Software Process Improvement

Case Study

Funded by the European Commission Project Number 23873

Austria No.8 (English) **September 1998**

GRZ Linz GmbH Organizational and technological infrastructure for reuseable components

Overview

Genossenschafts-Rechenzentrum (GRZ) Linz G.m.b.H. participated in the ESSI project SPIRE and introduced a reuse-oriented project methodology. This activity will help to further improve the productivity and quality throughout the organization through an established reuse process. The objectives were achieved through software reuse supported by organizational measures, method improvements, and technology. The focus of this process improvement was to enhance the adoption of reusable software components mainly for developing graphical user interfaces and database routines in the PC development group. A further goal was to demonstrate the efficiency of intra- and interdepartmental exchange of reusable software.

The Organisation and its Environment

Genossenschafts-Rechenzentrum (GRZ) Linz G.m.b.H. is a service provider mainly for the Raiffeisen group of co-operative banks. Beyond the operation of a computer center for banks, the company is providing special services like networking management, backup solutions, technology introduction, training, and technical support. GRZ is developing software for all major platforms. The continuous improvement of the software process has always been a major objective of the company resulting in an ISO 9001 certificate, the use of proven methods and tools in all phases of the software life cycle, and a training program ensuring that the required skills are available. GRZ employs 139 people. Of these, 43 are software developers. Software projects can vary in size from small (around 1 software developer) to large (with up to 6 - 8 developers per project team).

Hermann Sikora, the CEO of GRZ, says: *"Quality is not just a catchword but has always been the prerequisite to satisfy the customers requirements. Quality is a result of continuously improving our development processes."*

Based on the assessment results, GRZ decided to establish a software reuse process within the organization. More specifically, the objectives of the improvement project were to

- increase the *productivity* to get products to the customers in a shorter time and at reduced cost, to
- foster the *reuse* of software, and thus to reduce *redundancy* in object-oriented software development, and to
- augment the *quality* of the products in terms of maintainability, portability, and reliability.

Hermann Lischka, Head of the Software Engineering service group of GRZ, says: *"We know that in order to better reuse the software developed in our company we have to develop an infrastructure that will help us to share the software across different projects and departments"*.

GRZ shows a well-balanced performance in all process areas investigated. The purpose of all processes is generally achieved. The mentor and the head of software engineering identified the following possible areas for improvement:

Risk Management:
In addition to the project management tasks performed, risks shall be identified, prioritized, and monitored in an organized manner.

Software Measurement:
The analysis of error causes, customer feedback, and product quality shall be intensified.

Effort and Cost Estimation:
The existing methods for effort and cost estimation shall be improved and new methods shall be introduced.

Software Reuse:
Crucial activities in the experiment were the extension of the software life cycle with a reuse-oriented methodology and the introduction of support tools.

SPIRE Partners are:

Starting Point

In the SPIRE initiation stage, an assessment of the software process was performed to determine the strengths and weaknesses of the current software engineering practice.

In the GRZ, an assessment workshop was organized with four participating groups: CEO, Quality Management, Software Development, and Project Management.

The groups were selected to identify potential differences in the view of the software process. Each team consisted of two to three persons. The task of the team members was to reach consensus when evaluating the various process areas, to analyze the results, and to draw preliminary conclusions.

Finally, GRZ decided to introduce an organizational and technological infrastructure for software reuse.

The measures were expected to lead to a sizeable reduction of coding and testing effort, to allow substantial savings of development time and cost, and at the same time ensure better product quality.

The focus of the experiment was to enhance the adoption of reusable components (mainly for GUI and database programming) in the PC development group. In a first step, it concentrated on (object-oriented) source code and program documentation.

Another objective was to show that intra- and inter-departmental exchange of reusable components is efficient. The objectives of the project are summarized in the following table:

Objective	Description	Improvement
DEV. TIME	Development time for new applications	-20%
REUSE	code required for new applications compared to current approach	-30% (for GUI and DB part of application)
MAINTAINABILITY	effort to implement typical modification request	-30%

The Improvement Project

The BMVC Application Framework

The C++ Application framework library BMVC was purchased and enhanced to provide an initial set of components for the reuse framework. BMVC is based on the Model-View-Controller Paradigm.

Josef Jungwirth is the key developer for the reuse library: *"Unlike other application frameworks BMVC shows a loose coupling of model, view, and controller. This helps us to decrease the dependencies among model and view, and so to strictly separate business rules from presentation issues. The use of the BMVC framework is therefore an important prerequisite to achieve a higher reuse rate and reduce redundancy."*

Reuse Library Management

In the course of the SPIRE project GRZ has developed guidelines and instructions on how to create, update and utilize reusable software elements. A reuse library and processes to produce, store and deliver reusable components have been established. The guidelines describe in detail how to create new elements, how to manage and reuse elements, and how to test newly developed elements. All project standards are published in the Lotus Notes Intranet to enable access to all relevant persons.

(1) How to produce reusable elements:

- Programming and documentation guidelines for new elements (C++)
- Classification and cross Referencing: The Cocoon utilities are used to process the application framework's C++ source files to automatically create a set of web pages that document the framework (classes, methods, global functions, types, ...). The programming and documentation guidelines for C++ had to be updated, as Cocoon requires a set of conventions in the C++ header files. In addition, the BMVC application framework was updated to work properly with the utilities.

