NordDesign 2008
August 21 – 23, 2008
Tallinn, Estonia

Towards Integrated Modelling of Product Lifecycle Management Information and Processes

Dag Bergsjö
Chalmers University of Technology
Göteborg, Sweden
dagb@chalmers.se

Amer Catic
Chalmers University of Technology
Göteborg, Sweden
amer.catic@chalmers.se

Johan Malmqvist
Chalmers University of Technology
Göteborg, Sweden
joma@chalmers.se

Abstract

This paper makes a review of the current standards and practices regarding, PLM architecture and modelling languages for information and system modelling. As well as it makes a short review of the trends in automotive and aerospace PLM applications in industry.

It is concluded that enterprise and PLM modelling can be and is being performed in industry at different abstraction levels. A more uniform and integrated approach towards modelling of both IT and processes is believed to be beneficial from several viewpoints, not least in order to better govern and manage the IT and process integration and architecture over time, in an ever changing business context.

Keywords: Product Lifecycle Management, PLM, Enterprise Architecture, Modelling.

1 Introduction

A far reached goal of any organization must be the optimization of the alignment of their business drivers (e.g. customer & partner needs, business strategy, organization culture, business, people, processes and technology). This optimization, not only provides for efficient and cost effective performance, but also helps execution of the organizational goals and objectives [1]. As a basis for such an assessment good modelling capabilities of the enterprise architecture (EA) and the description of the business model is essential. This model would constitute a map of the PLM landscape that would make the streamlining of people, processes and products [2] in product development easier, something that is a prerequisite for the implementation of e.g. a service oriented PLM architecture [3].

An overall definition of EA can be that EA is the blueprint of the company [4]. Every company is within this model organised into different enterprise activities that can be broken down further into processes and activities. The enterprise itself can then be seen (and modelled) according to these business activities [4]. From an information systems, or information management perspective the goal with EA is to integrate the business and IT resources in order to improve the enterprises competitiveness [5]. In order to define a modelling language for EA some generic requirements have been stated on the solution: Multiple consistent levels, Adequate representation of actions, Multiple systems of interest, and Existence of traceability relationships [5]. Regarding these points it is important that the

model allows for concurrent editing and that the model is allowed to grow and change over time.

In general it can be stated that there exist two types, or two approaches towards enterprise architecture, namely the IT viewpoint consisting of IT modelling tools and the process viewpoint consisting of process or EA tools [6]. The overall problem is that the domain specific tools lack in functionality regarding level of detail for process and business modelling as well as most UML based tools have problems with representing the hierarchical nature of businesses and processes.

Product lifecycle Management (PLM) refers to the management of the product definitions, e.g. product data within foremost product development but in the essence of the definition product data of the whole lifecycle of the product . Integrating the development over several departments, totalling thousands of employees, calls for powerful IT tools and systems, where information can be managed for instant access.

This paper will aim at streamlining PLM (The company's processes and IT systems) with the work in EA and modelling, in order to evaluate the possibility to successfully model PLM architectures within both a business process viewpoint as well as the IT viewpoint. This has not previously been done in the context of PLM. The following research question is stated:

- How can PLM architectures, consisting of information and processes, be modelled using principles found in theory and practice for enterprise architecture?

2 Method

This approach is based on semi structured and unstructured (unplanned conversation) interviews with PLM architects and experts, as well as an overlooking theoretical study involving the areas of PLM and EA. The industries of focus have been the automotive and aerospace industries mainly in Sweden.

The approach is verified with a case study performed in a large automotive manufacturer in Sweden. Functions with a high degree of software and electronic content have been modelled including business processes and information entities, such as documents and requirements, simulation and analysis software and information repositories.

3 Modelling of complex processes and PLM (Theoretical approach)

When working with product development in a highly digitalised environment, plans for how to structure and represent information must be made. The need of information and how that information is transformed during a product development project must be dealt with, as well as connections to the development process and the product during its entire lifecycle. This subject has been investigated in the literature and in several independent standards. The product dimension is however only one part of the equation. Process and Organisational aspects [2] have to be considered and modelled as well.

