1 An Analysis of Student Performance during the Introduction of the PSP: An Empirical Cross Course Comparison

Fernanda Grazioli, Universidad de la República
William Nichols, Software Engineering Institute
Diego Vallespir, Universidad de la República

1.1 INTRODUCTION

Almost every new product or system that we use in our daily life has a software component for its operation. Meanwhile, both the size and complexity of the software increase day by day. In this context, software engineering needs improved software quality, better cost and schedule management as well as reduced software development cycle time [Sommerville 10].

The Team Software Process (TSP) is a software development process for teams that satisfies these needs and which uses the Personal Software Process (PSP) for each team member [Humphrey 05a] [Humphrey 06]. The PSP is a defined and measured software process designed to be used by an individual software engineer to addresses the software businesses needs by improving the technical practices and individual abilities of software engineers, and by providing a quantitative basis for managing the development process [Humphrey 05b].

Given that the TSP is a process successfully used and it is qualified as the best software development process for medium and large scale projects [Jones 10], it is important to know whether the processes and the techniques of the PSP lead to develop high quality products. Therefore, the general goal of this study is to know if the different techniques and phases of the PSP (therefore, the PSP itself) produce positive changes in the aforementioned aspects of the software development.

The PSP is taught through a course. Several versions of the course use the same exercises, but introduce process phases and techniques in modified sequences. An earlier version of the course has several published studies demonstrating improvement in developer performance[1] with process insertion [Hayes 97][Rombach 08][Paulk 06] [Kemerer 09][Paulk 10], but the retrospective analysis left some threats to the validity of these claims. One threat to the validity of the claims of these studies is the confounding of the effect of introducing process phases and techniques insertions with the gaining of domain experience as related programs are developed.

Given this known problem (validity threat to prior experiments in PSP), the main goal of this study is to use the PSP data from the latest two course formats to determine whether the different techniques introduced improve several aspects of developers' performance, or if such improvement is only a consequence of gaining experience in the problem domain. A secondary goal is to document observations and results of the two recent course versions, which do not have yet published works.

Based on [Hayes 06] and [Rombach 08], and continuing our previous study of defect density in unit testing [Grazioli 12], we decided to evaluate the effects of the last two PSP course versions through three hypotheses, focusing on determining the main reason for the improvements and not just evaluating the effect size of the improvements. Therefore, we defined the particular goals of this study as:

[1] The term "performance" covers several aspects, such as improve the quality of the produced product, produce better estimations, and increase the code production rate, among others. It should not be confused with productivity.

- Analyze and compare the data collected at the PSP levels in two different courses for the purpose of evaluating performance improvements of engineers with respect to ***yield / production rate / size estimation accuracy*** from the viewpoint of a researcher in the context of the PSP training of engineers in "PSP for Engineers I/II revised" course and the training of engineers in "PSP Fundamentals and Advance" course.
- In case of improvements, determine if these are due to the specific techniques introduced or if such improvements are only a consequence of the experience gained in the problem domain.

1.2 DATA SET

We used data from the eight program course version, PSP for Engineers I and II (PSPI/II), taught between June 2006 and June 2010 and from the seven program course version, of PSP Fundamentals and Advanced (PSP Fund/Adv), taught between December 2007 and September 2010. These courses were taught by the Software Engineering Institute (SEI) at Carnegie Mellon University or by SEI partners, including a number of different instructors in multiple countries.

We analyzed 347 subjects in total, 169 from the PSP Fund/Adv course and 178 from the PSPI/II course. From this we made several cuts and run data cleaning algorithms to include only the students who had completed all programming exercises, in order to clean and remove errors and questionable data. We determine other cuts on the data set, by performing an analysis and assessment of the data quality based on the data quality theory.

1.3 STATISTICAL MODEL

In our context, there are several subjects performing the same task (programming) but following different processes (PSP levels). This is a repeated measures experiment. What we want to do is to notice changes in the individuals when they change the applied process.

To know whether engineers improve their performance during the course, we studied the changes in engineers' data over seven different programming assignments. Rather than analyzing changes in group averages, this study focuses on the average changes of individual engineers. Some engineers performed better than others from the first assignment, and some improved faster than others during the course. In order to discover the pattern of improvement in the presence of these natural differences between engineers, the statistical method known as the repeated measures analysis of variance (ANOVA for repeated measures) is used [Tabachnick 13].

