Journal of Enterprise Information Management

Framework for internet-supported inter-organizational product development collaboration

Eva-Maria Kern, Wolfgang Kersten,

Framework for internet-supported inter-organizational product development collaboration

Eva-Maria Kern and Wolfgang Kersten
Department of Business Logistics and General Management, Hamburg University of Technology, Hamburg, Germany

Abstract

Purpose – The purpose of this article is to introduce a framework for internet supported inter-organizational product development, which enables companies to efficiently configure their development processes according to their needs.

Design/methodology/approach – Three ideal types of partner integration are identified and specific design guidelines for each type are proposed. Current approaches for product development collaboration are analysed. Based on their shortcomings the main approach of this paper evolves. Additionally the main results of interviews with experts are used to develop and introduce a framework which has been tested partially in the shipbuilding industry.

Findings – The paper identifies three types of partner integration and proposes specific design guidelines.

Research limitations/implications – The framework introduced in this paper provides guidance for future research in the area of product development collaboration.

Practical implications – The framework further supports managers in designing efficient and effective inter-organizational product development collaboration by choosing the appropriate level of partner integration.

Originality/value – The paper presents a practical and usable framework for internet supported inter-organizational product development collaboration. The main focus is on designing the partnership interaction.

Keywords Product development, Internet, Economic cooperation, Business policy

Paper type Research paper

Introduction

The product development phase has a significant impact on the efficiency of the entire supply chain. Especially in the development of complex products, which is usually accomplished in processes that are distributed among producers and their suppliers, there is potential for increasing the ability to compete.

Driven by modern information and communication technologies (ICT) numerous approaches for organizing product development collaboration have been developed during recent years. Their core element is the use of ICT to support design activities, data and information management, as well as the interaction between development partners. In this paper we concentrate on those tools of ICT which support interaction between the development partners. Computer Supported Cooperative Work (CSCW) is the all-encompassing term for all aspects of computer-supported cooperation (Wigand *et al.*, 1997). ICT can on the one hand offer tools to enable personal communication such

Journal of Enterprise Information Management
Vol. 20 No. 5, 2007
pp. 562-577

1741-0398
DOI 10.1108/17410390710823716

as e-mail, video-conferences and chats. On the other hand it can enable partners to get access to common material. Common material includes all objects of information, which are used in and arise during the collaboration process (Krcmar *et al.*, 2000). Asynchronous tools for using common material allow file-transfer, file-sharing, data exchange, or data base sharing. Synchronous tools such as shared whiteboards, viewers, or applications, enable distributed development partners to work on the same task and therefore support close interaction and collaboration.

In this context, Willaert, Graaf and Minderhoud defined the concept of Collaborative Engineering, which is:

> . . . a systematic approach to control life cycle cost, product quality and time to market during product development, by concurrently developing products and their related processes with responses to customer expectations, where decision making ensures input and evaluation by all life-cycle disciplines including suppliers, and information technology is applied to support information exchange, where necessary . . . (Willaert *et al.*, 1998).

They interpreted collaborative engineering as a logical consequence of concurrent engineering and pointed out the important role of ICT. A similar technology-driven approach was developed by the Aberdeen Group, which described Collaborative Product Commerce as a "new category of software and services that uses internet technologies to tie together product design, engineering, sourcing, sales, marketing, field services and customers into a global net" (AberdeenGroup, 1999). Accenture avoided the pure technology orientation when it defined Collaborative Product development as business capability to enable fast and efficient product development across dispersed organizations (Accenture, 2001). The integration of technologies, processes and strategies is defined as main base for successful implementation. Monplaisier and Haji as well as Monplaisier and Salhieh point out the enabling function of ICT and concentrate on its efficient use to support interaction (Monplaisir and Haji, 2002; Monplaisir and Salhieh, 2002). Kersten and Kern outline the importance of a holistic design for cross-organizational product development and define Collaborative Engineering as the internet-supported design of distributed product development processes, considering technological as well as organizational and human aspects (Kersten and Kern, 2003). Other approaches in this area follow the same basic ideas (Bullinger, 2003; Contero, 2005).

All the approaches mentioned have the common thread of offering a wide framework providing very general guidelines for the design of product development collaboration. None of them supports companies in choosing the right configuration for their specific development tasks or situations, based on their actual interaction needs.

Driven by this situation the main aim of our paper is to develop a framework for internet-supported inter-organizational product development collaboration in supply chains. This supports companies in selecting the appropriate approach for designing distributed development processes according to their specific interaction needs.

Our paper is structured as follows: In the next section we describe our basic approach: focusing on interactions for organizing product development collaboration. Based on this, we explain the key dimensions which have to be considered in designing collaboration. Insights in industry confirm our findings. In the third section we develop our framework. Its main requirements are described and the basic concept is explained. We introduce a portfolio and identify three ideal types of partner integration which differ in the level of intensity of collaboration. For each type we propose specific guidelines for the design of partner integration, considering IT support as well as strategic, organizational, and personnel aspects. In the penultimate section we describe briefly our test application in the shipbuilding industry. Finally, we summarize our conclusions from the paper.

Organizing product development collaboration

Inter-organizational product development processes can be differentiated on the basis of the interaction required for addressing the common development target. Therefore, not every product development task requires the same intensity of collaboration between the partners. Due to cost-benefit considerations the intensity and design of integration have to be based on the requirements of the existing development situation.

