process to create a new artifact or system to meet desired needs. It is a process in which the basic sciences and engineering sciences are applied to convert resources optimally to meet a stated objective. In the past, engineering design emphasized the coordination of individuals working on separate tasks. More recently, collaborative approaches to engineering design have become increasingly necessary. It is crucial to achieve effective and efficient collaboration in engineering design, with improved support from modern information technology.

1.1 The Need for Collaboration in Engineering

There are two major reasons to require collaboration in engineering design. One is that increasingly complex systems demand effective collaboration among multidisciplinary designers [1], [2] because it is impossible for individuals working separately to accomplish the design tasks; the other is that market globalization requires companies to complete the product development process in the shortest period and with the highest quality. Designers must collaborate closely with suppliers, manufacturing partners and customers to speed up the development cycle.

For example, Airbus has 39 sub-contractors and vendors from multiple European countries involved in its development program. It was reported that around 26% of project meetings of Airbus contractors required international partners and there are more than 400 one-day trips taken by its engineers to collaborate with stakeholders each day. On the average, 49% of the Airbus engineers' daily activities are spent on discussions and meetings with other stakeholders [3].

Marsh [4] observed that designers spent an average of 24% of their working time to acquire and disseminate information, and the majority of this information was obtained from personal contacts rather than formal sources; Stewart [5] estimates that probably only 20% of an organization's knowledge is effectively used; Vijaykumar [6] concludes that about 50% of the old queries were answered by colleagues. A study of designers in German industry, conducted by Badke-Schaub and Frankenberger [7], shows that 88% of critical actions are determined while interacting with colleagues, although more than 80% of their working time they work individually.

All of the above indicates designers spend a lot of time on the interactions with their colleagues or partners (the collaboration happens in both intra-company and inter-company); also, the information and knowledge derived from their co-workers is abundant. Moreover, some organizations do not manage knowledge effectively.

Other group interactions, such as team learning, discussing, negotiating, evaluating, making decisions, and so on, are essential elements of collaboration and shared knowledge creation. These are the reasons why some commercial technologies and research areas have been very active in recent years and also progressed dramatically [8], [9], such as Computer Supported Cooperative Work (CSCW), Computer Supported Collaborative Design (CSCD), Group Support System (GSS), and other projects providing collaborative and distributed solutions.

Hence, it is critical to establish a shared workspace to facilitate collaboration among distributed teams, both synchronously and asynchronously. The workspace must allow people to efficiently exchange or share resources, capture and record new knowledge created in design process, and ensure that people have access to the knowledge they need, when they need it.

1.2 Benefits of Collaborative Computer Tools

Effective collaborative computer tools enable geographically distributed designers to collaborate, both synchronously and asynchronously. These tools allow participants to conveniently share all information and knowledge and communicate, regardless of time or place.

Effective collaborative computer tools help manage and leverage an organization's technological knowledge and information. Traditionally, some design information is stored in a physical form, like books, manuals, and paper-based drawings in library, etc. Other information is communicated informally, e.g., conversations among colleagues, suppliers, etc., and the process is poorly recorded and unorganized. Newcomers or novices must rediscover much of this information by repeating the same process. However, storing all of the project-related files in an online repository allows people to freely access all the information of the project and previous experience from different workplaces at any time. Designers will learn experiences and lessons from the old projects.

Effective collaborative computer tools can help designers solve conflicts in the early design stage and decrease product development lead-time and manufacturing costs [2]. “Teamwork” paybacks and “task-work” paybacks are two types of benefits that industry can expect from successful applications of collaborative engineering. Improvements related to teamwork can be found in better communication among team members, common understanding, collaborative generation of new ideas, faster decision-making, and increased employee morale and responsibilities. Positive effects on task-work include improved product innovation, better technology integration and utilization, shortened development cycle, and lower development and manufacturing cost [2]. For instance, associated with their global product development projects, Hewlett-Packard reported a 135% RoI (Return on Investment) after one month and 240% RoI after three months in travel costs alone. Canon achieved significant reductions in design iterations, total costs, and lead-time by using a collaborative design tool to develop laser printers [10].

Certainly, not everything will be solved after installation of collaboration tools. On one hand, the existing software is not capable enough to meet the demands of collaboration; on the other hand, many companies have the information and knowledge access problems, decision independence problems, management problems, and agent access problems, etc. (see more detail in [11]). Many problems need to be solved in this field to ensure effective collaboration.

1.3 The Objectives of This Thesis

Effective collaboration is very important in engineering design. A better understanding of the essence of collaborative engineering design (CED) is needed in order to develop effective computer support tools. The objectives of this thesis are as follows:

1) Survey the literature from multiple disciplines to better understand the essence of CED, including design process, generic design activities, learning in design, collaboration science, collaborative knowledge building (CKB), and so on.
2) Study knowledge classifications and engineering knowledge representations to provide support for knowledge management in collaborative engineering.
3) Establish a CKB-oriented model for collaborative engineering design, and then propose detailed requirements of computer support for CED.
4) Review the existing collaborative technologies and make it clear if they can meet all the requirements of CED.
5) Identify required functionalities that have not been achieved by current computer tools.

1.4 Main Contributions of This Thesis

The main contributions of the thesis are concluded as below.

1) The thesis compares and contrasts the relevant research literature from different fields including engineering design, social science, collaboration science, information technology, education and business management.
2) The thesis proposes an integrated CKB-orientated model of CED for both engineering design and learning. The model integrates and extends key ideas and elements from different fields: Stahl's and Singh's CKB model of collaborative learning; Lu's ECN-based model of collaborative engineering; Nonaka's SECI model of organizational knowledge creation; and Sim and Duffy's model of engineering design activities. The purpose of the model is to describe and explain how knowledge is created in collaborative engineering design.
3) The thesis identifies general and detailed requirements for collaboration tools based on the integrated CKB-oriented model of CED, and compares them to the capabilities of existing collaboration tools. Gaps between requirements and existing capabilities are identified.

1.5 The Structure of This Thesis

This thesis is divided into seven chapters, and is organized in the following way.

Chapter 2 first reviews definitions of engineering design and the common design process, and then describes design as a knowledge transformation process. Finally it discusses the recurring generic activities in design and collaboration.

Chapter 3 first clarifies the meaning of terms coordination, cooperation and collaboration, then surveys the different views of CED. It also reviews and compares technology-oriented and social science-oriented research approaches. Finally, the ECN-based collaborative engineering approach of Lu et al. is described.

Chapter 4 first reviews the application of knowledge management in engineering, and then identifies types of knowledge, engineering knowledge classifications and representations respectively.

Chapter 5 compares and contrasts two existing knowledge building theories, Nonaka's organizational knowledge creation and Scardamalia's knowledge building theory. Existing views of learning in design are presented, and an argument is made that design and learning are intertwined and CKB has the potential to integrate activities of both in a single collaboration model.

Chapter 6 proposes an integrated knowledge building-oriented CED model and illustrates it in detail. This model incorporates the key elements of Stahl's CKB model, Lu's ECN-based collaborative engineering model, Nonaka's knowledge creation theory, and Sim and Duffy's model of a design activity. This is the main contribution of the thesis.

Chapter 7 proposes a set of general requirements of computer support for CED evolved from the integrated model in Chapter 6. After existing collaborative tools are reviewed, specific functional requirements for CED are identified and several additional functions needed to support CED are identified and described in detail.

2 Engineering Design Process

This chapter surveys accepted definitions of the engineering design process, then presents the view of design as a knowledge transformation process, followed by the introduction of recurring activities both in design and collaboration.

2.1 Definition of Engineering Design

Ertas and Jones define engineering design as "the process of devising a system, component, or process to meet desired needs. It is a decision making process (often iterative) in which the basic sciences, mathematics, and engineering sciences are applied to convert resources optimally to meet a stated objective [12]." Eder and Hosnedl describe design as "a process of formulating a description for an anticipated process system and/or an object system that is intended to transform an existing situation into a future situation to satisfy needs [13]." Similar definitions can be found in other design references and textbooks, e.g., Dieter and Schmidt [14].

Different companies and industries have their own design processes. However, most design processes are very similar, with minor differences. Dieter and Schmidt describe Morris Asimow's engineering design process consisting of the following seven phases: 1) Conceptual Design; 2) Embodiment Design; 3) Detail Design; 4) Planning for Manufacture; 5) Planning for Distribution; 6) Planning for Use; 7) Planning for Retirement of the Product [14]. Most researchers emphasize the first three phases of Asimow's design process. Dieter and Schmidt expand these three phases into eight distinct stages as shown in Figure 1. Other authors, including Paul and Beitz [15], Hubka and Eder [16], and Ullman [17], describe the design process in a similar way.

The stages of a systematic design process are not rigidly fixed, but rather provide guidance to designers. The process includes iteration and feedback loops at every level. Design procedures are not fixed and the process can be decomposed into generic activities such as defining, generating, evaluation, deciding, synthesis, etc., proceeding interactively and repetitively. In each activity a rational action is executed by designer(s) to achieve a desired goal [18].

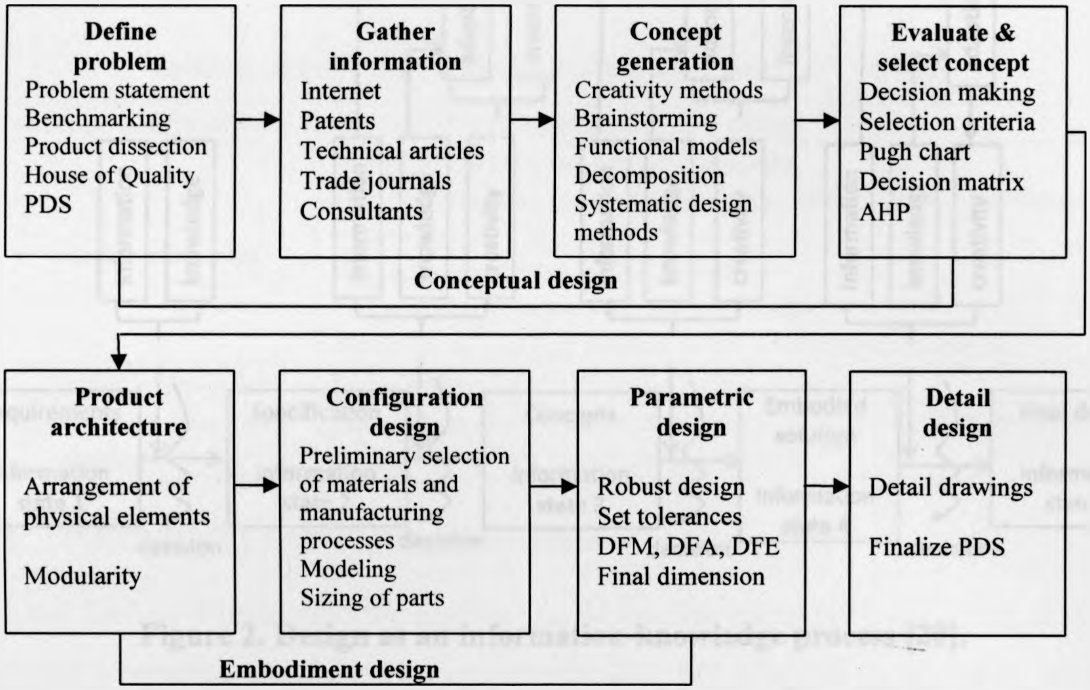


Figure 1. Sequential engineering design process [14].

**Notes: PDS- Product Design Specifications AHP-Analytic Hierarchy Process*

DFA- Design for Assembly DFM- Design for Manufacture DFE- Design for the Environment

2.2 Engineering Design as Knowledge Transformation

Eder and Hosnedl [13] and Sim and Duffy [18] describe design as a knowledge/information transformation process. The input knowledge is what is known at the beginning, and the output knowledge is what is learned about the solution. Reddy et al. [19] propose the concept of "artifact theory" to interpret the knowledge-creating process that unfolds during the product design process: "To reflect the knowledge building aspect of the design process, we extend this view and propose that design is a process of constructing a theory of the artifact, not merely constructing a manufacturable description." In other words, designers are creating knowledge. The artifact theory is the output knowledge of the design process.

Hicks et al. [20] describe the design process as shown in Figure 2.

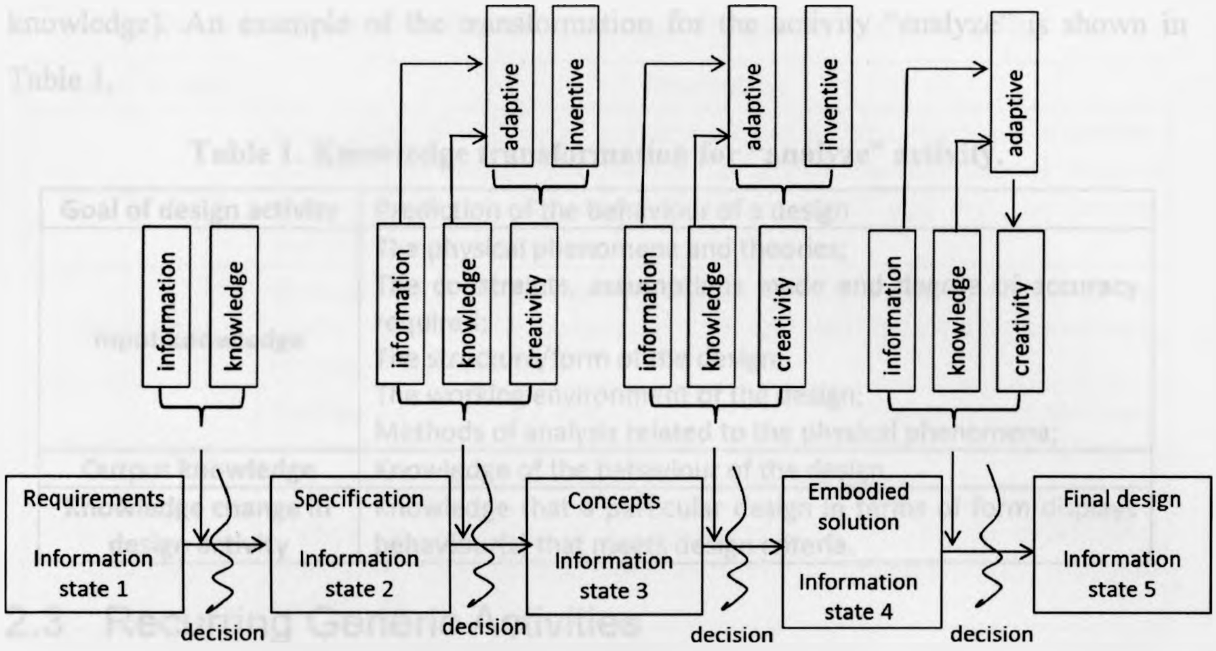


Figure 2. Design as an information-knowledge process [20].

The diagram illustrates design as an information and knowledge transformation process. During the process new knowledge is generated and used to make decisions and carry forward the design procedure. In addition, Hicks et al. believe that two types of creativity occur: adaptive creativity (adapt and extend existing knowledge to a new situation) and inventive creativity (purely original).

Sim and Duffy [18], [21] represent generic design activities as goal-directed knowledge transformation processes as shown in Figure 3.

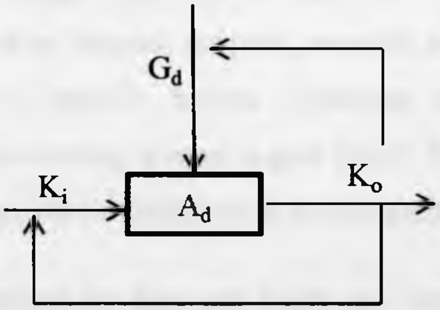


Figure 3. Model of design activity [21].

K_i refers to input knowledge (existing knowledge); **A_d** stands for design activity; **G_d** represents the goal of the design activity; **K_o** represents output knowledge (new knowledge). An example of the transformation for the activity “analyze” is shown in Table 1.

Table 1. Knowledge transformation for “analyze” activity.

Goal of design activity	Prediction of the behaviour of a design
Input knowledge	The physical phenomena and theories; The constraints, assumptions made and degree of accuracy required; The structure/form of the design; The working environment of the design; Methods of analysis related to the physical phenomena;
Output knowledge	Knowledge of the behaviour of the design
Knowledge change in design activity	Knowledge that a particular design in terms of form displays behaviour(s) that meets design criteria.

2.3 Recurring Generic Activities

Sim and Duffy [18] surveyed the engineering design literatures [15], [17], [22], [23] and extracted a common set of 27 recurring generic activities involved in the design process. Table 2 lists some of the most frequently encountered activities based on Sim and Duffy's ontology.

Briggs et al. [24] identified six fundamental and recurring patterns of collaboration used typical in group activities involving idea generation, problem solving and decision-making. The six patterns are shown in Table 3. A new research field known as *collaboration engineering* has emerged to develop tools and methods to facilitate these patterns of collaboration. Briggs and his colleagues have proposed the concept of *ThinkLets*, which are defined as “named, scripted, reusable, and transferable collaborative activities that give rise to specific known variations of the general patterns of collaboration among people working toward a goal [24].” ThinkLets are able to support collaboration by providing guidance similar to an expert human facilitator or coach [25].

The generic activities identified by Sim and Duffy are very similar to the patterns of collaboration identified by Briggs et al. These generic activities are found in most human problem-solving and critical thinking processes, and are not unique to design.

Table 2. The descriptions of some high frequency activities~~.~~ [18].

Design activity	Description
Decomposing	Break down the complex problem, object or task into a set of smaller problems to reduce complexity. Designers may decompose task in structure way, function-oriented way, or aspect way (e.g., mechanical and electrical, hydraulic)
Generating	Generate concepts to meet the requirements of customers or perceived needs. Usually, the generated concepts are described qualitatively instead of quantitatively.
Synthesising	Configure entities of a domain to construct a realisable system structure to meet requirements. Specifically, combine concepts or parts into a whole, e.g., components to sub-assembly, sub-assembly to assembly.
Analysing	Predict the behaviour of a design structure by analysis techniques: qualitative techniques, approximate techniques and detailed techniques.
Decision making	Select the best solution among several alternative possibilities based on some criteria.
Evaluating	Assess if the design satisfies the objectives, e.g., check to make sure the system won't fail; compare concepts to find the best solution, etc.
Information gathering	Identify and gather relevant up-to-date information to support the design task.
Selecting	Select from several options, such as choose a working principle for equipment, and pick a component from a catalogue.

Table 3. Patterns of Collaboration~~.~~[24].

Pattern of Collaboration	Description
Generate	Moving from having fewer concepts to having more concepts
Clarify	Moving from less to more shared understanding of the concepts under consideration and of the words and phrases used to express them
Reduce	Moving from having many concepts to a focus on fewer concepts worthy of further attention
Organize	Moving from less to more understanding of the relationships among concepts
Evaluate	Moving from less to more understanding of the relative value of the concepts under consideration
Build consensus	Moving from having fewer to having more group members who are willing to commit to a proposal

3 Collaborative Engineering Design

This chapter first clarifies the meaning of terms coordination, cooperation and collaboration, then surveys the different views of CED. It also reviews and compares technology-oriented and social science-oriented research approaches. Finally, the ECN-based collaborative engineering approach of Lu et al. is described.

3.1 Defining Collaboration

Most researchers agree that coordination, cooperation and collaboration are three distinct levels of human collective endeavors. Lu et al. [2] distinguish coordination, cooperation and collaboration in terms of participants, resources, goals and task structure as shown in Table 4.

Table 4. Collective human endeavor characteristics [2].

