

People, process and tools: A Study of Impact Analysis in a Change Process

Summary of Master Thesis
Artour Klevin, Lund 2012

The common problem with Software Configuration Management (SCM) and processes is that they lessen flexibility and adds man hours on paperwork instead of working on the product. However there are great benefits as well, for example, extra security in the case something goes wrong or ability to collaborate in big teams.

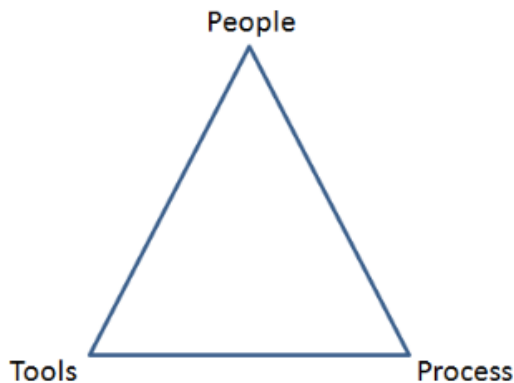


Figure 1 - The people, process and tools triangle

In this thesis three areas have been explored and we believe that they are fundamental for a successful business, they are: process, people and tools (see figure 1). The ambition of this master thesis is to give people who read it, a better understanding of possible ways to efficiently improve software development. Also create some concrete solutions for the site. With help of: better processes, smarter tools, more committed and educated people.

Software development companies have to offer support and maintenance of their products. Maintenance of a complex system is very difficult and costly process, to make it possible to maintain the software techniques like Impact Analysis is used. An Impact Analysis is a list of all deliverables that have to be updated

because of a proposed change to the system. It is used to help ensure that the system is performing correctly after the change is implemented and to verify that new faults are not introduced. In the change process the Impact Analysis is a vital part of the process, which helps to keep the system stable.

The perceived problem in the early stages of the thesis was that there were many simple systematic errors and that there was a need for faster feedback on the Impact Analysis to developers. The CCB also wanted to find simple errors faster and Safety wished for better quality on the information written in the Impact Analysis answers. The focus of the company was a tool as a solution to the problems. Therefore a tool solving the most pressing issues was developed, while at the same time informal interviews with developers were carried out to find other problems.

Initially developers complained about how laborious it was to write the Impact Analysis. But when they have talked about it for longer periods of time other problems got more focus. These problems were often only vaguely related to writing the Impact Analysis, like the problem with legacy code or that the Impact Analysis questions have grown uncontrollably and that the whole Impact Analysis is now more a burden than something beneficial. Also, there already exists tools for finding the answers to some of the questions in the Impact Analysis, but they are not used much. This has led us to the conclusion that the mentioned problems about the Impact Analysis are just indicators of deeper dissatisfactions. This leads to a wider analysis of the problems, which gave many new problems that had to be

addressed. Distributed across all tree areas of interest that are covered in this thesis.

One of the first findings was that the Impact Analysis is used more for documentation than economic and efficiency concerns, which is the common reason for having it. At the site it is used for a much wider purpose and gives great control of the change. One of the ideas about improving efficiency of the change process is making the process more lightweight during the initial step. In figure 2 the process is visualized first how it is today. Then below the suggested process is shown and last the difference in time spend on a case that is deferred is shown. Today the amount of work that needs to be done before the first decision can be made is too vast, which leads to long lead time before decisions can be made. There are two ways to improve the situation the company is facing today. The first is that right information should be written down at right time. Only the information that is actually needed for the evaluation should be gathered and rest can be gathered when the decision to implement the

change have been taken. The second is that the developers should be educated in how to find answers without looking at the code and implementing parts of possible solution before it is decided that it should be implemented.

Another problem found is that developers don't see the purpose of spending time on tasks that are waste of their time. The information gathered by the Impact Analysis has to be used so that employees have a reason to write good answers on the questions. There should also be better knowledge amongst developers why the process is the way it is, this can be done by educating them in a different ways. For example, involve them in the decision making when the process is created.

Our final conclusion is that a tool is the gateway to a solution, but without proper support of process and education of people. It will not perform at its full potential, because there are always deeper problems below the surface. A good solution will cover all three areas and is the way to a successful software development.

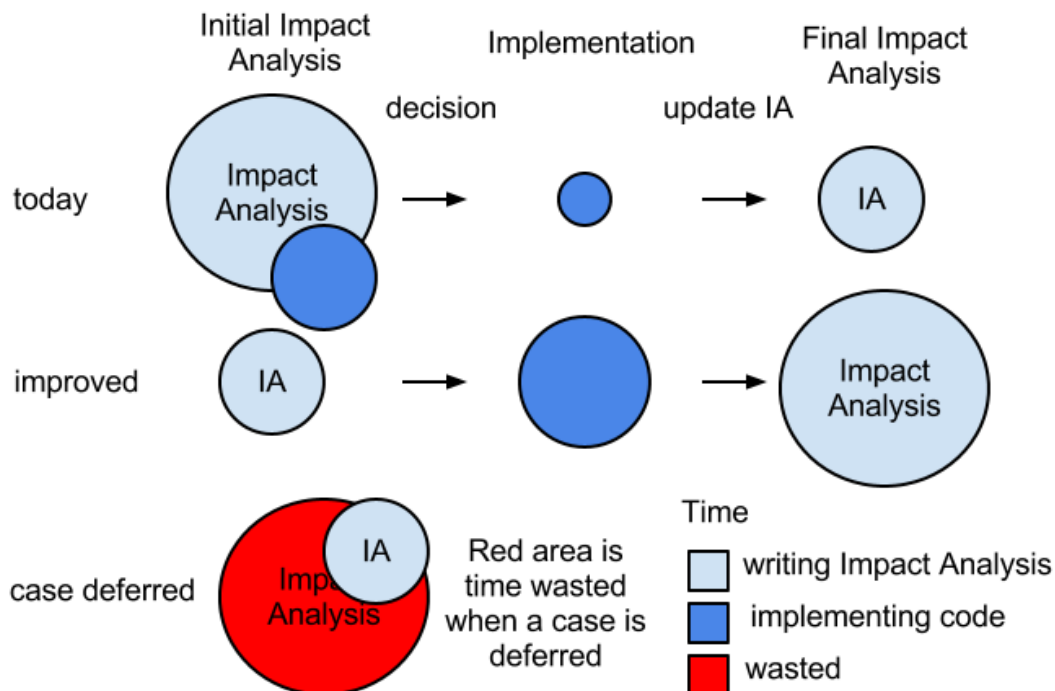


Figure 2 - The situation today versus the suggested one