



# **Maximizing Personal Productivity to Reach Business Top Performance with Personal Software Process**

**Chaiyoot Chamnanlertkit**  
Industry Development Center  
The Stock Exchange of Thailand



## IT Service Business in the new paradigm

- Resource issues and the inability to recruit talented people are driving users to rely even more on Enterprise Service Provider and are moving them to a resource person role, rather than a supplier of services role.
- It is crucial for both parties to understand that the nature of the relationship required is highly dependent on the type of service to be delivered, from efficiency focused through effectiveness focused to competitive advantage.



 **While the software business has been troubled almost from the outset**

- **Unpredictable schedules**
- **Missed commitments**
- **Poor-quality products**



**Productivity** is a crucial concern for business and most important driver for achieving **satisfactory results** and **significant cost-advantage**



# Productivity : Software Development Business Perspective

## Team Leader and user

Development professionals who are credible and professional and can be trusted to do what they commit to do.

## Developer

Fun job, a rewarding development experience, and the satisfaction of doing professional work.

## Management/Stakeholder

Quality software on predictable schedules and for committed costs



Professional



## Software engineering professional

- Software engineers really believed, they can get a quality product if they would concentrate on finding or preventing the defects before the start of test
- Evidence that this strategy works. By using a defined and measured personal software process, engineers can improve the quality of their products by five to ten times while also improving their productivity.

To improve product quality, you must improve process quality.



A powerful counter to this criticism is the widespread adoption of the Personal Software Process (PSP) methodology.



# About the Personal Software Process (PSP)

## PSP Developed in 1993 by Watts S. Humphrey

- **Disciplined** and **structured** approach to developing software.
- This personal software process (PSP) is a promising way for engineers to **understand their own performance** and to see **how to improve** it.
- By using the PSP concepts and methods, engineers in almost any technical field can improve their **estimating and planning skills**, make **commitments** that they can meet, manage the quality of their work, and **reduce the number of defects** in their products.

• PSP came from questions about the Software Engineering Institute's (SEI) capability maturity model (CMM). Many viewed the CMM as designed for large organizations.

• The SEI thus started a process research project to examine ways individual engineers could apply level 5 process principles. After several years of research, means were devised to adapt 12 of the 18 CMM key process areas to the work of individual software engineers.



# Principal PSP quality measurement

**PSP quality measures is yield:** the percent of all **defects removed** before the first compile or test, percentage improvement in the average numbers of defects found per thousand lines of code (KLOC)

Where found	Class A	Class B	Class C	Class D
<b>Total defects</b>	<b>53.4%</b>	<b>45.8%</b>	<b>55.1%</b>	<b>80.1%</b>
<b>Compile defects</b>	<b>68.8%</b>	<b>76.6%</b>	<b>75.7%</b>	<b>88.1%</b>
<b>Test defects</b>	<b>68.8%</b>	<b>81.7%</b>	<b>64.2%</b>	<b>83.2%</b>

The numbers of engineers in classes A, B, C, and D were 4, 12, 6, and 19 respectively, in classes A, B, and C were moderately experienced engineers while most members of class D were little experience

Average LOC/Hour	Class A	Class B	Class C	Class D
<b>Exercises 1 and 2</b>	<b>19.9</b>	<b>31.4</b>	<b>11.4</b>	<b>13.8</b>
<b>Exercises 9 and 10</b>	<b>36.3</b>	<b>38.6</b>	<b>26.9</b>	<b>22.3</b>
<b>Percent Improvement</b>	<b>82.4%</b>	<b>22.9%</b>	<b>136.0%</b>	<b>61.6%</b>

**All groups improved, While many factors will influence these rates**

## PSP : Improvement areas

The PSP is a technique that engineers can apply to most structured personal tasks to improve their

**Predictability**

**Quality**

**Productivity**



**Practices for making individual commitments**

**Practices to ensure high-quality work products**

**Feedback loops to achieve continuous improvement**



## The quality strategy used with the PSP

- Tracking Quality
- Continuous Improvement





# Tracking Quality

- Defect history provides answers
  - “How many defects will I probably inject?”
  - “Where am I likely to find those defects?”
- By estimating defects, individuals can recognize poor quality and take corrective action



# Continuous Improvement

- Data is captured and analyzed after each development cycle
- Estimated and actual size and effort builds personal history
- Regression continuously improves estimation  
Analysis of defects results in actions to prevent
- Similar defects in the future



# Requirements to Implement PSP

## Individual

Discipline and commitment

Objectivity regarding performance data

## Organization

Strong  
management  
support

Investment in training  
and process support

Respect for data  
confidentiality



## PSP Results

- Predictability
- Quality
- Organizational Capability



“Good engineering practices. The programmers have to have data and quality plans. They also have to track quality and they have to own it. That involves a change from today’s typical attitude. That’s exactly what we’re talking about. That’s what PSP and TSP are designed to do.”

*Focus on Watts Humphrey  
A CAI State of the Practice Interview  
October, 2006*