Distributed product development presents a particular challenge to the knowledge management in small and medium-sized companies. Barriers for information exchange arise from the separation of organisational units, separation of projects and separation of the development process. However, to make reliable decisions in the early phases of the development, information must be provided to designers gathering requirements and elaborating the product concept. In this paper, a methodical approach is presented, which aims on a better integration of information in value-added networks. The principal core of the approach is an information model, combining partial models of requirements, system components and suppliers. Introducing this with a web-based tool, a step towards better decisions in development could be made.

Keywords: product lifecycle, information and knowledge management; collaborative design; systems architecting; conceptual design

1. Introduction

Todays significance of distributed product development has brought a new emphasis to an efficient knowledge management. Thus, for many companies it has become necessary to perform the development process across multiple geographic locations in mostly independently operating departments or companies. Consequently, especially small and medium-sized enterprises (SME) often participate in OEM centered value-added networks. In this way, they are able to focus their special competences in collaboration. Such networks, for instance, can be found in the development of mechatronic products, which integrate components of the mechanical and electrical engineering and information technology. An example for a network of a system developer is shown in Fig. 1.1.

Each company assumes a specific role in these networks, whereas the degrees of cooperation differs widely: Companies could collaborate as producers of single components, while others cooperate as dependable partners in design and production. Especially with mere producers knowledge exchange is very limited. Even though, both system developer and supplier could benefit from closer cooperation [2].

1.1. Barriers for Knowledge

Apart from the designers creativity, one key factor for successful product development is the available knowledge as basis for decisions [1]. However, this knowledge is often not sufficiently available when it is needed for decision-making, but rather exists in later stages of the process. For example, during conceptual design, decisions on needed functions or suitable solution principles are made as a basis for the product to be developed. The knowledge about different options for product implementation is not available as the manufacturer gets involved only later in the process. Thus, the concept engineer has to make assumptions. As it turns out later, these decisions might be incorrect or non-optimal, expensive and time-consuming iterations are needed.
Besides this temporal discrepancy of knowledge availability, boundary conditions as independently acting organisational units or functional separation of project processing complicate the knowledge exchange between knowledge carriers. These barriers promote the emergence of so-called islands of knowledge during the design process [5].

1.2. Objective

In many cases, even in SME applicable methods and tools are missing to handle and communicate product and process information across geographical distances and different disciplines efficiently. The methodical approach presented in this paper aims to facilitate a profitable cooperation with suppliers due to an enhanced exchange of knowledge. As basis, an information model is described, enhancing an easy sharing of information in distributed product development.

1.3. Structure

In following chapter 2, the barriers for knowledge in distributed product development are analysed and general challenges of information exchange are identified. On this basis, in chapter 3 the problems of an efficient knowledge management are clarified, before the idea of the methodical approach is presented in chapter 4. Thereafter, the information model as its core element is described in detail in chapter 5. The paper ends with a conclusion and an outlook.

2. Knowledge in distributed product development

Knowledge is the competence of individuals in dealing with complexity. Thus, the success in complex product development with many different options to choose is mainly based on knowledge [3]. However, as knowledge is bound to individuals in the form of tacit knowledge, a direct exchange is not possible [4]. Rather, a resource is needed, which could be extracted by knowledge carriers and from which other person could generate knowledge. This resource is information [5].

2.1. Knowledge in the Process of Product Creation

A knowledge-based product development aims on the provision and combination of knowledge at earliest possible decision points. Thus, decisions in the early phases of the process determine most of the costs, whereas they are caused only later. However, the knowledge necessary for decision-making is often non-sufficient available at the time needed within the process [6].

Especially in distributed product development, this temporal discrepancy is difficult to overcome. Available knowledge in early phases is mainly based on experience of single persons. However, reliable information is hold by departments or companies, that are only later involved in the process. For example, knowledge regarding manufacturing limitations could only be considered by the company that will later participate as a supplier of a subsystem. Often, this supplier is not even defined and will be chosen later. The decision could only be made on basis of estimations founded on the experience of a specific person, but not expert.

2.2. Islands of knowledge in distributed product development

Besides the temporal discrepancy, as an extra dimension, the organisation is added to the challenges in distributed product development to bring knowledge together. Barriers between organisation units prevent combination of knowledge. As a third dimension the inaccurate connections between current projects and experience from other projects hinders the access to existing knowledge.

These three dimensions process, organisation and projects creating barriers between interrelated knowledge, are represented in Fig. 2.

