Using Discrete Event Simulation (DES) to Model Software Development Process

Abstract

This project seeks funding to support the research on expanding the knowledge of the software development process within the corporate environment in conjunction with Q-Labs. Software development today has a well defined workflow that can be optimized and explored through discrete event simulation in order to maximize resources and total profit of the process as a whole. The software development process and methodologies are used in organizations throughout the United States, many of which are in the defense industry that requires a rating based on process models to obtain U.S. Government contracts. Currently, the Capability Maturity Model (CMM) is one of the leading process improvement models used extensively for avionics software and government projects in North America, Europe, Asia, Australia, South America, and Africa. The objective of this proposed research is to model the software development process with an emphasis on the CMM in an attempt to extrapolate real-world development practices and behaviors present within the current Q-Labs adoption of this development system. The goal would be to balance and streamline this process, thus improving the overall development efficiency. Q-labs, a company that provides change management and model-based process improvement services, will provide the study with real-world data that will validate the findings of the research conducted through Embry-Riddle Aeronautical University (ERAU). This research will establish a partnership between ERAU and Q-Labs for possible long-term collaboration.

Background

With the advancement of today's computer technology, software has become one of the nation's major industries. Despite the ubiquitousness of software systems, today's software still faces challenges in terms of its development and maintenance. It is still difficult to develop a high-quality software product in a productive, efficient, and timely fashion. This is largely due to the fact that today's software structure has become quite complex with many developers and designers involved, throughout the initial design phase. Design activities have also introduced many new dynamics into the development process such as crew resource management (CRM) techniques and other team oriented programs that focus on project collaboration and optimization. New dynamics introduce more complexity and results in the entire project becoming difficult to be controlled.

From a software designer's perspective, concerns about the development process might include (Raffo and Kellner, 1999):

- What development phases are essential?
- Which phase could be minimized in order to reduce costs and cycle time without sacrificing quality?
- What is the value of applying tools in the development process?
- How do we predict the benefits of a new process before commitment of resources?
- How do we compare different process alternatives?

The criticality of software to many organizations have led researchers and industries to spend tremendous amounts of time and effort studying the software development process in an attempt to gain control of it.[Raffo and Kellner, 1999]. Among these efforts, software process modeling is found to be a very effective way to control the process by "increasing process understanding and enabling processes to be formally defined and prescriptively applied" [Raffo and Kellner, 1999]. There are several modeling approaches that have been demonstrated to be effective in addressing some aspects of the software design. For example, the Analytic Summary Model focuses on the high-level cause-effect quantitative relationships between input and output factors of the software development process while the Analytic Structural Model captures the interrelationships, dependencies and structure of the software development process at a more detailed and lower level. [Raffo and Kellner, 1999]. Process modeling, on the other hand, provides "operational guidance" to the critical development steps in order to streamline and delve into the details of the software design process [Raffo and Kellner, 1999]. The amount of research that can be found in this modeling arena is vast, including Software Process Simulation Modeling (SPSMs) [Raffo, et al., year], applying process modeling in software process redesign (SPR) [Scacchi, 2000; 2001], studies on software process maturity [Clark, 1997], and applying discrete event simulations (DES) in software project management [Raffo and Harrison, year; Raffo and Vandeville, year].

Despite the amount of current research conducted in this area, there are few examples of its application to the software industry. Particularly the application of DES due to the dynamic nature of the different software projects compared to the traditional

manufacturing systems. Also, the fast emerging technology of computer hardware and software make it difficult to have a unified DES model for most projects.

In recent years, requirements and design standards from higher maturity organizations, (e.g., those at CMMI Levels 4 and 5 and employing Lean/Six Sigma techniques) have urged software designers to take these high quality standards into process planning and control, to maintain a high level of product quality, and yet at the price of minimal efforts on time and cost. A successful implementation of the CMM within the software development process requires specific development transitions, which are subject to frequent time and resource inconsistencies between comparative projects. Inconsistencies within this development process need to be explored and modeled through discreet event simulation. This would help mitigate risk and provide a focused and strategic answer to these unsettled development problems.

