IESE

Fraunhofer Einrichtung
Experimentelles
Software Engineering

The PERFECT Handbook

Volume 4
Application of PERFECT

Perfect

Authors:
Andreas Birk, Peter Giese, Ralf Kempkens,
Dieter Rombach, Günther Ruhe (Eds.)

In part supported by CEC
through ESPRIT Project
9090 "PERFECT"

Has appeared also as
PERFECT technical reports

IESE-Report No. 062.97/E
Version 1.0
December 1997

A publication by Fraunhofer IESE

Fraunhofer IESE is an institute of the
Fraunhofer Gesellschaft.
The institute transfers innovative software development techniques, methods and tools into industrial practice, assists companies in building software competencies customized to their needs, and helps them to establish a competetive market position.

Fraunhofer IESE is directed by
Prof. Dr. Dieter Rombach
Sauerwiesen 6
D-67661 Kaiserslautern

Preface

Motivation

Software organizations need to continuously improve their product development processes in order to fit the changing needs and characteristics of the software business. Therefore, every software organisation needs to identify and further develop its software engineering core competencies. An improvement strategy should be built on principles of organizational learning. An important prerequisite for the success of improvement programs is that they are planned and executed systematically. Empirical control is necessary to track and manage changes to software development practices.

PERFECT Improvement Approach (PIA)

ESPRIT Project 9090 “PERFECT” has developed a measurement-based improvement approach that can be tailored to company-specific needs and characteristics. It integrates various operational methods for specific tasks of systematic process improvement as well as tool support for these tasks. The “PERFECT Improvement Approach” (PIA) is based on the Quality Improvement Paradigm (QIP) / Experience Factory (EF) [1] and the principles of continuous improvement and organizational learning in the software domain. Important underlying infrastructure technologies are goal-oriented measurement, explicit modeling, and comprehensive reuse.

PERFECT Handbook

This report is an edition of the main result from PERFECT, the PERFECT Handbook. The PERFECT Handbook consists of twelve parts, so-called “booklets”. They are summarized in four volumes as a series of Fraunhofer IESE technical reports. The list of contents is presented in the table below.

Volume	Part
1 Improvement Methodology	Handbook Introduction
	PERFECT Improvement Approach
	PIA Experience Factory: The PEF Model
2 Infrastructure Technologies	Goal-Oriented Measurement Using GQM
	Process Modelling
	Experience Base
	Impact Analysis
3 Process Enactment	APEL - Abstract Process Engine Language
	A platform for supporting the PERFECT methodology
4 Application of PERFECT	Launch of the Experience Factory
	A PIA Scenario
	Experiences with PIA

Volumes of the PERFECT Handbook

The volumes of the PERFECT Handbook address "Improvement Methodology","Infrastructure Technologies", "Process Enactment", and "Application of PERFECT". "Improvement Methodology" describes the basic principles and core elements "PERFECT Improvement Approach" (PIA). "Infrastructure Technologies" addresses core competencies that a software organisation should have in place for the successful conduct of an improvement program. "Process Enactment" refers to tool support for the execution of software processes. "Application of PERFECT" introduces recommendations, explanations, and experiences for the set up and performance of improvement programs according to PIA.

This volume

This volume on "Applications of PERFECT" contains the three handbook parts that address issues of applying the PERFECT Improvement Approach. "Launch of the Experience Factory" describes the process of setting up a PIA based improvement program. "A PIA Scenario" illustrates the course of a PIA based improvement program using an example case. "Experiences with PIA" summarises the experiences of the PERFECT application projects from applying PIA.

PERFECT consortium

The PERFECT project started in October 1993 and ended in November 1996. The consortium consisted of CAP Gemini Innovation (France), Daimler Benz (Germany), University of Grenoble / Laboratoire LSR (France), Q-Labs (Sweden), Robert Bosch (Germany), Siemens (Norway), SINTEF (Norway), and University of Kaiserslautern (Germany). Fraunhofer IESE has taken over past-project exploitation from University of Kaiserslautern.

Handbook contributions

The PERFECT Handbook is a joint publication of the PERFECT consortium. Most Handbook parts are result of close collaboration of multiple consortium members, while one partner had particular responsibility. University of Kaiserslautern had main responsibility for the two Handbook parts "PERFECT Improvement Approach" and "Goal-Oriented Measurement Using GQM". They have been written in collaboration with Q-Labs.

PERFECT tool platform: APEL and GQM*aspect*

In addition to the handbook, PERFECT has developed the tools "APEL" and "GQM*aspect*". APEL is a process modeling and enactment environment that consists of process modeling language and editors, process model compilers and interpreters, as well as a process enactment platform. APEL has achieved a new and outstanding level of integration of process, product, and measurement concepts that has not been provided prior to PERFECT by any other software engineering environment [2]. GQM*aspect* supports the planning phase of GQM-based measurement programs. It is a platform-independent tool that offers functions for editing and generating GQM abstraction sheets and GQM plans. It also facilitates the reuse of GQM plans from previous measurement programs [3].

PERFECT web pages and online PIA scenario

Continuously updated information about PERFECT is provided through the PERFECT web pages [4]. They provide online access to PERFECT results and offer the interactive web-based PIA scenario. The PIA scenario illustrates and teaches the course of PIA-based improvement programs. In particular, it illustrates the interaction of QIP steps on organizational level and project level with the Experience Factory [5].

Usage of PERFECT results

Results from PERFECT are used in several other initiatives and research projects; among them are ESPRIT project "PROFES" [6], German government funded project "SoftQuali" [7], Fraunhofer IESE's Spearmint project [8], and several improvement programs at industrial software organisations (e.g., [9]).

Kaiserslautern, December 1997

Andreas Birk
Peter Giese[1]
Ralf Kempkens
Dieter Rombach
Günther Ruhe

References

[1] Victor R. Basili, Gianluigi Caldiera, and H. Dieter Rombach. Experience Factory. In John J. Marciniak, editor, *Encyclopedia of Software Engineering*, volume 1, pages 469–476. John Wiley & Sons, 1994.

[2] J..Estublier, S. Dami, A. Amiour. *APEL: A Graphical Yet Executable Formalism for Process Modelling.* Automated Software Enginnering (ASE), March 1997.

[3] Markus Hoffmann, Andreas Birk, Frank van Els, Ralf Kempkens. GQMaspectV1.0: User Manual. Fraunhofer IESE Technical Report No. 020.97/E, 1997.

[4] http://www.iese.fhg.de/Perfect

[5] http://www.iese.fhg.de/Services/Projects/Public-Projects/Perfect/PIA-Scenario/welcome.html

[6] http://www.ele.vtt.fi/profes/

[7] http://www.iese.fhg.de/Services/Projects/Public-Projects/Softqual.html

[8] http://www.iese.fhg.de/Services/Projects/Internal-Projects/Spearmint.html

[9] http://www.iese.fhg.de/Services/Projects/Industrial-Projects/Allianz.html

1 Peter Giese is now with Kiefer&Veittinger GmbH, Mannheim

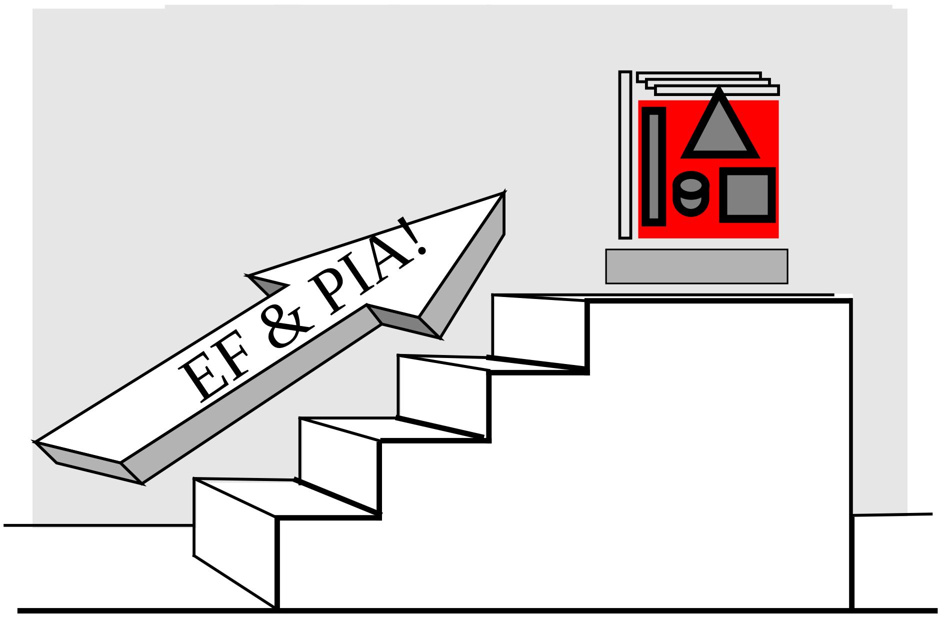
A booklet from the

Perfect

ESPRIT project 9090

Handbook

Launch of the Experience Factory



 D-BL-Launch-2-PERFECT9090

Table Of Contents

1. Executive summary

The Experience Factory according to the PERFECT Improvement Approach adds new value to your software organization. The Experience Factory supports organizational learning in the context of software processes and in software development. However, the added value of the EF depends upon the perspective:

- The EF as an *operational tool for projects*, supporting them to reach their goals more efficiently.
- The EF as a *tactical tool in the short term evolution of an organization*. New technologies for software development are systematically investigated and made to practices within the organization, in order to meet increased market demands.
- The EF as a *strategic tool for an organization*, i.e. to support strategic planning, design, implementation and management. The artefacts related to the organizations software processes and the artefacts related to the management of these, thus become important strategic components.

The growth towards a more strategic culture/attitude in an organization, implies a major change in the infrastructure, the organizational climate, leadership styles etc. This will require new and different capabilities/skills on all levels in the organization. Managers capability and skill profiles must evolve as well as the profiles of the software development staff. Furthermore we expect to see a number of new roles.

The full implementation of an EF thus imply a major undertaking in terms of changing the organizations internal state of affairs. This cannot be done in one go. Consequently we propose to implement the EF in several steps according to the PEF Capability Growth Model. We launch a vision for the EF in the long run, but proceed in steps which can be practically grasped. In this way the EF is successively integrated into an existing software business.

Obviously the introduction of an EF in a software organization is a great challenge and implies hard work. For that reason, we end this summary with the following recommendation:

Think big, but start small. Start a vision process (think big) in order to clarify the actual purpose of introducing the PIA in relation to your own business. Then focus on projects (start small).

2. Introduction

For your convenience, we start with some frequently used abbreviations

PIA = PERFECT Improvement Approach

PEF = PIA Experience Factory

EF = Experience Factory[1] organization according to PEF model

1. The concept “Experience Factory” refers to the works of Victor Basili et. al. [BASI94].

2.1 Purpose of this booklet

To anyone who is in charge of a software business or a software process improvement program:

This is a booklet with the intention to inspire the start-up of an improvement programme according to the PERFECT Improvement Approach. The booklet cover two major topics.

First of all we consider the strategic business value of implementing an experience factory. New and alternative ways of understanding the experience factory concept and the purpose of it's use are presented.

Secondly we present a short framework for how to launch the EF. A model is presented to support the successive integration of an EF into an existing software development organization. Since every business is characterized by their own situation, we only give hints on possible pathways rather than present ready to use solutions.

It is recommended to have some basic insights in the PEF before reading this booklet, see [PEF] chapters 1 - 3.

2.2 Problem statement

This document address the following three questions

What is the motivation for an EF from a business perspective?

What are the main considerations for starting up an EF?

How can the EF be introduced in increments?

2.3 Context

This booklet is about to launch the PIA in a software organization. The focus is however on the experience factory part of the PIA. It is however of importance to view the experience factory in the context of the whole organization. We use the organizational model of the PIA as a reference model, see figure 1. Note the following:

PIA is defined by the PEF and the tool platform.

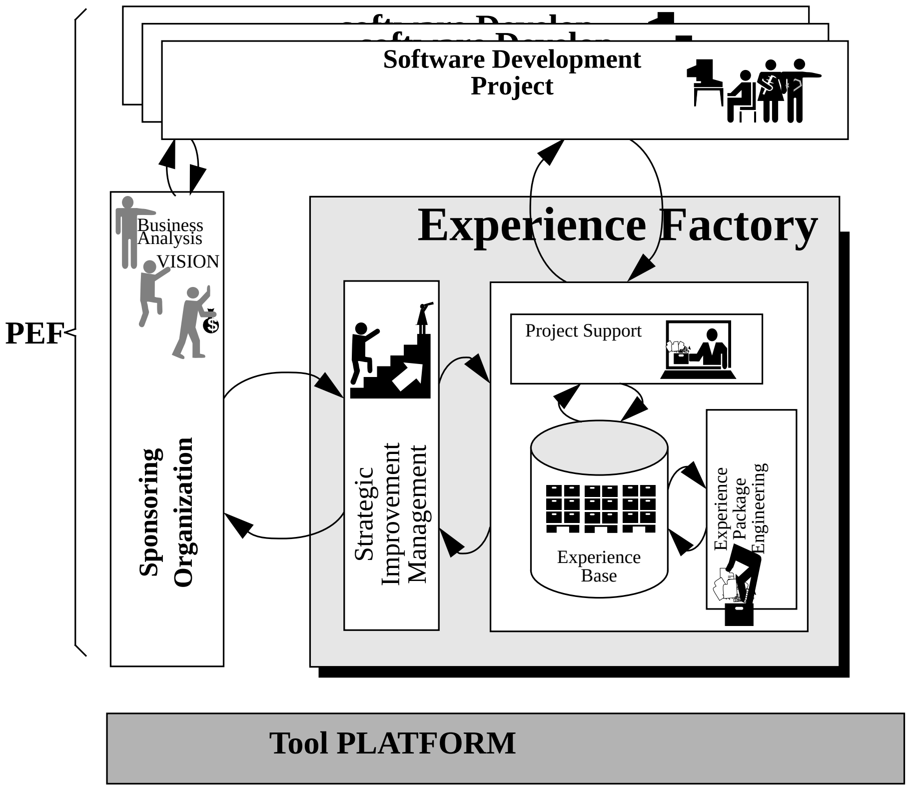
PEF is defined by the Experience Factory, the Software Development Projects, and the Sponsor Organization.

The Sponsor Organization is responsible for the business as a whole and must among others develop business plans, set long-term goals, supply resources for the EF and perform adequate follow-up. The development projects on the other hand have the more operational contact with the EF, and thus reuse software processes and experience of their use.

The EF is the organizational unit for maintaining the organizations software processes, supporting their use and improving them based upon the collection of experience.

The emphasis on the way the EF operates in relation to the software development organization, lies in the word *support*. software projects know their job and have experience. The EF makes this knowledge and experience systematically available to the organization. Furthermore, the EF operates to make the improvement work efficient *in cooperation with the projects and the Sponsor Organization*.

Figure 1: PIA model: PEF + Platform



3. Motivating the EF from a business perspective

3.1 Overview

Through the EF, the software organization will be able to increase its awareness of the software processes and the implications these have on performance (business as well as technical). The software processes and the PEF support processes, thus become important variables in the short and long term perspective of an organization's business development.

The basic motivation for introducing the EF varies. We can however identify three basic perspectives related to the EF and its application in a software organization:

Operational perspective: The EF supports ongoing and close in future projects to make best use of the organizations software processes.

> The EF will support the organization to operate according to the best practises so far. This is an invaluable aspect of not letting the organization to degenerate from a position it has reached through improvement programs.

Tactical perspective: The EF will support the organization to reach its short term business goals (about 1 - 3 years).

Given the current business strategies and core product technologies, the EF will support to achieve the stated business goals through improving the organizations software processes. New technologies for software development are systematically investigated and made to practices within the organization, in order to meet increased market demands. See further [PEF] chapter 3.1.

Strategic perspective: The EF is an important component in the formulation and implementation of a competitive strategy.

Business strategic forecasts give information about the future market demands in terms of new products, increased performance, shorter life-cycles etc. The EF serves as a research-tool to perform investigations in new process technologies to meet these demands pro-actively.

Furthermore, the EF gives strategic feedback on the feasibility of different business strategic alternatives from the perspective of software processes. The EF can also be supportive in the implementation of new or changed strategies.

In this chapter we will elaborate on the motives for an EF from the strategic perspective. The tactical and operational perspectives are well covered in the PEF booklet.

3.2 About making software processes explicit

Through the EF, the software processes and the related artefacts are made **explicit**. Whatever aspects of the software processes are made explicit, these can be looked upon, understood, evaluated and thus be used in a conscious way to support an organizations business evolution. The following list serves as examples in this context:

- The EF is a physical organization, with resources dedicated to activities that concern software processes and the use, the maintenance and evolution of these.
- The EF experiments with new process technologies. In this way the EF produce visible results towards the business planning part of the organization. The organization will have better data to support decision making in important business issues.
- The EF, in cooperation with the project organization, will set the standard of the current process technologies and their use. The EF will be the storehouse of the best practises in relation to software development.
- The EF operates as an active change agent when introducing new software practises in the software development organization.
- The EF creates a visibility of software processes issues in relation to product issues. The business potential lies in the explicit coordination of strategic software process management and strategic software product management.
- The EF activities are based upon testable and validated data. This data is

a foundation for the interpretations packaged as visible “experiences”, i.e. experiences from the use of software processes given a certain context.

- The EF is funded to develop and maintain a platform of tools that support projects in the use of the software processes. Furthermore the EF evaluates vendors tools and integrate these in the organization when feasible.

3.3 Strategic considerations of managing software processes

In this section we will discuss the EF as a key factor in the formulation of the competitive strategy of an organization. In terms of software processes and their management, we want the EF to justify the following statement by the chairman of the General Electric Company

Trying to understand where we will sit in tomorrow’s world, and not where

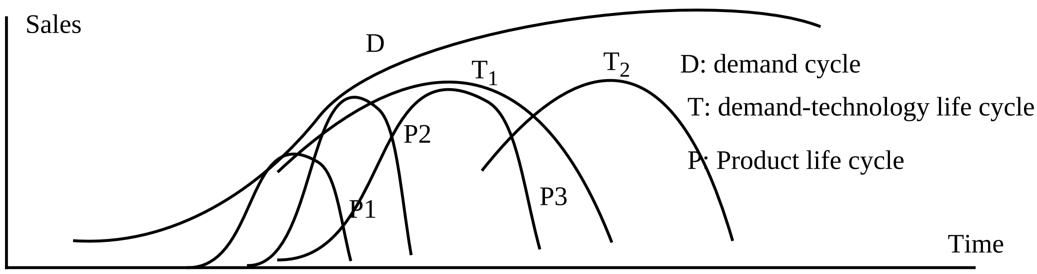
- Trends in organizational styles: structure, culture, management style.
- Structural change of a business sector.
- Diversification of the current business.

In the software intensive industry these demands are directly or indirectly related to different aspects of the software process (experience, technology, process models etc.). The EF plays the role of making these aspects explicit and evaluated against business performance. In this sense the EF makes the software processes manageable from a business perspective.

3.3.2 The need for a competitive strategy

The software business is a technology intensive branch of industry. We conceive that the demand, product and technology life cycles get shorter with time. This implies that at some time in the life-span of a company, discontinuous changes in products and even technologies are likely to be necessary in order to stay or grow in business. New strategic business areas must be entered, and others left. See figure 2.

Figure 2: Triple life-cycle: demand-technology-product.



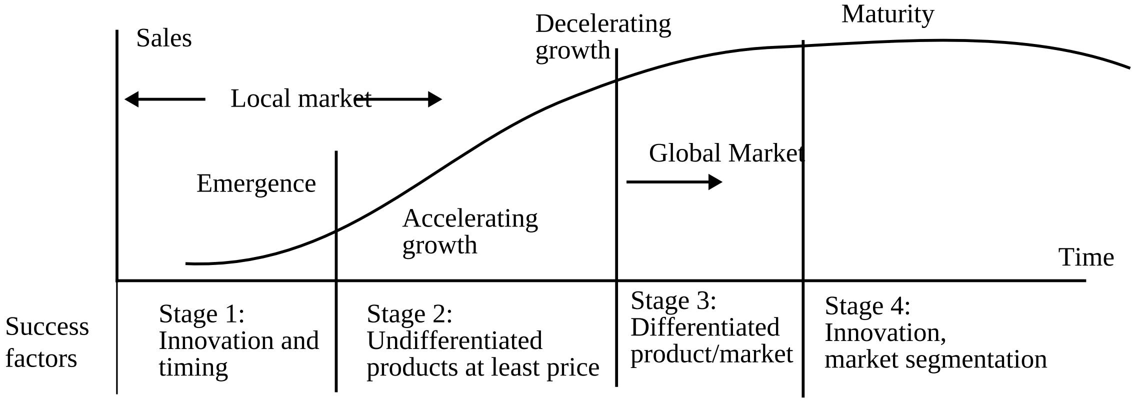
For the software industry it is thus important that a number of technology variables are included in the strategic design process. Essential variables in our context are the software processes, the software management processes and their relation to the mentioned life-cycles.

Moreover, if we look into the details of a typical evolution of a competitive strategy, we see how the success factors of a strategic business area change with time. The challenge to management is to anticipate transitions in the life-cycles stages and to revise the company's strategy accordingly.

An example of this is e.g. when a company expands it's business from a local to a global basis. The local market may have come close to saturation, while novel market places are still in the earlier stages of accelerating growth. Figure 3 illustrates this example and point out how the success factors change with time (see bottom line of the figure).

Now the interesting question is how the EF can support a company in the formulation, implementation and evolution of a competitive strategy. This is the issue of the next paragraph.

Figure 3: Typical stages of an example demand cycle



3.3.3 Formulation of a competitive strategy: a starter

How should the EF operate in order to support a company in the formulation, implementation and evolution of a competitive strategy?

The answer is highly related to the environmental conditions and can be understood through the Strategic Success Hypothesis[1] [ANSO90] chapter 2.1.2, which states the following:

A company's[2] performance potential is optimum when the following three conditions are met:

(1) Aggressiveness of the company's strategic behaviour matches the turbulence of its environment.

(2) Responsiveness of the company's capability matches the aggressiveness of its strategy.

(3) The components of a company's capability must be supportive of one another.

In this hypothesis we find the critical parameters which are essential in the formulation of the business strategy of a company. The parameters are:

Turbulence of the environment is reflected in the time-frames of the demand/product and technology life cycles.

Aggressiveness tells us something about a company's extrovert behaviour on a market, i.e. the degree of discontinuity which a company introduces into succeeding generations of its products and technologies.

The *responsiveness* on the other hand is a parameter for how a company responds to environmental changes, i.e. changes which the company is prepared to perceive, accept and is capable of handling.

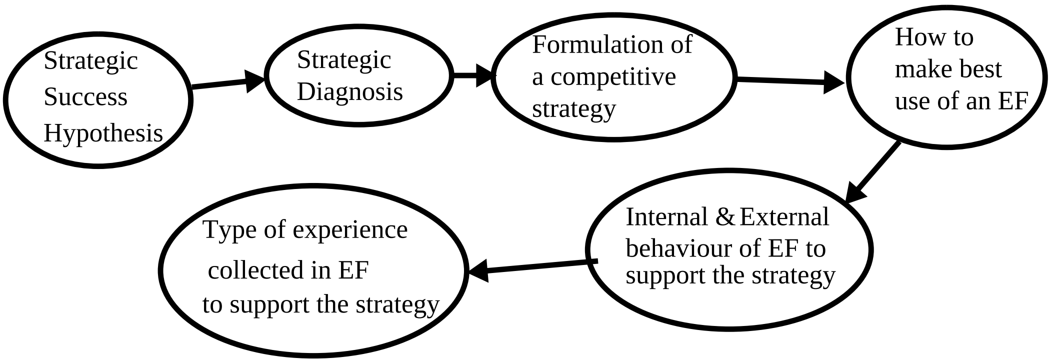
By *components* is meant skills and mentality of managers, culture of the company, power structure, etc. Important components in a software business can e.g. be "Strategic product management", "Strategic process management" and

1. The hypothesis has been validated by empirical research.
2. The word "company" is used in relation to having or developing a competitive strategy. "Organization" is used otherwise in a more general meaning.

Experience Factory.

An assessment of these parameters will give us important information about the way an EF should operate from a business perspective. The assessment procedure is named “Strategic diagnosis” [ANSO90] chapter 2.1.5. To clarify, we illustrate the sequence of derivation in figure 4.

Figure 4: Derivation of EF from a business perspective



From a general point of view (not software specific), the idea is that a successful business should strive to have a matching triplet of aggressiveness, responsiveness and turbulence. We thus need a scheme with attributes according to which we can classify these parameters.

One simple example of such a scheme is found in table 1, where three levels of turbulence are matched with the aggressiveness and responsiveness parameters[1].

Following this example, it is likely to find a software company to operate in the region of level 2-3. For such a company we conclude that in order to work at its optimum, it must be a fore-runner by seeking new change instead of adapting to change. Furthermore the company must consider discontinuous change in its products, competitive environments and marketing strategies.

The actual set-up of these parameters combined with the wanted set-up of these parameters give essential information about the way an EF should operate in an optimum way. However what this means in detail in a specific case can only be understood from the actual situation. Some examples will be given in the next paragraph.

1. The details of such a scheme and the proper procedure for how to make the diagnosis must be worked out thoroughly. Some guidance for this work is given in [ANSO90].