	Participants	Resource	Goal	Task structure
Coordination	Large community	Limited and exchanged	Multiple and competing	Pre-defined, same layer in hierarchy, uni-direction
Cooperation	Mid-size group	Limited and shared	Multiple and Private	Pre-defined, across layers in hierarchy, bi-direction
Collaboration	Small team	Limited, shared, complementary	Single and common	Undefined, non-hierarchical, multi-direction

Coordination is the most basic level of collective endeavor. It occurs among organizations or different departments in one organization. It is influenced by regulations that inform each department as to when and how it must act. Departments have different functions and don’t share resources except exchanging limited information. Effective coordination forms an integrated and harmonious body and increases efficiency. Therefore some researchers say that “coordination is about efficiency” [26].

Cooperation is the second level of collective endeavor. It usually happens in mid-size groups, in which participants share some resources and methods. Group members have their own subtasks and separate goals, but work together reciprocally or in compliance for mutual benefit.

Thus it can be seen that working together through coordinated or cooperative activities generally provides benefits. However the general roles of individual participant or a unit stay the same and the work itself does not vary much.

There are times when cooperation and coordination are not enough. For instance, if current groups do not have good means to serve a new customer group, it may be necessary to join into a collaborative relationship with another group by forming a team. In Merriam-Webster Online Dictionary [27], collaborate is defined as "work jointly with others or together especially in an intellectual endeavor". Briggs et al. [24] define collaboration as "joint effort towards a group goal". Lu et al. [2] believe "collaboration aims at achieving a common goal and collective results" that could not be accomplished by individuals alone. They point out that in collaborative teams, besides sharing resources and outcomes, the most important thing is to share a common goal. Alberts et al. [28] describe collaboration as actors working together and actively sharing "data, information, knowledge, perceptions or concepts" to achieve a common purpose.

Noble [29] describes collaboration from a cognitive perspective, focusing on the problem-solving aspects of group work. He defines it as "the mental aspects of group problem solving for the purpose of achieving a shared understanding, making a decision, or creating a product." Michael Schrage argues collaboration is not about agreement, but about creation. In his book "Shared Minds", he states that: "collaboration is the process of shared creation: two or more individuals with complementary skills interacting to create a shared understanding that none had previously possessed or could have come to on their own. Collaboration creates a shared meaning about a process, a product, or an event [30]." Therefore, from Schrage's point of view, the goal of collaboration is not to establish a positive relationship between partnering groups (which coordination does), but the pursuit of a specific result.

Collaboration relies on both cooperation and coordination of efforts, but goes far beyond these two working relationships. It is about using information to create something new, so a great deal of time and communication is required while collaborating. During collaboration, differing views and conflicting ideas are discussed, negotiated and discoursed, and then merged into something that was previously unimaginable. "Unlike coordination, collaboration seeks divergent insight and spontaneity, not structural harmony; and unlike cooperation, collaboration thrives on differences and requires the sparks of dissent [26]."

Designing complex systems requires collaboration among multidisciplinary stakeholders who coordinate to plan tasks, cooperate to resolve reciprocal dependencies, and co-construct knowledge to identify shared goals and solutions [31].

In this thesis, collaboration will contain the following attributes:

1) Two or more people working together, no matter where they are located;
2) Team members work collaboratively toward a shared team goal, which cannot be accomplished by working individually;
3) They share resources, knowledge and outcomes;
4) The result is creation of something new that meets the shared team goal.

3.2 Different Views of Collaborative Engineering Design

Collaborative engineering design is related to several different research fields or areas, including Concurrent Engineering, Collaborative Engineering, Computer Supported Cooperative Work (CSCW), and Computer Supported Collaborative Design (CSCD). Often, different names are used for similar or identical ideas. The work in different fields has much in common, but has evolved somewhat in parallel, and in some cases the cross-fertilization between the fields is weak.

Turino defines Concurrent Engineering as "a systematic approach to the integrated, concurrent design of products and their related processes, including manufacturing and support [32]". It is intended to cause the product developers from the very outset to consider all elements of the product life cycle, from conception to disposal, including cost, schedule, quality and user requirements [14]. However, Kamrani [33] defines "the integrated, concurrent design of products and related processes, including manufacturing, product service, and support" as Collaborative Engineering. It is a different name for essentially the same idea.

The ECN working group of CIRP (College International pour la Recherche en Productique) defines Collaborative Engineering as a new socio-technical engineering discipline, which “facilitates the communal establishment of technical agreements among a team of interdisciplinary stakeholders, who work jointly toward a common goal with limited resources or conflicting interests [2]”. The International Journal of Collaborative Engineering (IJCE) [34] was established to publish research in this new area. IJCE defines Collaborative Engineering as a discipline that “studies the interactive process of engineering collaboration, whereby multiple interested stakeholders resolve conflicts, bargain for individual or collective advantages, agree upon courses of action, and/or attempt to craft joint outcomes which serve their mutual interests.”

Another related field is Computer Supported Cooperative Work (CSCW). Wilson [35] defined CSCW as a generic term, “which combines the understanding of the way people work in groups with the enabling technologies of computer networking, and associated hardware, software, services and techniques.” Research in CSCW seeks to understand how people and organizations interact with one another, and to integrate this understanding with the development of computer based tools to support real world settings.

Groupware, also referred to as collaborative software, workgroup support systems or simply group support systems, is software designed to help people involved in a common task achieve their goals. Many people regard CSCW and Groupware as the same thing; however, according to Shen et al. [8], the term CSCW is widely used in the research community while Groupware is used more in commercial software products.

Computer Supported Collaborative Design (CSCD) studies the application of CSCW technologies in design, especially engineering design and software design. Sprow [36] suggested that CSCD can also be called “Cooperative Design, Concurrent Design, or Interdisciplinary Design”. Shen et al. believe CSCD is not just CSCW in design, but an application of “collaborative engineering” to product design [8]. The most widely applied CSCW technologies in collaborative design systems include groupware technologies, which facilitate interactions among design team members, and context awareness technologies, which enhance coordination among team members. A fundamental task of CSCD is to develop computer systems to support group interactions amongst geographically distributed participants.

A review of existing CSCD systems (see Section 3.3.1) shows that most tools emphasize CAD modeling, simulation and optimization software, engineering database sharing and exchange, agent-based collaborative design, PLM, and project management, etc. Researchers in CSCD put more effort into computer tools, while people in collaborative engineering put more emphasis on collaborative activities including discourse, negotiation and decision-making. Furthermore, the use of computers does not appear in the definitions of collaborative engineering.

In this thesis, Collaborative Engineering Design (CED) is defined as *a knowledge-intensive process of devising a technical system, which involves multidisciplinary people working jointly, sharing resources and building new knowledge while solving problems.* Collaborative engineering design includes a set of human-centered socio-technical activities, which can maximize the gain of integration of the “social teamwork by groups” and the “technical task-work by individuals” [2]. What’s more, the essence of collaborative engineering design needs more in-depth study and the better it is understood, the more support can be obtained from the technologies.

3.3 Technology-oriented versus Social Science-oriented Approaches

This section reviews technology-oriented approaches, social science-oriented approaches and socio-technical-oriented approaches to collaborative engineering.

3.3.1 Technology-oriented Approaches

Technology-oriented approaches focus on developing computer tools to support task-work in collaborative engineering. Most of these tools fall into one of two categories: general groupware tools and CAD-oriented collaboration tools.

Hundreds of general groupware products now exist in the marketplace, and more appear monthly. Mittleman et al. [37] surveyed over 250 existing groupware tools, and identified common collaboration technologies (see detail in Section 7.2.1), including joint authoring, online meeting, file management, information access technologies, and so on. However, none of these general groupware tools can deal with CAD models.

On the other hand, CAD-oriented tools emphasize sharing and collaborating on geometric models. Li and Qi [9] identified three types of CAD-oriented tools: visualization-based collaborative systems, co-design collaborative system and concurrent engineering-based collaborative system. The first group allows users to view, mark-up, measure or make cross-sections and assemble models via internet or intranet. Examples of these tools include Cimmetry Systems AutoVue, Actify SpinFire, SolidWorks eDrawing, RealityWave ConceptStation, and Autodesk Streamline. The second group provides more interactive capabilities to support synchronous co-modeling/co-modification design, and asynchronous assembly-based design, such as real-time data sharing, to jointly view, annotate and edit a model. These kind of tool includes CollabCAD, IX SPeeD, Alibre Design, OneSpace, etc. The major features of the third group include integrated service tools to optimize design activities, such as manufacturability analysis, manufacturing cost evaluation, CAE simulation, etc.; also, it can facilitate the communications and data transfer across the organization boundaries. From Li and Qi's work, most of the concurrent engineering-based collaborative systems are still at the research stage.

Product data management (PDM) and product lifecycle management (PLM) are also important capabilities in collaborative design. PDM/PLM systems are similar to document management systems, but with additional functionality specifically for managing CAD data.

There is a lack of integration of general groupware and CAD-oriented tools, and it is not clear that these technologies have been developed with appropriate consideration of human behavior and social dynamics.

3.3.2 Social Science-oriented Approaches

Collaborative engineering can be regarded as the practical application of collaboration sciences in the engineering domain in terms of understanding human aspects of collaboration and interactions in teamwork [2]. The relevant fields include communal communication, human collective behavior, collective decision-making, organizational science, social cognition, and social choice, etc. The knowledge and theories developed in these fields can be associated and integrated to support collaborative engineering.

Monge and Contractor [38] studied various configurations of team communication networks to identify the optimal mechanism for information exchange in community-based actions. Many researchers have studied collective decision-making [39][40]. Organizational science [41] recommends that people engage in collaborative endeavors as members of a purposeful "team", and align their decisions and actions with shared team goals ahead of their individual interests. Social cognition [42] studies how people understand, influence and connect to others in social settings, and how this influences their own decisions. On the other side, social cognition also studies how individual perceptions and behaviors influence group decisions. Social choice investigates how individual intents can be appropriately considered to form a group intent which is acceptable to every group member [43].

All of the above disciplines are good foundations for effective collaborative engineering in term of human collaboration. While they are often short of rigorously validated studies of practical projects, the theories and frameworks from social science research provide useful guidance.

3.3.3 Socio-technical-oriented Approaches

Although information technologies can enable teams to simultaneously discuss and handle shared design representations, technological systems alone are not able to provide an answer to the complicated problems proposed by teams. Carey and Kacmar [44] found that the introduction of technical systems produces behavioral and operational changes, rather than the desired improvements in productivity or quality. They found that existing technologies increase task complexity, and they doubt that using such technologies within an information-rich context is a good choice.

Socio-technical research argues that both technical and social subsystems within an organization should be optimized together to maximize performance. In terms of the technical subsystem, the technologies, procedures, and methods employed by an organization must be considered. In terms of the social subsystem, team members, their communication, interactions, and relationships with one another and the wider organization all play significant roles. Hammond et al. [45] reviewed some literature on collaboration of engineering design process and proposed a conceptual model for distributed engineering collaboration based on socio-technical theory. The model indicates bandwidth of interactions is reduced in the distributed communication contexts, which causes two major kinds of changes in group interactions. On one side, participants must seek to maintain a comfortable level of communication by using compensating mechanisms, such as limiting the amount of data considered or increasing mental effort; on the other side, changes in the social presence perceived by participants influence timing, amount and content of interaction, etc. Hence, an optimized socio-technical approach must provide designers with appropriate technological support, and at the same time, enable management to mediate the group interactions by appropriate protocols, training, and methods for achieving virtual design team success.

3.4 ECN-based Collaborative Engineering Process

Engineering Collaboration via Negotiation (ECN) is a research hypothesis developed by the CIRP/ECN-WG (working group). It is a "guided teamwork process which a collaborative engineering team can employ to achieve a task-work agreement". An *"Interaction→Perspective→Preference→Agreement"* mechanism is applied to attain Participative Joint Decisions (PJD) that underline the task-work assignments [2]. Lu et al. propose an ECN-based collaborative engineering process in which participants "employ a dynamic, socio-technical co-construction process to collaborate with each other reciprocally to reach participative joint decisions…" The emphasis of ECN is on sharing perspectives to reach a common understanding, followed by discourse and negotiation of individual preferences to reach a consensual agreement.

An ECN-based collaborative engineering process consists of four stages as shown in Figure 4. Each stage is informed by appropriate theories from different disciplines.

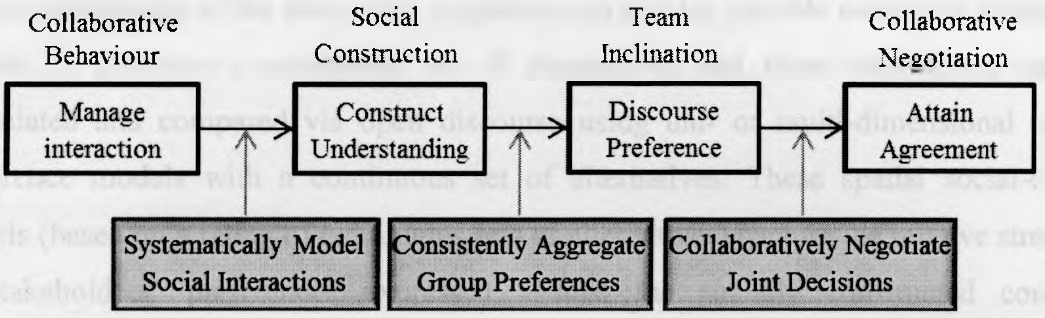


Figure 4. ECN-based collaborative engineering process [2].

Stage 1: Manage social interactions

In this initial stage, organizational behavior theory suggests the modeling of team social interactions and collaborative behaviors as an "organizational man" working in a "small and induced" team [46]. An organizational man seeks to satisfy all stakeholders' preferences (satisfactory or good enough for all), whereas a traditional economic man attempts to optimize (find the best alternative). In this phase, people usually choose team members, develop clear team goals, clarify resources and constraints, establish a baseline interaction procedure and behavior criteria, etc. Collaborative behaviors of the team must be carefully organized to ensure that team members engage in the social interactions. Stakeholders' perspectives can be changed in the social interactions.

Stage 2: Construct common understanding

At the start of a project, stakeholders have their own diverse and possibly conflicting understandings of the task-work. The different viewpoints have to be "calibrated, eliminated or minimized" as much as possible. In light of theory of social cognition, "minds can be shaped by others" and individuals change their preferences during social interaction, but this process must be "properly managed and strategically guided" [47] to achieve a common understanding. Social construction theory is applied to systematically guide stakeholders toward establishing a common understanding.

Stage 3: Discourse group preference

Once common understanding is obtained, it can be used as an anchor for stakeholders to consistently and fairly discourse and compare their dissimilar preferences to attain a single group preference. This is a challenge in collaborative engineering. Based on the domain knowledge of the task-work, engineers can employ suitable numerical simulation models to generate a continuous set of alternatives and these alternatives can be formulated and compared via open discourse using uni- or multi-dimensional spatial preference models with a continuous set of alternatives. These spatial social-choice models (based on a rating of continuous sets of alternatives) capture the relative strengths of stakeholders' preferences expressed against the socially constructed common understanding of task-work at Stage 2. Then, in the next stage, interpersonal comparisons of preference strengths can be implemented via negotiations.

Stage 4: Obtain team agreement

Given a robust group preference established by all of the members in the above stages, stakeholders can now directly and proactively participate in collaborative negotiations to make joint decisions that lead to a consolidated team agreement for the task-work at hand. Owing to the carefully organized team membership and well managed social interactions in Stage 1, the socially co-constructed common task-work understanding in Stage 2, and the consistently established group preference in Stage 3, the collaborative negotiation activities at this stage can be systematically supported and guided by negotiation analysis techniques from the decision sciences. This completes the ECN-based collaborative engineering process, resulting in a Participative Joint Decision.

According to ECN, collaborative engineering is a process of designing a system or artifact under the collaboration of multidisciplinary stakeholders through a series of activities, such as sharing resources, analysis, evaluation, negotiation, making-decision, etc. The focus here is negotiation and decision making. Knowledge creation, however, is not explicitly addressed in the ECN model.

4 Engineering Knowledge Management

This chapter reviews the application of knowledge management in engineering firstly, and then identifies types of knowledge, engineering knowledge classifications and representations respectively.

4.1 Introduction of Knowledge Management in Engineering

Knowledge Management (KM) has been recognized as an important part of organizations since the 1990s, and knowledge is considered as a competitive element for individuals, organizations and nations [48]. Although there is no universal KM definition yet, most are similar. Karadsheh et al. [49] list eight different definitions of KM. Most researchers agree that Knowledge Management includes capturing, discovering/acquiring, creating/generating/identifying, retrieving, sharing, reusing, evaluating and applying all information assets of an enterprise [50], [51]. The objective of KM is to improve the organizational innovation, reaction, efficiency and capability, in other words, to present the right knowledge to the right people at the right time.

Specifically, the common applications of KM system are: 1) codifying, sharing and transferring best practices within organization; 2) creating corporate directories, which is also referred to as internal expertise mapping; and 3) creating knowledge networks and then amplifying knowledge [52]. KM can increase the creativity and innovation of collaborative design. A KM system provides designers with a convenient way to capture and share knowledge [5].

Researchers recognize that KM is a very important component in engineering design, because design is a knowledge-intensive task and geographically distributed designers need to share knowledge resources. Zhen et al. [53] propose a novel distributed knowledge sharing model for spreading and sharing knowledge among engineers in collaborative product development teams; Ouertani et al. [54] state a standardized approach to trace and share product knowledge and key constructs to support traceability during the product development process; McMahon et al. [55] introduced some technologies applied to KM in engineering design, including human and organizational methods, groupware, information search and retrieval technology, knowledge organization, acquisition and structuring, etc.; Mezher et al. [56] built a knowledge management system for a real design firm, functions of which include creating knowledge, storing knowledge, updating obsolete knowledge according to feedback, and disseminating knowledge to all designers for current and future application. An expert system was built to capture tacit knowledge as well.

Knowledge creation is one of the key activities in KM. Eight of eleven papers surveyed by Karadsheh [49] clearly describe it, but several different terms are used, such as knowledge creation, knowledge building, and knowledge-generating. For example, Peachey and Hall [57] present that knowledge creation and generation focus on the different methods of generating new knowledge from both internal and external organization; Sun and Gao [58] claim that knowledge creation in the organization emphasizes creating new products or new ideas, enhancing ideas and services; Bouthillier and Shearer [59] portray creating new knowledge from different sources by either combining internal knowledge with other internal knowledge, or analyzing information to create new knowledge; Lei et al. [60] knowledge creation is based on both the human cleverness and existing knowledge.

4.2 Types of Knowledge

To design a knowledge management system, it is important to understand clearly the definitions of data, information, knowledge and classifications of knowledge.

4.2.1 Data, Information and Knowledge

Many researchers distinguish between data, information and knowledge [20] [49]. Table 5 compares three different views; the first two definitions are from the engineering domain and the third is from the business domain [61]. Although different terms are used in the definitions, the meanings are consistent: *Data* is usually described to be textual, either in a numeric or alphabetical form, with insufficient context on its own; *Information* is the combination of text and data to describe a fact in an either subjective or objective way, something that can be pointed to, found, lost, written down, accumulated, compared, and so on; *Knowledge* is something "broader, deeper and richer" [14] than the former two, which is created from the data and information through human's activities and then employed to solve problems. It is harder to transport, receive, assimilate or quantify (it is possible to have too much information but not too much knowledge [55]).

Table 5. Definitions of data, information and knowledge.

	Defined by Hubka and Eder [16]	**Defined by Dieter and Schmidt** [14]	**Defined by Kahn and Adams** [61]
Data	It is information without implied context.	It is a set of discrete and objective facts about events.	Data viewed as a set of facts.
Information	It is meaningful data which states assigned meaning of a static or dynamic phenomenon or thought.	It is data that has been treated in some way, and then it conveys a message.	Represented as categorized, reviewed and scrutinized data.
knowledge	It is meaningful information that is assigned based on the theoretical and practical context to a static or dynamic phenomenon or thought.	It is a mix of experience, values, contextual information and expert insight that provides a framework for evaluating and incorporating new experiences and information.	Knowledge is the result of merging information with practice, perspective and expression.

