



Maintaining High Process Capability in a Student Project Course

Author

Rout, Terence P, Seagrott, John

Published

2007

Conference Title

20TH CONFERENCE ON SOFTWARE ENGINEERING EDUCATION & TRAINING,
PROCEEDINGS

DOI

[10.1109/CSEET.2007.35](https://doi.org/10.1109/CSEET.2007.35)

Downloaded from

<http://hdl.handle.net/10072/17886>

Link to published version

<https://www.computer.org/csdl/proceedings/cseet/2007/12OmNzBwGmA>

Griffith Research Online

<https://research-repository.griffith.edu.au>

Maintaining High Process Capability in a Student Project Course

Terence P Rout and John Seagrott

School of Information and Communications Technology,

Griffith University, Queensland, Australia

T.Rout@griffith.edu.au

Abstract

We have provided a focus on process management and improvement as a basis for conducting student group projects. This paper summarises the lessons learnt from eight years of experience in improvement-focussed projects. The approach has been based upon the establishment of the course as a separate and identifiable organization unit, with its own set of process assets. The responsibility for establishment of these assets resides with the students. Assessment for the course is based on an intensive assessment of the capability of the processes implemented by the project team. The project was assessed on three criteria: the process capability achieved; the achievement against a target capability profile defined by the student team in an Improvement Plan; and the comparison of the capability achieved compared to previous year's achievements. We have found that it has been quite possible for the student project to achieve capabilities equivalent to Level 3 (as defined in ISO 15504) for a significant set of processes under their control. This work demonstrates the feasibility of maintaining organizational identification in a student project course, with the accompanying benefit of achieving improvement in process capability. The educational benefit of the course derives from exposure to process improvement as a critical goal for the students.

1. Introduction

This paper discusses experiences with a capstone software development project undertaken by software engineering undergraduates at Griffith University. There are two primary goals for the project: to provide students with an industry relevant software development task, and to give students experience with using, assessing and improving a set of software development processes. The project also introduces students to technology used in industry.

The need to introduce software engineering students to industry strength software development tasks is well known [1][5][9][13]. This sort of project recognizes the fact that it is “*important that students are aware of the challenges and proven techniques related to industrial development of software*” [19].

The course provides an avenue to increase awareness of, and allow students to apply software engineering best practices, which is very valuable in preparing students for industry [2]. The project also allows students to develop valuable industry skills such as communication, teamwork and project management [6].

We have been lucky to receive continued support throughout the life of the course from the same industry affiliate. This has allowed each yearly intake of students to progressively add to the functionality of the same evolving piece of software in a manner similar to that used by [5]. The importance of this type of development is significant, as most industrial projects do not start from scratch [3][5]. The development projects have been related to the development of a software product used in the testing of software.

The importance of process in the development of software is large [4][5][16][17]. Our course differs from most other software development projects offered in educational

institutions due to our focus on process assessment and improvement in addition to the use of processes. The course described by Suri and Sebern [16] is similar to our course, but little detail is given on the results of the improvements made each year. There is also no discussion or evidence of a systematic process assessment model being used to help identify improvement opportunities.

The course applies a model-based approach to improvement, similar to those employed in industry [4, 16]. The course employs process assessment conformant with ISO/IEC 15504 [7] as the basis for evaluation, and uses ISO/IEC 12207 [8] as the basic reference model for process establishment, deployment and improvement.

In this course, each new intake of students use, assess and improve the assets provided to them the previous project team. A set of standard assets is maintained in a repository that can be accessed by students undertaking the course each year. The students take the set of standard assets, consisting of standard processes, tailoring guidelines, templates and forms, and tailor them to be appropriate for use in the current project. The CMMI [4] describe a defined process as being "*a managed process that is tailored from the organization set of standard processes according to the organization's tailoring guidelines*". These defined processes are then taken, with the other assets, and deployed for use in the project.

The students use the results of three main types of assessments as a basis for improvement identification: results from the previous year's final assessment, results from self-assessments and results from a mid-year RAPID assessment [12]. The improvements are made to the set of standard organizational assets discussed above, ready for the next time the processes are used.

