

Research Proposal

Study On Method Selection for Specific Software Project

Xiaohua Zhou
Xiaohua.Zhou@drexel.edu

November 28,2003

Study On Method Selection for Specific Software Project

Background

The philosophy of software development is a very big issue. It can be observed from different perspectives or levels including society-level, organization-level, project-level and individual-level. In each level, there exists the variety of methods, policies and principles, which can't be easily judged as bad or good without full considerations on context.

At the societal level, open source, as Linux is developed, is a very good strategy to ensure software quality and the security of information systems. However, this strategy also brings side effects such as version confusion. At the organizational level, the Capability Maturity Model is the most famous framework to guide the improvement and optimization of software process though it doesn't cover everything (domain, systems, project and people) and it is costly and difficult to be adopted. At the project level, there are many methods ranging from agile ones (e.g. Extremely Programming, XP) to plan-driven ones (e.g. TSP). At individual level, Personal Software Process is a famous framework to guide software engineers.

Regarding project-level methods, we can place various agile and plan-driven development approaches along a spectrum of increasing emphasis on plans. In this context, the term "plan" includes documented process procedures that involve tasks and milestone plans, and product development strategies that involve requirements, designs, and architectural plans. (Barry Boehm, 2002) Naturally, we will ask: *what factors does the extent of emphasis on plans for specific software project depend on? What is the relationship of the emphasis on plans over these factors? Negative or positive?*

Many qualitative researches have pointed out that the real project environment include architecture, requirement, size, developers, customers, refactoring, and primary objective. The environment of the project will affect the selection of emphasis on plans. So we will study on the second question here: What is the relationship of the emphasis on plans over these factors? Due to limits of time and effort, we only study the causal relationship of the extent of emphasis on plans over three factors, architecture, developers and customers.

Hypotheses

1. There is a negative relationship of the extent of emphasis on plans over the performance of all developers in the team.
2. There is a negative relationship of the extent of emphasis on plans over the

participation of the customer.

3. There is a positive relationship of the extent of emphasis on plans over the adaptability of the architecture.

Variables

Hypothesis 1:

Dependent variable: the extent of emphasis on plans

Independent variable: the performance of developers in the team

Hypothesis 2:

Dependent variable: the extent of emphasis on plans

Independent variable: the participation of the customer

Hypothesis 3:

Dependent variable: the extent of emphasis on plans

Independent variable: the adaptability of the architecture

Terms

1. Emphasis on Plans (EP)

Planning work includes: (1) have various meetings to discuss plans; (2) prepare written documents before coding; (3) Strictly carry out the plan; (4) Periodically check and track the plan. So a plan-driven team will take much time and effort on planning process. Usually, man-month or man-day is used to measure the cost of project. The percent of planning cost can be an indicator of emphasis on plans.

We should point out that some methods such as agile approaches also have planning process, which distinguish them from unplanned and undisciplined hacking method. But their plans focus on short-term goal only. So the time span of plans is also indicative of the emphasis on plans.

2. Performance of Developers (PD)

Since there is more than one developer in the team, performance of developers refers to the mean of all members including analysts, designers, programmers, and testers. The implications of performance here include (1) technical skills such as designing, programming, testing and documenting; (2) project experiences; (3) independent working ability; (4) communication skill; (5) teamwork spirit.

3. Participation of Customers (PC)

Software development is a game that both developers and customers participate in. Participation of Customers describes the extent customers cooperate with the development team, which can be very loose or very intensive in the practice.

In this context, participation of customers can be measured by (1) the time and effort customers contribute to the project, (2) the customer's proficiency in project related area, (3) communication skills, (4) satisfaction over the performance of developers, (5) common goal and vision.

4. Adaptability of Architectures (AA)

The adaptability of architectures depicts the ease with which software allows differing system constraints and user needs to be satisfied. An adaptable architecture not only meets the current requirements but also meet the foreseeable requirements. But an inadaptable architecture can meet the current requirements only.

Method

Web-based survey will be used to collect data in the study. Subjects will be asked to self-report the information concerning the last project he was responsible for through web-based questionnaire. After data collection, we will apply qualitative analysis to test hypotheses.

The advantages of web-based survey here:

- (1) Low cost.
- (2) High response rate considering subjects are Internet experts and surfer.
- (3) Large sample size.
- (4) Convenient for data process.
- (5) Strength in reliability.

The weakness of this method obviously lies in its weak validity. In order to get stronger validity, we will use multi-item scale to measure each variable. Subjects will be asked a group of questions from different perspectives for each variable.

