A Knowledge-Based Approach for PLM Implementation Using Modular Benefits Dependency Networks

Bas Koomen[1,2](✉)

[1] University of Twente, Enschede, The Netherlands
s.p.koomen@utwente.nl
[2] Cadmes B.V., s-Hertogenbosch, The Netherlands
bas@cadmes.com

Abstract. Industrial companies face significant challenges when they engage in the implementation of Product Lifecycle Management. Research has shown that organizations have difficulties in defining concrete and measurable goals and relating enabling technology to business benefits. Moreover, implementation service providers rely heavily on tacit knowledge when it comes to operational details. This paper proposes a conceptual framework as a methodology for implementation teams. It allows teams to reuse implementation knowledge on a detailed level, related to contribution to benefits and business goals. The methodology is derived from emerging, set-based product and process development methodologies and also from benefit management strategies for information systems. The goal of this methodology is to increase the probability that Product Lifecycle Management implementation contributes to the business benefits of organizations and therefore lower the economic risks. The paper describes the method and the result of two explorative case studies.

Keywords: Product Lifecycle Management · IT benefits management · Knowledge reuse

1 Introduction

Product Lifecycle Management (PLM) is a business practice to manage all product-related information and communication during the product's entire life, from concept to recycling [1–3]. PLM has become more important over the past years with the increasing complexity of products. Producing companies need to be more flexible in terms of product needs, volume, resource efficiency, adapt to (personalized) customer needs, and work in a networked supply chain [4].

PLM, as a business approach, is supported by PLM software. PLM software includes many applications, like Computer-Aided Design (CAD), Computer-Aided Manufacturing (CAM), and Product Data Management (PDM).

The implementation of PLM software is a challenge for many companies and many PLM implementation projects fail to achieve their goals [1, 5, 6]. In a review of published literature about PLM implementation [7], we found that structured guidelines for implementation projects exist. They mostly come down to four phases: 1) prepare

Published by Springer Nature Switzerland AG 2020
F. Nyffenegger et al. (Eds.): PLM 2020, IFIP AICT 594, pp. 553–562, 2020.
https://doi.org/10.1007/978-3-030-62807-9_44

the organization and define strategy and goals, 2) Analyze and measure the current state of the organization and processes, 3) Design a future state for the organization, processes, and ICT infrastructure, and 4) Implement and maintain the future state. It is a substantial investment for companies to perform all the tasks in each phase, and it requires process and management skills to perform them with sufficient quality. Large companies may have these skills in their organization, but Small and Medium-Sized Enterprises (SME) regularly don't.

In an empirical study on 9 PLM implementation projects in SME [8], we found that these companies are "cutting corners" in the hope that a technical software implementation will fix their business problems. Implementation service providers are likely to respond by delivering functionality that the company is requesting, with little evidence that the investment contributes to improved business processes.

Research on Enterprise Systems (ES) and Information Systems (IS)[1] shows that only the software itself does not bring value to an organization [10]. It is the way the organization can change its processes and culture that has the most influence on the value of an ES/IS investment. Therefore, they need to see the relation between enabling function, operational benefits, and business value [11]. Section 2.1 elaborates on the theory of IT Benefits Management.

Another valuable insight from our research is that the PLM implementation, as an organizational subject, is a design process. This insight is supported by the idea that an organization is an artifact, and therefore designed [12]. Furthermore, van Aken [13] has made the connection between the design of physical objects (products, buildings) and organizations. He also noted that in the design of physical objects, the designer, the maker, and the user can be separated. In organizations, this separation mostly absent.

Following these ideas, we looked into several product design methods. We found that Lean Product and Process Development (LPPD) [14] addresses issues that we recognized in our research on PLM implementation. This method emphasizes the contribution of functional and technical decisions to real customer value. Decisions should be delayed until evidence on value contribution is present. The concept of Set-Based Concurrent Engineering, as part of LPPD, enables the decision making process and efficient reuse of knowledge from previous projects. Section 2.2 contains more detailed background information on LPPD.

In this paper, we formulate a method that combines the principles of both IT Benefits Management and Lean Product and Process Development (Sect. 3). We explored elements of this new method in two case studies within a PLM implementation service provider (Sect. 4). The case studies showed promising results, and we gained information to improve the method. We believe that this method can help implementation service providers to improve their service and deliver better quality across all four phases of a PLM implementation. Moreover, it is likely that it lowers the implementation cost, particularly in the context of SME.

[1] Enterprise Systems and Information Systems are a generic descriptions of "means by which people and organizations, utilizing technology, gather, process, store, use and disseminate information." PLM can be classified as an Enterprise Systems or Information System [9].