Below we define terms and independent variables that must be clear to understand the analyses:

- Subject – A student who performs a complete PSP course.
- Course Type – Refers to a PSP course version. It can be PSP Fund/Adv or PSPI/II.

- Program Assignment or Program Number – Refers to an exercise that a student has performed during the PSP course. Values go from 1 to 7. Program assignment 8 of the PSP I/II course version is not going to be analyzed as there is no way to compare it with another assignment in the PSP Fund/Adv course version.
- PSP Level – Refers to one of the six process levels used to introduce the PSP in these course versions. It can be PSP0, PSP0.1, PSP1, PSP1.1, PSP2, PSP2.1. Each program assignment has a corresponding PSP level according to the PSP course version. As we want to analyze the introduction of phases and techniques during the courses, we group PSP0 and PSP0.1 and we group PSP1.0 and PSP1.1, and analyze PSP2.0 and PSP2.1 separately.
- Yield = 100 * Defects removed before compile phase / Defects injected before compile phase
- Production Rate = (Actual A&M LOC / Actual Minutes) * 60
- Size Estimation Accuracy = (Estimated LOC – Actual LOC) / Estimated LOC

To analyze whether performance improvements are due to the programming repetition or due to the phases and techniques introduction, we defined and used an indirect statistical method of analysis. This method consists of three steps in which the relationships between program number, PSP level, course version and engineers' performance are examined applying ANOVA. Figure 1 presents flow chart of this method.

The first step tries to find out whether are there differences between the two courses by comparing the variable under study for each program assignment (comparing the same program in different courses). For a program, when there is no statistical difference it is discarded. If there are significant differences when there is no PSP level difference within the courses for that program, then the level cannot be the root cause of the differences in the variable under study. But, when the differences are found when there is a level difference for that assignment, then we should move forward to the second step in order to find if the PSP level could be the root cause of the changes.

We know that in each course, each program assignment is completed following a specific PSP level. The second step looks at each course separately, and tries to find if the differences between the course programs assignments are taking place when the PSP level has changed or if the differences are taking place even when the PSP level has not changed between two assignments. If there are significant changes between programs assignments with the same PSP level, this can lead us to think that the effects on the dependent variable are due to the repetition of exercises and not due to a specific technique introduction. Otherwise, if the significant changes are only between programs assignments with different PSP level, then we must study (in the third step) the behavior of the engineers' performance through the PSP levels, when grouping the program assignments by PSP level.

The third and last step looks at each course separately again, and tries to find if the differences between the PSP levels are taking place when a specific technique that is expected to improve an aspect of the engineers' performance is in fact introduced. If there are significant changes between

PSP levels where the technique is introduced, this will be showing that the technique introduced is the factor affecting the introduced engineers' performance and not the program repetition.

Figure 1 shows a flowchart that represents in a clear graphic way the flow of the third step analysis procedure that we propose and followed for each dependent variable.

