Proc. of CONQUEST 2001, Nürnberg, 29-40.

Software Process Improvement at ABB

Abstract

ABB has a long tradition of improving software processes and of applying CMM in SPI projects. This paper presents the structure and results of a company wide SPI initiative called ASPI. The initiative aims to complement the existing local SPI projects by harmonizing processes, methods and tools to be used in R&D software development throughout ABB in a common framework. We give an overview of the structure of this framework and present in detail the common decision model for product development projects which forms an important part of the framework. This model is generic and has to be tailored to the needs of local ABB R&D *units. We conclude by discussing the experiences gained during introducing the decision model and give a brief outlook on the SPI activities planned in future.*

1 Introduction

Many companies in the software business have realized that substantial gains in productivity of software development and quality of software products can only be achieved by improving the software process. Such improvements have turned out to be costly and bearing a high risk of failure, since the software process is a complex system of relationships among processes, technologies and people. Results and experiences of industrial software process improvement (SPI) initiatives have been recently published. For instance Wohlwend (1994) describes Schlumberger's and Komiyama (2000) presents NEC's SPI program.
ABB is a group of companies operating in different countries and business areas. The companies are managed locally. As a consequence, smaller software process improvement initiatives have emerged independently at different ABB companies, with specific improvement objectives. The results and experiences of these improvement activities are discussed in detail in Lichter (1995) and Welsch (1997).
In this paper we present the structure and results of a company-wide SPI initiative called ASPI (**A**BB **S**oftware **P**rocess **I**nitiative). This initiative is funded by the ABB software engineering R&D program and is carried out by consultants from ABB Corporate Research. The initiative aims at defining and setting up global software process elements and coordinating local SPI efforts.

The paper is organized as follows. Chapter 2 presents all elements of ASPI. Chapter 3 describes in detail the R&D control model for software development, which is the core element of ASPI. In chapter 4 we introduce the ABB Gate Model. This model is part of the R&D control model and serves as an overall decision model. Chapter 5 presents how the Gate Model is introduced and implemented in local ABB business units. Chapter 6 summarizes our conclusions and gives an outlook on SPI activities planned in the future.

2 Overview of ASPI

To complement the existing local SPI activities, to gather global synergies (c.f. Thomas, 1994) and to coordinate and support a continuous approach to establishing and improving good software processes, ABB started in 1999 a company wide SPI initiative (ASPI). ASPI stresses business and project management issues and tries to link SPI activities to business goals (see Debou, 2000).

Organization

ASPI was started by ABB's Corporate Research Organization with the goal to install a continuous improvement culture in ABB within three years. Meanwhile, a unit in ABB's group-wide process organization has been created to own the SPI activities. As shown in Figure 1, several organizations contribute to ASPI:

- **ABB Group Processes**: The ABB group has a central process organization called Group Processes, who owns all common ABB processes, e.g. for product development or business project execution. A unit in this organization has been established to centrally coordinate the different SPI projects. This unit initiates the local SPI activities by addressing local management and monitors the status of the SPI activities. It further ensures consistent application of the standards developed within ASPI.

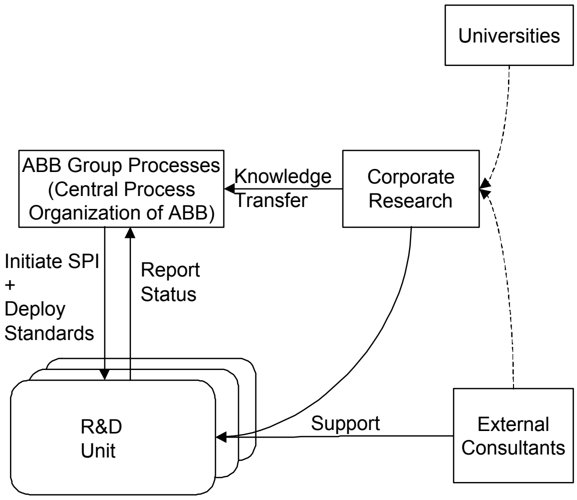

Fig 1: Organization of ASPI

- **Corporate Research**: After setting up the initiative, Corporate Research evaluates, packages, and transfers the knowledge from external partners (universities, consultancy companies) to the process organization. Experts from Corporate Research further support business units in SPI projects.
- **R&D Units**: Selected units are required to budget for SPI, create SPI plans, conduct improvement activities and report the status of the activities to the process organization. In return part of the external cost of the unit's SPI activities is covered by corporate fund of the ABB group.

