

# An Experience on Software Development Organizational Adoption and Implementation of the Preliminary Measurement and Estimation to Appraise Software Projects

**Jutawan Saemu**

*Avalant Co., Ltd.*

*Silom, Bangrak, Bangkok, 10500 Thailand*

*jutawan\_saemu@yahoo.com*

## **Abstract**

*Increasing productivity and quality, and decreasing cost and time to market are common goals of organizations developing software products. It is the fact that the source of good products comes from effective processes. To know what the current being, the measurement methodology is an approach to survey any attributes among software development life cycle. Furthermore, the results of measurement are the index to show organizational capability. This paper proposes an experience of a software solutions development company in Thailand that applies the software engineering discipline e.g. software measurement, project estimation to evaluate the internal software process and the project productivity.*

**Keywords:** Software Development Organization, Project Measurement, Metrics, Project Evaluation, Function Point (FP) Analysis, Productivity.

## **1. Introduction**

Software development and maintenance of existing software is a growing constituent of our economy. Quality and timely delivery of software within reasonable costs are critical for the success of many business organizations today. So the way to gain effective products is to construct effective processes since the beginning of state.

Software Engineering Methodology is one of the major solutions, recently proposed for avoiding and managing the problems frequently associated with software development, is to treat the entire development task as a process that can be measured, controlled, and improved [9]. This approach helps to force discipline into the methods and practices adopted during software development. In general, an organizational process can be defined as a logical organization of people, technology, and practices into work activities

designed to transform information, materials, and energy into a specified end result [6]. A software process can be defined as a set of partially ordered activities performed during the development or evolution of a software product or system [1].

Avalant Company Limited is established two years ago with the commitment of the software solutions provider. There are two service lines under organization roof that consists of *E-Business Service Line* and *Solution and Technology Service Line*. ***E-Business Service Line*** provides best-in-class e-business technology solution that solves client's business problems. The consulting services in this area include the Application Development, Business & Application Integration and Portal Strategy and Development. ***Solution and Technology Service Line*** provides client with infrastructure development, build and maintenance services. Develop standard methodology and framework to accelerate solution development and deployment for the clients. Currently, Avalant employs about 90 staffs to maintain 12 – 15 projects (with different software size scale). The common project roles are defined into seven roles: project manager, business lead, technical lead, analyst, software engineer, tester, and developer.

After the projects go on in a period of time, we found some problems with both operation and management including the increasing of new staffs and there are many development teams, the existing monitoring will be not thoroughly. Then we would like to know the current status of project in several of project attributes, in order to set the direction of problem solving and process improvement in appropriate way.

This paper presents the experience of preliminary measurement of software project by illuminating our approach to define the metrics and measurement process by next section explains steps of measurement preparation. Section three shows measurement performing. Section four introduces the supporting tools. Section five reveals the measurement output. Finally, the conclusion displays in section six.

## 2. What we plan?

To be effective, a plan must be started as a set of targets, the achievement or non-achievement of which can be unambiguously measured [2]. We investigate the as is components in our organization from existing software processes in the past and the on-going project. Our main goal is to measure basic project attributes in order to analyze what process should be improve and to create the baseline for next repeatable in other projects.

We defined the tasks plan for the project appraisal as the details in Table 1.

**Table 1: Task plan**

<b>Task No.</b>	: 1
<b>Task</b>	: Identify metrics
<b>Description</b>	: Define the metrics for measuring project attributes.
<b>Output</b>	: List of metrics (see Table 2)
<b>Task No.</b>	: 2
<b>Task</b>	: Select subjects (project)
<b>Description</b>	: Select the projects that there are not different in the development technique but the project size maybe different in order to compare the result of measurement.
<b>Output</b>	: List of selected projects.
<b>Task No.</b>	: 3
<b>Task</b>	: Collect data.
<b>Description</b>	: Gather project data from required attribute list to be the raw data of measurement.
<b>Output</b>	: <ul style="list-style-type: none"> <li>- Individual person-hour</li> <li>- The number of unique function types: <ul style="list-style-type: none"> <li>o External inputs</li> <li>o External outputs</li> <li>o Queries</li> <li>o External files or interfaces</li> <li>o Internal files</li> </ul> </li> <li>- The number of project defect.</li> </ul>
<b>Task No.</b>	: 4
<b>Task</b>	: Perform estimation and measurement.
<b>Description</b>	: Pool project data into calculation model to find measurement results.
<b>Output</b>	: <ul style="list-style-type: none"> <li>- Project person-hour</li> <li>- Project size (represent in format of adjusted function point count)</li> <li>- Defect ratio.</li> </ul>

## 3. What we do?

According to the model of estimation measurement, we follow function point analysis to measure software size. Then the team of process improvement introduces the concept of project measurement through the project representative to let them help us in function estimation. The other data e.g. project work hour, number of defect we automatic get from the system. Avalant has already

developed the system to manage about the individual work effort and to maintain defects of project by using Avalant Timesheet System and Avalant Change Request System respectively that the details will be show in section 4.1 – 4.2.

