J Intell Manuf
DOI 10.1007/s10845-011-0607-4

Formalization of design chain management using environment-based design (EBD) theory

Xuan Sun · Yong Zeng · Wei Liu

Received: 19 December 2009 / Accepted: 24 April 2011

Abstract This paper aims to develop a formal conceptual model for the design chain management (DCM) from its informal definition by using the Environment-Based Design (EBD) theory. This effort is different from the existing approaches to developing conceptual models in that the model is derived step by step from a natural language description of the design chain management. The derived DCM model is called a wheel model since the model would evolve because of its internal conflicts. An industrial example is used to show how the proposed formalization can be used to solve a problem in the DCM.

Keywords Design chain management · Formalization · Environment-based design (EBD) · Conceptual model · Wheel model

Introduction

It is fairly well known that up to 80% of product cost is determined by the decisions made in the early design stages; therefore, effective management of product design plays an important role in the entire product lifecycle. In the last two decades, product design activities have been moving to a collaborative environment across extended enterprises, due to the advancement of technologies, the globalization of markets, the segregation of customer demands, and the competition at home and abroad. Collaborative Product Development (CPD) has thus become an operational model for product R&D activities in a wide range of industries (Wang and Hwang 2005). By properly leveraging and synergizing knowledge, technologies, and resources among all the collaborators in product development, the CPD approach may effectively reduce the cycle time and cost while maintaining quality in the development of complex products (Venkatachalam et al. 1993; Chokshi and McFarlane 2008). The application of CPD has resulted in the disaggregation of vertically integrated industry into a multi-layered network of horizontal suppliers and partners, each of which can concentrate on their core competences for certain features and functions of a product. The CPD aims to develop and manufacture innovative products for the fulfillment of new market requirements through integrating all necessary skills and competences across the boundaries between the collaborating organizations and enterprises. Subsequently, an increasingly complex design chain for the development of highly sophisticated product is widely seen in various industrial sectors such as aerospace, automobile, and telecommunication (López-Ortega and Ramírez-Hernández 2007; Mun et al. 2011).

Clark (1991) and Twigg (1997a,b) positioned a design chain as the relationship between a product assembler and its part suppliers. Poirier and Reiter adapted the concept of design chain to product development chain, which is a system through which organizations develop products and services to meet customer requirements (Poirier and Reiter 1996). O'Grady indicated that the product development chain encompasses product assembler, the suppliers and the customers (O'Grady and Chuang 2001). There may be a considerable mesh of suppliers to suppliers, called second-tier suppliers, third-tier suppliers and so on (O'Grady and Chuang 2001; Muckenhirn and Meier 2008). It is generally

X. Sun · Y. Zeng (✉) · W. Liu
Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada
e-mail: zeng@ciise.concordia.ca

W. Liu
Department of Management, Northwestern Polytechnical University, Xi'an, China

Published online: 08 December 2011

agreed that design chain involves participants throughout the product development process, from concept, detail engineering, process engineering, prototype manufacturing, through to post-launch activities (Twigg 1998). Each participant, both internal and external to a focal firm, contributes their capabilities (knowledge and expertise) necessary for design and development of a product (Twigg 1997a,b).

Effective design chain management is of paramount importance for industries to develop innovative and high quality products within a reduced lead-time and cost (CIMdata 2006; Ajoku 2007). Research in this field is mainly concerned with the early involvement of suppliers in new product development (Carlisle 1989; Dowlatshahi 1998; Akira 2001), selecting suitable design chain partners for co-development success (Lin 2003; Nagarajan et al. 2004; Wang and Lin 2006), the design processes collaboration in the design chain (Choi et al. 2005; Shiau and Wee 2008), and the cross-industry diagnostic tool for design-chain management- Design Chain Operation Reference-model (DCOR) (Supply-Chain-Council 2004) as well as its extension (Wu et al. 2007). All those research results come from researchers' ad-hoc observations. In order to develop more effective tools to support design chain management, the design chain management (DCM) needs to be modeled based on a formal foundation.

This paper intends to develop a formal conceptual model of design chain management through the formalization of its informal and generic description. This task is taken as a design problem. We will use a new design methodology—Environment-based design (EBD)—to solve this ill-defined problem.

The rest of this paper is organized as follows. In "Literature review", we discuss the relevance of DCM with collaborative product development and supply chain management, with a particular focus on the review of previous reference models of DCM. Section "EBD" introduces a scientific approach of design: EBD, as the formalization methodology. Starting with an informal description of design chain management, Sect. "Formalization of design chain management" formalizes the DCM into a conceptual model following the EBD theory. Section "Refinement of DCM conceptual model" refines the design chain conceptual model by further considering the details of its parts. An example is given in Sect. "Example" to show how the proposed formalization approach works. In the final section, we summarize our study, discuss the limitations, and suggest future directions.

Literature review

In this section, we will first review the basic concepts of collaborative product development (CPD), supply chain management (SCM), and design chain management (DCM), which provide the context for the research presented in this paper. Then we will introduce existing reference models of design chain management, particularly the Design Chain Operation Reference (DCOR) model.

CPD, SCM and DCM

Design chain management (DCM), collaborative product development (CPD) and supply chain management (SCM) all involve product, process, data, and organization. It would provide a better focus for the research on managing design chains by distinguishing these three acronyms—DCM, CPD and SCM,

Collaborative product development

The increasing competition in the globalized market has led collaborative product development (CPD) to become an inevitable trend for today's new product development (NPD). Organizations have to collaborate more with their suppliers, their customers, and other relevant parties in the current business environment, to respond rapidly to key customer needs, market opportunities, and technology changes, and to reduce the cost, time to market and development risk (Littler et al. 1995). Collaboration issues (Littler et al. 1995), supplier involvement (Dowlatshahi 1998), information technology (Huang et al. 2000; Wang et al. 2002), and theoretical modeling (Case and Lu 1996) are main streams in this research. According to CIMdata, "collaborative product development" includes "collaborative product design" and "design chain management" while DCM and CPD place different emphasis on the objects, goals and efforts (CIMdata 2006). DCM focuses on the design chain and aims to improve the performance of design chain. It is committed not only to planning the organization-wise distributed design processes but also to managing the involved participants and the relationship among those participants during the life cycle of the design chain. The design chain lifecycle can be generally separated into formation, operation and extinction stages. Nevertheless, CPD focuses on product and aims to help manufacturers reduce costs, shrink time-to-market, and deliver products that best meet the customer demands. It provides a platform to different stakeholders for them to work together on the development of a product through sharing information and resources, organizing development process and integrating business systems during the life of a product (Kiritsis et al. 2003).

Supply chain management

The term supply chain management (SCM) was originally introduced by consultants in the early 1980s (Oliver and Webber 1992), which has subsequently gained tremendous attention. Lambert et al. (1998) defined SCM as the integration of business processes from end user through original suppliers that provide products, services, and information to add values for customers. The supply-chain council (SCC) proposed a supply chain operations reference model (SCOR) for benchmarking supply chain processes and designing IT solutions for SCM (Stewart 1997). Cooper et al. (1997) identified seven business processes within supply chain management. Although the consensus remains that SCM is more than simple logistics, most supply chain literature examined procurement and value-adding activities, without explicitly defining product development as its part (Twigg 1998). However, most of the product's competitive characteristics and costs are determined and committed by the activities of the design chain (Lambert and Cooper 2000). Focus will necessarily shift in the early product development process (Twigg 1998). Furthermore, though SCM can lead to a significant reduction in production costs, it does not address other important factors that drive product competitiveness, especially innovation and time-to-market. Therefore, design chain management is becoming as important, if not more, as the logistics and production supply chain. While design chain is a subset of supply chain (Shiau and Wee 2008), managing the design chain is more challenging than managing the other components of a supply chain due to its inherent complexity such as diverse design processes, dynamic design environments, higher task uncertainty, more complex information (often in an incomplete form), and new buyer–supplier relationship (Twigg 1998; Shiau and Wee 2008).

In summary, the relationship among design chain management (DCM), collaborative product development (CPD), and supply chain management (SCM) can be illustrated in Fig. 1. Three pillars support the design chain management: product, design process, and chain management. The ultimate objective of design chain management is to lower R&D costs, reduce time to market and support product or technology innovation.

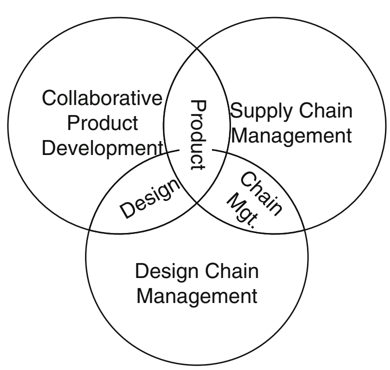


Fig. 1 Relationships among DCM, CPD and SCM

Design chain operations reference (DCOR) model

Respective design chain management reference models have been developed to tackle design process and operations (Supply-Chain-Council 2004; Choi et al. 2005; Wu et al. 2007), design authorities distribution (Twigg 1998), supplier performance measurement (Nagarajan et al. 2004), collaboration strategy (Chuang and Yang 2004), process change management (Shiau and Wee 2008), process maturity model (Fraser et al. 2003). The proposed models and corresponding objectives are listed in Table 1.