Common Framework for Product Development

Since 1999, ASPI has been working intensively towards two goals:

1. Harmonize processes, methods and tools to be used in R&D software development throughout ABB in a common framework.
2. Create a culture of continuous self-improvement in the R&D units with the goal to work on an efficient and mature process level.

These goals complement each other by approaching the R&D units from both top-down through the ABB-wide common framework and bottom-up through local SPI projects, as indicated in Figure 2. On the top-down side ASPI is providing a Common Framework for Product Development and Maturity, consisting of the following elements:

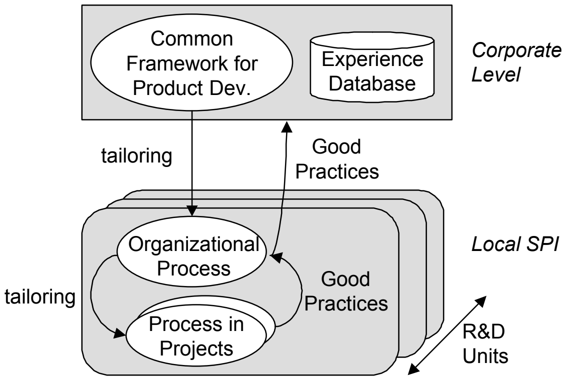

Fig 2: Company and local SPI levels

- **A model for R&D execution**
  This model defines a common language and procedures in the areas Project Decision Model, Project Management Model, and Development Model. It is covered in more detail in section 3.
- **A model for execution of process improvement projects**
  This model is called the Improvement Engine. It is in spirit similar to the SEI Ideal model (McFeeley, 1996) and defines the phases of a CMM-based improvement cycle.
- **A set of 4 common metrics**
  The Common Framework defines metrics for performance baselining, covering time, effort, quality and functionality.

The Common Framework establishes a common language throughout the company and provides common management procedures. Furthermore, it generates management commitment to SPI activities by providing a set of tools to support planning and tracking of such activities.

Experience Database

Another key element of ASPI is the experience database (EDB). The goal of the EDB is to facilitate sharing and reuse of experience. In contrast to the experience factory approach (Basili, 1994), ASPI had to construct the EDB with minimum administrative overhead and the goal of short payback time. Thus, we are targeting a lightweight approach to experience reuse, comparable to the environment described by (Houdek, 1998).

In the EDB, experiences are represented using a structured template. Content is controlled by a small organization, called the experience control board (ECB). There are 4 levels of experiences with different requirements on elaboration and degree of reuse of the experience, as sketched in Figure 3. The levels are

- **Entry level experiences**: This is the most informal type of experience, e.g. an observation made in one unit. It must contain contact information, experience description and lessons learnt. An example for this category is the description of a daily build process of one R&D unit.

- **Experience**: In this category a certain level of reuse is required. Based on the application of the experience success factors have to be given as part of the experience package. An example for this category is a code review seminar conducted and analyzed at different R&D units.
- **Good Practices**: On the next level, a proven experience can become a good practice. Good practices must contain a cost-benefit analysis and a guide for introducing the practice. The experience package must contain all necessary artifacts for applying the practice. An example for this category is a test process used consistently in different R&D units of a business area.
- **Approved Policies**: A good practice can become a mandatory process element upon decision by the process organization. An example for this category is the ABB Gate Model.

The classification of the experience into one of the levels is decided by the ECB after review.

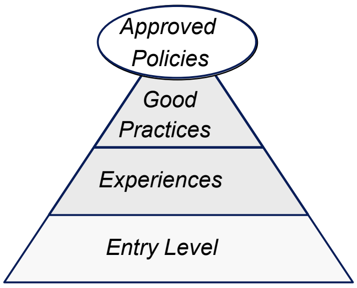

Fig 3: Levels of experience packages

The EDB is accessible for every ABB employee through a portal, which contains the Common Framework, the experiences, discussion groups, announcements, etc. The challenges in sustaining the EDB are (1) to keep the users interested in the EDB and (2) to ensure that new experiences are constantly added. The integration in a Web site helps achieve (1). ASPI publishes additions to the Common Framework, announcements and edited articles regularly on the EDB web site. Challenge (2) is harder to address, since the ECB is too small to edit all experiences and the R&D units themselves need incentives for creating them. We solve this through the funding scheme of SPI projects. In the application for the corporate funding share, the consultant / change agent has to identify a candidate for an experience out of the project.