These types are correlated and can be converted to one another. Consider a document containing a table of numbers indicating product sales for the quarter. As they stand, these numbers are *data*. An employee reads these numbers, recognizes the name and nature of the product, and notices that the numbers are below last year's figures, indicating a downward trend. The data has become *information*. The employee considers possible explanations for the product decline (perhaps using additional information and personal judgement), and comes to the conclusion that the product is no longer attractive to its customers. This new belief, derived from reasoning and reflection, is *knowledge*.

The relationship between data, information and knowledge is shown in Figure 5 [62]. From Figure 5, the presumption of hierarchy is data, information, and knowledge (knowledge is the highest level). However, they cannot be isolated from each other, that is, there is a continuum from data to knowledge, without clear boundaries.

Tuomi [63] argues knowledge must exist before information can be elaborated and before data can be collected to form information. As such, "initial data" do not exist - even the most primitive piece of data has already been influenced by the thought or knowledge processes that cause its identification and collection. Tuomi also argues that existing knowledge becomes information when it is articulated, verbalized, and structured; information becomes data when the information is assigned a fixed representation and standard interpretation. Based on this argument, knowledge never exists outside of a knower since it is always indelibly shaped by one's needs and one's initial knowledge storage.

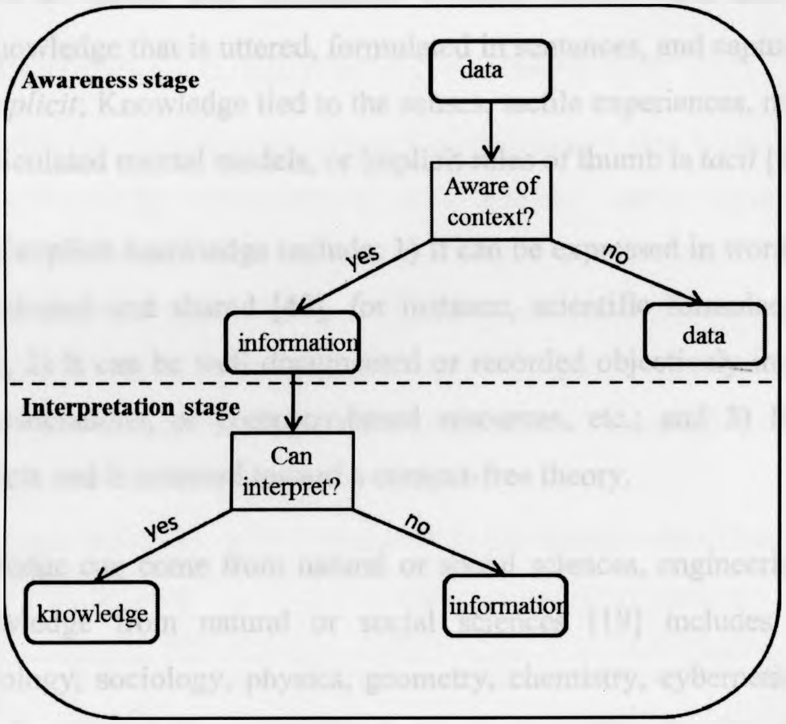


Figure 5. The relationship among Data, Information and Knowledge [62].

From Tuomi's viewpoint, once information is processed in people's mind it becomes knowledge; once the knowledge is articulated and presented in the form of text, words, graphics, or other symbolic forms, then it is converted into information. Hence, the inference of this argument is that knowledge only exists in people's mind; also, knowledge for some people could be information for other people and vice versa. If so, there is no meaning to strictly distinguish knowledge, information and data.

Since there is no separate term that includes all three of these types, the term "knowledge" will be used to include data, information and knowledge as defined above. In order to capture and use knowledge in a design process, it is must be clear what kinds of knowledge the designers acquire and where they can obtain it. Reasonable classification of engineering design knowledge is the prerequisite for knowledge supply to product designers; also it is the precondition of Knowledge Management in an organization.

4.2.2 Explicit and Tacit Knowledge

It is important to define and distinguish between *explicit* and *tacit* (or implicit) knowledge. Knowledge that is uttered, formulated in sentences, and captured in drawings or writing is *explicit*; Knowledge tied to the senses, tactile experiences, movement skills, intuition, unarticulated mental models, or implicit rules of thumb is *tacit* [48].

The features of explicit knowledge include: 1) It can be expressed in words and numbers, easily communicated and shared [48], for instance, scientific formulae, and universal principles, etc.; 2) It can be well documented or recorded objectively in public domain, professional associations, or company-based resources, etc.; and 3) It is about past matters or objects and is oriented toward a context-free theory.

Explicit knowledge can come from natural or social sciences, engineering science, and practice. Knowledge from natural or social sciences [19] includes mathematics, philosophy, biology, sociology, physics, geometry, chemistry, cybernetics, psychology, art, mechanics, knowledge theory, medicine, economics, optics, heuristics, work science, acoustics, etc. Knowledge from engineering science includes strength of materials, thermodynamics, manufacturing technology, material science, fluid mechanics, manufacturing and production sciences, etc. Knowledge from engineering practices includes successful or failure previous design cases (successful cases could be used again, but failure cases should be avoided), patents, industrial or technical standards (established by authorities or their own organizations), design formulae and rules (usually from handbooks and manuals, catalogues, or derived from experts experience or experiments) [53].

The features of tacit (or implicit) knowledge include: 1) It is not easily visible and expressible [48], usually existing in people's brains. For instance, the background expertise and relationships that a salesman builds up over many years covering a territory; 2) it is subjective, intuitive, and hard to capture, store and share with others, e.g., the skill of a craftsman developed through years of experience; 3) it is created in a specific and practical context.

Tacit knowledge is generally deep rooted within people's memory and spread through human interactions, either face-to-face or through virtual space.

4.3 Taxonomies of Engineering Knowledge

Many other knowledge classification schemes or taxonomies also exist. When searching for engineering knowledge classification, some authors' names appeared frequently, including Vincenti, Ropohl, Faulkner, and De Vries [64], [65]. Their knowledge classifications are summarized in Table 6. The first four classifications are compared and analysed in the book "Philosophy of Technology and Engineering Sciences" [65], and except Faulkner's, the other four classifications are also compared by Broens and De Vries [66].

Table 6. Summary of five engineering knowledge classifications.

Practical classifications		Philosophical classifications		
Vincenti [67]	Faulkner [65]	Ropohl [64]	De Vries [68]	Bayazit [69]
Fundamental design concepts	Related to natural world	Structural rules	Physical-nature knowledge	Procedural Knowledge
Criteria and specifications	Related to design practice	Technological laws	Process knowledge	Declarative Knowledge
Theoretical tools	Related to experimental R&D	Functional rules	Functional-nature knowledge	Normative Knowledge
Quantitative data	Related to final product	Technical Know-how	Knowledge of physics-function relations	Collaborative Knowledge
Practical considerations	Related to knowledge	Socio-technical understanding		
Design instrumentalities				

Table 6 gives an overview of the five different classifications. Roughly, Vincenti and Faulkner categorize engineering knowledge from a practical viewpoint, whereas Ropohl and De Vries categorize knowledge from a philosophical viewpoint.

Bayazit classifies knowledge as procedural, declarative, normative and collaborative. Procedural knowledge directs a designer to go through the design process in a sophisticated way and declarative knowledge includes knowledge about functions, materials, shapes, manufacturing processes, economic, and social knowledge, etc. Sim and Duffy [18] categorize tacit knowledge into three subclasses: declarative knowledge (synonymous with knowing that, or knowledge of how things are), procedural knowledge (knowing how) and causal knowledge (know why). In the engineering design domain, these three types of knowledge can be generally referred to as design object knowledge, design process knowledge and design rationale knowledge, respectively. Explicit knowledge can be divided into procedural and declarative knowledge as well.

From the literature, Vincenti's classification has received both recognition and criticism. Houkes [65] thinks Vincenti's classification performs badly in terms of exclusiveness and completeness. For instance, his catalog is partly guided by the distinction between codifiable theoretical tools and quantitative data, and uncodified practical considerations. However, practical considerations may be codified without turning into either data or tools. Ropohl [64] doesn't think some of Vincenti's categories seem specific to technical knowledge, such as "criteria" and "quantitative data". On the contrary, Broens and De Vries [66] believe Vincenti's, Ropohl's and Bayazit's classification are very similar. They surveyed mechanical engineers and found that 43% designers and engineers considered Vincenti's classification to be better than other traditional classifications, such as Dewey Decimal Classification (DCC) and the Universal Decimal Classification (UDC), which are divided by disciplines and sub-disciplines.

To sum up, it is not necessary to say which of the five classifications is the best one. Obviously, there is no universal classification; they are all valid, and depend on the perspectives. However, it is useful to identify and exploit existing knowledge classifications where they are useful, rather than developing new ones.

4.4 Engineering Knowledge Representation

In the environment of collaborative engineering design, sharing and exchanging information and knowledge is extremely critical, so making information and knowledge explicit, context aware and sharable are the topics of knowledge representations. In the field of Artificial Intelligence (AI), knowledge representation must support computer-based "reasoning". In AI, the fundamental goal is to represent knowledge in a manner that facilitates inference (e.g., drawing conclusions) from knowledge elements and creates new elements of knowledge. It analyzes how to formally think - how to use a symbol system to represent a domain of discourse (which can be talked about), along with functions that allow inference (formalized reasoning) about the objects [70].

In AI, these representations should support machine inference mechanisms, so the knowledge representation must be fully explicit. However, the author of this thesis argues that an Intelligent Assistant system is required in collaborative engineering, not Artificial Intelligence. Knowledge representations are used for supporting human information processing, not for computer reasoning. Because humans can understand partially implicit knowledge (e.g., sketches), knowledge representations in CED do not need to be as formal or structured as in AI systems. In other words, in CED KR must support human understanding and reasoning, not computer-based reasoning.

Engineers use a combination of different modes and representations to record and communicate explicit knowledge. Several of these are described below.

Verbal representation: Verbal communication is perhaps the easiest way to communicate knowledge. However, it is an informal way to represent knowledge, which may bring problems such as ambiguous and incomplete expression, since the meaning might be different when the speaker's facial expression is different. Verbal communication tends to be transient unless it is explicitly recorded in another form.

Textual representation: Written text is a common way to represent knowledge in explicit form. Careful writing can reduce ambiguity and misunderstanding, and a permanent record is easily maintained. It is applied widely in messages, documents, spreadsheets and other files.

Graphical representations: Graphical representations include photographs, graphs, charts, diagrams, drawings and freehand sketches. Graphical representations are very important in engineering.

Mathematical representations: Mathematical representations include mathematical models, calculations, etc. Much design knowledge is represented mathematically.

Physical representations: Physical representations include mock-ups, prototypes, experiments, etc.

Multi-media representations: Multimedia representations include video, audio, animations, and interactive CAD models. Multimedia representations require computer support.

4.5 Semantic Networks

Mental models can be considered in two different but related ways in Ref. [71]. One is the view of cognitive scientists, who define mental models as "an internal scale-model representation of an external reality". It is built on-the-fly in peoples' minds when they perceive or observe something, or are told something by others. The other view is from the field of Human-Computer Interaction, which regards mental model as "a set of beliefs about how a system works". HCI practitioners aim to help humans make sense of an increasingly complex world. In sum, mental models, such as perspectives, beliefs, schemata, paradigms, and viewpoints, help people to perceive and define their world [72].

The semantic network (Figure 6) is one of representational formats of mental models [73]. A semantic network represents semantic relations among objects. It is a directed or undirected graph consisting of vertices, which represent objects, and edges, which stand for relationships between objects.

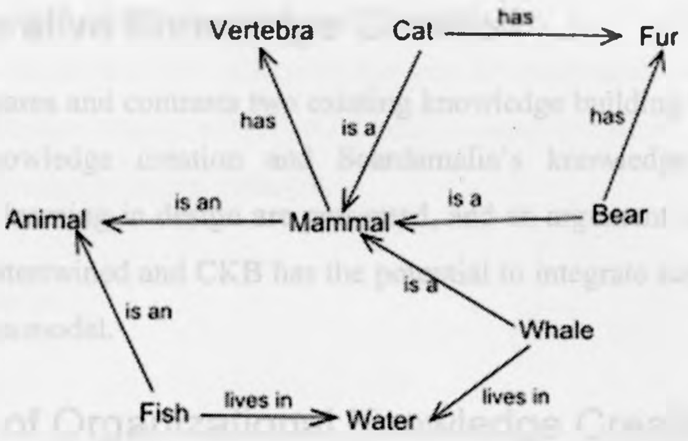


Figure 6. An example of a semantic network from Wikipedia.

5 Collaborative Knowledge Creation

This chapter compares and contrasts two existing knowledge building theories: Nonaka's organizational knowledge creation and Scardamalia's knowledge building theory. Existing views of learning in design are presented, and an argument is made that design and learning are intertwined and CKB has the potential to integrate activities of both in a single collaboration model.

5.1 Theory of Organizational Knowledge Creation

In 1995, Nonaka and Takeuchi introduced their Organizational Knowledge Creation theory, which involves developing new content or replacing existing content within the organizations' tacit and explicit knowledge. They believe that "knowledge creation is a spiral process, starting at the individual level and moving up through expanding communities of interaction, which crosses sectional, departmental, divisional, and organizational boundaries [48]." The following sections describe the primary features of this theory.

5.1.1 Two Dimensions of Knowledge Creation

The key to knowledge creation is mobilization and conversion of tacit knowledge; the core of the theory is how organizational knowledge spiral emerges. The basic framework contains ontological and epistemological dimensions (see Figure 7).

The *ontological dimension* of knowledge creation represents the idea that the knowledge of individuals is amplified organizationally and crystallized as part of the organizational intellectual capital. Knowledge creation is not limited to individuals, but also occurs at group, organizational, or inter-organizational levels.

The *epistemological dimension* concerns the conversion between tacit knowledge and explicit knowledge. Nonaka and Takeuchi argue that tacit knowledge consists of both cognitive and technical elements. Cognitive elements focus on mental models, in which people build images of the reality, working models of the universe and vision for the future, by processing and manipulating analogies in their minds. The technical elements of tacit knowledge include specific know-how, crafts and skills. In contrast, explicit knowledge tends to be objective and can be transmitted in a formal and systematic language. Only a small fraction of knowledge can be expressed explicitly. Therefore sharing tacit knowledge among employees through communication is the key point of how the knowledge creation spiral emerges.

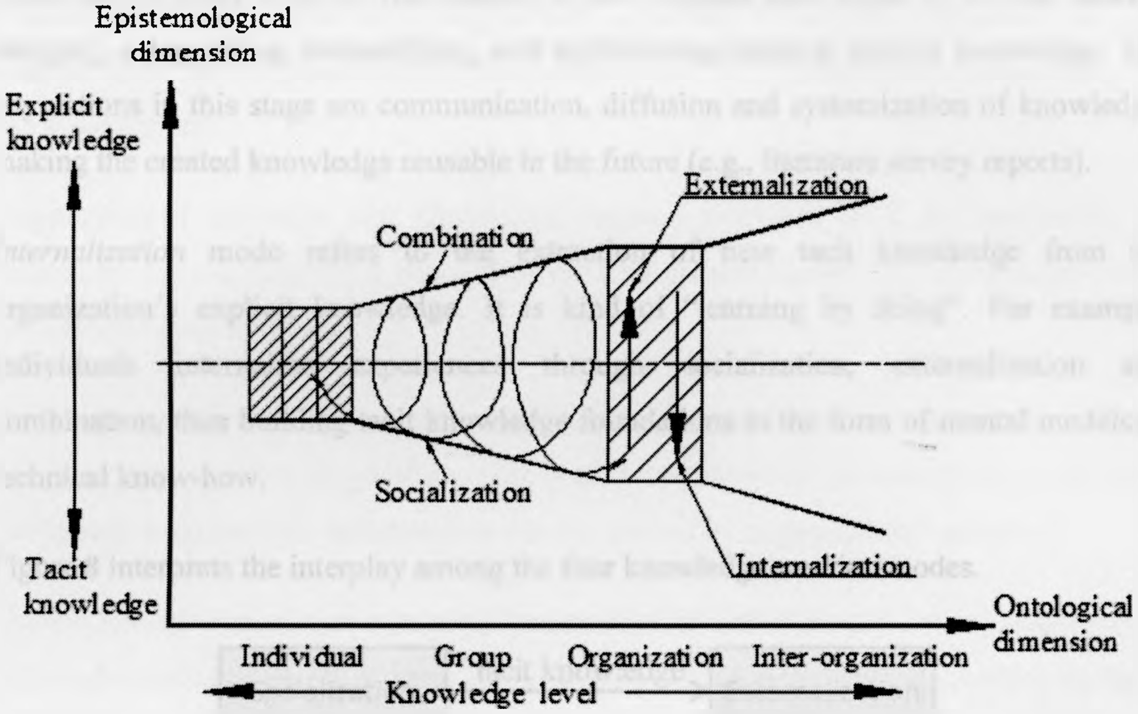


Figure 7. Two-dimension-spiral of knowledge creation [48].

5.1.2 Four Modes of Knowledge Conversion

Nonaka and Takeuchi believe that knowledge is created and expanded during the interaction between tacit and explicit knowledge; they call this interaction *knowledge conversion*. The interaction is a social process and has four different modes: Socialization, Externalization, Combination and Internalization, abbreviated as SECI.

Socialization is defined as conversion of tacit knowledge to new tacit knowledge through social interaction and experience sharing among organizational members by gathering, spending time together, or living in the same environment. People learn from each other by observation, imitation and practice, even without language.

Externalization means converting tacit knowledge to new explicit knowledge. In this phase individuals commit to the group and the individuals' knowledge is fused and integrated to become new organizational intellectual capital (e.g., articulation of best practices or lessons learned).

Combination mode involves the creation of new explicit knowledge by sorting, adding, merging, categorizing, reclassifying, and synthesizing existing explicit knowledge. The key actions in this stage are communication, diffusion and systemization of knowledge, making the created knowledge reusable in the future (e.g., literature survey reports).

Internalization mode refers to the extraction of new tacit knowledge from the organization's explicit knowledge. It is kind of "learning by doing". For example, individuals internalize experiences through socialization, externalization and combination, thus building tacit knowledge foundations in the form of mental models or technical know-how.

Figure 8 interprets the interplay among the four knowledge creation modes.

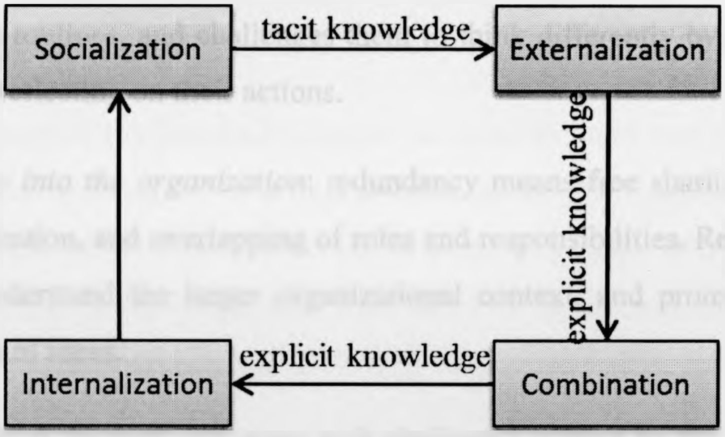


Figure 8. Knowledge creation modes.