Table 1:

Environmental turbulence	**1 Changing**	**2 Discontinuous**	**3 Surprising**
Strategic Aggressiveness	**Anticipatory** Incremental change based on extrapolation	**Entrepreneurial** Discontinuous change based on expected future	**Creative** Discontinuous change based on creativity
Responsive-ness of Capability	**Marketing** Seeks familiar change	**Strategic** Seeks new change	**Flexible** Seeks novel change

3.3.4 Making use of the EF from a business perspective

The EF in itself is a strategic tool for management to better align the company's aggressiveness, responsiveness to the environmental turbulence. To give some examples, this means that the PIA/EF concept can be used to

(1) Give strategic feedback in the formulation and planning of new competitive strategies for a company. An example of this would be when a company *plans* to enter a new strategic business area, e.g. a producer of Telecom equipment going from public to mobile telecommunications.

The efficiency of this kind of transitions depend on the type of experience the company has collected so far in its history. Which experience is relevant to pass over to the new situation, which aspects are currently unknown?

(2) Support to implement new strategies in a company or an organization, i.e. put new strategies into operations (w.r.t. software processes). An example of this would be when a company has *decided* to enter a new strategic business area. The timing and the implementation aspects of the anticipated changes must be considered.

(3) Drive the design of new strategies for software development based upon the investigation in new breakthrough technologies. In this way a company will be able to impact its current business through higher performance (PQTC). (Partly this area belongs to the tactical aspect of the EF).

(4) Technology intensive industries depend to a large extent upon research and development. The EF is a research platform w.r.t. to the software processes, their performance and implications on business. Through the EF a company is able to create a balance between research and development, where the latter is normally the dominant aspect.

(5) Support a company in its geographical growth, e.g. on a global basis. The experience collected in earlier stages of the demand cycles (1 and 2 in figure 3), will make the transition more efficient.

3.3.5 Example on performance improvement

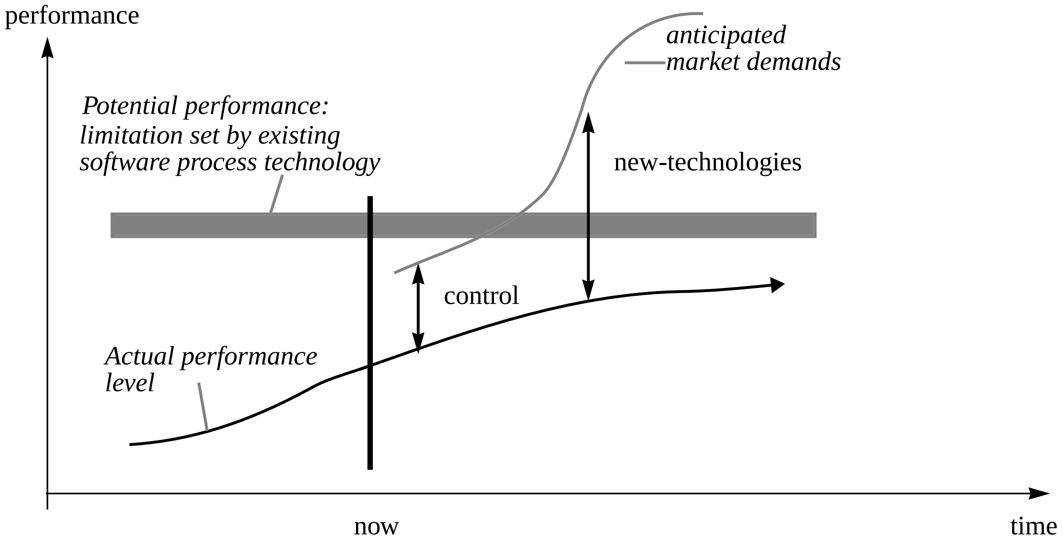
In this short example we discuss an organizations ability to have control over its *performance* evolution. The example shows the importance of having the EF to operate in a business context. The EF is thus not only an internal matter for an organization or its projects.

We want the example organization to be able to meet the increased market demands in a pro-active way. The pre-requisite is however that the organization is aware of its *actual* performance level and its *potential* performance level. This knowledge is possible to have through the EF, based upon the collected experience.

In figure 5 we describe two possibilities to meet increased market demands. Up to a certain limitation, quality control can be used to improve (control in the figure). The limitation is set by the existing process technology (horizontally shadowed region). In the case that the demands are expected to be higher than this limitation, the infusion of new technology must be considered (new-technologies in the figure).

The important point here is that the organization is able to merge the business context and the technology context. In doing so, the right decisions and actions can be made w.r.t. to the evolution of the organization. The business context in this example is given by the knowledge of the anticipated market demands (Sponsor Organization responsibility). The technology context is give by the knowledge of the actual and potential performance levels, furthermore by the knowledge of the potential of new technologies (Experience Factory responsibility).

Figure 5: Gap analysis of performance demands



3.4 Summary

The technology intensive software industry must not only compete with process performance factors such as time (timing), quality and reliability. In order to stay in business, software companies and organizations are under constant pressure to do new innovations at an increasing pace. Discontinuous changes in products as well as technologies must be anticipated. We also see the need for the diversification of a business to new strategic business areas.

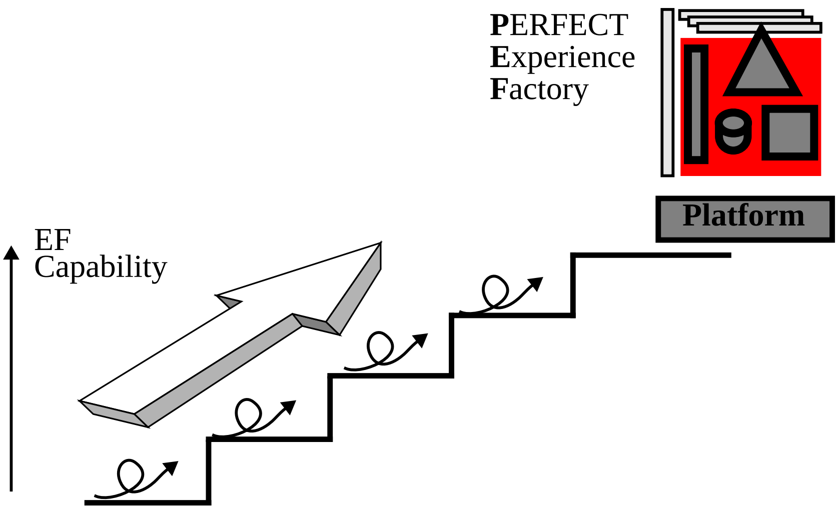
We conclude that the software processes and the experience of using these are important strategic assets. The EF plays the role of making these assets explicit and thus manageable from a business perspective.

4. To Launch the PIA: Implementing the EF

4.1 Overview

The full implementation in the EF implies a major undertaking in terms of changing the organizations focus and behaviour with respect to software development and software process management. For this reason we propose to implement the EF in several steps according to the PEF Capability Growth Model[1].

Figure 6: Stepwise implementation of the PIA



For each step taken we can consider an implementation model according to three phases.

- Pre-Launch, i.e. Platform building in terms of planning, commitment building and sponsorship.
- Launch - i.e. “take-off” according to plans.
- Post-Launch, i.e. implementation through building of skills, capabilities and organizations climate/culture. Evaluation of implementation progress.

In this section we will primarily look in to the Pre-Launch phase. A full picture of the implementation model will not be given, since we can find descriptions elsewhere in the literature. One source of information is [ANSO90], which represents the business perspective of managing strategic change. See e.g. the ‘Accordion method’, a change management process for infusing a new strategy in an company. Another source is Software Productivity Consortium, who defines a cyclical process for implementing change [SPC93]. The SPC-process can be generalized for this purpose.

4.2 Implementation model

The three phases of the implementation model identify the necessary activities for

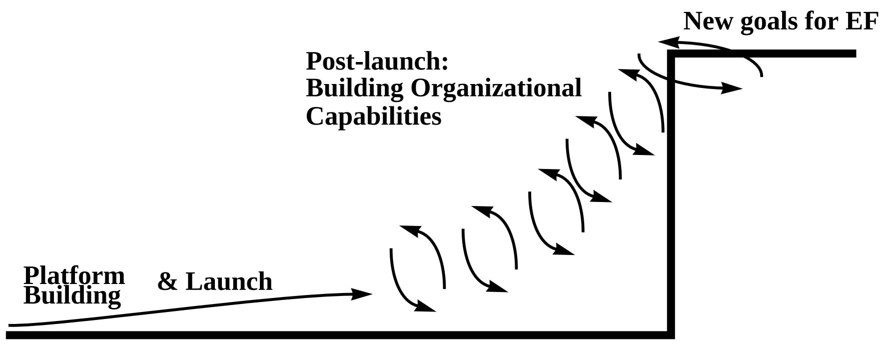
1. The details of the PEF Capability Growth Model are further elaborated in chapter 5. Now we only consider the steps as abstract entities.

evolving the EF one step further according to the goals set (see figure 7).

As indicated in the figure, the hard work does not lie in the actual Launch nor in the Pre-Launch activities. The Launch is more of symbolic act in order to empower and commit the EF and the organization to go for the next stage, while the Pre-Launch is more of a planning and preparation phase.

Why do we say that the Post-Launch phase is difficult? The reason for this is simply that, in this phase the implementation concerns the “real world”. The many circles of the Post-Launch Phase indicate that goals cannot be implemented in just one stroke. There has to be a repetitive action pattern, which allows the organization to successively adopt to new ways of thinking and behaving. These repetitive cycles show up in many ways. E.g. on “the horizontal level”, it is necessary to spread information concerning the improvement programme over and over again in different parts of the organization. On the “vertical level” we see that one project or one person must repeat the same type of activities several times, in order to be able to integrate a “new” piece of knowledge into the existing business.

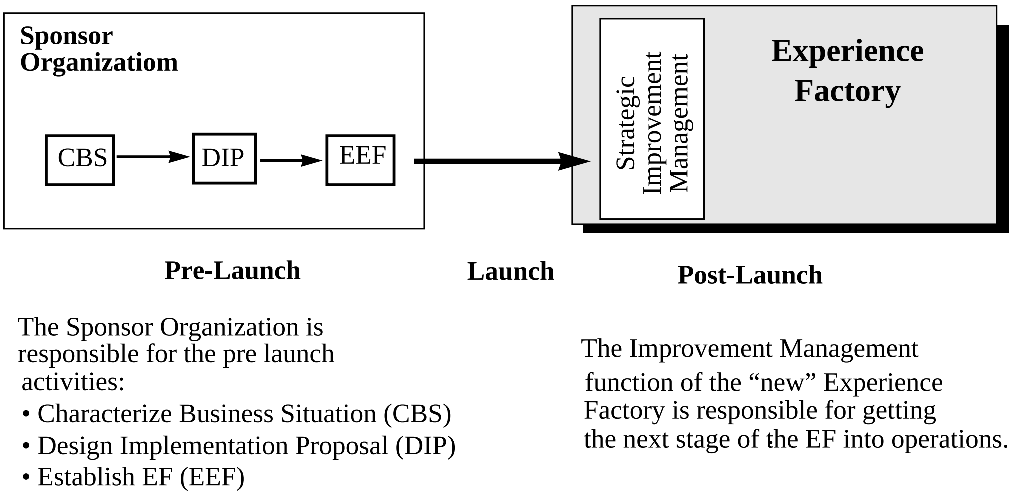
Figure 7: Implementation Model



By mapping the PEF Organization with the three activity phases, we get the organizational implementation view as shown in figure 8.

The Sponsor Organization is responsible for the Pre-Launch phase, i.e. the planning part of the next stage in the evolution of the EF. It is also responsible for initiating the actual Launch. The EF on the other hand is responsible for the Post-Launch phase, i.e. to get the next stage of the EF into operations in relation to the Project and Sponsor Organization.

Figure 8: Organizational implementation view



Note however that it is not responsibility of the EF to build Organizational Capability. The EF rather operates as a facilitator in this respect. The evolution of Organizational Capabilities can only come about when each of the different organizational entities take responsibility individually and together, in order to reach the goals set.

Note also that the conditions for the EF are set by the Sponsor Organization through the Pre-Launch activities as shown in the figure. In turn, the EF sets the conditions for the Project Organization through the EF-function *Strategic Improvement Management.*

We will know proceed with a more detailed guide for the Pre-Launch phase and elaborate on the activities related to the boxes indicated CBS, DIP and EEF.

4.3 A Pre-Launch guide

4.3.1 Overview

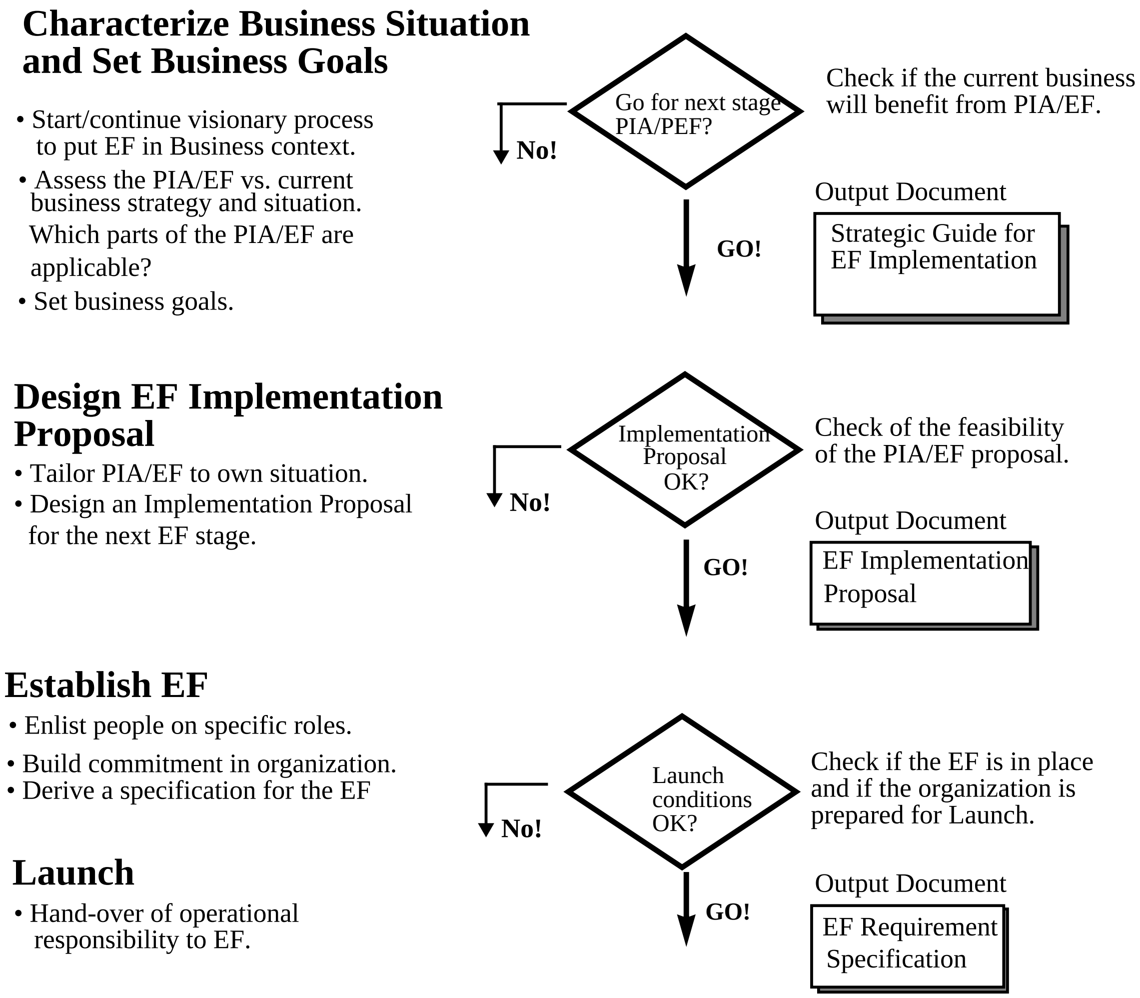
The Pre-Launch model is defined according to three phases (see also figure 9):

- Characterize Business Situation and Set Business Goals.
- Design Implementation Proposal.
- Establish EF (Pre-execution).

Each phase ends up in a decision point where certain criteria must be fulfilled in order to be able to continue with the next phase. The Pre-Launch model represents a generic description for each step in the PEF capability growth model. However the very first time PIA/PEF is considered the whole concept must be judged and planned for, not only "the next stage".

The next few paragraphs elaborates on the details of the Pre-Launch model.

Figure 9: Pre-Launch overview



4.3.2 Characterize Business Situation and Set Goals

The focus of this phase is to decide on the EF from a strategic and business point of view. It is highly related to the strategic success hypothesis, which means that the EF should be used in such a way that the business is developed towards its optimum. The output from this phase is the document *Strategic Guide for EF Implementation*. The document is a conclusion of the following activities:

(1) Start a vision and assessment process in order to align the EF to the prevailing business strategy. In other words, where are we and where do we want to go?

- Assess the current EF vs. the PEF reference model and the PIA in general. What has been achieved so far?
- What is the current business status? Typical parameters are: Business vision, current business needs, challenges and strategies?
- Which business needs or challenges are expected to be addressed by the EF and which are not? Which parts of the PIA/EF are applicable?

- In what way should the EF operate from a business perspective, that support the external state of affairs to be achieved? What does the parameter triplet (environmental turbulence, aggressiveness, responsiveness) mean in this context?
- Is the planned next stage EF in harmony with other company strategies?

(2) Set business goals with respect to the EF.

(3) Perform a simple ROI case study. Cost benefit analysis can be based upon existing, although not fully complete, data.

(4) Other aspects to consider for assessment:

- Need of competency/skill development w.r.t. management, software process and software development.
- The political and cultural climate of the organization is supportive to EF.
- Short term goals implications vs long term goals, i.e. judge the risk of short term goals taking over the business.
- Realities of timing, resistance and power structure.

4.3.3 Design Implementation Proposal

In this phase, the focus is to design an implementation proposal for the PIA initiative in general and specifically for the next EF capability stage. The output from this phase is an *EF Implementation Proposal*, addressing:

- Suggestion for how PIA/EF is to be tailored according to the needs of this particular organizatiom.
- Organizational targets, i.e. departments, projects, roles involved etc. Directives concerning priorities and order of introduction.
- Possible software targets, which relates to products, versions, systems, sub-systems etc.
- Pointing out critical resources that are necessary in order to implement the next stage of the EF.
- Pointing out key positions in the organization for commitment building.
- Guidance for planning containing e.g.: Economical budget, resources, primary milestones and primary deliveries.
- Proposals of evaluation and award mechanisms.
- Definition of success criteria: e.g. related to EF and projects capability/performance.
- Planning of a LAUNCH-session: a symbolic act for commitment and empowerment of the EF and the organization to go for the next stage. The improvement proposal is primarily addressed to the Improvement Management of the PEF model.

4.3.4 Establish EF

Enlist people

- Enlist people to the EF roles as defined by the PEF booklet.
- Enlisting a senior software engineer (especially the first time) to the EF.
- Identify an Improvement Manager responsible for running the EF ("Improvement management").

Commitment building

- Increase awareness in organization through seminars on the concept, directed to projects, software support groups like QA, department managers etc. Presentation of the implementation proposal and leading ideas.
- Secure support from critical management positions.
- Secure the proper resources (people, time, money and support).

EF Requirement Specification

- Develop a requirement specification for the EF Support processes according to the next development stage. It has to be derived from the "Strategic Guide" as well as the "Improvement Proposal" from the previous phases.

4.3.5 Expected roles in Pre-Launch

The question is how to organize the Pre-Launch phases. One suggestion is to make use of high performance teams [POKR95]. The motivation for this is among others:

- The Pre-Launch is characterized by three phases, each one limited in time and driven by specific tasks/goals. The Pre-Launch organization is only temporary and is not an ordinary part of the organization.
- The tasks to be performed require a high degree of cross-competence in order to be successful. We need skills from areas such as management, business analysis, marketing, experience factory and software development.
- The combination of brain power from different people and disciplines leads to higher creativity.
- The EF concept is dependent upon good communication in the organization between the three parties (Sponsor, EF and software development). The use of the team work concept at an early stage, support this.

We choose to describe the roles of the team organization from a general point of view, see figure 10 [POKR95]. We are not specific about how to organize teams in the Pre-Launch phases since every business case has its own requirements. Moreover, the composition of the teams varies in the different phases, since the tasks to be performed are different. The temporary roles are:

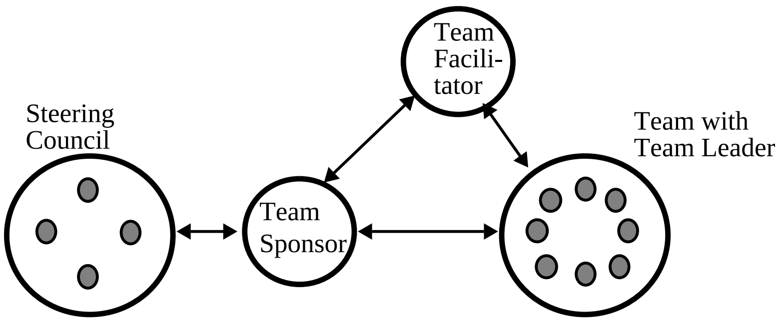
A high performance team is defined by a group of willing and trained individuals who are united around a common goal, sharing responsibility for their task and who are empowered to implement decisions. The team consists of the Team Leader and Team Members.

The Team Facilitator is the group process consultant who coaches the Team Leader, advise the Team Sponsor on team dynamics, trains the team to use team methods and improvement tools. The Team Facilitator is an important factor of speeding up the group work process. The Team Facilitator should be trained in group dynamics and team tools, attend team meetings and meet the Team Leader frequently.

The Team Sponsor should provide for direction and guidance, authorize time and resources, finance the project, committing to follow through and promote team success.

The Steering Council is a model management team, who has the overall purpose clear. Typically the steering council would be staffed by stakeholders in different key positions, to back up and keep track on the overall progress.

Figure 10: Generic team organization



5. The PEF capability growth model

The purpose of the PEF growth capability model is to achieve a stepwise introduction of an EF, i.e. an experience factory according to the PERFECT Improvement Approach. Thus a company or an organization will be able to successively implement a strategic management capability w.r.t. software processes.

5.1 Model overview

We choose to define the PEF Capability Growth Model in the *spirit* of the well known Capability Maturity Model for software [PAUL93]. This choice, at least at a first glance, can easily be misunderstood and will thus need some motivation and elaboration.

What we want to achieve in a business or organization:

- Provide support to the successive evolution of an organizations capability to develop it’s software.
- To develop a capability of strategic change management w.r.t. software processes. The EF is a key component in this context.

What we do:

- We focus on organizational capability. Different parts/aspects of an organization can and will of course vary in their capabilities. Organizational capability is a collective matter and represent an average “value”.

- We use *the main objective of each CMM level* as a guide for what we want the EF to support. The reason is that the successive evolution as described by CMM is natural; starting with a limited scope (project focus) and complexity, and then successively increasing the software organizations capability of software development and process improvement.
  Another way of viewing the situation is that the CMM levels represent successive levels of increased information complexity, which the organization is capable of handling.
- We define a number of successive steps, which each prescribe a capability focus of the EF.
- Each EF capability focus prescribed corresponds to and supports a certain capability focus of the software development organization.

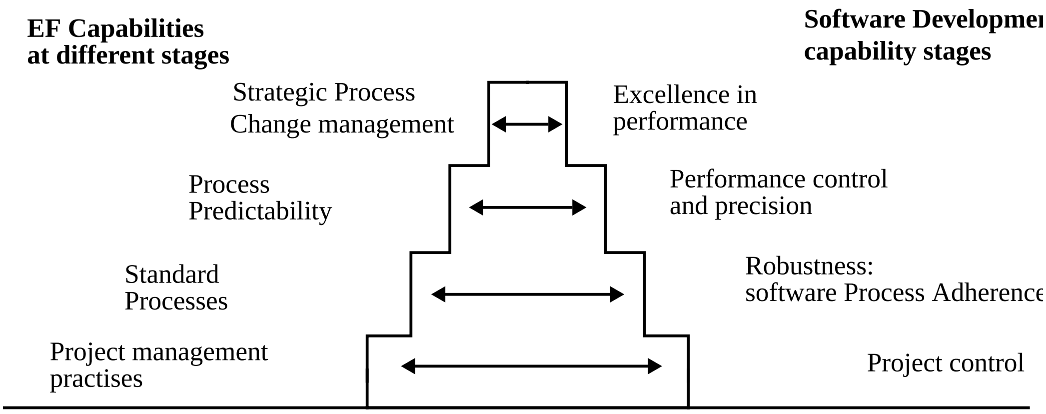
What we don’t do:

- We do not take stand either against or with, the way the CMM is designed w.r.t. the key process areas, their grouping or sequence.

With this basis we are ready to present an overview for how the EF and software development capabilities match at different levels, see figure 11.

If we look in to the right side of figure 11, we find a short characteristic of the capability of the software development organization.

Figure 11: EF Capabilities vs. Software Development Capabilities



The ultimate goal is “excellence in performance”. The starting stage is however to have control over single projects (*Balance chaos with order*). The mid stage 3 prescribe a robustness on the organizational level, in terms of software process adherence (*Know your actions*). Stage 4 prescribes performance control and precision on an organizational level (*Know the results of your actions*). Stage 4 is the for-runner to the fifth, which is characterized as “excellence in performance” (*Know how to act and change in order to perform at an optimum level*).

If we go to the left side of figure 11, we find the characteristics of the EF stages, which describe the support to the software development organization in the corresponding stages. The starting focus of the EF is on project management practises in order to support “project control”. The definition of standard processes and support of their use is the EF focus in order to achieve “robustness”. In stage 4 the focus of the EF is on having the organizations software processes under statistical control, i.e achieving process predictability.