Basic approach – focusing interaction

To address issues of inter-organizational collaboration, our basic approach is to determine the appropriate level of collaboration intensity, based on the interaction needs of the development partners. Interaction needs can be understood as the kind and quantity of interactions that development partners need to fulfil their common objectives. Interaction intensity in product development can be specified by frequency of interaction, direction of interaction, and richness of transferred information.

- Frequency of interaction describes the number of interactions per time unit.
- Direction of interaction specifies whether one or both partners depend on the interaction.
- The richness of transferred information is a measure of the complexity of exchanged information. Text, for example, has lower richness than a drawing that contains equivalent information.

To realize different degrees of interaction intensity, one of three different formal mechanisms of interaction: coordination, cooperation, or collaboration, can be selected. All of these mechanisms are based on communication, which is defined as the exchange of information (Gronau, 2003).

Coordination is the management of dependencies between activities and the support of (inter)dependencies among actors (Bordeau and Wasson, 1997; Fischer, 1994). The aim of coordination is to integrate the contributions of the development partners. It therefore has an integrating and tuning function during task fulfilment. This mechanism is only sufficient if the subtasks can be clearly specified and can be solved by each partner individually. The richness of information needed for coordination is low to medium; the direction of interaction can be unidirectional or reciprocal, with low to high frequency.

Cooperation can be defined as the shared production of goods or services between distributed agents, organizational units, or organizations. This mechanism has to be used if the target of the development partners involves common problem solving, where each partner depends on the input of the other partner. The direction of interaction is necessarily reciprocal. Interaction frequency is usually medium to high.

Collaboration is a special case of cooperation and implies the joint execution of one task. Examples of collaboration in the field of engineering are the common distributed edition of a document or the common definition of product properties. With cooperation the subtasks are fulfilled uniquely by the development partners, who exchange start and end dates, task descriptions, and takeover conditions, to assure a common work result. With collaboration an integration of work fulfilment takes place. A clear distinction among partial actions is no longer possible. Collaboration is a coordinated, synchronous activity that is the result of a continued attempt to construct and maintain a shared conception of a problem (Dillenbourg *et al.*, 1995). The target of collaboration is integrated problem solving. This mechanism is characterized by reciprocal interaction at high frequency, and usually requires the transfer of rich information.

According to the described interaction mechanisms, we have defined three levels of collaboration intensity for product development: coordination, cooperation and collaboration. Which type should be chosen depends on the required interaction intensity caused by the interaction need of a specific development situation.

The required interaction intensity in product development increases:

- with increasing process-related interdependencies between the defined subtasks; and
- with increasing necessity of integrating the knowledge and experience of the development partners (Pfaffmann, 2001). The necessity for knowledge integration can be caused by specific characteristics of the development task as well as the partners' competencies in solving related problems. Development tasks can be specified by their complexity, level of innovation, and dynamics (Picot *et al.*, 1988). Complex development tasks are characterized by a high variety of elements and many interrelations between these elements. The level of innovation required determines how much knowledge or experience a company has available for solving this specific development task. The dynamics of a development task describe the number of changes must be made to the task specifications during the development process. These three characteristics determine the ex-ante-specifiability of a development task. This determines, in turn, how much a task can be specified at the beginning of the development process. The higher the ex-ante-specifiability, the better the concrete result, the solution approach to be used, and the subtasks and the relevant interfaces can be predefined. Therefore a high ex-ante-specifiability causes only a low interaction need during product development collaboration. The higher the competencies of each development partner concerning the solution of their development tasks, the less they need the knowledge of their partners to complete their subtasks. The need for interaction therefore increases with decreasing level of competence of one or all of the partners (Pfaffmann, 2001).

To determine the required interaction intensity according to the above described influence factors "interdependencies of subtasks, ex-ante-specifiability of development task and partner competencies" we use the relationships shown in Figure 1.

Ex-ante-specifiability of development task and interdependencies of subtasks are directly represented as the y-axis and x-axis in Figure 1. If product development is sequential interdependent, this means that only the result of one subtask is needed to start the other subtask. Reciprocal coordinated interdependence requires coordination between partners while processing their subtasks, whereas reciprocal integrated interdependence has a need for permanent communication and common problem solving between partners (see Wheelwright and Clark, 1994; Thompson, 1967)

