Adopting and Institutionalizing a Product Line Culture

Günter Böckle[1], Jesús Bermejo Muñoz[2], Peter Knauber[3], Charles W. Krueger[4], Julio Cesar Sampaio do Prado Leite[5], Frank van der Linden[6], Linda Northrop[7], Michael Stark[8], and David M.Weiss[9]

[1] Siemens AG, Otto Hahn Ring 6, D-81730 Munich, Germany
Guenter.W.Boeckle@mchp.siemens.de
[2] Telvent, C/ Tamarguillo 29, 41006 Sevilla, Spain
jesus.bermejo@sainco.abengoa.com
[3] Fraunhofer IESE, Sauerwiesen 6, D-67661 Kaiserslautern, Germany
knauber@iese.fhg.de
[4] BigLever Software, Inc, 10500 Laurel Hill Cove, Austin, TX 78730, USA
ckrueger@biglever.com
[5] Departamento de Informática, PUC-Rio,
R. Marquês de São Vicente, 225, Rio de Janeiro, Brasil 22453-900
julio@obatala.inf.puc-rio.br
[6] Philips Medical Systems, Veenpluis 4-6, 5684 PC Best, The Netherlands
Frank.van.der.Linden@philips.com
[7] Software Engineering Institute, Carnegie Mellon University,
Pittsburgh, PA 15213-3890, USA
lmn@sei.cmu.edu
[8] University of Maryland, College Park, MD 20742, USA
mstark@cs.umd.edu
[9] Avaya Labs, USA
weiss@avaya.com

Abstract. The strengths of product line engineering have been described before. But how can an organization make the move from developing one-of products to product line engineering without major interruptions in the day-to-day work? This paper describes how to perform the transition to product line engineering and lists the various strategies for such a transition. It also describes how to create an adoption plan and how to institutionalize product line engineering in an organization.

1 Introduction

There are several approaches to successfully bringing products to the market. For software-intensive systems, the product-family approach promises significant improvement in time to market and various other success parameters. The basic idea behind the product line approach is planned reusability. Once a software organization deploys such an approach, producing a new product that belongs to a family can be done at a fraction of the cost and time of a usual production process.

This document describes how an organization can perform the transition from a standard product development of one-of products to product line engineering. It is primarily the result of a working group at the Dagstuhl symposium on Product Family Development, April 16 - 20, 2001.

G. Chastek (Ed.): SPLC2 2002, LNCS 2379, pp. 49–59, 2002.

2 The Task

Transforming an organization to create products as members of a product family requires the directing of the business towards product-family development by installing corresponding processes, organization structures, and methods. Central to the introduction of the product line culture lies the challenge of convincing the different organization-decision levels about the appropriateness of investing in this technology. For mechanical and electronic systems, product line engineering is current practice. However, for software and for software-intensive systems, particular topics have to be considered.

This paper aims at describing a general process strategy for the adoption and institutionalization of a product line culture for software-intensive systems. Business cases have to be developed to justify adopting a product line approach and to define the actions required to accomplish that. A business case depends on which decision level in the organization we are trying to convince. Different roles have different interests and have to be addressed in different ways by such a business case to achieve the implementation of a product line culture.

3 The Big Picture

Figure 1 depicts the overall context of directing a business towards a software product line. It shows the major phases of the transition process.