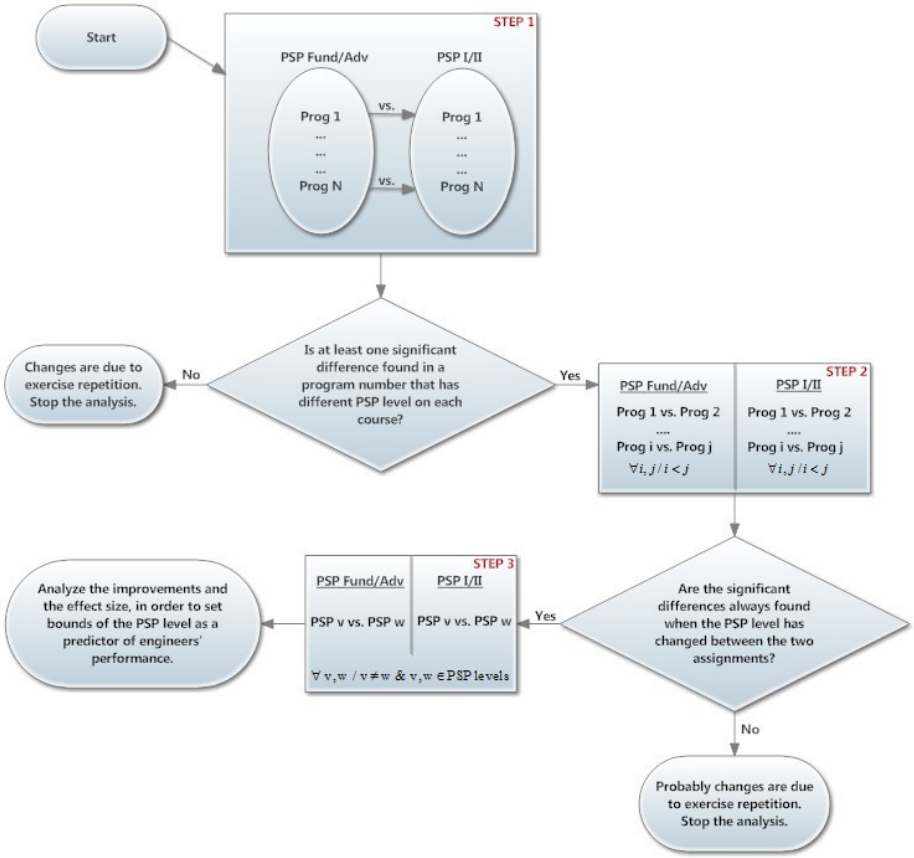

Figure 1: Three Step Analysis Approach Flowchart

1.4 RESULTS

This section presents a summary of the results obtained for the three hypotheses. We should remember that following the same approach, in a previous study that share the same main goal, we analyzed performance improvements of engineers with respect to defect density in unit testing, were we found significant improvement with a mean reduction of a factor of 2.3. That result suggests that improvements in defect density in unit testing are most plausible regarding mastering PSP techniques rather than programming repetition [Grazioli 12].

1.4.1 Yield

After following the analysis procedure for Yield, for each course we only found significant difference between assignments with different PSP level, and we did not find significant difference in process yield between PSP0 and PSP1. According to the design and code review introduction in PSP level 2 these improvements were expected. The left plot of Figure 2 shows the estimated marginal means of Yield vs. program number, for both courses. The graphic shows how the two courses have low yield during assignments with PSP level 0 or 1, then an important increment on yield on the first PSP2 introduction.

Looking at the two-way ANOVA results of step three, in both courses we find out that there is significant difference between PSP0 and PSP2, PSP2.1. We also found that there is significant difference between PSP1 and PSP2, PSP2.1. The right plot of Figure 2 shows the 95% confidence intervals of Yield for each PSP level, for both courses.

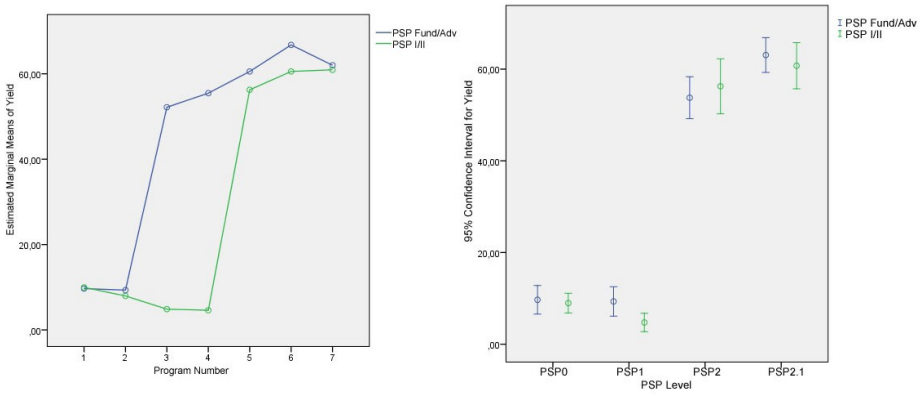

Figure 2: Estimated Marginal Means and 95% Confidence Interval of Yield

Our results show significant improvement in the process yield with a mean increase of a factor of 1.9. Our results also support that design and code reviews techniques are the main reason of the improvements rather than the learning effect.