CMM-based SPI activities

The top-down approach of ASPI is complemented by bottom up SPI activities within the different local software development units. ABB has a long tradition of the application of CMM in such SPI projects (c.f. Welsch 1997). Over the years ABB has developed a toolbox of several types of CMM assessments:

- **CBA IPI**
  This „official“ CMU-SEI assessment type is applied by larger ABB software development units, who have been investing in SPI for a longer period. The effort of this method is justified by its scientific accuracy. ABB works with CMU-SEI to educate internal assessors for this method.
- **ABB Mini CMM Assessment**
  The first ABB Mini CMM Assessment method was created in 1993 based on the study of the CMM (Paulk et al., 1993) and has been revised based on our experiences with the CBA IPI. It tries to generate comparable results with fewer assessors and less effort. While the redundancy in the assessment team (2 to 4 assessors) is not as high as in the CBA IPI, we maintain consistent results by strict requirements on education and experience of lead assessor and assessment team.
- **CMM Onboard**
  Since 2000, ABB is also using the CMM Onboard process developed by Q-Labs AB. CMM Onboard is a process, which allows development projects to reach CMM level 2. In this process a graphical representation of the CMM level 2 in the form of a board is created and constantly updated during the improvement project. We have found this process useful especially for small R&D units, who are at the beginning of their SPI efforts.

Over the last years, restructurings and mergers have impacted many R&D units. While it is difficult to achieve high maturity levels in such a dynamic environment, the knowledge from CMM levels 2 and 3 is seen by most units as critical for establishing a dependable process culture.

3 The R&D Execution Model for Product Development

The R&D execution model for product development consists of four layers (see Figure 4). The product planning process forms the topmost layer of the model. It serves to systematically plan and manage the product portfolio.

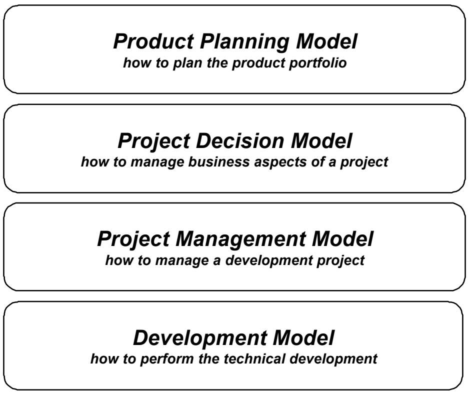
Product Planning Model
how to plan the product portfolio
Project Decision Model
how to manage business aspects of a project
Project Management Model
how to manage a development project
Development Model
how to perform the technical development

Fig 4: Layers of the R&D execution model

In order to develop a product and to control the development progress, the R&D execution model defines a Project Decision Model (called Gate Model), a Project Management Model as well as a Development Model.

Gate Model

The ABB Gate Model is the common decision model for product development. The ABB Gate Model helps to make the project status visible and provide relevant data as the basis for business decisions. This is achieved through 7 defined decision points (called Gates G0 to G6) during the development project's lifetime. One additional check point (G7) after the project is used for checking the results of the project and feeding back experiences to the organization.

Project Management Model

The Project Management Model helps the project manager to run a project according to the ABB Gate Model and to provide the information required for the gate decisions. It introduces a common terminology for project management activities throughout ABB and establishes ABB-wide procedures for project steering.

Development Model

The development model refers to the actual software engineering process of the development unit. Due to the great variety of businesses and products in the ABB group, there is not much potential for standardization on this level. However, the Common Framework provides guidance on how different software lifecycle models (sequential, incremental, evolutionary) are used with the ABB Gate Model and Project Management Model.