2 Theoretical Background

This section describes the principal elements from Benefits Management and Lean Product and Process Development, upon which we based our implementation method.

2.1 Benefits Management

Benefits Management, as described by Ward and Daniel [9], is management in a way that benefits from the use of Information Systems (IS) are realized. "Benefits" are referring to business benefits: an advantage for a specific stakeholder or group of stakeholders that want to obtain value from an investment.

On the one hand, benefits are realized by organizational business changes, enabled by enabling changes and IS/IT enablers. On the other hand, benefits enable the achievement of investment objectives, derived from the company's business drivers. These interdependencies build a benefits dependency network (BDN). Figure 1 illustrates the interdependencies with arrows, pointing towards the business benefits. Each arrow represents an n-to-n relation. For example, multiple business changes can support a single benefit, and one single business change can support multiple benefits.

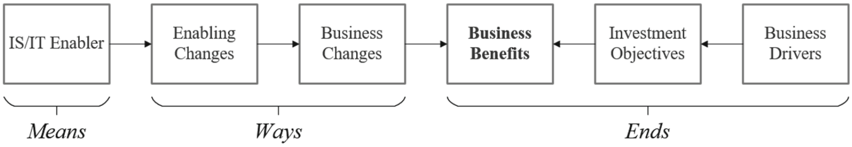

Fig. 1. Interdependencies within a BDN. Six categories in three areas of interest. Modified from [9]

Benefits Dependency Networks are used in various ways. Peppard, Ward, and Daniel [11] have investigated the use of BDNs in Customer Relationship Management (CRM) implementation projects. They found three starting areas of interest (Means, ways, or ends) in the BDN applicable in individual projects. In each scenario, the level of uncertainty increases with every step.

1. **Problem-based interventions**. In these cases, the project starts with a business driver to eliminate an existing problem or constraint. The event flow in this scenario:
   a. Define performance targets for the improved state (investment objectives and potential business benefits).
   b. Identify enablers and changes needed to achieve potential benefits.
   c. Decide which enablers and changes to implement, based on contribution to intended benefits.
2. **Innovation-based interventions, delivered from new ways of working.** Here, the focus is on the business and enabling changes. The flow of events:
   a. The start is a global vision about the future of the company.

   b. Describe new ways of working that would deliver benefits in the context of the vision.
   c. Determine how IT can enable these changes
   d. Evaluate the feasibility and the contribution to the benefits and the vision.
3. **IT innovation-based interventions.** These interventions start with the capabilities of IT enablers. The typical flow in this scenario:
   a. Understand the new capabilities of IT.
   b. Determine what changes to the ways of working are needed.
   c. Assess the benefits and the potential investment objectives that could be realized by the changes.

It is helpful to identify in which scenario a company is going to invest in IS/IT. We suspect that in many cases, a company is making investments based on scenario 3, but the project is managed as a scenario 1. In case of such a misinterpretation, evaluations are made on wrong arguments, and there is no evidence of the value contribution of IT enablers. A skilled application of a BDN can prevent these mistakes and let stakeholders focus on the right subjects. The research of Peppard et al. [11] have shown successes with this approach, delivering more proven benefits of IT investments.

2.2 Lean Product and Process Development

While investigating the design methodology at Toyota in the '90s, Ward, Liker, Cristiano, and Sobek II [15, 16] discovered set-based concurrent engineering (SBCE) as a key differentiator. As a design methodology, it was able to deliver more customer value in less time and effort. Rapid learning and reuse of existing knowledge are key elements in SBCE.

The term Lean Product and Process Development (LPPD) was introduced later by Ward and Sobek II [14]. It puts SBCE in the context of a specific design management concept. Based on their work, other researchers have refined some aspects of LPPD. For our method, we used a number of these elements:

Visual Management. In LPPD, it is essential to present information visually, in order to have all stakeholders maximally involved. Cloft and Kennedy [17] describe a method of causal mapping. These causal maps contain all relevant target parameters, input parameters, and internal parameters of a design problem.

Interdependencies of these parameters are presented as nodes, linking the parameters. The nature of these interdependencies is either known or unknown. In the case of unknown, there is a knowledge gap to be filled.

This visual approach helps to position reusable knowledge in context or narrow down on areas where research is needed. The research team can distribute tasks efficiently and publish the outcome in the causal map.

Set-Based Concurrent Engineering. SBCE includes the following practices, characteristic to its nature [18]:

- Requirements are not defined as exact point values but in ranges or sets. As the design progresses, these ranges can be refined based on knowledge acquired during the design process.