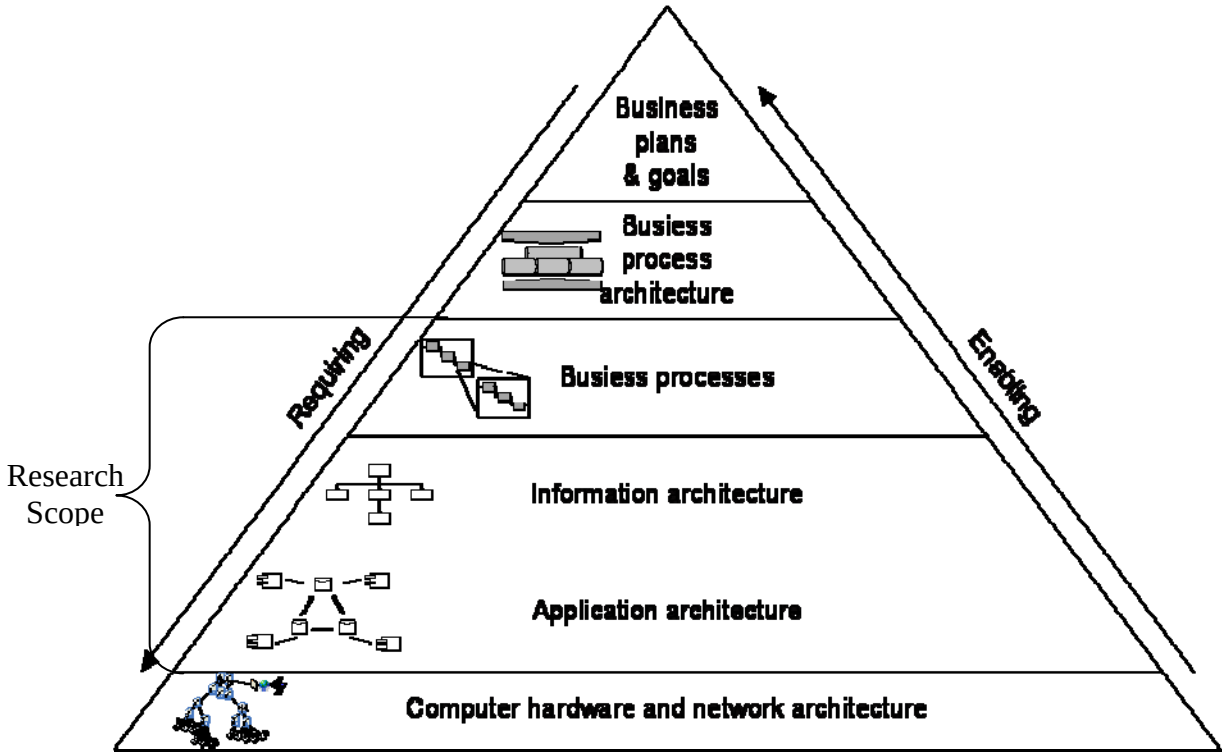

Figure 1. *Enterprise architecture [7]*

The concept of enterprise architecture addresses the need to keep a consistency throughout the different ways of representing an enterprise, the different enterprise architectures. The concept was set forward by Zachman as a response to the efforts to structure different parts of the enterprise at different levels without considering and making sure that the structuring effort was in harmony with the enterprise as a whole. The consequence being that an incredible amount of redundancies are introduced due to the fact that different parts of the enterprise have different architectures. The differences do not only exist between different parts of the enterprise but they also differ at different levels within the part of the enterprise. When we talk about differences we mean that the different ways of structuring the organization, processes, information, applications and infrastructure in different parts are not in harmony which leads to inconsistencies and redundancies which increase costs. An example stated by Zachman from the field of banking states that customer data was stored redundantly in 129 files and that statistically an average data fact is stored 10,8 times redundantly [Zachman, 1996].

The phenomenon described by Zachman in general has been observed in the PLM area by e.g. Svensson, 2003. Different engineering departments have been optimizing their own set of IT-tools without adherence to a larger architecture. This led to a sub-optimization of the IT-landscape with as well large overlaps as inconsistencies and “holes” in the IT-landscape. Both of these leading to large inefficiencies in the engineering processes. An example in Figure 2 could be that department A is using one set of CAD and PDM tools to create and manage their data while department B is using another set of CAD and PDM tools. The reason for the different tools might be that they have differing functionalities which provide excellent support for some activities performed at the different departments. The overlapping area means that a subset of the data created and managed is identical. Besides the risk that the data may not be identical leading to two different representations of the same product component there is a cost issue directly related to the redundancy.

