

TaskPit: A Software Development Task Measurement System for Software Process Analysis

Pawin Suthipornopas, Pattara Leelaprute,
Akito Monden, Hidetake Uwano, Yasutaka Kamei,
Kenji Araki, Kingo Yamada, Ken-ichi Matsumoto

2015-07-07

1 Abstract

In this paper we propose a tool called TaskPit to automatically monitors software development tasks such as programming, testing, and documentation. TaskPit automatically records the time, keystrokes, and mouse clicks for each task, where a task is associated with a set of software applications and windows titles. It also records the time series data of size and the number of documents associated with each task. This paper also describes two case studies using TaskPit. The first case study has total of 12 people working in 6 days of measurement. (a) Leader is too engaged in the development work rather than management, (b) not many people are communicating by e-mail, (c) the time spent writing e-mail is not corresponding to the length of that e-mail, could use improvements in writing the correct form of e-mail to specific type of person. A clue of improvements from this project is found in the second case study where a single developer was measured in 13 business days. As a result (d) on working days there are more than half of unplanned work, but (e) the total work done passed the estimated number because the use of TaskPit to help manage their work as a developer. It is suggested that TaskPit could help in this improvement.

2 Introduction

As Tom DeMarco said 「 You can't control what you can't measure 」 it is essential to control the process of developing software [1]. For this purpose, various product processes metrics and measurement have been applied to the developing site [2]. In many of the environment of development site; the software scale, work hours, and bugs are measured and used for project management, and quality assurance of the development [4] [3].

References

- [1] Tom DeMarco. *Controlling software projects: Management, measurement, and estimates*. Prentice Hall PTR, 1986.
- [2] Robert B Grady. *Practical software metrics for project management and process improvement*. Prentice-Hall, Inc., 1992.
- [3] Lawrence Putnam and Ware Myers. *Five core metrics: the intelligence behind successful software management*. Addison-Wesley, 2013.
- [4] 直美 and 誉田. ソフトウェア品質会計: *NEC* の高品質ソフトウェア開発を支える品質保証技術. 日科技連出版社, 2010.