Through the successive development of an organizations capability according to stages two to four, the foundation for strategic process change management is laid. In stage 5, the organization is now capable of systematically managing their software processes from a business perspective. In terms of the strategic success hypothesis, this would mean the software organization more easily can optimize its prevailing strategy according to the dynamics of the environment. Furthermore, the organization can more easily coordinate its different key strategic components, of which the software process (and related artifacts) is one.

5.2 The Model in terms of PEF support processes

We will now provide a more detailed view of the PEF capability growth model. This will be done through mapping of the main PEF support processes (9 in number) to the different stages in the capability growth model as described in figure 11.

The result of the mapping is packaged in table 3. In the left hand column, we find the main focus of the EF at each stage, while the right hand column address the PEF model support processes with a corresponding short description. For further details about the PEF support processes, see PERFECT booklet “PIA Experience Factory - The PEF model” [PEF] chapter 4.

Table 2:

Main PEF focus	PEF Support Processes
Stage 2 focus Support single projects to have project control	**Ad hoc** support to projects with respect to project management issues. On request, supply of project management practises that support project control - such as planning/tracking/requirements engineering. **Improvement Program cycle (IPC)**, to define activities to support the organization to reach it’s improvement goals. This implies e.g to direct the current practices within the software organization and EF towards the next stage in the capability growth model.
Stage 3 focus Supply standard software processes and support for consistent use.	**Process Model Support (PMS)**, to provide project support for consistent selection, tailoring, instantiating and use of software processes (including tools and methodology). PMS is a motivating and supporting activity-area for organizational process adherence. **Process Model Engineering (PME)**, to provide the pre-requisite for process adherence. Standard Processes are defined, documented, maintained and kept under configuration control.

Table 2:

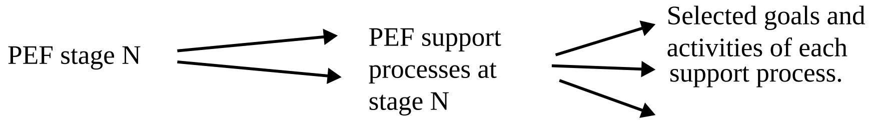
Main PEF focus	PEF Support Processes
Stage 4 focus Supply support and means to have performance predictability of organizations standard processes	**Process Control Support (PCS)**, to provide support for projects to develop their measurement plan. This includes also support to visualize the measures in terms of quality models. This provides the basis for understanding projects and organizations performance in terms of PQTC. **Process Experience Support (PES)**, to guide projects "on-line" to achieve the project goals. The guidance is based upon experience from previous use of the project process. **Process Control Engineering (PCE)**, to integrate quality models with the organizations standard processes. Establish a long-term measurement program for all processes.
Stage 5 Support for organization to maintain excellence in performance through strategic management of software processes	**Process Experience Engineering (PEE)**, to provide support for the continuous improvements of the organizations software processes, based upon statistical analysis. This enables the organization to fill the gap between the actual and the potential performance within the scope of the actual technologies. **New technology investigation (NTInv)**, to give the organization the opportunity to consider/select new technologies in order to meet discontinuities in current or anticipated market demands. **New technology introduction (NTIntro)**, to define activities for efficient technology transfer of new technologies in to the organization and into projects.

5.3 Further improvement of the PEF capability growth model

The model presented in paragraph 4.2 lacks in flexibility. Potentially it could be necessary to let one support process to occur at several stages, although with different scopes.

We can achieve this flexibility by looking deeper into how the PEF support processes are modelled. In [PEF] chapter 4, we see that each support process is described in terms of a number of goals and a number of activities. This means that a stage in the capability growth model can be described according to the structure in figure 12.

Figure 12: Modelling of PEF stages



We thus increase the resolution of the growth model through the use of the goals and activities related to each support process. A particular PEF support process can thus be allowed to appear at different stages, although attached with different goals and activity descriptions.

The work to develop the high resolution growth model, can be part of the tailoring activity for an organization who is about to adopt and implement the PIA. The subject is not further developed here.

6. To consider when implementing an EF

In this chapter we conclude with a number of important considerations, however easier said than done:

(1) A coherent improvement strategy for a company is a prerequisite for success. If an EF is to be part of a company's business strategy, it has to be integrated and tailored according to the existing business plans and strategies. This can be done through a vision process where strategic value of an EF is judged.

(2) The PEF presents a vision for the internal state of affairs which should support the vision for the external state of affairs.

(3) Consider evaluation criteria and an evaluation process in order to drive the EF in the wanted direction. It is extremely important to recognize what has been achieved. The PEF Capability Growth Model can be used as input for the evaluation mechanism. See e.g. [KAPL94], for a detailed elaboration and suggestions in this issue.

(4) Continuous commitment building throughout the organization is a good thing to do. This also implies the handling of resistance to change that follows with changed action-patterns. Behavioural resistance is related to groups and individuals who are threatened in their culture and position of power. Systemic resistance occur when operating and strategic activities compete for the organizational capacity. See further [ANSO90] and [SENG90].

(5) Avoid the EF to be the fad of the year or yet another cover-up. Argyris, see [ARGY90], develops useful insights for how to detect and cope with "organizational defence mechanisms", i.e anti-learning patterns of behaviour.

(6) Experience show that without the early involvement of projects, improvement programmes have a hard time to succeed. A mixed top down and bottom up approach is therefore preferable.

(7) Think big but start small. Start a vision process (think big) in order to clarify the actual purpose of introducing the PIA. Then focus on projects (start small).

(8) Facilitate organizational learning from a general point of view. To achieve a optimal leverage effect of introducing the PIA, learning in the context of software should be supplemented by learning in terms of more general organizational and human aspects, such as e.g. leadership styles and capabilities, commitment culture and empowerment of people. See [SENG90], [SENG94].

(9) The costs for running an EF is roughly estimated to 1-5% of an organizations software development resources depending upon size and degree of ambition. To this figure, must be added costs for training and "transforming" the organization according to the chosen approach. A ROI case should be built in order to better judge the added value of the EF.

7. References

[ARGY90] Chris Argyris, “Overcoming Organizational Defences”, Needham, Mass.: Allyn and Bacon, 1990.

[ANSO90] Igor Ansoff, Edward McDonnell, “Implanting Strategic Management”, Pretence Hall, 1990.

[BASI94] Victor R. Basili, Gianluigi Caldiera and H. Dieter Rombach. “Experience Factory”. In John J. Marciniak, editor, Encyclopedia of Software Engineering, volume 1, pages 469-476. John Wiley & Sons, 1994.

[KAPL94] Craig Kaplan, Ralph Clark, Victore Tang, “Secrets of software Quality: 40 Innovations from IBM”, MC Graw Hill, 1994.

[PAUL93] Mark C Paulk et al, “Capability Maturity Model for Software, version 1.1, Software Engineering Institute, 1993.

[PEF] PERFECT Consortium, PIA Experience Factory - The PEF Model, D-BL-PEF-2-PERFECT9090, ESPRIT Project 9090 “Perfect”, 1996.

[POKR95] Sandy Pokras, “Building High Performance Teams”, Kogan Page, 1995.

[SENG90] Peter Senge, “The Fifth Discipline: The Art and Practise of the Learning Organization”, Currency Double Day, 1990.

[SENG94] Peter Senge et. al., “The Fifth Discipline Fieldbook”, Nicholaus Brealey Publishing Limited, 1994.

[SPC93] Software Productivity Consortium, “Managing Process Improvement: A guidebook for implementing Change version 1.00.06”, U.S. Department of Commerce, 1993.

A booklet from the

Perfect

ESPRIT project 9090

Handbook

A PIA Scenario

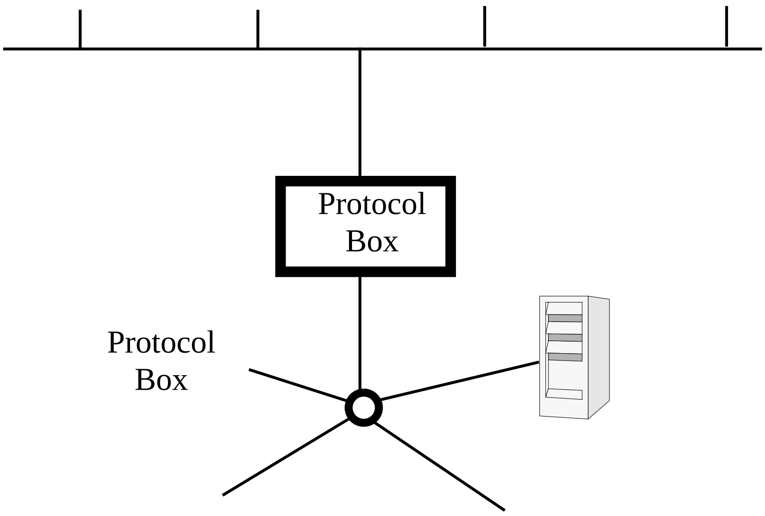


Table Of Contents

1. Executive summary

This booklet describes how an systematic process improvement approach can be initiated and executed in an organization. The example is described in terms of a scenario and is based on a fictitious company named ASDO which represents an “average” software organization. By taking into account the specific characteristics of ASDO, the different functions and steps needed can be described and motivated allowing a better understanding of the needed actions. The scenario starts by identifying the improvement needs of ASDO and shows the actual execution of a number of projects and how the artifacts of the PERFECT improvement approach are derived. During the scenario, examples from the applications within the PERFECT project are used in order to motivate the similarities and differences between the fictitious company ASDO and real organizations.

2. Introduction

This booklet presents a brief example of the introduction of a PIA/PEF in a fictitious company named ASDO. It is not intended as an in-depth description on how to fully utilize the concepts, technologies and tools developed within the PERFECT project, the focus is rather on showing how the general and basic ideas of PERFECT can be implemented throughout an organization in order to achieve systematic process improvement. In the scenario we have tried to discuss the decisions and choices any organization has to address when implementing a systematic process improvement programme. We have also tried to present how the major artifacts are derived during an improvement cycle. By also adding experiences from the PERFECT applications we try to focus on the specific characteristics of ASDO.

2.1 Purpose of this booklet

The purpose of this booklet is to give the different roles within an improvement program a better understanding on how the elements of the PERFECT improvement approach fits together. The intended audience of this booklet are any one with an interest in how systematic process improvement based on the PERFECT improvement approach is applied. The only answers this scenario will not address are detailed questions of the type how is this done or performed or what kind of relation that exist between artifacts. For this latter type of questions, the specific booklets of the different technologies have to be consulted. Generally the scenario will give guidance to the following types of roles:

- Improvement initiators and champions

  The improvement initiator tries to establish an improvement programme or tries to motivate the organization for a change. In the scenario we try to address some of the concerns of this group, i.e. how to relate the business goals with the improvement and measurement goals. Audience within this group will also be given insight how to motivate, launch and plan a systematic improvement programme.

- Managers and sponsors.

Managers and sponsors of any improvement programme are interested in the cost of the systematic process improvement programme and the expected ROI. In this scenario these roles will be explained and the understanding on which functions that have to take part, that systematic process improvement is a long term commitment over several project and when they can expect to calculate ROI.

- Improvement agents.

  The improvement agent plans and performs the improvement programme. The scenario presents a first overview of how to plan and follow up the improvement programme.

- Affected practitioners.

  For the practitioners it is important to understand the steps of the improvement programme. By reading this they will be able to grasp the different activities and technologies used and to understand their general relationships.

2.2 Problem statement

Systematic process improvement requires that a number of decisions, plans and actions are taken and implemented. This scenario tries to show how these requirements can be resolved within a systematic improvement programme.

2.3 Context

The scenario is based on the PERFECT Improvement Approach (PIA), where the PIA Experience Factory (PEF) model is an important part. Other parts of the PIA include the Platform, the Methods and the Experience Base. In the PERFECT project the following parts have been used as input for the scenario:

PIA (PERFECT Improvement Approach)

The overall steps of the PIA is based on the six steps of the QIP and the Experience Factory concepts. The PIA contains the framework for how systematic process improvement should be performed based on the fundamental PERFECT ideas. See the “PERFECT Improvement Approach” booklet, [PIA].

PEF (PIA Experience Factory)

The PERFECT Experience Factory model is integrated in the PIA. It contains and describes one instantiation of the QIP/EF concepts from the PERFECT point of view. Major elements of the PEF are the functions and roles needed. The PEF model is used as input for the scenario description. The PEF is described in detail in the booklet “PIA Experience Factory - The PEF model”, [PEF].

Launching

The launching of PIA/PEF describes how the organization should view and perform the introduction of systematic process improvement according to PIA/PEF. The launching concept is described in detail in the booklet “Launch of the Experience Factory”, [Launch].

GQM

The GQM technology package from PERFECT is the collection of the GQM-related results from PERFECT. It contains a definition of the GQM technology and material supporting the application of GQM. See the handbook “Goal-oriented measurement using GQM”, [GQM]

Tools

The PIA is supported by a platform and also support by an experience base. The scenario will not go in detail on how to utilize the platform and tools developed within the PERFECT project or the implementation of an experience base. The support tools, the platform and modelling concepts are discussed in the “APEL Abstract Process Engine Language”, [APEL], the “Process Modelling”, [ProcMod], the “A Platform for supporting the PERFECT methodology”, [Platform] and the “Experience Base”, [ExpBase], booklets.

Applications and other experiences

The application projects have implemented the PIA and given feedback continuously to the development of the PIA and thereby the PEF model. The experiences made in the application have been generalised and then included in the relevant parts of the PIA and the PEF. Experiences from the applications are described in the A-PIA booklet. See the “Applied PIA”, [APIA], booklet.

The Experience Factory is a logical and/or physical organization that supports project developments by analyzing and synthesizing all kinds of experience, and supplying that experience to various projects on demand. It has been used in several organization where most experiences are coming out of the work performed at the NASA/SEL [Basi94].

2.4 Document Overview

The booklet starts with an overview of the fictitious company named ASDO (A Software Development Organization) and presents its strengths and weaknesses. In the description of the introduction of the PERFECT approach, the first improvement cycle is presented in depth including the planning phases and execution phases. Subsequent chapters describes the planning of the next improvement cycles based on the experiences of the first cycle.

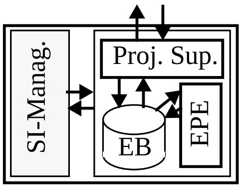
3. Organisation descriptions

3.1 An example organization – ASDO

3.1.1 Introduction to ASDO

ASDO is a medium size Software development company with 50 developers. The company is now for 10 years in business and specialized in developing protocol converters for LANs and WANs to share different types of equipment. The product sales increases annually with 50%. In the beginning there was not that much competition as this market was relatively small. But meanwhile company networks with different equipment and protocols are widely used and attracted strong competitors.

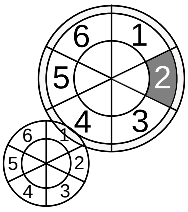
3.1.2 Product domain



The products of ASDO are small “Black Boxes” which are connected between different components and networks. These boxes automatically perform the needed protocol conversion without any external configuration. The hardware for the protocol converters is well standardized.

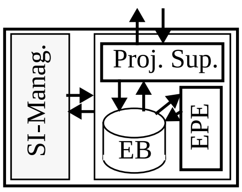
The main effort for a new product or a tailored and customized solution is the development of the software. ASDO has two main types of products. The software for the standard converters for standard solutions/networks has a size of approximately 200 KLOC. A new update of the software costs approximately 6000 man hours and takes less than 6–12 month. For large organizations with specific needs ASDO develops tailored and customized solutions. Their code size is approximately 500–1000 KLOC and the development of a new product costs 20000–30000 man hours with a lead time of 12–18 month.

3.1.3 ASDO’s improvement needs



The life cycle of the products becomes shorter as new technologies are implemented in the networks of the customers. The introduction and massive use of intranet technologies demands advanced and complex converters with high functionality. As the networks become a more integral part of the customers organisation the quality of the products is a very sensitive issue.

The specific problems of ASDO are the increased costs and the shorter time to market. Due to the growing size of the software the fault rate in the system tests has increased. As a result the number of customer complaints on service quality grew together with the costs for maintenance.



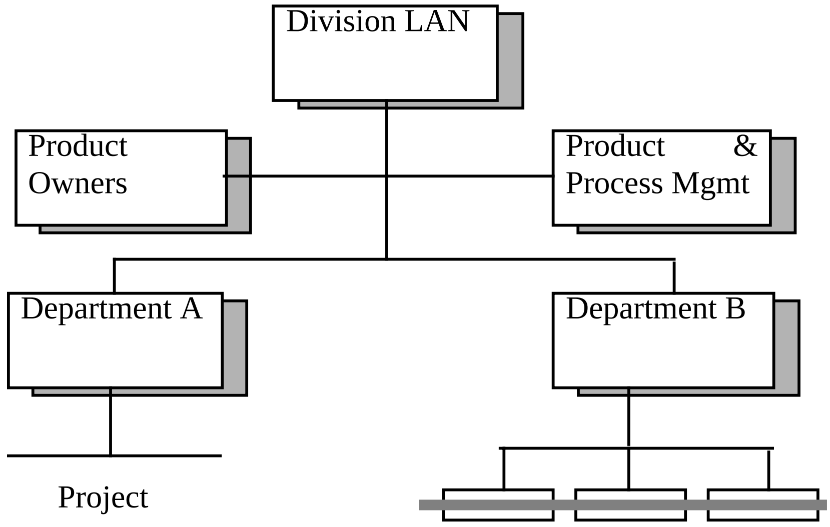
The management realized that ASDO grew to a point where the old development structure with its informal process reached its limits. It became harder to integrate new personnel. The loss of two project managers caused major delays in one project and the total failure of another.

The position in the market is quite comfortable but ASDO feels more pressure from competitors and the customers to cut down the time to market.

3.1.4 ASDO’s current organization

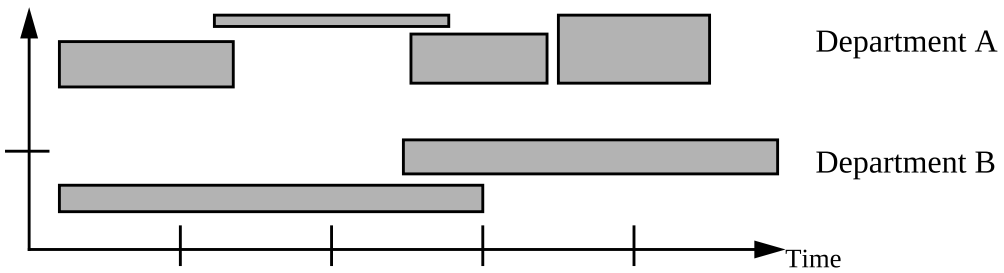
ASDO consists of two main departments. Department “A” is responsible for the standard applications, department “B” for the tailored applications. Both department are using principally the same defined process which is owned by the process management (see figure 1). Department “B” has three sections and runs it’s project through these sections as in a traditional matrix organisation (see figure 1). Department "A" is smaller and has only one section. Each department has it's own domain knowledge which is not shared with the other department.

Figure 1: Organization of ASDO



The departments run their projects in the same way. They use only a limited overlap for their sequence of projects. The projects within one department differ in size and duration (see figure 2).

Figure 2: Typical project schedules for ASDO



Most developers in ASDO have a background in electrical engineering. Only some newcomers have an education in computer science. None has a specific software engineering education but most of the developer are aware for the needs of it.

In the early days of ASDO the functionality of the protocol translators were quite small. The methods and techniques for developing software were well suited for these small developments. But as the projects grew in size and complexity the problems with the software reached a significant level. The first initiative was the definition of the process. The development process is now divided into several phases.

Some data are gathered in the process. But only the numbers of faults found in system test is formally defined. The specification of the lead time of a project is still vague.

ASDO uses an implicit waterfall process for the software development. The project organization is a strict hierarchy where each subproject leader is responsible for 5 – 10 developers. The different tasks of a project are split up by the management and assigned directly to the developers. There is no team work and every developer is only responsible for his tasks. The quality of the products is highly dependant on the individual developer.

Some initiatives for improving the process were performed with mixed results. The introduction of a configuration management system was quite successful while the introduction of C++ failed and was stopped. Currently only C is used as programming language.

Management and the developers see the need for further improvements. The lead times of the projects are relatively small so that ASDO can run some pilots of new technologies. The department "A" performs usually shorter projects that the department "B". It was decided by the management that any further improvement initiative is piloted by department "A" and if successful transferred to department "B" which runs the projects with a much longer lead time.

3.2 Application providers

3.2.1 Siemens

Siemens AS in Norway is an organisation of about 2.500 employees. It is organized as 9 different divisions, where each division is divided into a number of sections.

The organization where this application takes place is the Defence Systems section in the Telecom division. The section employs about 50 persons, among them 15 software developers. Embedded systems which have severe performance and memory constraints are developed using SDL-tool as CASE tool and C, assembler and SDL as programming languages.

Figure 3: Project Schedule at Siemens

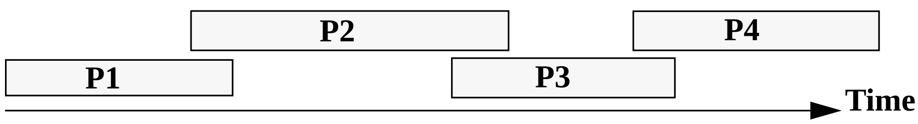


Figure 3 shows a typical sequence of the projects at Siemens defence. The overlap between the projects is 1–2 month.

3.2.2 Robert Bosch

Robert Bosch GmbH is a manufacturer of electrical and electronic components. It is divided into 4 major business areas, of which the automotive and telecommunications area account for about 75% of the approximately 150,000 employees. K3 is one of the nine business units within the automotive area. It manufactures fuel injection systems for gasoline. The development departments that are the focus of this study develop control software for the fuel injection modules.

The application is an embedded system. So the product complexity does not come from the software, it comes from the physics of the product. Two projects have been studied. Each project consists of developing an infrastructure and then producing perhaps 1 to 2 hundred variants of the product in the space of 2 to 3 years.

The production of the variants is the focus of the measurement/improvement programme. It is much closer to production in a factory environment than variant production for non-embedded system products.

3.2.3 Ericsson

Ericsson's 85,000 employees are active in more than 100 countries. Their combined expertise in switching, radio and networking makes Ericsson a world leader in telecommunications. Ericsson is a knowledge-based company with 120 years of experience in telecommunications. Ericsson is active in almost all sectors of the telecommunications field. The company is divided into five business areas, i.e. Public Telecommunications, Radio Communications, Business Networks, Components and Microwave Systems. These five business areas have a common core technology and strategy. They cooperate closely and to a very large extent provide one another with products and services.

The target product is embedded software for mobile phone systems (basestations). Software development is distributed over different sites in Europe (Germany, France, Greece, Spain) and US. The size of the project is approximately 300k manhours of design work within approximately one year. The design work is carried out by 3 to 6 design teams at each site with 5 to 6 designers per team.

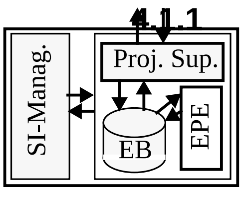
3.3 Comparison

The major difference between the ASDO organization and the three applications are the size of the complete organizations. ASDO is to be regarded as a small company where Siemens, Bosch and Ericsson are regarded as being large corporations. The impact on the process improvement programme w.r.t the size will be minor, some identifiable difference might be that the larger organizations might have more resources available. All organizations presents similar project structure and the same project schedules, where ASDO also includes two related but different departments. From a process improvement programme we regard them as almost the same if the focus is on the improving departments.

4. Mapping between organizations and PIA/PEF

In the following chapters the result of assessments of the organizations are compared with the PIA/PEF model to identify what parts of the PIA/PEF are already fulfilled by the organizations and can be reused in the improvement programme.

4.1 Mapping between ASDO’s organization and PIA/PEF



4.1.1 Strategic improvement management

Improvement programme cycle (IPC)

ASDO has started some initiatives for improving their process with improvement management. As a first step ASDO established a improvement programme cycle which is in the responsibility of a local task force. This should show the commitment of the sponsor for the improvement programme. The task force also maintains a characterization of the process, but this document is still an early draft. The identification of the weak areas with biggest improvement potential is done using inspired guesses as the measurement programme is not able to deliver appropriated results yet. The connection between the improvement goals and the short term business plans are somewhat vague and not clearly defined. The planning for the improvement programme has started but not reached its final state. As the measurement programme is still in an early state the monitoring and verifying of the achievements of the improvement programme is very difficult. The improvement cycle has not reached the finale state of analysing and packaging the experiences.

New technology investigation (NTInv)

The new technology investigation process is only partly in place at ASDO. The identification of the best candidate methods and technologies is done in an ad hoc manner considering the current load of the people and the availability of the resources. Only one pilot had a good preparation and the evaluation was supported by a small measurement plan.

New technology introduction (NTIntro)

The introduction of new technologies is still done in an ad hoc manner at ASDO. There is no other characterization of the impact of the new technology than a rough estimation. For the introduction ASDO only uses training courses before or at the beginning of a project. No transition plans exist and they have no change allies to perform the changes. The task force is also responsible for providing the infrastructure and the training material. The information flow between all concerned persons is rather low. Without a detailed plan for the introduction it is almost impossible for ASDO to track the transition in a proper way. Analysing and packaging the results is not done.

4.1.2 Experience package engineering

Process model engineering (PME)

In the definition of the software development process most of the details were captured. But the descriptions of some activities, criteria and products are still incomplete. Some planning for the tailoring has started but nothing has been applied yet. The documentation of the current process follows the existing guidelines of the organisation.