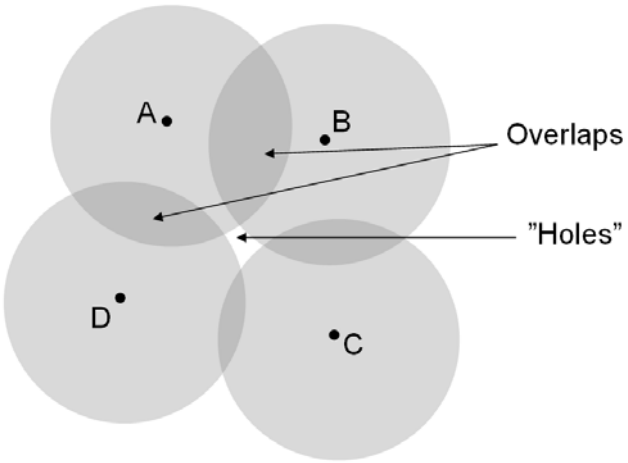

Figure 2. *Overlaps and missing support regarding different departments*

In order to address the issue of non-consistency and non-harmony in the enterprise, Zachman developed a framework that can be applied in order to structure the different architectures within the enterprise [8]. By applying this framework the enterprise architectures are described and inconsistencies can be identified. The framework is general in its terminology and is described briefly in the following passage.

On the vertical axis of a matrix there are the different levels of detail relevant to different roles in the enterprise. There are five levels:

- *Scope*
  The context is set at this level where it is defined what the enterprise does in general terms. This is the strategic level where business strategies and business objectives are defined in order for the general business vision and idea of the enterprise to be fulfilled.
- *Business model*
  At this level the concepts for how the business is built up are defined. Examples include business plans, business processes, logistics and general relations between organization units.
- *System model*
  At this level the logic for the different architectures is set e.g. application architecture, the data model, people and roles in the organization and so on.
- *Technology model*
  In the technology model the technological solutions which need to be implemented to realize the system models. Since this is still at a technological level it means that only the technologies to be used, and not the detailed solutions, are defined. An example is that a parametric CAD technology is to be used for solid modelling without stating which commercial software to use.
- *Detailed representations*
  This is the level at which the detailed issues are resolved. This level implements the technologies by applying solutions, commercial or internally developed.

On the horizontal axis there are different aspects which relate to the different process- and IT-related aspects of the enterprise:

- *Data*
  The data aspects relates to the information and how it is structured in information models at more generic levels and how it is structured in the databases in the more detailed implementation levels. The data is what is delivered from the enterprise processes.
- *Function*
  The function aspect describes the process in which the data is used from a high level process towards the detailed application and program processes performed in computer applications
- *Network*
  The network aspect is going from the geographic level towards the resource level and answers the questions where are the process performed.
- *People*
  The people aspect resolves who is doing what from complete organizations via roles to final identities of individuals
- *Time*
  The time aspect states the times frames in which the enterprise processes operate. From general time plans for e.g. product releases towards execution times for individual programs
- *Motivation*
  This aspect answers the question why the processes are performed in a particular manner and spans from general business objectives of the enterprise down to business rules controlling execution of programs.

Since similar reasoning, which led to the framework and idea of enterprise architecture as a means for better management and cost efficiency, can be found in the area of PLM by e.g. Svensson the ideas of enterprise architecture will be applied and benchmarked to the area of PLM. The paper focuses on different modelling techniques as well as modelling tools used in the area of PLM and relates to how these contribute to the documentation and description of the PLM related parts of the enterprise architecture which here will be referred to as the PLM architecture.

Information Modelling for Complex Products

To describe a primarily mechanical product and how it functions has been the theme of Hubka and Eder [9]. Their Theory of Technical Systems shows how a technical system interacts with a human operator to perform a change from an input to an output. The model has been refined by Andreasen in the Theory of Domains [10]. This thinking is further elaborated in the Chromosome Model [11]. The terminology of the chromosome model can be applicable for modelling of mechatronic systems, since there are no difference if functions in the Chromosome Model are realised with hardware or software [12].