Among the reference models listed in Table 1, the DCOR model (Supply-Chain-Council 2004) is the most recommended. It is initially developed by the Business Process Management organization of Hewlett–Packard and then conveyed to SCC in 2004. The major concepts in the DCOR model include five major design procedures—product plan, concept design, detail design, design review, and design amend under the basis of project management. They can also be classified into four levels: top level, which defines the scope and content for the DCOR model; configuration level, which helps a company's design chain to be "configured–to–order"; process element level, which defines a company's ability to compete successfully; implementation level, where companies implement specific design chain management practices. Through four-level hierarchy, DCOR model can be applied to not only analyze intra-organizational and inter-organizational design chain framework and processes but also identify in-depth the position of each design chain member and its impact on design chain performance.

Although DCOR model helps firms construct design chain models and formulate improvement plans, it remains in preliminary stages of development (Wu et al. 2007). The major focuses of the existing efforts have been on the efficiency and effectiveness of the information flow with the aim to improve the performance of the design chain system. The modeling of the design chain in terms of the product itself, among other issues, is often overlooked. The research presented in this paper attempt to develop a comprehensive conceptual model of design chain management, which can be used as a roadmap and theoretical foundation for supporting collaborative design.

Conceptual model of DCM

A holistic conceptual model of design chain management in the DCM context should enclose three interrelated models: operation model, management model and collaboration model, as depicted in Fig. 2 (Liu and Zeng 2009). Design Chain Operation Reference—model as a standard operation model for DCM is under the basis of design process and through five major design procedures—product plan, concept design, detail design, design review, and design

Table 1 Existing design chain management reference models

Model	Objective
Integration mechanisms (Twigg 1998)	To coordinate supplier-manufacturer joint design activities from technology, organization, and procedure aspects during the pre-project, design, and manufacturing phases
Design chain conceptual model (PRTM model) (Deck and Strom 2002)	To develop a good strategy for the design of a design chain that will have a governance structure to facilitate the collaborative work between partners
Process maturity model (Fraser et al. 2003)	To improve product development collaboration in the design chain by properly managing the development processes and the communication plans between partners
Standard DCOR (Supply-Chain-Council 2004)	To identify the principal process elements in the design chain, as a reference for firms to analyze and to improve design chain performance
Design chain collaboration complexity trend model (Chuang and Yang 2004)	To establish design chain collaboration strategies
Performance measurement framework (Nagarajan et al. 2004)	To assess supplier capability and compatibility in terms of design chain processes, tools and techniques
Product design chain collaboration framework (Choi et al. 2005)	To resolve major obstacles to collaboration during product design by offering design process reference model based on SCOR, service component reference model, technology and standard reference model
Collaborative design chain operations reference model (Wu et al. 2007)	To develop a collaborative design chain system in the product lifecycle
Distributed change control workflow (Shiau and Wee 2008)	To maintain the consistence among designs in a collaborative design network

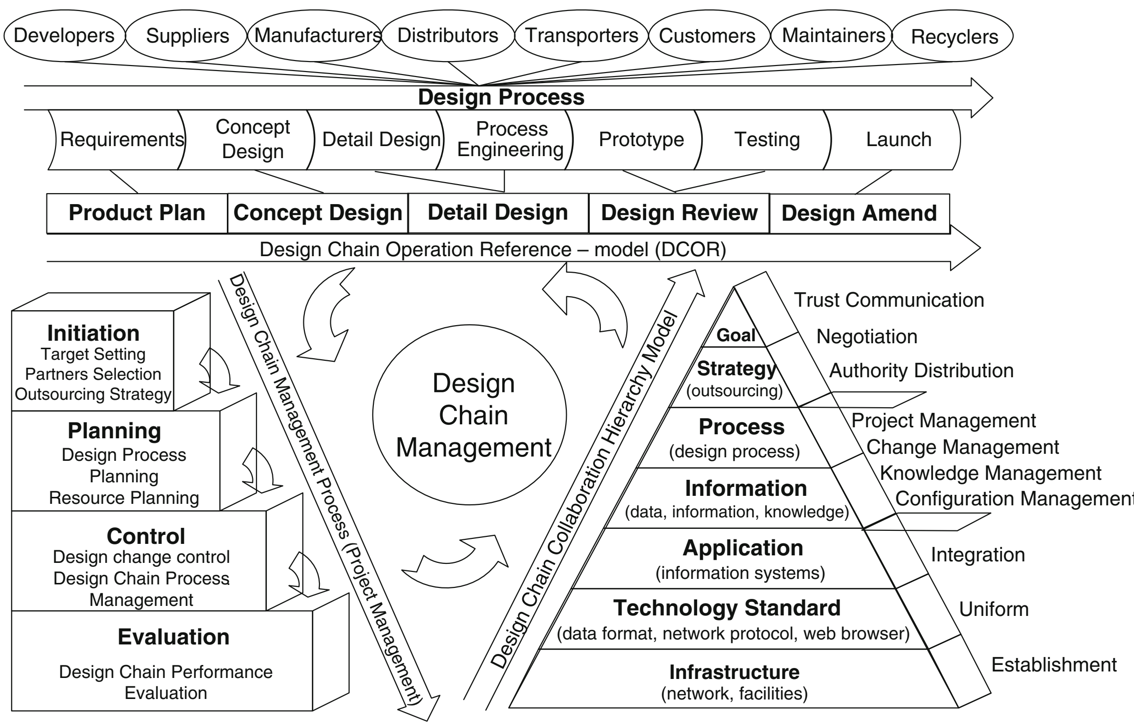


Fig. 2 Conceptual model of DCM (Liu and Zeng 2009)

amend. The principal process elements in DCOR can be mapped to the product design process from requirements, concept design, detail design, through process engineering, prototype, testing to product launch and the involvement of all stakeholders of product lifecycle management in the DCM. The management process defined in conceptual model of DCM integrates the design chain management process, which is also called project management, involves a series of management processes including initiation, planning, control, and evaluations. In the initiation phase, OEM sets competitive targets, selecting appropriate design partners and choosing outsourcing strategies. While the design process and resource planning is determined in the planning phase, change control in the control stage needs to be activated in the complex design chain environment once the resource, design processes are planned. At last, design chain performance evaluation is essential to improve the cooperation ability in the design chain, thus increase effective product innovation.

Environment-based design (EBD)

Intuitively, design is a human activity that aims to change an existing environment to a desired one by creating a new artifact into the existing environment. EBD is such a design methodology that provides step-by-step procedures to guide a designer throughout this environment change process. The underlying principles behind the EBD are that design comes from the environment, serves for the environment, and goes back to the environment.

EBD (Zeng 2004a, 2011) was logically derived from the observation above following the axiomatic theory of design modeling (Zeng 2002). As illustrated in Fig. 3, the Environment-Based Design includes three main activities: environment analysis, conflict identification, and solution generation. These three activities work together progressively and simultaneously to generate and refine the design specifications and design solutions.

This section will give an overview of the EBD methodology. The first subsection will be focused on the EBD process whereas the second will introduce a conceptual model for representing design.

Environment-based design (EBD): process

Mathematically, the EBD process can be represented by structure operation, denoted by ⊕. Structure operation can be defined as the union (∪) of an object O and the interaction (⊗) of the object with itself (Zeng 2004a, 2011).

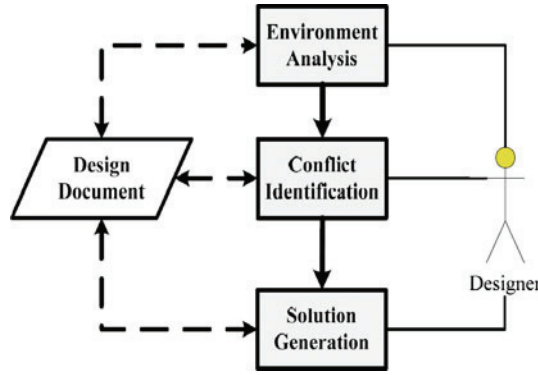


Fig. 3 Environment-based design: process flow (Zeng 2004a, 2011)

$$\oplus O = O \cup (O \otimes O), \tag{1}$$

where ⊕O is the structure of the object O. Everything in the universe can be seen as an object. Interactions between objects are also objects. Examples of interaction include force, movement, and system input and output. Structure operation provides a means to represent a hierarchical system with a single mathematical expression. The application of structure operation can be found in the representation of sketches (Zeng et al. 2004) and linguistic information in design (Zeng 2008).

Due to the capacity of human congnition and the scope of an application, a group of primitive objects can always be defined as (Zeng 2002, 2008)

$$\oplus O_i^a = O_i^a. \tag{2}$$

Equation (2) means that a primitive object is an object that cannot or need not to be further decomposed.

In the design process, any previously generated design concept can be indeed seen as an environment component for the succeeding design. As a result, a new state of design can be defined as the structure of the old environment (E_i) and the newly generated design concept (S_i), which is a partial design solution.

$$\oplus E_{i+1} = \oplus (E_i \cup S_i)\,. \tag{3}$$

It has been shown that the environment structure, which is ⊕E, includes the description of the design solution at design stage i and the design requirements for the design stage i+1 (Zeng 2004b).