Process control engineering (PCE)

The only measurement currently included in the process is the number of errors found in the system test. It is planned to include other measurements into the different processes, but this has not been started yet.

Process experience engineering (PEE)

Based on the currently available data the process experience engineering (PEE) part of the PIA is not done.

4.1.3 Project support

Project management support (PMS)

The project support for management is not specified yet. It is however a common understanding that the local task force will support the project management.

Project control support (PCS)

Currently they are only supporting the collection of the errors found in the system test. It is planned to extend this task as the measurement programme will be extended to other data and process steps.

Project experience support

There is no support for the project experience available.

4.1.4 Summary

Table 1:

PEF–Model		ASDO
Project Support	PMS	not done
	PCS	Only the system test is controlled by measurements and the results are reported to the project management
	PES	not done

4.2 Application providers

Table 1 and 1 show a summary of the assessments at Siemens and Ericsson before the improvement cycle.

Table 2:

PEF–Model		Siemens
Strategic Improvement Management	IPC	Nothing
	NTInv	Small test-cases. Continuous activity but ad-hoc.
	NTIntro	Ad-hoc
Experience Package Engineering	PME	Process described by Q-Labs as consultant.
	PCE	Nothing explicit. Only on “project control level”.
	PEE	Not very explicit. Subjective final reports made by project manager.
Project Support	PMS	Process models maintained. Not experience oriented.
	PCS	The project leaders had to read final reports from similar projects.
	PES	Ad-hoc

Table 3:

PEF–Model		Ericsson
Strategic Improvement Management	IPC	Ad-hoc; natural or accidental.
	NTInv	Ad-hoc; market / constraint driven, pilot project / studies.
	NTIntro	Done ad-hoc, but not need driven.
Experience Package Engineering	PME	Process documentation mainly but no modelling.
	PCE	Nothing
	PEE	Nothing

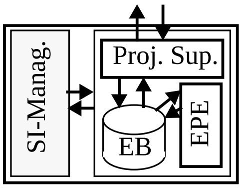
4.2.1 Tools

The local task force uses only standard word processor and spread sheets as tools.There are no tools for process modelling, data gathering or CASE available.

4.3 Comparison

Viewing the different organization, differences in background and taken steps

5.2 Context



The cornerstones for the improvement programme where set by the management based on their long term business analysis:

- The improvements must be introduced in a controlled way and justifiable by objective data.
- The improvement programme has to start with low risk projects.
- Any failure may not risk the whole company.
- Before applied to all departments the benefit of any new technology must be proven within ASDO.
- The time frame is 3–4 years for the whole programme.

The products developed in department “A” are mainly bookshelf products for standard situations. There is no direct delivery contract with a specific customer. Delays and failures are not visible outside ASDO. Department “B” develops directly for a specific customer and has rather hard requirements in their contracts. Any delays or failures in department “B” are directly visible for the customer and can cause the loss of new contracts and market shares. With this business situation it was decided to use department “A” as starting point for the improvement activities.

5.3 Launching

The management approved the following strategic plan for the improvement programme:

(1) Introduce PIA/PEF into department “A”

(2) Run some experiments to find the best solutions for ASDO using 3–4 projects in department “A”

(3) Transfer the successful improvements to department “B”

The task force started with an internal assessment to find out where ASDO is in relation to the PIA/PEF. The result is described in chapter 3.1.4. Based on that the following main goals were set:

(1) All processes are documented and understood with a proper baseline

(2) Control of projects

(3) The quality of the products is kept.

(4) The lead time is reduced by 50%

Based on these goals and the current knowledge about the organization the task force developed a “Strategical guide for implementing PIA” (SGI) and to achieve goal number (4) as early as possible they have chosen inspections as “PIA/PEF improvement proposal” (PIP) right from the beginning. Table 4 summarizes the improvement plan for ASDO.

Table 4:

PIA/PEF-model	ASDO
Characterization	- Typical CMM–level 1 organization - undocumented informal software process - need for explicit project management experience
Goals for ASDO	(1) All processes are documented and understood with a proper baseline (2) Control of projects (3) The products quality is kept. (4) The lead time is reduced by 50%
Chosen technologies	- Formal inspection - CASE tools
Execution	- 4 projects in department “A” with a time frame of 3 – 4 years are chosen. - The detailed plan is listed in table 8.
Analysis	- Are the goals reached? - Impact of new technologies - Improvements
Expected results	- Experience packages for the department “A” - Experience packages for the transfer to department “B”

For the ASDO it is possible to apply different approaches for reaching the next stage. A proper baseline for gaining control over the process could be applied. One problem with this approach is that there is no direct improvement visible and it takes at least one project cycle. In the case of ASDO it was decided to apply some well known improvements to get some visible results as early as possible to get and keep up the motivation for all participants of the improvement programme. Therefore the baselining measurement programme is accompanied with the introduction of inspections. Table 5 summarizes the activities at ASDO with the respect to the PIA/PEF model. Figure 4 shows the methodological view of the ASDO. The big cycle represents the steps of the strategic level (compare also table 4) and the smaller ones the project level with four projects.

Figure 4: Methodological view

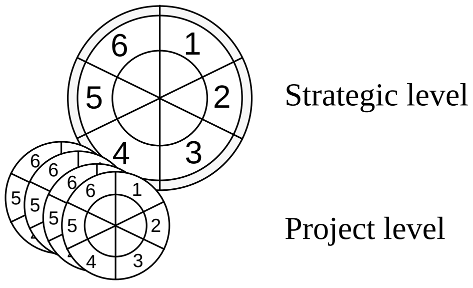


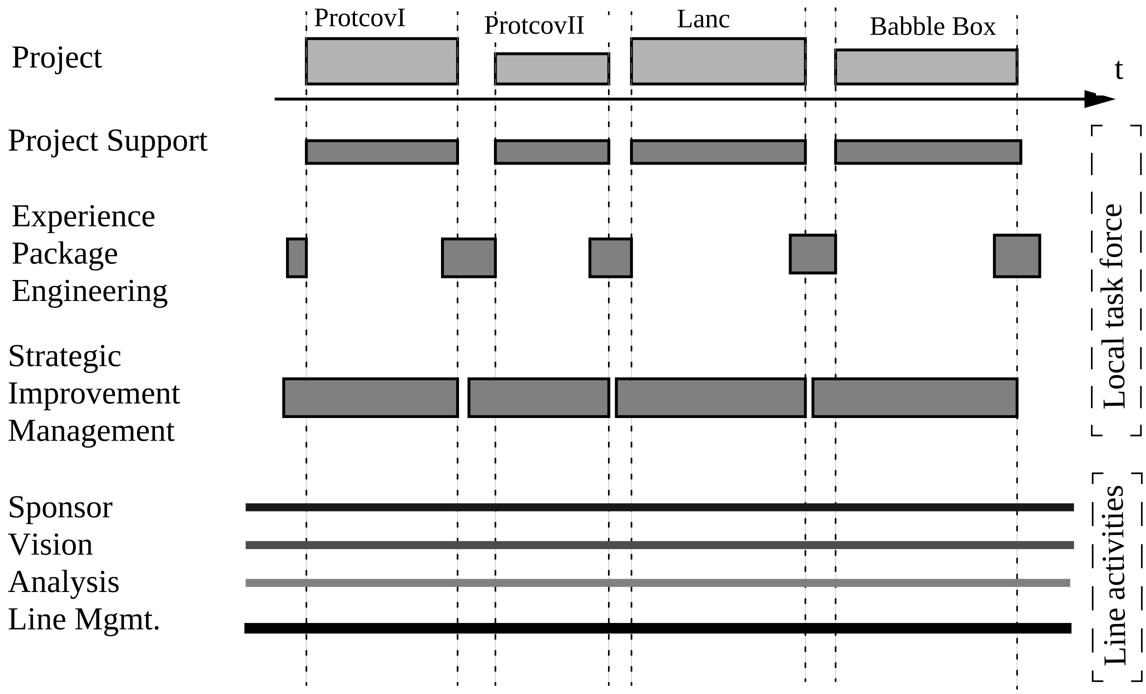
Table 5: Plan of the organization before the first project

PEF–Model		ASDO
Pre-Launching Activities	SGI	A strategic concept paper was developed describing the usage of department “A” as pilot organization. The transfer to department “B” is mentioned as a long term goal.
	PIP	Baselining and the introduction of inspections in all process steps.
	SPS	The project support is partly defined within the work assignments of the local task force. No authorities are clearly defined.
Strategic Improvement Management	IPC	This is the first cycle of the improvement programme with a project of department ”A” 3-4 years focus on department “A” Transfer of the results to department “B”
	NTInv	Improvement proposals are identified from the literature. No impact analysis has been performed.
	NTIntro	A rough plan for the introduction of inspections and CASE tools exists.
Experience Package Engineering	PME	The completion of the process description to be done by the local task force.
	PCE	All processes must be measured.
	PEE	Abstract process description to be developed by the local task force.
Project Support	PMS	The local task force will give support to the project management based on the measurement programme. Detailed descriptions are still missing.
	PCS	All phases of the project are measured
	PES	To be done by the local task force.

5.4 Detailed planning

The task force that was used to describe the current process at ASDO took over the responsibilities for the execution of an improvement cycle at ASDO. The task force is excluded from all development tasks and fully sponsored for the introduction of the PEF. Figure 5 gives an overview of the support activities for the next 4 projects in department “A”, which relates to the 4 small cycles in figure 4.

Figure 5: Project/support plan



The task force will play the role of an internal consultant group that gives support to the projects. No direct authority has been assigned. The task force started with setting up an internal information system where they could display their work and results. The group is responsible for setting up a main plan which initiatives will take place in department “A” for the next 2 years. All improvement initiatives must be coordinated by the task force. Table 8 gives an overview of the plan for department “A”.

Table 6:

Project	Aspect	
ProtcovI 10 month	Goals	Understanding the process and introduction of inspections (goal 1 + 3)
	Expected results	Documented process description and baseline
ProtcovII 6 month	Goals	Further understanding and stabilizing the process. Refinement of the process documentation (goals 1+ 2)
	Expected results	Refined process documentation and process control models First visible improvements.

Table 6:

Project	Aspect	
Lanc 10 month	Goals	Guidance of the project. (goals 1+ 2) Tuning the process. (goal 4) Automatization of the data gathering
	Expected results	Process models validated Control mechanisms refined Weak areas identified Measurable improvements
Babble Box 9 month	Goals	New technology introduction (Hypothesis: CASE tools) Reaching the improvement goals. (goals 1, 2, 3, 4)
	Expected results	Understanding the impact of the new technologies Measurable improvements

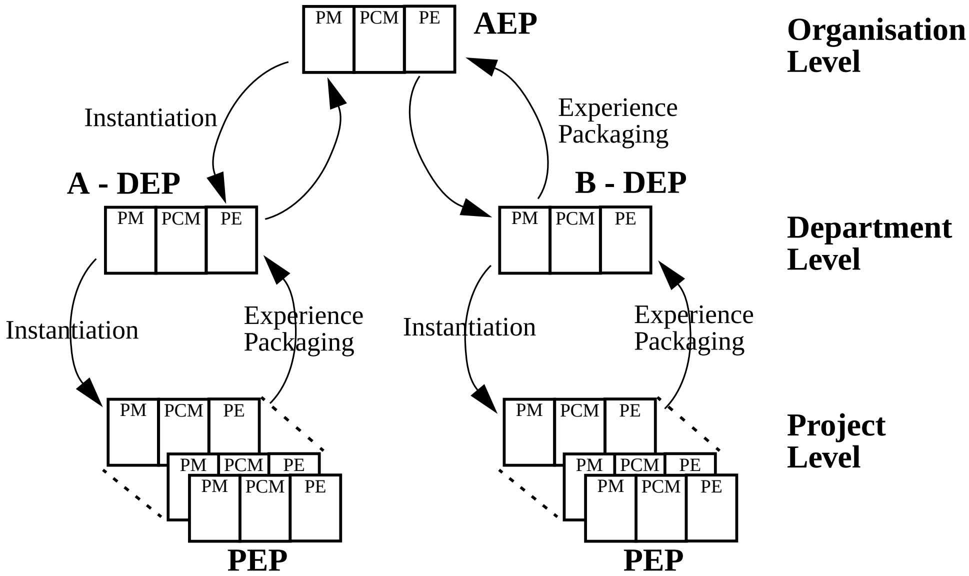
Besides from the direct project support the task force has to set up a document database for all documents and products. This database should also host all data coming from the measurement programme and the experience packages. For the implementation new intranet technologies should be used.

ASDO already has a process description which roughly divides the development into different phases but no formal specification exist. Inspections should now take place in all process phases. The inspection records will be the major input for the measurement programme of the baselining. All initiatives must be tailored to the small projects and groups inside department “A”. This is considered to be a possible problem and risk for the later planned transfer to department “B”.

6. Executing an improvement cycle at ASDO

This chapter gives a vision how improvement cycles at ASDO can look like. ASDOs current state will be often illustrated by experience packages (EP). An experience package is a collected set of consistent process models, process control models and process experience that form a consistent unit (see PERFECT Handbook “The PEF model”). At ASDO three sorts of experience packages are distinguished. project specific EPs (PEPs), department specific EPs (D.E.Ps) and ASDO wide EPs (AEP). How these experience packages are related to each other is shown in Figure 6

Figure 6: Relations between experience packages



6.1 Initial activities

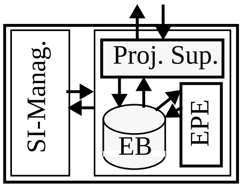
6.1.1 Initial experience package

Before any improvement programme was started ASDO had no explicit knowledge about process control models, process experiences and process models. The development process could be as best compared with an implicit waterfall process. They had only role descriptions and some responsibilities defined in the employee contracts and what on working or project meetings was determined. The EP in Figure 7 could be seen as well on department level as on project level because at the beginning there is no difference between them.

Figure 7: Experience package at start

6.1.2 First assumptions

In a first step there was a discussion about the problems and possible reasons (root cause analysis). It was realized that ASDO had no real control of their development process and that this couldn't be reached this in one step. A systematic approach seemed the best way to be successful. Therefore in a first phase ASDO collected



some guesses about their state and their goals. One goal is to decrease the development cycle from approximately 6 to 12 month by 50% in department “A” and to keep the product quality (especially the meantime between failure). In future ASDO wants here some more reliable data. But before the improvement could be started ASDO had to look from where they start, evaluate the assumptions and how the goals can be reached. Because of a limited budget ASDO couldn’t make a long term research and then start up new technologies. Therefore it was decided also to introduce inspections at the same time because ASDO was convinced that this technology could be a first good step in the right direction.

6.2 Project ProtcovI

6.2.1 Planning first project

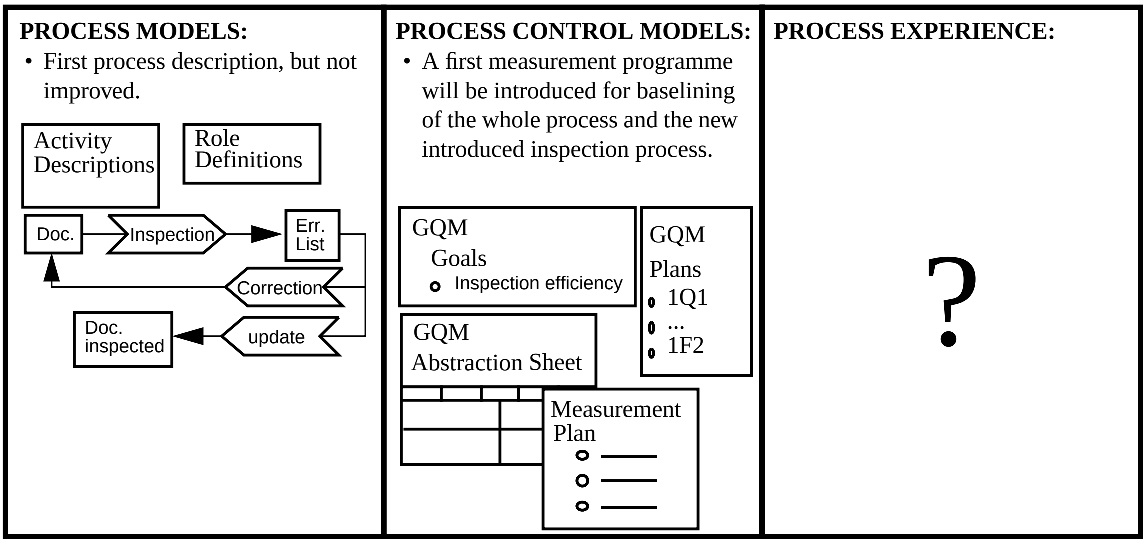
Table 7:

Project	Aspect	
ProtcovI 10 month	Goals	Understanding the process and introduction of inspections (goal 1 + 3)
	Expected results	Documented process description and baseline

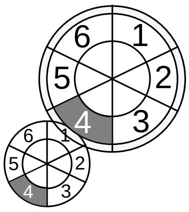
Before the first project for understanding and baselining was started ASDO collected their implicit knowledge. That means the existing processes were captured as far as possible. ASDO defined an own inspection process (based on inspection theory), made their goals explicit and determined how to measure them (using the GQM Paradigm). In addition to baselining and understanding ASDO wanted to measure what is called inspection efficiency (a self defined measure) to control the impact. This included measures like inspection effort, detected faults and fault density in the system test.

The first selected project was ProtcovI. It was a low profile and non critical project with a typical project size (50 man-months) like other department “A” projects. ASDO assigned five person 100% to ProtcovI. Figure 8 illustrates the experience package at start of ProtcovI. Table 7 shows the goals of ProtcovI.

Figure 8: PEP- A01[1] at project start of ProtcovI

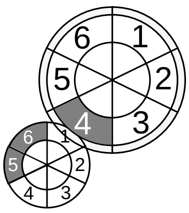


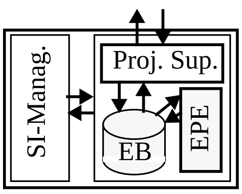
6.2.2 Executing ProtcovI



During the project execution ASDO collected the measures which they had identified with the GQM Approach before. ProtcovI was not so easy to manage because of the additionally impact from the task force group which controlled the measurement programme and the new inspection process. ASDO had especially forgotten to plan enough additional effort for preparation and meeting of the reviews and the collection of the data. Therefore some milestones were missed in the project. At the later phases of ProtcovI ASDO ran into time pressure and some employees fall back in the old behaviour and the data collection was partly omitted.

6.2.3 Results of the analysis





Because ASDO ran out of time the project was finished 1 month later than planed. But on the other hand the delivered product had a satisfying quality because the fault density in the system test was lower than in the previous projects. It was concluded from this that probably the mean time between failure is longer, but at the moment exists no reliable data about this. ASDO made first experiences and identified some problems and weak points. After the project the collected data were analysed and gave the following results (see also Figure 9):

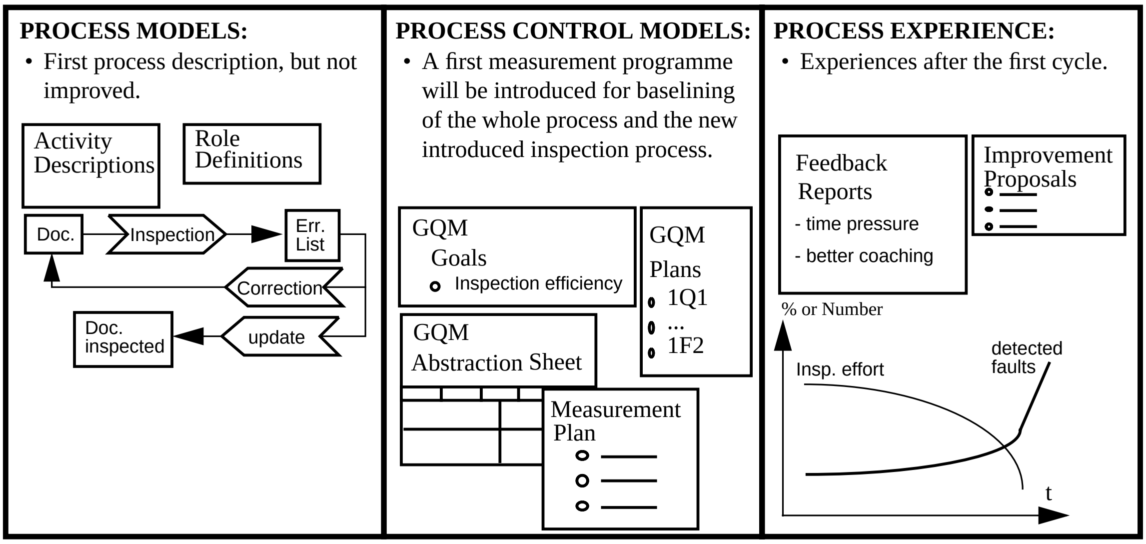
- In the last project phases the inspection effort decreased.
- In the later phases of ProtcovI were the most faults detected.
- The fault density (errors per lines of code) in the system test was 30% less than before.
- The time pressure increased to the end of the project.
- Because of some planning failures and the time pressure some persons partly fall back in the old behaviour.

1. EPs are numbered consecutively in every department, e.g. PEP-A01 means the first PEP in Department “A”

- The employees had problems applying the new processes.

ASDO got a better understanding of their processes and inspections were introduced, but not fully understood. In the next cycle ASDO had to stabilize the data collection process to got better results. The data material was not as good as expected and therefore the results of the analysis were also weak, but the made experiences gave some ideas what had to be changed in the next cycle.

Figure 9: PEP - A02 after analysis of ProtcovI

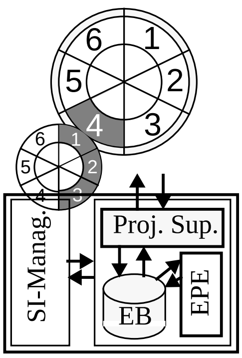


6.3 Project ProtcovII

6.3.1 Planning ProtcovII

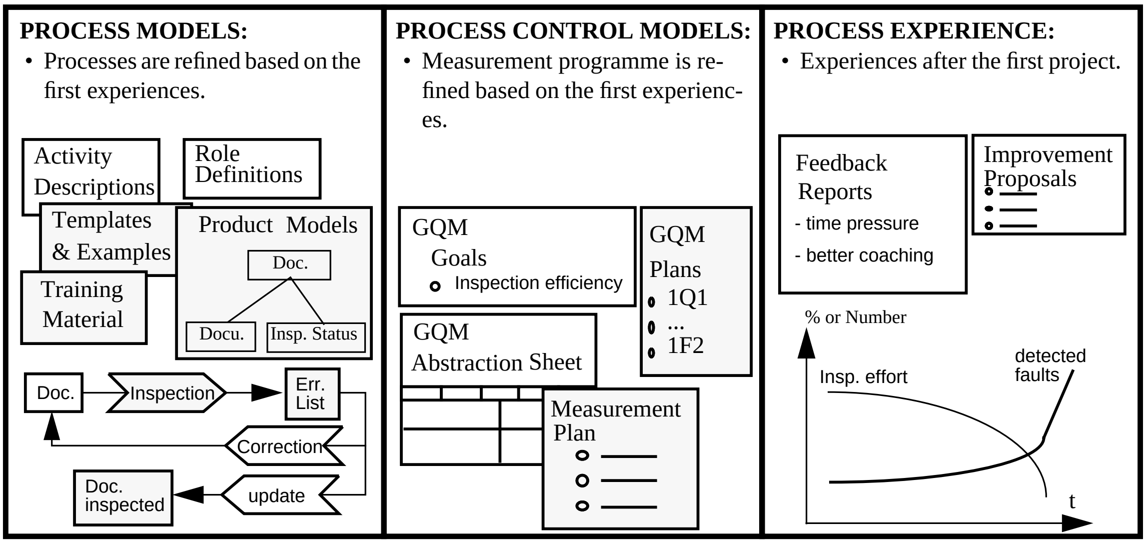
Table 8:

Project	Aspect	
ProtcovII 6 month	Goals	Further understanding and stabilizing the process. Refinement of the process documentation (goals 1+ 2)
	Expected results	Refined process documentation and process control models First visible improvements.

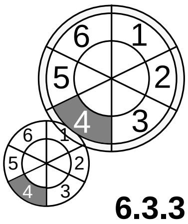


Based on the analysis of the data from the first measurement cycle ASDO made some refinements and extensions in their “Experience Base”. The inspection process was adapted, the structure of the error list and the inspected document changed a little bit to have a better handling. Templates and training material were created to get a better process compliance and a first product model was created. Also some problems with the measurement programme were tackled by refining the GQM- and the measurement-plan. After the first cycle the most changes were adaptations of the processes to the local environment (see Figure 4). Table 8 summarizes the plan of ProtcovII. Depending on which level the data is measured (project level or e.g. inspection level) ASDO got more or less data points per project from which the first trends could be concluded.

Figure 10: PEP - A03 at project start of ProtcovII[1]

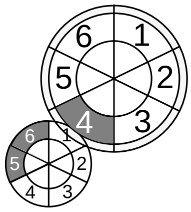


6.3.2 Executing ProtcovII

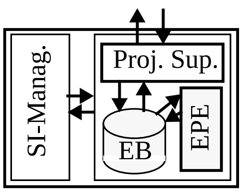


ASDO recognized that the processes became more stable during ProtcovII. The compliance of the inspection process was rather good and the data collection was nearly complete. No milestone was missed in ProtcovII and it was finished three weeks before schedule.