Property-Driven Development/Design [13] is a theoretical model evolved from the work of Hubka and Eder and of Andreasen. The focus is modelling of a product in a PDM/PLM system. According to the model, there is a distinction between characteristics and properties, where characteristics describe the structure, shape, and material of a product, while the properties describe the product's behaviour. Properties depend on several aspects including the chosen characteristics and cannot, in contrast to characteristics, be determined by the designer.

The function-behaviour-structure (FBS) framework was described originally by [14]. The FBS model treats design as a process where functions, behaviours, and structures are essential. The steps are linked together by processes that can be seen as the different stages of design. In Figure 16 there are eight processes linking function (F), expected behaviour (Be), behaviour derived from structure (Bs), structure (S) and design description (D) together. The model can be extended to describe even more processes [15]. The model makes it possible to distinguish between different types of information within the design process. Function information regarding "what it is for" can be used. The model also shows when (in which process) that particular information is usable.

Standards for Modelling of IT Systems and Data Exchange

Attempts to standardise information related to PLM system development for management of mainly product information are ongoing. Standardisation effort with connections to this are the ISO 10303 STEP. There exist two main modelling techniques, Unified Modelling Language (UML) and STEP Express. Both of these techniques, in the most basic sense, illustrate items, attributes, and relations.

UML is based on three main models: Functional Model, Object Model, and Dynamic Model. The standard is officially defined by the Object Management Group (OMG). In UML 2.0 there exist 13 types of diagrams for different types of information modelling tasks [16]. The most common one for information modelling is called the class diagram. In related research it has been shown by [17] that UML can be used as a powerful tool to model information in a way that reduces risks of misinterpreting the models. A model based on UML is easily understood even with no UML experience [18], which makes the model suitable for communicating product models in large projects involving different engineering disciplines.

SysML is a development of the UML standard to be less software-focused and better manage systems engineering information. Despite the limitations of UML, it is still possible to model complex relations involving both software and mechanical parts in UML diagrams [18]. The standard UML4SysML was finally adopted by OMG in May 2006 [19]. The standard is applicable to modelling in systems engineering contexts with a few exceptions, for example storage of items [20]. Comparisons of SysML to UML advantages for systems engineering can be identified.

ISO 10303 is known as STEP or the Standard for the Exchange of Product model data. The standard contains a number of Application Protocols (AP). These protocols contain adaptations of the ISO standard to certain businesses (Figure 17). The applicability of STEP is discussed by [21], e.g. the possibilities to perform configuration management on a single product to be accessible over its lifecycle as in AP-239. This would allow better control over produced products and enable easy software updates, much needed in the automotive industry. Express is the language adopted in the ISO standard 10303-11. Express is available both as a text-based and as a graphical specification.

Standards for Process Modelling

Business and process modelling languages are used to describe the sequential order of business activities and resources needed as well as control instruments. Commonly used models include IDEF0.

More detailed process descriptions can be performed in the language BPEL. In order to make a business process executable in a software there has been identified two different types of

business processes according to BPEL. Abstract and executable processes, where the latter has the possibility to be automated by a software. The abstract process requires inputs that are difficult to attain automatically. An extension of BPEL is called WS-BPEL and is focused upon execution of web services standardised by OASIS [22].

4 Modelling of complex processes and PLM (Empirical data collection)

When managing information systems and processes most companies studied use some means of modelling and updating their models and specifications regarding their PLM environment. This chapter deals with the empirical findings from several companies, meetings, interviews and workshops, it is structured according to the research scope presented in Figure 1.

Business processes

Business processes are defined on a high abstraction level and are commonly known throughout the organisation as "the product development process" or similar internal company specific name for the product development process. The over all processes are documented in word documents, leaflets and books (electronic and paper based). The over all use with the process documentation is to work as guidelines in order to assure that the right approvals, deliveries are performed at the gates. The processes are quite similar from company to company and show great resemblance to theoretical models e.g. [23, 24].