1.4.2 Production Rate

After following the analysis procedure for Production Rate, for each course we only found significant difference between assignments with different PSP level. There is a deterioration of production rate as engineer's move forward in the PSP level. The left plot of Figure 3 shows the estimated marginal means of Production rate vs. program number, for both courses. The graphic shows how engineer's production rate evolves during the complete courses.

Looking at the two-way ANOVA results of step three without course discrimination, we find out that there is significant difference between each PSP level compared in pairs. The right plot of Figure 3 shows the 95% confidence intervals of Production rate for each PSP level, considering both courses together.

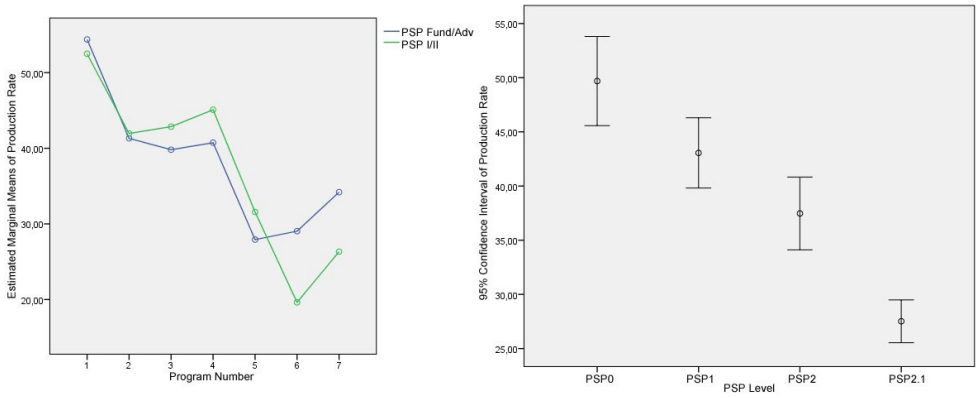

Figure 3: Estimated Marginal Means and 95% Confidence Interval of Production Rate

So, regarding production rate we found a mean reduction of a factor of 0.7. In our study both courses appear to be effective in demonstrating that the increments in the amount of design documentation and data tracking proposed by the PSP deteriorates the production rate during the PSP course. Our result differs from previous studies of the 10 program course version, which some find improvements and others find no real gain or loss [Hayes 97][Rombach 08][Paulk 10].

1.4.3 Size Estimation Accuracy

After following the analysis procedure for Size Estimation Accuracy, for each course we only found significant difference between assignments with different PSP level. According to the PROBE technique introduced, which is based on engineer historical data, these improvements were expected. The left plot of Figure 4 shows the estimated marginal means of ABS(SEA) vs. program number, for both courses. The graphic shows how the two courses perform differently, even we cannot see the specific effect of the introduction of the size estimation technique in PSP Fund/Adv course. Remember that in PSP Fund/Adv we cannot compare PSP1 to something previous, as there is not a previous assignment with a size estimation calculus done by the student. We can see the evolution of the rest of the course, but not specifically the PSP1 introduction. In this graphic of the estimated marginal means, the size estimation accuracy appears to be more consistent by the end of the courses.

Looking at the two-way ANOVA results of step three, in the PSP Fund/Adv course we found that there is significant difference between PSP1 and PSP2.1. But as we do not have assignments with PSP0, we cannot study the effects of introduction of PSP1. Regarding to the two-way ANOVA results for the PSP I/II course, we found that there is significant difference between PSP1 and PSP2, PSP2.1. The middle plot of Figure 4 shows the 95% confidence intervals of absolute value of size estimation accuracy for each PSP level, for both courses.

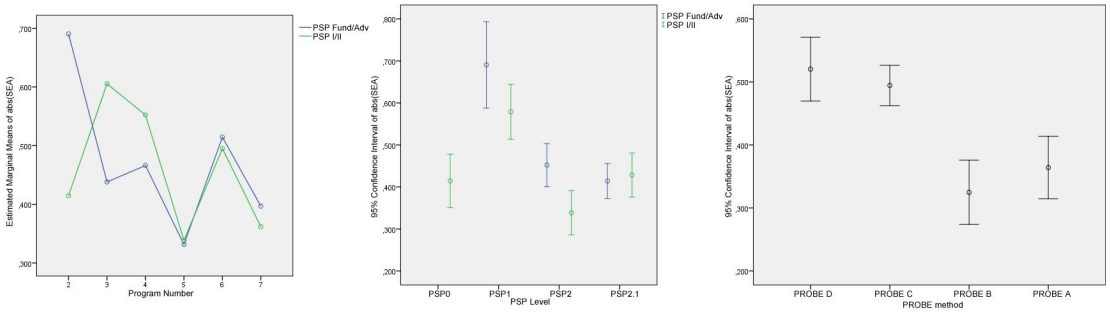

Figure 4: Estimated Marginal Means and 95% Confidence Interval of abs(Size Estimation Accuracy)