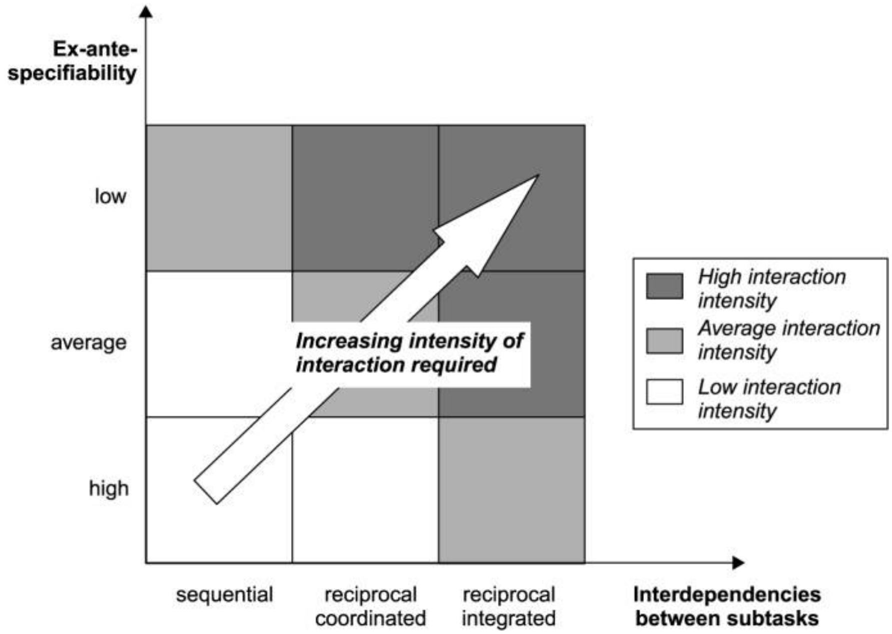

Figure 1.
Determination of the required interaction intensity

Partner competencies are indirectly represented here, based on the choice of portfolio axes, because they have an impact on both portfolio dimensions. Low competence or experience of at least one partner with the development task on the one hand decreases ex-ante-specifiability. On the other hand it influences the process-related interdependencies between the subtasks, because the less one partner knows the more the partner has to interact with the other partner to solve the assigned subtask.

The relationships shown in Figure 1 help to select an appropriate interaction intensity for efficiently designing product development collaboration.

Key dimensions of organizing product development collaboration

Designing inter-organizational collaboration requires on one hand to the ability to overcome spatial and time distances. That is why modern information and communication technologies (ICT) are required for integrating partners at the data and information level. On the other hand it means designing interrelations between tasks, human actors, information, and material in a way that the relevant targets of all involved companies can be reached as well as possible (Weinert, 2002). Therefore, besides the technical aspects of integration, aspects of organizational integration as well as integration of people have to be faced. A holistic framework for organizing product development collaboration must consider organizational, technological and human aspects (Warschat, 2003; Gronau and Kern, 2004).

To be able to develop a practically useable framework including all relevant dimensions and to address company needs, we carried out an empirical study to investigate the current challenges which industry must face while organizing product development collaboration (Kern, 2005). To collect the data, we interviewed experts from eight companies in six different industries: Plant Construction, Commercial Vehicle, Automotive, Locomotive, Aircraft and Shipbuilding Industry. Common characteristics of these industries are the high complexity of their products, a high division of labour, as well as a focus on network organization in product development. We used structured interview guidelines with open questions to avoid influencing the answers of the experts we interviewed. In the following, the main results of our study are outlined.

By asking for challenges and problems we received similar responses from the different industries. Experts identified the definition of one common target, as well as its common understanding, as essential pre-conditions for efficient collaboration. All experts named product technical specifications as core elements of the target agreement. Additionally they pointed out the necessity of specifying the inter-organizational development process. A common target definition is only possible if companies are willing to freely discuss their individual targets for product development. Therefore the experts identified openness as a main pre-condition for agreeing on a common target definition.

Another important challenge in designing inter-organizational collaboration is to overcome inter-company barriers. The higher the required intensity of collaboration, the more important is the integration of partners. To create intense collaboration, the development partners have to be aware of company specific differences and attitudes. Therefore they must be willing to get involved with each other, which requires the creation of trust. Only then it is possible to systematically take measures for overcoming the different possible barriers. Inter-organizational obstacles can arise from non-collocation, different company-specific characteristics, (e.g. company culture, organizational and technical pre-conditions), and different nationalities of the development partners.

- In situations with high interaction needs, such as the discussion of complex problems, the lack of face-to-face communication was identified as the main challenge from not being collocated. In these cases the use of ICT only is an inadequate substitute for personal meetings.
- Problems that could be identified due to different company cultures are differences in openness, in communication attitudes, and in reliability of the partners. This complicates building trust and decreases ambition for creating a cooperative atmosphere.
- Organizational differences that are identified as causing possible difficulties with collaboration are on the one hand different decision-making processes and decision structures which cause complications with making decisions. On the other hand different departmental structures and tools can lead to difficulties in finding an appropriate contact person.
- A major problem with collaborative product development was the level of heterogeneous ICT infrastructure. Different computer-aided design (CAD) systems and communication tools decrease the efficiency of collaboration by complicating data exchange and interaction between partners.
- Cultural interfaces due to differences in language, national attitudes, values, working mentality, way of thinking, problem solving, level of education, measurement systems, etc. can all influence the efficiency of collaboration. On the one hand the technical quality of the product may suffer, while on the other hand collaboration may become more difficult due to misunderstanding and complications at a personal level.

The more closely companies have to work together, the more the described barriers lead to inefficient work. To solve this problem, it would be appropriate to define measures according to the required level of collaboration intensity. But although companies tend to address the relevant key dimensions, they do not methodically design the related processes based on interaction needs. The expert interviews showed that the design of inter-organizational product development is mainly based on experience, and emerges during the ongoing collaboration process.

Therefore our aim is to develop a framework which includes not only the relevant key dimensions but additionally provides guidelines for different intensity levels of partner integration.