On a detailed level however documentation is sparse. Work descriptions connected to the employment certificate constitutes the most detailed description of the work to be performed. An attempt to in more detail map a development process for an automotive Adaptive Cruise Control (AICC) process in a swim line diagram is shown in Figure 3. It is not common to have these detailed maps of how to perform a development process specific for each development task, but they are created in order to test and evaluate specific use cases when introducing a new IT system. Theses maps act as a one time performance i.e. they are used for a specific use case and then they are not maintained or updated, and when a new use case needs to be created the whole process mapping is performed once again. There seems to be lacking tools and processes for keeping the process information correct and updated. It was stated by one of the internal IT department co-worker that if they where suppliers to the product rather than the IT system they would not have been hired, since they deliver solutions that are incomplete, that are not tested, and that lacks documentation.

Information architecture

The information architecture is in some cases well documented in information models, but updated and accurate documentation on a detailed level is difficult to get hold on. The information models have changed over time and are distributed over several IT systems. Information models drawn up using for example UML terminology are used both "as is" information models and "to be" information models. Most companies seem to have an idea about there information models and object structure. There is however doubt if the theoretical models corresponds to the actual implemented models. The addition of meta data and transformation of information from one IT system to an other also changes the information model in a way that is difficult to foresee.

Application architecture

The way applications communicate and the interfaces between them are to varying degrees documented. At the companies they range from lists with "allowed software" to well documented specifications. In some cases it is stated which IT-supplier that is supposed to deliver a specific software, and in other cases which application to use for different engineering tasks. The available documentation and models are often over simplified where an integration between two systems simply are illustrated by a connecting line.

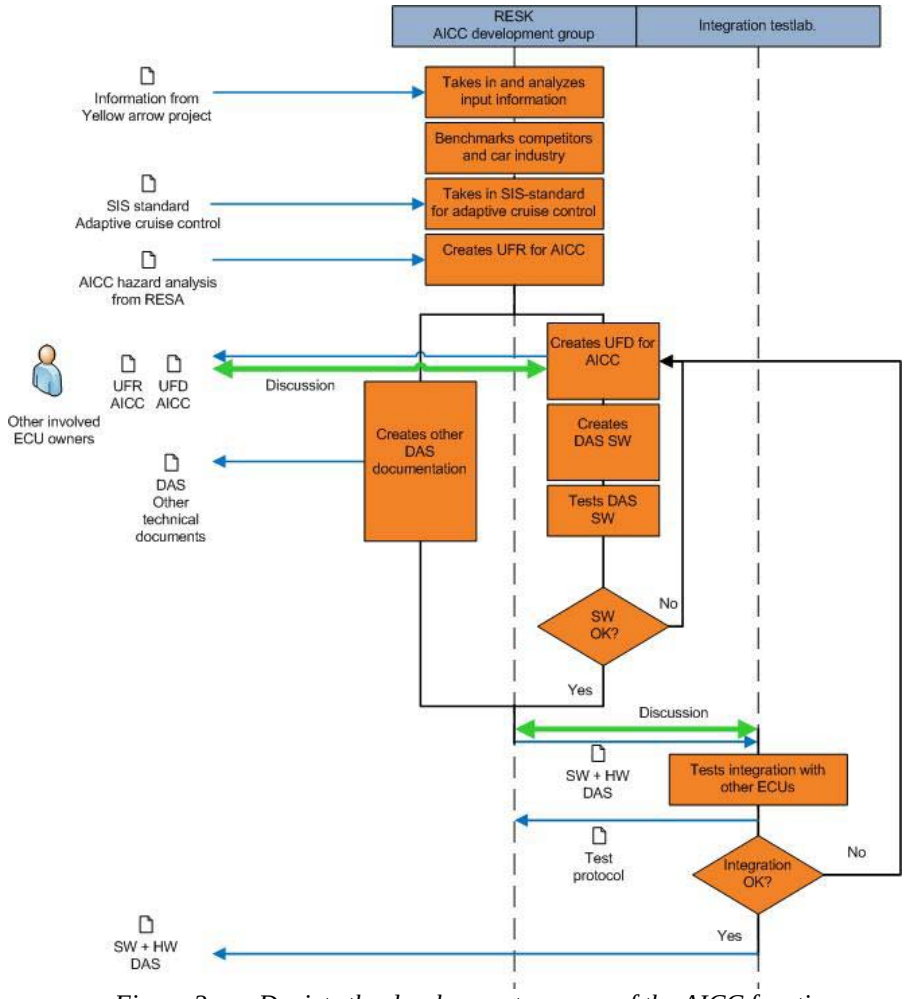

Figure 3. Depicts the development process of the AICC function.