6.3.3 Results of the analysis

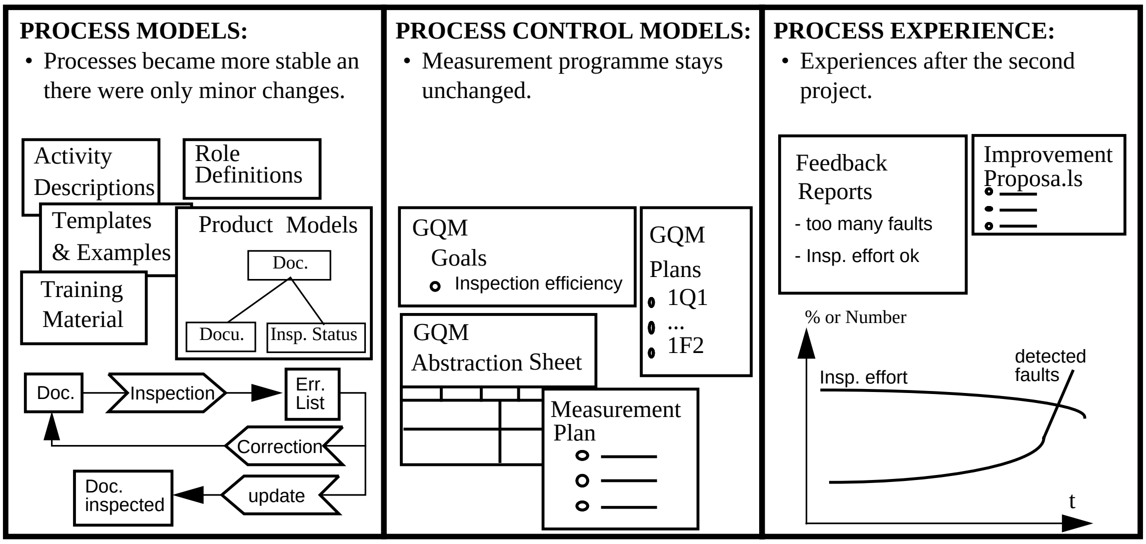


Nevertheless ProtcovII had no major problems ASDO was far away from the desired goals. ProtcovII was finished before time and process compliance was much better than in ProtcovI. This was concluded because the data was much more complete than in Protcov I and the inspection effort decreased not so heavy as in ProtcovI. But the results of the inspections were even worse than in ProtcovI. The fault density in the system test was higher than in ProtcovI.



1. The gray areas showed where ASDO made changes or extensions

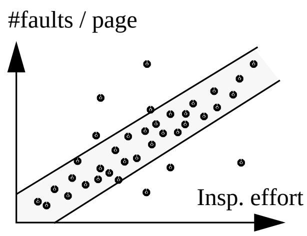
Figure 11: PEP - A04 after analysis of ProtcovII



6.4 Experience engineering after Protcov I and II

After the first two projects ASDO had the experience that the inspections need an appropriate effort to be efficient. Based on the data of the first two projects ASDO built a first model to guide the inspection process (see Figure 12). This model is part of the Experience Package of department "A" and can also be seen in the next project EP (Figure 13), which is an instantiation of the A-DEP. Guidance means, ASDO had enough data to recognize when the inspection process is probably not longer within the boundaries. If ASDO measures parameters outside the standard deviations during a project an alarm signal will arise and ASDO can react on this. But ASDO had not enough experience at this time to say which steps probably get them back into the right borders.

Figure 12: Experience with inspection effort

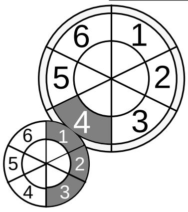


6.5 Project Lanc

6.5.1 Planning of Lanc

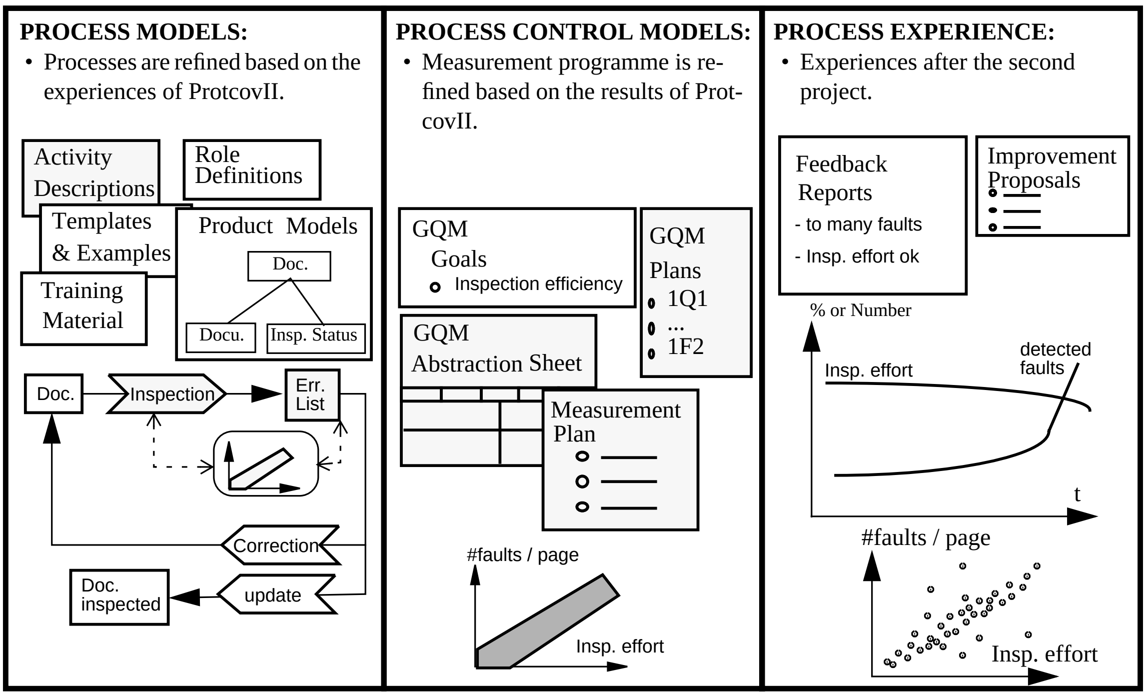
Table 9:

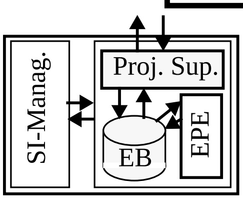
Project	Aspect	
Lanc 10 month	Goals	Guidance of the project. (goals 1+ 2) Tuning the process. (goal 4) Automatization of the data gathering
	Expected results	Process models validated Control mechanisms refined Weak areas identified Measurable improvements



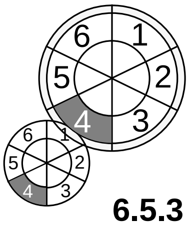
During the inspections in ProtcovII the assumption arose that the high fault density at the end are not faults in the later phases but problems in the earlier ones. Therefore ASDO adapted the GQM- and measurement-plan to find the reasons for the high fault density. Especially the measures in the inspection process were refined (more error classes than before). Due to this a full guidance and tuning of the process like planned (see Table 8) was not possible. To make the data collection easier ASDO purchased a tool which support the data collection and analysis.

Figure 13: PEP - A05 at project start of Lanc



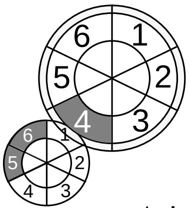


6.5.2 Executing Lanc



During the execution of Lanc only minor problems were recognized which probably arose because of the introduction of the measurement tool. The inspection effort model seemed reliable so far. Lanc was finished one month before schedule but in the system test were many errors detected which had to be corrected.

6.5.3 Results of the analysis



ASDO was satisfied with the results in so far as they were successful in quality and timely delivery (about 10% lead time reduction) but they were unsatisfied with the fault density in the system test. Because the inspection process worked well there had to be an other reason.

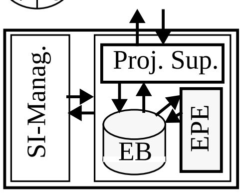
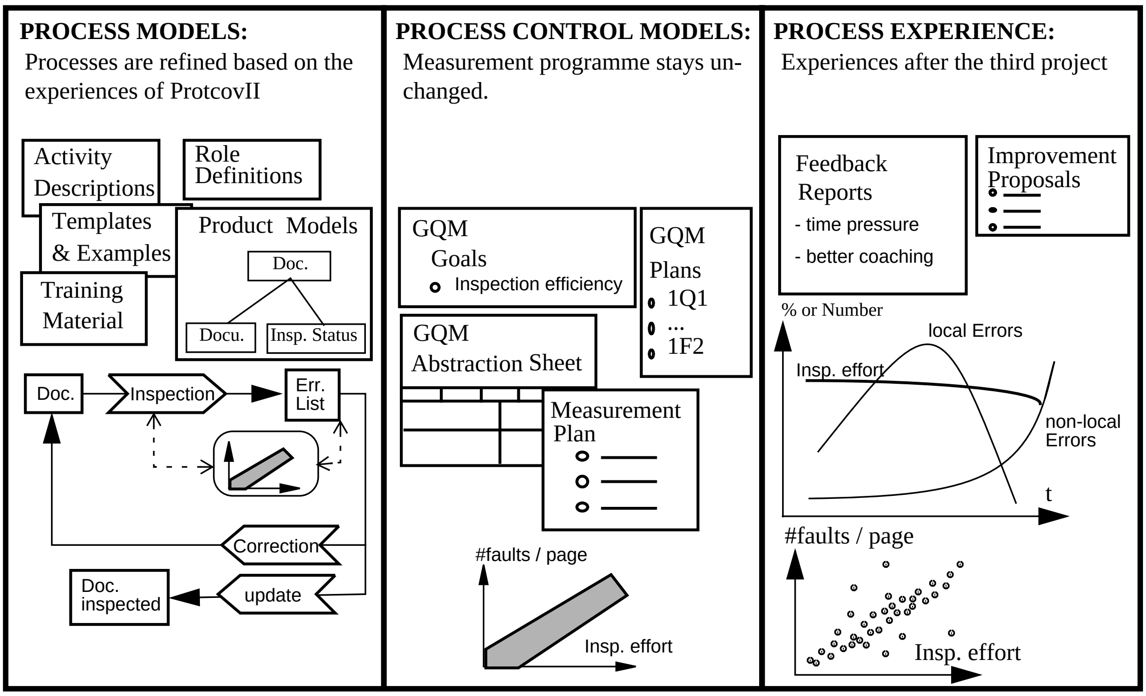
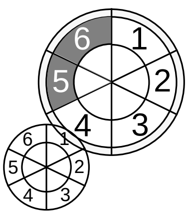


Figure 14: PEP - A06 after analysis of Lanc



6.6 Experience packaging after Lanc



ASDO made some very useful experiences in the first three projects. The planed goals are not reached so far, but the understanding of the processes is much better than before and the control of the processes is improved. The lead time was reduced about 10%, but this is much behind the expected result. The inspection process is accepted and works well.

ASDO recognizes that their problems are not in the coding phases what was assumed before but in the earlier phases. Therefore ASDO wants to investigate more in a better documentation and specification of the system. Thereby the documents become better inspectable and the inspection will deliver better results than before and become more useful.

The planned investment in a CASE-tool is postponed as long as the problems in the earlier phases are not solved and therefore the usefulness of such a tool is questionable.

6.7 Technology investigation

ASDO decided to investigate in new methods for specification and documentation. This was because it had been found that the greatest potential for improvement was in the earlier phases of the development process. It was however not obvious that the new methods actually would result in improvements when they were introduced in the forthcoming projects. Furthermore, it was not clear which out of the possible methods was the best. Thus, it was decided to investigate the new methods by analysing the potential effects before making any changes to the existing development process.

Two different CASE tools were identified as interesting to use. The first CASE tool mainly relied on specifying the system with data flow diagrams and the second could be used to specify the system with SDL. It was decided to investigate both the change proposals with the impact analysis method [ImpactA] and then choose the best one if this method was expected to result in enough improvement.

The new methods were incorporated into the process for the next project as suggestions of new process models. This was done by identifying the changes of the original process that were needed if the different CASE tools were to be used. These new process models could then be analysed as process change proposals. Next, a number of persons were identified as experts with experience from each phase of the original process. Some persons were only identified for one phase, but some persons were identified for more than one phase.

For each phase of the new process, the identified experts were asked to give their opinion on the effects of the changes of the original process. This was done by ask-

- The mean cycle time of the future projects would decrease with around 25% due to the change.
- The number of remaining faults would decrease by around 35% due to the change. This figure was uncertain, but it was not as uncertain as the corresponding figure for firs change proposal.
- The cost of the projects would decrease by 5% due to the change.

The result of the analysis was stored as experience (see figure 1). The analysis showed that the second change proposal seemed to result in more improvement than first change proposal, but the results of the analysis showed a great uncertainty. The uncertainty was however larger for the first change proposal than for the second. It was decided to not further work with the first change proposal for the moment, but to concentrate on the second change proposal.

7. The next improvement cycle

The findings in the previous projects suggested a major change for the main plan (see table 8). The figures showed clearly that the introduction of a CASE tool would not give the expected results. Therefore the overall improvement cycle (table 4) was revised and refined. The following table 4 shows the new overall improvement cycle at ASDO based on the experience of the first three projects

Table 10:

PIA/PEF-model	ASDO
Characterization	- Typical CMM–level 1 organization in department “B” - CMM–level 2 reached in department “A” - undocumented informal software process in department “B” - need for explicit project management experience in department “B”
Goals for ASDO	(1) All processes are documented and understood with a proper baselining department “A” and “B” (2) Control of projects (3) The products quality is kept. (4) The lead time is reduced by 50%
Chosen technologies	- Formal inspection - Document standards and formal methods for specification
Execution	- 2 projects in department “A” with a time frame of 1 year are chosen. - 2 projects in department “B” are chosen for the transfer of experience packages. - The detailed plan is listed in table 11.
Analysis	- Are the goals reached? - Impact of new technologies - Improvements

Table 10:

PIA/PEF-model	ASDO
Expected results	- Experience packages for the department “A” - Experience packages for the department “B” - First generic experience packages on ASDO level.

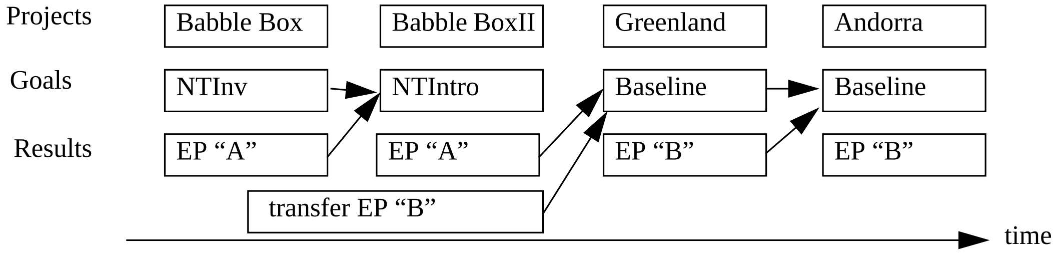
Table 11 shows the new revised plan. ASDO will stay with the strategic planing and start the transfer to department “B”

Table 11:

Dep./Project	Type	Planned activities
“A”: Babble Box 9 month	Goals	New technology investigation
	Expected results	Improve document standards and introduce formal methods for specifications.
“A”: Babble Box II 7 month	Goals	Further understanding and stabilizing the process and the usage of the formal methods
	Expected results	Refined process description and process control models. Preparation for the transfer of EPs to the department “B”
“B”: Greenland Bank 18 month	Goals	Introducing PIA/PEF into department “B”. Understanding and baselining.
	Expected results	Tailored EPs for department “B”
“B”: Andorra Automobiles 15 month	Goals	Further understanding and baselining. Introduction of inspections.
	Expected results	Tailored and refined EPs for department “B” and first generic EPs for department “A” and “B”

Figure 15 shows the time line for the projects listed in table 11 and the support activities performed by the local task force.

Figure 15: Time table for the IPC at ASDO



The table 12 describes the status of the PIA/PEF at ASDO after the project “Andorra Automobile" was finished in department "B".

Table 12: Status of the PIA/PEF at ASDO

PEF–Model		ASDO
Pre-Launching Activities	SGI	The strategic concept paper is mainly unchanged. The plan for department "A" is changed (see above).
	PIP	In department "A" the improvement proposals are based on a solid set of process data. "B" has not yet reached this level.
	SPS	The local task force has developed a standard procedure for department "A". "B" is due to the different size and type of projects still experimental.
Strategic Improvement Management	IPC	Well established in department "A". Accepted in "B" but still unstable.
	NTInv	Good baseline for department. "A" available, no major failures yet.
	NTIntro	Established and stable for "A", still risky for "B" but no failures yet.
Experience Package Engineering	PME	Done in "A" and "B"
	PCE	Guidance packages available for "A" but not yet for "B".
	PEE	First generic EPs are now finished but not yet evaluated
Project Support	PMS	Project management is aware of the support and using it.
	PCS	Works good for "A", guidance for "B" is not possible yet.
	PES	An experience feedback loop is not yet established.

7.1 Application providers

The following two tables show the status of the PIA/PEF after the improvement cycle at the application providers.

Table 13:

PEF–Model		Siemens
Strategic Improvement Management	IPC	Included in the development process description so that every project has to re-use and provide experience.
	NTInv	Running test-cases; ad-hoc
	NTIntro	Ad-hoc

Table 13:

PEF–Model		Siemens
Experience Package Engineering	PME	Generic process model is maintained. Tailoring done for every project.
	PCE	Some quality models (“defect slippage”) are quite mature.
	PEE	Experience still on project level. Experience packages not generalized.
Project Support	PMS	Process models maintained (the generic one).
	PCS	Normal effort / schedule control. Some feedback sessions used for control.
	PES	Data are analysed and stored in a database ready for reuse.

Table 14:

PEF–Model		Ericsson
Strategic Improvement Management	IPC	As a organizational change a Design Support Coordination Team was established. Methodological changes are now addressed by this new entity.
	NTInv	
	NTIntro	
Experience Package Engineering	PME	Quality-experiences are packaged in terms of (GQM) Quality models mainly. Work instructions reflect the packaged process changes.
	PCE	
	PEE	
Project Support	PMS	Systematic by “improvement” projects executed in parallel to the target project.
	PCS	
	PES	

8. Applicability in different organizations

In this chapter we discuss some FAQ (frequently asked questions) w.r.t the scenario and the application of the PERFERCT approach.

- Small, medium, large organisations.

The usage of PIA/PEF is actually independent of the size of the organization. Examples and experiences have shown that we have been able to implement the approach both on small, medium sized and large organizations. One of the PERFECT applications actually contained a large project distributed on 5 different sites as well as on small sites and projects. The major difference is in how to tailor and adopt the different functions and roles. A traditional mistake in small organizations is that all roles are instantiated on several people which does not need to be the case. A person could maintain several roles as long as the intention of each individual role is maintained. ASDO is regarded to be a medium sized organization in this respect.

- Revolutionary versus evolutionary.

The PIA/PEF uses an evolutionary approach. The evolutionary approach presented by PERFECT could be discussed in the light of the revolutionary approach. As the scenario describes an evolutionary approach of understanding the current baseline, controlling the project and finally based on experiences propose a change, the view of the revolutionary approach states that everything should be changed at one time. However, this is no contradiction, the only difference in view is that the revolutionary approach stresses the rapid technology change, the PERFECT methodology also supports the same view but helps to minimize the risks of technology introduction by focusing on what to do. If we based on our business needs need to do a rapid technology change, the chances of being successful are facilitated if we have an understanding of the current baseline.

9. Summary and conclusion

This scenario gives an overview of how to launch, implement and perform an improvement programme based on the PERFECT Improvement Approach. The examples presented are based on a fictitious company, but the company has been derived based on the different applications in PERFECT, experiences during the course of the project and experiences from other projects performed by the involved partners.

One conclusion drawn form the scenario is that systematic process improvement requires planning, long term commitment, and when put into practice that it will give the organization the possibility to systematically reach its goals. Experiences shown from application organizations also shows that several positive side effects are also noticed, e.g. increased communication inside the organization, better planning mechanisms and management commitment as examples. Drawbacks of the proposed framework are found to be the same aspects, i.e. planning and long term commitments that are mandatory concepts but often violated and thus obstacles in the improvement programme.

10. References

[Anso] Igor Ansoff, Edward McDonell, "Implanting strategical management".

[APEL] PERFECT Consortium, APEL Abstract Process Engine Language, D-BL-APEL-2-PERFECT9090, ESPRIT Project 9090 "Perfect", 1996.

[APIA] PERFECT Consortium, Experiences from using PIA, D-BL-APIA-2-PERFECT9090, ESPRIT Project 9090 "Perfect", 1996.

[Basi94] Victor R. Basili, Gianluigi Caldiera and H. Dieter Rombach. "Experience Factory". In John J. Marciniak, editor, Encyclopedia of Software Engineering, volume 1, pages 469-476. John Wiley & Sons, 1994.

[Basi94b] Victor R. Basili, Gianluigi Caldiera and H. Dieter Rombach. "Goal Question Metric Paradigm". In John J. Marciniak, editor, Encyclopedia of Software Engineering, volume 1, pages 528-532. John Wiley & Sons, 1994.

[Basi95] Victor R. Basili and Frank McGarry. "The Experience Factory: How to build and run one". Tutorial given at the 17th International Conference on Software Engineering, Seattle, Washington, USA, April 1995.

[ExpBase] PERFECT Consortium, Experience Base, D-BL-ExpBase-2-PERFECT9090, ESPRIT Project 9090 "Perfect", 1996.

[GQM] PERFECT Consortium, Goal-Oriented Measurement using GQM, D-BL-GQM-2-PERFECT9090, ESPRIT Project 9090 "Perfect", 1996.

[Gust95] PERFECT Consortium, edited by Anders Gustavsson and Christer Mattson. "The PERFECT Approach to Continuous Improvement using Experience". Presented at the ESI-ISCN conference in Vienna, Austria, September 1995.

[Höst95] Martin Höst, Claes Wohlin. "Impact Analysis of Process Change Proposals" in Software Quality, pages 311-323, edited by M. Ross, C.A. Brebbia, G. Staples, J. Stapleton, Computational Mechanics Publications, Southampton, United Kingdom, 1995.

[ImpactA] PERFECT Consortium, Impact Analysis, D-BL-ImpactA-2-PERFECT9090, ESPRIT Project 9090 "Perfect", 1996.

[Launch] PERFECT Consortium, Launch of the Experience Factory, D-BL-Launch-2-PERFECT9090, ESPRIT Project 9090 "Perfect", 1996.

[NASA90] NASA/SEL. "Manager's Handbook for Software Development". NASA/SEL, Goddard Space Flight Center, Greenbelt, Maryland, November 1990.

[NASA92] NASA/SEL. "Software Management Environment (SME)". NASA/SEL, Goddard Space Flight Center, Greenbelt, Maryland, September 1992.

[NASA94] NASA/SEL. "Software Measurement Handbook". NASA/SEL, Goddard Space Flight Center, Greenbelt, Maryland, July 1994.

[PEF] PERFECT Consortium, PIA Experience Factory - The PEF Model, D-BL-PEF-2-PERFECT9090, ESPRIT Project 9090 "Perfect", 1996.

[Pete85] Tom Peters and Nancy Austin. "A passion for excellence". Harper Collins Publishers, London, 1985.

[PIA] PERFECT Consortium, PERFECT Improvement Approach, D-BL-PIA-2-PERFECT9090, ESPRIT Project 9090 “Perfect”, 1996.

[Platform] PERFECT Consortium, A Platform for supporting the PERFECT methodology, D-BL-Platform-2-PERFECT9090, ESPRIT Project 9090 “Perfect”, 1996.

[ProcMod] PERFECT Consortium, Process Modelling, D-BL-ProcMod-2-PERFECT9090, ESPRIT Project 9090 “Perfect”, 1996.

[Romb93] H. Dieter Rombach, Victor R. Basili and Richard W. Selby, editors. “Experimental Software Engineering Issues: A critical assessment and future directions”. Lecture Notes in Computer Science Nr. 706, Springer-Verlag, 1993.

[Seng94] Peter M. Senge. “The fifth discipline: The Art & Practice of The Learning Organization”. A Currency paperback published by Doubleday, 1994.

[SPC92] Software Productivity Consortium, “Using new Technologies”. SPC-92046-CMC, 1992.

[SPC93] Software Productivity Consortium, “Managing Process Improvement”. SPC-93105-CMC, 1993.