Here we present the results for each year over which the project has been run, which allows us to see the use and improvement of the project assets by a variety of student groups on different development projects. The way in which the different groups deploy the assets is very interesting and varies greatly, and will also be discussed throughout the paper.

2. Project Courses at Griffith University

Information technology degree programs at Griffith University have traditionally incorporated a significant project course [10]. When a full four year program for software engineering was introduced, the project element was enhanced by incorporating a second full-year project in the final year of the degree, in addition to the existing course. This provided an opportunity to broaden the educational goals for the program and for the course.

The generic project course was designed from its inception to incorporate quality management as a basic learning objective [11]; in its original design, students were required to manage their projects so as to meet the requirements of **Australian Standard 3563 - 1988, Software Quality Management System** [15]. In achieving the goal, it was expected that the groups would produce, firstly, a statement of Quality Policy and a detailed Quality Plan; and secondly, well-defined and accepted standards for the various stages of the project life-cycle.

The course has evolved over time to a focus on project management, supported by a significant suite of standards, criteria and templates for the expected work products. Students are supported by the teaching team in the use of the provided assets; teams are small (4-6 members) and each project has a unique aim in developing a solution to a problem for a client who is normally external to the University.

With the introduction of a second project-based course into the four year degree program, a design was sought that would introduce new learning experiences, and would reinforce the performance of practices seen as representing best industrial performance. The "large group" model [14] was used, with the whole class forming a single project team. Assessment has

been heavily based on the use of process assessment approaches, with ratings of process capability in different processes being used as one basis for assessment of team members.

The emphasis on process management and process improvement as the principal learning focus for the course has increased with time. The course now requires the students, working as a group, to establish from existing process assets a defined process, and to apply this to the planning, executing, monitoring and controlling of a software engineering project of significant scope and complexity. At present – eight years after its initial establishment – the student group is regarded as representing a continuing organization, having a 100% staff turnover annually. The team has the following goals for the course:

- Deploy an existing set of process assets as a defined process to the project. The set of process assets has been developed and evolved by students enrolled in the course over the past seven years; Table 1 shows the contents of each Standard Process Description. Standard process definitions have been developed for most of the processes defined in ISO/IEC 12207 [8].
- Demonstrate the performance of the defined process using appropriate measures.
- Identify and prioritise improvements to the process assets.
- Implement the improvements, including where possible, deploying the new process assets.

Table 1 – Contents of Standard Process Description

Introduction

- Identification
- Document Purpose
- Process Purpose
- Process Outcomes
- Work Products
- Definitions and Acronyms
- References

Process Infrastructure

- Process Goals
- Process Interfaces
- Roles in the Process
- Tools in the Process
- Work Environment

Activity Descriptions

Graphical Overview of the Process

Tailoring Options

Traceability

Process Outcomes to Standard Activity

The determination of student grades for the project is based 60% on the achievement of the team and 40% on individual performance. In determining the team's achievement, a formal assessment of process capability, in accordance with the international standard ISO/IEC 15504 -- *Process Assessment* [7] is employed. Process profiles for a scope of up to 20 processes, drawn from ISO/IEC 12207 [8], are determined. The team's achievement is evaluated based upon the following criteria:

1. Assessed Capability Profile. This is based on the Capability Level achieved, where achievement of Level 3 in each process assessed would represent 100%.
2. Improvement against plans. This is based on a comparison between the achieved capability and the target capabilities defined in the Process Improvement Plan. Matching the target capability in each process would represent 100%.

3. Improvement over historical profile. This is based on a comparison between the achieved capability and the capabilities recorded in the final assessment for the previous year. The comparison is based on the individual Process Attribute Ratings, not just Capability Level Ratings, and thus includes provision for recognition of minor improvements.

In addition, a component addressing the team's ability to manage issues which have arisen during the year is incorporated. This is based on an evaluation drawn primarily from a Post-Project Review presented by the team, together with information from regular status reports.

