

Towards a Reference Framework for Software Product Management

Inge van de Weerd, Sjaak Brinkkemper, Richard Nieuwenhuis, Johan Versendaal, Lex Bijlsma

Department of Information and Computing Sciences

Utrecht University, The Netherlands

{i.vandeweerd, s.brinkkemper, rnieuw, j.versendaal, a.bijlsma}@cs.uu.nl

Abstract

In the last decade, software product management has received much practical attention, though research in this area is still scattered. In this paper, we present a reference framework for software product management, in which the key process areas, namely portfolio management, product roadmapping, release planning and requirements management, are identified, as well as the stakeholders and their relations. With this reference framework, we provide a structure for a body of knowledge for software product management.

1. Product management

Software is more and more developed and commercialized as a standard product. In companies specializing in software products, the role of product manager has emerged over the last years, and appears to be of strategic value, but complex to execute. The product manager is responsible for managing requirements, defining releases, and defining products in a context where many internal and external stakeholders are involved [11]. The domain of product management has been established, especially in technical sectors with physical products, since the industrial revolution in the 19th century [15]. Only relatively recently, also software product management has received attention in product software companies like Microsoft [9] and Alcatel [10], and, to a lesser extent, in scientific literature, e.g. [15]. Although several of the existing product management practices can be applied in software product management, specific challenges can be identified in software product management. Software products differ from other products in the fact that the manufacturing and distributing of extra copies do not require extra costs for the company [8]. Also, existing software products can be changed easily, and sold software products can be updated by using patches or release updates. There is also a downside to these advantages. The organization of requirements and the tracking of changes in the design are very complex. Also, due to the ease of making changes, the release frequency is relatively high in comparison with non-software products. Finally, the product manager has a lot of responsibilities regarding the product

functionality, but has not the management authority over the development team, so decision making requires consent of many players. We therefore claim that there is a need for an integration of research efforts in this key domain.

Knowledge on software product management for research and educational purposes is very fragmented. There lacks an integrated body of knowledge, as exists for example in software development [3]. The goal of this paper is to provide a structure for a body of knowledge for software product management, by providing a reference framework based on an overview of state-of-the-art literature, and industrial case studies.

In the next section we elaborate on the rationale for the reference framework, and the research method we have applied to develop it. Then, in section 3, we discuss the basic structure of the reference framework. The four process areas are elaborated on in section 4. The final section describes our conclusions and future research.

2. Rationale and research method

Reference frameworks have proven to be beneficial for research and practice in many fields. We mention the ISO/OSI layers for the layering of network services [13] or the ANSI/SPARC 3-schema architecture for database management systems [22]. The need for a reference framework for software product management is found in the desire to get an understanding of its complete domain. Varying research contributions and developments in the software industry can be positioned in it to interpret their consequences in a uniform context. Furthermore, such a reference framework also provides a starting point for:

- Definition of key terms and the identification of open research questions;
- Education of product managers and competence building;
- Development of improved, integrated tool support;

The research method we have applied for the conception of the reference framework is the following:

1. Field interviews and discussions with experienced product managers;
2. Literature review on both non-software product management and software product management;
3. Creation of a draft reference framework;

4. Validation by an extensive case study at a large product software company;
5. Validation with input from an industrial workgroup on product management;
6. Finalization of the reference framework.

The draft framework was amended several times after suggestions from both practitioners and researchers. Small augments might still be needed, but we are convinced that the basic structure has been established.

3. Basic framework structure

The basic structure of the reference framework is based upon the objects or artifacts of product management, and upon the set of stakeholders identified in the scope of work of the product manager.

3.1. Artifact hierarchy

Professional product management is essentially a matter of well-organized processing of issues related to requirements, products and releases [11]. A hierarchical ordering of these artifacts imposes a structure on the process areas. First, the scope of work of product management concerns the complete set of products of the company, the so-called *product portfolio*. Small or young companies may have a portfolio of just one *product*, whereas larger companies have several. Each product has a release sequence of past, present and future *releases*. Several internal versions may exist versus selected externally released versions for the market. The release numbering is usually determined by internal conventions, where major changes in the technical architecture are a reason to call it an X.0 release. Finally, each release definition consists of a set of selected *requirements*. Each requirement implies the addition of a technical or functional feature to the product.