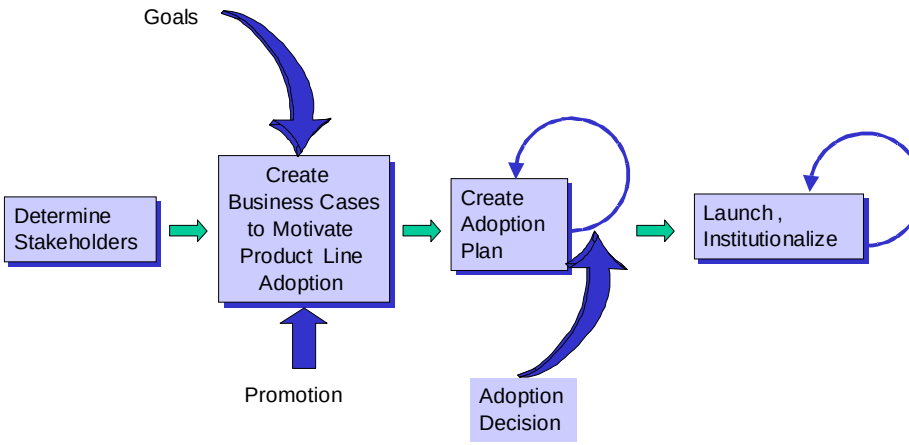

Fig. 1. The Big Picture - the Major Phases for Transition to Product Line Development

First of all, we have to consider the goals of a product line approach; these goals will vary according to the stakeholders we are addressing and their decision level in the organization. These goals will then be elaborated in business cases that promote the product line idea. The business cases and a first draft of the adoption plan will be the base for the decision to adopt a product line culture. After a positive decision, the

adoption plan will be elaborated in more detail and refined iteratively. After launch, the major concern is in institutionalizing the product line culture by integrating it in the organization culture. Launch and institutionalization are themselves iterative process phases, too.

4 The Stakeholders

The first step in the transition process is to determine the stakeholders. The different decision stakeholders (roles) are interested in different aspects of the product line concept. As a consequence, the business cases are role dependent and the plans will have to be adjusted accordingly.

There are often organization-specific roles that have to be considered. However, there are some general roles that will occur in most organizations:

- Product managers: Their interest is in marketing and customer analysis; their goals are revenue based. Time to market is important to them and their interest is in market breadth; often they are the ones who are involved in the assignment of budgets and strategy definition.
- Managers: They are interested in technical feasibility, but also in project planning, measuring progress, risk analysis, and getting presentable results early.
- Project managers: They are interested in technical feasibility, but also in project planning, measuring progress, risk analysis, and getting presentable results early.
- Engineers: They must be convinced that the approach is technologically sound and can help them in engineering; managers may ask their opinions about the approach.
- Quality assurance: They must be convinced that the approach increases quality. Managers may ask their opinions about the approach.

5 Business Cases

For each of these roles, a business case has to be developed that defines how product-family engineering helps them to achieve each of their specific goals. These business cases are used to convince the corresponding people of the advantages of the product-family approach.

Persons representing these roles will be involved in directing a business towards product-family engineering. They have to be motivated with the business cases and their tasks and responsibilities have to be defined.

The direction of the business case will first start with an adoption plan. It has to show certain measures that are relevant for the roles addressed in the business case. These are for instance revenues and profits, costs and return on investment (ROI), comparing one-of products to a family of products that is targeted to specific markets. More about business cases is described in [1] and [5].

6 The Adoption Plan

The persons doing adoption planning represent any one of the roles described in Section 4. In any case, the person(s) should have the responsibility to carry it out, trace it, and evolve it. The structure of the adoption plan has three major parts:

1. Characterization of current state: The state of the organization is characterized by its process, the staff and their expertise, the organizational structure, the project management methods, the engineering methods, and many other business parameters.
2. Characterization of desired state
3. Strategies, objectives, and activities to get from the current to the desired state

6.1 Characterization of the Current State

For characterizing the current state, the maturity of the organization has to be analyzed and described. The adoption of a product-family engineering approach starts with the assessment of the current state. Such an assessment may be similar to a Capability Maturity Model® (CMM) assessment; however, its purpose is not to obtain a number to compare the organization to other organizations. This assessment shall determine the strengths and weaknesses of the organization for product-family engineering and the points where special care (and thus measurement) has to be taken for the adoption. The following topics have to be assessed:

- Market characterization
- Competition in the market
- Degree of globalization
- Management commitment
- Organizational maturity
- Organizational stability
- Staff turnover
- Process maturity
- Project management maturity
- Expertise in product-family engineering (warm start vs. cold start)
- Domain expertise
- Other staff competence
- Contractors' expertise and reliability
- Communication with third parties
- Existing assets
- Software infrastructure

® CMM is registered in the U.S. Patent and Trademark Office.