The critical element in evaluating the team's performance is a comprehensive assessment of process capability for the project, conducted as the final event in the year. The assessment is conducted according to the requirements of ISO/IEC 15504 [7], and takes three days; a team of three or four members, all trained in performance of assessment and under the supervision of a recognised Competent Assessor, performs the assessment. All work products generated by the team during the course of the year – including also the archival process assets developed in previous year – are examined, and all team members are interviewed, both individually and in groups, by the assessment team. The Process Scope of the assessment is determined by the Course Convenor, and generally includes up to 20 processes drawn from ISO/IEC 12207 [8]; ISO/IEC 15504-5 [7] is used as the Process Assessment Model for the assessment.

These processes were assessed to Capability Level of 3 (out of a possible 5). Ratings are assigned for achievement of the following process attributes (PA) as defined in ISO/IEC 15504 [7][18]:

- PA 1.1 – Process Performance Attribute
- PA 2.1 – Performance Management Attribute
- PA 2.2 – Work Product Management Attribute
- PA 3.1 – Process Definition Attribute
- PA 3.2 – Process Deployment Attribute

Each Process Attribute was rated on a four-point scale of achievement, as prescribed in ISO/IEC 15504 [7][18]:

- F – Fully Achieved
- L – Largely Achieved
- P – Partially Achieved
- N – Not Achieved

From the resulting Process Profiles, the Capability Level achieved by each process was determined, again following the scheme in ISO/IEC 15504 [7].

The principal component in the evaluation of individual performance is a personal diary maintained by each student recording details of effort productivity, quality and lessons learned from the project. In addition to this, additional information from project records, a peer review of individual performance, and an overall evaluation by the teaching team members is incorporated.

The personal diaries are important element in the ongoing improvement of the course. This is the principal mechanism used to gain feedback from the students on the operations of the course on whether the learning objectives have been met.

3. Results - Process Capability Achievement and Improvement

3.1. Process Profiles

Figure 1 presents the results of the ISO/IEC 15504-5 [7] based process assessment performed at the end of the course each year. The results presented span from 2001 to 2006

and cover 17 processes. The diagrams included in Figure 1 show how the process ratings have changed over the years.

No clear pattern of overall improvement over time is evident; however, the capability levels achieved by the teams in different years are significant, with consistent achievement of Level 3 capability in multiple processes. The mechanism chosen for the course – where groups in each year identify and pursue their own improvement targets – results in varying priorities; the availability of resources (see below) also impacts on both the extent of improvements undertaken, and the extent to which existing capabilities can be maintained.