- Design limits and trade-offs between design parameters need to be explicit before design decisions are made.
- Sets of design alternatives are developed. The number of alternatives in the set gradually decreases as the design knowledge increases.
- (Sub-)system design and manufacturing development are done in parallel, with information shared between them. This helps to make better design decisions.

By developing multiple alternative solutions for sub-systems, the chance of success of the entire design project increases dramatically. Besides that, it generates a wealth of reusable knowledge for future design projects.

Structured Workflow. Kerga, Rossi, Taisch, and Terzi [19] have described a workflow to apply SBCE. This workflow consists of four steps:

1. Explore alternative solution sets for sub-systems, looking only at the sub-system itself.). Generate solution sets systematically by using existing knowledge (trade-off curves).
2. Eliminate incompatible solutions in the overlapping regions of the sets. Sets overlap when the sub-systems have design parameters in common. If the parameter value of one set does not exist in the other set, the solution is incompatible.
3. Narrow down the number of solutions by testing. Testing generates limit curves in the trade-off curve diagrams. Eliminate solutions that are outside the limits.
4. Select the final solution design from the remaining alternatives, based on optimal cost vs. performance.

A3 Methods to Reuse Knowledge. To solve problems, capture the acquired knowledge collaboratively, and reuse it visibly, Toyota uses the A3 method [20]. Nowadays, many companies have adopted various flavors of A3 problem-solving techniques (A3-thinking) as part of Lean Initiatives [21]. A3 methods guide teams through a structured process of reasoning to come up with solutions in a systematic way. The essential information of the problem-solving process is written down, supported by graphical information. Due to the standardized and structured approach, others who use the same structured approach can retrieve the information quickly. For our PLM implementation method, we focused on the following characteristics of A3 methods:

- A3s present information in a very compact way, lowering the barrier for users to use and maintain it.
- A3s are structured in a predefined way. It guides the thinking process of the readers and guides writers to put all essential details down in the same way and be complete in their input.

3 New Implementation Method

We derived our method from the two fields described in Sect. 2. They merge into a new method for repetitive PLM implementation in multiple companies. The method allows implementation service providers to link enabling technology to business improvements.

In this process, the method captures PLM implementation knowledge on a detailed level and in a modular way. Knowledge objects are reusable in future implementations in unique combinations that fit the specific customer's needs.

With this knowledge, an implementation service provider will be able to define and execute implementation projects that meet the real customer's requirements with higher quality and with less time and effort.

3.1 Visual Management

Where LPPD uses a causal map to visualize the interdependencies of the design parameters, we use a modified form of the BDN. We selected BDN, for its predefined context of IT implementation Initially, a BDN is made for a specific situation and describes a transformation from a current state to a future state of an organization. In our method, we want to reuse elements of the BDN. Therefore, our BDN has to present a possible future state instead of a change. The replacement for change is Use-Case in our method. Also, Operational Benefits should be described in absolute instead of relative terms.

We simplified the model based on the case study (Sect. 4.1) and merged the replacement for enabling change and business change into one category. Furthermore, we added more detail in the enabling technology by narrowing down on functions instead. These modifications resulted in the structure shown in Fig. 2. As with the BDN, described in Sect. 2.1, the arrows can describe n-to-n dependencies.

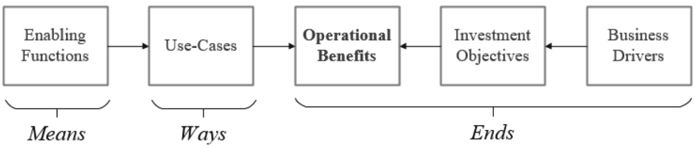

Fig. 2. Modified BDN as visual mapping for PLM implementations.

3.2 Set-Based Design

In our method, we subdivide the entire PLM implementation into PLM reference processes. PLM reference processes are common processes within the broader definition of PLM [22]. Examples are change management, project management, and product management. Potential benefits, related to a reference process, can be realized by multiple alternative Use-Cases: a set of solutions. How and how much each Use-Cases contributes to each benefit is information that is related to the dependency. In some cases, a Use-Case could be detrimental to a benefit, for example, when users have more administrative workload. This Use-Case then has a negative dependency on the benefit of "low workload". Benefits and Use-Cases can exist in multiple reference processes, creating an overlap between sets.

3.3 Workflow of the Method

In our method, we follow a similar workflow, as described in Sect. 2.2:

1. Find feasible alternative Use-Cases for Operational Benefits within a reference process.
2. Eliminate incompatible alternatives by evaluation of contribution to the benefits of overlapping sets.
3. Perform tests on remaining alternatives, based on the specific implementation requirement ranges. Measure the performance (contribution to Operational Benefits).
4. Select an optimal solution, based on cost vs. the estimated contribution to the achievement of the desired benefits.