Nonaka and Takeuchi stress that all of the four modes must be triggered continuously during the dynamic interaction involved in knowledge creation. In addition, they believe the new knowledge created in each conversion mode is different. Socialization results in "sympathized knowledge"; externalization produces "conceptual knowledge";

combination creates “systemic knowledge”; and internalization produces “operational knowledge”.

5.1.3 Five Conditions for Organizational Knowledge Creation

Providing the proper environment to facilitate and encourage group interaction is the role of an organization in the knowledge-creation process. Nonaka and Takeuchi list five conditions at the organizational level, which can drive employees to promote the knowledge spiral.

Organizational intention: the knowledge creation activities must be purposeful, as defined by the organization’s goals. The value of created knowledge is evaluated based on organizational intention.

Individual autonomy: the employees should be permitted to create knowledge autonomously as much as possible. This motivates individuals to contribute knowledge, and to self-organize their collaborative efforts, guided by organizational intention.

Fluctuation and creative chaos: an environment of “creative chaos” forces people out of their comfortable routines, and challenges them to think differently by questioning their assumptions and reflecting on their actions.

Build redundancy into the organization: redundancy means free sharing of information within the organization, and overlapping of roles and responsibilities. Redundancy allows individuals to understand the larger organizational context, and promotes sharing and cross-fertilization of ideas.

Maintaining internal diversity can cope with challenges caused by the environment. To maximize variety, every employee in the company ought to be assured of the quickest access to the broadest variety of necessary information and knowledge, going through the fewest steps.

5.1.4 Five-Phase Model of the Organizational Knowledge Creation Process

Nonaka and Takeuchi developed a theoretical framework of the organizational knowledge creation process, consisting of the following five phases.

Sharing tacit knowledge: the first phase is sharing of tacit knowledge among individuals through social interaction. The organizational context is usually a self-organizing team motivated by a problem related to organizational intention. Guided by organizational intention, team members act autonomously to select a diversity of members with complementary expertise. The members share redundant skills and knowledge, and are challenged by creative chaos.

Creating concepts: a shared mental model formed and clarified through continuous dialogues, and finally crystalized into explicit concepts by deduction, induction or abduction. This phase corresponds to externalization.

Justifying concepts: the organization must verify if the new concept is worthy of pursuit based on organizational intention. For engineering organizations, the normal justification criteria include feasibility, machinability, cost and profit, etc.

Building an archetype: the justified concept is converted into something tangible or concrete, that is, an archetype. For a new product development process, the archetype could be a prototype; for a service innovation, it could be a model of a novel managerial system. To build a prototype, designers from different disciplines are put together to develop specifications, manufacture, etc. The third and fourth phases are akin to combination.

Cross-leveling knowledge: in this phase the new concept, which has been justified and modeled, is expanded from a section to others in the division, across to other divisions and then beyond the organization in what called “cross-leveling” of knowledge. By now, the concepts and archetype become explicit knowledge in organizations in the form of engineering drawings, documents, patents, products, systems and /or services.

The organizational knowledge-creation process combines all of these elements as shown in Figure 9.

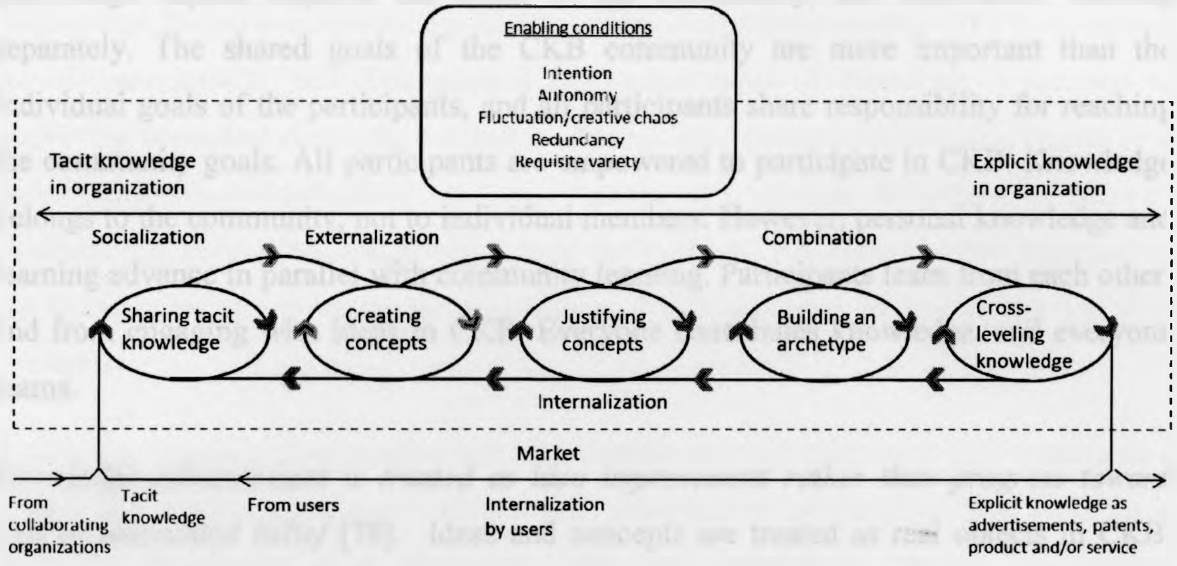


Figure 9. The model of organizational knowledge-creation process [48].

5.2 Collaborative Knowledge Building

Another perspective on knowledge creation is provided by the theory of knowledge building developed by Scardamalia and Bereiter to support deep learning in education [74], [75], [76], [77], [78]. They define knowledge building as "the production and continual improvement of ideas of value to a community, through means that increase the likelihood that what the community accomplishes will be greater than the sum of individual contributions and part of broader cultural efforts [76]." Thus, they believe knowledge building is not limited to education, but goes on throughout a knowledge society. The theory is called Collaborative Knowledge Building (CKB) in the literature.

5.2.1 The Features of CKB Theory

Based on the literature review of CKB theory, some of its primary features are summarized as follows.

Knowledge advancement is attained as a community outcome rather than individual achievement [78]. The state of knowledge in a specific field is not in the minds of even the most knowledgeable individuals, but within the community of that field. Advancing knowledge capital requires the effort of the community, not individuals working separately. The shared goals of the CKB community are more important than the individual goals of the participants, and all participants share responsibility for reaching the community goals. All participants are empowered to participate in CKB. Knowledge belongs to the community, not to individual members. However, personal knowledge and learning advance in parallel with community learning. Participants learn from each other, and from engaging with ideas in CKB. Everyone contributes knowledge, and everyone learns.

Knowledge advancement is treated as idea improvement rather than progress toward true or warranted belief [78]. Ideas and concepts are treated as real objects in CKB, which are considered to be improvable. Idea improvement is the core and explicit principle of CKB and it guides the efforts of participants. Ideas and concepts must be sought from a number of sources and perspectives, then they are shared, discussed, compared, connected, expanded, refined, etc.

People eventually obtain deep structural knowledge of something, not knowledge about something [78]. "Knowledge of" consists of both declarative knowledge (know-what) and procedural knowledge (know-how); while "knowledge about" is only declarative knowledge. Participants are motivated and empowered to manage and direct their own participation in CKB, without being directed by teachers or other managers. In CKB, participants work with problems that result in deep structural "knowledge of".

Knowledge building is a dynamic improvement process. In knowledge building, people advance the frontiers of knowledge in their community through purposeful activities such as "identifying problems of understanding, establishing and refining goals based on progress, gathering information, theorizing, designing experiments, answering questions and improving theories, building models, monitoring and evaluating progress, and reporting [76]". Higher level ideas emerge in the process, and there is no end to the improvement process.

Knowledge building discourse aims at idea improvement. Scardamalia and Bereiter believe that discourse in knowledge building aims at idea improvement [78]. It involves a set of commitments that distinguish it from other types of discourse: it commits to improving ideas and concepts, not just to sharing information or expressing opinions; it seeks "common understanding", not only agreement; it stresses expanding the basis of evidence and persuasion, respecting others' perspectives, not denying other viewpoints. By these criteria, argumentation and debate are encouraged in knowledge building discourse.

Knowledge building theory advocates constructive use of authoritative information. In a knowledge building community, people seek authoritative information. That is, CKB is built on a foundation of existing knowledge from authoritative sources. Participants identify and respect appropriate authoritative sources, and incorporate ideas from these sources. At the same time, participants evaluate sources critically.

Participants in a CKB community critically assess and evaluate their own progress toward the community goals. The purpose is to identify and address problems, issues and barriers to success.

CKB requires that participants in the knowledge building community recognize and follow socio-cognitive norms and values, such as contributing to collective knowledge advances, constructive and considerate criticism, idea improvements [76], and so on.

5.2.2 CKB Process Models

This section will introduce and compare two CKB process models proposed by Stahl [79] and Singh et al. [77] respectively.

Stahl proposed an initial CKB model (Figure 10) in 2000 based on the perspective of learning as a social process of collaborative knowledge building [79]. He decomposed CKB into two interacting cycles: personal understanding, and social knowledge building.

The convention in this model is that rectangles represent forms of knowledge and arrows represent transformative processes. The two cycles cannot be separated in reality, but depend on each other. In this model, they are separated for conceptual clarity.

The cycle of personal understanding is an individual cognitive process. Stahl believes that learning usually starts from the tacit pre-understanding, which is from the previous personal experience and concepts. However, people may encounter some problems making these understandings collapse completely. They have to seek solutions to mend the gap through feedbacks of new practice and feeling. When they can explain all of these implications, an updated personal comprehension is reached and can be proved right if no contrary points are found. Consequently, this comprehension will increasingly become a tacit pre-understanding as a new start point for next learning cycle. The tacit understanding is influenced by the people's culture background, past experience and feedback from previous social interactions, etc.

The social knowledge building cycle can build on and supplement the personal understanding cycles of several individuals. This process takes place in a social context and it is a social epistemological process. It begins with the public statements of several persons on a certain problem, followed by an extensive and refining discussion if disagreements or contradictions exist. During the discussion, the participants will clarify the different meanings in various perspectives and terminologies combined with personal viewpoint gradually changing until they arrive at a shared understanding. Then successful negotiation can result in an agreement which is acceptable to all participants. The agreement is regarded as new knowledge which should be formalized and become cultural artifacts. Individuals understanding is involved at every social phase, but cannot be explicitly represented in the model.

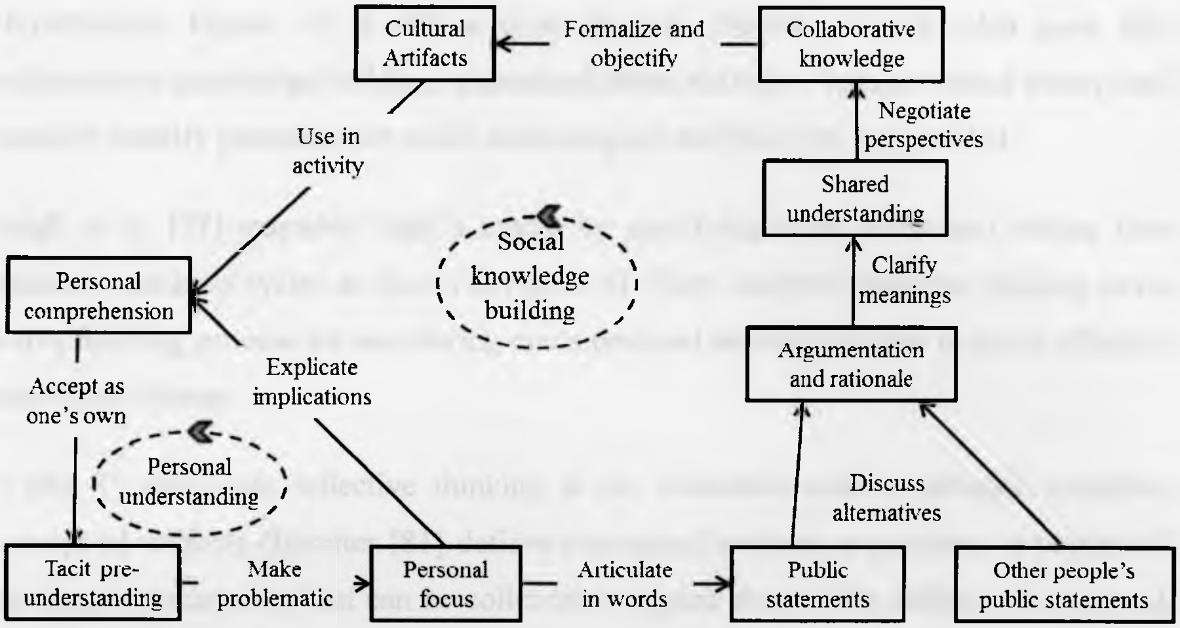


Figure 10. The first version of Stahl's model of knowledge building process [80].

Stahl also illustrated it would be wrong to suppose the CKB process always went through the same sequence as shown in Figure 10. Indeed, the identification of the particular set of elements is incomplete and approximate. To avoid misleading, Stahl and Thomas Herrmann [80] collaboratively developed another model to show the mediation of personal understanding and group knowledge building in a different form. In the model, the cycle of personal understanding includes five components:

1) tacit understanding of the world
2) experiencing breakdowns in understanding
3) reinterpreting meaning structures to reconcile contradictions
4) articulating ones' understanding
5) formally structuring knowledge

The cycle of social knowledge building activities including:

1) sharing perspectives
2) exchanging arguments and rationale
3) clarifying meanings
4) negotiating conflicts
5) formally structuring knowledge

Nevertheless, Figure 10 is still a good start to illustrate clearly what goes into collaborative knowledge building, understand those activities, design a sound theory, and possibly identify processes for which technological scaffolds can be provided.

Singh et al. [77] extended Stahl's model by modifying some terms and adding four reflective thinking cycles as shown in Figure 11. They interpret reflective thinking as an active thinking process for monitoring one's personal learning process to cause effective conceptual change.

"Cycle 1" represents reflective thinking at the individual level to develop reflective conceptual artifacts (Bereiter [81] defines conceptual artifacts as products or objects of thinking and reasoning that can be collectively argued about). The reflective conceptual artifacts are used by participants while discussing different perspectives. Tacit pre-understanding represents the individual's use of old experiences and knowledge. When faced with a problematic situation, a person uses reflective thinking to articulate tacit knowledge in the form of conceptual artifacts. It is through interacting with these artifacts that people interpret meaning, engage in discussions, develop a shared understanding and collaboratively build knowledge.

"Cycle 2" represents collaborative reflective discourse at the group level to develop shared understanding. Shared understanding is crucial in terms of ensuring that each perspective is understood and group members are on some level of common ground. There is also a combined action between individual reflective thinking and collaborative reflective discourse, with the former working at the individual level and latter at the group level.

"Cycle 3" and "Cycle 4" stand for the mediating role of reflective thinking for resolving contradictions. Although the four cycles are shown separately, "they are intertwined at various levels of abstraction in the CKB process [82]".

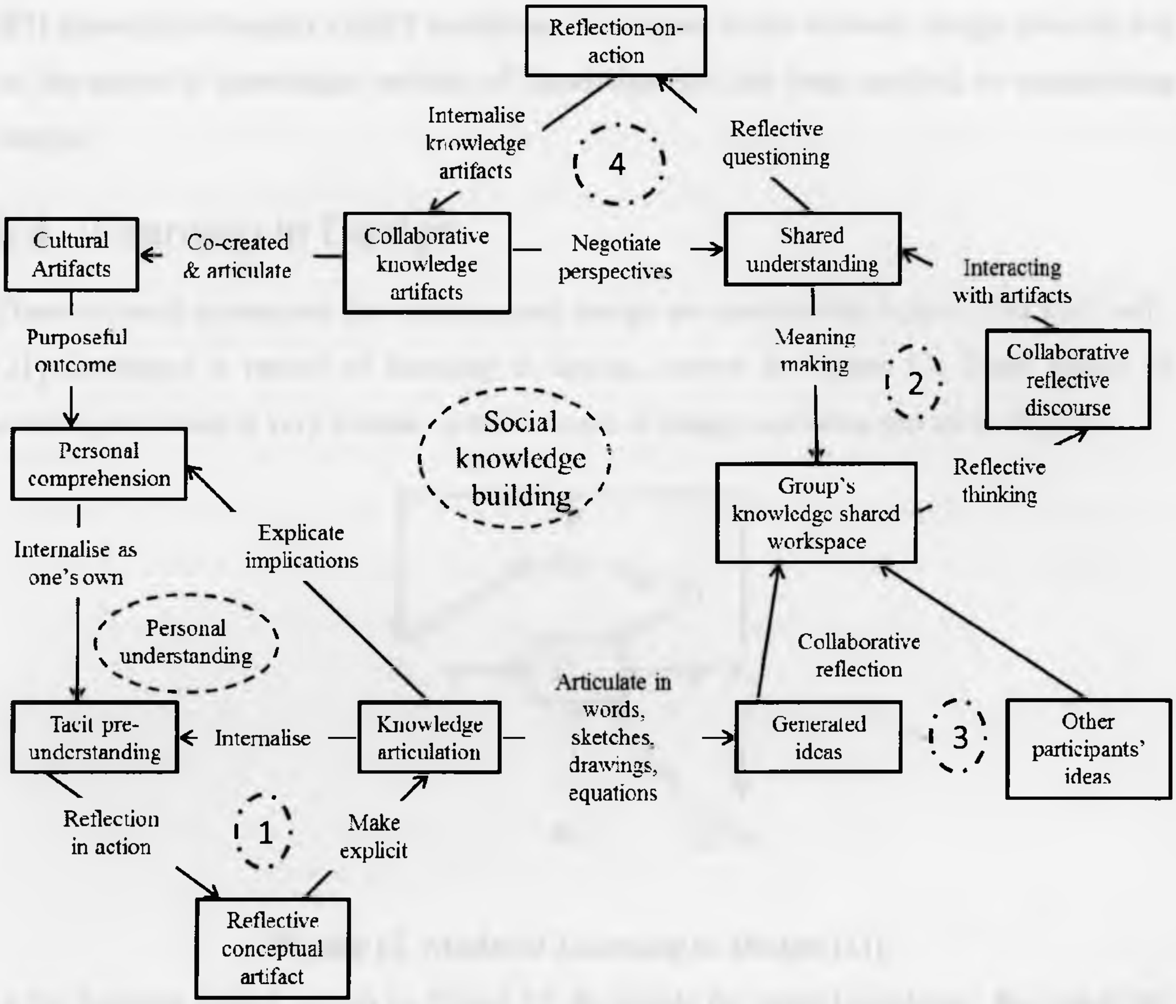


Figure 11. Extended CKB Model (adapted from [82]).

5.3 Comparison of the Two Knowledge Building Theories

Both of the two theories describe the knowledge building process as the combination of individuals' mental activities and group interactions. Both describe recurring knowledge transformation activities, and both recognize tacit and explicit forms of knowledge. Both theories also correspond well with Sim and Duffy's model of generic design activities. Both theories also support the ECN-based collaborative engineering process.

Nonaka's knowledge creation theory comes from organizational knowledge management, while Scardamalia's collaborative knowledge building theory is rooted in education. The theories appear to have been developed independently, and there are few if any cross references in the respective literature. A very recent paper by Dubberly and Evenson

[83] shows how Nonaka's SECI model can be mapped to the software design process, but to the author's knowledge neither of these theories has been applied to engineering design.

5.4 Learning in Design

There is broad agreement that learning and design are inextricably linked. Sim and Duffy [21] developed a model of learning in design, shown in Figure 12. Their model of learning activities is very similar to their model of design activities shown in Figure 3.