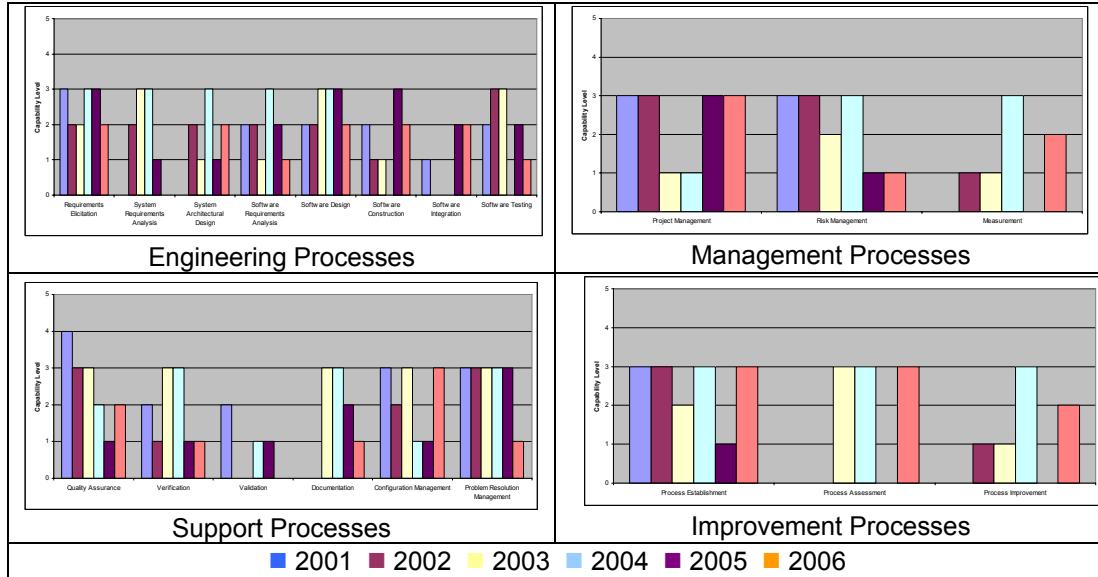


Figure 1 – Process Capability Assessment Results, 2001 - 2006

3.2. Process Improvement Achievements

Table 2 – Achievements in Process Improvement

Process 2004	Attribute	PA 1.1	PA 2.1	PA 2.2	PA 3.1	PA 3.2	CL
Measurement Process	Previous Target Achievement	L F F	L F F	L F F	L L F	L L F	1 3 3
	Previous Target Achievement	L F F	L F F	F L L	F L L	P L L	1 3 3
	Previous Target Achievement	P F F	L F F	P F F	F L P	P F F	0 2 2
2005	Previous Target Achievement	P F F	L F F	P F F	F L P	P L F	0 2 2
	Previous Target Achievement	P F F	L F F	P F F	F L P	P L F	0 2 2
	Previous Target Achievement	P F F	L F F	P F F	F L P	P L F	0 2 2
2006	Previous Target Achievement	P F F	L F F	P F F	F L P	P L F	0 2 2
	Previous Target Achievement	P F F	L F F	P F F	F L P	P L F	0 2 2
	Previous Target Achievement	P F F	L F F	P F F	F L P	P L F	0 2 2

The results presented in Table 2 are a selection of improvements made over the life of the course. In each year, based upon their own defined priorities, the team targeted processes for improvement, based upon the achievement of a desired Capability Profile. In Table 2, the results (as detailed Process Profiles) are shown, listing the previous achieved capability; the target profile as defined in the Improvement Plan; and the actual capability as assessed.

It can be seen that, where the team identified and targeted specific improvements, they had considerable success in meeting these targets. Comparison of these results with those in Figure 1 leads to the conclusion that this was often at the expense of maintaining capability in "non-targeted" processes; however, this was not the only reason for lower assessed capabilities. In many cases this arose because the process was not specifically exercised, because of the nature of the year's project, or because of resource constraints.

4. Issues and Observations

4.1. Class Size

The principal issue encountered over the time that the course has been running has been a decline in enrolments. The degree program was designed to stabilize with an output of approximately 20 graduates per year, and after 6 years, was tracking towards this target; however, from 2005, numbers in the final year of the four-year program, and therefore numbers enrolled in the Project, declined sharply. The enrolment data is shown in Figure 2.