With these results, we do not really see directly that the introduction of the estimation technique improves the size estimation accuracy, because in PSP2 and PSP2.1 are introduced the design and code reviews and design templates, not the estimation' techniques.

In order to get a clearer idea of the relationship between the estimation techniques introduction and the size estimation accuracy, we propose to analyze the data in a different way. Not looking at the PSP level, but looking at the specific PROBE method that is applied in each assignment. To do this, we execute again the third step of the indirect analysis method, but this time reorganizing the student data by PROBE method (A, B, C or D). We found that there is a significant difference between PROBE A and PROBE C, D as well as a significant difference between PROBE B and PROBE C, D. The right plot of Figure 4 shows the 95% confidence intervals of the absolute value of size estimation accuracy for each PROBE method, for both courses together.

With the available data it is very difficult to separate the possible causes of size estimation improvement: the introduction of the formal estimation technique and the experience in the problem domain. With the presented results it is clear that data shows and support the hypothesis that the engineer's size estimates improves. But we cannot determine if the introduction of the size estimation technique is the main reason of that improvement because:

- PROBE A and B cannot be applied until there are a minimum of three historic points
- It takes accumulated data for the size estimation technique to become effective
- The estimation process take multiple repetitions to stabilize
- The estimation technique is not just one technique. In fact, it is a package of three different methods, and student varies it application during the course
- The PSP level introduction on the last two courses is not the optimal to study this hypothesis

So, regarding to size estimation accuracy results, we found significant improvement with a mean reduction of a factor of 2.6. For this particular dimension we were not able to discard the domain learning effect as the root causes of the improvements, as the estimation technique introduced in the PSP courses is based on historical data and needs repetition.

1.5 THREATS TO VALIDITY AND LIMITATIONS

To apply the repeated measures ANOVA some assumptions must be met: subjects must be randomly selected, observations on these subjects are independent, the dependent variables must be normally distributed and have equality of variances.

The researchers did not select the subjects; they were the ones that selected the course, and there is no precondition to do one course or another. So the random selection seems to be satisfied. But, on the other hand, some other biasing factor remains, because the students that took the PSP Advanced are more likely to go onto instruction or teaching. So, this group might respond better to the PSP instruction, and this could be seen as a threat to validity. As other potential factors, a completely independent observation of the subject is almost impossible to achieve as classes are working together with the same instructor and thus they do not only depend on the sole quality of the instructions. Given the quite large set of data, the large number of different instructors, and numerous different classes this assumption should however not be completely violated.

The analysis of the collected data showed that the requirement for normal distribution of the dependent variables is not fully met. However, the data are mounded without severe outliers. Nevertheless, different transformation techniques were applied to better meet this assumption for each hypothesis to reach a more normal distribution variable. Fortunately, an ANOVA is not very sensitive to moderate deviations from normality; simulation studies, using a variety of non-normal distributions, have shown that the false positive rate is not affected very much by this violation of the assumption [Glass 72][Harwell 92][Lix 96].

The PSP training aims at providing engineers with techniques to improve their daily work with 7 or 8 assignments, depending on the course version. The data is collected within a class set-up where the attendees can concentrate on the assignment and are not distracted by colleagues, working on multiple projects, etc. The investigation thus can only show the improvements achieved during the duration of the class. A general translation of the achieved improvement effects to generally improved workplace performance must however be seen very carefully. The results show trends and it can be interpreted that the trend might continue and finally lead to the assumed results. It is also not directly possible to conclude that the results are immediately valid for large scale projects, when the engineers are working in multiple project teams, and the project is executed over a long time span.

1.6 CONCLUSIONS

The analyses executed in this work substantiate that trends in personal performance observed during PSP application are significant, and that the observed improvements or deterioration represent real change in individual performance, not in the average performance of the group.