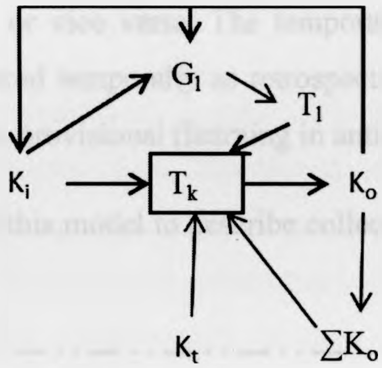


Figure 12. Model of Learning in Design [21].

In the learning model shown in Figure 12, **K_i** stands for input knowledge; **K_o** stands for output knowledge or knowledge learnt; **G_l** is the goal of learning; **T_k** represents knowledge transformation activities. There are seven opposed pairs of transformers:

1) abstraction/detailing;
2) association/disassociation;
3) derivations (reformulation)/randomization;
4) explanation/discovery;
5) group rationalization (or clustering)/decomposition (ungrouping);
6) generalization/specialization;
7) similarity comparison/dissimilarity comparison.

T_l represents the reasons that trigger learning. Four triggers are identified:

1) provisional learning trigger;
2) in situ trigger;
3) retrospective trigger;

4) rationale trigger.

$\mathbf{K_t}$ represents knowledge of transformers, which determines the appropriate transformer to apply; $\mathbf{\sum K_o}$ stands for the accumulation of output knowledge. As a result of learning activities, there are multiple types of output knowledge.

They also analyze the interaction between designing and learning. Three links between them were identified as epistemic link, teleological link, and temporal link. The epistemic link is related to knowledge acquisition and transformation process during the design and learning process. The teleological link is concerned with the goals, that is, the design goal can precede a learning goal or vice versa. The temporal link refers to the design and learning activities can be linked temporally as retrospective (learning from experience), in-situ (learning as needed) or provisional (learning in anticipation of need).

Wu and Duffy [84] extended this model to describe collective learning in team design, as shown in Figure 13.

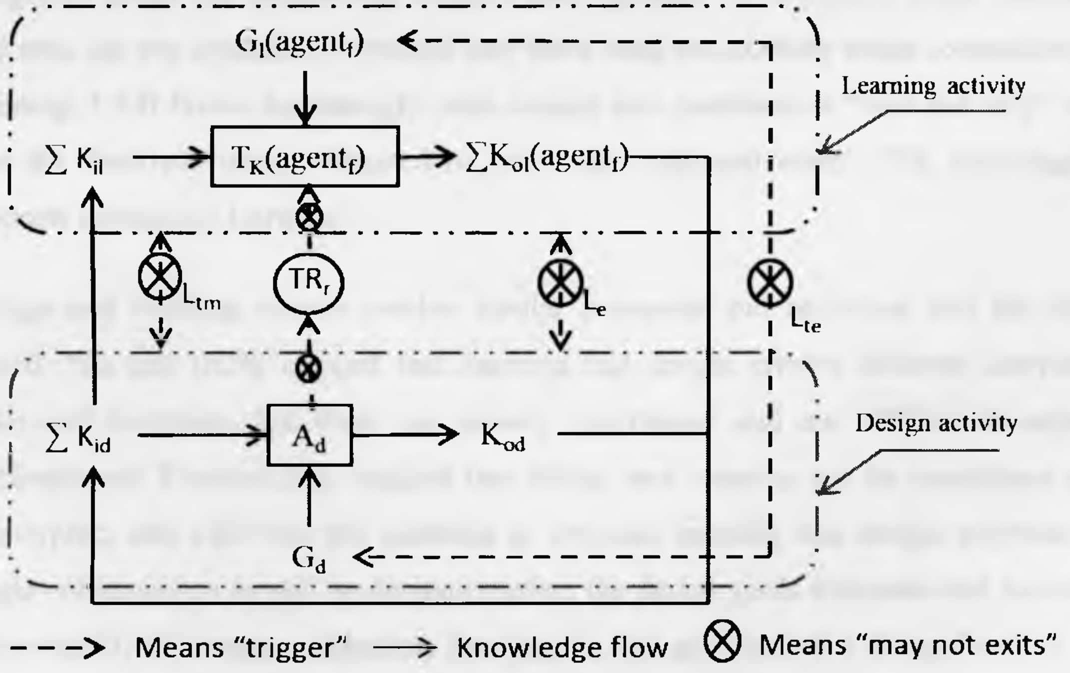


Figure 13. A model of collective learning in design (adapted from [84]).

Here, $\mathbf{\sum K_{il}}$ is the sum of input knowledge from all sources to support team learning; $\mathbf{\sum K_{id}}$ is the sum of input knowledge from all sources to support team design; $\mathbf{G_l(agent_f)}$

represents the learning goal of agent_f (f=1, ..., m, m is the number of agents who conduct learning activity); $\mathbf{G_d}$ stands for design goal; A_d represents design activity; $\mathbf{K_{od}}$ is output knowledge of team design; $\sum\mathbf{K_{ol}}$ (agent_f) is the sum of learned knowledge of agent_f; T_K(agent_f) is knowledge transformer of agent_f; $\mathbf{TR_r}$ refers to rationale trigger; $\mathbf{L_{tm}}$, $\mathbf{L_e}$ and $\mathbf{L_{te}}$ represent temporal link, epistemic link and teleological link respectively.

According to this model, agents (team members) require and transform knowledge from each other through their interactions, such as conversations and team meetings. The learning goal and the design goal interact with each other in two ways: the learning goal can precede the design goal, or vice versa.

CKB has been developed primarily to support intentional or goal-driven learning. Bereiter and Scardamalia use the term “intentional learning” to refer to cognitive processes that have learning as a goal rather than an incidental outcome. From this point, Sim and Duffy’s learning in design is also intentional learning since it has learning goal. People set goals for themselves, monitor their progress toward those goals, understand and seek out the conditions in which they learn best, and actively make connections and meaning. CKB favors increasingly deep inquiry into questions of “how and why” rather than the shallower inquiry directed by questions “what and when” [78], accordingly, it supports intentional learning.

Design and learning clearly involve similar processes and activities, and are closely linked. Wu and Duffy suggest that learning and design involve different knowledge, goals and activities, but these are closely intertwined and are difficult to separate. Dubberly and Evenson [83] suggest that design and learning can be considered to be isomorphic, and CKB has the potential to integrate learning and design activities in a single collaboration model. In design practice, the design goals dominate and learning is a byproduct; in design education, learning is the goal and the design artifact is a byproduct. In many situations, the design and learning goals coexist on an equal footing, or with alternating priority at different stages.

6 An Integrated CKB-oriented Model for Collaborative Engineering Design

In this chapter, an integrated knowledge building-oriented CED model is built and illustrated in detail. The model incorporates the key elements of Stahl's CKB model, Lu's ECN-based collaborative engineering model, Nonaka's knowledge creation theory, and Sim and Duffy's model of a design activity.

6.1 Compare CKB Theories and ECN-based Collaborative Engineering Model

So far, three models of knowledge creation have been reviewed: Stahl's and Singh's CKB models from the collaborative learning literature, Nonaka's knowledge creation process from the knowledge management literature, and ECN-based model from the engineering design literature. These theories have evolved independently, yet the common themes are clear. The differences are mainly due to differing terminology and perspectives. New insights can be gained by comparing the theories, and mapping them to each other as shown in Figure 14.

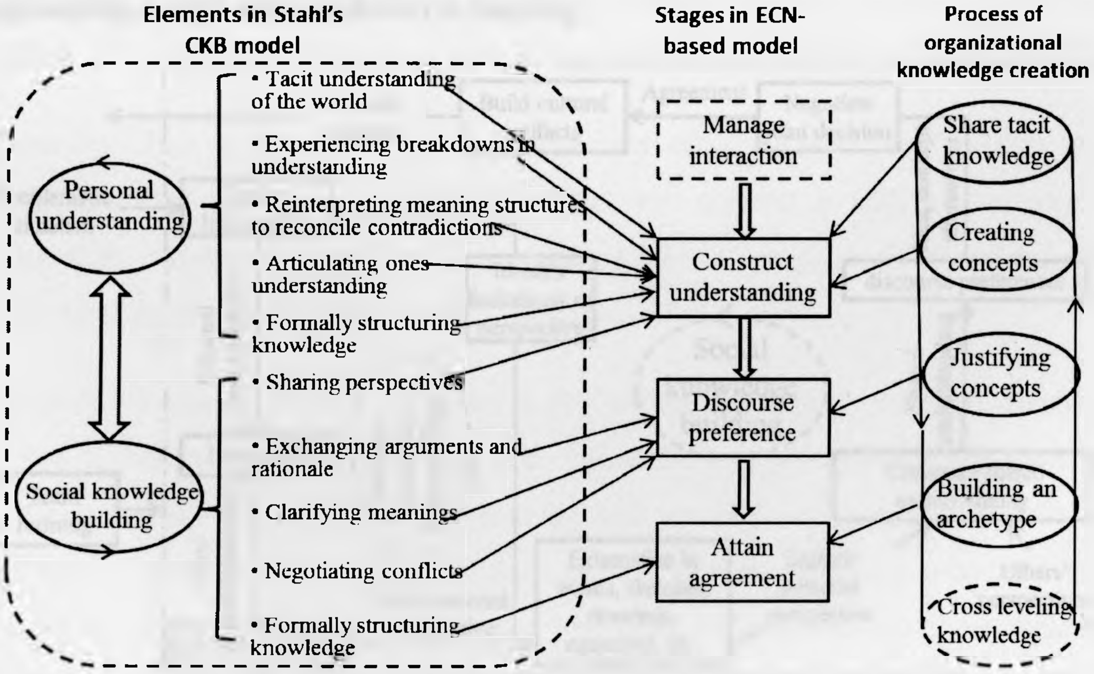


Figure 14. The corresponding relation graph between CKB elements and ECN stages.

Most of the elements in knowledge building correspond to the phases of collaborative engineering. “Cross-leveling knowledge” in Nonaka's model is equivalent to the new cultural artifacts in CKB (shown in Figure 10, not in Figure 14). While not explicitly represented in the model, the outcome of ECN is a design artifact, which can be considered as a cultural artifact or artifact theory. ECN includes a “manage interaction” step not explicitly modeled in the other processes. Nevertheless, it does not mean it may not exist in the other two the knowledge building environments since preliminary team formation is required for any collaborative activity.

6.2 Integrated Knowledge-building-orientated Model for CED

An integrated model for CED is proposed in this section, incorporating the key elements of Stahl’s CKB model, Lu’s ECN-based collaborative engineering model, Nonaka’s knowledge creation theory, and Sim and Duffy’s model of a design activity. In Figure 14, the convention is that the boxes represent activities and arrows represent knowledge, to be consistent with the design activity model in Figure 3. The purpose of the model is to describe, understand and explain how knowledge is created in both collaborative engineering design and collaborative learning.

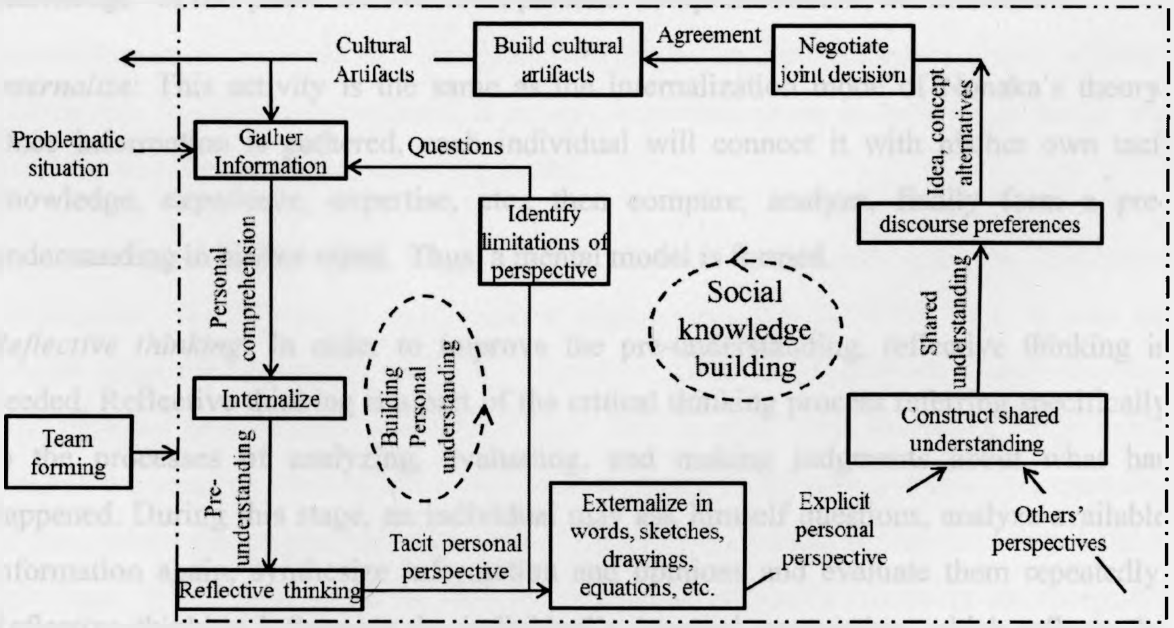


Figure 15. An integrated knowledge-building-orientated model for CED.

The CED process starts from a problematic situation and ends up with cultural artifacts. The new knowledge about the cultural artifacts feeds back to the personal understanding cycle, supporting learning from experience. The model primarily consists of two expanded cycles: building personal understanding cycle and social knowledge building cycle. The two cycles are intertwined—the individual mind is involved at each social phase.

It is necessary to point out that the model does not depict a design sequence, but design activities that may recur frequently in any design phase (conceptual design, embodiment design or detailed design). So, it doesn't mean any problematic situation must go through all steps one by one in this model. Some steps might be skipped.

Next, each design activity in this model will be elaborated.

Gather information: This could happen either at the beginning when a problem occurs or during the process when more information is needed. Any relevant information should be collected as "input knowledge", such as a detailed description of the problem, engineering principles, previous research, requirements of clients, etc. Information could come from customers, colleagues, handbooks, research papers, suppliers, etc. "Output knowledge" of this phase is a low-level personal comprehension of the information.

Internalize: This activity is the same as the internalization mode of Nonaka's theory. Once information is gathered, each individual will connect it with his/her own tacit knowledge, experience, expertise, etc., then compare, analyze, finally form a pre-understanding in his/her mind. Thus, a mental model is formed.

Reflective thinking: In order to improve the pre-understanding, reflective thinking is needed. Reflective thinking is a part of the critical thinking process referring specifically to the processes of analyzing, evaluating, and making judgments about what has happened. During this stage, an individual may ask himself questions, analyze available information again, synthesize information and opinions and evaluate them repeatedly. Reflective thinking influences the individual's personal perspective, which reflects the person's ideas, values, priorities, beliefs, biases, preferences, knowledge, background, and expertise, etc.

Identify limitations of perspectives: If the personal perspective is not complete and still has some questions which cannot be answered satisfactorily, the stakeholder will need go back to search for more information. This happens very often in engineering design and in learning. A question seems solved, but it may raise another question. This could cycle again and again as understanding improves.

Externalize personal perspective: Personal perspectives include ideas, opinions, assumptions, questions, concerns, etc. Stakeholders need convert tacit perspectives into explicit form, so that they can communicate with others. In engineering design, words are not strong enough to support externalization; thereby, other representations may be involved, such as calculations, pictures, sketches, drawings and multi-media. This is also externalization in Nonaka's SECI model.

Construct shared understanding: All personal perspectives are externalized and shared in a group workspace where they can be compared, contrasted and discussed. The output of the group interaction is shared understanding, which may include identification of similarities and differences, clarification of contradictions, limitations, unified terminologies, agreements and disagreements, etc. The interactions can also influence and modify personal perspectives. It is very important phase in both CKB model and ECN-based collaborative engineering model.

Discourse preferences: This stage has the similar connotation with the "discourse preferences" in ECN-based collaborative engineering, but for CED here is also the central stage for knowledge building. Discourse seeks to improve ideas and concepts, thus creating new collaborative knowledge. This process aims to evolve many initial ideas into a small number of better and more complete ideas, with a shared understanding of their relative strengths and weaknesses. Differing individual preferences may exist at this stage. If the negotiation of different perspectives finally results in an acceptable group preference, then such an outcome is regarded as new knowledge. It embodies in new concepts or alternatives for solution.

Negotiate joint decisions: In the above stage, consolidated group preferences are negotiated by stakeholders to arrive at joint decisions. The outcome is an agreement supported by the group.

Build cultural artifacts: Cultural artifacts are permanent knowledge objects that become part of shared knowledge beyond the community. Cultural artifacts include research papers, reports, engineering drawings, physical prototypes, and the design artifacts themselves. A record of the discourse used to reach agreements is also a valuable cultural artifact, as it reveals rationale and can be used to guide similar projects in the future.

The integrated CKB-oriented model shown in Figure 15 is based on existing validated models: Stahl's model is validated in collaborative learning, Nonaka's organizational knowledge creation model has been implemented in some companies and Lu's ECN is validated in a case study. If we accept the premise that the nature of collaboration is essentially the same in collaborative knowledge building, collaborative learning and collaborative design, then it is reasonable to assume that the combined model is also generally valid and can be used to guide collaboration tool developers. The model is a first step, and further validation and refinement should be done as future work.

7 Computer Support for CKB-based Engineering Design

This chapter proposes a set of general requirements of computer support for CED evolved from the integrated model in Chapter 6. After existing collaborative tools reviewed, specific functional requirements for CED are identified and several additional functions needed to support CED are identified and described in detail.

Chapter 6 presented collaborative engineering design as a process of collaborative knowledge building. Engineers need computer tools to support the whole process. That is, all of the recurring activities in the rectangle boxes in Figure 15 need to be supported; the whole process need to be facilitated appropriately; and all of the knowledge transformed in the arrow lines need to be captured, stored, organized, and prepared for future retrieval.

Most existing collaboration tools are not based on a theory of collaborative work. For example, Microsoft SharePoint is a leading collaboration tool used by many large organizations including the World Bank [85], Pfizer Global Research and Development [86], and University of Maryland University College [87]. A literature search using "SharePoint" as a keyword turned up no evidence that SharePoint is based on a theoretical framework, and no studies were found relating SharePoint to an existing theory. As a result, there is no evidence that the capabilities of tools like SharePoint correspond to the capabilities required to support collaborative knowledge building and design.

Some required collaborative functionalities are put forth by researchers, such as Scardamalia [75] and Stahl [80]. They all believe a shared workspace is needed to support and mediate activities in collaboration, and integrate day-to-day work of the community. Scardamalia advocates the tool should be a self-organizing system of interactions of participants and their ideas without needing an external organizer; it should be able to facilitate collaborative knowledge-building strategies, textual and graphical representation of ideas, and reorganization of knowledge artifacts; it can restructure the flow of information so that questions, ideas, criticisms, suggestions, and the like are contributed to a public space equally accessible to all; it is able to link those contributions, and support socio-cognitive practices. Stahl argues that a knowledge building environment should retain a record of all the new built knowledge; it is preferable to be built on asynchronous, persistent collaborative technologies and be implemented as a Web-based environment via internet; it should help people to express their ideas, discuss with others, clarify disagreements or misunderstandings, negotiate shared understandings, etc.; it should provide facilities, such as searching, filtering, linking, and so on; beyond that, other functions like compiling and formatting sets of notes, delivering information automatically when it needed, etc.

The system is not necessary an Artificial Intelligence system, but an Intelligent Assistant. This means that “an intelligence-amplification system” (machine + a mind) can beat an “Artificial Intelligence system” (a mind + imitating machine working by itself) [88]. For instance, for a CAD drawing, computer does not need to know what it stands for, just keeps it in a certain way and displays it when people need it. Individuals should interact with computers in natural ways, having technologies adapt to users, instead of users adapting to technologies.

7.1 General Requirements for CKB in Engineering Design

Based on the integrated CKB-oriented collaborative engineering design model, some specific activities and corresponding requirements are identified in the following sections.