The dynamic and uncertain nature of the software development process has made simulation a desirable tool for such a purpose. The value of using simulation, especially discrete event simulation (DES), has been well recognized [Kelton et al., 2007; Kellner et al., 1999]. Some of the benefits include: formula independent, stochastic modeling; application focused designs for different modeling purposes, including but not limited to, strategic planning and management; process control and operational management,;process improvement and technology adoption, a better understanding of the entire design process; and a better facilitation of the training and learning required to implement the design process.

This project would demonstrate the feasibility of developing and implementing a true DES modeled approach to software development by using a readily available simulation tool, ARENA 10. Investigating and modeling a real world software development problem provided by a national software development company, Q-Labs and the University will gain quality insight and headway into the problems faced by thousands of others utilizing the software development process.

Methodology

The following Figure 1 illustrates a typical software design process currently used by Q-Labs and other professional software development companies.

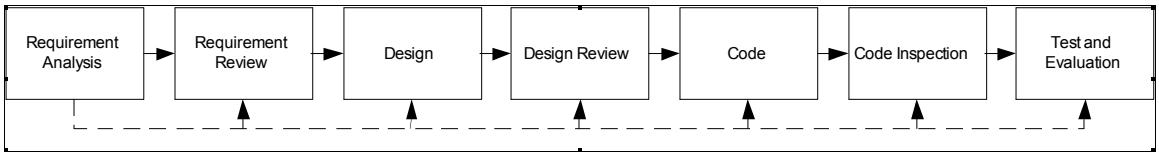

Fig. 1 Typical Software Development Process

A DES model will be built to investigate this process; the main collection and modeling objectives include:

1. Estimation of process performance – the findings of the research will allow accurate prediction and improvement of the performance of their software development process implementation. Performance for the process include:

estimated time for completion of tasks and project, estimated cost estimates for project, estimated resources (mainly the workforce) utilization (Equation 1), and the estimated rework efforts for different stages.

$$B(t) = \begin{cases} 1 & resource\ busy \\ 0 & resource\ idle \end{cases}$$

The utilization of is the area under $B(t)$, divided by the length of the project

$$\frac{\int_0^T B(t)dt}{T} \qquad (1)$$

2. Identification of process bottlenecks – the findings of will allow for identifying bottlenecks in the current software development process and determine the most advantageous changes to the current system to provide maximum throughput of the system.
3. Evaluation of process alternatives and proposed improvements – the proposed improvements resulting from the research will provide real-world recommendations and cost-cutting alternatives to the current Q-Lab implementation and other industries using the software development process.
4. Estimation of the performance of process elements lacking data – any processes in which data cannot be obtained or accurately depicted will need to be estimated in terms of a performance ranked rating as compared to the other processes within the software development system at Q-Labs.

Industry Modeling Approach

Data will be obtained from one completed project/release from Q-Labs to develop the baseline model. Historical data to be collected includes the following items for each of the design elements

- Definition of process elements
- Effort for process elements
- Time for process elements
- Defects found in process elements
- Effort to fix defects in a process element

Upon completion of the data collection, a statistical analysis of the data will be performed by using SPSS and ARENA Input Analyzer to estimate the distribution of the data sample, and by conducting the Goodness of Fit Test (GFT) on these data sets.

Simulation Study Approach

Discreet event simulation software, Arena 10 by Rockwell Automation will be used to map out the specific implementation of the software development process used in the Q-Lab environment. Data will be collected onsite at Q-Labs (Melbourne, Florida) by the student PI. After the modeling is constructed and verified, numerous simulations will be evaluated and recorded to find noticeable patterns and predictable outcomes for changes in the development process. The three most advantageous implementation solutions

developed from the simulation process will be presented to the Q-Labs partner for validation of the findings. The effectiveness of the research conducted will be validated against real-world data produced by Q-Labs. A successful modeling of the current software development system will yield expected results from the Arena software simulations and be compared to the real-world data obtained from Q-Labs. Using subsequent studies or projects from Q-Labs will provide real world cross validation that can be easily applied to other software development projects throughout the United States. Upon acceptance and validation of the simulation results by Q-Labs, the research will then be concluded with a non-disclosed report of the findings from the study and process used to develop these findings. Improvement recommendations will be evaluated based on the response and acceptance from Q-Labs.