Supplier software API:s are normally well documented and known (even though they may change over time). However company specific or highly customized legacy IT systems are often not documented and documentation that exist is often out of date. In one of the studied companies it was not known to what extent the application architecture was documented, but there existed a core PLM team that had the knowledge of the architecture. There was also a discussion whether the core team withheld this information on purpose or if they did not have the possibility to make the documentation available.

5 Analysis and discussion

Focus on IT and IT-systems

When talking to PLM architects and similar professionals it is clear that they express a large need for process and engineering focus. It is however evident that it is very popular to focus on IT and IT-systems very early in the projects. The IT systems and PLM systems part is the hands on work that generates something to show and test, and hence the modelling and mapping of PLM processes and IT architecture is not that mature. In the companies there is knowledge about existing tools and modelling languages for enterprise architecture and software development but these are in practice not used. Modelling in these cases are restricted to common tools such as Visio and PowerPoint that do not include specialized functions for IT and process modelling.

The documentation of IT systems are often inconsistent an unknown. When asked about the documentation of PLM systems, integration etc. most interviewees seem to think that there exist some models or documentation somewhere but they don't know where it is, or how to get it. It also seems like they don't need it or want it. If this should be compared to the product and the product documentation there is a huge gap in the availability of product related documentation and specifications and the availability of IT related documentation. It was also mentioned by several interviewees that the specific company is not a supplier of IT solutions and that this reflects the lack of documentation and understanding of the need. This can however be challenged in an environment where many engineers spend more time on managing IT, and IT-systems than actually working with engineering tasks [25]. When engineers spend more time on IT than on engineering, is the core business of the company then IT or engineering, and should not a larger effort be placed on managing IT resources?

Process and project support

In general processes are rather well defined (at least at a high abstraction level) but processes show a varying degree of details. It is also not known to what extent that the defined processes actually work or how much the engineers comply with them in their work. The process and project templates are normally easy accessible on the company intranet or on paper documents. It seems that the most commonly used tool for process are standard Office applications. Many of the processes are hard coded into the information systems used such as Teamcenter or SAP. These commercial systems have functions for extracting data models etc. that can be used in order to explain a process or a product part.

There seem to be a missing logical step from information models to implementation, i.e. descriptions that are necessary for the implementation. An information model conveys little information about the implementation and often lacks important aspects e.g. about the underlying database architecture of database tables and queries. This makes it difficult to use an information model as the sole requirement document of a PLM implementation, even though this is often the case.

In order to better support the integrated modelling of process and information architecture a conceptual tool has been developed that shows the possibilities of an integrated modelling approach (Figure 4). The same process as in Figure 3 has been modelled using a similar flow but that incorporates data repositories. The ability to integrate such an internet based model with a Wiki-like interface makes it possible to maintain and update the model over time without incorporating extra resources to architects and IT specialists, since it is the everyday user that uses the Wiki-like interface. This approach is just emerging and future research and assessments will have to confirm the applicability of integrated wiki and EA tools.

Human factors

Human factors of engineering are often mentioned in the interviews as being of importance, along with several other things. However it is evident that many of the people working with PLM introductions and improvement projects have a large filter towards the business processes, as described by etc [26]. It is difficult to map people and roles in connection to PLM and process modelling and to some extent these functions are not supported in the IT systems invested in. A typical industrial case is when it is more convenient to change the business process to fit the IT system already selected which in turn leads to a more complicated solution for the engineer that is supposed to benefit from the new PLM system.