[SPC94] Software Productivity Consortium, “Process Engineering witht he Evolutionary Spiral Process Model”. SPC-93098-CMC, 1994

A booklet from the

Perfect

ESPRIT project 9090

Handbook

Experiences with PIA

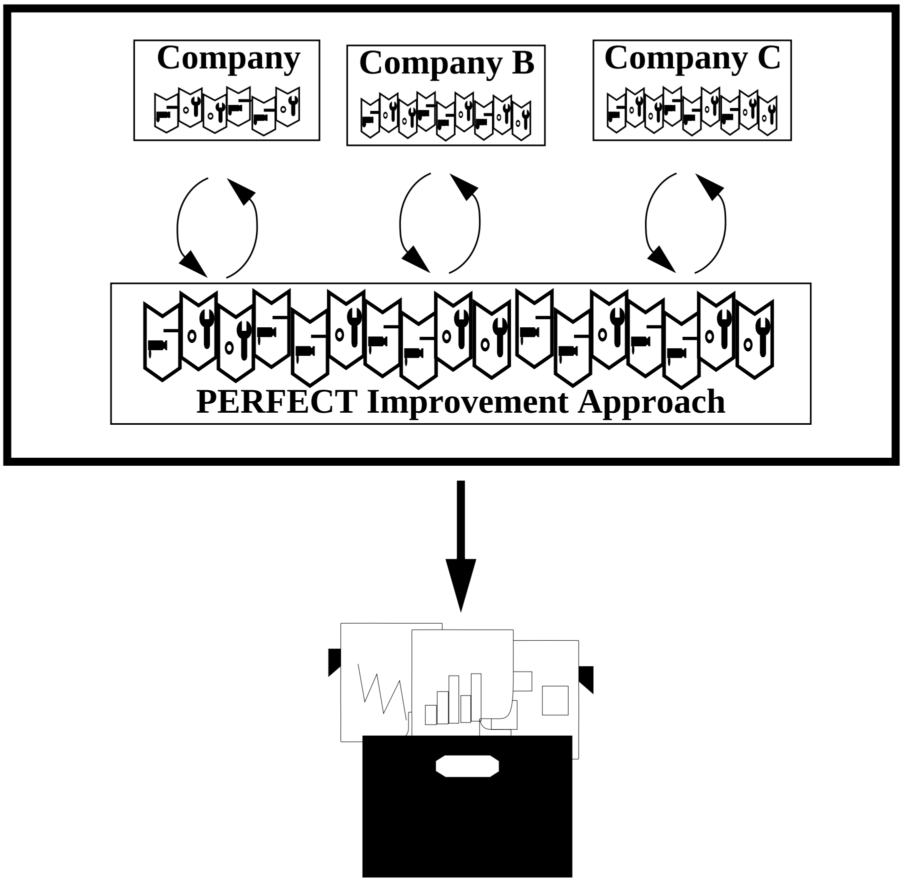
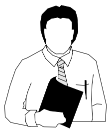


Table Of Contents

1. Executive summary



It has been a well-recognized fact in manufacturing technology for quite some time now that well-structured, easily understandable and thereby controllable manufacturing processes are an essential prerequisite for the achievement of quality and productivity standards. Now these ideas are applied to the field of software development. The quality of software products is closely linked to the maturity and the quality of the accompanying engineering process. High software quality can only be guaranteed by systematic software development processes and so the primary step towards maintaining market competitiveness is establishing and maintaining programs for continual optimization of software development processes on the basis of measurable project experience. The PERFECT Improvement Approach helps in obtaining this goal.

Performing an improvement initiative according to the PERFECT improvement approach leads to establishing a stable, systematic and continuous improvement program. Depending on the market the companies operate in, the maturity of the organisation, and the quality level they have to achieve the instantiations of the PERFECT Improvement Approach vary considerably.

Robert Bosch GmbH, Siemens AS and Q-Labs/Ericsson were application providers for the PERFECT project. All application projects were in the area of embedded systems, ranging from telecommunication to automotive electronics, and had similar overall improvement goals. However size and structure of the application projects were quite different: comparatively small, short, single site projects as well as large, long, multi site projects are covered.

Information about how the PERFECT applications set up their improvement programs is given in the following chapters. This includes information about the effort they spent, the strategic goals they wanted to achieve, the derived project goals and how they succeeded in establishing a continuous improvement program. Knowing how the PERFECT Improvement Approach was applied and the experiences were gained leads to a better understanding of the approach and helps in introducing and maintaining an improvement program using the PERFECT methodologies.

2. Introduction

During the development of the Perfect Improvement Approach and its methods industrial pilot projects were used to evaluate the developed methods and tools. The experiences gained there helped in refining and maturing the methods and increased their usability for practitioners through providing questionnaires, tutorials and guidelines.

Experiences also form the core of an effective improvement initiative. Normally the experiences gained serve to optimize the software development process. The experiences gained by the application providers, however, shall help you to introduce and establish a software process improvement initiative according to the PERFECT Improvement Approach efficiently.

2.1 Purpose of this booklet

The purpose of this booklet is to highlight and present the experiences made by the application providers using the PERFECT Improvement Approach (PIA) during the PERFECT project. Application providers were Bosch, Siemens, and Q-Labs with several projects in different environments ranging from telecommunication to automotive electronics. We show how the PERFECT Improvement Approach was applied and the experiences gained, which lead to a better understanding of the approach and help in introducing and maintaining an improvement program using the PERFECT methodologies.

This booklet answers the following questions:

- How was the PERFECT Improvement Approach applied?
- Which problems occurred and how were they solved?
- What is the benefit in applying PIA?

Furthermore lessons learned and recommendations how to apply PIA are presented.

2.2 Context

For understanding the experience stated in this booklet information about the basic concepts of the perfect improvement approach is needed. The main concepts are only mentioned here. For details, explanations and information about the connection of elements of the PERFECT Improvement Approach please refer to the corresponding booklets:

- Booklet on the PERFECT Improvement Approach [PIA]
  The PERFECT Improvement Approach helps in introducing and maintaining company-specific process improvement programs. It can be looked at from three different perspectives: a methodological perspective, covering the activities involved in systematic improvement, an organisational perspective, which addresses the roles and organisational entities necessary for systematic improvement, and a functional perspective, which deals mainly with tool support and other support resources required for systematic improvement. Generic

models for each perspective make the improvement principles operational and form the core of the PERFECT Improvement Approach.

- Goal-Oriented Measurement using GQM [GQM] (p. 4 - 10)
  The GQM processes are specialisations of the methodological perspective and cover measurement-aspects. A GQM product is every kind of document or explicitly represented information which is relevant within the GQM process, among them are the following: GQM goal, GQM plan, GQM Abstraction Sheet, the measurement plan, and the feedback session report.
- The PEF Model [N-A1.1-B](p. 5 - 25)
  The organisational model underlying the PERFECT Experience Factory contains three organisational entities, the sponsoring organisation, software development projects and the experience factory. Within the experience factory three corresponding organisational entities or roles are defined, strategic improvement management, experience package engineering and project support. Strategic improvement management forms the interface to the sponsoring organisation, project support supports the software development projects, and experience package engineering deals with building and maintaining experience packages. Each of these organisational entities/roles has been further refined.
- APEL [APEL](p. 3 - 7)
  APEL, which stands for Abstract Process Engine Language, is a graphical formalism for process description and support. It provides a high-level language and a functional interface needed for the support of software projects and their improvement according to the PERFECT Improvement Approach.

2.3 Document Overview

A short summary of the application experiences is stated in section 3. Section 4 describes how Bosch applied the PERFECT Improvement Approach and the resulting experiences. In section 5 the experiences made by Siemens are covered whereas section 6 contains the experiences within the PICME project.

3. Overall experiences with the PIA

Robert Bosch GmbH, Siemens AS and Q-Labs/Ericsson were application providers for the PERFECT project. Although all application projects are in the area of embedded systems and have similar overall improvement goals, they differ considerably in how they instantiated the improvement approach due to their different characteristics. Size and structure of the application projects are quite different: comparatively small, short, single site projects as well as large, long, multi site projects are covered.

Within Bosch the application projects consisted of two phases. First developing an infrastructure for the product, which includes a stable hardware platform and a set of library modules, and then producing hundreds of variants of the product. The production of the variants is the focus of the measurement/improvement program and is very close to production in a factory environment. Siemens based the improvement program on a series of similar, nearly consecutive projects of about one year execution time each, which have the classical software development phases specification, analysis, design, coding and test. Experiences dealing with a multisite project were gained within the Q-Labs/Ericsson AS project with about one year execution time.

The application projects have used the PERFECT improvement approach since the start of the PERFECT project. As major elements of the improvement approach have been developed, refined and improved at the same time, the experiences do not cover all aspects of the improvement approach evenly. That applies particularly for the PERFECT Experience Factory (PEF) and the functional perspective. The approach has been highly appreciated, especially the methodology with the strategic component and the separation into project organisation and experience factory organisation.

The PERFECT Improvement Approach has been evolved and improved based on the experiences made by the PERFECT applications. Some of those experiences expressed the dire need of more guidance how to apply the approach. Especially the elements:

- the different phases within an improvement program and
- the iterative improvement process

result from the experience of the application providers.

Although those ideas are now integrated in the PERFECT Improvement Approach, they are described briefly because of their importance for setting up and maintaining a successful improvement initiative. The text in the subsequent paragraphs contains common experiences of the application providers covering different aspects of the PIA.

Phases of an improvement program

According to the experience made a division of the improvement program into several phases with different focus was necessary. The PERFECT application providers pointed out that understanding the software development process and subsequent guidance based on this understanding are prerequisites for achieving process improvement through changing the software development process in a controlled way. So as stated in the booklet PERFECT Improvement Approach [PIA] an improvement program has different phases. According to their experiences *explicit* improvement can only take place in the last phase where the software development process is changed in a controlled way. However due to better understanding and better guidance implicit improvement (i.e. better performance) can also be achieved during the early phases of an improvement program. In order to switch from one phase to the next it is usually necessary to perform *several* runs through a (part of a) process model. During the understanding phase these runs serve to understand and stabilize the current process (or process parts) and discover the possibilities for improvement. The duration of each phase depends on the duration of the process which is to be improved and on how many of those processes can be executed in parallel. Hence they determine the earliest point in time when an improvement due to changes can be achieved.

Incremental introduction of the improvement program

Establishing and maintaining an improvement initiative means getting familiar with the different aspects of the methodology including the key technologies GQM, explicit modelling and comprehensive reuse, and establishing the necessary infrastructure, especially assigning responsibles for the improvement program activities. Improving software development processes is a quite complex task, hence the understanding for how to apply the methodology, how to put the key technologies, especially GQM, into practice, and how to set up the necessary infrastructure in this particular environment is only achieved gradually. According to the experiences of the applications it is beneficial to *start with few, small projects, a limited set of goals and a small organisational entity* for the handling of the improvement program. Additionally especially the *iteration of the steps* of the methodological perspective turned out to be useful to accelerate and promote the understanding and to keep the motivation of all persons involved through providing early success. During the course of an improvement program the number of projects and persons involved, the number and complexity of goals and the size of the necessary organisational entity to run the improvement program can increase according to the level of understanding gained through the iterative implementation.

Improvement goals

The first step of an improvement program is to identify the improvement goals of the organisation. As experience showed that they are a critical success factor for the improvement initiative, they must be chosen very carefully. There is a close link between improvement goals, its inherent improvement potential and the benefit that can be achieved.

To give an idea of the problems attached imagine the following situation: Compare two imaginary improvement programs within identical companies, both improvement programs go through three consecutive phases - from understanding to changing. Both improvement programs have the overall goal to reduce time to market. In the first improvement program the goal to reduce rework is derived. So the whole development process with an average duration of 8 months is investigated. The second improvement program has chosen as subgoal to increase the efficiency of the test process. Here the test process with an average duration of 10 days is investigated. In the first case the overall benefit which can be achieved is based on the improvement potential of the whole development process and probably greater than in the second case where the scope of the improvement is limited to the test process. On the other hand in the first case the time frame in which any improvement can be achieved is quite long (three consecutive phases with the length of the development process i.e. approximately 2 years), whereas in the second case the results can be obtained earlier.

Strategic level

The strategic perspective strongly supports careful planning of the improvement program as a whole and hence provides the long term focus for the improvement program. Careful planning involves assigning well defined roles to each project within the improvement program. This long term focus leads to *lasting* improvements. This experience was stated by the ERATMOS project.

Organisational perspective

The main characteristic of the organisational model, which covers the roles and organisational entities involved in systematic improvement, is the logical or physical separation of project organisation and experience factory organisation. Siemens put the separation into practice assigning two persons to the experience factory organisation. One of them works also partly in the PERFECT application projects as product engineer, so establishing despite the logical separation the close link that is necessary to obtain good results. As the development group concerned is small this informal way proved to be beneficial.

However the instantiations vary considerably. Besides establishing the separation between project and improvement team already on the project level Bosch introduced measurement program responsibles, a Change Control Board (CCB) which evaluates the improvement proposals and accepts or rejects them, and a steering committee, which monitors the progress of the improvement program and provides guidance about its direction. According to their experience there has to be a tight connection between those two teams.

Both instantiations have been reported to work well. The type of instantiation depends obviously on the size of the project and the size of the development team. Common experience was that the close link between the experience factory organisation and the project organisation is indispensable.

Explicit modelling

Process modelling and goal-oriented measurement are used excessively during the early stages of an improvement initiative. So most of the experiences reported so far for the key technologies deal with goal-oriented measurement (GQM) and process modelling. GQM experiences are covered in detail within the next chapters by each of the applications.

Process modelling (including also modelling of the products, and roles involved in the software processes) forms the basis for any improvement program. APEL, the process modelling approach developed in PERFECT, was evaluated in off-line experiments by the PERFECT Applications. All of them found the modelling capability very good, Bosch stated that it is also the best tool they know of to model processes.

General experiences of the applications with explicit process modelling during the application projects using other tools (e.g. MVP-L) can be summarized as follows: The PERFECT improvement approach requires that all views of the process are considered when developing the process model. Small but annoying inconsistencies can be found and removed during the formal description of the current software development process. According to the experience of the applications, developing the product model before the process model proved to be useful. An already stable product model is helpful during the discussion of the process and helps keeping the focus on the process view. In most cases explicit modelling, which promotes better understanding, leads to implicit improvements such as better planning and more focused decisions during software development and project management activities.

4. The Bosch improvement program

Robert Bosch GmbH is a manufacturer of electrical and electronic components. It is divided into 4 major business areas, of which the automotive and telecommunications area account for about 75% of the approximately 150,000 employees.

K3 is one of the nine business units within the automotive area. It manufactures motor management systems for gasoline. The development departments that are the focus of this study develop control software for the fuel injection modules.

4.1 State before the PERFECT project

Bosch has introduced the following process improvement initiatives at the corporate level:

- „12 Quality Principlesorganisationescribing the Bosch Quality Culture
- „Quality Assurance of Control Unit Software“
- CIP: a continuous improvement program based on workshops

Although fault and effort statistics were being kept in K3 before PERFECT, it was felt that a consistent and consequent approach to measurement was lacking. The need for measurement for improvement was identified.

4.2 Improvement program

4.2.1 Goals of the improvement program

Following a strategic management decision, the following goals were set for K3 as a whole:

- Productivity increase of 30%
- Time to Market reduction of 50%
- Maintaining the Quality Standard „Zero Defect Delivery“

4.2.2 Projects within the improvement program

Two K3 projects have been studied in the PERFECT project. In the Project1, variants of an already existing product have been produced continually by a team of 5 developers. The Project2 has started in July 1995 and will continue for a number of years. A new product line will be developed by a team of 12 developers.

When one applies the approach, one must consider how one’s environment differs from either the norm or from the expectations of the approach. The question presents itself “What is significantly different about Bosch or about K3”. First of all, the application is an embedded system. The product complexity does not come from the software dimension, it comes from the physics of the application domain.

Secondly, as far as the hardware/software infrastructure is concerned, there is a single architecture that holds for one product generation. A product generation can loosely be defined to be a set of products that follow the same physical principles for motor management, have the same hardware and have a certain software commonality (programming language, process elements like test, specifications, etc.) The “project” therefore consists of developing an infrastructure (stable hardware platform, a set of library modules and perhaps a couple prototypical products) and then producing perhaps 1 to 2 hundred variants of the product in the space of 2 to 3 years. A product is defined internally to be one variant that is delivered to one customer. Note that in other types of software development, a product is often singular and is the end result of the specification, analysis, design, coding and test tasks. In the case of project2, these steps are employed both in the preparation of the platform and the production of the variants. Given the large number of products produced, the production of the variants has similarities to a production phase in manufacturing, which has no analogue in other types of software development.

This steady-state process; the production of variants was/is the focus of the measurement/improvement program. It is much closer to production in a factory environment than variant production for non-embedded system products.

In the overall process under consideration, there is not a lot of specialization of tasks. There are 4 developer roles: feature specifier, module developer, integrator and tester. Each role has its process, which is independent of (or more precisely, sequential to) the others.

4.2.3 PIA strategic level - first steps

A collection of data was available from the state before PERFECT. Additionally the characterization questionnaire was filled in. The goals were the same as stated above

- Productivity increase of 30%
- Time to Market reduction of 50%
- Maintaining the Quality Standard „Zero Defect Delivery“

4.2.4 PIA project level

Project1 products were being developed before the PERFECT project started. In this case the characterization task consisted of gathering existing process documents and developing an unambiguous and sufficiently detailed (for measurement purposes) process model, which was checked for consistency during individual interviews with each and every developer.

Lessons Learned

- All views of the process must be considered when developing the process model. As it happens, the interviews were only with the developers, while one of the GQM Plans was from the view of the Project Manager. The process model had to be expanded to incorporate some subtle nuances that only the Project Manager sees.
- A complete Product Model is necessary for process modelling. There was no Product Model at the start and it had to be developed through inter-

views with the developers. It is better to develop the product model before the process model, as it minimizes the confusion when the process details are discussed.

- Interviews over the process model are a good occasion to get application information before doing the GQM plans. It is very helpful during the GQM-Interviews.
- Use a formal language when specifying the process model. It is very helpful for discovering the conceptual inconsistencies that will plague a measurement program when not caught.

A number of weaknesses in the current situation came to light during the characterization step. Since a baseline is a necessary precondition for an improvement program, there could be no intent to introduce improvement measures. The problems had to be first understood and characterized. These were discussed with the different process participants and a set of goals relating to understanding and characterization was developed.

Given there was no GQM program in place at the time, the crossover to the model selection step was a bit indistinct. Going from consensus about the direction to a set of concrete goals is an iterative process.After establishing the goals, questions and metrics were derived. A representative for each view was chosen to act as an expert for the GQM Plan developer. More precisely, individual development team members were chosen to be responsible for each of the plans corresponding to the goals. The plans were developed, questions and then measures, iteratively through a series of interviews between the GQM Plan developers and the Responsibles.

Lessons Learned:

- It is important to nominate one person from the development team to act as an expert and be responsible for a GQM Plan. Without a constant and consistent discussion partner the GQM planner will have difficulties developing detailed definitions for terms and in developing hypotheses about complex relationships.
- Exact and detailed definitions of the terms used to formulate the goals should be developed before the interview process. Terms had completely different connotations in different environments. This lead to misunderstandings that were only discovered later.
- Textual representations of GQM plans are difficult to read for the uninitiated. This lead to the development of Abstraction Sheets.

The measures developed for the GQM plan were reconciled with the process model. This produced a plan which specified which measures were to be gathered at what point of time, or under what circumstances, during the execution of the project.

A set of questionnaires were developed so that the appropriate people could record the measurements as close to the event as possible. All project members were given the chance to review the questionnaires and change requests were honoured. This measure was taken to gain acceptance from the group as well as to hopefully increase consensus about the questions.

At the beginning of the project, the questionnaires were interpreted manually and the results were entered into a database. This was primarily due to the fact that there was no data-gathering tools available, but it also served the purpose of quickly establishing the appropriateness of the questions on the questionnaire. Since the entry was done off-site, this system suffered a marked rigidity with respect to changes in the process so that the quality of questionnaires was sometimes affected. Later in the project, a data-gathering tool was developed and validation occurred within the development group.

Lessons Learned:

- The questionnaires are only successful when they list all of the possible outcomes exactly. Although the Questions and Measurements were seem-identification of improvements. The initial feedback sessions dealt with validation and improvement of the questionnaires. Subsequent feedback sessions helped to identify the first improvements.

Lessons Learned:

- The feedback sessions last about an hour and a half, and occur in the middle of development. Given that they occur reasonably frequently, they disturb the normal development process, or at least they are not unnoticeable. There were problems relating to lack of attendance because of deadlines. It is important that all team members attend, but especially the Plan Expert and all the creators of the data discussed in the session. The validity of the interpretations can be affected when team members are not present. It can also destroy the usefulness of the session. It is therefore important the importance of the sessions be recognized by management and that the staff be given the time to attend the session.
- The other side of the first point is that there is usually more data than time to discuss them. The sessions must be carefully planned and the planner must ensure that the appropriate team members attend the session and that the session be directed at those who attend.
- Similarly to the first point, the staff must be given the time to attend the feedback sessions, but management must also specifically plan for the interruptions and „budget“ for them.
- The presence or absence of management can affect the effectiveness of the sessions. Some members were not as forthright as they could have been in the presence of their boss. Managers were present because the GQM Plan was developed for their view. So we have separate feedback sessions for management and staff. When there are differences in the resulting interpretations, subsequent joint sessions are arranged.

In contrast to the Project1, the Project2 is new. That is, the product line did not exist and that the development team must be formed. There was no existing process and although there are product and process models, they were not verified against practice. A process model was defined for the „development unit“ (more than Project2 and less than the whole business unit) which was supposed to be the new standard.

As this is a new project, the goals are essentially the same as for Project1; that is, to understand effort, reliability, reusability and functional consistency. Given that the goals were the same, the Project1 GQM plan could be and was adopted for Project2. Again, the GQM analysis was reconciled with the process model to produce the measurement plan. As the development program started, it became clear that the process had been too idealistically defined, and that the developers didn't work in the expected ways. Part of the problem lay on the fact that there are now 36 developers rather than the 12 that had originally been planned. The process model had to be adapted (redefined) to pass into the new conditions. This had the expected impact on the measurement plan and the data collection tools. In all, the measurement plan was delayed about a month.

The first feedback session had to be delayed because of incomplete data. The new developers who were brought into the project had not filled out the questionnaires properly. The major *problems* were:

- They had not properly understood the questions.
- They hadn't filled out the questionnaires at all because they didn't know when to do it.
- They didn't find the tool - it wasn't installed on their PC or it didn't catch their eyes in the menu

Finally the feedback session was held. It was more a simulation to show the developers how it would be done and to provide motivation for collecting the data diligently. There was also a retraining session.

The business unit was reorganized and the feature developers were assigned to a department which is not a participant in the Project2 program. The parts of the measurement program that dealt with feature development; what might be termed the requirements and specification phases, thus became irrelevant.

During the analysis it was discovered that for many measures there was no data or too little data for a reliable estimate. This phenomenon had several *causes* other than those outlined in the previous section:

- The questionnaires had too many questions and answering them took too long
- The measurement plan was developed for a fully functioning department. Since the department is actually in start-up and the process not yet stable some of the questions were inappropriate. Some examples:
    - Questions about the distribution of internal and customer change requests were not relevant as at that time all development efforts were being directed towards developing a common platform. There are no external customers right now.
    - Questions about coverage of the total functionality were not relevant because many of the customer feature requirements are incomplete.

It should be noted that the unnecessary load caused by the irrelevant and inappropriate questions did lead to some dissatisfaction with and rejection of the program from the team members.

One of the most important insights coming out of the first proper feedback session for that project is that, because the formal process was being followed more closely by the software developers, the feature developers were getting more feedback about their errors. The resulting load overloaded them and the work had to be redistributed. Here is an example of how process conformance in the early stages of a measurement program can lead to process change, possibly leading to process instability that negates early improvement measures.

Lessons Learned:

- Take things one step at a time. Planning for an ideal situation at the beginning of a program with an immature process leads to wasted effort and dissatisfaction in the team.

In the Project1, one third of the errors found after delivery of the product couldn't have been caught by testing. This was considered to be a very significant statistic. The result of this is that after the measurement program has been introduced in Project2 and is reasonably stable, an inspection process will be introduced so that errors can be caught before testing.

The development team had a number of phases, where they had to introduce changes to process elements that were not functioning according to the mechanisms they were expected to use. So far the development team has not replaced process elements with other process elements that use other mechanisms.

As was described, hundreds of products are produced during the “production” phase. The process is fixed and rigid. Each developer repeats his part of the process on a regular basis, which is ideal for quick stabilisation and optimization. The goal of the process is to produce new products quickly, cheaply and reliably. Perturbations due to the introduction of new technologies are not welcomed unless they offer the solution to an immediate and dire problem. We have therefore adopted the philosophy that the production process should remain as stable as possible between generations. Revolutionary improvements will be introduced with the next generation.

4.2.5 Experience Factory

The experience base consists of the PERFECT reporting documents, the Project1 and Project2 program documents and K3’s process and product documents.

The change-over when Project 1 had just ended and Project 2 was just beginning was done somewhat informally by the GQM program manager. The need to have people for packaging experience for reuse is not immediate. Nonetheless the Experience Factory organisation is being formed. Later on two engineers were added to the Project-level Experience Factory; one is responsible for measurement/improvement and the other administers the inspection program.

4.3 Experiences

4.3.1 GQM experiences

Although project1 and project2 are closely related, and in the beginning they appeared to be very similar, the addition and then deletion of functional specification, the addition of the lower-level modules and changes in the product mandate (drastically increased production demands) all made them very different.

We probably realized GQM in a textbook manner, from the use of interviews, to the use of process models and the development of measuring plans, to the use of feedback sessions for the analysis of the data.

At the beginning, the GQM plans for Project2 were a logical extension of the work done in Project1. Some work had been done in characterization and the goal, in the first step, was deeper understanding. Plans for functionality coverage and reusability were added to the plans for effort, and reliability and the latter plans were refined to give more precise answers. The GQM plans were revised after the reorganisation. This was done by re-examining the hypotheses and then carrying the changes forward to the measurement plan. Although the hypotheses remain substantially the same, some rigor has been lost because the GQM plans themselves were not updated because of time constraints. Although this is somewhat regrettable, it is likely to occur often in projects where there is more data to be processed than manpower to do it. It is therefore to be expected that the plans will not evolve for the better, but rather in the best case may fulfil their original purpose.