7.1.1 Developing Individual Perspectives

The system should provide a private workspace analogous to a personal notebook, to help individuals to construct and record their evolving personal perspectives. The workspace should allow the user to input, restructure, organize and otherwise manipulate information of any kind, including documents, sketches, calculations, notes, etc. The tool should encourage articulation of rationale, reasoning, reflection, etc. It should be easy to create links and associations between the private and shared workspaces, and to selectively share the private perspective with other members of the group.

These tools would be used primarily to support individual thinking, so much of the knowledge required to interpret the explicit form would remain tacit. For example, a designer would understand the context or background of a sketch or note in a personal notebook, without needing to provide an explicit explanation. This tacit knowledge needs to be further articulated before the perspective can be shared and understood by others.

7.1.2 Constructing a Shared Understanding

Individual perspectives exist mostly in tacit form, and must be made explicit and shared with others in a shared workspace. While sharing personal perspectives and comparing with others, an individual's perspective can be influenced by the "social interaction". They may rearrange their own opinions and clarify them again. The perspectives themselves evolve as a consequence of discourse.

After reshaping individuals' perspectives, the tool should help to form a common understanding and then formalize and record it. Team members develop a deep and shared knowledge of the issues: what are the contrary parts, what are the unanimous parts, what are limitations, what are strengths, and a shared glossary may be created as well.

7.1.3 Discoursing Preferences

The shared understanding is used as a starting point for all participants to discourse further and reach a group preference. Discourse is a key part of CKB, and includes activities such as questioning, commenting, discussing, arguing, debating and negotiating. Tools are required to allow different perspectives to be discussed, compared and contrasted to identify areas of agreement, disagreement, conflict, contradiction, or omission, etc.

7.1.4 Consensus Building and Decision Making

After discourse the group preferences, improved ideas, concepts and alternatives are created. Then, the team needs to build consensus and establish agreements in order to progress toward the team goal. Many minor decisions need to be made at every stage in engineering design, and it is important to record the agreement and the rationale.

Tools are needed to allow group members to check the degree of consensus with a variety of voting methods to choose from. Typically, a vote result will reinforce a discussion and negotiation. Individuals can privately compare their own vote with the group's total result. Areas of disagreement on key aspects of a policy or plan will often be highlighted, which also encourages negotiation.

7.1.5 Building Cultural Artifacts

Finally, it is important to formally document and record agreements in the form of cultural artifacts. The formal representations include design reports, publications, engineering drawings, prototypes and designed artifacts. Tools are required to support the co-authoring of these artifacts, and their maintenance. Cultural artifacts represent new contributions to public or community knowledge. The cultural artifacts themselves evolve through ongoing collaborative knowledge building, and their evolution should also be documented. In engineering, this is usually in the form of design revisions and version management.

7.2 Capabilities of Existing Collaboration Tools

This section surveys the capabilities of existing collaboration tools. A set of generic capabilities is identified by reviewing and combining the research of Mittleman et al. [37] and Büchner et al. [89].

7.2.1 Mittleman's Classification of Existing Groupware Capabilities

Mittleman et al. [37] analyzed over 250 groupware products, and extracted a set of fundamental groupware capabilities found in at least some products. These capabilities are described in Table 7. Few of products provide all of these capabilities, but the best ones provide most of them.

Table 7. Fundamental groupware capabilities (adapted from [37]).

	Capability	Description	Examples
1	**Core functionality**	The primary capability provided by a tool.	The core functionality of Audio Conferencing Tools is to provide a continuous channel for multiple users to send and receive sound.
2	**Content**	Possible content for contributions to a collaboration system.	
	Text	A block of textual information	Skype can only accept text

	Links	Reference pointers with labels	messages, audio-stream, and video stream; SharePoint supports text, links and graphics.
	Graphic	A pictorial image, object or diagram	
	Data-Stream	A continuous data flow (sound channel or desktop sharing)	
	Hypermedia	Combinations of the content types above	
3	**Relationships**	Associations established by a tool based on users intent: collection, list, tree and graph.	The relationships may be syntactic or semantic.
	Collection	Connotes membership in a set of otherwise unrelated objects	Searching for members of "NBA".
	List	A list of an ordered set of objects.	Before/after, bigger/smaller
	Tree	A set of objects in hierarchical relationships with each object (except the root) having only one parent, but having zero-to-many children.	System, subsystem, component.
	Graph	An organization where each object can have zero-to-many links to other objects.	Parents, siblings, children, cousins…
4	**Supported actions**	The system allows users to deal with the content and relationships.	
	Add	Contribute content to the group.	Upload new file to a repository
	Receive	Ability to receive, view, or read contributions to the system.	Obtain new contribution from partners
	Associate	Establish relationships among contributions.	Organize ideas into categories
	Edit	Modify or revise the existing content.	Revise text already contributed to a session
	Move	Change relationships among items.	
	Delete	Remove or erase anything in content.	
	Judge	Render or rate an item, an opinion.	Based on its merits to vote
5	**Action parameters**	Describe characteristics of actions that impact user's experience of contributions and of one another.	
	Synchronicity	Describes expected delay between the time that a user executes an action and the time other users respond on that action.	For e-mail, users don't expect immediate response; while for audio conference, they hope a reply within a very short time.
	Identifiability	Refers the degree to which users can determine who executed an action.	It can be made anonymously or pseudonymously.
6	**Access control**	Restrict the users' rights and privileges to entering a session and making some actions.	For instant message, all users may add, edit or delete their own input, but not to other's.
7	**Session persistence**	To what degree the contributions are temporal or permanent.	Data stream of audio disappears after input in Skype, but messages may exist for long time.
8	**Alert mechanisms**	The capability to interrupt or notify participants of something or someone in	RSS (Really Simple Syndication); IM arriving by making a sound or

		the system demands their attention.	popping up a momentary visual cue to notify receiver;
9	**Awareness indicators**	The ways by which users can know what others have access to a session, the nature of their roles, and their current status.	In some systems, people can learn who is currently active and what he is doing.

Next, they categorized the core functionality of groupware tools into four relatively stable categories based on their primary functionalities: jointly authored pages; streaming technologies; information access technologies; and aggregated systems which combine several technologies (Table 8).

Table 8. Classification of core capabilities (adapted from [37]).

Core capabilities	Description	Examples
Jointly Authored Pages	A shared workspace to which one or multiple participants can contribute, usually simultaneously. The input data structures of pages might include text, graphics, numbers, or other digital objects.	
Conversation tools	Optimized to support dialog among group members.	Email, instant message, blogs, chat rooms, threaded discussions…
Shared editors	Optimized for the joint production of deliverables like documents, spreadsheets, or graphics.	wiki, multi-cursor word processor, multi-cursor whiteboards, Google spreadsheet
Group dynamic tool	Optimized for creating, sustaining, or changing patterns of collaboration among people making joint effort toward a goal.	Some tools for idea generation, idea clarification, idea organization, e.g., GroupSystems Catagorizer…
Polling tools	Optimized for gathering, aggregating, and understanding judgments, opinions, and information from multiple people.	GroupSystems Moodmeter, GroupSystems Vote…
Streaming technologies	Technologies that provide a continuous feed of changing data.	
Desktop/Application	Sharing Optimized for remote viewing and/or control of the computers of other group members.	NetMeeting, VNC, MSN messenger,
Audio Conferencing	Optimized for transmission and receipt of sounds.	freeconferencecall.com, Skype, POTS
Video Conferencing	Optimized for transmission and receipt of dynamic images.	NetMeeting, MSN Messenger
Information Access technologies	Technologies that provide group members with ways to store, share, find, and classify data objects.	

Shared File Repositories	Provide group members with ways to store and share digital files.	Teamware Office, ScanR, eFax
Social Tagging Systems	Provide means to affix keyword tags to digital objects so that users can find objects of interest, and so they can find others with similar interests.	Eurekster, Iijit, Sproose, Wink
Search Engines	Provide means to retrieve relevant digital objects from among vast stores of objects based on search criteria.	Google, Ask, Yahoo
Syndication Tools	Provide notification of when new contributions of interest have been added to pages or repositories.	RSS feed, e.g., Pluck, Bloglines, Firefox, FeedDemon, NetNewsWire, MyYahoo, NewsGator Online/Outlook
Aggregated systems	Technologies that combine of other technologies and tailor them to support a specific kind of task.	SharePoint, BSCW, Microsoft project

7.2.2 Büchner's Collaboration Services Catalog

A similar survey of existing collaboration tools was done by Büchner, Matthes and Neubert [89] in 2008. Their study analyzed several integrated web-based Enterprise 2.0 tools. Enterprise 2.0 is a set of technologies and services providing collaboration and communication services specifically for enterprises. McAfee [90] identified six underlying Enterprise 2.0 technologies: Search, Links, Authoring, Tags, Extensions, and Signals (SLATES). *Search* means that users search for information using search engines rather than following navigation structure; *Links* connect related content, and support searching; *Authoring* allows people to easily contribute new content to an information platform either individually or jointly; *Tags* are metadata including labels and keywords that can be attached by users to categorize and give meaning to content; *Extensions* leverage the other technologies to provide additional services like automatically categorizing content and making recommendations; *Signals* inform users when new content of interest appears. SLATES are not restricted to Enterprise 2.0 tools, but also underlie many Web 2.0 technologies including wikis, blogs, Facebook, and Twitter [90], [91].

However, Büchner et al. [89] argue that SLATES are "fuzzy and not used by all tools the same way", so they are not clear enough to describe or evaluate Enterprise 2.0 tools objectively. Büchner et al. designed a unifying multi-dimensional services catalog and conducted a functional analysis for eight representative tools: Alfresco Share, Atlassian Confluence, GroupSwim, Jive SBS, Liferay Social Office, Microsoft Office SharePoint Server, Socialtext, and Tricia. They categorized the common concepts into three service contexts, 13 service categories and 49 services. The three service contexts are content-centric services, user-centric services and orthogonal services (those services that are neither user-centric nor user-centric are assigned to this class). The service contexts and categories are summarized in Table 9. Furthermore, they evaluated all capabilities of the 8 tools, rating from 0 to 4 in a technical point of view: "0" means no capabilities at all; "4" represents complete coverage of the service they defined. They observed that none of the available tools provided complete coverage of all of the service categories, and each tool had different strengths and weaknesses.

Table 9. Services Catalog (adapted from [89]).

Service context	Service category	Service
Content-centric	**Authoring:** collaborative web-based creation and manipulation of content respectively content objects (e.g., wiki pages, comments, files).	WYSIWYG-Editor (What-You-See-Is-What-You-Get)
		Support for tables, images, and media objects
		Input support for link creation
		Auto save
		Description of all content objects by rich markup text
		Spell checking
		Concurrent editing
		Offline editing
	Link management: dealing with the references among content (e.g., files) and container objects (e.g., wikis, directories).	Human-readable permalinks for all content objects
		Stable URLs for containers and actions
		Labeling of invalid links
		Search for invalid links
		Automatic propagation of link updates
	Tagging	Tag support for all content objects
		Input support for tag creation
		Tag usage overview
		Private Tags
	Search	Full-text search over all content
		Search content of files
		Highlighting of search hits
		Advanced search operators
		sorting
		filtering

	Version management: tracing the evolution of all the content objects within their life-cycle	Safety net through content revisions and audit trail
		Annotation and classification of revisions
		Human readable presentation of revision differences
		Restore
		Access control for versions
		Undelete
	Desktop File Integration: the flexibility of accessing to files in the tool.	File access: additionally of web access, files can be accessed by standardized protocols, like SMB, WebDAV and FTP
		Metadata: Embedded file metadata is adopted and can be accessed and manipulated.
User-centric	Access control	Creation of groups and invitation of new members by users
		Uniform, flexible, and fine granular access control concept for all content types
		Functional groups for access control
		Content of any type may be made available for anonymous users
		Smooth transition between the usage modes not logged on and logged on
		Spam avoidance
	Feedback	Comments to content of any type
		User ratings
		Anonym post of comments
	Social networking	Support for social network building, e.g., inviting others to be a “friend” and the invitation can be accepted or declined.
		Fine granular access control for user profile properties
	awareness	Tracking of other users' activities
		Tracking of activities on content and container objects
		Support for different message channels
	Usage analytics	Usage statistics down to the level of individual content items
		Search words statistics
Orthogonal	Consistent GUI	Consistent presentation of actions and views
	Personalization	Adaptable look & feel for certain functional areas

7.3 Specific Functionality for CKB in Engineering Design

Both Mittleman’s and Büchner’s classifications identify the available capabilities of existing tools, but neither of them comments on whether these capabilities completely match the requirements of collaborative work. The two classification schemes are largely consistent, but with some significant differences.

This section proposes a set of required functionalities to support collaborative engineering teams, as shown in Table 10. The functionalities are based on the core functions identified by Mittleman and Büchner, supplemented by additional functions to fill the gaps. Totally, there are 6 categories of functions in Table 10. The first five will be elaborated respectively in the following sub-sections; the last one is a group of functionalities which have been explained in Table 7, and the name, user-centric, is borrowed from Büchner's catalog. "Syndication" is put under the "alert mechanisms" since they are all about the issues that notify users.

Table 10. Categorizing required functionalities for CKB in engineering design.

Categories	Sub-categories		Description
Content	Text		Refer to Table 7
	Tables and graphs		Refer to Table 7
	Media objects		Audio, video, animation, Flash, etc.
	Sketches and drawings		These are functionalities specifically required for CED, which are not mentioned in other general groupware.
	Mathematics and calculations		
	CAD models		
Supported actions	Add		Refer to Table 7
	Receive		Refer to Table 7
	Move		Refer to Table 7
	Delete		Refer to Table 7
	Judge		Refer to Table 7
	Associate		Build relationships by keywords, links, tags, etc.
	Edit		Including joint-authoring on documents or wikis, etc., and co-edit on CAD models, drawings, etc.
	Search		Combine the features in both Mittleman's and Buchener's schemas, and add semantic search
	Discourse	Annotate	attach a comment, note, or freehand markup to any content object
		Comment /suggest	Comments provide feedback. Suggestions propose an idea, improvement or action.
		Question/ answer	Questions seek answers
		Discuss	Discussion considers or examines by commenting, talking over or writing about, etc., especially to explore solutions.
		Argue	Arguing presents alternative viewpoints to clarify pros and cons of different positions.
		Negotiate	Negotiation seeks to find common ground between different positions.
		Decide	Deciding has the goal of reaching an agreement on

			the best position, concept, idea or action.
Relationships	Collection		Refer to Table 7
	List		Refer to Table 7
	Tree		Same definitions as Mittleman's classification but need display different views of these relationships.
	Graph		
Workspace	Private		Personalized space
	Shared		For articulate personal perspectives and discourse
	Knowledge capture and storage		Facilitate the capturing, recording, sorting and storing of information/knowledge gathered or generated during collaboration
Project management			Calendar, schedule, team forming and workflow, etc.
Users-centric functionality	Access control		Refer to Table 7
	Identifiability		Refer to Table 7
	Alert mechanisms		Alert mechanisms are addressed in Table 7. Syndication belongs to this subcategory, and interpretation is in Table 8.
	Awareness indicators		Refer to Table 7

7.3.1 Content

This category refers to all kinds of contents that can be input to the system. Mittleman's group found that the content types of existing collaboration tools included text, links, graphic, data-stream and hypermedia; Büchner's group believes besides those types, other media objects like video and Flash should be able to be embedded as well by editors of Enterprise 2.0 tools. The author argues this is not an exhaustive list, for engineering domain, sketches, drawings and CAD models are engineers' languages. Engineers cannot work without formulas and calculations either. Engineers generate ideas in their mind, which is tacit knowledge and invisible to others. Therefore, an engineer needs to externalize them firstly and then articulate personal perspectives to other team members. Most groupware can only recognize words, pictures, audio/video channels, or links, while engineers use a mix of knowledge representations.

Users want to input content as conveniently as possible. When users jointly work together from distributed places, everything has to been done by computer. Engineers need to be able to conveniently input sketches and formulas, even do some simple calculations (embedded in the collaboration tool, not the calculators in Windows) by keyboard, mouse or pen (stylus) input. These capabilities are not available in most text-oriented groupware products. One product, Microsoft OneNote, supports freehand sketches and annotations, but some other functionality is weak, such as unorganized co-authoring and comments, etc.

7.3.2 Supported Actions

Supported actions can be applied to all of the above content types. The definitions of add, receive, move, delete and judge in Mittleman's comparison scheme are accepted in this thesis, so they won't be described again (please refer to Table 7). This section will introduce the other actions required by CED, such as associating, editing, searching and discoursing on all contents.

7.3.2.1 Associate

The *associate* action establishes relationships between content objects. The two primary actions are *tagging* and *linking.*

Tags are metadata elements used to describe and classify content objects. *Tags* include user-defined keywords and labels, as well as automatically-generated metadata like timestamps, content author, content type, etc. *Tags* support the relationship types *collection* and *list*. *Tags* can form a bottom-up categorization system. Tags can be based on a pre-defined ontology or taxonomy, or they can be user defined. Most systems allow users to either choose an existing tag, or define a new one. Büchner's group believes the tool should have the capability to show the frequency of tag usage both numerically and visually in the form of a tag cloud [89].

Link is one of the six components in McAfee's SLATES. The most common usage is the web link, which connects one web page to another. The functionality is very helpful for any collaborative systems, but is still too limited. It should be easy and natural to create links between any content objects, at any level of granularity. For example, it should be easy to create a link

1) From an equation in a document to a bookmark in a textbook;

2) From a citation or quotation in a document to a bookmark in an annotated copy of the source reference;
3) From a statement in a report to rough notes in a private or shared notebook;
4) From a geometric feature in a CAD model to a comment or sketch.

7.3.2.2 Edit

Edit refers to the creation and modification of content. A large amount of knowledge or information should be expressed explicitly, documented, recorded, and then maintained throughout the process of collaboration, such as design reports, publications, engineering drawings, prototypes, designed artifacts, etc. Tools are required to support the *co-authoring* and *co-design* of these artifacts.

Co-authoring requires a shared workspace in which one or multiple participants can contribute to the creation of content objects, either synchronously and asynchronously. The services of these tools usually match the common single-user office tools, such as word processor and spreadsheet, but with enhanced attributes. For instance, ideally, if a user wants to edit a document, it should be possible to work on the shared documents without explicitly uploading and downloading files. SharePoint server 2010 allows users to edit the file in place simultaneously, as if it was on a local or network drive. With Google Wave, Google Docs, users can co-edit a document in a shared workspace as well [92]. Identifiability is an important attribute for joint authoring, which makes it easy to view and track the contributions of different participants if desired.

However, if someone wants to edit a file and does not expect interruptions from others or to avoid repeated work, he/she may *check out* the file, then others have to wait or only read until he/she checks it in again. This can also be regarded as an occasion of "awareness"—users may know who else are working in the same file or window as well.

Co-design: engineers and designers need to be able to share and co-design CAD models and drawings, and these are the focus of many collaborative engineering software systems. Required functionality includes at least the following:

1) Interactive, shared viewing of CAD geometry (rotate, zoom in/out, make cross-section, perform limited dimensioning and measuring);
2) Interactive, shared annotation and markup of CAD geometry;
3) Ability to associate relevant forms of discourse with specific elements of a CAD model or drawing, e.g., a discussion about a tolerance or surface finish;
4) Joint creating and editing of CAD models.

Some of the capabilities have been achieved in different systems, which have been mentioned in Section 3.3.1.

7.3.2.3 Search

Search provides a means to retrieve relevant content from vast stores of objects based on search criteria. The system should be able to engage in full-text searching over all content objects, including comments, tags, and the relationships of the content objects. Advanced search supports AND, OR, and NOT operators. In addition, the tool can sort and filter the results. For example, the default is to display all the results by relevance; users can sort them by date or by author's name, etc., or filter them by content type, tags, modification date, and modifier.