Therefore, the EBD process can also be graphically illustrated in Fig. 4, which implies the recursive evolution of design requirements and design solution (Zeng and Cheng 1991; Zeng and Jing 1996).

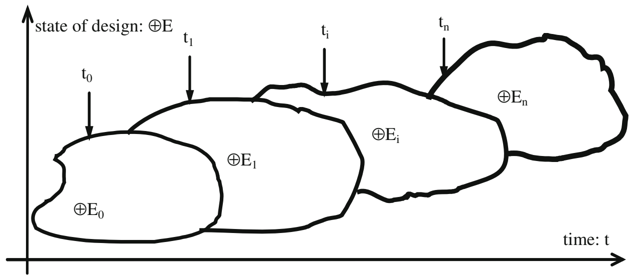


Fig. 4 Environment based design: process (Zeng 2004b)

The following explains the three activities included in EBD (Zeng 2004a, 2011; Zeng and Yao 2009):

Step 1: Environment analysis: define the current environment system $\oplus E_i$.

$$\oplus E_i = \oplus \left(\cup_{j=1}^{n_e} E_{ij}\right) = \cup_{j=1}^{n_e} (\oplus E_{ij}) \cup \cup_{j_1=1}^{n_e} \cup_{\substack{j_2=2 \\ j_2 \neq j_1}}^{n_e} \left(E_{ij_1} \oplus E_{ij_2}\right), \quad (4)$$

where n_e is the number of components included in the environment E_i at the *i*th design state; E_{ij} is an environment component at the same design state. It should be noted that decisions on how many (n_e) and what environment components (E_{ij}) are included in E_i depend on designer's experience and other factors relevant to the concerned design problem.

Step 2: Conflict identification: identify undesired conflicts C_i between environment components by using evaluation operator K_i^e, which depends on the interested environment components.

$$C_i \subset K_i^e (\cup_{j_1=1}^{n_e} \cup_{\substack{j_2 = 1 \\ j_2 \neq j_1}}^{n_e} \left(E_{ij_1} \otimes E_{ij_2}\right). \quad (5)$$

Step 3: Solution generation: generate a design solution s_i by resolving a group of chosen conflicts through a synthesis operator K_i^s. The generated solution becomes a part of the new product environment for the succeeding design.

$$\exists C_{ik} \subset C_i, K_i^s : c_{ik} \rightarrow s_i, \oplus E_{i+1} = \oplus (E_i \cup s_i). \quad (6)$$

The design process above continues with new environment analysis until no more undesired conflicts exist, i.e., $C_i = \Phi$.

Environment-based design (EBD): representation

As shown in the last subsection, environment structure can be used to represent design mathematically throughout the design process. A critical component in the structure operation is the interaction operation $\otimes$. To facilitate the application of the structure operation, the Recursive Object Model (ROM) is proposed to represent the information appeared in design (Zeng 2008). The ROM includes two kinds of objects, which are primitive and compound objects, and three kinds of relations: constraint, predicate and connection. Table 2 shows the graphic symbols in the ROM.

ROM is the foundation of EBD. In environment analysis, designer transforms original design problem and requirements described by natural language into a ROM diagram, which will enable the designer to define and understand the design problem more clearly. In conflict identification, ROM diagram can show the relations between environment components clearly, which can be used to identify the root conflict.

Formalization of design chain management

The purpose of formalization is to turn the informal into a formal structure. In this section, we will start with an informal definition of DCM and finish with a formal conceptual model of the DCM. The EBD methodology is adopted to formalize the design chain management.

An informal definition of the DCM is given as follows:

The term Design Chain Management was defined as the management of the participants, both internal and external to a focal firm that contributes the capabilities (knowledge and expertise) necessary for the design and development of a product which, on completion, will enable full-scale manufacture to commence (Twigg 1998). Consequently, the design chain involves participants throughout the product development process, from concept, detail engineering, process engineering, prototype manufacturing, through to post-launch activities (Twigg 1998), including designers, suppliers, manufacturers and customers.

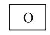
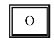
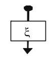
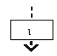
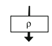
The following subsections will show the EBD process of formalizing the definition above into a formal DCM conceptual model.

Environment analysis

According to the EBD, a design problem is implied in the environment in which the product is expected to work. The objective of environment analysis is to identify the environment components and their relations implied in a design problem, as is shown in Eq. (4).

Environment can be generally classified into natural, built, and human environments, denoted by E^n, E^b, and

Table 2 Recursive Object Model (ROM) (Zeng 2008)

Type		ROM symbols	Description
Object	Object		Everything in the universe is an object
	Compound object		It is an object that includes at least two objects in it
Relations	Constraint relation		It is a descriptive, limiting, or particularizing relation of one object to another
	Connection relation		It is to connect two objects that do not constrain each other
	Predicate relation		It describes an act of an object on another or that describes the states of an object

E^h, respectively (Zeng 2002). Human environment refers to everyone who has a direct impact on the system; natural environment represents every natural aspect that is not subject to human control; and built environment includes every creation of human being that directly affects the system.

$$\oplus E = \oplus \left(E^n \cup E^b \cup E^h\right). \tag{7}$$

Therefore, the first step in formalizing the DCM is to identify the major components and their relations implied in the informal definition of the DCM. This can be achieved by generating the ROM diagram for the informal definition, which is shown in Fig. 5.

Figure 5 can be simplified into Fig. 6 following the rules defined in Zeng (2010), which can show the structure of DCM system clearly.

From this ROM diagram, it is clear that the main objects are participant, capability and process since they have the most constraint relations. Furthermore, the most critical constraining objects are firm and product. These five objects constitute the first level of environment components, as is given in Table 3. They can be seen as primitive objects as defined in Eq. (2), that is,

$$\oplus E_i = E_i, \forall i = 1, \ldots, 5. \tag{8}$$

The relations between these five components are given in Table 4.

Mathematically, the ROM diagram shown in Figs. 5 and 6 are part of the following structure.

$$\oplus \Omega = \oplus[(E_1 \cup E_2 \cup E_3) \cup (E_4 \cup E_5)]. \tag{9}$$

According to Eq. (1), Eq. (9) can be expanded as follows:

$$\begin{aligned}\oplus \Omega = {} & [\oplus(E_1 \cup E_2 \cup E_3)] \cup [\oplus(E_4 \cup E_5)] \cup [(E_1 \cup E_2 \cup E_3) \\ & \otimes (E_4 \cup E_5)] \cup [(E_4 \cup E_5) \otimes (E_1 \cup E_2 \cup E_3)] \\ = {} & [(\oplus E_1) \cup (\oplus E_2) \cup (\oplus E_3) \cup (E_1 \otimes E_2) \cup (E_1 \otimes E_3) \\ & \cup (E_2 \otimes E_1) \cup (E_2 \otimes E_3) \cup (E_3 \otimes E_1) \cup (E_3 \otimes E_2)] \\ & \cup [(\oplus E_4) \cup (\oplus E_5) \cup (E_4 \otimes E_5) \cup (E_5 \otimes E_4)] \cup \\ & [(E_1 \otimes E_4) \cup (E_1 \otimes E_5) \cup (E_2 \otimes E_4) \cup (E_2 \otimes E_5) \cup \\ & (E_3 \otimes E_4) \cup (E_3 \otimes E_5) \cup (E_4 \otimes E_1) \cup (E_4 \otimes E_2) \cup \\ & (E_4 \otimes E_3) \cup (E_5 \otimes E_1) \cup (E_5 \otimes E_2) \cup (E_5 \otimes E_3)].\end{aligned} \tag{10}$$

According to Eq. (8), Eq. (10) can be simplified as

$$\begin{aligned}\oplus \Omega = {} & [E_1 \cup E_3 \cup (E_1 \otimes E_2) \cup (E_1 \otimes E_3) \cup (E_1 \otimes E_4) \\ & \cup (E_1 \otimes E_5) \cup (E_3 \otimes E_1) \cup (E_3 \otimes E_2) \cup (E_3 \otimes E_4) \\ & \cup (E_3 \otimes E_5)] \cup [E_2 (E_2 \otimes E_1) \cup (E_2 \otimes E_3) \cup (E_2 \otimes E_4) \\ & \cup (E_2 \otimes E_5)] \cup [E_4 \cup (E_4 \otimes E_1) \cup (E_4 \otimes E_2) \\ & \cup (E_4 \otimes E_3) \cup (E_4 \otimes E_5)] \cup [E_5 \cup (E_5 \otimes E_1) \\ & \cup (E_5 \otimes E_2) \cup (E_5 \otimes E_3) \cup (E_5 \otimes E_4)] \\ = {} & M_{ts} \cup M_{pc} \cup M_{pd} \cup M_{pm},\end{aligned} \tag{11}$$

where

$$\begin{aligned}M_{ts} = {} & [E_1 \cup E_3 \cup (E_1 \otimes E_2) \cup (E_1 \otimes E_3) \cup (E_1 \otimes E_4) \\ & \cup (E_1 \otimes E_5) \cup (E_3 \otimes E_1) \cup (E_3 \otimes E_2) \cup (E_3 \otimes E_4) \\ & \cup (E_3 \otimes E_5)],\end{aligned} \tag{12}$$

$$M_{pc} = [E_2 \cup (E_2 \otimes E_1) \cup (E_2 \otimes E_3) \cup (E_2 \otimes E_4) \cup (E_2 \otimes E_5)], \tag{13}$$

$$M_{pd} = [E_4 \cup (E_4 \otimes E_1) \cup (E_4 \otimes E_2) \cup (E_4 \otimes E_3) \cup (E_4 \otimes E_5)], \tag{14}$$

$$M_{pm} = [E_5 \cup (E_5 \otimes E_1) \cup (E_5 \otimes E_2) \cup (E_5 \otimes E_3) \cup (E_5 \otimes E_4)]. \tag{15}$$

