Software Engineering Strategies: Aligning Software Process Improvement with Strategic Goals

Reinhold Plösch[1], Gustav Pomberger[1], and Fritz Stallinger[2]

[1] Kepler University Linz, Institut für Wirtschaftsinformatik - Software Engineering, Altenberger Straße 69, 4040 Linz, Austria
{reinhold.ploesch,gustav.pomberger}@jku.at
[2] Software Competence Center Hagenberg, Softwarepark 21, 4232 Hagenberg, Austria
fritz.stallinger@scch.at

Abstract. Aligning software process improvement with the business and strategic goals of an enterprise is a key success factor for process improvement. Software process improvement methods typically only provide little or generic guidance for goal centered process improvements. We provide a framework for developing software engineering strategies that are aligned with corporate strategies and goals. Strategic objects as an important part of our framework can be directly aligned with SPICE or CMMI processes. This allows that any process improvement action can be systematically aligned with strategic goals.

Keywords: Software process improvement, software engineering strategy, functional strategy, strategic goal, CMMI.

1 Introduction and Overview

Aligning software process improvement with the business and strategic goals of an enterprise is a key success factor for process improvement. Intensive research has been performed on defining best practice models for software lifecycle activities (e.g. [1], [2]) as well as methods for guiding software process improvements, ranging from guidance for single improvement actions (e.g. [3], [4]) to the management of overall improvement programs (e.g.[5], [6]). Although these methods generally consider the existence of strategic goals and stress the importance of aligning process improvements to business goals, they generally provide little and typically only generic guidance on how to define the details of, prioritize and select process improvements.

In the remainder of this paper we present a method for development of functional software engineering strategies as a mediator between business goals and software process improvement. The main results presented are (a) an understanding of the role of engineering strategies in the overall strategy development context of an organization, (b) a meta-model for describing engineering strategies, (c) the identification of the strategy objects relevant for software engineering and their mapping to CMMI process areas, and (d) a method to guide the development of engineering strategies.

2 Developing Software Engineering Strategies

In order to understand strategy development at the software engineering level we first relate software engineering strategies to the overall strategy development efforts in an

R.V. O'Connor et al. (Eds.): SPICE 2011, CCIS 155, pp. 221–226, 2011.

organization. Fig. 1 illustrates the overall strategy development process of an organization. We will not discuss all the steps but refer to [7] and [8] for a detailed discussion. According to [8], typically a distinction is made between the corporate strategy, various division strategies, and various functional strategies.

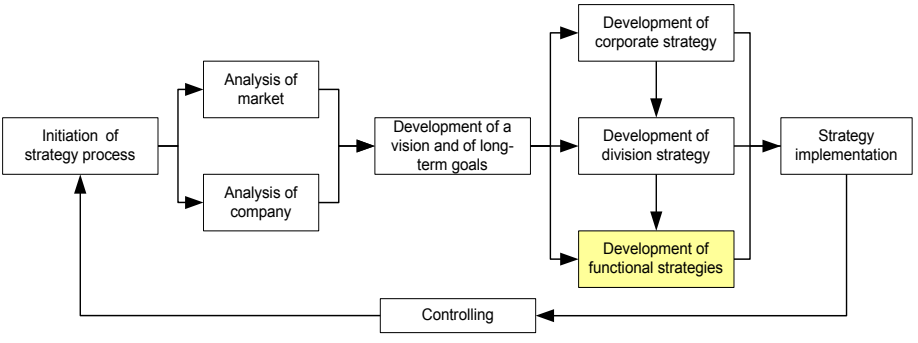

Fig. 1. Overall strategy development process

Corporate Strategy: The central issue on this level is to determine which market segments should be addressed with which resources. This has to be understood against the background of the core tasks of a company – resource allocation, diversification decisions and the coordination of the more or less independent divisions (management of synergies).

Division Strategy: The division strategy refines the corporate strategy. The major questions to be addressed by the division strategy are how to develop a long-term unique selling proposition compared to the market competitors and how to develop a unique product or service. The competitive advantages of a division are related to its capabilities and resources and to the customer needs and structures of the market.

Functional Strategy: Functional strategies define the principles for the functional areas of a division in accordance with the division strategy and therefore refine the division strategy in the distinct functional areas. Examples of such functional areas are marketing, finance, human resources, engineering, software development, etc.

