Project LEAP: Personal Process Improvement for the Differently Disciplined

Carleton A. Moore

January 4, 1999

Research Area

Process Improvement, Measurement, Personal Software Process

Problem

Software developers and managers have faced the problem of producing quality software since the beginning of the computer age. Many people have studied the software quality problem and have proposed solutions, including better testing, better organizations, better practices, better project tracking, better programming environments, and many other factors that potentially affect the development of software. We can categorize these different solutions into two groups: (1) solutions that focus on software development as a group effort and (2) solutions that focus on the individual software developer. Some of the many suggestions that involve groups of software developers include: the Capability Maturity Model, Clean Room development, software quality assurance groups, and Formal Technical Review. These organization level methods help improve the quality of the software, however they may not be enough.

In the past four years, there has arisen a new focus on the individual software developer. In "A Discipline for Software Engineering"[6] Watts Humphrey introduced the Personal Software Process, also known as PSP.

> PSP is a self-improvement process designed to help you control, manage, and improve the way you work. It is a structured framework of forms, guidelines, and procedures for developing software. Properly used, the PSP provides the historical data you need to better make and meet commitments and it makes the routine elements of your job more predictable and more efficient.[6]

PSP, an empirically based process improvement method, focuses on the individual software engineer. In PSP, software engineers record the time they spend programming, the defects they find in their software and the size of the software. Based upon these measurements, the engineer can track their productivity, make better predictions for future projects, gain insight to what types of

errors they make, and learn how to remove defects earlier in their development process. The PSP, as described by Humphrey, is a manual process. The engineer records, transfers and analyses the data all on paper forms. After many projects, the engineer accumulates a large paper database of their historical data.

After using the PSP for two years, we noticed three general problems. First, we started to question the quality of the data recorded. We noticed that we did not record all of our defects, in part because the overhead of recording each defect is too expensive. Anne Disney and Philip Johnson conducted a study to look at the data quality of PSP data. They found that there are significant data quality issues with manual PSP.[2, 3]

Second, our experiences with industrial partners and management practices and Robert Austin's book "Measuring and Managing Performance in Organizations"[1] made us think about the issues of measurement dysfunction in PSP and review data. An organization may pressure their members to produce "good" results. There are many ways that the members can manipulate the personal data collected in the PSP to get the "right" results.

Third, after four years, the results with long term adoption of PSP are mixed. Pat Ferguson and others report excellent results with PSP adoption at Advanced Information Services, Motorola and Union Switch and Signal[5]. However, Barry Shostak and others report poor adoption of PSP in industry[7, 4].

These issues started us thinking about designing an automated, empirically based, personal process improvement tool. Our goal is to reduce the collection and analysis overhead for the engineer, and the measurement dysfunction of the collection process. This should improve the benefits to the engineer and the long term adoption of empirically based process improvement. To pursue this work, we initiated Project LEAP, <http://csdl.ics.hawaii.edu/Research/LEAP/LEAP.html>, and began developing the Leap Tool Set, <http://csdl.ics.hawaii.edu/Tools/LEAP/LEAP.html>.

We designed the Leap Tool Set to automate much of the collection and analysis of PSP data. Based upon our design goal to reduce measurement dysfunction, we allow the user to control who sees what data and change the data that is shared. We also reduce the process constraints on the user. Leap does not require the user to follow a predefined, fixed process. Leap also incorporates collaborative review support. This allows the developer to gain insight from other developers. This group input is an important feature lacking in the PSP. We made these design decisions in an attempt to improve the usability and adoption of the Leap Tool Set.

Research Questions

We intend to deploy the Leap Tool Set in both academic and industry settings in order to investigate the following research questions:

- What are the strengths and weaknesses of the Leap Tool Set?
- What are the strengths and weaknesses of empirically based process improvement?

- What types of insights can we gain through empirically based process improvement? What types of insights can we not get through empirically based process improvement?
- What are the barriers to adoption of the Leap Tool Set
- What are the benefits to the users of Leap?
- How do users improve after using Leap?
- What are the kinds of improvements we can make using Leap beyond improved estimation?
- Is the integration of collaborative review and personal data collection appropriate?
- Is Leap an appropriate form of automated support for personal process improvement?