3.4 Reuse Knowledge

In our method, we capture knowledge in predefined forms for each object (e.g., Enabling Function, Use-Case). The form contains a clear structure to guide the user. The structure varies per object class. For example, the Operational Benefit form currently contains a section for a description of the benefit (what is the advantage?), potential performance indicators (how can the benefit be measured?), and hyperlinks to related use-cases and investment goals.

The content should not exceed the amount of information that would fit on an A3. The form also includes references to other objects to represent potential dependencies. Physical A3 could be used for this purpose, but a digital method allows to hyperlink between forms. In our case study, we used an online wiki structure.

4 Explorative Case Studies

During the development of our method, we ran two explorative case studies to get feedback on elements of the method.

4.1 Explorative Case Study 1 – Modular Benefit Dependency Network

Our learning goal for this study was to observe if and how practitioners could apply our method in the early stage of development. We invited a technical and commercial consultant of a PLM implementation service provider to participate in this study.

First, we asked them to identify an industry segment within their SME customer base. Important was that they knew both operational and business contexts in this segment. Then, we trained them to use our modified BDN and let them create a potential Business Drivers, Investment Objectives, and other BDN obects that apply to this industry segment. The commercial consultant focused on the “ends” and the technical consultant on the “means” and “ways”. They worked on the Benefits collaboratively. The knowledge was written down in wiki pages, using the wiki app in Microsoft Teams.

We gained several insights from this study. Both consultants were able to apply the method without much difficulty after some coaching in the first meetings. The commercial consultant claimed that this method helped him to structure a consultative conversation with customers. Instead of argumentation of the value of software products, he believes he can discuss value for the organization. Moreover, he believes he is better able to position more relevant solutions to help the customer to realize improvements. The technical consultant was also positive about the visibility of how technical enablers should be used to realize certain benefits. However, he suggests that more detail is needed in the enabling functions to make it usable as a detailed specification. With the current detail level, the deployment of a specified implementation still relies too much on the knowledge of the consultant in the project.

4.2 Explorative Case Study 2 – Alternative Enablers

In this study, we looked more deeply into the enabling functions. We participated in the early stage of a PLM implementation with an SME company that (among other subjects) was asking for a solution to manage design changes (reference process).

First, we engaged with the customer to ask for the desired Operational Benefits that they expect from change management and how they could measure them. The customer came up with eight potential benefits and realization criteria. Example: One Operational Benefit was "traceability of change" with the criterion "all changes must be traceable from origin to execution".

After that, we made a list of potential Use-Cases to execute design changes. We found potential nine Use-Cases within the software platform of the customer, requiring various apps as Enabling Function.

Finally, we linked the Use-Cases to all Operational Benefits and rated the individual dependencies by estimated contribution. The rating is a five-point scale: strong negative, negative, neutral, positive, and strong positive. Figure 3 shows a simplified version of this rating.

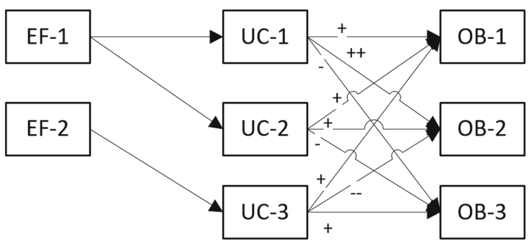

Fig. 3. Illustration of dependency rating for Use-Cases.

After the evaluation of the dependencies, three potential Use-Case remained feasible. They have to be evaluated with the stakeholders for final selection (yet to be performed).

We learned from this study that the method helps to make an objective selection of potential solutions to a problem. The customer shifted the attention from a request for functionality to a formulation of business outcomes. We expect to see more support from users when they are involved in the evaluation of the remaining Use-Cases.

5 Conclusion

In this paper, we described how we succeeded in forming a method for PLM implementation, based on LPPD and Benefits Management. The method can be used by PLM implementation service providers to find a match between enabling technology and their customers' business value. The explorative case studies gave encouraging feedback, growing confidence that the method improves the quality of PLM implementation. SME customers will benefit the most because, potentially, the reusable knowledge will replace costly business consulting.

We plan to make improvements to our method. Based on case study 1, we learned that we need to review the knowledge forms and improve them for clarity. We also need to write better instructions for the proper use of the method. Furthermore, the dependencies currently are only hyperlinks in a wiki system. The wiki system does not support meta-data on these links. In case study 2, we rated the dependencies manually in a spreadsheet. Another planned improvement is the ability to give priorities to Operational Benefits and make use of potential maturity levels to help organizations to formulate their ambition for individual reference processes [22].