Developing the framework

Based on our approach of focusing on interaction, the state of the art in this field, and the results of our empirical study, we derived requirements for a practically useable framework that internet supported inter-organizational product development collaboration should fulfil:

- The main idea is to determine the level of collaboration intensity according to the interaction needs of the partners in their specific development situation.
- ICT is one main element, because it enables spatial collaboration.
- Organizational, technical and human aspects must be considered.
- The framework has to provide specific guidelines which can be easily implemented in industry.

In the following we set up and describe our framework with the help of a portfolio of interaction types.

Setting up the framework

The basic approach of our framework is to specify first the required interaction intensity to undertake the defined development task. This specification must be discussed with experts from all the participating development organizations, as described in Figure 1. Based on this approach we choose the required interaction intensity as the x-axis dimension of the portfolio in Figure 2.

Integration is necessary if interactions between the distributed development partners are to be realized. An enabler to overcome spatial distances is modern ICT. As shown in section 1, ICT offers various functionalities that can help to integrate the development work, so technical integration is a core element of our framework. Different groups of tools help to support different interaction intensities. For this reason we the different classes of ICT tools described in section 1 as the y-axis dimension, indicating functionalities to support personal communication or the synchronous or asynchronous use of common material.

The resulting portfolio shows nine possible types of product development collaboration. Based on cost-benefit considerations, the extent of technical integration has to be selected in a way that the interaction needs of the development partners can be satisfied as well as possible. We identified only three of the nine possible types that were able to meet the requirements economically.

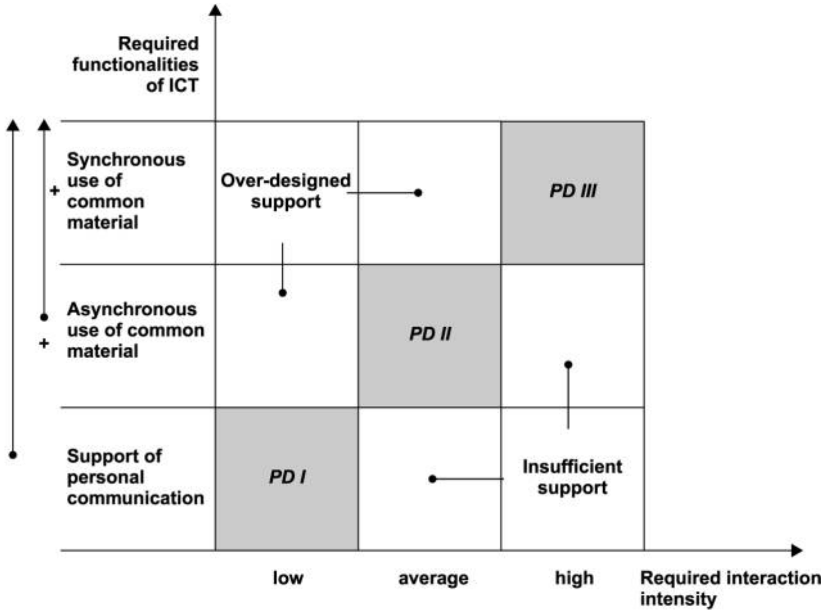

Figure 2.
Portfolio for selecting the appropriate interaction type

As Figure 2 shows, only the types situated on the diagonal of the portfolio offer the right range of ICT support. We named them after their corresponding interaction mechanisms coordination (PD I), cooperation (PD II) and collaboration (PD III):

- PD I is characterized by usage of the coordination interaction mechanism. To efficiently support this type of PD, ICT tools to support personal communication are sufficient.
- PD II is based on the cooperation interaction mechanism; its target is common problem solving. In addition to tools supporting personal communication, it requires tools that enable the asynchronous processing of common material. This is evident because complex development tasks have to be undertaken which involve the transfer of rich information such as CAD drawings. This transfer cannot be handled efficiently with tools that only support personal communication.
- PD III uses the collaboration mechanism because it involves integrated problem solving. In addition to tools supporting personal communication and asynchronous processing, the use of tools for synchronous processing are also inevitable because of the high interaction intensity required.

All the other types of support in Figure 2 represent real existing designs, but they do not use ICT according to interaction needs. Whereas the three types shown at the top left have an over-designed functional range, the types situated at the lower right do not offer sufficient support to realize the required interaction intensity. This results in an overly time consuming coordination effort in order to achieve the desired level of collaboration. Companies should therefore try to select ICT support according to the three ideal types we have identified.

It should be pointed out that the portfolio in Figure 2 helps to define different types of product development and shows which ICT tools should be chosen to support different intensities of interaction. How to design organizational and key human dimensions like personal integration according to the PD-type is described in the following section.

Levels of integration for product development collaboration

As pointed out, ICT is an enabler for designing cross organizational product development collaboration. According to our empirical findings and theoretical insights, collaboration requires further integration of the partners at different levels. In addition to technical integration through ICT, we have identified the following five important levels of integration:

(1) The initial point of product development collaboration and therefore an inevitable precondition of successful collaboration is the integration of partner targets. The development partners are connected through a common development task. Depending on the required interaction intensity for solving this development task, additional needs for target integration can arise at the process level as well as the company level. The higher the intended intensity of interaction, the more the partners have to agree upon a common design for cross-company processes and/or company cooperation strategies.