4 The ABB Gate Model

Decision models for product development projects have been discussed by Cooper (1993). They have paramount importance for most product-based enterprises, since they

- guarantee product quality and readiness of the different functions in the organization to launch a new product, and
- enable timely business decisions on project continuation or termination based on project status and market criteria.

Controlling projects by gates

A gate is a decision point in a project where those who are responsible for the outcome of the project evaluate the achieved results from a business point of view and determine whether to continue the project or not. A decision to continue may of course include alterations to the project such as changed scope or plan.

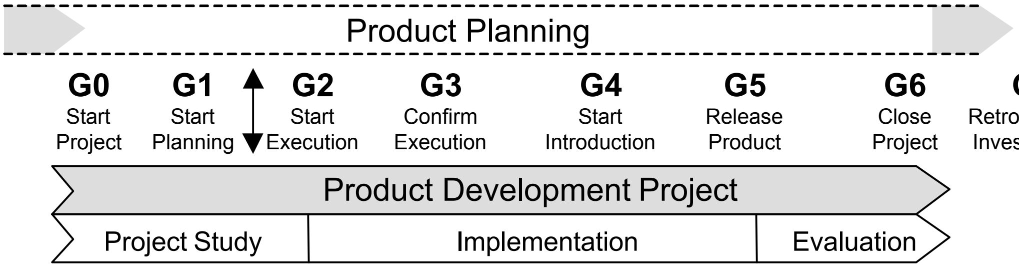

Fig 5: The ABB Gate Model

At each gate, the status as well as the business opportunities and risks for the project are discussed in a gate meeting. To keep the gate meetings focused on the business decisions, the ABB Gate Model includes a gate assessment before each gate. Figure 5 depicts the Gate Model in the context of product development projects.

It is important to understand that the Gate Model works on the top level of the product development project. A product development project typically consists of different parallel activities executed by different functions in the organization (e.g. marketing, sales, development, service, training,

quality management).In the Gate Model the progress and readiness of all these functions is checked and the feasibility of the project is assessed taking all these functions into account. Thus, the Gate Model is a decision model for the whole product development project, while a software development process is a technical model, which covers only the software development part.
As shown in Figure 5, a product development project is divided into three phases: the project study phase, the implementation phase and the evaluation phase. Gates are used to control the progress during theses phases.
The gates serve the following purposes.

- **Gate 0: Start Project**
  Initiates the project study phase. The focus between G0 and G1 is on analysis of the requirements.
- **Gate 1: Start Planning**
  Defines the scope of the project. The requirements agreed here will control the planning made between G1 and G2.
- **Gate 2: Start Execution**
  Marks the agreement on requirements, concept, and project plan. The focus from G2 to G3 is on specification of functions and architecture.
- **Gate 3: Confirm Execution**
  Confirmation that target dates can be met and that the project executes according to the project description and plan. After G3, the focus is on implementation.
- **Gate 4: Start Introduction**
  Release for acceptance testing. Focus is on validation, on preparation for the market introduction and on production preparations.
- **Gate 5: Release Product**
  Hand-over of the results to the line organization. G5 indicates also that the project activities should be finished and focus in the period to G6 is on finalizing any remaining issues.
- **Gate 6: Close Project**
  The project is terminated.
- **Gate 7: Retrospective Investigation**
  A follow-up of the project to check if the results are satisfactory, and to feed back experiences to the organization.

Staffing a gate oriented decision model

In order to apply the Gate Model in development projects, the corresponding activities (e.g. planning the gates, performing gate assessments) have to be carried out. For this end the Gate Model introduces the following roles.

- **Project Sponsor**: The manager responsible for the development and maintenance for the affected product lines. The project sponsor should have the authority to start and stop the project, i.e. have influence on the product portfolio and the economical power to increase or decrease the funding of the project.
- **Gate Assessor**: The gate assessor has the overall responsibility to report the status of the project that indicates whether the project is ready to pass a gate or not. The gate assessor is typically from QA, an external assessor or the project sponsor. It is important that the gate assessor is objective.
- **Gate meeting participants**: The gate meeting participants appointed at G0 will have the responsibility to assist the project sponsor in evaluation of the project at the gate meetings. The participants at the G1 – G7 meetings should be the responsible managers for development quality, sales, marketing, service and training as well as the project sponsor, product manager and the project manager.