As the type of work differs when dealing with artifacts from the distinct hierarchy levels, this hierarchy gives rise to a subdivision of software product management into four process areas: *portfolio management* to deal with the products in the product portfolio; *product roadmapping* to deal with the different releases of each product; *release planning* to deal with the collections of requirements of each release; and *requirements management* to deal with the content of each individual requirement.

3.2. Stakeholder interaction

Product managers are confronted with a large number of requirements, originating from different internal and

external stakeholders. We distinguish the following internal stakeholders [11]:

- The *Company board* is responsible for the definition and communication of strategy, vision and mission.
- *Research & innovation* explores new opportunities for product innovations and finds ways to incorporate improvements or new features into the products.
- The consultants of the *Services* department are responsible for the implementation of the software product at the customer organization.
- *Development* has as main responsibility the execution of the release plan.
- *Support* stands for the helpdesk to answer questions and for the small defect repair unit.
- *Sales & marketing* is the first contact with a potential customer. Through these contacts new requirements can be gathered.

The following external stakeholders are recognized [16]:

- The *Market* is an abstract stakeholder, standing for potential customers, competitors and analysts.
- Most companies have different kinds of *Partners*: (1) implementation partners, (2) development partners, and (3) distribution partners.
- *Customers* often have new feature requests, which can be communicated to *Services*, *Sales & marketing*, *Support*, but also directly to the product manager.

Observe that the stakeholder names are generic, so that naming or grouping may differ in product software companies.

4. Reference framework

Except in requirements engineering, there is little literature explicitly addressing the domain of software product management. In this section, we provide an overview of state-of-the-art research on (software) product management. In Figure 1, a reference framework for software product management is visualized. In the remaining of this section, each of the process areas, defined in section 3.1, is provided with an explanation.

4.1. Portfolio management

Portfolio management entails the decision making about the set of existing products, introducing new products by looking at market trends and the product development strategy, making decision about the product lifecycle, and establishing partnerships and contracts. We also position product line management in this area. A software product line is a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment

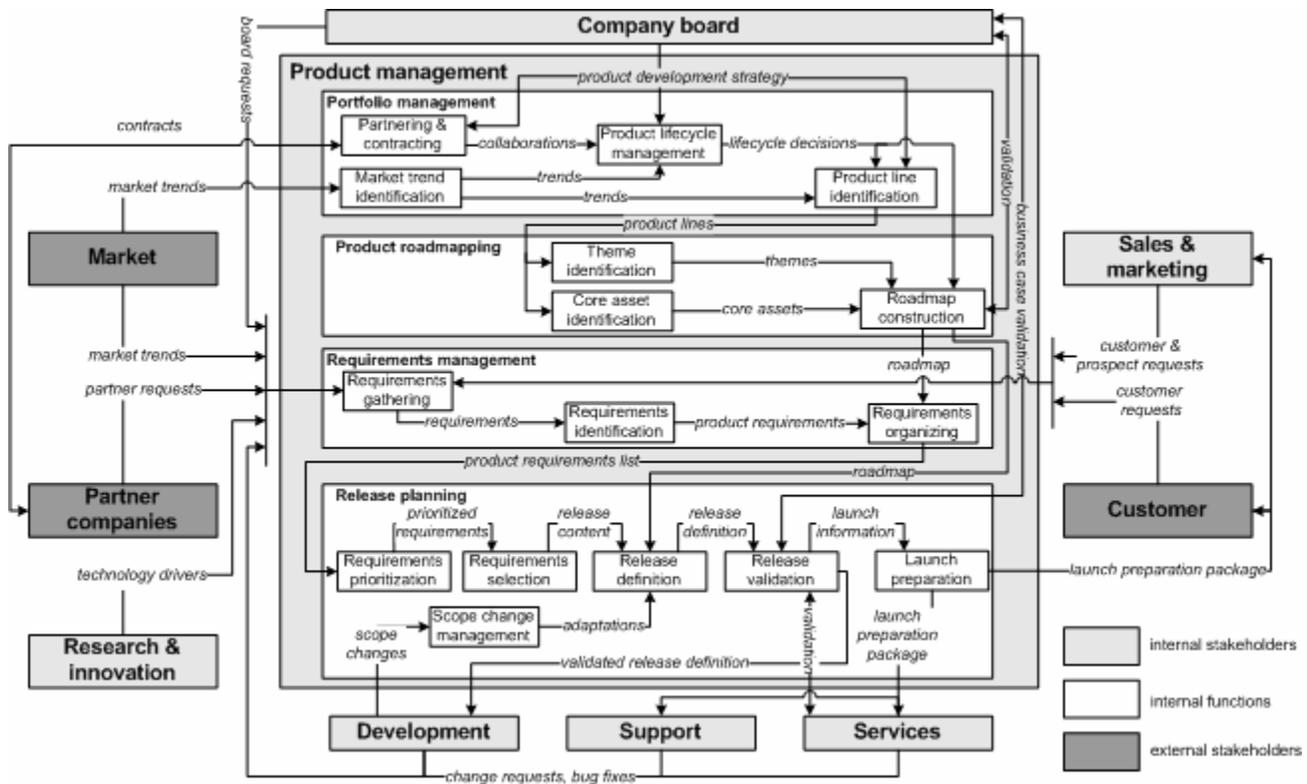


Figure 1. Reference framework for software product management

or mission and that are developed from a common set of core assets in a prescribed way [7]. This also comprises issues like variability across different product lines [20]. Several case studies have shown that introducing product lines organizations improves performance [2].

Looking at the reference framework, we see portfolio management on top. It contains four main processes: *partnering & contracting*, *market trend identification*, *product lifecycle management* and *product line identification*. Input is received from the Company board, Market and Partner companies.

4.2. Product roadmapping

In [23], roadmapping is called a metaphor for planning and portraying the use of scientific and technological resources, elements and their structural relationships over a period of time. It is complex due to dependencies on other related products (even from partners), technology changes, and the distributed development [6]. Roadmapping has, just like product line management, its origins in a sector distinct from the software industry, the manufacturing industry, where it is used for business oriented long-term planning and technology forecasting [16], however, also in the product software industry roadmaps are used for planning purposes [23].

In the reference framework, *product roadmapping* receives input regarding product lines from portfolio management. This input is used to *identify* themes and core assets [17] that can be used later on in the requirements organization. This information is gathered and described in the *product roadmap*.

4.3. Requirements management

Requirements management contains the activities of gathering, identifying, revising and organizing incoming requirements from the various stakeholders. Requirements management is a key area in product software companies [5]. Especially analyzing requirements costs a lot of time in product software companies, due to the (often) high amount of incoming new requirements per month. A solution to this problem is the use of linguistic engineering to link customer wishes to requirements [18].

Requirements management starts with *gathering all requirements* from within the company and from external stakeholders. These are translated to *product requirements* (i.e. requirements that will be implemented in the product) by removing the duplicates, connecting requirements that describe a similar functionality, and by rewriting them. Then, the requirements are organized per product and core asset.

4.4 Release planning

Software release management is the process through which software is made available to, and obtained by, its users [12]. Especially on the area of release planning, where the set of requirements for the next release is determined, much research has been carried out. Examples are release planning using integer linear programming [1], the analytical hierarchy process [14], stakeholders' opinions on requirements importance [21] and linear programming techniques using requirement interdependencies [4].

In the reference framework, release planning starts with the product requirements *prioritization*. After the prioritization, product requirements are *selected* that will be implemented in the next release. This can be done in multiple ways, as described in the listed literature. When the product requirements are selected, a *release definition is written* that is *validated* by different stakeholders. A business case is sent to the company board. When this has been approved by the board, a *launch preparation package is constructed* and sent to the stakeholders.

5. Conclusions and further research

In this article, we presented the reference framework for software product management. Furthermore, we provided an overview of state-of-the-art literature on software product management.

We are convinced that the proposed reference framework for product software management is a first step to position this important industrial domain in the field of scientific research in requirements engineering and to provide a structure for an integrated body of knowledge. Currently, we are developing a tool for operational support for the product manager [19]. In the future, we hope to contribute to further refinements of the reference framework and to one of the open research questions we identified, namely: which steps should be taken to introduce product management in a product software company successfully?