(2) Teamwork is an essential characteristic of product development. In cross-organisational collaboration, team members of different companies have to work together. Typically, team members are not collocated; due to a lack of ongoing face to face contacts, team building processes turn out to be more difficult and slower than in traditional, collocated teams. Therefore it is essential for efficient collaboration to integrate team membership at a personal level. This may be achieved by personal meetings, temporary exchange of team members, or the so-called ongoing collocation (Groher, 2003).

(3) Organizational integration means creating an appropriate organizational structure. Product development collaboration is usually executed as a project, it is appropriate to implement organizational integration on the basis of a project organization. Depending on the selected division of work and the required interaction intensity, two relevant specifications of project organization can be defined: interface-oriented or team-oriented (Müller-Stewens and Gocke, 1995). In the interface-oriented model, also known as black box development, the producer undertakes the specification of product components (Wildemann, 1996). The problem solving activities of the supplier are then in the technical realisation of predefined specifications. The supplier is completely responsible for the development results as well as being in control of the development process (Groher, 2003). To integrate the development partners, milestones are defined at which specified work packages are to be finished. The team-oriented model is characterized by tight collaboration between producer and supplier. Depending on the level of insight the producer has with the supplier's development process, glass-box and grey-box models can be differentiated. Glass-box development means close cooperation in specifying and developing the product. Competencies of both partners are necessary for a successful result.

Important bases in this situation are trust and openness, because both partners have to get deep insights into the developments undertaken by the other partner. Grey-box development is a type of glass-box development, but with a lower level of producer insight into the development activities of the supplier. The main contribution of the supplier is specific knowledge; the main contribution of the producer is the best possible integration of the supplier's contributions (Wildemann, 2000). Contrary to the black-box development, where only contact persons are named, partner integration in team-oriented models is realized by defining a inter-organizational project team.

(4) Creating a common knowledge base can support the collaboration process. Therefore an integration of partners by combining their knowledge in relation to the development task can increase the efficiency of product development. Additionally, common collective knowledge is developed during the collaboration process. This includes knowledge, not just about the development task, but about the collaboration process itself such as specific partner competencies and characteristics. The common knowledge base results in simplifying repeated collaboration, but this can cause a strong dependency between partners. A precondition to success in this situation is mutual trust. A first step in creating a common knowledge base is the evaluation of the common project. Beside relevant aspects of technical problem solving, experience derived from collaboration should be included. In a second step the insights gained must be prepared, edited, and made available as lessons learned. A possibility for tightly integrating the development partners is by implementing an inter-organizational knowledge management system to ensure that a target oriented common knowledge base is created.

(5) Trust is a main precondition for successful product development collaboration. Trust is the expectation of a person regarding the attitude of interaction partners. Each inter-organizational collaboration requires a minimum of trust, because this fosters cooperative behaviour, leading to a reduction in conflict (Jones and George, 1998). To a certain extent, trust can be created, especially in the early phase of product development collaboration, by contracts or verbal arrangements. Real trust has to develop during the ongoing process of collaboration. It mainly results from positive experience with development partners (Beck, 1998). A key to building trust is open attitudes of the development partners as well as the creation of a common cooperation culture. Cultural integration is the development of a common cooperation culture based on trust, enabling the development partners to overcome cultural differences and work together effectively. People have to become aware of cultural differences and they must learn to accept them. For this reason it is essential to develop inter-cultural knowledge through specific training and experience.

Figure 3 summarizes the five suggested integration levels and their specific modes of design.

Implementation of all integration measures requires a significant amount of planning and effort. To maximize the likelihood of success, it is necessary to select the integration measures appropriate to the specific needs of interaction.

Integration levels	Modes of design			
Integration of targets	Development task	Process (aspects of time)	Process (aspects of content)	Strategy
Personal integration	Personal meeting	Timely exchange	Co-Location	
Organizational integration	Black-Box	Grey-Box	Glass-Box	
Integration by knowledge and learning	Project evaluation/lessons learned	Cross-organizational knowledge management		
Integration by trust and culture	Contract based trust	Common cooperation culture		

Figure 3.
Levels of integration and their specific design modes

Formulating guidelines for the design of ideal interaction types

Our framework characterizes three ideal interaction types. In the following we formulate design guidelines for each type, addressing all the relevant levels of partner integration.

PD I is used for coordination between development partners (Figure 4). In this situation, mostly explicit knowledge is exchanged. Due to the required intensity of interaction, the target integration level must include development tasks as well as the timely matching of the development process. The development tasks as well as the subtasks have to be specified in detail. All milestones, when the contributions of the partners have to be delivered, have to be defined. Due to the low interaction intensity it is not obligatory that contact persons meet personally; nevertheless occasional personal meetings may be helpful. For organisational integration the black-box model

Integration levels	Modes of design			
Integration of targets	Development task	Process (aspects of time)	Process (aspects of content)	Strategy
Personal integration	Personal meeting	Timely exchange	Co-Location	
Organizational integration	Black-Box	Grey-Box	Glass-Box	
Integration by knowledge and learning	Project evaluation/lessons learned	Cross-organizational knowledge management		
Integration by trust and culture	Contract based trust	Common cooperation culture		

Design mode required   *Design mode possible*

Figure 4.
Design guidelines for PD I

should be selected. The integration by knowledge and learning can be carried out using project evaluation. Integration by trust and culture can usually be accomplished through contracts.