Hence, the structure of DCM environment can be represented as the union of the structure of participant, process, capability,

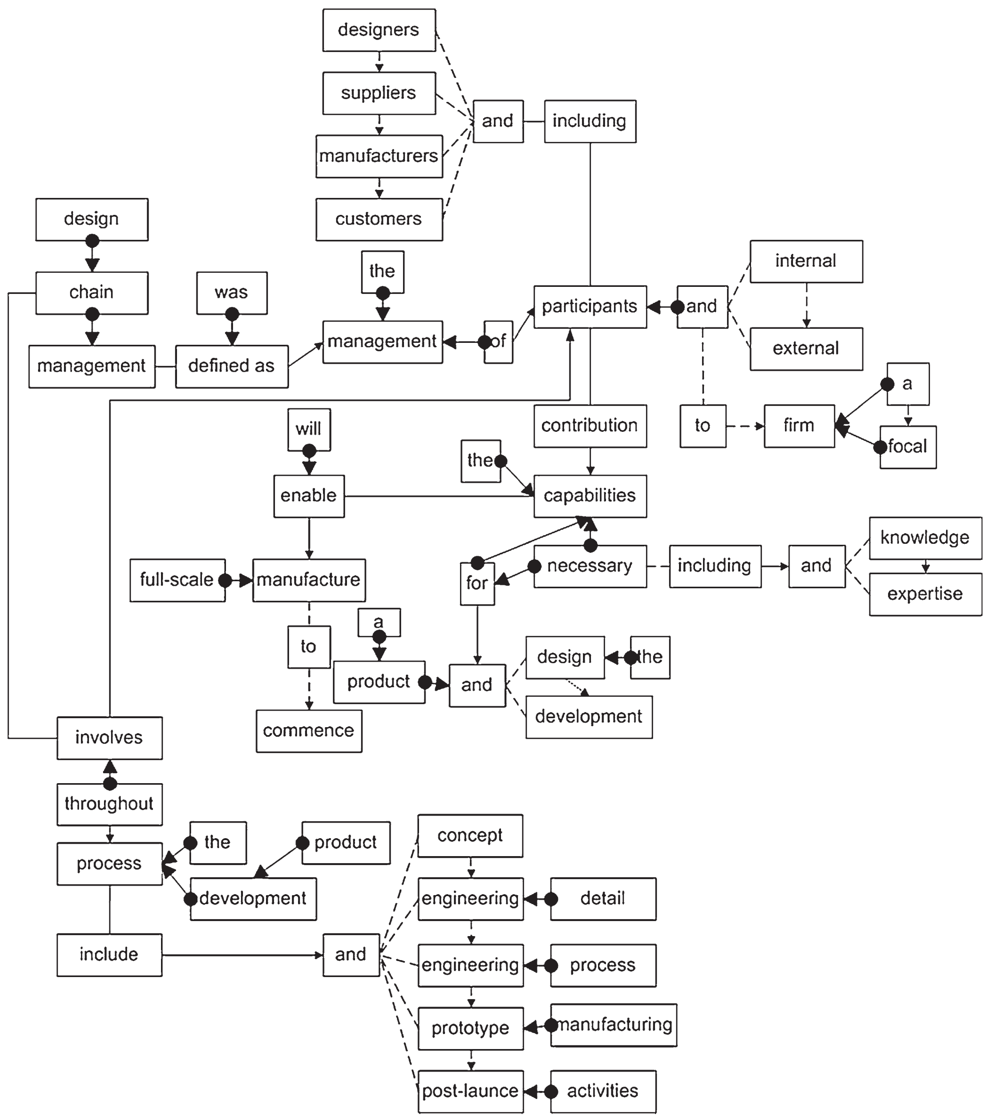


Fig. 5 ROM diagram of informal definition of DCM

product, firm, as well as the relations among them. This structure is given in Eq. (11) and illustrated in Fig. 7, which can be further refined by decomposing its components.

Conflict identification and solution generation

In the DCM environment, from Table 4, it is obvious that the relations between environment components are independent, which means there is no conflict in the current DCM system. Further problems of the design may come from the decomposition of the identified environment components.

Interpretation of the formal model

In Eq. (11), M_{ts} means that participant (E_1) needs to take into account capability (E_3) of all relevant cycles, such as sources, time, technology and so on, in order to work out a reasonable and feasible work task. This leads to a task centered model in the DCM. M_{pc} shows all the related processes in DCM, and the involved participants, the necessary capabilities in different processes, as well as the relations between them. For example, the designers in design process and the manufacturers in engineering process. That means a process centered model is needed in DCM. Similarly, M_{pd} shows product and the relations between product and other primitive objects, so the product is a very important object in DCM which should be focused on; hence, a product centered model is very necessary. M_{pm} represents the focal firm and the relations between the firm and other objects in DCM, which pays close attention to the performance of the whole DCM system and show the reason why a performance centered model is important in DCM.

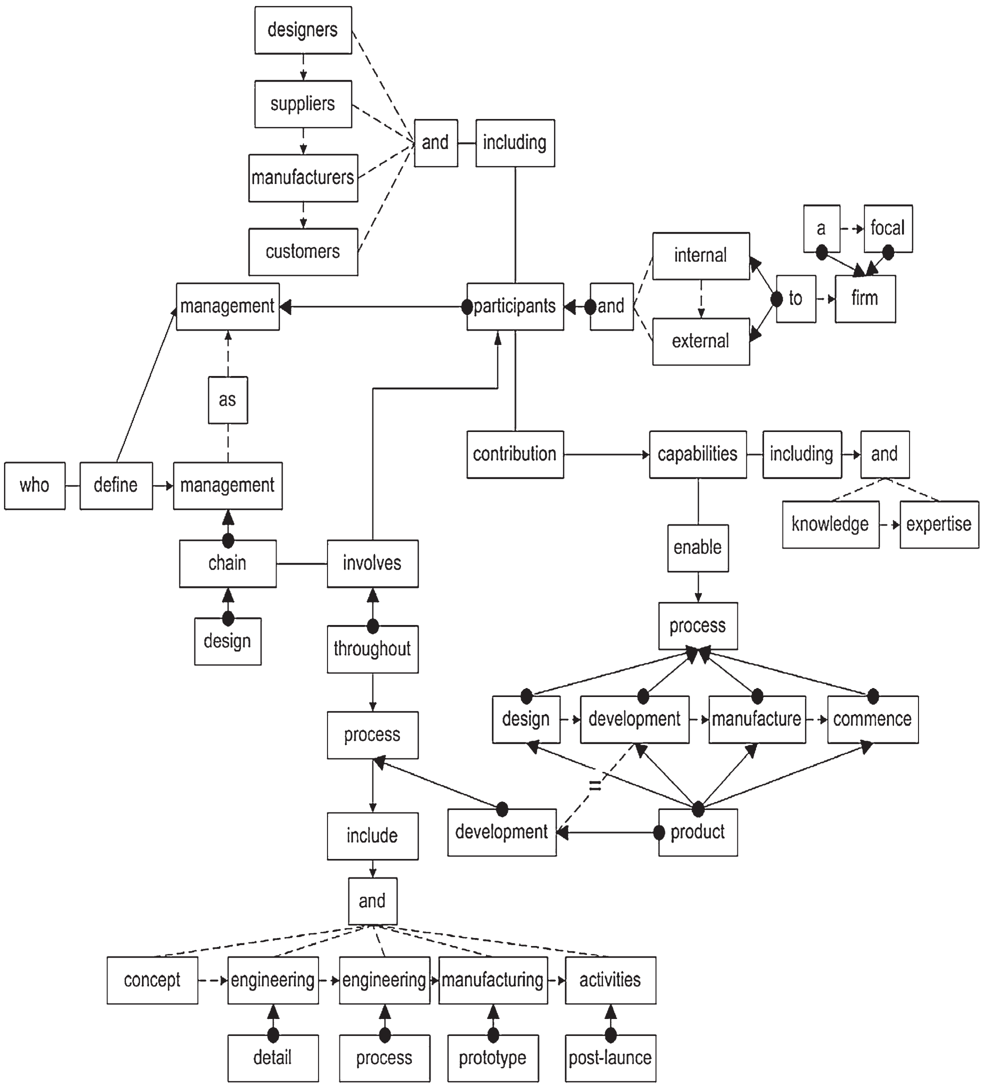


Fig. 6 Simplication of ROM diagram for DCM Definition

Table 3 Environment components—1

E_1	E_2	E_3	E_4	E_5
Participant	Process	Capability	Product	Firm

Consequently, DCM can be regarded as a management process, providing a set of solutions to address various unacceptable conflicts among the relations between its environment components. A representative model of design chain management can be derived as

$$
\begin{aligned}
S_{DCM} &= \oplus\left(M_{ts} \cup M_{pc} \cup M_{pd} \cup M_{pm}\right) \\
&= (\oplus M_{ts}) \cup \left(\oplus M_{pc}\right) \cup \left(\oplus M_{pd}\right) \cup \left(\oplus M_{pm}\right) \\
&\quad \cup \left(M_{ts} \otimes M_{pc}\right) \cup \left(M_{ts} \otimes M_{pd}\right) \cup \left(M_{ts} \otimes M_{pm}\right) \\
&\quad \cup \left(M_{pc} \otimes M_{ts}\right) \cup \left(M_{pc} \otimes M_{pd}\right) \cup \left(M_{pc} \otimes M_{pm}\right)
\end{aligned}
$$

