December 1998

Experience Using State-Based Simulation Models in Industry to Support Process Improvement

David Raffo
Portland State University

Joseph Vandeville
Northrop Grumman Corporation

Experience Using State-Based Simulation Models in Industry to Support Process Improvement

David M. Raffo
Portland State University

Joseph V. Vandeville
Northrop Grumman Corporation

Introduction

The process of developing software is seldom static. The process changes constantly in response to market demands for the faster, more reliable and cheaper production of software [4]. The process is also driven by rapid evolution of software development and quality standards, as well as advances in processes, methods and tools. Given that process improvement is necessary for a company to remain competitive, key questions for software project managers become:

- What is the best approach for making decisions about process changes?
- How can the risk associated with potential process alternatives be assessed?

This paper discusses an approach and rationale regarding how to make process changes which is being implemented at Northrop Grumman's Surveillance and Battle Management Systems in Melbourne, Florida (SBMS Melbourne). Part of this approach uses an analysis method developed by Raffo [7] which enables a firm to quantitatively predict the impact of a potential process change in terms of development cost, product quality, and project schedule. This approach uses stochastic simulation models of each process alternative and supports a quantitative assessment of risk or uncertainty associated with the process change. SBMS Melbourne is currently incorporating this approach as a key component of its process improvement methodology. This work provides a comprehensive treatment of process metrics which have been implemented at SBMS Melbourne and incorporates them into a meaningful framework for making management and control decisions regarding software development projects. This approach supports the successful implementation of earlier CMM Levels which focus on defining processes and implementing metrics. We see this approach as being key to successfully achieving higher level CMM practices related to Quantitative Process Management as well as Process and Technology Change Management.

Background

A software process improvement program was started at SBMS Melbourne in 1990 as the result of recommendations from a quality action team formed to evaluate the existing software development process. The organization chose the Software Capability Maturity Model (CMM) [5] developed at Carnegie Mellon University's Software Engineering Institute (SEI) as the method to achieve the desired software process improvements. SBMS Melbourne now conducts annual software process assessments and is operating at a CMM Level 3.

The Capability Maturity Model specifies key process areas (KPAs) [6] which are generally agreed to be fundamental practices required to produce quality software in a timely, cost effective, and controlled manner. The Capability Maturity Model describes a framework which provides an evolutionary path from ad hoc, chaotic processes to mature, disciplined software processes. A well defined process structure which includes the collection of product and process metric data is necessary as a foundation for enabling a company to effectively adapt to change - accepting new projects, implementing new technologies, assimilating new workers, and so forth in an organized, and controlled fashion. As the company matures, the process definition and collected metric data form the basis for making decisions regarding the management of technical and process change.

Given our current stage of maturity, SBMS Melbourne's current goal is to have a quantitative understanding of the performance of the existing process and of the quality of the software products produced by this process. With this understanding, new technology including process improvements can be incorporated into the organization in an orderly manner.

In early 1995, SBMS Melbourne established a procedure for the introduction of new process technology. Key elements of this procedure are the identification, analysis, implementation and evaluation of proposed process changes. With the increasing maturity level of the organization, this procedure is being improved by the incorporation of process modeling. This paper provides more detail about the SBMS procedure and how it is being integrated with process modeling.

Identifying Process Change Candidates

The Software Engineering Process Group (SEPG) at SBMS Melbourne is the focal point for the identification of software process change candidates. Ideas for process change originate from many sources both within and external to the organization. Some of these sources of candidate changes include the following:

- Process models (e.g., CMM, ISO)

- Corporate initiatives (e.g., Software Council, Inter-divisional cooperation)
- Internal initiatives (e.g., employee suggestion program, quality action teams, interdepartmental cooperation)
- Government and industry initiates (e.g., Software Engineering Institute, Rome Laboratories, Software Productivity Consortium)
- University cooperation (e.g., Software Engineering Research Center, Software Engineering Research Forum, local university programs)
- Internal project experience (e.g., metrics, lessons learned)

Analyzing Process Change Candidates

When candidate process changes have been identified, they undergo an analysis using the current process as the basis of comparison. In the past, at SBMS Melbourne, this analysis has considered the benefits associated with implementing the change, the cost associated with the change (e.g., rework of methods, training, loss of data continuity, impact to productivity) and any risks associated with both the introduction or non-introduction of the change.