The Gate Assessment Process

The ABB Gate Model requires each project to perform gate assessments to prepare gate decisions. The gate assessment reviews the status of the project and prepares the relevant information for the gate meeting. In order to prepare the gate assessment report checklists containing the central aspects to be assessed are used.

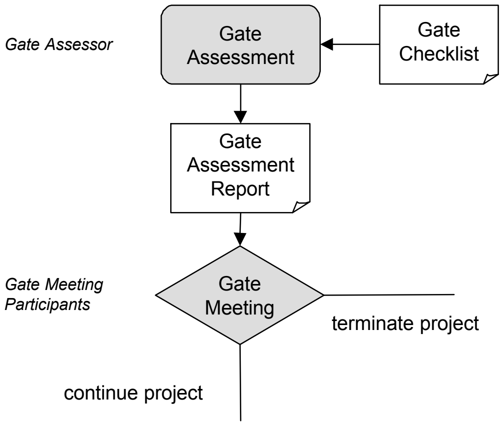

Fig 6: Gate meeting procedure

The manager responsible for the affected products (the Project Sponsor) initiates the gate process in collaboration with the project manager. The first step is to perform an assessment and to prepare the gate decision material. The input to the assessment is documents prepared by the project and gate assessment checklists. The assessment is typically done over an extended period and involves the project manager and a gate assessor. The output from the assessment is signed checklists and an assessment report.
Based on the assessment report, the decision to

continue (possibly with changes) or terminate the project is made at the Gate Meeting.

5 Implementing the Gate Model

The implementation process

To get the most value of the Gate Model, every R&D unit has to incorporate the Gate Model into its own processes. To implement the ABB Gate Model there are a number of concrete steps that should be performed. By experience these should be taken in the following order:

1. Sort out the structure in R&D management regarding how decisions are made and map the ABB Gate Model roles to the organization.
2. Decide what project types the ABB Gate Model should be applied to. Decide what project "size" factors that should be the limits for use of the Gate Model.
3. Make the gate checklists complete regarding extra checkpoints and document references.
4. Decide how metrics should be collected and reported.
5. Run a pilot project.
6. Decide on how the Gate Model should be rolled out and roll it out.

In order to support the business units to introduce and tailor the Gate Model a one-day training course is available. The course is intended for line managers and project managers. It explains the Gate Model and includes several small exercises and discussions on how it fits in the software process culture of the R&D unit addressed. Typically, this course forms the starting event in the local gate model implementation process.

Experiences

The common ABB Gate Model as described above has been piloted since 1999 and officially rolled out throughout the company in March 2000. It is today used by 1200 software developers in ABB's critical R&D units. In these two years, we have discovered the following benefits over earlier ABB models and models from the literature:

- **Common language.** The mandatory elements of the Gate Model (numbering and names of the Gates, main items in the checklists) establish a common culture in the company. Everyone in the R&D units knows what "G2" means, and which main deliverables have to be prepared for that gate. This commonality enhances cooperation and gives better visibility of the status of critical product development projects.
- **Clear separation between Decision Model and Development Model.** The gate model is the interface between the development project and the product line organization. For the product organization, it is important to understand, if they will make profit with the project result and if the progress of all parts of the project is sufficient. The Gate Model provides this information at critical points of the development project without burdening the gate meetings with the intricacies of the development process. This is particularly important for incremental development models, like the Rational Unified Process. In an incremental project, there will be technically motivated internal iterations, which are not visible in the Gate Model. Other iterations will create a product that has to be maintained by the organization. These external deliveries are visible in the Gate Model, and a Gate 5 is required before the product becomes operational at a customer site.
- **Business and quality focus.** Early (local) versions of the ABB Gate Model interpreted the gates mostly from a quality assurance perspective and did not explicitly address the business perspective of the project. The gate meetings were often restricted to checking the existence of documents and discussions of technical problems. These deficiencies have motivated the current gate process, which separates assessment from gate meeting, and leads to shorter, more focused gate meetings, and more conscious business-oriented go / no go decisions.