Because of the followed approach, we are able to suggest that the PSP is the root cause of the improvements rather than the domain learning effect in process yield and in defect density in unit testing. Since PSP level changes so rapidly in the PSP Fundamentals and Advance course and in the PSP I/II revised course, the program number and the PSP process level are tightly correlated in a way that makes separating the effects difficult. This is one of the reasons why we were not able to reject the learning effect in the other two hypotheses. However, the results of our analysis related to these hypotheses lead us to think that the process phases and the introduced techniques are probably one of the main reasons of the changes, so further research and experimentation is necessary to confirm it.

With our results, we show that the use of PSP produces positive changes regarding the improvement quality of the software product, which is one of major needs of software development. Given the size and complexity of modern software projects, success requires that all individuals produce high quality software products with predictable cost and schedule. It is, therefore, essential to base organizational processes on practices that work at an individual level and satisfy these needs. This work suggests that PSP has demonstrated the capability to address these needs.

1.7 REFERENCES/BIBLIOGRAPHY

[Glass 72]
G. V. Glass, P. D. Peckham and J. R. Sanders, "Consequences of failure to meet assumptions underlying effects analyses of variance and covariance" *Rev. Educ. Res.,* vol. 42, pp. 237-288, 1972.

[Grazioli 12]
F. Grazioli and W. Nichols, "A Cross Course Analysis of Product Quality Improvement with PSP" *In Proceedings of the TPS Symposium 2012: Delivering Agility with Discipline. Special Report, Software Engineering Institute, Carnegie Mellon University CMU/SEI-2012-SR-015,* pp. 76-89, September 2012.

[Harwell 92]
M. R. Harwell, E. N. Rubinstein, W. S. Hayes and C. C. Olds, "Summarizing Monte Carlo results in methodological research: the one- and two- factor fixed effects ANOVA cases," *J. Educ. Stat.,* vol. 17, pp. 315-339, 1992.

[Hayes 97]
W. Hayes and J. W. Over, The Personal Software Process (PSP): An Empirical Study of the Impact of PSP on Individual Engineers, Technical Report CMU/SEI-97-TR-001, Software Engineering Institute, Carnegie Mellon University, December 1997.

[Humphrey 05a]
W. S. Humphrey, TSP: Leading a Development Team, Addison-Wesley, 2005.

[Humphrey 05b]
W. S. Humphrey, PSP: A Self-Improvement Process for Software Engineers, Addison-Wesley Professional, 2005.

[Humphrey 06]
W. S. Humphrey, TSP: Coaching Development Teams, Addison-Wesley, 2006.

[Jones 10]
C. Jones, Software Engineering Best Practices: Lessons from Successful Projects in the Top Companies, McGraw Hill Professional, 2010.

[Kemerer 09]
C. Kemerer and M. C. Paulk, "The Impact of Design and Code Reviews on Software Quality: An Empirical Study Based on PSP Data" *IEEE Transactions on Software Engineering,* vol. 35, no. 4, 2009.

[Lix 96]
L. M. Lix, J. C. Keselman and H. J. Keselman, "Consequences of assumption violations revisited: A quantitative revew of alternatives to the one-way analysis of variance F test" *Rev. Educ. Res.,* vol. 66, pp. 579-619, 1996.

[Paulk 06]
M. C. Paulk, "Factors Affecting Personal Software Quality" *Cross-Talk: The Journal of Defense Software Engineering,* vol. 19, no. 3, pp. 9-13, 2006.

[Paulk 10]
M. C. Paulk, "The Impact of Process Discipline on Personal Software Quality and Productivity" *ASQ Software Quality Professional,* vol. 12, no. 2, pp. 15-19, 2010.

[Rombach 08]
H. D. Rombach, J. Münch, A. Ocampo, W. S. Humphrey and D. Burton, "Teaching Disciplined Software Development" *Journal of Systems and Software,* vol. 81, no. 5, pp. 747-763, 2008.

[Sommerville 10]
I. Sommerville, Software Engineering - 9th Edition, Addison-Wesley, 2010.

[Tabachnick 89]
B. G. Tabachnick and L. S. Fidell, Using Multivariate Statistics, Harper Collins, 1989.