(2) How to manage and use reusable elements:

- Adoption of MS VSS Tool: MS Visual Source Safe is used as the supporting environment to perform the configuration and version management (CVM) tasks.
- CVM Procedures: The German V-Model was used as a starting point to develop the procedures and guidelines for CVM.

(3) Test Procedures for new components:

- Evaluation of reusable components (providing a component evaluation guide and a test policy)

Training Package

A training package has been developed to introduce to all relevant people the BMVC framework, the management procedures, and the tools. A tutorial sample application will help to enable the productive use of the library.

Baseline Project
In this application experiment, the key person with know-how of the tools and procedures will encourage the people of the baseline project to utilize them. This is crucial for getting the whole personnel committed to the new methodology. Furthermore, the baseline project allows a detailed evaluation of the experiment's success.

The Results

The project demonstrates that SPI can be achieved through a combination of organizational measures, method improvements, and technology introduction. It is expected that in this case the organized, methodical and controlled reuse will further improve quality and help to deliver high quality solutions in diminishing time. It is expected that the follow-up experiences will help to learn more about reuse barriers, the scope of reuse, and the impact of the programming language on reuse.

In SPI an improvement project action plan should be feasible and address the special needs of the company. The period to carry out the experiment was limited to six months. This time span was definitely too short to carry out the improvement activities, to gain experiences, **and** to measure the benefits. For GRZ this means that precise figures on how the objectives have been reached will be available only after the official end of SPIRE.

Hermann Lischka, the head of the GRZ software engineering group says: *"SPIRE was a big help to initiate SPI activities in our company and to focus on the software reuse process. We expect a major business benefit when applying the procedures in several projects for a longer period of time"*

Lessons Learned

Assessment Methodology:
SynQuest is a computer-based questionnaire for guided self-assessments and provides a small-scale and efficient way to start an improvement project. The experiences from the assessment workshop can be summarized as follows:

- In a three-hour period, it was possible to evaluate the status of the organization and to detect potential areas for improvement. SynQuest therefore helped to reduce the cost and effort of the assessment.
- The assessment workshop actively involved senior management and engineering staff to integrate different viewpoints of the software process. This helps to create more ideas for further improvement activities.
- The questionnaire with the underlying help system eased the understanding and awareness of software quality management.

V-Model:
One major task in the experiment was to develop procedures and guidelines for the management of the reuse library. The mentor proposed to use the German V-Model as a source for this task. While the V-Model provides an impressive source of externally documented knowledge in the areas of software engineering, quality assurance, configuration management, and project management, it is difficult to interpret and implement the framework in a real world organization. The major problem is the language problem caused by different terminology used in the standard and in the company. This makes it often difficult to compare the company's practices against the idealized list of practices proposed in the standard.
In the project, we therefore decided to develop a set of use cases (user-oriented scenarios) describing typical situations occurring in configuration management. It turned out that the outline used in the V-Model to structure the submodels had to be changed to better fit the real usage scenarios of the organization. Nevertheless, it has to be mentioned that is worth to look at the V-Model documents and use them as captured engineering experience.

Cocoon, MS Visual Source Safe, LotusNotes:
The selected tool environment provided an adequate means to support the defined processes and reuse methodology.

SPIRE Management, Project duration:
The change from Schilling to EURO and the Year 2000 problem are high priority issues especially in the banking domain. There is always the danger that an externally funded and controlled project like SPIRE will be deferred due to business constraints. However, SPIRE helped in getting the improvement project started.

Plans for the Future

- GRZ expects to continuously learn more about *reuse barriers* and how to overcome them.
- *From C++ to Java*: The change of the mainstream programming language requires the adaptation of some of the developed guidelines.
- *Enhance the scope of reuse*: Reusable elements may also include software requirements and design documents. The intensified or obligatory use of an graphical notation (e.g. Unified Modeling Language) and the adoption of an object-oriented CASE Tool may be considered in the future.

Acknowledgements:

This case study is published by ARC Seibersdorf for the SPIRE project.

Thanks are due to the staff of GRZ Linz, especially
Hermann Lischka and
Josef Jungwirth

Genossenschafts-Rechenzentrum Linz (GRZ)
Goethestr. 80
4020 Linz
Telephone +43 732 6929 0
Fax +43 732 6929 220

and our mentor Dr. Paul Grünbacher

SPIRE:

more information is available on the web site
http://www.cse.dcu.ie/spire

SPIRE Partner contacts:

Centre for Software Engineering,
Tel:- +353-1-704 5750
Fax:- +353-1-704 5605

MARI (Northern Ireland) Limited,
Tel:- +44 1232 669500
Fax:- +44 1232 669800

Etnoteam
Tel :- +39 2 261 621
Fax :- +39 2 261 107 55

IVF
Tel :- +46 31 706 60 00
Fax :- +46 31 27 61 30

Austrian Research Centers - Seibersdorf
Tel :- +43 2254 780 3117
Fax :- +43 2254 72 133

Software Industry Federation,
Tel :- +44 1232 333939
Fax :- +44 1232 333454