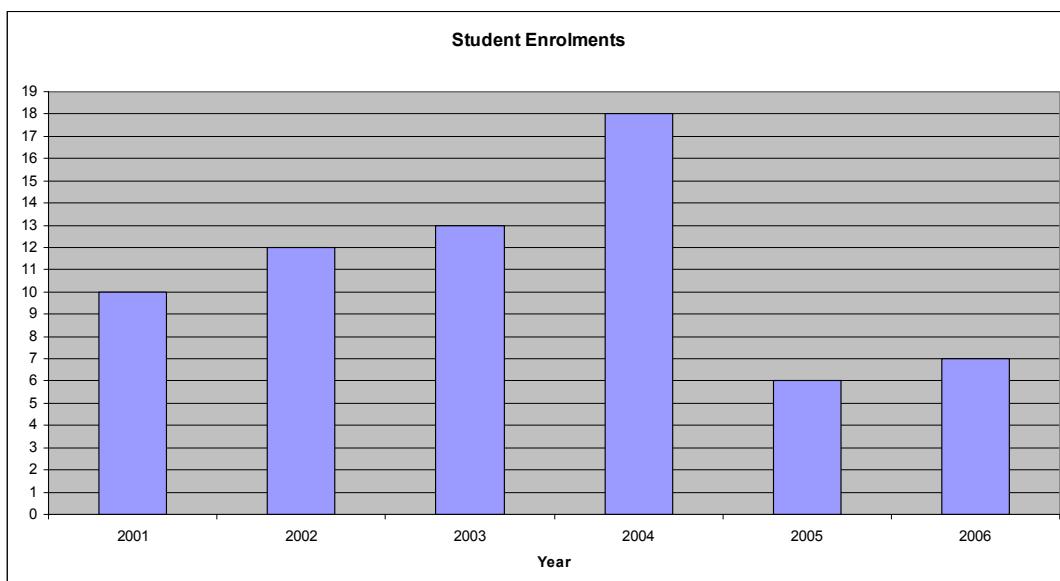


Figure 2 – Enrolment numbers in Project Course

The decline in numbers can be attributed to two factors: the general decline in enrolments in IT-related courses following the collapse of the "Internet Boom", and the loss of funding for specific scholarships in software engineering, that had been offered by Software Engineering Australia (Qld).

While it would be expected that such a dramatic decline in numbers would have had a major impact on the success of the course, this did not in fact eventuate; in fact, the limited resources provided the pressure needed to encourage more significant usage of the course's Process Assets. Over the last two years, students in the course have (of necessity) limited the development of new standard processes in favor of maintenance of existing standards; the

primary focus in terms of process improvement has been to determine more appropriate development approaches and tailoring guidelines suited to small project teams.

4.2. Turnover of Personnel

The second most significant factor in the operation of the course was a more generic one, and derived from the fact that there was a complete turnover of personnel within the "project organization" at the completion of the term of the course. Since the design of the course was based around the concept of a continuing "virtual enterprise" with ongoing continuity of mission and goals, this posed a considerable problem. The turnover is inherent in the progress of the students; the key issue is to manage its impact on the use of the standard process assets. The issue was addressed in several ways.

In the first instance, a continuing relationship was built up with an external organization that acted as client for the projects. The company was a consultancy firm, specializing in the provision of software testing services. Over the period of the course, the client organization provided realistic projects in the form of internal support systems designed to interface with their internally-developed testing environments. The second key strategy was to involve previous students in the delivery of the course, wherever possible. Several graduates from the course were employed by the client organization, providing a combination effect.

4.3. Focus on the Project!

In the actual operations of the course, the most significant issue identified, has been the need to maintain a focus on the technical work of the project. There is a tendency for the students to become locked in to the definition and analysis of processes rather than focusing on the use of the process to support technical work of the project. This has reflected itself in an unwillingness to deploy processes, where the standard process assets are seen as being of inadequate quality.

A key lesson for the students from the project has been to accept the existing process assets as having value, and to deploy these and seek actively improvements in both the standard processes, and in the mechanisms for tailoring of these standards to apply to the work of the project. In the records of capability, presented above, this can be seen in periodic failure to achieve even performed capability in some of the fundamental engineering processes. Here the students were so concerned with weakness of the available assets that they proceeded basically without any process support, and as a result failed to address one or more of the basic expected outcomes of the process.

5. Summary and Conclusions

The results presented here demonstrate the ability to achieve relatively high levels of process capability in a student project course, based upon the continuing evolution of process assets defined and maintained by the students over successive years. The maintenance of a "virtual enterprise" and the recognition of the course as such an entity by the students is critical to these achievements.

The educational benefit of the course is seen as deriving from exposure to process improvement as a critical goal for the students, rather than as "another academic topic". Through the development of real and verifiable goals for ongoing improvement (over previous historical achievements) the students become acquainted with many of the challenges and decisions that face improvement initiatives in the wider industrial environments.

The principal threat to the future of the course is similar to the threats that impact on industrial organizations: the intervention of market forces (as reflected in the decline in

enrolments). Without an increase in the number of students in the degree programme as a whole, it is difficult to see how the course can continue; this is a generic issue impacting on the industry generally, and one without any easy solution.