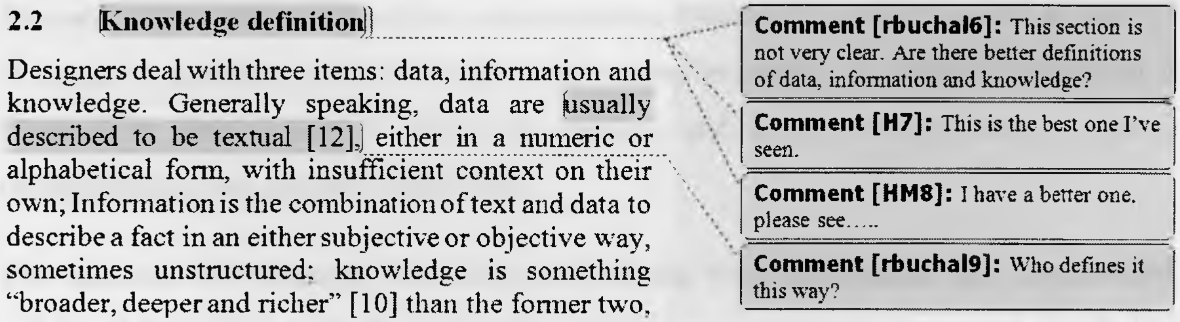
7.3.2.4 Discourse

A number of more specific activities can be identified within the theme of discourse. They include annotating, commenting, suggesting, questioning, discussing, arguing, negotiating and deciding. These functionalities can support users in building shared understanding, discoursing group preference and making decisions, etc.

Annotating is the capability to attach a comment, note, or freehand markup to any content object. This is the digital equivalent of margin notes and red pencil annotations on paper. The important requirement is to place the annotation within the content object itself, rather than separate from it. It is the difference between placing notes at the end of a document, and putting notes in the margins and on top of the document.

It is useful to distinguish different forms of dialog: comments, suggestions, questions, discussions, arguments, and negotiation. Comments provide feedback. Suggestions propose an idea, improvement or action. Questions seek answers. Discussion considers or examines by commenting, talking over or writing about, etc., especially to explore solutions. Arguing presents alternative viewpoints to clarify pros and cons of different positions. Negotiation seeks to find common ground between different positions. Deciding has the goal of reaching an agreement on the best position, concept, idea or action. There is no clear boundary between these activities, and all involve discussions and dialogs.

It should be possible to embed discourse within the evolving knowledge structure, rather than as a separate activity. Tools like Microsoft Word provide some of these capabilities, but some limitations are found too. For instance, Microsoft Word allows users to insert comments into a document, but it does not support embedded discussions. When multiple persons want to discuss the content of a shared Word document, they are forced to use comments which are not threaded or shown in chronological order. What can be done now is like the screen shot shown in Figure 16. In this discussion, three people involved, “H” and “HM” answered the question that “rbuchal” asked and the comment of “H” and “HM” are parallel to rbuchal’s. Moreover, once the comment is created on the side which is default, it cannot be moved any more. However, the ideal display should be listing all the comments in a logical way to clearly see their relationships when they are responding to one topic. In other tools, like Igloo Online Communities [93], this function is even worse - all the comments can only be added in the end of the article chronologically. In OneNote, comments are not physically connected with the targeted content, so the links can be lost if content is rearranged on a page.



2.2 Knowledge definition

Designers deal with three items: data, information and knowledge. Generally speaking, data are usually described to be textual [12], either in a numeric or alphabetical form, with insufficient context on their own; Information is the combination of text and data to describe a fact in an either subjective or objective way, sometimes unstructured; knowledge is something “broader, deeper and richer” [10] than the former two,

Comment [rbuchal6]: This section is not very clear. Are there better definitions of data, information and knowledge?

Comment [H7]: This is the best one I’ve seen.

Comment [HM8]: I have a better one. please see…..

Comment [rbuchal9]: Who defines it this way?

Figure 16. An embedded discussion using Microsoft Word comments.

Comments or threaded discussions should be embedded right beside the content to which they refer and can be moved along with the commented content. Besides that, comments should be threaded and displayed in chronological order. Comments should be identified by reviewer's name, and it should be possible to sort or search the comments by reviewers' names or by time. For example, users may hope to see the comments given by Mr. Right from yesterday to today.

The functionality required for discourse is not specifically mentioned in Büchner's catalog, while it is partly mentioned in the core functionalities of Mittleman's classification scheme. Technologies, like instant messaging, chat rooms, blogs, wikis or threaded discussions are helpful to articulate individual's perspective and establish shared understanding about an issue; some collaboration tools, like ThinkLets [24], [25], GroupSystems [94], can help users to brainstorm, converge, organize and evaluate ideas, and build consensus. GroupSystems has whiteboard, electronic brainstorming, group outliner, topic commenter, vote, alternative analysis and survey.

Stahl discussed the required computer tools to support discourse [80]. Table 11 describes four of them.

Table 11. Computer support for discourse (adapted from [80]).

Discourse activities	Form of computer support
Discuss alternatives	Discussion forum
Argumentation and rationale	Argumentation graph
Clarify meanings	Glossary discussion
Negotiate perspectives	Negotiation support

Discussion forum is an interactive communication system that enables people to respond to notes or questions posted by one another asynchronously. A threaded discussion is required to form a tree of divergent opinions and then converge them to shared understandings and acknowledged ideas.

The structure of a threaded discussion can become very complicated and unstructured, and most discussion tools do not guide users to a conclusion or agreement. Tools like *argumentation graphs* can make the structure of discussion explicit and formalized. Such a function could help individuals understand their knowledge-building process, and "*pointing out where additional evidence is needed or where alternatives have not been explored*" [80].

To construct shared understanding, a *glossary discussion* is needed to clarify and define the meaning of important terms. The discussion can help participants reach a common understanding by exchanging perspectives or negotiating conventions. A glossary is formed as a result.

Most likely, negotiation is the most delicate phase in CKB since all the big differences meet at this point. *Negotiation support* is needed to make explicit all kinds of viewpoints and ensure that they are considered in the negotiation process. During negotiation, multiple perspectives converge to a shared group perspective.

7.3.3 Relationships

Knowledge building requires tools to support linking, relating, classifying, abstracting, summarizing, filtering of many different content objects represented by documents, drawings, notes, comments, etc.

Relationships are the associations users can establish between content objects. Currently, hyperlink or hypertext can connect things, but it usually leads the user to jump from the current window to another window to see the information he/she is interested in. It is possible that after several "jumps", the user gets lost, especially when they are facing some new items. In this situation, if the tool can display a whole picture in one screen to show their relationships, that will be a big improvement.

In this thesis the author accepts the definitions of the four relationships identified by Mittleman et al.: collection, list, tree and graph (see Table 7). However, they are not explained deeply enough. Moreover, in CED people need to not only establish these kinds of relationships, but *show them in different views*. Specifically, content objects and their relationships need to be represented by a network of linked nodes, which can be organized in different ways through views, such as a tree in Figure 17, a graph in Figure 18.

Nodes in Figure 17 and 18 may be questions, ideas, opinions, comments, resources, etc. and they themselves may be expressed by documents, charts, tables, graphics, animations, videos, links to other applications and applets, and so on. A view is a description of relationships among nodes. Nodes can be moved around in views, downward or upward; one node also can be shown in different views from different perspectives.

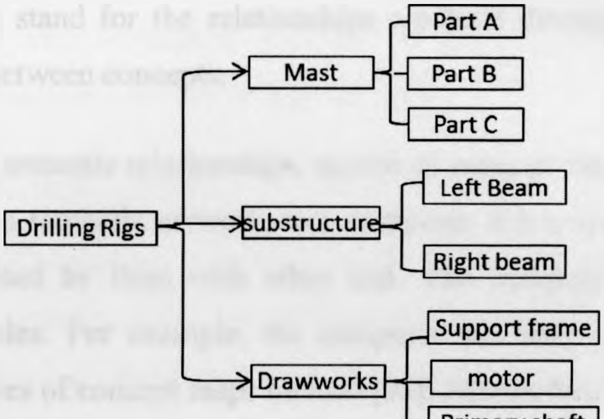


Figure 17. An example of a tree.

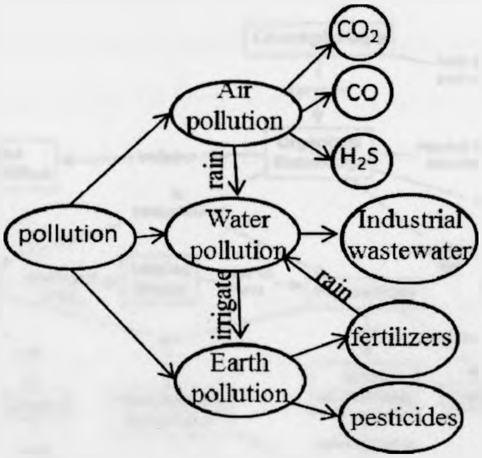


Figure 18. An example of a graph.

The tools should allow users to choose different views to show relationships of contents, such as list or tree, and in a certain display mode, users can choose different levels, such as tree or graph depth. That is, users can select to expand or collapse the trees and graphs, while, for collections and lists, users can choose to display different levels, like abstraction or detail, etc.

Relationships may be either syntactic or semantic. The important difference is that semantic relationships are about meaning, while syntactic relationships are just grammatical rules. Syntactic relationship of the contents in arbitrary sequences of characters is performed through the computation of string similarity. Semantic relationships, instead, stand for the relationships are built through the computation of meaning relatedness between concepts.

Tools should support semantic relationships, similar to concept maps as shown in Figure 19. To a human, it is a semantic network; to a computer, it is a syntactic network – just text in boxes connected by lines with other text. The computer does not know the meanings, just the rules. For example, the computer can only process this based on syntax. The key features of concept maps include [95]: *Hierarchical structure* and *Cross-links.* Hierarchical structure implies that the map is read from top to bottom, and any node may be expanded into lower-level maps. Cross-links allow links from one map to another.

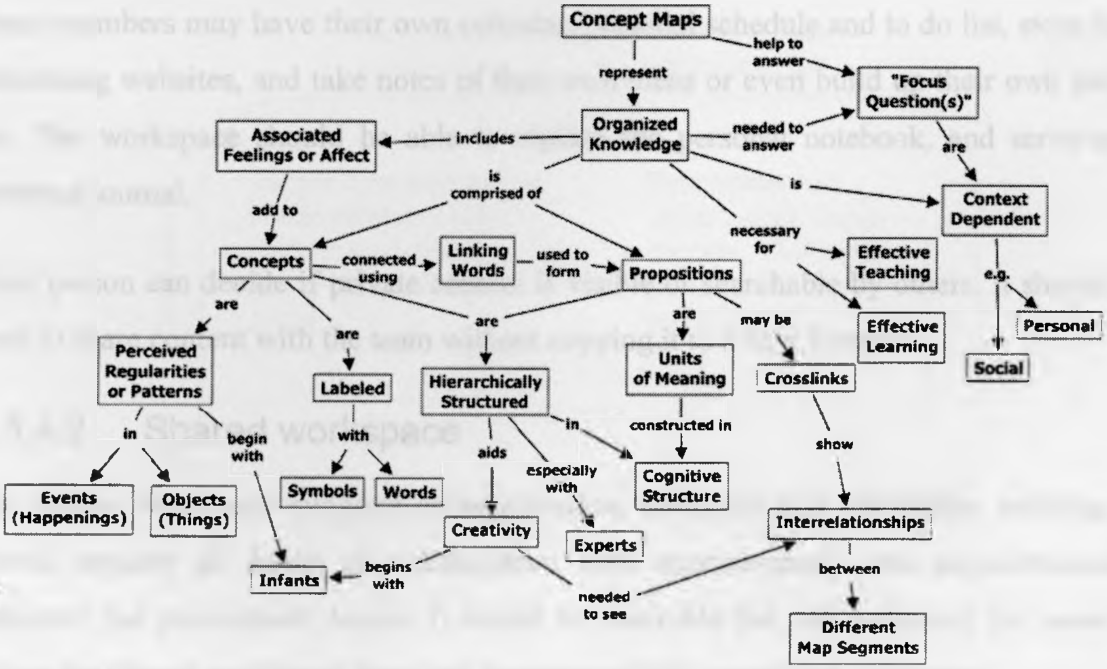


Figure 19. An example of a Concept Map from Wikipedia.

Concept maps are an effective learning tool, and have been implemented in design courses to help students to achieve a higher level of understanding [96], [97]. It is a powerful tool for organizing, associating, integrating ideas, and building knowledge as well.

7.3.4 Workspace

The tools must provide an integrated environment or workspace to support all individual and team activities. The workspace provides a platform for all communication, information sharing, discourse and knowledge building.

7.3.4.1 Private workspaces

Private workspaces are needed for individuals. Each member has a profile which includes contact information, location, short biographical descriptions, etc. The information is very important to create and maintain relationships between team members in a virtual workspace, and help them become acquainted with one another like in a real office.

Individual task work can be captured and stored in this area. Furthermore, individuals can customize the space style (e.g., interface) according to their own interests and intentions. Team members may have their own calendar, personal schedule and to do list, store their interesting websites, and take notes of their own ideas or even build up their own blogs, etc. The workspace should be able to replace the personal notebook, and serve as a personal journal.

Each person can decide if private content is visible or searchable by others. It should be easy to share content with the team without copying it to a new location.

7.3.4.2 Shared workspace

The shared workspace supports communication, discourse and knowledge building. It should support all kinds of collaboration both synchronously and asynchronously, wherever the participants locate. It would be desirable but not necessary for users to access the shared workspace from any computer which connecting to internet.

The well-known Time/Space Matrix of CSCW [98], lists all the possible ways that people could collaborate. Figure 20 is adapted version for CED. They can be collocated (same place) or distributed (different place). The collaboration could be synchronous (same time) or asynchronous (different time). Collaboration tools should support collaboration in all the four quadrants of the grid.

Synchronous communication: Tools are needed to enable participants communicate by text, voice or images in real time from different places. With the help of tools such as instant messaging, videoconferencing, shared whiteboards, application sharing, and electronic meeting and decision rooms, designers can discuss and communicate like they are in same office. For the collaboration of same time/same place, tools are needed as well, tool like ThinkLets can support generating ideas, organizing ideas and voting in one meeting room, etc.; tool like OneNote is helpful to record meeting notes, etc.

Asynchronous communication: Many collaborative activities do not require team members to interact simultaneously, and often asynchronous communication is preferable. Common asynchronous communication technologies include emails, discussion forums, blogs, etc.

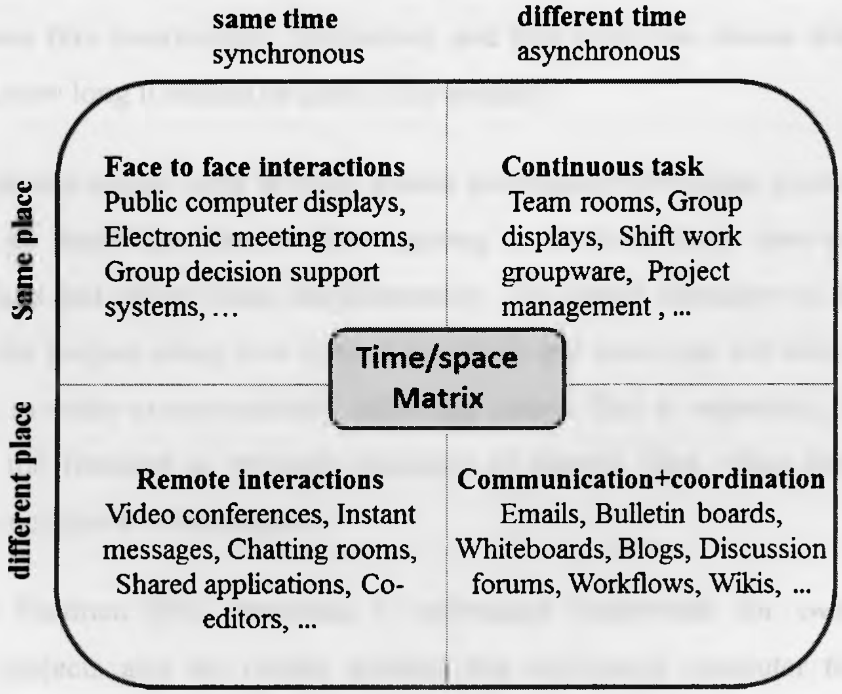


Figure 20. Time-Space Matrix for CED (adapted from [98]).

7.3.4.3 Knowledge capture and storage

During collaboration of the team members, lots of information and knowledge is gathered or generated. The shared workspace must facilitate the capturing, recording, sorting and storing of them. In particular, it must capture and store all discourse process, design history, rationale, etc. Also it serves as a file repository for documents, spreadsheets, pictures, drawings, audio and video files, and so on.

In traditional engineering design, only the design outcomes are recorded, and important knowledge including rationale, arguments, alternative concepts, etc. are not captured except in the personal notes and memories of the participants. This key knowledge is difficult to access or reuse in the future, and over time it is lost as people leave or forget. The informal knowledge generated during CKB has great value as a future resource, and should be preserved and made it searchable later. For instance, during the process of building common understanding, discoursing group preference and making decision, a large number of discussion, feedback, comments, questions and answers are necessary to be kept in history record, because the design rationale evolves in these activities and it is valuable to check out when it is needed. It is useful or even essential to keep track of modifications of each file--who made what changes and when. Of course, it is also helpful to make this functionality customized, and then users can choose what should be recorded and how long it should be kept in the memory.

The system should enable users to easily access information they need, to recognize if the content will be useful for them before opening it, to disseminate new documents to relevant persons and inform them simultaneously. The shared repository is able to notify everyone in the project when new content is added, and users can tell who is retrieving and using it. In order to conveniently index and search files in repository, the software should have the function to structure metadata of electric files, when they are firstly stored in the repository of workspace.

Hameri and Puttinen [99] presented a web-based framework for two distributed engineering projects and the results showed that web-based computer tool not only advanced their punctuality, cost control and workflow, but also accumulated large amounts of information and new knowledge while the project progressed. This information can be not only used to refine the organization and focus on the truly value-adding activities, but saved as organization memory so that many years later, new employees can still find out “why and how”.

7.3.5 Project Management

Traditional project management aims at improving efficiency and works on coordination rather than collaboration. However, collaboration relies on coordination and cooperation, so project management functions are also required for CED. Other researchers like Mittleman et al. [37], Kwintiana Ane [100], and Shen [8] agree that project management is a necessary part in collaborative engineering. When an engineering project is started, the first tasks are to form the teams, set up schedules, manage resources, make organizational charts and work regulations, etc. The tool should support project managers to do project tracking, control and in particular plan-adjustment and evaluation. Certainly, don’t forget to set the common goal for the collaborative teams.

Team forming: usually, an engineering project involves several teams that bring different backgrounds and expertise for the task. The system should display all the team members’ names and who is the corresponding person in a group. Job descriptions of a team and each person should be elaborated clearly. In addition, team goals, primary team interaction procedure, and behavior criteria, etc. should be declared in this part. Collaborative behaviors of the teams are vital and must be carefully organized to guide the team members to engage in the collaboration.

Workflow: the system should be able to manage and define a series of tasks sequences to produce final outcomes. So, for example, in a manufacturing setting, a design document might be automatically routed from designer to a technical director then to the production engineer. At each step of the workflow, each person or a group is responsible for a specific task. Once the task is complete, the tool is capable to inform the individuals responsible for the next task and ensures that they receive the data they need to execute their subtasks of the process. In addition, the system should ensure uncompleted tasks are followed up. Though, an efficient workflow is very useful, it does not mean all the tasks have to follow certain procedures.