![Fig. 2. Dimensions of separation causing islands of knowledge (see also [7])]({})

These dimensions of knowledge separation are the barriers to be overcome for an efficient knowledge management.

2.3. Information pathologies

In the process of knowledge exchange, problems could occur, that effect a lack of information needed for decision-making. Four types of avoidable mistakes could cause these so-called information pathologies [8]: information that is not produced at all, is not acquired, is not or incorrectly transmitted and information that is misunderstood or not put to use.

In interdisciplinary collaborative product development, many information pathologies could be identified, which cause non-optimal decisions. Assuming that all required information exists at any time of the process, following main problems were identified in current research:

1. **Production**: Information does not exist at all, because it is not produced by individual persons (as knowledge carriers). Especially the innovative strength of SME often depends on individual persons and their tacit knowledge, that is not directly transmittable and long-term storable.

2. **Acquisition**: Information is not acquired, i.e. information exists, but the decision-maker does not know, that it could be acquired. In distributed development, this could be caused by complex organisational structures with various departments and companies, but also by separated project
processing. The decision makers do not overview the information.

3. Transmission: Information could not be transmitted correctly because of a lack of communication technology or compatibility of data formats. Nevertheless, there is a high availability of web-based technology, it is often not used efficient in collaborative development. This is also caused by different tools and data formats used by the different partners.

4. Understanding: Information is available for the decision maker, but this person does not know its relevance, because the understanding of the complete complex system is missing.

Fig. 3 shows these four pathologies of information exchange in a distributed product development environment. In this example, the information represents parts of a comprehensive model as all information reference to the same product development project.

Fig. 3. Process of information exchange in distributed product development showing the four information pathologies.

3. Problem Clarification

The methodical approach presented in this paper focuses on preventing information pathologies in product development with various distributed partners. Thus, the forming of islands of knowledge is prevented due to the use of methods and tools improving the exchange of information.

3.1. Current Status

The approach introduced, is based on an analysis of various distributed product development processes [9]. Companies of different types collaborate in an interdisciplinary product development process of a mechatronic product. One company takes in the role of the system developer and acts as the centre of the network, as shown in Fig. 1. Hence, the product development process is mainly defined by the system developer which integrates different other companies with specific expertise.

In case studies, particularly project documentation and information flows were analysed and current challenges and potentials for improvement were identified. The approach presented in this paper address these specific points in these specific networks. Nevertheless, it could be transferred to similar processes that are characterised by the following:

- Various persons, departments or companies are integrated as independent acting stakeholders at different stages in the process.
- The collaborating partners vary from project to project, but are mostly taken from the same pool of favoured partners.
- The product development task is initiated through customers. The products are customised, but base on known partial solution.

3.2. Process of product development

To describe the challenges and potentials, in the following a process visualisation is chosen, representing a typical procedure beginning with the customer inquiry and ending with a close-to-production prototype (see Fig. 4).

From each of the phases results an interim outcome as a milestone, which represents the decisions made in the previous phase. These decisions define costs, caused later in the process. Besides the main process set by the system developer, three further paths of collaborating suppliers are described. These represent independently acting organisational units involved later in the process.

In the phases from the inquiry to the tender to the requirements, the customer (as a resource of information) is involved into the process. Based on the requirements given by the customer, the system developer is called to define the system design and concerning system specifications. Thereafter these decisions affect the activities of the suppliers.

The procedure in the considered networks is similar to the V-Model given in [10]. Based on the definition of the task, a cross-domain system design is performed, which is concretised in the specific domains and finally integrated to the whole product. Before the start of the project, the system developer already works in advance on the tender of the project. In this preliminary stage, the launch of the project is uncertain. Because of this reason the effort must kept small.

The introduced development procedure is a general description and real processes will differ from this. Especially, unplanned iterations could be necessary. For instance, if problems or errors are identified during the domain-specific design, the realisation or integration, then these have to be corrected through changes in the system design.
3.3. Challenges and potentials

The introduced process highlights the essential challenge of distributed product development: Suppliers are integrated late in the process (usually beginning in domain-specific design), but far-reaching decisions already have to be made in the early stages without involving suppliers (project preparation, task clarification and system design). Unfavourable decisions could effect different problems. For example, if the costs and time deadlines given with the tender to the customer could not be met, the system developer has to bear extra costs. If the requirements are not completely gathered or the system design is not feasible, the decisions have to be corrected in downstream phases. At the worst, iterations are necessary and phases have to be run through again.