Table 4 Relations bewteen components—1

	E_1	E_2	E_3	E_4	E_5
E_1		Plan, execute, monitor …	Contribute	Control, manage	Constitute, manage …
E_2	Influenced by		Influenced by	Contribute	Owned by
E_3	Belong to	Enable		Determine	Contribute
E_4	Produced by	Influenced by	Influenced by		Developed by
E_5	Contain	Manage, control …	Influenced by	Control, manage …	

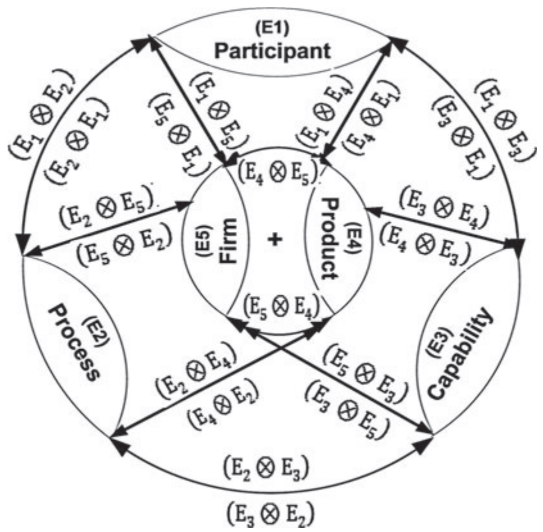


Fig. 7 Structure for conceptual model of DCM

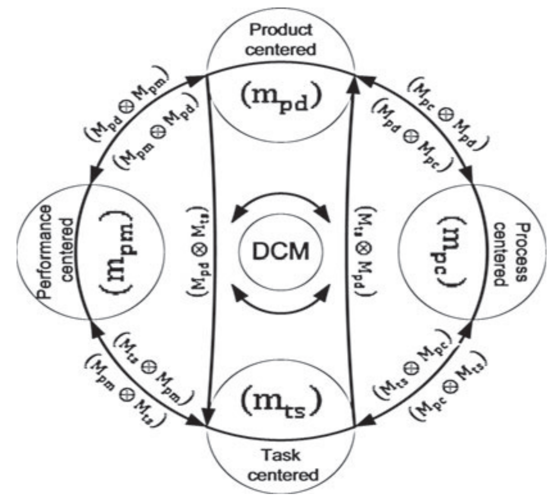


Fig. 8 Stucture of wheel model

$$\cup\left(M_{pd}\otimes M_{ts}\right)\cup\left(M_{pd}\otimes M_{pc}\right)\cup\left(M_{pd}\otimes M_{pm}\right)$$
$$\cup\left(M_{pm}\otimes M_{ts}\right)\cup\left(M_{pm}\otimes M_{pc}\right)\cup\left(M_{pm}\otimes M_{pd}\right), \quad (16)$$

where again M_{ts} represents a task centered model; M_{pc} represents a process centered model; M_{pd} represents a product centered model; and M_{pm} represents a performance centered model.

Refinement of DCM conceptual model

According to Eq. (16), the structure of conceptual DCM model—Wheel Model can be generated as in Fig. 8. In this wheel model, four different types of models—task centered model, process centered model, product centered model and performance centered model work together to resolve problems in the DCM process from different aspects. Diverse elements and aspects such as involved participations, design process, design chain operation, design chain management process, and design chain collaboration, are taken into account, which are significant for the effective development of potential DCM methodologies or supporting tools. It is called the wheel model since any internal conflict between the components of DCM will lead to an evolution of the model, where the internal conflict is like the torque acting on the wheel of DCM model.

In order to complete the wheel model, the four models included in Eq. (16) should be developed. The following derives the product centered model M_{pd}. The EBD methodology is used again to further refine the product lifecycle management (PLM) by decomposing its components.

Liu et al. (2009) described components and their relationships in the PLM environment as depicted in Fig. 8. It provides the foundation for identifying the components, subcomponents and all possible relations about a product in the DCM process.

Environment analysis

Table 5 shows the environment components residing in DCM. The human environment includes all involved participants, such as suppliers, manufacturers, customers, developers, transporters, distributors, maintainers, managers, and recyclers. The natural environment contains natural resources such as time and space. The built environment includes objects such as products, organizations, standards, design data, design knowledge, IT tools, design processes and design strategy.

In the environment of DCM, various relations exist between two components or from a component to itself or among multi-components. Relations can be described in words including design, use, share, inquiry, communication, and so on. In most cases, there are various relations between two components and each relation may have its attributes, properties, features and levels, etc. For example, Clark and Fujimoto (1991) observed the relation of design authority between suppliers and manufacturers, and divided it into three levels: full authority, part authority and no authority, while Twigg extended it to eight levels (Twigg 1998). According to (Sharma 2005), at least three categories of relations between human and data can be extracted such as response, requirement/feedback and decision supporting. Furthermore, there are even relations between two relations. Table 6 gives an example to show the relations between E11 (developers) and other components (Fig. 9).

Table 5 Environment components in PLM

Human environment: E^h	
Design chain participants	E_{11}: developers; E_{12}: manufacturers; E_{13}: distributors; E_{14}: maintainers; E_{15}: suppliers; E_{16}: transporters; E_{17}: recyclers; E_{18}:managers;
Built environment: E^b	
Processes in PLM	E_{21}: requirements; E_{22}:conceptual design; E_{23}: detail engineering; E_{24}: process engineering; E_{25}: prototype manufacturing; E_{26}: testing; E_{27}: goal; E_{28}: strategy; E_{29}: performance;
Product design information	E_{31}: configuration; E_{32}: specification; E_{33}: BOM; E_{34}: engineering changes; E_{35}: cost; E_{36}: parameters;
Product design standards	E_{41}: STEP; E_{42}: PDML; E_{43}: U3D; E_{44}: XML;
IT Tools	E_{51}: SCM; E_{52}: PM; E_{53}: CAD; E_{54}: CAE; E_{55}: PDM; E_{56}: EDM; E_{57}: ERP;
Products	E_{61}: mechanical products; E_{62}: aerospace products; E_{63}: electronic products; E_{64}: service; E_{65}: automatic products;
Natural environment: E^n	
E_{71}: time; E_{72}: space;	

Table 6 Relations between components

$E_{11} \otimes E_{12}$	Communicate, cooperate	$E_{11} \otimes E_{21}$	Plan, execute, monitor
$E_{11} \otimes E_{31}$	Create, change, share	$E_{11} \otimes E_{41}$	Adopt
$E_{11} \otimes E_{53}$	Use	$E_{11} \otimes E_{61}$	Develop
$E_{11} \otimes E_{71}$	Spend	$E_{11} \otimes E_{72}$	Consider

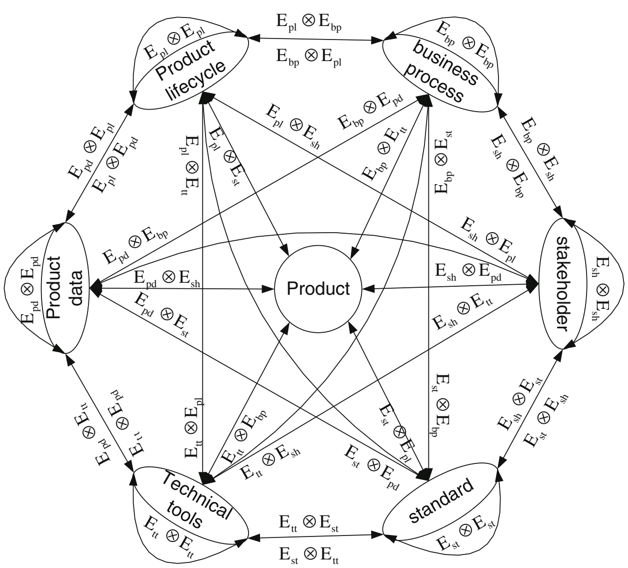


Fig. 9 PLM Relational graph in DCM environment (Liu et al. 2009)

Conflict identification

According to the system component relations identification in the environment analysis part, the relations among the relations between system components are identified. This relation analysis could help find out potential conflicts, which will be resolved in the next step for further analysis. Table 7 gives an example to show the important possible conflicts which are generated from environment system relations.