The measurement consists of 18 questions, composed of 4 multi-item scales (see appendix). Multivariate analysis will be conducted for each item in a scale.

Subjects

Software Project Mangers are subjects in the study since they are the best one who know the whole story of the project.

Sampling is an important issue in the study. We plan to randomly select 500 subjects from registered users of some professional websites for software project manager, www.allpm.com, for example. The nature of these websites ensures that selected subjects are project managers, at least quasi-project manager in the real world. Besides, registered user of these websites can be considered representative of the population of project managers since the registration process of these websites has no special limitations.

Threats to Validity

Sources of Internal Invalidity

The survey is possible subject to the artificiality. The subject report that their team took a quite plan-driven method doesn't mean that the team chose a plan-driven method in real world. The underlying reason may be, for example, that the subject traditionally thinks that a project manager ought to make the project quite organized. The subjectiveness of some questions will also threaten the validity of the research. For example, the judgment of the performance of the team is rather subtle.

Sources of External Invalidity

The sampling of subjects possibly threatens the external validity of the survey. Even if the sampling is ideal, the respondents may not be very typical. One explanation is that those managers of large projects have high possibility not to response because they are very busy.

Data Collection

We plan to cooperate with www.allpm.com to complete the task of data collection. The questionnaire (see appendix) will be posted on the website of www.allpm.com. Then send the instructions and link to the selected subjects via email. Subjects' answers will be automatically recorded in a database. The online survey will stop two weeks later.

Table 1 Original Data Sheet

Subject	EP					AA			PD					PC				
	1	2	3	4	5	1	2	3	1	2	3	4	5	1	2	3	4	5
#1																		
#2																		
#3																		
...																		

Data Analysis

Take the example of first hypothesis to show how to do data analysis. First, for each sample, calculate the scale score of EP and PD by obtaining the mean of their multi-items' score respectively.

Second, check the internal validation of scale by multivariate analysis.

Third, divide samples into three groups according to the performance of developers (PD). The three groups are respectively: low, middle and high. Further, we calculate the mean of EP of three groups. If the mean of low group is greater than the middle group and the middle group greater than high group, the hypothesis is true. Or the hypothesis is false.

Table 2 Groups by PD

PD	Low	Middle	High
Mean EP of Samples			

References

Paulk, M.C., B., and Chrisis, M.B. Performance maturity model, version 1.1. IEEE Software, July 1993, 18-27.

Bamberger, J. Essence of the performance maturity model. Computer, June 1997, 112-114

Ferguson, P., Humphrey, W.S., Khanejpoori, S., Macke, S., and Matvya, A. Result of applying the personal software process. Computer, May 1997, 24-31.

Beck, K. Embracing change with extreme programming. Computer, October 1999, 70-77.

Boehm, B., (2002) Get ready for agile methods, with Care, Computer, p. 64-69. January.

Appendix: Questionnaire

Extent of Emphasis on Plans

	Strongly disagree	strongly agree
1. Number of meetings for plan discussion.	1 — 2 — 3 — 4 — 5 — 6	
2. Well written documents	1 — 2 — 3 — 4 — 5 — 6	
3. Long-term plans	1 — 2 — 3 — 4 — 5 — 6	
4. Strictly carry out the plan	1 — 2 — 3 — 4 — 5 — 6	
5. Periodically check the plan	1 — 2 — 3 — 4 — 5 — 6	

Performance of Developers

	Strongly disagree	strongly agree
1. Strong technical background	1 — 2 — 3 — 4 — 5 — 6	
2. Many years project experiences	1 — 2 — 3 — 4 — 5 — 6	
3. Independent working ability	1 — 2 — 3 — 4 — 5 — 6	
4. Good communication skills	1 — 2 — 3 — 4 — 5 — 6	
5. Teamwork spirit	1 — 2 — 3 — 4 — 5 — 6	

Participation of Customer

	Strongly disagree	strongly agree
1. Enough efforts on the project	1 — 2 — 3 — 4 — 5 — 6	
2. Good communication skills	1 — 2 — 3 — 4 — 5 — 6	
3. Proficient in project-related area	1 — 2 — 3 — 4 — 5 — 6	
4. Satisfied with your performance	1 — 2 — 3 — 4 — 5 — 6	
5. Common goal and vision	1 — 2 — 3 — 4 — 5 — 6	

Adaptability of Architectures

	Strongly disagree	strongly agree
1. Adaptable to changing requirement	1 — 2 — 3 — 4 — 5 — 6	
2. Architecture for heterogenous platform	1 — 2 — 3 — 4 — 5 — 6	
3. Architecture for long-term use	1 — 2 — 3 — 4 — 5 — 6	