Thus, the methodical approach aims for better decisions in early phases of the process. This should be reached through an efficient provision of information to the decision makers in the respecting phases. For this, the basis is to overcome the three dimensions causing islands of knowledge:

- **Temporal discrepancy (process):** For decisions made in specific phases, information referring to other phases should be available. This could be achieved by a long-term established knowledge base, including information on necessary decision and its effects. For example, components specifications needed in the domain-specific design could already be defined in the tasks clarification in agreement with the customer (and not afterwards between suppliers, system developer and customer).
- **Separated organisational units:** Between departments and collaborating companies, an information exchange about requirements and organisational restrictions should take place. For example, manufacturing restrictions from suppliers are known to the engineers of the system developer carrying out the system design.
- **Project-specific separation:** Information from other projects as requirements and product information should be prepared for reuse in new projects. A connection of this information could facilitate the detection and using in current work. For example, requirements from older projects on similar systems could be also defined for actually developed systems and are not left out.

4. Methodical approach

Basic idea of the approach presented in this paper is to suitably prepare and provide knowledge, normally available at unfavourable time and place, to decision-makers in early phases of the development process. In this way, islands of knowledge are connected.

4.1. Cross-company Information Exchange

To close the gaps between islands of knowledge, the four information pathologies are basis for the new approach:

1. The approach guides the designer through the process and enable him to produce information, using modelling tools.
2. Connections between information on a meta level allow to recognise links between produced information and enable decision makers to detect and acquire relevant information.
3. A cross-company used platform allows a direct transfer of information.
4. A consistent way of describing information with the models allows a comprehensive understanding of the information in actual context.

The current work aims on rectifying information pathology two and four. The previous chapter has shown, that here most potentials could be exploited in distributed product development, because information generally exists, but is not connected in a right way to be detected by decision makers. A comprehensive information model, integrating cross-company knowledge, enhances the correct understanding.

4.2. Connection of information on a meta level

There are many kinds of information relevant for decisions. In this paper, the focus lies on three types of models, in between which connections are often overseen: A requirements model, a system structure model and a supplier model. These three models are connected on a meta level, which map relations between the different types of information (see Fig. 5).

![Fig. 5. Basic information for decision-making, connected on a meta level.](image)

The requirements model consists of specifications of the product as well as restrictions resulting from processes and resources of companies [11]. The requirements refer to the system or system parts. The system model itself contains the system elements (as system parts or components) and the connections in between (as structure of the product). In this way, especially interfaces between system elements are modeled, improving the distributed development.

The suppliers are linked to both partial models: Requirements could be caused by them or be relevant for them [12]. System elements connect requirements and suppliers, as the suppliers takes responsibility for it and, therefore, should be involved in its specification.

5. Information Model

The information model can be divided into three levels: the knowledge base level, the meta level and the project level.
Thereby, the meta level connects all information on an abstract layer and establishes the link between the other two levels. In this way, it fulfills two tasks: First, it enables to identify relevant information for active projects on the project level within the knowledge base. Second, if no information can be identified, knowledge gaps to be closed are exposed. Fig. 6 shows the schematic structure of the information model. Each level contains the same three types of partial models.

Fig. 6. Structure of the information model on three levels.

The meta level consists of abstract elements of each partial model. These elements (abstract requirements, abstract system elements and abstract suppliers) represent project-independent place holders for concretised elements from the knowledge base. In development, these abstract elements are used as checklists (requirements), product templates (system elements) and competency filters (suppliers) for modeling the partial models on project level. The abstract elements are connected within a partial model and with elements of the other partial models. Thus, abstract requirements are connected to correspondent abstract system elements and abstract system elements to abstract suppliers as denoted on the meta level in Fig. 6.

The knowledge base level consists of existing elements classified in the requirements pool, the system elements pool and the suppliers pool. Elements on this level represent information from other projects. All elements own a specific set of specified attributes. Requirements for instance own attributes e.g. representing unique ids, names and values. Moreover, these elements are linked to further data, like documents or CAD models.

On project level, the partial models for active projects are derived. In these models, the specified elements and their relationships are modelled based on abstract elements from the meta level. The elements of this level have correspondent elements in the knowledge base. Thus, the project level shows a project specific subset of elements from the knowledge base level describing the product.

5.1. Process of product concretisation

Especially the early stages of the development process (project preparation, task clarification and system design, cf. Fig. 4) are supported through the use of the information model. By progressively building up the model, information from different participants of the product development are fetched and integrated. In this way, a full set of product requirements is generated, forming the basis for an efficient development and realisation of the product in the value-added network. In the following, the process of product concretisation will be introduced with an example. The steps of the approach are shown in Fig. 7.