Arena 10 - Discreet Event Simulation Software

Rockwell's Arena 10 software is the World's leading simulation software that has been used successfully by many organizations to advance the efficiency and productivity of their business. With an animated Arena simulation model, businesses can design new facilities and explore new processes before the system is in place. The software can compare operational strategies and help to select the best solution for a given set of requirements. ARENA 10 will be used in the study as the tool for simulation model.

Timeline

- Procurement of hardware and software – July 2007
- Data Collection and Modeling System – August to December, 2007
- Subsequent data collection – January-February , 2008
- Model Validation – March - April, 2008
- Result analysis – May, 2008
- Presentation of results and report – June, 2008
- Pacific Northwest Software Quality Conference – October, 2008

Evaluation of Work

Technical reports required by the program, proceedings, one peer–reviewed journal article, submission into International Council on Systems Engineering (INCOSE) magazine, and one student thesis will demonstrate the scientific validity and excellence of this work.

Significance

The purpose of this study is to apply discrete event simulation to model the dynamics of the software development process. At this time, understanding the nature of the development process in contemporary software systems is still extremely limited. This is largely due to the dynamic evolution, complexity, and uncertain nature of the problem along with the difficulty of measuring both software design performance and processes. The DES model proposed in this study will provide a feasible and practical formula independent approach to integrating different performance measures for the software process into a single model, enabling practitioners and researchers to obtain a more comprehensive understanding of the dynamic development process. The results of this research will be of particular value to the software industry. Through the validation of the model, the results derived from the model could easily be extended to a wide range of

applications to ultimately benefit software design, research, and industry. Knowledge regarding the staffing model and resource planning at certain design phrases gained from this study will enable appropriate business decisions to be justified as new cost cutting techniques.

A Graduate student forms a leading and integral part of this proposal; Graduate students will get cross training between Human Factors and Software Engineering. Embry-Riddle Aeronautical University was for many years a traditional four-year college, but graduate programs are now flourishing. The Human Factors and Systems Department has developed a Master's of Science program in both the Human Factors track and the Systems Engineering Track. Research and education related to both areas are in a rapid growth phase at Embry-Riddle Aeronautical University. It is no doubt that a project such as this would provide a magnificent opportunity to contribute to this growth and allow for future students to benefit from real world studies and research opportunities made possible through this unique partnership.

Q-Labs, Inc. is a leading consulting company measurably improving the process of software and systems development. The Human Factors Department at ERAU and Q-Labs has established a partnership in the field of software engineering and systems engineering. This project will enable this partnership to grow and enable both parties to benefit from the collaboration. This will enable ERAU to provide valuable services to the local industry and provide a broader impact to local communities.

Adequacy of Resources

We have requested funding support for the hardware and software required for this study. The Human Factors department will provide lab space for programming and regular meeting among collaborators.

Q-Labs, Established in 1989, maintains operations in France, Germany, Sweden, UK, US, and China. The consultancy team, numbering 120 employees, continues to flourish in all countries meeting the increasing demand of optimizing IT resources and enhancing products, productivity, and quality. The local branch in Melbourne, Florida, has specialized in software process improvement for over 16 years and has successfully implemented projects for hundreds of customers in various industries worldwide. Industries include: the areas of automobile, finance, aerospace/defense/avionics, telecom, and technology. Q-Labs has maintained historical data logs on software projects that will be used in this study. Q-Labs will also provide workspace for the student PI to collect data and interview designers.

Future Proposal Development for External Funding

Upon completion of the project, the PIs along with the Human Factors Department at Embry-Riddle will explore the possibility of collaborating/subcontracting with Q-labs for future software process assessment. Q-labs is already very interested in testing the feasibility of such a model and partnership with ERAU. Current software researchers at NASA, DARPA, the Office of Naval Research, and the Department of Defense also have interests in this research topic. This project will allow the PI to further develop his ideas

and to pursue subsequent funding from these agencies. The proposed study will enable ERAU to provide preliminary results and publications.