In principal these strategies can be developed independently from each other; nevertheless they must all adhere to the division strategy and therefore also to the corporate strategy. While on the corporate and on the division level the emphasis is on the *effectiveness* (doing the right things) of the corporation or division, the functional strategies have their focus on the *efficiency* (doing the things right) of the respective functional areas. This distinction between the different kinds of strategies ensures that business goals are translated from the corporate strategy to the functional strategies.

In a next step we need to understand the structure of functional software engineering strategies. Fig.2 depicts the conceptual framework for the description of functional strategies. The strategic goals formulated in the software engineering strategy are refinements of strategic goals on the corporate and respectively divisional level, mapped on the functional area. A strategy object is a topic (e.g. architecturemanagement) that refines one or more strategic goals. As the strategy objects—and therefore also the strategic statements—are targeted towards the functional strategic goals it is also assured that the divisional or corporate goals are not violated.

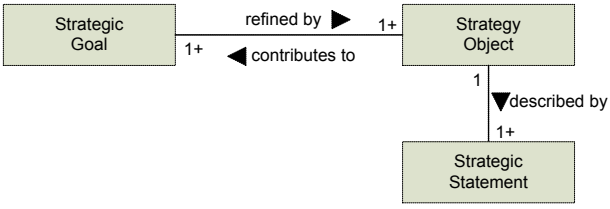

Fig. 2. Strategy description - conceptual framework

Each strategic goal has a description and explanation of the strategic goal, lists expected benefits, describes how to reach the strategic goal and contains a description of how to measure its realization. Furthermore, each strategic goal is prioritized. Table 1 provides an example description of a strategic goal from a real-world project. The verbalization of strategic goals is not an easy task and should be based on knowledge from a detailed analysis of the organization. A strategic goal of a software engineering strategy must not violate corporate or division goals or visions.

Table 1. Example for description of a strategic goal

ID:	G-SALE	*Priority:*	A-Goal
Strategic Goal:	Selected software products have to be sellable separately, i.e. without selling the underlying hardware product.		
Explanation of strategic goal:	The selected software products must meet conditions, so that they can be sold independently of other products (hardware and software) on the automation market.		
Description how to reach the strategic goal:	This is achieved by appropriate abstraction of the runtime environment, isolation and independence from other products, extensive tests, appropriate actions for the protection of intellectual property, documentation and consulting and support offers.		
Description how to measure the realization of the strategic goal:	Guideline for achieving this goal is that by the end of the first quarter of 2011 product X is sellable alone and independently of other products.		

Examples of strategy objects that are typically refined during the strategy development process include architecture management, quality management, requirements management, standards management, etc. The description of strategy objects comprises their definition, identification of typical topics dealt with, and examples of strategic statements. Table 2 gives an example of a description of a strategy object from a real-world project.

Table 2. Example for description of a strategy object

ID:	O-WORK
Definition:	Work Organization is the systematic arrangement of effective and efficient software development and project execution.
Strategic statement 1:	In the areas of Firmware (incl. Technology), Human-Machine-Interface and Tools the following developer teams have to be formed: OEM development, product development, and maintenance
Strategic statement 2:	Each software developer is member of one of these teams. For capacity reasons a developer may temporarily join another team, but the number of these developers should be kept low.

The general approach of the method for the systematic development of software engineering strategies is to conduct a strategy development process as shown in Fig. 3. The typical process is structured into the development and prioritization of strategic goals, strategy objects and strategic statements.