Fig. 7. Process of product concretisation integrating different partial models on the meta level and knowledge base level.

The process starts with the elicitation of an initial set of customer requirements in step 1. Therefore, from the meta level abstract requirements are used as checklists to describe the general purpose and usage site of the product, e.g. the required lifetime of the product. From former projects, established requirements could serve as reference.

Based on this initial requirements, in step 2, a predefined system template for the current project can be chosen from the meta level. These templates describe product concepts and consist of configurations of abstract elements for requirements and system elements combined with supplier information. For example, one product family could be represented by one template. The template selected here must meet the requirement defined in step 1.

In step 3, the selected template gives guidance to concretise the requirements of the product. Therefore, the abstract system elements are replaced by established elements from the product portfolio or new elements. The specified templates serve as a filter to identify matching system elements.

After the system with its elements is defined, in the 5th step, the suppliers for the different elements are selected. The selec-
tion is based on the supplier pool in the knowledge base. By considering the results of the steps 2 to 4, a subset of possible suppliers can be generated for each system element. From this subset a project specific supplier can be chosen.

Within this process, the meta level supports the designer as a guide to clarify relevant issues. Furthermore, it structures information filtering elements of the knowledge base for the usage in current projects. The cross-company connection of information ensures the overall view on the aspects relevant for far-reaching decisions. After the completion of a project, the project specific information located on the project level (cf. Fig. 6) is transferred to the knowledge base.

In the long term, the companies benefit from a documentation of project and product information through the hole development process, which help to overcome the barriers between interrelated knowledge (cf. Fig 2).

5.2. Implementation of the information model

Main advantages of the methodical approach are brought to bear especially in the first phases of the process. In the project preparation, information from beyond the barriers of knowledge (causing the islands of knowledge) improve the estimation of costs and durations in the tender. Estimations could be made more accurate and efficient. In the following phase, the requirements could be gathered based on checklists, combining experience of different companies. The system design could be created, based on existing products. Thereby, the models are carried from phase to phase and continuously concretised.

The presented methodical approach could be applied based on a software tool in development processes. This tool supports the procedure of producing, acquiring, transferring and understanding of information. Depending on the role of the user, specific functions and interfaces are available.

For the cross-company use, the tool has to be used by each partner in the development network as system developer or supplier. Like in a social-media network, each partner creates profiles, assigning to competences and specifying restrictions. The system developer brings together the information of its specific network of supplier. Suppliers could appear in various networks of different system developers.

6. Conclusion

The introduction of the methodical approach to companies collaborating in value-added networks improves the product development process in critical early phases, in which important cost defining decision have to be made, but the information is incomplete. Especially using the proposed information model, integrating partial models of requirements, systems and suppliers, a step towards a better information exchange could be made.

The advantages for the system developer addresses two levels: Firstly, the efficiency of processes is raised, through methods, guiding the designer through decision points, supported by templates and checklists. Secondly, in the long term use, storing of knowledge is supported, which secures the success of the company.

On the other hand, employees have to be trained for the use of the tools. Furthermore, in the introduction stage, a considerable effort is required, to set a starting basis for the knowledge base. Only if it contains various abstract elements as starting point for development, it could be used efficiently.

For the suppliers, the use of the tools leads to effort for create and maintain their profiles, including information about competences and product and project-specific restriction in the tool, to be listed in the suppliers pool of the system developer. Thereby, firstly they apply for orders. Secondly, they ensure an accurate preparation of information relevant for them, before they themselves are involved in a project.

Currently, the described functions of the methodical approach are implemented to a software prototype, which is used in case studies to validate the expected advantages in development projects. Another important aspect is to examine the general acceptance of the introduction of a cross-company used tool. Often, initial reservations concerning the know-how protection and growing dependencies exist, that have to be overcome. However, exactly this is the aim of the methodical approach: to get closer together to improve cooperation and strengthen the whole collaborating network.

Acknowledgements

The results presented in this paper were achieved within the research and development project Synergetische Produktentwicklung mechatronischer Produkte in Wertschaffungsnetzwerken (SynProd), funded by the German Federal Ministry of Education and Research (BMBF) within the Framework Concept Research for Tomorrows Production and managed by the Project Management Agency Karlsruhe (PTKA). The author is responsible for the contents of this publication.

References