Based upon these research questions we have developed the following testable hypotheses:

- Automating the PSP will lead to improved adoption of personal software process improvement.
- Reducing the constraints on developers imposed by PSP will lead to improved adoption.
- Training in a LEAP compliant personal software process improvement tool set will lead to adoption of the tool set in professional practice.
- The LEAP case studies will provide valuable insights into the strengths and weaknesses of this technology.

Evaluation

To evaluate these hypotheses we plan on conducting two case studies. We will conduct the first case study on senior level undergraduates in Spring, 1999. The undergraduates will learn about empirically based process improvement and software engineering processes. We will train the students how to use the Leap Tool Set and they will use it to record their software development of several projects. This process is similar to the training of the PSP. The students will submit reports on their progress and findings. We will interview the students to determine their feelings and attitudes toward process improvement and the Leap Tool Set. We will also conduct a survey of their perceptions of the Leap Tool Set and their environment. The interviews and the surveys will allow us to predict the students' adoption of the Leap Tool Set and empirically base process improvement practices. A few months after they finish the class we will contact the students and ask them to fill out a survey. This survey will determine if they are still using Leap, and/or any of the empirically base process improvement concepts in their work. This survey will allow us to determine the level of adoption of Leap and empirically based process improvement, and any barriers to adoption of Leap.

We plan on conducting the second case study in an industrial research group during Summer, 1999. We will introduce the Leap Tool Set as automated support for collaborative review and personal process improvement. We will train the members on the use of Leap and help them conduct reviews and analyses. We will also train them on personal process improvement. During the training we will conduct a survey and interviews to determine their perceptions of Leap and personal process improvement. Again the surveys and interviews will allow us to predict the adoption of Leap. A few months after the training we will contact the members and conduct a survey and interviews. This survey will determine the level of adoption of Leap, any barriers to adoption, and the member's attitude toward personal process improvement.

Contributions

We expect the following contributions from this research:

- We have made the Leap Tool set available for down loading at <http://csdl.ics.hawaii.edu/Tools/LEAP/LEAP.html>. Leap is implemented in Java and runs on Windows, Unix, and Macintosh. The Leap Tool Set will provide a novel form of automated support for empirically based process improvement, including time, defect, size, and pattern recording and analysis. We will soon release the Leap Tool set as an Open Source system.

- Insight into adoption of personal process improvement. The results of this research will provide new insight into barriers to adoption of personal process improvement. The research will provide new insight into why users adopt or refuse to adopt personal process improvement in their work. We designed the Leap Tool Set to overcome some of the known barriers to adoption. If Leap is not adopted, then this suggests that these barriers are not key barriers.

- Insight into empirical process improvement. The use of Leap helps us learn what improvements we can make using empirical measurement. The case studies may provide insights into the limits of empirically based approaches to process improvement.

- Insight into process improvement issues. PSP uses the classic waterfall software development model, fixed forms, and a single size definition. Leap relaxes many of the constraints that PSP imposes. The case studies should help us learn if these constraints are required for process improvement.

References

[1] Robert D. Austin. *Measuring and Managing Performance in Organizations*. Dorset House Publishing, 1996.

[2] Anne M. Disney. Data quality problems in the Personal Software Process. M.S. thesis, University of Hawaii, August 1998.

[3] Anne M. Disney and Philip M. Johnson. Investigating data quality problems in the PSP. In *Proceedings of the ACM SIGSOFT Sixth International Symposium on the Foundations of Software Engineering*, pages 143–152, Lake Buena Vista, FL, November 1998.

[4] Khaled El Emam, Barry Shostak, and Nazim Madhavji. Implementing concepts from the Personal Software Process in an industrial setting. In *Proceedings of the Fourth International Conference on the Software Process*, Brighton, England, December 1996.

[5] Pat Ferguson, Watts S. Humphrey, Soheil Khajenoori, Susan Macke, and Annette Matvya. Introducing the Personal Software Process: Three industry cases. *IEEE Computer*, 30(5):24–31, May 1997.

[6] Watts S. Humphrey. *A Discipline for Software Engineering*. Addison-Wesley, New York, 1995.

[7] Barry Shostak. Adapting the Personal Software Process to industry. *Software Process Newsletter #5*, Winter 1996.