The focus of PD II is common problem solving, which requires a more intensive interaction and therefore a tighter integration of development partners than PD I (Figure 5). In addition to the integration of development tasks and the timely matching of development processes, it is necessary to agree on the design of processes. Additionally the partners should define a common cooperation strategy. Personal integration for this type requires personal meetings between the team members because they have to work closely. The exchange of implicit knowledge can be required. Personal exchange or even collocation should be discussed. For organizational integration the grey-box model is appropriate. In any case a project evaluation should be carried out. If repeated collaboration is planned or likely, it could be helpful to implement inter-organizational knowledge management. PD II requires more extensive strategic considerations than PD I. Therefore the development partners should try to create a common cooperation culture in addition to the contract-based creation of trust. Due to the more intensive integration of PD II, repeated or medium to long term collaboration should be a planned prospect.

The aim of PD III is integrated problem solving (Figure 6). Development partners must collaborate extensively to solve common development tasks. It is of high relevance to not only communicate about product and process, but also about a common strategy. The definition of strategy goes well beyond the requirements of PD II, since PD III requires a high degree of trust and openness between the partners. In order to realize an appropriate intensity of interaction, far reaching measures have to be taken at all integration levels. For personal integration, collocation is essential, because tight collaboration demands a detailed matching of working and problem solving methods, processes and tools. The exchange of implicit knowledge has a high value in a PD III situation. A trusting atmosphere is essential, because the development

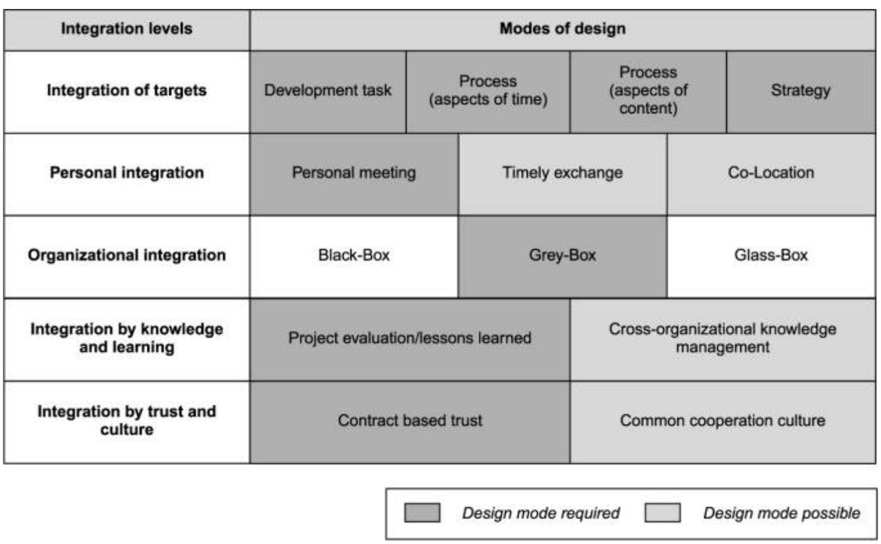

Figure 5.
Design guidelines for PD II

Integration levels	Modes of design			
Integration of targets	Development task	Process (aspects of time)	Process (aspects of content)	Strategy
Personal integration	Personal meeting	Timely exchange	Co-Location	
Organizational integration	Black-Box	Grey-Box	Glass-Box	
Integration by knowledge and learning	Project evaluation/lessons learned	Cross-organizational knowledge management		
Integration by trust and culture	Contract based trust	Common cooperation culture		

Design mode required *Design mode possible*

Figure 6.
Design guidelines for PD III

partners work together on the same development tasks. Therefore organizational integration should be in the form of a glass-box model, with a maximum of openness to be able to achieve the full performance potential of all the partners.

The integration by knowledge and learning must be realized by implementing a cross organizational knowledge management. This type requires a common cooperation culture. Due to the huge integration efforts that have to be made for realising the adequate interaction intensity, PE III can only be economically realized in long term cooperations.

Test application in the shipbuilding industry

The proposed framework was partially tested in a joint research project with the ship-building industry (Bronsart *et al.*, 2002). Based on a defined collaboration scenario, a ship yard and three of its development partners used our guidelines to design the organizational and technical integration of the partners. Due to its challenging nature the rudder system design phase was chosen. Within this test case the shipyard is responsible for the design of the whole ship, whereas a supplier delivers the design of the rudder machine. The concept of the rudder machine room as an interface to the ship hull is done by an engineering consultant. A certification society is responsible for the certification of the rudder machine and its final assembly in the hull (Koeppen *et al.*, 2004).

The defined development task is characterized by a low ex-ante-specifiability as well as reciprocal integrated interdependencies of partner subtasks. Thus it requires a high interaction intensity and PD III must be selected as the appropriate interaction type. For organizational integration this leads to the glass-box approach, to provide for high transparency of development processes and the implementation of a comprehensive information and communication structure. In our case we decided to set up an internet-based information and communication platform (IC-Platform). As described in section 3 the full scale of ICT functionalities for personal communication as well as asynchronous and synchronous use of common material was implemented.