In the Gate model introduction projects, we have further found the following critical success factors:

- **Staffing of the sponsor role.** Since the sponsor is ultimately responsible for the business decision, it is important that the sponsor is known and has the competency to make this decision. To identify the sponsor for a project is an important and non-trivial step in a large, matrix-structured enterprise.
- **Training Course.** The one-day training course, which consists of lectures as well as two simulated gate meetings has proved essential. We have observed, that R&D units, who have simply downloaded the Gate Model from the EDB without training have suffered from misinterpretations. Without training, the decision making process of the Gate Model can be misunderstood, or the gate model can be mistaken as a sequential development model instead of a decision model.

6 Conclusions, Lessons Learned and Outlook

In this paper we have presented ABB's approach to improving its software product development processes. While the focus of former SPI initiatives was to improve local processes as well as technological skills, ASPI is a company wide initiative focussing on product and business driven improvements. Although ASPI is not completed, the following can be summarized:

Because ASPI is a top-level management driven program it is visible throughout the company and has been recognized as an important endeavor. This is in our opinion a prerequisite to develop and implement common processes in a multinational company. Moreover, this initiative has proven to canalize and to complement local SPI activities.

We can identify two essential success factors of our initiative.

- First, there is a clear separation of common management processes that have to be applied at all units, and local technical processes that are under control of each unit. For this reason the common processes (e.g. the Gate Model) are defined in a generic way with hot spots for local tailoring and interfaces to local processes. Tailoring guidelines help to implement these processes in the local environment.
- Second, the chosen SPI organization (ASPI, the ABB Group Processes responsible unit and the local SPI organization at the business units) guarantees that both the common processes are introduced and used and that experiences and good practices are collected and analyzed.

Another important aspect of this SPI initiative is that at first product and business oriented processes are covered (by means of the Gate Model). Thereby the product development management was involved right from the start in the process definition. This raises the awareness on the importance of software and the necessity to invest in SPI activities as well as the acceptance of the resulting common processes.

After the common Gate Model is defined, approved and introduced we are currently working on a common Project Management Model supporting project managers to run a project according to the ABB Gate Model and to provide the information required for the Gate Decisions. It introduces a common terminology for project management activities throughout ABB and establishes company-wide procedures for project steering. Additionally, we plan to provide a project management toolbox containing small but useful tools supporting management activities.

References

Basili, V., G. Caldiera, D. Rombach (1994): The experience factory. In Marciniak (ed.) Encyclodedia of Software Engineering, vol 1. John Wiley & Sons, S. 469-476.

Cooper, Robert G. (1993).: Winning at New Products: Accelerating the Process from Idea to Launch, 2nd edition. Perseus Press, 1993.

Debou, C., A. Kuntzmann-Combelles (2000): Linking Software Process Improvement to Business Strategies: Experiences from Industry, Software Process: Improvement and Practice, No 5, pp 55-64.

Houdek, F., K. Schneider und E. Wieser (1998): Establishing experience factories at Daimler-Benz - An experience report. In Proceedings of the 20th International Conference on Software Engineering (ICSE) 1998, IEEE Press, Pages 443-447.

Komiyama, T., T. Sunazuka, S. Koyama (2000): Software Process Assessment and Improvement in NEC – Current Status and Future Direction, Software Process: Improvement and Practice, No 5, pp 31-43.

Lichter, H, C. Welsch, M. Zeller (1995): Software Process Improvement at ABB Kraftwerksleittechnik GmbH, In P.Elzer, R.Richter (eds.) Proceedings of MSP'95 Experiences with theManagement of Software Projects, Karlsruhe, IFAC Conference Proceedings, Elsevier.

McFeeley, Robert (1996). IDEAL: A User's Guide for Software Process Improvement (CMU/SEI-96-HB-001, ADA 305472). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

Paulk, M., B. Curtis, M. Chrissies, C. Weber (1993): Capability Maturity Model for Software, Version 1.1. Carnegie Mellon University.

Thomas, M., F. McGarry (1994): Top-down vs. bottom-up process improvement, IEEE Software, 11(4), pp 12-13.

Welsch, C., H. Lichter (1997): Software Process Improvement at ABB - Common Issues and Lessons Learnt . Proceedings of Software Quality Management SQM 97, Bath UK, March 1997.

Wohlwend H., Rosenbaum S. (1994): Schlumberger's Software Improvement Program. IEEE Transactions on Software Engineering (20), No. 11, S. 833 - 839.