Solution generation

Blake and Mouton first presented five conflict management styles as problem-solving, smoothing, forcing, withdrawal and sharing (Blake and Mouton 1970). Thomas later relabeled them and proposed five dimensions of conflict-handling intensions, including avoiding, accommodating, competing, compromising and collaborating (Thomas 1992). Conflict resolution techniques include problem-solving, superordinate goals, expansion of resources, communication, and restructuring the organization. Table 8 gives the corresponding solutions to the conflicts in Table 7.

Therefore, according to all the conflicts obtained from conflict identification and corresponding solutions stemmed from solution generation, a PLM model is derived, as shown in Fig. 10. It must be noted that there can be more conflicts in the PLM process than those listed in Tables 7 and 8, which are just examples to show how to get the product centered model by using the EBD.

There are six parts implied in the PLM wheel model: Product Lifecycle; Product Data; Standards; Business Processes; Technology Tools; and Stakeholders. This wheel model classifies and summarizes the types of conflicts that may appear in the PLM, and supplies corresponding solutions for each type of conflicts. The product lifecycle level (C1S1, C2S2) refers to the conflicts appearing in the product development phase, production phase, distribution phase, operation phase and retirement phase. Product data level

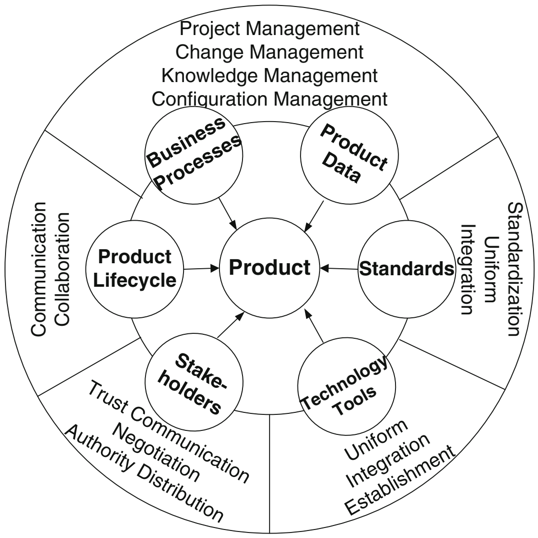


Fig. 10 PLM Wheel model

Table 7 Conflict identification

Relations	Possible conflicts
$E_{29} \otimes E_{18}, E_{18} \otimes E_{35}$	Collaboration performance and transaction cost (C1)
$E_{12} \otimes E_{15}, E_{15} \otimes E_{29}$	Various suppliers have different ability in product development process(C2)
$E_{12} \otimes E_{32}, E_{15} \otimes E_{32}$	Inconsistency of engineering data due to change (C3)
$E_{11} \otimes E_{36}, E_{15} \otimes E_{36}$	Incompatibility between Information sharing and protection (C4)
$E_{11} \otimes E_{23}, E_{12} \otimes E_{23}$	Inconsistency between design processes of different participants (C5)
$E_{18} \otimes E_{34}, E_{11} \otimes E_{34}$	Incompatible perspectives of different participants (C6)
$E_{14} \otimes E_{26}, E_{14} \otimes E_{26}$	Release more design control authority to increase communication (C7)
$E_{27} \otimes E_{28}$	Process contradictory (C8)
$E_{11} \otimes E_{26}, E_{11} \otimes E_{26}$	Incompatible between different tools (C9)
$E_{11} \otimes E_{27}, E_{12} \otimes E_{27}$	Incompatible goals of different participants (C10)
$E_{11} \otimes E_{28}, E_{18} \otimes E_{28}$	Inconsistent strategies among different participants (C11)

(C3S3, C4S4) represents conflicts about data format, sharing mechanism, technology standards, information semantics, product parameters and product related knowledge in PLM. There are many standards in the stages of the product lifecycle, the origin, the scope and the development process (Nyere 2006). At the standards level (C5S5, C6S6) includes all the problems about standards. Business process level (C7S7, C8S8) represents all conflicts about product market strategy, product portfolio planning, product platform planning, customer requirements, product specification, conceptual design, detailed design, design analysis, prototyping and testing, process planning, inventory management, sourcing production, inspection, packing, distribution, operation and service, disposal and recycle (Ameri and Dutta 2004; Wu et al. 2007). Technology tools level (C9S9) includes conflicts in the technology tool applications. Stakeholders level (C10S10, C11S11) contains problems in different participants, such as diverse competitive goals among different partners and the inconsistent strategies among different participants.

The collaboration in product lifecycle phase and information sharing mechanism in product data phase can be supported by communication, collaboration, sharing mechanism and project management. Business processes phase is operated through the functionalities such as data management, change management, configuration management, document management, and knowledge management and so on, while standards phase works well based on standardization, uniform and integration. Trust communication, negotiation and authority distribution are the core strategies to solve the problems between different participants in stakeholder phase. Finally, uniform, integration and establishment enhance the efficiency in technology tools phase.

Once we get the product centered model, the refinement for the wheel model can be constructed based on Fig. 11.

From the example above, it is clear how to get the process centered model, the task centered model and the performance centered model by using EBD methodology. Therefore, the whole wheel model can be completed when these three models are generated.

Example

In this section, protection of confidential information in design chain management is chosen and addressed for an industrial product. The objective of this section is to show that the proposed conceptual model of design chain management can clarify and resolve unresolved conflict effectively.

Background

In this part, a real industrial product—Compressed Natural Gas Dryer (Geng and Li 2008) is studied. It is a dual tower heat— reactivated equipment for natural gas dry procedure, whose primary function is to provide dry natural gas.

Firstly, it is necessary to explain why we are concerned about the confidential information protection for this product. Generally speaking, natural gas must be dried before it can be put into use. The Compressed Natural Gas Dryer introduced in this section has been the major product of a company ABC, which owns a big share of the market. Company ABC's competitors have always wanted to obtain the key technical parameters of the dryer for their own product development. One way they have tried is to go through company ABC's suppliers and customers, in an attempt to

Table 8 Possible solution generation

Conflict descriptions	Suggested solutions
Collaboration performance and transaction cost (C1)	Web based collaboration software, collaboration performance assessment (S1)
Various suppliers have different ability in product development process(C2)	Design partner selection (S2)
Inconsistency of engineering data due to change (C3)	Collaboration and communication (S3)
Incompatibility between Information sharing and protection (C4)	Collaboration and secure information sharing strategy (S4)
Inconsistency between design processes of different participants (C5)	Mapping existing processes to the predefined process templates (S5)
Incompatible perspectives of different participants (C6)	Collaboration and communication (S6)
Release more design control authority to increase communication (C7)	Centralization (S7)
Process contradictory (C8)	Process collaboration and coordination (S8)
Incompatible between different tools (C9)	Standardization and coordination (S9)
Incompatible goals of different participants (C10)	Goals collaboration (S10)
Inconsistent strategies among different participants (C11)	Strategies collaboration and communication (S11)

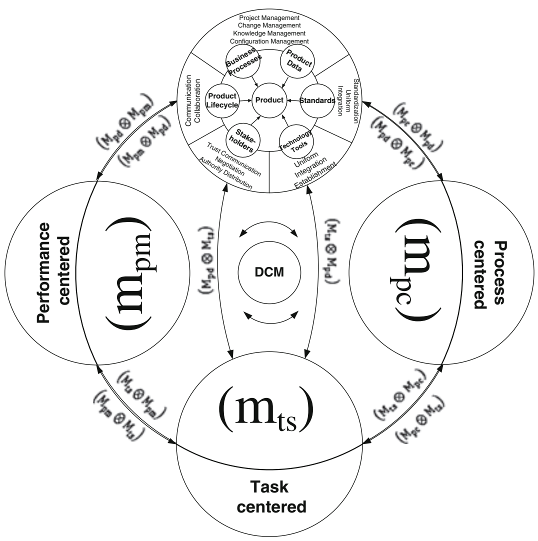


Fig. 11 Development for conceptual model of DCM

reverse engineer the dryer. One of the company ABC's earlier dryer was copied by one of its competitors successfully. In order to avoid the leakage of the key technical information of their newly upgraded product, the company adopted the methodology similar to what we have derived below.

Solutions from the presented formalization approach

According to the analysis based on conceptual model of DCM given in Eq. (16), Fig. 8 and Table 7, there exists a conflict between developers and suppliers regarding the access to the confidential information held by the developers. This is shown in Fig. 12. It is hard to make a decision for developers about what design parameters can be safely shared with their suppliers. On the one hand, suppliers need design parameters to supply proper assembly parts. On the other hand, developers want to protect product data for legal and/or competition purposes.

This conflict can be mathematically represented as

$$\mathrm{C}\left[\mathrm{require}\left(\mathrm{E}_{15}, \mathrm{E}_{36}\right), \mathrm{protect}\left(\mathrm{E}_{11}, \mathrm{E}_{36}\right)\right], \tag{17}$$

where require $(\mathrm{E}_{15}, \mathrm{E}_{36})$, shows the relationship "require" from suppliers E_{15} to product parameters E_{36} whereas protect $(\mathrm{E}_{11}, \mathrm{E}_{36})$ shows the relationship "protect" between developers E_{11} and product parameters E_{36}.