4.3.2 Development and production phases

It has just recently become clear that projects in K3 are actually in two phases: infrastructure (or basic development) and production (or reuse). There are certainly certain relationships between the product components that can be studied during the development stage, but the two phases do not seem to be very similar with respect to their processes.

The first 6 months of the project2 project were characterized by rapid, unforeseen growth and by major organisational restructuring. In the end, the measurement plans and their associated goal of understanding, were hindered by the chaotic environment. It is perhaps a lucky accident that the infrastructure was largely developed in this time frame. In any case, this phenomenon has just been recognized and the data gathered in this time frame was invalid anyways, so it is very difficult to say if the differences in the phases really were significant, and how.

In the future, the different natures of the phases must be better taken into account. It may be that there will be two separate improvement programs and that the QIP cycles will repeat when the phases repeat.

4.3.3 Introduction of separate teams

In our project experience, it was necessary to introduce the separation between the project team and the improvement team already on the project-specific level. The reason is that the introduction of a measurement-based improvement process requires a lot of specific knowledge that normally goes beyond the normal project budget and competencies.

On the other hand, it is also evident, that there has to be a tight connection between the two teams, just as between the project-organisation and the Experience Factory. It is indispensable to have deep knowledge of the development process that is going to be improved. Beyond this, the improvement team does have to have good working relations to the project team, because the whole measurement approach relies heavily on valid data. They are only obtained, if the general cooperation is well received and accepted by the developers. This exigency can be fulfilled for example by including some experienced developers in the support team and it can be assisted by locally integrating the project and the support team.

Process improvement (PI) has always been a separate department in K3. Members of the PI departments were members of the (PERFECT) support team, however. The support, or improvement, function has been assigned to the PI department and it is the intention that when the GQM/QIP/EF methodology is well-enough understood and when they have the manpower, that the PI department will realize and administer the improvement program and experience factory for subsequent product generations.

There is a strict separation between the responsibility to evaluate and recommend a proposal and the responsibility for carrying out the improvement. The support team is responsible for documenting the current situation, investigating cause/effect relationships, investigating possible solutions and possibly recommending a solution. The development process remains the property of the development department, however. They are therefore responsible for the implementation of the changes.

We have introduced a Change Control Board (CCB) with representatives from the management of the business unit, the development team and the support team. It is this committee's responsibility to evaluate the cost/benefit, technical and organisational aspects of the improvement proposals suggested by the support team and to accept or reject them.

There is a steering committee composed of representatives of the development department, the support team, the upper-management of the business unit, the corporate support organisation, academia and consulting. The committee is responsible for monitoring the progress of the support program and for providing guidance about the direction of the program.

4.3.4 Bias of strategic orientation versus ready-for-use concepts

The emphasis in the PIA is on strategic and long term concepts. In this respect, it is a valuable introduction into the spirit of the overall PIA framework. But from a practitioners viewpoint, this can turn out to be also a general shortcoming:

The importance of this shortcoming from a practitioners' perspective is also stressed by the fact, that in practice the general basis and the operational core of PIA is formed by concrete projects. Of course, the overall framework and organisational infrastructure is needed to install an organisation-wide continuous improvement process. But in order to become effective it has to be translated in concrete projects. Our evaluation in the two subsequent paragraphs (i.e. *Pre-Measurement assessment, Iterative Introduction)* shall help to bring more balance between strategic and ready-for-use concepts.

Pre-Measurement assessment of the organisation's maturity

PIA generally speaks of adapting the initial steps to the organisation's maturity. However, it does not give further help how to determine the maturity of an individual organisation. Our experience was, that the organisation's maturity really matters!

Some parts of the GQM-plans that had been elaborated at the beginning of the process, did not fit the process after a certain time because the process was still changing. Unfortunately, the right balance between process maturity and going into detailed creation of GQM-plans is left open in the PIA approacWÏrmicew ttm667 u.04

could not really answer. This too, had a negative impact on the motivation and the commitment of the project team for the overall goal of process improvement and forced us to make extra-investments into additional motivation-sessions.

Iterative Introduction of the measurement-program

Because of our experiences with introducing the measurement-program and adapting it stepwise to the concrete process, we prefer the following procedure that shall help to preserve the project team's motivation:

Make a first version of GQM-plans, questions and metrics etc. But *tell the project team*, that this is a preliminary-version, in order to avoid frustration, if not all of the questions and measures fit the process. Then evaluate the preliminary results together with the project team.

Feed back the results of the evaluation into the next version of the measurement program and start collecting valid data with valid questions with a motivated project team.

5. The Siemens improvement program

Siemens AS in Norway is an organisation of about 2.500 employees. It is organized as 9 different divisions, where each division is divided into a number of sections.

The organisation where this application takes place is the Defence Systems section in the Telecom division. The section employs about 60 persons, among them 20 software developers. Embedded systems which have severe performance and memory constraints are developed using a SDL CASE tool and also C.

5.1 State before the PERFECT project

The idea to introduce process improvement was developed in 1992. One year before the PERFECT project started we introduced a new process model based on Cleanroom in a large project (P1). In the same project we also introduced formal use of the SDL design language with automatic code generation. This was partly handled by engaging Q-Labs as consultants. These actions formed the first steps and served as basis for the following process improvement process. When PERFECT started a new organisation was formed. A full time resource was put on the job of enhancing what was already developed, and to introduce systematic quality improvement. Half a year later the project was strengthened by a person working part-time in PERFECT and part-time in development projects, which were denoted as “the PERFECT application projects”.

Among the data already collected were problem/change reports from all phases, review reports and time account plans containing hours per activity. Other data were available in the form of monthly status reports and final reports. The monthly status report has traditionally been used to administrate the project. The final reports contain information about what went according to plan, what went wrong, what should have been made better, how many hours were used in the different phases, and on different modules. This experience data are so far not used systematically, but instead regarded as useful input for the planning process of a new project.

5.2 Improvement program

5.2.1 Goals of process improvement program

The departments goal is to introduce “Systematic quality improvement”. This is regarded as the mean to fulfil the “Meta-goals”

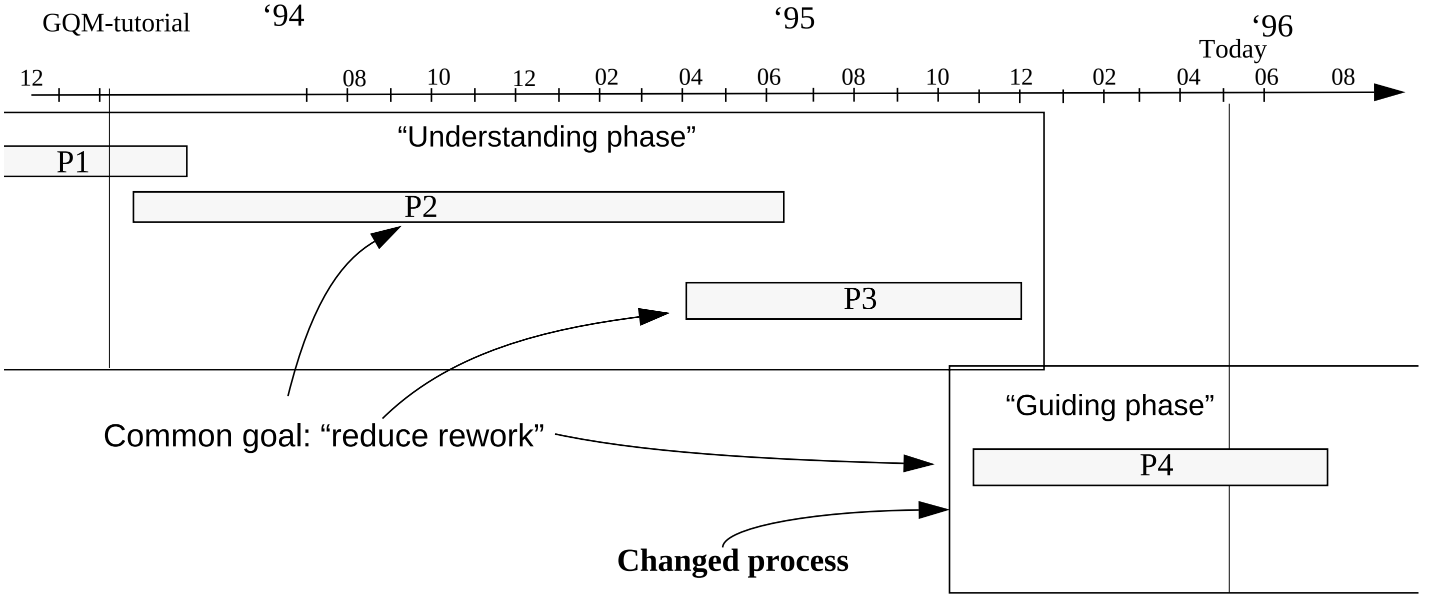
- Reduce development cost (Effort)
- Reduce lead time (Time To Market)
- Maintain Quality (Reliability)

These are regarded as long-term goals for the department. The short-term improvement goal for the department is more like “Describe and understand the development process”, since this is regarded as a prerequisite for introduction of systematic process improvement.

5.2.2 Projects within the process improvement program

As the first projects measured within an improvement program start usually without explicit quality models or accurate data from previous projects, the collected data can't be compared. So the first projects measured in any organisation will be more or less for the purpose of making a baseline and no measurable improvement will be obtained. However, a better understanding of the current situation will be gained. In addition the first experiments with the PERFECT-methods were mostly done to increase the understanding of the PERFECT ideas, which in the beginning were not described in any document.

Figure 1: Improvement program



PERFECT methods have been applied on three projects now (P2, P3, P4). The first project (P1) was a project where "everything" was changed: Cleanroom, formal SDL with automatic code generation, and intensive inspection of every project document were introduced. The next three projects were all commercial projects modifying existing products.

Table 1:

	P2	P3	P4
staff/persons	4.5	6.5	6.5
duration/months	12	9	9
effort/hours	7000	8700	6800
area	Communication	MMI and numerical functions	Communication

The amount of changes was, however, so extensive that parts of the systems needed to be redesigned. The application area was similar in all projects, however the P3 had a substantial part dealing with MMI- and numerical functions whereas the P2 and P4 projects concentrated on communication issues. The project manager in P2 participated in the original development of the product that is modified.

5.2.3 PIA strategic level - first steps

According to the PIA the improvement program started with the characterize step. During that step the „Characterization Questionnaire“, developed in the PERFECT project was filled in. A brainstorming session was held to find what goals to focus on for a limited GQM-analysis. The main goals of the organisation served as input. Sub-goal “*reduce rework*” was from this selected as a goal we wanted to start looking at. However, when using the brainstorming method you should keep in mind that people are very focused on their most recent experiences, whose influence on the projects to come they tend to overestimate. The problems with unstable requirements in the first project lead to a strong focus in the GQM plan for the next project on the effect of changing requirements. But there the requirements were stable. In the choose models step a process model was developed due to the lack of already existing quality and process models. This was done in close cooperation with the project manager and the project participants.

5.2.4 PIA - project level

We then tried to collect data from the still ongoing project (P1) in order to understand more about causes of rework and how much rework we were doing. This project was run as a traditional project in Siemens, and no particular effort was spent in collecting data. However, rework on documents from early phases had been identified as special activities in the activity plan. As more than half of the project was already finished, we were just able to make a first vague model of rework and of effort distribution. The effort distribution could be compared to previous projects and showed (as suspected) that the main workload was moved from test phases to analysis phases. The lesson we learned was that although we had collected a lot of data, no conclusions could be drawn. The data were inaccurate and important information was lacking so we were left still ignorant about the nature of rework. We learned that measurement has to be planned carefully in the beginning of a project and has to be related to goals that the project participants can relate to and see the need for. The project had an explicit process model, however, it is hard to measure without having an explicit process model for the change management process.

For the next project (P2) we used a more goal-oriented approach - starting with a GQM-tutorial on “reduce rework” which lead to a much more useful set of metrics. Effort on change, phase of discovery (i.e. correction) and phase of introduction for every change were identified as basic metrics that had to be collected during the project. At the same time we enhanced and tailored the process model and made it more stable. Data were collected according to the GQM-tutorial performed before the start-up of the project. A preliminary analysis of the collected data showed that changes coming from the specification documents (i.e. early phases) cost much, but we also learned that the number of (and also the cost of) changes in the category of *defects* was small, so rework wasn’t any large problem. The effort on rework of class ‘Defect’ was 135 hours in total, while the total project effort was 7.000 hours. From this data, together with data on effort per activity in the process model, we were able to describe a defect slippage model. We deduced that for this project the development process model worked well with respect to defect prevention. At the same time the data supports the conclusion that our process model is vulnerable to specification changes which proved to be expensive. After a thorough analysis we did not find any good candidates for process changes for improvement. After finishing the project P2 we found that we needed to expand the focus so that we would avoid missing useful data.

For the next project (P3) the main focus was still on reducing re-work, but we wanted to emphasize on learning what *requirement stability* meant for the performance of projects. This expansion came up after having performed a survey among all the software developers in the department asking them to come up with a prioritised list of the impacting factors on the efficiency of the department, which were used when formulating the GQM-goals for the projects to come. The question to be answered was: “What adds to the “minimum effort” of software development projects in Siemens Defence?”. This was the best and most sufficient way we could think of for how to go from high-level improvement goals to GQM-goals. This time quality models were derived from the GQM-analysis. We derived 68 Metrics from 8 GQM-goals. A measurement plan was developed and questionnaires for data collection were made. The different questionnaires were sent out on the end of every phase, in addition to a start-up questionnaire and a termination questionnaire. The process model was tailored to the needs of the new project. Data were collected according to the GQM-analysis performed before the start-up of the project. Most of the raw data on changes and review were put into a simple database for analysis. A more comprehensive database solution is currently developed.

5.2.5 PIA strategic level - analyse

Table 2: Defect slippage

			Origin (software development phase) of rework cause						
Resources spent (sw dev. phase)	Sum	%	1	2	3	4	5	6	7
1	-	-	-						
2	7.0	-	-	7.0					
3	5.3	1.1	4.0	1.3	-				
4	115.2	23.5	33.7	0.5	74.7	6.3			
5	6.5	1.3	-	-	1.5	4.5	0.5		
6	90.4	18.4	-	-	38.8	40.1	9.0	2.5	
7	273.3	55.7	1.5	6.5	5.8	170.2	65.3	-	24.0
Sum	497.7	100	39.2	15.3	120.8	221.1	74.8	2.5	24.0

	Sum	%	Origin (software development phase) of rework cause						
Resources spent (sw dev. phase)			1	2	3	4	5	6	7
%	-	100	7.9	3.1	24.3	44.4	15.0	0.5	4.8

For some GQM-goals we have (after three projects) understood enough to enter the guiding phase, while for others we still try to understand! But when it comes to pure “understand re-work” we have developed a “defect slippage model” that tells us a lot of how the development process is performed. The model covers both number of defects, effort used on defects, changes that are not defects and defects per module. Based on the findings from these types of measurements we have for the P4-project imposed changes in order to take a step ahead.

Suggestions for change

In the P3 project we had some disturbing results on both defect slippage and effort distribution. At the same time the project had a hard time to deliver on time (just made it) with satisfactory reliability. Extracted from the defect slippage data of P2 and P3 we can have a look at defects originating from phase 4:

Table 3: Defect slippage originating from phase 4 in P2 and P3

Defect detected in	P2 #	P2 %	P3 #	P3 %
4	8	15,5	0	0
5	4	7,5	5	5
6	29	57	45	40
7	10	20	62	55
Total	*53*	100	*112*	100

In addition we saw that P2 used 100 hours correcting defects originating from phase 4 while P3 used 183 hours. When these figures (together with some other findings) were presented to the project a quite engaged discussion took place. Some project members had already expressed concern with the way inspection and team building had suffered during summer holiday. This was during phase 4 and the lesson learned was simply: “ensure good team building and inspection climate”. When looking back the project members also agreed on that integration test wasn’t performed well enough. This was partly due to some problems with testing MMI-functions on the host machine (integration phase) and that the time constraint made them not reviewing the test-logs well enough. Another aspect was that Message Sequence Charts weren’t well maintained, so that the input to the activity of making test scripts wasn’t as good as it should have been.

Changes made

Based on the findings from the two projects the following changes were made for the P4 project:

- More emphasis on getting the teams/reviews to work
- Using requirement lists in all phases to ensure full design coverage and full integration- and system test-scripts coverage.
- Designer shall participate in test-log interpretation

Improvement

We want to distinguish between implicit and explicit improvement. Implicit improvement is better performance due to better understanding of the process, explicit improvement is the result of a significant change in the development process that leads to better performance. Both kinds are of equal importance, however the explicit improvement can usually be proved better as there normally exists data from before and after the change!

At Siemens we have experienced that a lot of PIA principles leads to implicit improvements. All these factors have shown to be important both for the P2 and the P3 project. The well defined and documented process model lead to better understanding, less mistakes and less uncertainties about the process to follow. Moreover it proved to be a new useful tool for education as it makes newcomers productive much faster than before. In combination with goal oriented measurement better estimation was achieved. No activities are forgotten and effort data from previous projects are more directly applicable. Additionally the reasons for *not* performing so well become visible, which in turn makes us capable of adjusting the process for the next project to come. Goal oriented measurement and the mature process model also lead to easy motivation for small changes in the way of working (without really changing the explicit process model).

The project (P4) has now completed the last phase within time and budget. Compared to the P3 project, the total number of defects discovered through the project have been reduced by 20%. This indicates that the focus on getting the review teams to work better have succeeded. The number of defects introduced in phase 4 is half the number from the P3 project, even though the product size (in terms of lines of code) is 1.5 compared to P3.

The use of requirements lists have been a success when it comes to the preparation of the system test. The resources used on the production of the system test document is reduced from 654 hours in the P3 project to 431 hours in the P4 project. The system test document from P4 covers more tests than the document from P3.

Table 4: Defect slippage originating from phase 4 in P3and P4

Defect detected in	P3 #	P3 %	P4 #	P4 %
4	0	0	4	8
5	5	5	0	0

Defect detected in	P3 #	P3 %	P4 #	P4 %
6	45	40	26	51
7	62	55	21	41
Total	*112*	100	*51*	100

The requirement lists have also been used to complement the test cases in the integration phase. In this way we hoped that more defects would be discovered on the host machine and not on the target machine. The project has reported that some of the additional host tests introduced because of the requirement lists did discover defects.

Table 4 shows that the percentage of defects discovered in the phase 6 has improved. Still the figures are not as good as those from the P2 project (Table 3). One possible reason brought up by the project is that the P4 project has developed 3 different products which communicate with each other. The integration test on the host machine, however, was run as three separate tests. The complete integration was not tested until the system test on the target machine.

In total, we believe that the data show that better review teams and the use of requirement lists have lead to fewer defects being introduced and that the defects have been discovered earlier. But we still see that the integration test on the host machine can be further improved. Detailed discussions with the project team in the feedback session we had lead to the proposal to work in the next project more on selecting good test data in the integration phase.

Effort

Rough estimates for executing the PIA on the project level in % of the project effort is found in Table 5. We will probably be able to save some time in the first three steps if we decide to go for a „standard set of metrics“ only.

Table 5:

steps of the improvement approach	Effort
steps 1-3	10%
step 4 (data collection within a project)	1%
step 5 (analyse)	3-5%

If additional goals come up there will be less to save. In step 4 we will have to work with the questionnaires to make them easier to use and the answers easier to interpret. But there is not much to gain. In step 5 we expect to gain some (1-2%) from the database tool we have developed

5.2.6 Experience Factory

The measurement plans and the analysed data have until now been documented and stored in different word documents and spreadsheets. The measurement plan was very hard to keep updated during the project. The analysed data from the projects was spread on many documents and it turned out to be hard to keep the overview of all the data. In order to solve these problems, a student has developed a prototype experience base. The prototype covers two areas, definition of GQM-plans covering goals, questions, metrics and values, and a “core part” covering the “core data” we want to collect in all projects. The core data includes project characteristics, resource data (hours per phase and activity) and the defect slippage model data. The prototype has a report function which allows to automatically generate many of the analysis we so far have done manually (in spreadsheets). The experience base shall be available to all employees in the department. Only a few persons will be allowed to write to the database. We believe that this is important in order to keep the data from different projects consistent and comparable.

Mainly two persons have been involved in the improvement program activities on the experience factory side. One had the roles of Product Engineer, Project Supporter and Experience Package Engineer. She works 50% as a Product Engineer in the projects. The other 50% she has been in charge of collecting data, analyzing data and package data. She has obviously the role of feeding results back to the projects and to help interpreting the data found. The other one had the role of Improvement Program Manager. He reports to the line organisation and maintains sponsorship. He has QA-roles in the projects to keep a close connection to the projects and the project manager. Currently we aren’t investigating or introducing new technologies, so the other roles have not been established so far.

5.3 Experiences and lessons learned

The principle of QIP is generally very good. The problem is to understand that the six steps are not necessarily executed in sequence, and by different organisational entities.

We have found it an advantage to have a very close connection to the projects. It really helps when we come to the analysis of data, where we have always found it necessary to incorporate “inside information”.

The danger is that we have not always found it necessary to make all characteristics or goals/questions explicit. It makes it possible to scale down the overhead effort of the improvement program, but it also makes it tempting to leave too many “obvious things” out.

When it comes to suggestions for improvements based on findings and analysed data the project participants themselves have lately been the most active persons. This we regard as being very positive, much better than if an “Improvement program manager” should have imposed changes on the projects!

We have seen an example on the opposite, namely that one project (the P3) should improve traceability of requirements by starting to use the tool RTM. The project manager agreed on the goal, but did not really approve the solution. It turned out project to come requirement tracing was achieved with no new tool at all but rather with some creative use of the word processor. And the new traceability turned out to be an help when it came to verification!

5.3.1 GQM Experiences

In the first project we were completely dependent on help from GQM experts. In the third project we were a bit overwhelmed by the effort needed to make a proper GQM plan and measurement plan. We discovered that the amount of GQM-goals (8) lead to so many metrics (68) that it was hard to keep them consistent with for instance the data collection forms. The need for tool support for handling metrics definitions, collection of data and analysis became obvious.

A GQM-analysis has been performed as a start into the understanding phase. This process was a time-consuming and difficult one, but still very giving. We have learned that to handle a large set of metrics will become expensive, and we have to put that in relation to our size and goals. Developing the GQM-plan was also difficult and time-consuming. Lesson learned for GQM-starters: ensure to get coaching from a GQM-expert, or use the new GQM-handbook.

Most of the participants found the interview session using the abstraction sheets promising. This is regarded as a useful tool for gathering and structuring experiences.

Some of our metrics is subjective (Example: “How much effort is used in re-planning due to requirement changes?”). We experienced that this kind of data should be collected as soon as possible after the trigger. Otherwise the value will be very limited. It is also important to formulate the question as precise as possible and with enough context information.

So far we have not used feedback sessions more than twice. GQM-experts tell us that this is too seldom and they are probably right. But we must be aware of that the time of a project participant is very precious. Another very important issue is that 90% of the most interesting data coming from a typical project are collected in the test phases. In our development process this is also a short period of time. Therefore the need for feeding back findings several times during a project seems limited. The least requirement we put on ourself is one feedback session during the detailed design phase and one in the end. If a preliminary analysis during the project shows some alarming results we should react and feed this back at least to the project manager.

Using GQM “by the book” either requires a very limited set of goals, or good tool support.

5.3.2 Conclusions - lessons learned

When reading these lessons learned one should bear in mind our characteristics, most of all that we are a relatively small department running projects with duration around 10-12 moths.

- The PIA requires patience; you must be willing to “stay in the understanding phase” until you are ready to make real explicit improvements.

- In order to really understand the process it has to be described and given a few (2 to 3) projects to stabilize. This takes a lot of time and effort!
- It is possible to improve significantly also in an understanding phase (“implicit improvement”), provided that you are able to combine the process model with goal-oriented measurement.
- The EF give the basis for detecting problems and suggesting candidates for change, but it is the project participants themselves that must identify and develop the specific change for improvement.
- PIA does not guarantee improvements, but the consequence of changes (and circumstances that just happens) will be monitored, so that you will have a better basis for making decisions for the next projects.
- If you organize the experience factory very closely to the projects you can afford to be more pragmatic (not so formal) to reduce the overhead of the improvement program itself. But be aware: it still will take more effort than expected, if not you are probably too pragmatic!
- Use goal-oriented measurement, but don’t overdo it! We saw that we had to limit the GQM-effort so we ended up with some “core set of metrics” that are to be collected and analysed in all projects. These metrics are effort on phases, activities and modules and defect slippage. For the other data that we are collecting we have goals that are evaluated in a “goal-oriented way” on the start-up of new projects. If we discover problems that we do not understand well enough, or if we will impose substantial changes, GQM will probably be used again.
- Without a common process model, it is very hard to compare data from different projects. Example: in order to compare the distribution of resources through the development phases, the activities must be the same.
- It is hard to achieve reliable data if the data collection cannot be connected to a specific activity in the process model. Example: our defect slippage model is connected to the Change Management process, which is a well defined process. The data collection is then clearly defined and integrated with the execution of this process. For more subjective data we have used questionnaires asking people after each development phase. The accuracy of the data is limited by the granularity of the process model.
- We have had some problems using questionnaires to collect data. We have not put enough effort into formulation of questions. This has lead to different interpretations by different data providers. Another problem with the questionnaires have been that we have distributed them too late, i.e. data have been lost.
- It is important to have a role which has the responsibility of packaging the analysed data into the experience base. This role has to ensure that the data are harmonized and consistent.

