



# Software Process Improvement Case Study



Funded by the European Commission Project Number 23873

Ireland No. 002 (English)

October 1998

**Overview** – *Peregrine Systems Ltd.* is a successful and fast growing software development company which provides a specialised range of Business Process Automation application software products to clients in the financial services and government industry sectors.

The company’s main business objective is the development and growth of its customer base. The impact of this growth is that the organisation is becoming increasingly involved in larger projects, therefore increasing the size of software development teams. Management at Peregrine became aware that their current approach to source code management would not be adequate to handle this change. It was becoming increasingly necessary to introduce source code management procedures and also the tools to support them. Additionally, a requirement was identified for an improved defect reporting and tracking system.

The purpose of this project was to improve existing procedures for source code management and defect control, and to identify and implement suitable tools to support these procedures. The aim was to provide an improved level of service to an increasing customer base.

The goals of the project were achieved over the period of the improvement project. Tools to support source code management and defect control were evaluated, selected, purchased and installed. In addition, appropriate procedures were developed and implemented.

Peregrine’s capability in the areas of source code management and problem resolution has increased significantly over the life of the project. This enabled the company to handle the added burdens of growth in their customer base.

## The Organisation and its Environment

Established in 1992, Peregrine Systems is now the leading provider of solutions for chargeback processing to the card payments industry. The company focuses on enhancing the profitability of card programmes with innovative software solutions.

ICS and MERCS, the companies strategic products, deliver competitive advantage to card issuers and merchant acquirers by automating the chargeback processing task. Both systems are equipped with a knowledge of the Visa and MasterCard/Europay chargeback regulations and thus remove many of the knowledge intensive tasks in the business process. Peregrine’s other case management tools address process fraud, voucher request management, ATM disputes and other cardholder disputes.

The company is also responsible for supporting several other products, developed both in-house and by third parties, over a large customer base spread throughout Europe. Peregrine employs a staff of thirty personnel organised into small project teams dedicated to handling specific tasks.

## Starting Point

To achieve its main business objective of continued growth, Peregrine identified specific business needs to be addressed.

- To support a larger customer base.
- To reduce the number of reported software defects.
- To manage multiple technical platforms.
- To improve the scoping and resourcing of projects.

An increase in the size of both their customer base and the technical platforms on which they deliver their software, resulted in an increase in the number and size of the projects undertaken. There was also a significant increase in the amount of software, documentation, etc. produced due to multiple releases and versions of the software.

In order to achieve these growth objectives, management identified two main areas in the software development process for improvement. These areas were Source Code Management (SCM) and Defect Control (DCS) which were confirmed as key areas requiring attention in the independent analysis carried out at the start of the project. Peregrine decided to implement an improvement project to address these concerns.

## The Improvement Project

Peregrine developed a definitive approach for the implementation of the improvement project.

The approach consisted of five basic objectives to be achieved consecutively.



These objectives were:

- To select a commercially available tool to support source code management and defect control.
- To install and configure the tools.
- To migrate existing data into the tools for use during the pilot project.
- To evaluate the performance of the tools.
- To develop a rollout strategy for other company projects.

Picture not included

**Michael Foley**

Michael Foley was appointed as Project Manager for this project. This was a part-time role with the remainder of his time being devoted to his main role of product support.

**Gary Ramsay, Software Development Manager** says: *'The choice of Michael as Project Leader was crucial to its success. In a previous company he had experience of both successful and unsuccessful source code management projects and this gave the project a tremendous start in narrowing down the products which were worth considering.'*

### **Tool Selection**

The identification of the most appropriate tools was achieved by assessing Peregrine's requirements and reviewing the functionality in commercially available tools. The definition of the requirements was used to create a checklist against which to score the commercially available tools.

The bulk of information about available products was obtained from the WEB, which included promotional materials from the commercial entities (independent evaluations contained in commercial publications). This information was used to gain an understanding of the strengths and weaknesses of the various tools. A subsequent comparison was made against the checklist of requirements, and an overall assessment made to determine the combination of tools to be selected. Additionally, compatibility between the two products was desired in order to integrate SCM and DCS functions. After final analysis, Visual SourceSafe and TrackRecord were selected as SCM and DCS tools.

**Michael Foley** says: *'We selected TrackRecord from Numega and Visual SourceSafe. The Key reason for our decision is that TrackRecord allows you to design your own screens, which meant that we could define different screen layouts for our diverse product range.'*

### **Installation and Configuration**

Both tools are functionally rich and in order to use them as productively as possible, it was important that they were configured correctly before use.

**Michael Foley** says: *'The initial plan was to organise training for TrackRecord through a third party entity. However, due to scheduling difficulties with that party, it was necessary for me to learn the program myself, and then train the staff. I thought this would be a problem but in fact it forced me to get to grips with the product quickly and I was able to concentrate on what I knew we needed.'*

Using the various options available in TrackRecord and SourceSafe a number of trial projects were set up. Various tests were conducted to determine the most effective configuration. Once the tests had been completed, an optimum Visual SourceSafe and TrackRecord configuration was set up for the pilot project.

Integration between the two products was the key to obtaining the maximum benefit of tracking defects to particular software versions. Although the installation of each product was relatively straightforward, integration between the two was a more complex procedure.

**Michael Foley** says: *'Integration between the two products proved to be more difficult than anticipated. Documentation provided by each of the vendors was not specific enough in terms of integration with other software packages.'*



### **Migration of data for Pilot Project**

Once the optimum configuration for Visual SourceSafe had been established, migration of data for the pilot project was a relatively straightforward process of importing modules into the system.