According to the conceptual model of DCM, collaboration and sharing mechanism will be used to show how the conflict in Eq. (17) can be resolved. For the conflict (S4) between developers and suppliers mentioned in Sect. "Refinement of DCM Conceptual Model", available solutions have to be designed and performed until suppliers can access the required data.

The first candidate solution would be to merge developers E_{11} and suppliers E_{15} into a new object E_{11}^{*}.

$$\begin{aligned}\mathrm{E}_{11}^{*} &= \mathrm{E}_{11} \oplus \mathrm{E}_{15} \\ &= \mathrm{E}_{11} \cup \mathrm{E}_{15} \cup \left(\mathrm{E}_{11} \otimes \mathrm{E}_{15}\right) \cup \left(\mathrm{E}_{15} \otimes \mathrm{E}_{11}\right).\end{aligned} \tag{18}$$

The interaction $(\mathrm{E}_{11} \otimes \mathrm{E}_{15}) \cup (\mathrm{E}_{15} \otimes \mathrm{E}_{11})$ can be a collaboration relationship established based on a confidentiality agreement between the developers E_{11} and suppliers E_{15}, shown as collaborate $(\mathrm{E}_{11}, \mathrm{E}_{15})$. As a result, the solution can be represented as

$$\mathrm{C}\left[\mathrm{require}\left(\mathrm{E}_{11}, \mathrm{E}_{36}\right), \mathrm{protect}\left(\mathrm{E}_{15}, \mathrm{E}_{36}\right)\right] \rightarrow \mathrm{E}_{11.}^{*} \tag{19}$$

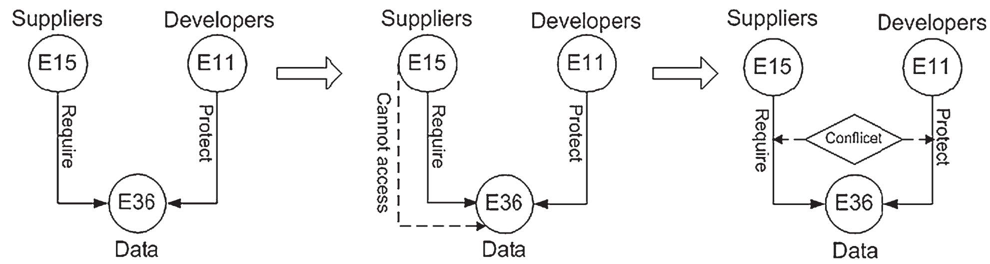


Fig. 12 Conflict between developers and suppliers

The second candidate solution (Zhang et al. 2011a) would be to divide the object E_{36} such that

$$E_{36} = E_{36}^1 \oplus E_{36}^2 = E_{36}^1 \cup E_{36}^2 \cup (E_{36}^1 \otimes E_{36}^2) \cup (E_{36}^2 \otimes E_{36}^1). \tag{20}$$

In this decomposition, the suppliers would be able to access the parameter E_{36}^1 necessary for their work while the developers will protect their data e_4^2. The key becomes how to devise a reliable interaction mechanism $(E_{36}^1 \otimes E_{36}^2) \cup (E_{36}^2 \otimes E_{36}^1)$. As a result, the solution can be represented as

$$C[\text{require}(E_{11}, E_{36}), \text{protect}(E_{15}, E_{36})] \rightarrow E_{36}^1 \oplus E_{36}^2. \tag{21}$$

The solutions can be chosen based on different situations. It is better to divide the data into two irrelative parts, so developers can protect what they try to keep confidential and suppliers can get what they want. However, if the data cannot be divided, a confidentiality agreement between the developers and suppliers can be established, which can largely reduce the information leakage risk level.

Confidential information protection technique

Since the most important confidential information is included in the regeneration process, the confidential information risk analysis will be only focused on this process. The paramers to be protected are listed in the priority of protection level (Table 9).

Paramerter T1 plays a key role for the generation process. It is regarded as the top level parameters for confidential protection. The recovering T1 process is shown in Fig. 13.

Table 9 Priority of protection level

Priority of protection level	
Top level	
Process parameters	Pressure Flow rate Temperature
Empirical coefficient	Used for engineering design
Moderate level	
Mechanical parameters	Product model for purchasing parts of equipments

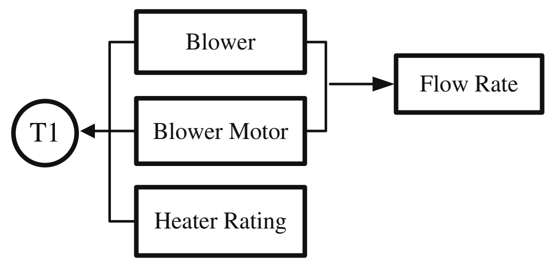


Fig. 13 Recovering T1 process

If one supplier is designed to provide all these three equipments (blower, motor and heater), the risk level of information leakage would be very high. According to the previous analysis, there are two solutions. One is that company makes a collaboration relationship established based on a confidentiality agreement; however, since paramerter T1 plays a key role for the generation process, a more stable solution should be considered. The principle of extraction will be applied in this case. Instead of one supplier providing all three parts, the heater, blower and blower motor should be split into small assembly parts and respectively distributed to independent suppliers, the risk level is getting lower. The company can devide these important equipments into small pieces and manufacture the core parts themselves, so it is hard for competitors to know exactly what are the original equipments.

Figure 14 describes the methodology used for increasing production complexity against the malicious reverse engineering of the product confidential information. The key is to break the relationships between products or within one part, which leads to less probability for tracing parameters.

The details of the techniques are given in Zhang et al. (2011a,b).

Conclusion and future work

This paper contributes to the design chain management by formalizing design chain systems into a comprehensive DCM conceptual model in the context of Product Lifecycle Management (PLM). Design chain involves various complex components and relationships, which present considerable

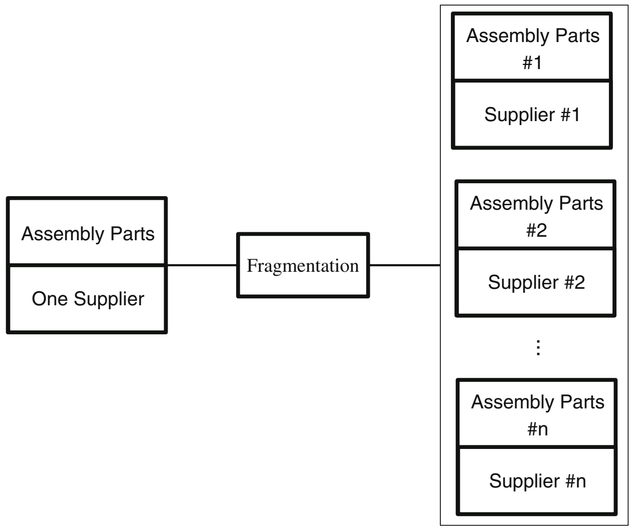


Fig. 14 Conceptual DCM model 1

challenges for DCM. Furthermore, informal explanations of the relationships in the DCM bring additional ambiguities. The formalization process leads to a wheel model for the design chain management. Existing literature attempted to address complete collaboration and design process management issues; however, the informal foundation and static nature of those design chain management reference models not only restrict clear definition of DCM but also limit its future development. Therefore, we propose a derivable mathematical representation to clarify unresolved conflict toward an improved conceptual model, which integrates the product design process, the collaboration hierarchy model, DCOR model and the design chain management process. The mathematical representation reveals unacceptable conflicts between environment components implied in the design chain environment, which presents clear definitions of the design chain participants, design process and the relationship between available technologies and models in DCM. Furthermore, models are constructed to remove the unacceptable conflicts in the DCM, which also identifies new problems for the design chain collaboration.

Future work includes the identification of more relations and conflicts by refining the DCM components and their relationships. As a result, the wheel model will be completed by adding process, task, and performance centered models.

Acknowledgements The research reported in this paper is partially supported by NSERC through a CRD project (PJ 350114-06). We are grateful to the financial support from NSERC, CRIAQ, Pratt & Whitney Canada Corp., Bombardier Inc., CMC Electronics Inc., and Rolls-Royce Canada Limited. We also thank Lina Cao for her help in improving the literature review of this paper. The valuable comments from the anonymous reviewers have greatly helped us to enhance the quality of the paper. We specially thank the guest editor Dr. Chih-Hsing Chu for his patience in waiting for our revision.

References

Ajoku, P. N. (2007). Combining lean initiatives with theory of constraints in distributed product design chain management. *International Journal of Electronic Business Management, 5*(2), 81–92.

Akira, T. (2001). Bridging inter- and intra- firm boundaries: Management of supplier involvement in automobile product development. *Strategic Management Journal, 22*(5), 403–433.

Ameri, F., & Dutta, D. (2004). *Product lifecycle management needs, concepts and components* (p. 2). Ann Arbor, MI: University of Michigan.

Blake, R. R., & Mouton, J. S. (1970). The fifth achievement. *Journal of Applied Behavioral Science, 6*(4), 413–426.