The challenge in the near future will be to convince the management that an explicit Experience Factory organisation has to be established. It is vital that the expenses of such an organisation will not exceed its benefits. The software-group at Siemens Defence systems is a small one (20 persons), so we must be careful not to overdo things. There will be a lot of limitations on the number of different explicit models, the granularity in the models and metrics, the number of advanced metrics and so on.

Hopefully we will be able to spread the methods and ideas to other parts of Siemens, not only to software organisations, but also to other process oriented parts of the organisation.

6. The PICME project

Ericsson's 85,000 employees are active in more than 100 countries. Their combined expertise in switching, radio and networking makes Ericsson a world leader in telecommunications. Ericsson is a knowledge-based company with 120 years of experience in telecommunications.

Ericsson is active in almost all sectors of the telecommunications field. The company is divided into five business areas, i.e. Public Telecommunications, Radio Communications, Business Networks, Components and Microwave Systems. These five business areas have a common core technology and strategy. They cooperate closely and to a very large extent provide one another with products and services.

The target product is embedded software for mobile phone systems (basestations). Software development is distributed over different sites in Europe (Germany, France, Greece, Spain) and US. The size of the project is approx. 300k man-hours of design work within approx. one year. The design work is carried out by 3 to 6 design teams at each site with 5 to 6 designers per team.

6.1 State before the PERFECT project

The PICME project is an improvement program for a project adding functionality to an already existing product. In the project (x-1), which preceded the target project some process improvement initiatives were undertaken, inspired largely by ideas and techniques from the Cleanroom process, namely the use of teams, increments, and reviews. The introduction of these new techniques created difficulties for the project requiring a lot of additional effort to assimilate the new ways of working. It seems that the difficulties encountered in project (x-1) were partly due to an ad hoc approach to process improvement, so the target project improvement program was based on the systematic improvement model developed by the PERFECT project.

The organisation has a program going on for measurements of productivity, quality and lead time. The measurements were not so well defined which created problems when analyzing the results from different sites (sub-projects). At least they were not detailed enough to identify improvement areas.

The experiences from earlier projects were not used in a *systematic* way as suggested in the PERFECT approach. However, explicit models and process improvement concepts are in place, used and maintained, e.g. MEDAX, the software development process (waterfall oriented), CME-20/CMS-40 product line, PROPS project model (incorporating Demming's Plan-Do-Check-Act process), quality models for productivity, quality and lead time, TEMPUS time reporting system, and DELTA, as an "experience base". It stores all kind of project information, e.g. intermediate and final project (progress) reports, measurement data (PQT), inspection records, effort data (per activity) etc.

These captured experiences are kept in an experience base. However, they are mainly used for project planning and project execution, less for project guidance and only implicitly for (systematic) process improvement.

Initially, the goals of the PICME project were simply to provide support to the Target project to enable it to achieve its performance goals in terms of

- Productivity (20% higher than in previous projects),
- Quality (a factor 2 better than in previous projects), and
- Lead time (to be reduced by approx. 30% within the design phase, compared to previous projects).

These PQT goals translated, based on the analysis and experiences from previous projects, into the following goals for PICME:

- To introduce a systematic approach to process improvement.
- To introduce measurement in a systematic way with the aim of building a quantitative baseline.
- To achieve a performance improvement in the Target sub-projects by increasing process adherence.
- To stabilize the development process used in the previous project and strengthen the usage of (Cleanroom based) technologies, as: teamwork, extensive reviews/ inspections, incremental development and work allocation, and feature oriented design.

6.2 PICME improvement program

One overall objective of the improvement program was to introduce a systematic way of running an improvement program following the PERFECT approach.

First, the focus was on understanding the development process. So only minor changes to the existing process were suggested. “To stabilize the used process” was the theme for PICME, which implies training, coaching and monitoring of existing processes.

To be able to run an improvement program in the complex environment of PICME an process improvement organisation had to be defined first. This multi-site organisation consists of:

- A multi-site Process Improvement Coordination Team (PICT), intended to coordinate and harmonize the activities across all sites in the project.
- Process Improvement Teams (PITs) in each site, to ensure that the process improvement program maintained close contact with the design teams, i.e. to ensure a good two-way flow of information, ideas, and feedback.
- Process Improvement consultants (Q-Labs)

This organisation was deliberately ‘bottom-up’, i.e. the driving force behind the program was intended to be the site PITs to ensure that the improvement proposals accurately reflect the real needs of the users.

6.2.1 PERFECT improvement approach (PIA)

A goal assessment, at the first GQM workshop, was one of the major steps taken during the initial phase of the PICME project. The target project and improvement project goals and their interdependency were clarified. The terminology of short term goals and long term goals was introduced.

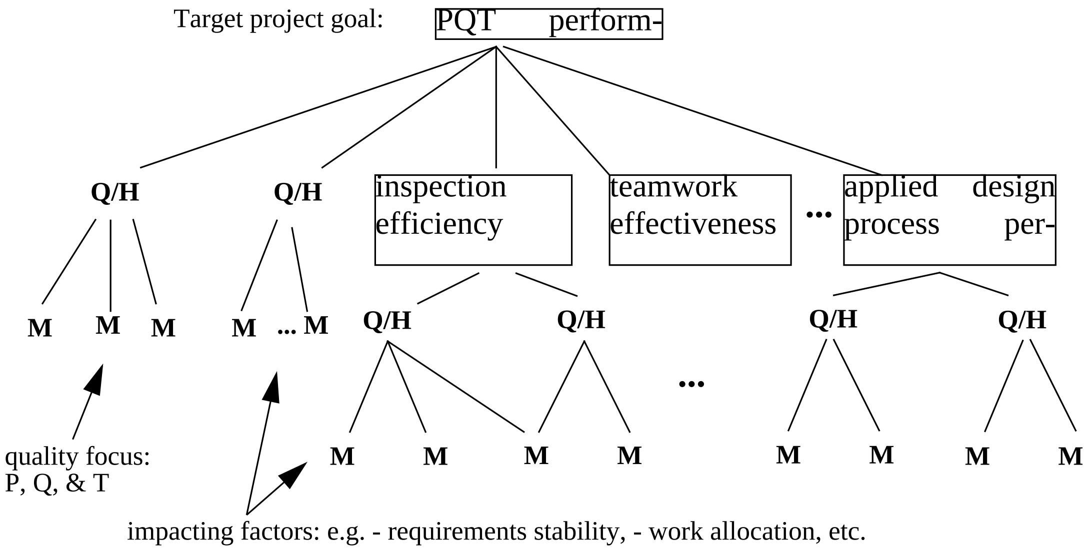
- Long term goal.
    - Over the next five years the goal is to achieve a general improvement in the organisation's productivity, to reach higher product quality with shortened lead time.
    - This should be accomplished by establishing a continuous and systematic process improvement program.
- *Short term goal.*
    - Within the period of the *Target* project, the goal was to reach the PQT targets specified and to understand the improvement proposals in a quantitative way by establishing a baseline.
    - That means to gain a better understanding of the impact on these targets of the process changes being made but also a better understanding of other process and environmental factors which impact these targets.

These two goals reflect the PIA organisational levels of improvement programs (methodological perspective) as they are

- the strategic level with the corresponding long term goal and
- the project level with the corresponding short term goal described above.

Following PIA's scientific method, described in the methodological perspective, the goals stated above were translated into GQM-based measurement goals. The following goal-tree (figure 2) formalizes the integration of strategic and project organisational levels into a GQM-based measurement goal.

Figure 2: GQM-based measurement goal (goal-tree).



However, referring back to the goals of the project and analyzing the performance of the project, the results of the PICME project are mixed:

- *To introduce a systematic approach to Process Improvement.* The PERFECT

Approach was presented as a framework for the program. However, the approach was not fully applied during the project. It represents a useful conceptual model and has been applied in structuring parts of the work and the organisation but was not fully understood by the members of the project and was not easy to apply in the way envisaged by the approach. The most significant results are the institutionalization and continuation of goal-oriented measurement using GQM and some organisational changes as proposed by the PERFECT Experience Factory (PEF).

- *To introduce measurement in a systematic way with the aim of building up a quantitative baseline.* The GQM measurement approach was applied in the project, it was set up with the concrete (see goal-tree above) project measurement goal (project level) and is carried out over a couple of projects (strategic level). Although the measurement baseline is incomplete in parts and the integrity of some of the data is questionable, it represents a significant achievement that the sites (sub-projects) were able to initiate such a measurement program for the first time. Most importantly, they continue to maintain quality models defined in the scope of the PICME project, e.g. PQT performance, inspection efficiency, teamwork effectiveness, applied design process performance.
- *To achieve a performance improvement in the Target sub-projects by increasing process adherence.* There are two processes to be considered, the “applied process”, i.e. the one that the teams actually execute, and the “defined process”, which is described in the process descriptions. Where process adherence is less than 100%, it indicates that there is a deviation between these two. There are two ways to reduce this difference:
    - bring the defined process more in line with the actual behaviour of the design teams by modifying the process descriptions
    - bring the applied process more in line with the process descriptions by modifying the team behaviour.

  The assumptions behind this goal are:
    - that process adherence can be increased,
    - that an increase in process adherence will improve performance.

In the PICME project, both strategies were used to try to close the gap. The design process descriptions were rewritten to try to simplify them and to increase their understandability and usability. The main means to modify team behaviour was the coaching effort which was only partially successful.

Measures of process adherence were introduced as part of the measurement program. Process adherence was generally higher than in the past (although there is still considerable room for further improvement) and it seems that this has had a slight positive impact on the performance of the project. However, the performance goals (PQT) were only partly achieved. The lead time goal was achieved, i.e. lead time reduced by 30%, although, it caused a high time pressure during the project. This had a negative impact on the quality goal, the sites, except one, did not reach their target. The achieved quality, measured after design, during function test and before operation, was worse than in previous projects, e.g. more than factor 3 at some sites.

6.2.2 PERFECT Experience Factory (PEF)

To refine and instantiate the PIA was a goal of the PICME project. So far, the achieved result (compared with the PEF) describes one possible example instantiation of the PIA. The road-map to obtain a PERFECT Experience Factory could be summarized as follows:

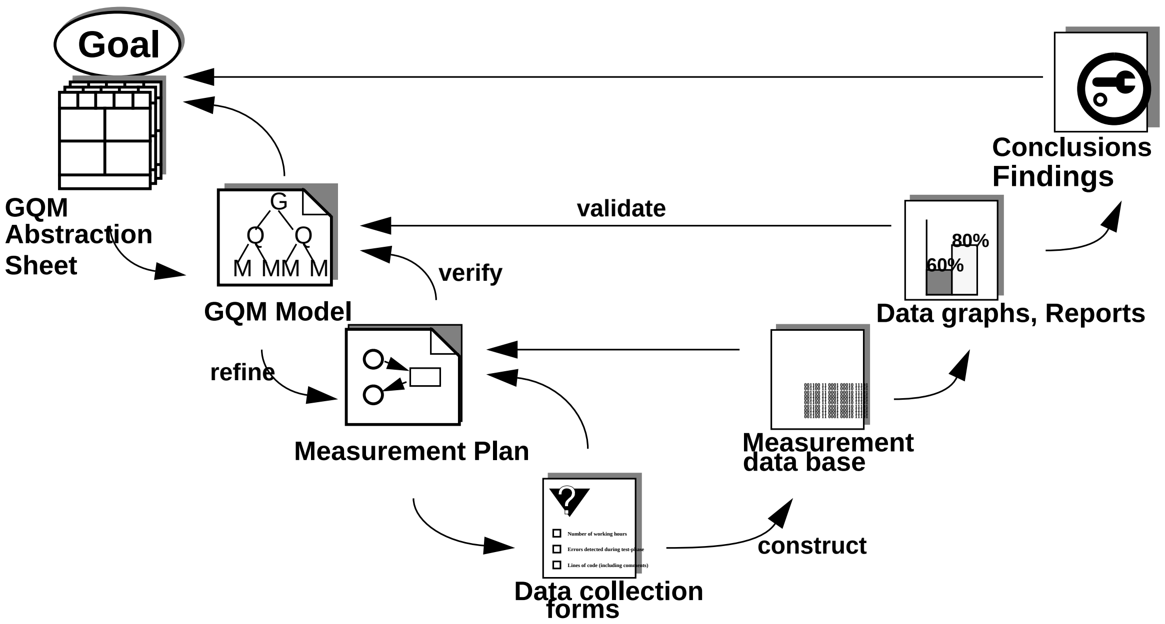
- *Obtain commitment.* The necessary sponsorship (and budget) for such an ambitious program is secured and in place. The achieved management directive ensure the work on continuation and enhancement of the PEF adaptation.
- *Establish structure.* The organisational structure of the PICME project, i.e. PICT and PITs, was taken as an initial structure and mapped to the PEF, e.g. Experience package engineering and Project support. A newly created Design Support Coordination Team (DSCT), which has the necessary vision, experience, seniority and power, is mapped to the Strategic improvement management entity as described in the PEF. It was decided to initiate and establish a node oriented, i.e. product line oriented, experience factory organisation.
- *Establish process.* Currently work instructions, function and role descriptions are developed which are inspired by the PIA and based on experiences from the PICME program. The experiences are packaged for reuse in terms of quality models (e.g. PQT performance, inspection efficiency, teamwork effectiveness, applied design process performance, etc.), process models (e.g. inspection process, GQM-based measurement process, etc.), and product models (e.g. GQM templates, inspection records, etc.). Experiences are actually reused, after some adaptations, in the ongoing target project and prepared to be reused in the coming target project which will be started in spring 1997.
- *Produce baseline.* The Target project performance and experiences provided the first draft baseline which is analysed and prepared to be changed systematically.
- *Prepare for process change*. Changes to the applied processes are proposed by the DSCT based on the achieved quantitative measurement results and the qualitative coaching results from the executed projects. Following the PIA ensures a *systematic* process improvement.

Establishing the organisational structure first (PEF) before applying PIA was a suc-

The models used to apply the GQM Approach are based on PERFECT results.

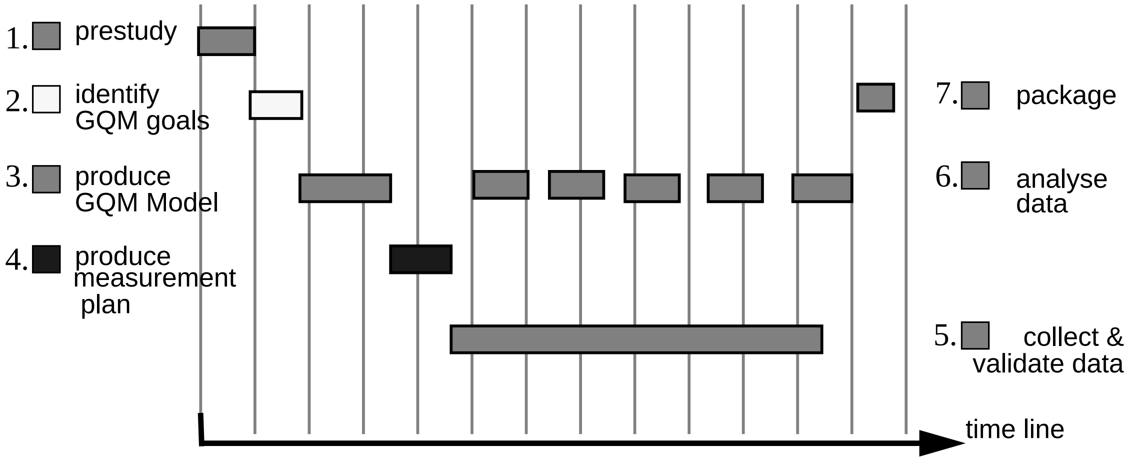
The following figure illustrates the main products and the relations between them details can be found in the booklet on goal oriented measurement using GQM.

A brief overview of the high-level/main process steps involved to produce the products described above is given below. This is a tailored measurement process defined prior to the process described in the GQM booklet. The effort mentioned to each step is qualitative rather then quantitative. It is dependent on the nature and number of measurement goals (reused vs. new goals, one vs. many), on the experience of the measurement team (first GQM measurement program vs. experienced), tool support (not at all vs. full support), etc. Exact figures were not collected for each step, but for the measurement program as a whole.



- **Prestudy**, i.e. identify inputs and environment, characterize organisation, identify improvement goals, characterize projects, identify project goals; with approx. 5% of total measurement program effort.
- **Produce high level GQM model**, i.e. identify measurement goals, quality focus, and impacting factors using GQM abstraction sheets; effort planned was 15% of total measurement program effort.
- **Refine and modify GQM models, i.e.** interview people, refine GQM models, validate GQM models; with approx. 15% of total measurement program effort.
- **Produce measurement plan**, i.e. define collection procedures, check against project plan; with approx. 15% of total measurement program effort.
- **Collect and validate data**, i.e. perform data collection procedure, validate data; with approx. 20% of total measurement program effort.
- **Analyse data**, i.e. translate results into (quality) models and perform feedback sessions; with approx. 20% of total measurement program effort.
- **Package**, i.e. analyse models for (dis)similarities, compare to goals, derive conclusions; with approx. 10% of total measurement program effort.

A time line overview, which is not to scale, relates the high-level process steps as follows. Steps 1 to 4 could be characterized as set-up activities which are needed in order to plan the measurement program, steps 5 and 6 are typical ongoing or periodic activities which are needed to execute the measurement program. Step 7, packaging, reports the conclusions and experiences in light of the measurement goals set during the planning step. The next chapter describes how the activities were performed distinguishing between measurement planning, data collection, data analysis, interpretation & packaging.



6.2.3.2 Execution of the measurement program

Measurement planning - prestudy results

During the prestudy phase of the measurement program some organisational and project characteristics were identified which are considered to be relevant for objectives of the measurement program, i.e.

- Although experiences with measurement in general are available from the PQT measurement program, the organisation had no experience with goal-oriented measurement. It was introduced for the first time into a project.
- A half day tutorial on goal-oriented measurement was held with participants from the GQM team (the PICT), and the line, i.e. target project representatives.
- The scope of the measurement program was set to Target project and was limited by the PICME-20 process improvement project. PICME-20 focused on improving the design processes, inspections, team work, work allocation, etc.

Measurement planning - GQM model development

The GQM Model was developed in several working sessions by the PICT in an iterative way. The selected measurement goals aimed to understand the impact of the selected improvement areas on productivity, quality and lead time. The result of this phase was a multi-site GQM Model with PQT as a top level GQM goal and inspection efficiency, team work effectiveness, work allocation fitness and applied design process performance as GQM sub-goals.

Initially one generic model for all sites was developed, but changes and refinements due to measurement plan evolvement and site characteristics led to site-specific variants of the model.

Measurement planning - measurement plan development

Each site was responsible to develop, based on the generic GQM model, its own measurement plan. The measurement plan had to be specific due to different (sub) project plan characteristics at each site. Although the PICT initially started to develop a measurement plan which was valid for all sites, a considerable amount of time passed until local adaptations at each site were defined.

Data to be collected was planned to be done through existing sources such as DELTA (experience base), time reporting system, inspection record, etc. Minor modifications to the inspection record were necessary to collect the data for the metrics defined in the GQM model. Few (subjective) data were planned to be collected by the means of questionnaires.

Data collection

The delayed definition of the measurement plan caused also a significant delay in the collection of the data, i.e. up to the late design phases. One site hasn't collected enough data to be able to analyse one sub-goal completely.

Data collection worked best there, where tool support was already available and the data to be collected was defined as part of the local quality plan, e.g. inspection records. In order to ease the data collection and to support the data analysis during the local feedback sessions, a spread sheet application was developed by the PICT. It constitutes a first draft of a measurement data base for storage, processing, validation and analysis of the collected data. It has been experienced, that data collection with the usage of questionnaires is not straightforward. The questionnaires where sometimes ambiguous or distributed too late to collect reliable data. Measures must be defined very precisely eliminating ambiguities. The usage of a glossary helped, but was not sufficient.

Data analysis, interpretation and packaging

Data from the sites were reported to the PICT according to the defined GQM model. The PICT maintained the site specific quality models, i.e. GQM models and corresponding data, in a spread sheet application which was used to prepare for feedback sessions at the sites.

Feedback sessions were held for each site. The purpose of the feedback session was to analyse the reported data according to the hypothesis stated in the GQM model. The PICT conducted the feedback sessions together with the GQM responsible, quality coordinator, and sub-project leader at each site. As a result of the feedback session structured interviews were performed with sub-project personnel, i.e. designers, team leaders, inspectors, moderators, quality coordinators, coaches. The purpose of the interviews was to validate the findings and conclusions drawn from the data analysis and to supplement missing quantitative results by qualitative information.

The main effort went into the analysis and packaging of the measured data. Currently the rough-set approach is analysed ("How could Rough Sets be used for

GQM-based Data Analysis”; Q-Labs, KL/QLS 96:0354) for its applicability and aid to

- package experiences in terms of rules, i.e. 100% correct classification of the objects from the learning set (measurement results obtained through the (GQM-based) quality models)
- investigate if right impacts were considered by detecting inconsistencies in measured data indicated through contradicting rules
- systematically compare hypothesis and actual experiences stored in rules
- prepare project guidance, i.e. early recognition and evaluation of project situations based on similar experiences from past projects

Continuation

Following issues were initiated based on experiences from GQM-based measurement in PICME:

- GQM-based measurement is continued, but tailoring and adaptation to local site specific standards and characteristics has still to be done.
- Node specific GQM process and product (templates) descriptions were developed.
- Costs and benefits of the GQM-based measurement program is evaluated more precisely. Roughly, the cost applying GQM was below 1% of the target project effort and approx. 20% of the improvement project effort. The cost include the refinement the GQM methodology, training and coaching of project personnel, set-up and maintaining of quality models, data collection and interpretation, feedback and interview sessions, and packaging of the final results.
- The ambition level of the measurement program is reduced and tailored based on experiences from PICME
- Feedback sessions are performed more regular.
- Project personnel and line management are involved more closely either through direct coaching or by regular project progress reports.
- GQM measurement responsible were appointed and established.
- GQM-based measurement is now becoming a local, i.e. site oriented, organisational (line) not a project issue any more. This will expand the usage to other product lines too.

Conclusions

Although the defined measures could not always be collected up to the defined detail already the model definition led to a better understanding of the project environment targeted by the process improvement approach. The quality models developed aid the analysis and packaging of the PICME program results.

An early involvement of the target project personnel into the measurement program, i.e. measurement planning, would increase the precision of the model and the reliability of the reported data. The requested effort would probably not exceed the actual value or would even be less, due to higher commitment from the involved personnel.

6.3 Experiences summary

A brief summary of what was used and experienced from the PERFECT Approach within PICME is given below.

PIA

- PIA was used for understanding and structuring the goals and the approach of the PICME program. It clarified the relation of strategic and project goals.
- PIA is an useful approach for making process improvement systematic.
- Packaging and reuse of experiences is not addressed well enough.
- To make the PIA more practicable it must address the characteristic and type of different organisational environments as well as the cost involved.

PEF

- PEF was used to aid the establishment of an organisational structure (roles, functions, responsibilities) of an improvement program. The existing organisational structure could be mapped to the PEF for better guidance. It is beneficial to have the organisational structure changed and settled before starting the improvement program.
- PEF makes the improvement program continuous.
- As an instantiation of the PIA it needs more refinement on the process level.

GQM

- GQM was used to baseline the project performance and, in a quantitative way, to understand the impacts of the technologies used and introduced. It helped to understand where potential improvement could be made.
- GQM is an useful approach to achieve evidence for improvement.
- An ambitious analysis task and the formal GQM process could make an improvement program to fail. GQM should be used smoothly and in a pragmatic way.
- It seems paradox that the PERFECT GQM Approach focuses mainly on the process and less on possible objects (‘product’) to be analysed. Addressing this issue more in-depth would increase the understandability and usefulness of the measurement approach.

Conclusion

The approach initiated by the PICME improvement program is continued in our organisation. The PEF is now gradually build up for one product line. Other product lines and sites are investigating its usage. GQM is extended in its use to other product lines and sites.

7. Bibliography

[APEL] PERFECT Consortium, *APEL (Abstract Process Engine Language)*, D-BL-APEL-2-PERFECT9090, ESPRIT Project No. 9090 “Perfect“, 1996.

[GQM] PERFECT Consortium, *Goal-Oriented Measurement using GQM*, D-BL-GQM-2-PERFECT9090, ESPRIT Project No. 9090 “Perfect“, 1996.

[PEF] PERFECT Consortium, *PIA Experience Factory - The PEF Model*, D-BL-PEF-2-PERFECT9090, ESPRIT Project No. 9090 “Perfect“, 1996

[PIA] PERFECT Consortium, *PERFECT Improvement Approach*, D-BL-PIA-2-PERFECT9090, ESPRIT Project No. 9090 “Perfect“, 1996.

The sections 4,5 and 6 are based on the „Application reports“

[N-A1.1-B] PERFECT Consortium, *Bosch Application Report*, D-N-A1.1-B-PERFECT9090, ESPRIT Project No. 9090 “Perfect“, 1996.

[N-A1.1-S] PERFECT Consortium, *Siemens Application Report*, D-N-A1.1-S-PERFECT9090, ESPRIT Project No. 9090 “Perfect“, 1996.

[N-A1.1-P] PERFECT Consortium, *PICME Application Report*, D-N-A1.1-P-PERFECT9090, ESPRIT Project No. 9090 “Perfect“, 1996.

Document Information

Title:	The PERFECT Handbook Vol. 4: Application of PERFECT
Date:	December 15, 1997
Report:	IESE-062.97/E
Version:	1.0
Status:	Final
Distribution:	Public