To define the detailed measures for organizational integration as well as for the design of the IC-Platform, we carried out a detailed process analysis, including possible disturbance variables. This ensured a common process understanding, resulting in the definition of an efficient inter-organizational development process. Based on this we were able to create an information flow model, which included the required communication structure as well as the content and objects of communication. This approach was used for the design of the IC-Platform and the choice of ICT functionalities. The target of the organizational and "technical" integration was to minimize collaboration effort. The overall performance and productive efficiency were tested by the industrial partners, and approved the applicability of our proposed approach for partner integration to design product development collaboration.

Conclusion and outlook

The design of distributed product development processes is a core element of supply chain management. For this reason, companies need guidelines for an efficient and effective design for development collaboration. Our analysis identified three appropriate types of partner interaction, and developed specific design guidelines for each type. The framework described can help companies to configure their cross organizational development processes according to their needs. It therefore has a great potential to contribute to company competitiveness. The framework was partially tested in a joint research project with the ship-building industry. Based on a defined collaboration scenario, one ship yard and three of its development partners used our guideline to design organizational and technical integration. The results showed that the resulting design of product development collaboration could fulfil the requirements of the industrial partners. Nevertheless, the framework as a whole must be tested further in practice in order to analyse its usability in industrial applications.

References

AberdeenGroup (1999), "Collaborative product commerce, delivering product innovations at internet speed", *Market View Point*, Vol. 12 No. 9.

Accenture (2001), *Collaborating to Compete, A New Way Forward*, Pan-European Collaborative Product Development Survey 2001, November.

Beck, T.C. (1998), *Cost-efficient Network Cooperation: Optimization of Complex Partnerships between Companies*, Wiesbaden (in German).

Bordeau, J. and Wasson, B. (1997), "Orchestrating collaboration in collaborative telelearning", in Boulay, B. and Mizoguchi, R. (Eds), *Artificial Intelligence in Education*, IOS Press, Amsterdam, pp. 565-7.

Bronsart, R. *et al.* (2002), *Proposition for the Joint Research Project "ShinCoS" (Shipbuilding Information and Communication Structure)*, Rostock (in German).

Bullinger, H.-J. (2003), "Collaborative development – potenzials of success in development networks", *Supply Chain Management*, Vol. II, pp. S. 61-S. 64 (in German).

Contero, M. (2005), "Collaborative engineering", in Li, E.Y. and Du, T.C. (Eds), *Advances in Electronic Business*,Volume I, IGI Publishing, Hershey, PA, pp. 87-122.

Dillenbourg, P., Baker, M., Blaye, A. and O'Malley, C. (1995), "The evolution of research on collaborative learning", in Reimann, P. and Spada, H. (Eds), *Learning in Humans and Machines: Towards an Interdisciplinary Learning Science*, Pergamon, London, pp. 189-211.

Fischer, T. (1994), *Coordination of Business Control Tasks in a Framework of Integrated Enterprise Information Systems*, Renningen (in German).

Groher, E. (2003), *Designing the Integration of Suppliers in the Product Development Process*, München (in German).

Gronau, N. (2003), "Collaborative engineering communities – architecture and integration approaches", in Khosrow-Pour, M. (Ed.), *Information Technology and Organizations, Trends, Issues, Challenges and Solutions*, *Proceedings of the 2003 Information Resources Management Association International Conference, Philadelphia, PA, 18-21 May*, pp. 792-795.

Gronau, N. and Kern, E.-M. (2004) in Camarinha-Matos, L.M. (Ed.), *Virtual Enterprises and Collaborative Networks – IFIP 18th World Computer Congress TC5/WG5.5 – 5th Working Conference on Virtual Enterprises, Toulouse, 22-27 August*, Kluwer Academic Publishers, Dordrecht, pp. 329-38.

Jones, G.R. and George, J.M. (1998), "The experience and evolution of trust: implications for cooperation and teamwork", *Academy of Management Review*, Vol. 23, Jg., pp. 531-46.

Kern, E.-M. (2005), *Distributed Product Development: Framework and Guidelines for the Organizational Design*, Habilitationsschrift, Technische Universität Hamburg-Harburg (in German).

Kersten, W. and Kern, E.-M. (2003), "Distributed product development as a core element of supply chain management", *Proceedings of the 2003 IEEE Conference on Emerging Technologies and Factory Automation, Lisbon, 16-19 September*, Volume 2, pp. 35-39.

Koeppen, B., Meyer, Ch. and Kern, E.-M. (2004), "Definition of the research target and analysis of development processes", Project Internal Working Report, Joint BMBF-research Project ShinCoS (in German).

Krcmar, H., Zerbe, S. and Schwarzer, B. (2000), "New forms of organization and ICT: challenges for their design considering global teams as example", in Wojda, F. (Ed.), *Innovative Organizations*, HAB Forschungsberichte Band 12, Stuttgart, pp. 291-321 (in German).

Monplaisir, L. and Haji, T. (2002), "Fundamental concepts in collaborative product design and development", in Monplaisir, L. and Singh, N. (Eds), *Collaborative Engineering for Product Design and Development*, GBV, Stevenson Ranch, CA, pp. 2-10.

Monplaisir, L. and Salhieh, S.M. (2002), "Collaborative product design and development: a case study integrating computer-supported collaborative work with quality function deployment", in Monplaisir, L. and Singh, N. (Eds), *Collaborative Engineering for Product Design and Development*, GBV, Stevenson Ranch, CA, pp. S. 126-S. 139.