Under previous methods, different versions of the source code were archived for each incremental release. A decision was made to migrate only the current version of the source code. A set of standard procedures was developed for the use of Visual SourceSafe.

Although TrackRecord facilitated the entry of previously documented defect reports, it was decided that only new defect reports would be entered into the system.

The initial plan was to migrate only a pilot project to TrackRecord, and then to perform a company wide rollout. However, several other projects came into acceptance testing at that time. Typically, at this stage of product development a number of issues would be raised by the client. It was felt that TrackRecord would be ideally suited to handle these issues. Therefore, it was decided to take this opportunity to roll out TrackRecord immediately to all projects.

**Ruth Clark, Project Leader** says: *‘On previous releases we had tracked defects using paper based defect reports and a summary spreadsheet. We always ended up spending valuable time updating the summary reports, which could have been better spent resolving defects. This seemed to be a big improvement with accurate summary details always available.’*

Following successful migration of a number of projects to TrackRecord, cross training of staff members was conducted on a per team basis. The prior involvement of project leaders in the configuration process greatly assisted the training of team members.

### **Evaluation of performance**

During a six-week period of usage a log was kept of all issues raised by the project staff regarding the SCM/DCS. These comments included technical problems with the application, difficulties in understanding functionality and suggestions for improvement.

**Neil Watson, Project Leader** says: *‘TrackRecord greatly simplifies the production of various support statistics, such as the number of calls per customer per week, number resolved within x number of days etc.’*

Over the period of evaluation, procedures for using Visual SourceSafe were refined to accommodate conditions in the normal work environment. Team members of the pilot project felt that Visual SourceSafe greatly improved version control and source code maintenance.

**Aidan Leonard, Software Engineer** says: *‘I found that using Visual SourceSafe gave me greater confidence when making changes to code. I could be sure that I had the correct version and that my changes would be saved.’*

### **Development of Roll-Out Strategy**

Following the successful pilot project, the intention was to develop a strategy for the rollout of Visual SourceSafe to other projects. Firstly, it was necessary to investigate whether the existing procedures used for the pilot project would be suitable for use in other technical environments.

The comparison of projects confirmed their compatibility with the current configuration. This rollout strategy included a timetable for the rollout to take place over a three-month period.

### **The Results**

Over the duration of the project, Peregrine achieved its primary objective of obtaining a fully integrated SCM/DCS system that enabled the tracking of individual defects to specific lines of code. This system also provided an efficient means of managing the sharing of generic code over projects with common functional requirements.

Within the six-week evaluation period, it was proved that the SCS would greatly improve maintenance of source code and enable Peregrine to handle projects with larger teams, and to maintain a quality product.

**Gary Ramsay** says: *‘The new procedures and support tools are definitely an improvement over the previous paper based system. I feel confident that the high standards we set ourselves will be maintained in larger more complex projects, which we are becoming increasingly involved in.’*



TrackRecord was implemented across the entire company, not just in the pilot project as initially planned. At the end of the project, a staff survey was conducted to elicit specific comments with regard to the efficiency of TrackRecord in error tracking, reporting and any suggested modifications to the user interface and standard procedures. The survey results showed that 90% of the staff felt that TrackRecord improved Pergerine's performance in tracking and reporting defects.

The final independent assessment proved that the capability levels in the company had increased significantly in the areas of configuration management and problem resolution. The SPICE rating for configuration management went from 0 to 1.75 while problem resolution went from 0 to 1.5. The staff attitude survey taken before and after the project showed a positive increase of 1.4. This showed that there has been a positive change in the attitudes of the staff to SPI over the timeframe of the improvement project.

Positive aspects resulting from the improvement project itself are that management and staff now have more confidence that they have a true understanding of the software status and therefore can better manage the development. The ease of availability of information on status is also a great improvement.

### **Lessons Learned**

Effective training is a definite requirement to facilitate initial installation and configuration, as well as to enhance user confidence in the tools.

**Michael Foley** says: *'Integrating the two systems together took longer than planned. Although initially it looked like a straightforward task, it took several calls to the technical support helpdesk to make the product work together seamlessly.'*

Honest feedback from user staff assisted in refining the software configuration, which in turn increased the effectiveness of the system and provided a more user-friendly working environment.

### **Plans for the Future**

Peregrine intends to employ the new SCM/DCS as standard on all new projects.

Monthly reports on the status of reported defects will be produced for senior management to provide a greater visibility of the current status.

As a result of the feedback about process improvement, Peregrine plans to apply Software Process Improvement to other areas of the software development life cycle. Another project is already underway in the area of testing to review how tools can be used to improve testing practices.

**Brian Caulfield, Managing Director** says: *'This project has proved that process improvement can have an impact even in a short timeframe. It is an area where we will continue to devote resources as it can deliver competitive advantage in an even more competitive market.'*

#### ***Acknowledgements:***

This case study is published by CSE Ltd, Dublin for the SPIRE project.

Thanks are due to the staff of Peregrine, especially Michael Foley and Gary Ramsay.

Peregrine Systems Ltd.,  
Block 2, Harcourt Centre,  
Harcourt Street, Dublin 2.

Tel: +353 1 4780077  
Fax: + 353 1 4780122

*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

##### **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  
Fax :- +43 2254 72133

##### **Software Industry Federation,**

Tel :- +44 1232 333 939  
Fax :- +44 1232 333 454