6.2 Characterization of the Desired State

After having characterized the current state by a CMM-like assessment, the desired state is described in the same structure so that an easy comparison of current and future states is possible. Thus, the topics to be considered are

- The markets that shall be addressed by the product family
- The competition in these markets
- The degree of globalization that is necessary for these markets
- The management commitment necessary for moving to product line engineering
- The organization structures proposed for product-family engineering and the requirements for the organizational stability
- The staff stability required to prevent know-how from getting lost
- The processes proposed for product-family engineering and methods to ensure that the processes are really enacted
- The required project management skills and methods
- The expertise of the staff that is necessary for product-family engineering
- The domain expertise required for the targeted markets and products
- Other competencies of the staff required for product-family engineering
- The expertise that has to be delivered from contractors because it cannot or need not be built up in the organization; the reliability and kinds of contracts required
- The communication with third parties (e.g., communication infrastructure to contractors, communication rules, milestones)
- List of other assets needed for product line engineering
- The software infrastructure needed for product line engineering (e.g., tools for product planning, scoping, requirements engineering, architecture modeling, design, design management, asset management)

6.3 How to Get from the Current to the Desired State

There are several strategies for getting from the current state to the desired state. The adoption plan has to characterize and prioritize them and suggest the best-suited strategy for transition. Figure 2 lists the major strategies for this transition.

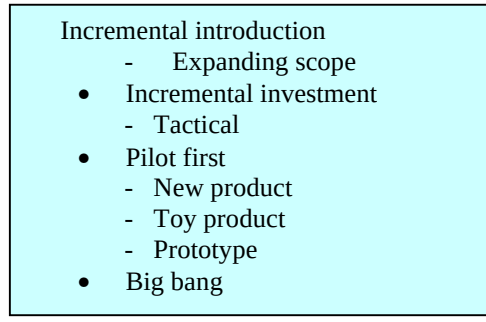

Fig. 2. Product Line Adoption Strategies

6.3.1 Incremental Introduction

The expanding scope strategy may start with a single department doing product line engineering and when they are successful, other departments are added incrementally. Mostly, there will be several departments developing products. One department for developing core assets may be installed and their results used by the other departments, as they get ready. The other departments may then be reorganized accordingly.

6.3.2 Incremental Investment

At the beginning a certain, limited investment is assigned to product line engineering while the major part remains in conventional product engineering. It has to be planned carefully which activities will be funded to start the product line; typically this is in writing reusable components or making existing components reusable. However, the product line scoping and planning stage has to be introduced as early as possible to ensure the significance of the efforts and to make them measurable and predictable. The budgeting plan for incrementally introducing product line engineering may be informal or more or less rigorous but planning and measuring is central!

The tactical strategy is driven by problems with the conventional approach, e.g., with configuration management for multiple related one-of products. Reusing core assets will reduce the complexity of configuration management. This approach may start informally. However, the scoping of and the planning for the further development and application of cores assets has to be performed early so that the results can be made measurable and predictable.

6.3.3 Pilot First

The New Product strategy is started as the first member of a new product line, or an existing series of related products, is extended by a new product for which certain activities of product line engineering are applied. For this approach, particular methods are important, like architecture recovery.

The Toy Product strategy recognizes that the risk of creating a new product with a new approach may be assessed as being too high in certain environments. In this case, a toy product may be created that is sufficiently close to the organizations' products, so that some of the results can be reused if product line engineering is introduced later for the "real" products.

The Prototype strategy recognizes that a prototype may be the first product in a new product family or a new product in a series of existing related products. In either case, the product line engineering that will be applied during the pilot project has to be planned beforehand, and the process has to be determined accordingly.

6.3.4 Big Bang

In cases where the management of an organization is convinced of the advantages of a product line engineering approach and where it is essential for the business to achieve the business results of this approach as early as possible, a big-bang approach may be

the best one. In this strategy, the product-family engineering approach is adopted for the new products of the organization immediately.

Whatever the strategy for product line adoption may be, it is always necessary to introduce review points in the transition process where the current state of the adoption is evaluated and the results of the new engineering approach are compared to the conventional approach. This requires the introduction of measurements during the transition process that can be used for these evaluations. These reviews have to adjust the transition process, and may even lead to a change in the adoption strategy.