Müller-Stewens, G. and Gocke, A. (1995), *Cooperation and Concentration in the Automotive Industry*, Chur (in German).

Pfaffmann, E. (2001), *Competence-based Management in Product Development*, Wiesbaden (in German).

Picot, A., Reichwald, R. and Nippa, M. (1988), *The Meaning of the Development Task for the Development Time – Approaches for the Design of the Development Time*, ZFBF-Sonderheft 23, Zeitmanagement in Forschung und Entwicklung, pp. 112-137 (in German).

Thompson, J.D. (1967), *Organizations in Action*, McGraw Hill, New York, NY.

Warschat, J. (2003), "Virtual engineering", in Bullinger, H.-J., Warnecke, H.J. and Westkämper, E. (Eds), *Neue Organisationsformen im Unternehmen, Ein Handbuch für das moderne Management, 2*, neu bearbeitete und erweiterte Auflage, Berlin, pp. 842-860.

Wheelwright, St. C. and Clark, K.B. (1994), *Revolutionizing Product Development*, Simon & Schuster, Frankfurt and New York, NY.

Weinert, P. (2002), *Organization*, München (in German).

Wigand, R., Picot, A. and Reichwald, R. (1997), *Information, Organization and Management*, Wiley, Chichester.

Wildemann, H. (1996), *Development Strategies for Suppliers, 3*, neu bearbeitete Auflage, München (in German).

Wildemann, H. (2000), *Analyzing Sourcing Potenzials – Programs for the Collaborative Opening of Rationalizing Potenzials*, München (in German).

Willaert, S.S.A., de Graaf, R. and Minderhoud, S. (1998), "Collaborative engineering, a case study of concurrent engineering in a wider context", *Journal of Engineering and Technology Management JET-M*, Vol. 15 No. 1, pp. 87-109.

About the authors

Eva-Maria Kern is an assistant professor at the Department of Business Logistics and General Management, Hamburg University of Technology (TUHH). She studied plastics engineering, and holds a PhD in Process Engineering and a Master of Business Administration in Generic Management. Eva-Maria Kern is the corresponding author and can be contacted at: e.kern@tu-harburg.de

Wolfgang Kersten is a professor and President of the Hamburg School of Logistics (HSL) and professor and head of the Department of Business Logistics and General Management at Hamburg University of Technology (TUHH). He studied industrial engineering at the Technical University of Darmstadt and received his PhD from the University of Passau.

This article has been cited by:

1. AkoumianakisDemosthenes, Demosthenes Akoumianakis, KtistakisGeorge, George Ktistakis. 2017. Digital calendars for flexible organizational routines. *Journal of Enterprise Information Management* **30**:3, 476-502. [Abstract] [Full Text] [PDF]
2. Lyubov L. Tmanova, Jessica S. Ancker, Stephen B. Johnson. 2015. Integrating an Informationist Into Graduate Education: Case Study With Preliminary Results. *Medical Reference Services Quarterly* **34**:3, 296-310. [Crossref]
3. Vinayak Kalluri, Rambabu Kodali. 2014. Analysis of new product development research: 1998-2009. *Benchmarking: An International Journal* **21**:4, 527-618. [Abstract] [Full Text] [PDF]
4. Ricardo L. Punzalan. 2014. Understanding Virtual Reunification. *The Library Quarterly* **84**:3, 294-323. [Crossref]
5. Mahdi Bohlouli, Fabian Merges, Madjid Fathi. Knowledge integration of distributed enterprises using cloud based big data analytics 612-617. [Crossref]
6. Demosthenes Akoumianakis. 2014. Ambient affiliates in virtual cross-organizational tourism alliances. *Computers in Human Behavior* **30**, 773-786. [Crossref]
7. Demosthenes Akoumianakis, George Ktistakis, Giannis Milolidakis, Konstantinos Michailidis. Enacted virtual partnerships: Imbrications of representations of boundary artifacts and cloud services 1-7. [Crossref]
8. Mahdi Bohlouli, Alexander Holland, Madjid Fathi. Knowledge integration of collaborative product design using cloud computing infrastructure 1-8. [Crossref]
9. Demosthenes Akoumianakis, Nikolas Vidakis, Anargyros Akrivos, Giannis Milolidakis, Dimitrios Kotsalis, George Vellis. 2011. Building 'Flexible' vacation packages using collaborative assembly toolkits and dynamic packaging: The Case Study of the eKoNES. *Journal of Vacation Marketing* **17**:1, 17-30. [Crossref]
10. Demosthenes Akoumianakis, Giannis Milolidakis, Anargyros Akrivos, Zacharias Panteris, Giorgos Ktistakis. Clinical Practice Guideline Management Information Systems: Cancer Guidelines as Boundary Spanning Tranformable Objects of Practice 230-237. [Crossref]
11. Demosthenes Akoumianakis. 2009. Practice-oriented toolkits for virtual communities of practice. *Journal of Enterprise Information Management* **22**:3, 317-345. [Abstract] [Full Text] [PDF]
12. Leandro Gabrieli, Marcelo Cortimiglia, José Luis Ribeiro. 2007. Modelagem e avaliação de um sistema modular para gerenciamento de informação na Web. *Ciência da Informação* **36**:1, 35-53. [Crossref]
13. Demosthenes Akoumianakis. Cross-Organization Virtual CoP 265-282. [Crossref]