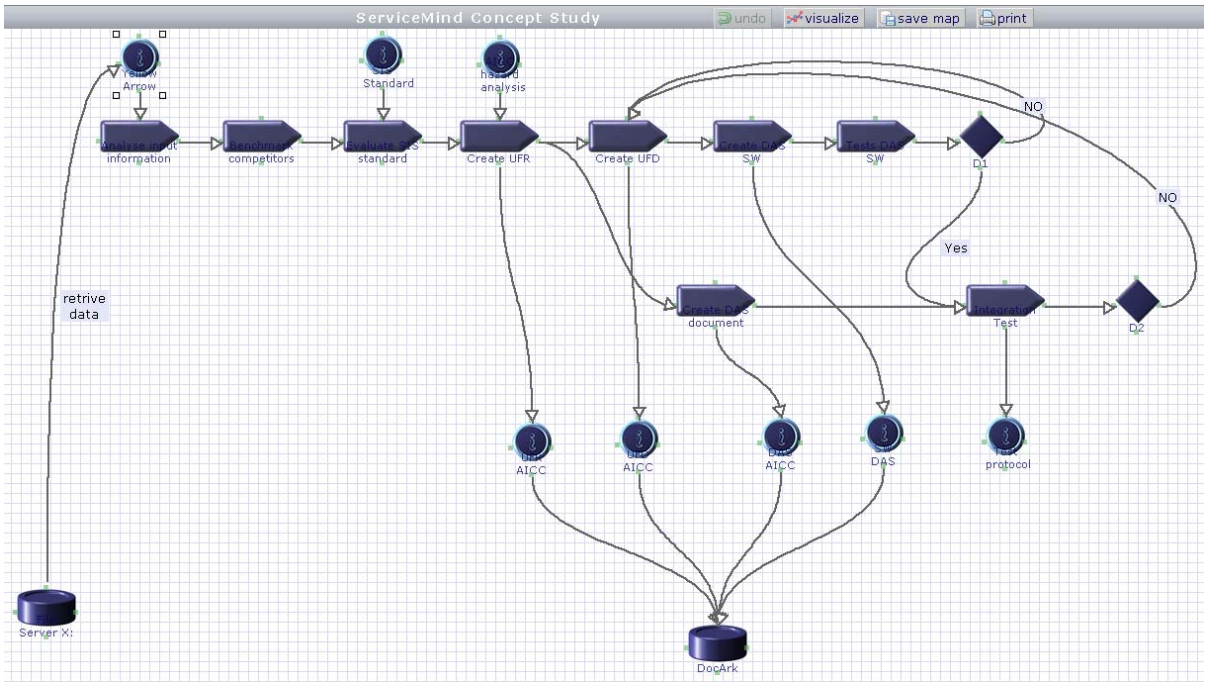

Figure 4. Alternative modelling of process from Figure 3

It is believed by the authors that a structured process for modelling and identifying requirements early on in the PLM-introduction project will benefit the technical solution. Further regarding the organisational impacts of user needs, when comparing the user-need aspect to general product development theories, it is shown that the need is the basis of any product development project [27]. Moreover, [28] states that the user (not necessarily the buyer) should be the main focus of a product development project. Transferring the same thinking to a PLM introduction and adaptation process should put the user and the use of the PLM system in focus. Doing this should make the PLM suppliers create PLM systems that are appreciated by the intended user, and not focus so much on the paying customer (the business perspective). In the end, a happy user will be appreciated by the business and hence be willing to pay for such a solution concept.

6 Conclusions

It is concluded that enterprise and PLM modelling can be and is being performed in industry at different abstraction levels. A more uniform and integrated approach towards modelling of both IT and processes is believed to be beneficial from several viewpoints, not least in order to better govern and manage the IT and process integration and architecture over time, in an ever changing business context.

The major challenges lie in finding modelling approaches that are beneficial for the organisation when implementing and performing changes to the IT and process environment without requiring huge mapping efforts. The needed level of detail and what the appropriate abstractions level should be is believed to be quite individual from company to company. Just as with IT and PLM it is very difficult to say what is appropriate for a unique company since the legacy and traditions vary. However, more structure and good documentation is essential when companies are taking the initiative to improve and make their PLM more efficient.

Acknowledgement

The authors would greatly acknowledge the support from ProViking (SSF) and Vinnova. The researchers also acknowledges the contribution of Jim Larsson who drew Figure 3.