The objectives and activities for transition depend on the strategy chosen. They include ways of integrating the transition process into the everyday work so that the process of developing products for the customer is not disturbed. Also included are activities from the business cases, including the necessary training for the staff and management. These activities for transition shall include estimations of effort and cost so that a decision about adoption can be supported. These estimations can be derived from the business cases.

7 Launching

The initial version of the adoption plan is used to decide about the product line adoption. If the adoption is chosen, the product line is launched.

The following are common parts of the launch for the activities described above:

- Recognition of stakeholders, their interests, needs, biases and culture.
- Determination of roles and responsibilities in the current and future state, together with a corresponding migration path.
- Definition of a product-family engineering process for the organization, considering the products, skills, environment, and so forth.
- Definition of the production process.
- Motivation of all affected stakeholders, using the business cases (see Section 5).
- Cultivation of champions and angels for all-important activities.
- Introduction of data collection, metrics, and tracking.
- Plans for funding.
- Plans for staffing, including consultants and providers.

Champions are staff members who are convinced of a new idea (e.g., product line engineering) and try to convince other staff members, too. They support others in applying the approach; they present practical tips and support the proliferation of the idea. Angels are members of management or senior engineers who have influence due to their role in the hierarchy or due to their experience and expertise. They give the necessary weight to the launch of an approach like product line engineering.

8 Institutionalization

Just launching the product line is not sufficient. It has to be institutionalized so that the involved managers and staff will consider it as part of their working culture.

Permanent organizational fixtures are then installed that support and enforce product-family engineering.

Institutions are considered stable structures [3]. “Organizations become institutions as they become valued in and of themselves: their disappearance would create drastic problems for various members and groups in the environment” [4].

Software product line institutionalization occurs when the product line processes are perceived as stable and indispensable to a given software organization. Iacono and Kling [2] believe that large scope computer-based information systems should be viewed as social institutions. As such, those systems are a complex web of interactions among actors, equipment, roles, procedures, rules, and activities. However, such systems can only be viewed as an institution when they are stable and indispensable to the organization.

This said, we do not contradict the fact that those systems, to fight entropy, need to be in constant evolution. Once a product line approach is institutionalized, it continues to evolve, but does so in a stable position and with a high demand from the organization that will see it as an indispensable part of its way of doing business. Although institutions strive for immortality [6], of course they are not immune to changes.

The institutionalization has to consider the organizational structures and ensure that product-family engineering is engrained in the organization’s processes. Incentives have to be defined and installed; funding has to be put into place by proven funding sources; and measures that reflect product-family engineering have to be installed and used as bases for reviews. There are two aspects to understanding institutionalization: what determines the extent to which an action or structure is institutionalized and the consequences of actions or structures being institutionalized. Iacono and Kling [2] list several determinants and consequences of institutionalized computer-based information systems. Using their list as a basis, we point out the main determinants and consequences of institutionalized software product line below.

8.1 Determinants

- Social arrangements such as policies, procedures, conventions, and the distribution of resources are developed and utilized by the main actors (audience).
- There are key stakeholders in the environment who wish to maintain or increase their control, despite the fact that no actor or coalition has complete control.
- There are negotiations, commitments, and future obligations among the main actors (audience) that are not easy to reverse.

The successful launch of a product line, for a given family of products, will bring new social arrangements to the organization. These will be seen as new policies, procedures, conventions, and distributions of resources in software production, but as well on the cooperation with customers and quality assurance. When these ensembles of resources and people are in place, we have a strong determinant towards making a product line an institution.

The champions of the idea as well as their angels will drive to maintain or increase control over the organization as a natural consequence of product line success. This is a main driving force towards institutionalization. Champions and angels will strive to

use marketing strategies to ensure that the whole organization perceives the advantage of product lines over traditional methods.

Of course in the process of marketing the product line, several commitments and future obligations will have to be established by the key stakeholders who have other interests in the organization. These commitments and obligations will persist since they represent a huge investment of the organizations.