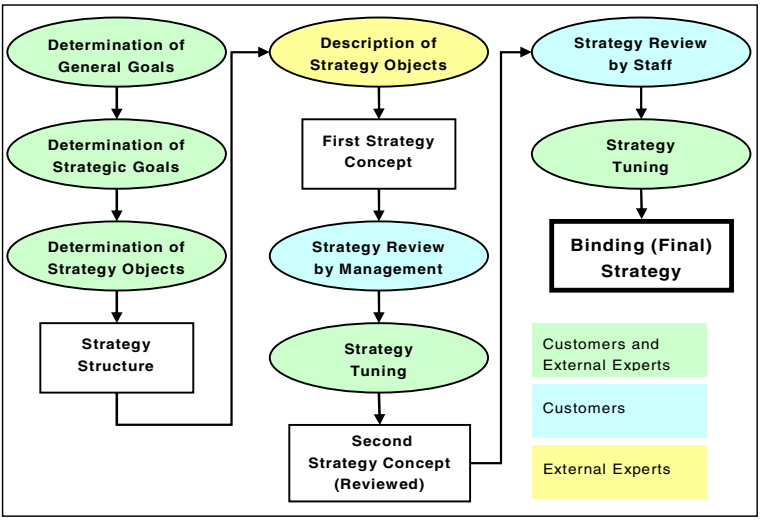

Fig. 3. Process of engineering strategy development

Table 3. Mapping of strategy objects to CMMI Process Areas

CMMI Process Area	**Strategy Object(s)**	
Causal Analysis and Resolution	Quality Management	P
Configuration Management	Configuration Management	F
Decision Analysis and Resolution	Architecture Management, Component Management, Innovations Management	P
Integrated Project Management	Project Management, Work Organization	F
Measurement and Analysis	Quality Management, Process Management	P
Organizational Innovation and Deployment	Innovation Management	F
Organizational Process Definition	Process Management, Work Organization	F
Organizational Process Focus	Process Management, Work Organization	F
Organizational Process Performance	Process Management, Quality Management	F
Organizational Training	Project Management, Work Organization	F
Product Integration	Component Management, Quality Management, Test Management	L
Project Monitoring and Control	Project Management	F
Project Planning	Project Management	F
Process and Product Quality Assurance	Quality Management	L
Quantitative Project Management	Quality Management, Process Management	P
Requirements Development	Component Management, Product, Product Management, Domain Engineering	L

Table 3. (*continued*)

Requirements Management	Requirements Management, Change Management	F
Risk Management	Risk Management	F
Supplier Agreement Management	-	N
Technical Solution	Architecture Management, Component Management, Domain Engineering	L
Validation	Quality Management	L
Verification	Quality Management, Test Management	F

From our experience in applying the strategy development process, we identified a large number of strategy objects that are typically of interest for organizations. We therefore mapped the CMMI process areas to the strategy objects in order to facilitate that process improvements can be systematically aligned with the strategic specifications. The result (cf. Table 3) shows that most process areas are semantically connected with the strategy objects of our approach. This gives us the possibility to systematically crosscheck, whether process improvements are aligned with strategic decisions and goals. The mapping defines in more detail which process areas (i.e., improvements of a process area) have to be aligned with which parts of the functional software engineering strategy. The mapping is quantified by an N-P-L-F-scale adapted from [1].

3 Experience and Conclusion

From our experience in applying the strategy development process in various software development organizations (with more than 100 software developers, each) using our conceptual framework of strategic goals, strategy objects, etc., we can draw the following conclusions: (a) the structure helps focusing on the strategic goals; (b) as the strategy objects are linked to the strategic goals, the strategic statements are automatically targeted towards the goals of an organization; (c) the mapping of the CMMI process areas to strategy objects allows aligning identified improvements to the strategy objects and therefore (by means of the link of the strategy objects to the strategic goals) to the business and strategic goals of an organization.

References

1. ISO/IEC 15504:2003. Information Technology - Process Assessment. International Standards Organization (2003)
2. CMMI for Development, Version 1.2. Technical Report CMU/SEI-2006-TR-008. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA (2006)
3. Shewhart, W.A.: Economic control of quality of manufactured product. D.Van Nostrand Company, New York (1931)
4. Dion, R.: Process improvement and the corporate balance sheet. IEEE Software 10(4), 28–35 (1993)

5. McFeeley, B.: IDEAL: A user's guide for software process improvement. Handbook CMU/SEI-96-HB-001, Software Engineering Institute, Carnegie Mellon University, Pittsburgh (1996)
6. ISO/IEC 15504-7:1998. Information Technology - Software Process Assessment - Part 7: Guide for use in process improvement. International Standards Organization (1998)
7. Simon, H., von der Gathen, A.: Das große Handbuch der Strategieinstrumente – Alle Werkzeuge für eine erfolgreiche Unternehmensführung. Campus, Frankfurt/Main (2002)
8. Venzin, M., Rasner, C., Mahnke, V.: Der Strategieprozess – Praxishandbuch zur Umsetzung im Unternehmen. Campus, Frankfurt/Main (2003)