References

1. Schekkerman, J., *How to Survive in the Jungle of Enterprise Architecture*. 2003, Victoria, Canada: Trafford.
2. Eppinger, S.D. and V. Salminen, *Patterns of product development interactions*, in *Proceedings of International Conference on Engineering Design - ICED 2001*. 2001: Glasgow.
3. Bergsjö, D., et al. *Product Lifecycle Management for Cross-X Engineering Design*. in *ICED'07*. 2007. Paris, France.
4. PRESLEY, A., et al., *Engineering the Virtual Enterprise: An Architecture-Driven Modeling Approach.* The International Journal of Flexible Manufacturing Systems, 2001. **13**: p. pp. 145-162.
5. Lê, L.-S. and A. Wegmann, *Definition of an Object-Oriented Modeling Language for Enterprise Architecture*, in *Proceedings of the 38th Hawaii International Conference on System Sciences*. 2005: Hawaii, USA.
6. Lê, L.-S. and A. Wegmann. *SeamCAD: Object-Oriented Modeling Tool for Hierarchical Systems in Enterprise Architecture*. in *Proceedings of the 39th Hawaii International Conference on System Sciences*. 2006. Hawaii, USA.
7. Zimmerman, T., *Implementing PLM Across Organisations - For Multi-Disciplinary and Cross-Functional Product Development*, in *Product and Production Development*. 2008, Chalmers University of Technology: Göteborg.
8. Zachman, J.A., *A Framework for Information Systems Architecture.* IBM Systems Journal, 1987. **25**(3).
9. Hubka, W. and W.E. Eder, *The Theory of Technical Systems*. 1988, Berlin, Germany: Springer-Verlag.
10. Andreasen, M.M. *The Theory of Domains*. in *Proceedings of the EDC-workshopp-Understanding Function and Function to Form Evolution*. 1991. Cambridge, UK.
11. Andreasen, M.M. *Designing on a "Designers Workbench"*. in *Proceedings of the 9th WDK Workshop*. 1992. Rigi, Swizerland.
12. Hallin, K., et al. *Modelling Information for Mechatronic Products*. in *Proceedings of ICED 03*. 2003. Stockholm, Sweden.
13. Weber, C., H. Werner, and T. Deubel, *A different view on product data management/product life-cycle management and its future potentials.* Journal of Engineering Design, 2003. **14**(4): p. 447-64.
14. Gero, J.S., *Design Prototypes: a Knowledge Representation Schema for Design.* AI Magazine, 1990. **11**(4): p. 26-36.
15. Gero, J.S. and U. Kannengiesser, *The Situated Function–Behaviour–Structure Framework.* Design Studies, 2004. **25**: p. 373–391.
16. OMG. 2006 [cited 2007-01-28].
17. Merlo, C., et al., *Compared implementations of PDM systems based on UML specifications.* International Journal of Product Lifecycle Management, 2005. **1**(1): p. 52-69.
18. Johansson, G. and J. Detterfelt. *A UML Based Modeling Approach for Multi Domain System Products*. in *Proceedings of NordPLM'07*. 2006. Göteborg, Sweden.
19. OMG, *OMG SysML Specification*. 2006.
20. Bock, C., *SysML and UML 2 Support for Activity Modeling.* Systems Engineering, 2006. **9**(2): p. 160-186.
21. Pratt, M.J., *ISO 10303, the STEP standard for product data exchange, and its PLM capabilities.* International Journal of Product Lifecycle Management, 2005. **1**(1): p. 86-94.

22. OASIS, *Web Services Business Process Execution Language Version 2.0*. 2007: http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf.
23. Pahl, G. and W. Beitz, *Engineering Desing - A Systematic Approach*. 1996, London, UK: Springer.
24. Ulrich, K.T. and S.D. Eppinger, *Product design and development*. 3. ed. 2004, Boston: McGraw-Hill. xviii, 366.
25. Bergsjö, D. and D. Malvius. *Use of Information Management Systems from Designers' Perspective*. in *NordDesign 2006*. 2006. Reykjavik, Iceland.
26. Malvius, D., D. Bergsjö, and S. Molneryd, *Balancing Operational and Strategic Impacts on Information Management* in *IDETC/CIE 2007*. 2007, Proceedings of the ASME 2007: Las Vegas, USA.
27. Bergsjö, D., *Management of Mechatronic Product Data in PLM Systems - Perspectives on Business and User Needs versus Technical Challenges*, in *Department of Product and Production Development*. 2007, Chalmers University of Technology: Göteborg.
28. Ottosson, S., *Dynamisk Produktutveckling in Swedish: Dynamic Product Development*. 1999, Tervix, Floda, Sweden.