The management of process and technology change are CMM Level 5 activities which rely on having a Level 4 quantitative process and product management infrastructure. Without this infrastructure the analysis associated with predicting the impact of a proposed process change tends to be weak by omitting elements which have a significant impact on overall project performance. For example, the analysis of the proposed change may be too narrowly focused, missing the fact that the influence of a change may extend well beyond its area of introduction. An example of this in practice, would be changing the process to conduct Unit Testing before Code Reviews and to analyze this change only with respect to how it impacted Unit Testing and Coding. Depending on the change however, there may be a significant impact to other development phases, such as Integration Testing or Maintenance. We view the capability of assessing the impact of the process change on "downstream" phases to be critical.

Another area of concern is capturing the richness and complexity of the software development process which involves multiple interacting elements involving people, processes and products. Software as a product undergoes many transformations as it progresses from a statement of requirements to validated code. These transformations are typically managed and controlled through a series of processes which represent different development phases. Some of these processes operate concurrently, others sequentially, and many overlap. Some methods of software development are spiral in nature, where the same process may be performed many times. In short, the software product evolves from one abstract form to another, yet in each phase or process activity, performance characteristics such as cost, schedule and quality are measured and tracked with respect to plans.

The organization's goal is to quantitatively analyze a process change to determine not only how it affects the phase in which it is introduced, but also how it impacts the performance of the software development process as a whole. To achieve this level of analysis, the organization has adopted the use of process modeling based on the work done by Raffo [7] and at the SEI by Kellner [3], [2] and Humphrey [1]. This modeling approach involves using a stochastic simulation of a project's software development process using historical metric data for portions of the process which are unchanged and estimates of performance data for the area of process change.

The model is constructed using the existing standard software process. The model is then populated with historic metric data which represents the effect of executing that process. This AS-IS process model can then be modified with the incorporation of the proposed process change. Data related to the process change area (e.g., cost, schedule, and quality data) can be estimated or derived from other sources to produce the TO-BE process model. The TO-BE model which has the process change embedded within it can then be used to simulate how a project will perform. The results of the simulation of the TO-BE process can be compared to the results of the AS-IS process to determine if the proposed process change is an improvement.

At SBMS Melbourne, this approach provides a thorough analysis of process alternatives. Moreover, we believe that a sound and complete analysis of a potential process change "up-front" is essential and can greatly reduce the risk of incorporating a potentially bad process change into the existing development process. The reduction in risk is accomplished by:

- creating a better defined, designed, and planned process change;
- providing a more complete analysis that captures important elements which have previously been left out;
- predicting overall project performance impacts as well as impacts for every phase of the development process;
- capturing uncertainty associated with the development process through the use of stochastic simulation models; and
- being able to make a consistent comparison of several process change alternatives and selecting the best of the group based on predetermined criteria.

Evaluating Process Changes

Once a process change has been selected for adoption, it is implemented and evaluated to determine if it has, in fact, been a process improvement or not. As mentioned in the previous section, a thorough analysis of the proposed process change, up-front, can play a significant role in reducing the risk associated with implementing the change.

The implementation of a process change is in itself typically a non-trivial task. Planning must be done, procedures and work instructions may require modification, standards may need to be updated and personnel are likely to need training or re-training. These expenses are considered to be part of the analysis of the impact of a process change and are typically non-recurring costs.

When implementing the process change, the usual practice for many companies is to use the software development process itself as the "test bed" for evaluating the process change. A project is selected to be the "guinea pig" to see how well the proposed change will work. An advantage of using a simulation model is that the proposed process change can be evaluated prior to its introduction into the target project process. This method offers a significant reduction of risk associated with process change. Moreover, at SBMS Melbourne, we see that process modeling provides earlier evaluation of the success of the new process. This allows us to implement the process change on a greater portion of the project or apply the process change earlier to similar projects. This approach again, significantly reduces the risk associated with the process change and increases the potential benefit.

Conclusions

Many companies constantly evaluate the way they do business and attempt to improve the performance of their processes to meet their business goals and the needs of their customers. In such an environment, process change seems to be inevitable, and the success of a company may depend on how well it can manage this change.

At SBMS Melbourne, key elements of the organization's quality policy statement include customer satisfaction, the production of world class quality systems and continuous process improvement. To achieve these goals, the organization's process improvement program focuses on a quantitative understanding of the performance of the existing process and of the quality of the software products produced by this process. With this understanding, new technology including process improvements can be incorporated into the organization in an orderly manner. A critical part of this process improvement strategy has been to adopt the use of process modeling and use sophisticated analysis and evaluation tools based on the work done by Raffo [7]. These stochastic simulation process models are being jointly developed by SBMS Melbourne and Portland State University.

References

References available upon request from author (davidr@sba.pdx.edu).