To draw rigorous scientific conclusions about the effectiveness of the method, we have to perform a more extensive and a more formalized case study. This is planned for the next phase of this research after the refinement and formalization of the method.

References

1. Stark, J.: Product Lifecycle Management. Volume 1, 21st Century Paradigm for Product Realisation. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-17440-2_1
2. CIMdata: About PLM (2017). https://www.cimdata.com/en/resources/about-plm. Accessed 2019 Jan 26
3. Gartner: IT Glossary - Product Lifecycle Management (2018). https://www.gartner.com/it-glossary/product-lifecycle-management-plm.. Accessed 2019 Jan 26
4. Smart_Industry: Dutch Smart Industry Report. Smart Industry Netherlands: Zoetermeer NL (2019)
5. Wognum, P.M., Kerssens-van Drongelen, I.C.: Process and impact of product data management implementation. Int. J. Product Dev. **2**(1), 5–23 (2005)
6. Siddiqui, Q.A., Burns, N.D., Backhouse, C.J.: Implementing product data management the first time. Int. J. Comput. Integr. Manuf. **17**(6), 520–533 (2004)
7. Koomen, B.: PLM in SME, what are we missing? An alternative view on PLM implementation for SME. In: Chiabert, P., Bouras, A., Noël, F., Ríos, J. (eds.) PLM 2018. IAICT, vol. 540, pp. 681–691. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01614-2_62

8. Koomen, B., Hoogeboom, M., Schellens, V.: PLM implementation success rate in SME. an empirical study of implementation projects, preliminary findings. In: Fortin, C., Rivest, L., Bernard, A., Bouras, A. (eds.) PLM 2019. IAICT, vol. 565, pp. 47–57. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-42250-9_5
9. Ward, J., Daniel, E.: Benefits Management: Delivering Value from IS & IT Investments. Wiley, Hoboken (2005)
10. Carr, N.G.: IT doesn't matter. Harvard Bus. Rev. **81**(5), 41–49 (2003)
11. Peppard, J., Ward, J., Daniel, E.: Managing the realization of business benefits from IT investments MIS Q. **6**(1), 3 (2007)
12. Simon, H.A.: The Sciences of the Artificial. MIT, Cambridge (1969)
13. Aken, J.E.V.: Strategievorming en Organisatiestructurering : Organisatiekunde vanuit Ontwerpperspectief [Strategy Building and Organizational Structuring; Organization Science from a Design Perspective.], 2nd edn. Kluwer, Deventer (2002)
14. Ward, A.C., Sobek II, D.K.: Lean Product and Process Development, 2nd edn. The Lean Enterprise Institute, Cambridge (2014)
15. Ward, A., et al.: The second Toyota paradox: how delaying decisions can make better cars faster. Sloan Manag. Rev. **36**(3), 43–61 (1995)
16. Sobek II, D.K., Ward, A.C., Liker, J.K.: Toyota's Principles of Set-Based Concurrent Engineering in Sloan Management Review, pp. 67–84 (1999)
17. Cloft, P.W., Kennedy, M.N., Kennedy, B.M.: Success is Assured: Satisfy Your Customers On Time and On Budget by Optimizing Decisions Collaboratively Using Reusable Visual Models. Productivity Press, Cambridge (2018)
18. Kennedy, B.M., Sobek II, D.K., Kennedy, M.N.: Reducing rework by applying set-based practices early in the systems engineering. Process Syst. Eng. **17**(3), 278–296 (2014)
19. Kerga, E., et al.: A serious game for introducing set-based concurrent engineering in industrial practices. Concurrent Eng. **22**(4), 333–346 (2014)
20. Shook, J.: Managing to Learn: Using the A3 Management Process to Solve Problems, Gain Agreement, Mentor and Lead. Lean Enterprise Institute (2008)
21. Sobek II, D.K., Smalley, A.: Understanding A3 Thinking: A Critical Component of Toyota's PDCA Management System. Productivity Press, New York (2008)
22. Informatik, V.: Leitfaden zur Erstellung eines unternehmensspezifischen PLM-Konzeptes: Product-lifecycle-Management; transparente Prozesse und konsistente Informationen im Produktlebenszyklus [Guideline for creation of a company specific PLM concept: Product Lifecycle Management; Transparent Processes and Consistent Information in the Product Lifecycle]. Informatik. VDMA-Verl, Frankfurt (2008)