8.2 Consequences

- Social arrangements between the main actors persist over time, making it difficult for small coalitions to change the status quo.
- People believe that the social arrangements will persist.
- Stable arrangements ease cooperation along routine behaviors and constrain substantially different behaviors.
- Change is localized.

Once a product line is institutionalized, change is localized, usually by add-on technologies, and people believe that the product line will persist. The product line framework will guide the cooperation among software personnel, customers, and quality assurance personnel in such a way that changing procedures or conventions will be difficult.

On the other hand, once a product line is institutionalized, the organization will benefit from better efficiency overall. With social arrangements working properly and the technology infrastructure in place, the organization will be producing software products faster, better, and cheaper.

9 A Bottom-Up Approach

A common challenge that organizations face with all of the adoption strategies is the "adoption barrier" – the overhead associated with starting the software product line. Typically resources have to be shifted from existing projects, and rarely do existing projects have resources to spare.

BigLever Software [7] promotes a *bottom-up* approach to address this problem. BigLever Software GEARS is a technology that allows organizations to gradually convert existing one-of-a-kind software and processes into product line software and processes [9]. The premise of GEARS is that any existing software product is a valid software product line with only one member and that new members can be added incrementally using the GEARS *infrastructure* to manage variation among the product members.

For example, during two weeks in August 2001, two engineers at Salion [8] allocated approximately 25% of their time using GEARS to convert their one-of-a-kind enterprise software product into a software product line with three members. Although the three initial members varied in simple ways, Salion had truly adopted a product line approach that could incrementally grow more sophisticated over time.

The key to this low adoption barrier is GEARS' *separation of concerns*. GEARS is an extension to the current arsenal of software engineering languages, tools, and

techniques. It provides a *software product line infrastructure* that can be inserted into any collection of source files and directories in order to introduce parameterized variation. The GEARS infrastructure is comprised of four basic constructs:

- Feature declarations – parameters of variation for the software product line.
- Product definitions – assignment of values to feature-declaration parameters for each product member.
- Automata – the encapsulated mapping from parameter values to source-code variants among the product members.
- Actuator – the mechanism for instantiating product members from the product source code and the GEARS infrastructure.

When using the GEARS technology during software product line adoption, BigLever distinguishes between three different adoption strategies that can be used independently or in combination:

- Extractive – a "bottom-up" approach where one or more existing legacy systems are reused as the basis for the product line.
- Reactive – an "incremental" approach where product line members are built on demand as new product requirements are identified.
- Proactive – a "big-bang" approach where a significant number of the product line members are identified and built upfront.

10 Summary

This paper demonstrates the importance of the following steps for a successful transition from producing one-of products to product line engineering:

- Prepare business cases that determine the activities, efforts and costs from the perspectives of the major stakeholders, as a basis for deciding about the transition.
- Make an adoption plan with the activities needed for the transition so that staff and management will support the transition.
- Institutionalize product line engineering in the organization so that it becomes part of the company culture.

References

1. The Business Case Web Site, http://www.solutionmatrix.com/
2. Iacono, S. and Kling, R.: Computer Systems as Institutions: Social Dimensions of Computing in Organizations, in *Proceedings of the Ninth International Conference on Information Systems*, 1988, pp.101-110.
3. Meyer, J.W. and Rowan, B.: Institutionalized Organizations: Formal Structure as Myth and Ceremony, American Journal of Sociology, vol. 83(2), 1977.
4. Perrow, C.: Complex Organizations, A Critical Essay, Soctt Foresman and Company, 1979.
5. A Framework for Software Product Line Practice - Version 3.0, http://www.sei.cmu.edu /plp/framework.html

6. Selznick, P. Leadership in Administration: A Sociology Interpretation, New York – Harper and Row, 1957.
7. BigLever Software, Inc.: *www.biglever.com*, Austin, TX. 2001.
8. Salion, Inc.: *www.salion.com*, Austin, TX. 2001.
9. Krueger, C.: Using Separation of Concerns to Simplify Software Product Family Engineering. April 2001. Proceedings of the Dagstuhl Seminar No. 01161: Product Family Development. Wadern, Germany.