References

- [1] M. van den Akker, S. Brinkkemper, G. van Diepen, and J. Versendaal, "Flexible Release Planning Using Integer Linear Programming", *REFSQ'05*, 2005, Porto, Portugal, Essener Informatik Beitrage, Band 10.
- [2] J. Bosch, "Product-Line Architectures in Industry: A Case Study," *ICSE*, Los Angeles, CA, 1999, pp. 544-554.
- [3] P. Bourque and R. Dupuis, (ed.), *Guide to the Software Engineering Body of Knowledge*, 2004 edition, IEEE Computer Society, Los Alamitos, CA, USA, 2004.
- [4] P. Carlshamre, "Release Planning in Market-Driven Software Product Development Provoking and Understanding", *Req. Eng.* 7, 2002, pp. 139-151.
- [5] P. Carlshamre and B. Regnell, "Requirements Lifecycle Management and Release Planning in Market-Driven Requirements Engineering Processes", *DEXA '00*, 2000, p. 961.
- [6] E. Carmel, *Global Software Teams*, Prentice Hall: Upper Saddle River, NK, 1999.
- [7] P. Clements and L. Northrop, *Software Product Lines: Patterns and Practice*. MA: Addison Wesley, 2001.
- [8] M.A. Cusumano, *The Business of Software*, Free Press: New York, 2004
- [9] M.A. Cusumano and R.W. Selby, *Microsoft Secrets*, Simon and Schuster, New York, 1995.
- [10] C. Ebert and M. Smouts, "Tricks and Traps of Initiating a Product Line Concept in Existing Products" *ICSE '03*, IEEE Comp. Soc., Portland, OR, USA, pp. 520-525.
- [11] L. Gorchels, "The Product Manager's Handbook: The Complete Product Management Resource (2nd edition)", NTC Business Books, 2000.
- [12] A. van der Hoek, "Software release management", *ESEC/FSE '97*, Springer: Heidelberg, Germany, 1997, pp. 159-175.
- [13] Information Technology - Open Systems Interconnection - *Basic Reference Model: The Basic Model*. International Standard, ISO/IEC 7498-1. 2nd ed. Geneva: ISO, 1994.
- [14] J. Karlsson and K. Ryan, "A Cost-Value Approach for Prioritising Requirements", *IEEE Software* 14(5), 1997, pp. 67-74.
- [15] T. Kilpi, "Product Management Challenge to Software Change Process: Preliminary Results from three SMEs Experiments", *Software Process Improvement and Practice*, 3(3), 1997, pp. 165-175.
- [16] L. Lehtola, M. Kauppinen, M., and S. Kujala, "Linking the Business View to Requirements Engineering: Long-Term Product Planning by Roadmapping", *RE'05*, IEEE Comp. Soc., 2005, pp. 439-446.
- [17] M. Moon and K. Yeom, "An Approach to Develop Requirement as a Core Asset in Product Line", *Lecture Notes in Computer Science*, 3107, 2004, pp. 23 - 34.
- [18] J. Natt och Dag, V. Gervasi, S. Brinkkemper and B. Regnell, "A Linguistic-Engineering Approach to Large-Scale Requirements Management" *IEEE Softw.* 22(1), 2005, pp. 32-39.
- [19] R. Nieuwenhuis, I. van de Weerd, L. Bijlsma, S. Brinkkemper, and J. Versendaal, "The Software Product Management Workbench", *CAISE Forum*, 2006.
- [20] K. Pohl, G. Böckle, F. van der Linden, "Software Product Line Engineering", Berlin, Springer-Verlag, 2005.
- [21] G. Ruhe, M.O. Saliu, "The Art and Science of Software Release Planning", *IEEE Softw.* 22(6), 2005, pp. 47-53.
- [22] D. Tschritzis and A. Klug, "The ANSI/X3/SPARC DBMS Framework Report", *Information Systems I*, 1978, pp. 173-191.
- [23] J. Vähäniitty, C. Lassenius, and K. Rautiainen, "An Approach to Product Roadmapping in Small Software Product Businesses", *ECSQ2002*, *Conference Notes*, Center for Excellence Finland, 2002, pp. 12-13.