8 Conclusions and Future work

Computer supported collaboration in engineering design has become inevitable because of the increasing complexity of design tasks and distributed multidisciplinary design teams. The thesis conducts a broad literature review to explore the essence of collaborative engineering design. It can be concluded that CED incorporates many generic design activities; each activity can be modeled as a knowledge transformation process; and the whole design process is full of knowledge building activities. Computer tools are needed to support the whole process of collaborative design. Unfortunately, after the existing tools for collaboration are reviewed, there is no evidence that the capabilities of tools fully correspond to the demands of collaborative knowledge building settings.

The thesis proposes an integrated CKB-orientated model of CED for both engineering design and learning. The model integrates and extends key ideas and elements from different fields: Stahl's and Singh's CKB model of collaborative learning; Lu's ECN-based model of collaborative engineering; Nonaka's SECI model of organizational knowledge creation; and Sim and Duffy's model of engineering design activities. The purpose of the model is to describe and explain how knowledge is created in collaborative engineering design.

Based on the new model, a set of specific functionalities for CKB in collaborative engineering design are elaborated. These are grouped into six categories: content, supported actions, relationships, workspace, project management and user-centric functionality. To support CED, an integrated collaboration environment must recognize all kinds of content types, including sketches, drawings, formulas and calculations; it must support creation of links among any content objects, at any level of granularity; it should allow different relationships among associated objects to be viewed in the form of graphs; it should support full-text searching over all content objects, including comments, tags, and the relationships of the content objects; and it should capture and store all the information and knowledge created during discourse.

Further work is required to refine and validate the CKB-oriented model, and to experimentally validate that the proposed collaboration environment supports CED. Experimental validation of the environment is problematic, as no integrated environment currently exists, and using tools that satisfy only a subset of the requirements may not lead to valid conclusions.

References

---

[1] Robin, V., Rose, B., and Girard, P., “Modelling collaborative knowledge to support engineering design project manager,” *Computer in Industry*, vol. 58, pp. 188-198, 2007.

[2] Lu, S. C-Y. , Elmaraghy, W. and Schuh, G. “A Scientific Foundation of Collaborative Engineering,” *Annals of the CIRP*, vol. 56/2, pp. 605-634, 2007.

[3] VIVACE, “Collaborative Methods and Tools”, VIVACE Forum, CERFACS, EADS, Warwick, U.K., September, 2005.

[4] Marsh, J. R., “The Capture and Utilisation of Experience in Engineering Design,” PhD dissertation, Department of Engineering Cambridge University, Cambridge, 1997.

[5] Stewart, T.A., *Intellectual capital: the new wealth of organization*, Doubleday/ Currency, 1997.

[6] Vijaykumar, G. and Chakrabarti, A., “Understanding the Knowledge Needs of Designers During Design Process in Industry,” *Journal of Computing and Information Science in Engineering*, vol. 8/1, pp. 11004-12, 2008.

[7] Badke-Schaub, P. and Frankenberger, E., “Design representations in critical situations of product development”, in *Proceedings of the 4th Design Thinking, Design Representation Symposium*, 1999.

[8] Shen, W., Hao, Q., and Li, W., “Computer Supported Collaborative Design: Retrospective and perspective,” *Computers in Industry*, vol. 59, pp. 855–862, 2008.

[9] Li, W. D. and Qi, Z. M., “State-of-the-art technologies and methodologies for collaborative product development systems,” *International Journal of Production Research*, Vol. 44/13, pp. 2525 — 2559, 2006.

[10] Smith, S. S.-F., “An Evaluation of Internet-Based CAD Collaboration Tools,” *The Journal of Technology Studies*, pp. 79-85, 2004.

[11] Crabtree, R. A., Fox, M. S. and Baid, N. K., “Case Studies of Coordination Activities and Problems in Collaborative Design,” *Research in Engineering Design,* vol. 9, pp.70-84, 1997.

[12] Ertas, A. and Jones, J., *The Engineering Design Process,* New York, N.Y., John Wiley & Sons, Inc. 1996.

---

[13] Eder, W. E. and Hosnedl, S., *Design Engineering: A Manual for Enhanced Creativity*, Boca Raton: CRC Press. 2008.

[14] Dieter, G. E. and Schmidt, L. C., *Engineering design*, Boston : McGraw-Hill Higher Education,1983.

[15] Pahl, G., and Beitz, W., *Engineering design*, The Design Council. Springer, Berlin Heidelberg New York, 1996.

[16] Hubka, V, and Eder, W.E. *Design Science*. Springer, Berlin Heidelberg New York, 1996.

[17] Ullman, D. G., *The Mechanical Design Process*, Boston: McGraw-Hill Higher Education, 2010.

[18] Sim, S.K., and Duffy A.H.B., "Towards an Ontology of Generic Engineering Design Activities," *Research in Engineering Design,* vol.14, pp. 200–223, 2003.

[19] Reddy, J.M., Finger, S., Konda, S. and Subrahmanian, E., "Design as Building and Reusing Artifact Theories: Understanding and Supporting Growth Of Design Knowledge," *The Design Productivity Debate*, A.H.B. Duffy, Ed. Springer Verlag, 1998.

[20] Hicks, B.J., Culley, S.J. , Allen, R.D. and Mullineux, G., "A Framework for The Requirements of Capturing, Storing And Reusing Information and Knowledge in Engineering Design," *International Journal of Information Management*, vol. 22, 263–280, 2002.

[21] Sim, S.K., Duffy A.H.B. (2004). Evolving a model of learning in design. *Research in Engineering Design,* vol.15, pp.40-61, 2004.

[22] Hubka, V., *Practical studies in systematic design*. UK: ButterWorth Scientific Co., 1988.

[23] Pugh S., *Total design: Integrated Methods for Successful Product Engineering*, Addison Wesley, Wokingham, 1991.

[24] Briggs, R.O., Kolfschoten, G.L., Vreede, G.J. De, and Dean, D.L., "Defining Key Concepts for Collaboration Engineering," *AMCIS Proceedings*, 2006.

[25] De Vreede, G-J., Kolfschoten, G.L., and Briggs, R.O., "ThinkLets: a Collaboration Engineering Pattern Language," *International Journal of Computer Applications in Technology*, vol. 25, Nos. 2/3, 2006.

[26] Denise, L., "Collaboration vs. C-Three (Cooperation, Coordination, and Communication)," Innovating, vol. 7/3, 2007.

[27] Merriam-Webster online *dictionary*, http://www.merriam-webster.com/.

[28] Alberts, D.S., Garstka, J.J., Hayes, R.E., and Signori, D.A., *Understanding Information Age Warfare*, Command and Control Research Program, 2001.

[29] Noble, D., "A Cognitive Description of Collaboration and Coordination to Help Teams Identify and Fix Problems," in *Proceedings of the 2002 International Command and Control Research Technology Symposium*, 2002.

[30] Michael Schrage, *Shared Minds*, NY: Random House, 1990.

[31] Jin, Y. and Geslin, M., "Argumentation-based Negotiation for Collaborative Engineering Design," *International Journal of Collaborative Engineering*, vol. 1, January/February 2009.

[32] Turino, J., *Managing Concurrent Engineering*, Van Nostrand Reinhold, New York, 1992.

[33] Kamrani,A.K., "Collaborative Design Approach in Product Design and Development", *Collaborative Engineering: Theory and Practice*, Kamrani,A.K., and Nasr , E.A. (eds.), Springer Science + Business Media, LLC, 2008.

[34] International Journal of Collaborative Engineering (IJCE), Publisher: InderScience. http://www.inderscience.com

[35] Wilson, P., *Computer Supported Cooperative Work: An Introduction*. Online version1991

[36] Sprow, E., "Chrysler's Concurrent Engineering Challenge," *Manufacturing Engineering*, vol. 108/4, pp. 35-42, 1992.

[37] Mittleman, D.D., Briggs, R.O., Murphy, J. and Davis, A., "Toward a Taxonomy of Groupware Technologies," *Collaboration Researchers' International Workshop on Groupware*, LNCS 5411, pp. 305–317, 2008.

[38] Monge, P. R. and Contractor, N. S., *Theories of Communication Networks*, Oxford University Press, 2003.

[39] Thurston, D. L., "Real and Misconceived Limitations to Decision Based Design with Utility Analysis," *Journal of Mechanical Design*, *Transactions of the ASME*, 123/2: 176-182, 2001.

[40] Fudenberg, D. and Tirole, J., *Game Theory*. MIT Press, Cambridge, MA, 1983.

[41] Olson, M., *The Logic of Collective Action: Public Goods and the Theory of Groups*, Harvard Economic Studies1995.

[42] Aronson, E., Wilson, T. D., and Akert, R., *Social Psychology*, Prentice Hall, 2006.

---

[43] Fishburn, P. C., *The Theory of Social Choice*, Princeton University Press, Princeton, N.J., 1973.

[44] Carey, J. M., Kacmar, C. J., "The Impact of Communication Mode and Task Complexity on Small Groups Performance & Member Satisfaction," *Computers in Human Behavior*, 13/1: 23-49, 1997.

[45] Hammond, J., Koubek, R.J., and Harvey, C.M., "Distributed Collaboration for Engineering Design: A Review and Reappraisal," *Human Factors and Ergonomics in Manufacturing*, vol. 11/1, pp. 35–52, 2001.

[46] Cyert, R. M. and March, J. G., *A Behavioral Theory of the Firm*, Blackwell. (online) 1992

[47] Arias, E. E., H., Fischer, G., Gorman, A. and Scharff, E., "Transcending the Individual Human Mind: Creating Shared Understanding through Collaborative Design," *ACM Transactions on Computer-Human Interaction*, vol.7/1, pp. 84-113, 2000.

[48] Nonaka I., Takeuchi H., *The Knowledge Creating Company: How Japanese Companies Create the Dynamics Of Innovation*, Oxford University Press, Oxford. 1995.

[49] Karadsheh, L., Mansour, E., Alhawari, S., Azar, G., and El-Bathy, N., "A Theoretical Framework for Knowledge Management Process: Towards Improving Knowledge Performance," *Communications of the IBIMA*, vol. 7, 2009.

[50] Awad, E. M., and Ghaziri, H. M., *Knowledge Management*, Prentice Hall, 2003.

[51] Malhotra, Y. and Galletta, D. A., "Multidimensional Commitment Model of Volitional Systems Adoption and Usage Behavior," *Journal of Management Information Systems*, vol. 22, pp. 117-151, 2005.

[52] Alavi, M., and Leidner, D., "Review: Knowledge Management and Knowledge Management System: Conceptual Foundations and Research Issues," *MIS Quarterly*, vol. 25/1, pp. 107-136, 2001.

[53] Zhen, L., Jiang, Z. and Song, H., "Distributed Knowledge Sharing for Collaborative Product Development," *International Journal of Production Research*, vol.49/10, pp. 2959-2976, 2011.

[54] Ouertani, M.Z., Baina, S., Gzara, L. and Morel, G., "Traceability and Management of Dispersed Product Knowledge during Design and Manufacturing," *Computer-Aided Design*, vol. 43/5, pp. 546-562, May 2011.

[55] Mcmahon, C., Lowe, A. and Culley, S., “Knowledge Management in Engineering Design: Personalization and Codification,” *Journal of Engineering Design*, vol. 15/4, pp. 307-325, 2004.

[56] Mezher, T., Abdul-Malak, M.A. et al. (). Building a Knowledge Management System in a Design Firm: the case of XYZ Structural Department. Journal of Cases on Information technology, Vol. 11, Issue 3, pp.1-17, 2009.

[57] Peachey, T. and Hall, D., “Knowledge Management and the Leading IS Journals: An Analysis of Trends and Gaps in Published Research System Sciences,” *International Journal of Knowledge Management*, vol.1/3, pp.55-69, 2005.

[58] Sun, Z., and Gao, G., “HSM: A Hierarchical Spiral Model for Knowledge Management,” In *Proceedings the 2nd International Conference on Information Management and Business*, 2006, pp. 542-551.

[59] Bouthillier, F., and Shearer, K. "Understanding Knowledge Management and Information Management: The Need For An Empirical Perspective," *Information Research Journal*, vol.8/1, 2002.

[60] Zhang, L., Ren, S., Jiang, X., and Liu, Z., “Knowledge Management and its Application Model in Enterprise Information Systems,” *International Symposium on Technology and Society*, IEEE, pp. 287 -292, 2000.

[61] Kahn, B., and Adams, E., “Sales Forecasting as a Knowledge Management process,” *The Journal of Business Forecasting*, pp. 19-22, 2000.

[62] Ahmed S., Blessing L. and Wallace K., “The Relationships between Data Information and Knowledge based on a Preliminary Study of Engineering Designers,” *ASME Design Theory and Methodology*, 1999.

[63] Tuomi, I., “Data is More Than Knowledge: Implications of the Reversed Hierarchy for Knowledge Management and Organizational Memory,” in *Proceedings of the Thirty-Second Hawaii International Conference on Systems Sciences, IEEE Computer Society Press*, vol. 16/3, pp. 107-121, 1999.

[64] Ropohl, G., “Knowledge Types in Technology,” *International Journal of Technology and Design Education*, vol. 7, pp. 65–72, 1997.

[65] Houkes, W., “The Nature of Technological Knowledge,” in *Philosophy of Technology and Engineering Sciences*, Anthonie W.M. Meijers , Ed. Publisher: Amsterdam, Boston: North Holland, 2009.

[66] Broens,R. and de Vries, M.J., “Classifying Technological Knowledge for Presentation to Mechanical Engineering Designers,” *Elsevier Science Ltd.* Pp. 457-471, 2003.

---

[67] Vincenti, W. G., *What Engineers Know and How They Know It*. Johns Hopkins Press, Baltimore, 1990.

[68] De Vries, M.J., "The Nature of Technological Knowledge: Extending Empirically Informed Studies into What Engineers Know," *Techné*: vol. 3, 2003.

[69] Bayazit, N., "Designing: Design Knowledge: Design Research: Related Sciences," in *Design Methodology and Relationships with Science*, de Vries, M J, Cross, N. and Grant, D. P., Eds. Kluwer Academic Publishers, Dordrecht, pp 121–136, 1993.

[70] Davis, R., "What is a knowledge representation?" *AI magazine*, vol.14/1, pp. 17-33, 1993.

[71] Davidson, M., Dove, L. and Weltz, J. "Mental models and Usability," Laura Dove, 1999. Online version: www.lauradove.info /reports/ mental models.htm.

[72] Nonaka, I., "A Dynamic Theory of Organizational Knowledge Creation," *Organization Science*, vol. 5/1, pp. 14-37, February 1994.

[73] Johnson-Laird, P. N., "Mental Models," in *the Foundations of Cognitive Science,* Posner, M., Ed, the MIT Press, 1989.

[74] Scardamalia, M., and Bereiter, C., "Computer support for Knowledge-building Communities," *the Journal of the Learning Sciences*, vol. 3/3, pp. 265-283, 1994.

[75] Scardamalia, M., "Collective Cognitive Responsibility for the Advancement of Knowledge," in *Liberal Education in a Knowledge Society*, Smith, B., Ed. Chicago: Open Court, pp. 67-98, 2002.

[76] Scardamalia, M., and Bereiter, C., "Knowledge Building," in *Encyclopaedia of Education*, New York: Macmillan Reference, USA, pp. 1370-1373, 2003.

[77] Singh, G., Hawkins, L. and Whymark, G., "An Integrated Model of Collaborative Knowledge Building," *Interdisciplinary Journal of Knowledge and Learning Objects*, vol. 3, pp. 85-105, January 2007.

[78] Scardamalia, M. and Bereiter, C., "Knowledge Building: Theory, Pedagogy, and Technology," in *Cambridge Handbook of the Learning Sciences,* Sawyer, K., Ed. New York: Cambridge University Press, pp. 97-118, 2006.

[79] Stahl, G., "A Model of Collaborative Knowledge-building," in *Proceedings of Fourth International Conference of the Learning Sciences*, Ann Arbor, MI, 2000.

[80] Stahl, G., *Group cognition: Computer Support for Building Collaborative Knowledge*, MIT Press, Cambridge, MA, 2006.

---

[81] Bereiter, C., *Education and Mind in the Knowledge age*, Mahwah: NJ: Lawrence Erlbaum Associates, 2002.

[82] Singh, G., Hawkins, L. and Whymark, G., "Collaborative Knowledge Building Process: An Activity Theory Analysis," *The Journal of Information and Knowledge Management Systems*, vol. 39/3, pp. 223-241, 2009.

[83] Dubberly, H. and Evenson, S., "Design as Learning – or "Knowledge Creation" – the SECI Model," *Interactions*, pp. 75-79, Jan.+Feb., 2011.

[84] Wu, Z. and Duffy, A.H.B., "Modeling Collective Learning in Design," *Artificial Intelligence for Engineering Design, Analysis and Manufactureing*. Vol.18, pp. 289-313, 2004.

[85] Mcclure, M., "A Case of Document - and Organizationwide – Collaboration," *EContent*, vol. 31/8, pp. 45-47, 2008.

[86] Barber, C.G., Haque, N., and Gardner, B., "'OnePoint' - combining OneNote and SharePoint to facilitate knowledge transfer," *Drug Discovery Today*, vol. 14/17-18, pp. 845-850, 2009.

[87] Diffin, J., and Chirombo, F., "A point to share: Streamlining access services workflow through online collaboration, communication, and storage with Microsoft SharePoint," *Journal of Web Librarianship*, vol. 4/ 2, pp. 225-237, 2010.

[88] Brooks, F.P., "The Computer Scientist as Toolsmith II," *Communications of the ACM*, vol. 39/3, pp. 61-68, 1996.

[89] Büchner, T., Matthes, F., and Neubert, C. "Functional Analysis of Enterprise 2.0 Tools: A Services Catelog," in *IC3K 2009*, CCIS 128, Fred, A. et al. Eds. Springer-Verlag Berlin Heidelberg, pp. 351–363, 2011.

[90] McAfee, A. P., "Enterprise 2.0: the Dawn of Emergent Collaboration," *MIT Sloan management Review*, Spring 2006.

[91] McAfee, A. P., *Enterprise 2.0: New Collaborative Tools for Your Organization's Toughest Challenges*, Harvard Business Press, 2009, http://andrewmcafee.org/2009/06/chapter-1-of-enterprise-2/

[92] Google Wave, http://wavety.com/google-wave-edit-google-docs/, accessed July, 2011.

[93] Igloo Online Communities, http://www.igloosoftware.com/company/communities, accessed July, 2011.

[94] GroupSystems EMS software, http://www.groupvision.ch/3_products_services/3f_system_overview_e.htm, accessed July, 2011.

[95] Novak, J. D. and Cañas, A. J., "The Theory Underlying Concept Maps and How to Construct and Use Them," Technical Report IHMC CmapTools, 2008 http://cmap.ihmc.us/Publications/ResearchPapers/TheoryUnderlyingConceptMaps.pdf

[96] Cornwell, P.J., Concept Maps in the Mechanical Engineering Curriculum, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.136.1198

[97] Sims-Knight, J. E. et al. "Using Concept Maps to Assess Design Process Knowledge," in *34th ASEE/IEEE Frontiers in Education Conference*, F1G 6-10, 2004.

[98] Johansen, R., *Groupware: Computer Support for Business Teams*, New York: The Free Press, 1988.

[99] Hameri, A.-P. and Puittinen, R., "WWW-enabled Knowledge Management for Distributed Engineering Projects," *Computers in Industry*, vol. 50, pp.165–177, 2003.

[100] Kwintiana Ane, B., "Tutorial on Integrated Web- And Agent-Based Computer Supported Collaborative Design," presented at the Fourth International Conference on Advances in Computer-Humans Interaction, 2011.