6. References

- [1] Ali, M.R., "Imparting Effective Software Engineering Education", ACM SIGSOFT Software Engineering Notes, Vol. 31, No 3, July 2006, pp. 1 – 3.
- [2] Bernstein, L., Klappholz, D. and Kelley, C., "Eliminating Aversion to Software Process in Computer Science Students and Measuring the Results", Proceedings of the 15th Conference on Software Engineering Education and Training (CSEET '02), February 2002, pp. 90 – 99.
- [3] Bothe, K., "Reverse Engineering: the Challenge of Large-Scale Real-World Educational Projects", *Proceedings of the 14th Conference on Software Engineering Education and Training (CSEET '01)*, February 2001, pp. 115 – 126.
- [4] Chrissis, M.B., Konrad, M. and Shrum, S., *CMMI – Guidelines for Process Integration and Product Improvement*, Addison-Wesley, Upper Saddle River, NJ, 2006.
- [5] Gehrke, M., Giese, H., Nickel, U.A., Niere, J., Tichy, M., Wadsack, J.P. and Zundorf, A, "Reporting about Industrial Strength Software Engineering Courses for Undergraduates", *Proceedings of the International Conference on Software Engineering (ICSE '02)*, pp. 395 – 495.
- [6] Goold, A. and Horan, P., "Foundation Software Engineering Practices for Capstone Projects and Beyond", *Proceedings of the 15th Conference on Software Engineering Education and Training (CSEET '02)*, February 2002, pp. 140 – 146.
- [7] International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC), *Information Technology – Process Assessment: Parts 1 - 5*, ISO, Switzerland, 2006.
- [8] International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC), ISO/IEC 12207 – Software Lifecycle Processes: Amd 1/2, ISO, Switzerland, 2004
- [9] Mayr, H., "Teaching Software Engineering by Means of a 'Virtual Enterprise'", *Proceedings of the 10th Conference on Software Engineering Education and Training (CSEET '97)*, April 1997, pp. 176 – 184.
- [10] Rout, T.P., "Quality, Culture and Education in Software Engineering", *Australian Computer Journal*, Vol 24, No 3, Aug 1992, pp. 86-91.
- [11] Rout, T.P. "Quality Assurance in student software projects – an exercise in action learning", *Proceedings of the Biennial Conference of the Australian Organisation for Quality (QUALCON 90)*, Adelaide, August 1990
- [12] T.P. Rout, A. Tuffley, B. Cahill and B. Hodgen, "The Rapid Assessment of Software Process Capability" in *Software Process Improvement*, ed. R.B. Hunter and R. Thayer, IEEE Computer Society Press, Los Alamitos, California, 2001
- [13] Runeson, P., "A New Software Engineering Program – Structure and Initial Experiences", *Proceedings of the 13th Conference on Software Engineering Education & Training (CSEET 2000)*, March 2000, pp. 223 – 232.
- [14] Shaw, M. and J. Tomayko, *Models for Undergraduate Project Courses in Software Engineering*, CMU/SEI-91-TR-10, Software Engineering Institute, Carnegie Mellon University, 1991
- [15] Standards Australia, Australian Standard 3563 – 1988, Software Quality Management System, 1988.
- [16] Suri, D. and Sebern, M.J., "Incorporating Software Process in an Undergraduate Software Engineering Curriculum: Challenges and Rewards", *Proceedings of the 17th Conference on Software Engineering Education and Training (CSEET '04)*, March 2004, pp. 18 – 23.
- [17] Upchurch, R.L. and Sims-Knight, J.E., "Designing Process-based Software Curriculum", *Proceedings of the 10th Conference on Software Engineering Education and Training (CSEET '97)*, April 1997, pp. 28 – 38.
- [18] van Loon, H., *Process Assessment and ISO/IEC 15504*, Springer, New York, 2004.
- [19] Wohlin, C. and Regnell, B. "Achieving Industrial Relevance in Software Engineering Education", *Proceedings of the 12th Conference on Software Engineering Education and Training (CSEET '99)*, March 1999, pp. 16 – 25.

Acknowledgements

The achievements of the project course have depended upon contributions from a large number of our colleagues at Griffith University; in particular, the efforts of Angela Tuffley, Bruce Hodgen and Don Abel must be recognised. The contributions of those professionals who have participated in the assessment process capability has been critical; particular mention is due to Michelle Morley, Danielle Tavares-Rixon and Fred Liang. The most significant contribution, however, has been that of the students whose achievements are largely responsible for the results we have presented here.