Carlisle, J. A., & Parker, R. C. (1989). *Beyond negotiation: Redeeming customer-suppliers relationships*. Chichester: John Wiley & Sons.

Case, M. P., & Lu, S. C. Y. (1996). Discourse model for collaborative design. *Computer-Aided Design, 28*(5), 333–345.

Choi, Y., Kim, K., & Kim, C. (2005). A design chain collaboration framework using reference models. *The International Journal of Advanced Manufacturing Technology* 26: 1–2.

Chokshi, N., & McFarlane, D. (2008). A distributed architecture for reconfigurable control of continuous process operations. *Journal of Intelligent Manufacturing, 19*(2), 215–232.

Chuang, W.-C., & Yang, H.-H. (2004). Design chain collaboration—a strategic view. *International Journal of Electronic Business Management, 2*(2), 117–121.

CIMdata. (2006). *Product lifecycle management.* A CIMdata Inc. Report, June 1, 2006.

Clark, K. B., & Fujimoto, T. (1991). *Product development performance*. Boston: Harvard Business School Press.

Cooper, M. C., Douglas, M. L., & Janus, D. P. (1997). Supply chain management: More than a new name for logistics. *The International Journal of Logistics Management, 8*(1), 1–14.

Deck, M., & Strom, M. (2002). Model of co-development emerges. *Research Technology Management, 45*(3), 47–54.

Dowlatshahi, S. (1998). Implementing early supplier involvement: A conceptual framework. *International Journal of Operations & Production Management, 18*(2), 143–167.

Fraser, P., Farrukh, C., & Gregory, M. (2003). *Managing product development collaborations—a process maturity approach*. *Proceedings of the Institution of Mechanical Engineers: Part B-Journal of Engineering Manufacture*.

Geng, Y., & Li, H. (2008). Confidential information protection for industry design. Montreal: Concordia Institute for Information systems Engineering, Concordia University, Master Project Report (Supervisor: Yong Zeng).

Huang, G. Q., Huang, J., & Mak, K. L. (2000). Agent-based workflow management in collaborative product development on the internet. *Computer-Aided Design, 32*(2), 133–144.

Kiritsis, D., Bufardi, A., & Xirouchakis, P. (2003). Research issues on product lifecycle management and information tracking using smart embedded systems. *Advanced Engineering Informatics, 17*(3–4), 189–202.

Lambert, D. M., & Cooper, M. C. (2000). Issues in supply chain management. *Industrial Marketing Management, 29*(1), 65–83.

Lambert, D. M., Cooper, M. C., & Pagh, J. D. (1998). Supply chain management: Implementation issues and research opportunities. *The International Journal of Logistics Management, 9*, 1–20.

Lin, H.-Y. (2003). *A robust decision model for selecting partners in design chain management*. Taichung: Graduate Institute of Industrial Engineering, Feng Chia University.

Littler, D., Leverick, F., & Bruce, M. (1995). Factors affecting the process of collaborative product development: A study of uk manufacturers of information and communications technology

products. *Journal of Product Innovation Management, 12*(1), 16–32.
Liu, W., Wang, M., Zhang, D., & Zeng, Y. (2009). *Development of secure collaboration models in product lifecycle management*. Stanford, USA.
Liu, W., & Zeng, Y. (2009). Conceptual modeling of design chain management towards product lifecycle management. In S.-Y. Chou, A. Trappey, J. Pokojski, & S. Smith (Eds.), *Global perspective for competitive enterprise, economy and ecology* (pp. 137–148). London: Springer.
López-Ortega, O., & Ramírez-Hernández, M. (2007). A formal framework to integrate express data models in an extended enterprise context. *Journal of Intelligent Manufacturing, 18*(3), 371–381.
Muckenhirn, R., & Meier, M. (2008). A minimal-invasive approach to configuration management supporting collaborative factory configuration scenarios. *Journal of Intelligent Manufacturing, 19*(6), 735–746.
Mun, J., Shin, M., & Jung, M. (2011). A goal-oriented trust model for virtual organization creation. *Journal of Intelligent Manufacturing, 22*(3), 345–354.
Nagarajan, R. P., Passey, S. J., Wong, P. L., Pritchard, M. C., & Nagappan, G. (2004). Performance measures and metrics for collaborative design chain management. In *10th international conference on concurrent enterprising, Escuela Superior de Ingenieros*. Seville, Spain.
Nyere, J. (2006). *The design-chain operations reference-model*. Supply Chain Council.
O'Grady, P., & Chuang, W.-C. (2001). Research issues in e-commerce and product development. *Cybernetics and Systems: An International Journal, 32*(7), 775–796.
Oliver, R. K., & Webber, M. D. (1992). *Supply chain management—logistics catches up with strategy. Logistics—strategic issues* (pp 63–75). M. G. E. in Christopher. London: Chapman and Hall.
Poirier, C. C., & Reiter, S. E. (1996). *Supply chain optimization: Building the strongest total business network*. San Francisco: Berrett-Koehler Publishers Inc.
Sharma, A. (2005). Collaborative product innovation: Integrating elements of CPI via PLM framework. *Computer-Aided Design, 37*(13), 1425–1434.
Shiau, J.-Y., & Wee, H. M. (2008). A distributed change control workflow for collaborative design network. *Computers in Industry, 59*(2–3), 119–127.
Stewart, G. (1997). Supply-chain operations reference model (scor): The first cross-industry framework for integrated supply-chain management. *Logistics Information Management, 10*(2), 62–67.
Supply-Chain-Council. (2004). Design-chain operations reference model(dcor) version 1.0. Pittsburgh, PA: Supply-Chain Council.
Thomas, K. W. (1992). Conflict and conflict management: Reflections and update. *Journal of Organizational Behavior, 13*(3), 265–274.
Twigg, D. (1997a). In N. Slack (Ed.), *Design chain management*. Oxford: Blackwell.
Twigg, D. (1997b). *Design chain management*. Oxford: Blackwell.
Twigg, D. (1998). Managing product development within a design chain. *International Journal of Operations & Production Management, 18*(5), 508–524.
Venkatachalam, A. R., Mellichamp, J. M., & Miller, D. M. (1993). A knowledge-based approach to design for manufacturability. *Journal of Intelligent Manufacturing, 4*(5), 355–366.
Wang, J., & Hwang, Y.-C. (2005). A study of the impact of activity overlapping on development lead-time in design chain. *Journal of the Chinese Institute of Industrial Engineers, 22*(4), 293–300.
Wang, J., & Lin, H. Y. (2006). A fuzzy hybrid decision-aid model for selecting partners in the design chain. *International Journal of Production Research, 44*(10), 2047–2069.
Wang, L., Shen, W., Xie, H., Neelamkavil, J., & Pardasani, A. (2002). Collaborative conceptual design–state of the art and future trends. *Computer-Aided Design, 34*(13), 981–996.
Wu, W.-H., Yeh, S.-C., & Fang, L.-C. (2007). The development of a collaborative design chain reference model for the motorcycle industry. *The International Journal of Advanced Manufacturing Technology, 35*(3–4), 211–225.
Zeng, Y. (2002). Axiomatic theory of design modeling. *Transaction of SDPS: Journal of Integrated Design and Process Science, 6*(3), 1–28.
Zeng, Y. (2004a). *Environment-based design: Process, model* (p. 40). Montreal: Concordia Institute for Information Systems Engineering, Concordia University.
Zeng, Y. (2004b). Environment-based formulation of design problem. *Transaction of SDPS: Journal of Integrated Design and Process Science, 8*(4), 45–63.
Zeng, Y. (2008). Recursive object model (ROM)—modeling of linguistic information in engineering design. *Computers in Industry, 59*(6), 612–615.
Zeng, Y. (2010). *ROM analysis: Generation, simplication and clarification of ROM diagram* (pp. 1–8). Montreal: Design Lab, Concordia Institute for Information Systems Engineering, Concordia University.
Zeng, Y. (2011). Environment-Based Design (EBD). In *ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (DETC 2011-48263)*. Washington, DC, USA.
Zeng, Y., & Cheng, G. D. (1991). On the logic of design. *Design Studies, 12*(3), 137–141.
Zeng, Y., & Jing, J. (1996). *Computational model for design*. Fourth international conference on computer-aided design and computer graphics, Wuhan, China, SPIE.
Zeng, Y., Pardasani, A., Antunes, H., Li, Z., Dickinson, J., Gupta, V., & Baulier, D. (2004). Mathematical foundation for modeling conceptual design sketches. *Transactions of the ASME: Journal of Computing and Information Science in Engineering, 4*(2), 150–159.
Zeng, Y., & Yao, S. (2009). Understanding design activities through computer simulation. *Advanced Engineering Informatics, 23*(3), 294–308.
Zhang, D. Y., Zeng, Y., Wang, L., Li, H., & Geng, Y. (2011a). Modeling and evaluating information leakage caused by inferences in supply chains. *Computers in Industry, 62*(3), 351–363.
Zhang, D. Y., Cao, X., Wang, L., & Zeng, Y. (2011b). Mitigating the risk of information leakage in a two-level supply chain through optimal supplier selection. *Journal of Intelligent Manufacturing*. doi:10.1007/s10845-011-0527-3.