After the introduction the source of input of project appraisal process, the steps below we describe the cycle of our measurement.

### 3.1 Data Collection

We collect work-hour into the system everyday. The project staff has own responsibility to input his/her tasks and effort to Timesheet system by weekly submit.

On the other hand, the defect information feed into the system by occurring chance.

The function size data is gathered by brainstorming and discussing with project representatives who know the project scope. They give the estimation data or count the exact number of each attribute we require.

### 3.2 Data Processing

We create the spreadsheet as the template of project information collection. When the project data complete, we enter them into the worksheet to get the measurement results.

### 3.3 Data Analysis

We distinguish the results according to project criteria e.g. project size (small, medium, large), development technique (web-based development, report development), development language (Java, PL/SQL, C, C#). Because of the first time of measurement, it is no historical data to compare then we use industry result to be the benchmark. After we repeat the measurement, maybe on the similar project domain or the other project with similar size criteria, we will get some information to indicate what area that we should improve.

## 4. What we use?

The motivation to construct the project measurement is to improve the existing processes to be more controllable and get better performance. By doing this, we use the many areas of knowledge to perform the project appraisal. The main discipline is Software Engineering. The other fields are the theory of Project Management, Software Metric, Software Estimation Techniques including the case studies and the papers. Furthermore, the other important to help in process of data collection is the software tool. As earlier mention, there are two systems to collect data for this project measurement. The details of them show in this section.

#### 4.1 Avalant Timesheet Systems®

**Tool Objective:** To collect work effort as individual person-hour into the system.

**System User:** Avalant Staffs.

- Functionality:**
- Record task, phase, type of activity, project name, work hour of staff.
  - Summarize as weekly report and automatic submit to project manager.
  - After project manager approve, the system will submit project report to project coordinator to collect as the work effort of each project.

#### 4.2 Avalant Change Request System®

**Tool Objective:** To record defect information of each project and disseminate to developers who made defect to fix including send to project manager, analyst, and clients to verify the result.

**System User:** Project team member.

- Functionality:**
- Record defect information (symptom, location, found date-time, target date, owner, etc.)
  - Report to project manager and client.
  - Record comment from related verifier.
  - While fixing complete, the developer submits the fixing result back to sender or founder.

### 5. What we gain?

Measurement has two broad uses: for assessment and prediction [5]. Examples of using metrics for assessment are: 1. Monitoring the advancement of a project in order to take the appropriate corrective decisions, 2. Evaluating a software product or process.

When using measurement for prediction, predictive measures can be used for: 1. Planning the resources and time needed for a certain project, 2. Predicting the outcome of a project, in terms of size, quality or other attributes of the delivered software.

In both cases, software measurement can provide a quantitative support to decisions that were previously taken on the basis only of subjective factors. The project manager can now take the decision of which tool or language to use or of the kind of process to adopt using data from previous projects instead of relying only on intuition.

After taking the project attributes into measurement process, the output we got is more useful. The direct outcome is individual project performance to show productivity of employee in each project and the average performance value of our organization. The measurement results are collected as historical data to use for the other measurements. Besides, the assessment results are the indicator of improvement as the ultimate empiricist, Lord Kelvin, [7] said, "If you cannot measure it, you cannot improve it".

**Table 2: List of metrics in this measurement.**

Type	Metric	Purpose	Support Tool
Basic Measurement	Function Size	To measure the size of projects and applications It can be expressed by Adjusted Function Points	IFPUG
	Work Effort	To measure the total of man-hour expended on application development.	
Productivity	Individual Work Effort	To measure the total of man-hour of software development per project.	Avalant Timesheet System
	Individual Work Effort per Phase	To measure the total of man-hour of software development per project phase.	
	Work Effort Ration per Role	To measure the percent of total man-hour of each role in project.	
Quality	Defect Found Ratio	To measure the quality of the new development and application enhancements delivered to the user. (Count all defects of project)	Avalant Change Request System
	Mean Time to Repair Ratio (MTTR)	To measure the average elapsed time needed to repair defects, showing how quickly defects are fixed.	
	Defect Detection Ratio	To measure the number of defects by phase are detected, the severity of defect, and the cause of defect.	
	Defect Remove Ratio	To measure the number of defects by phase are fixed or removed.	
Financial	Project cost	To measure the relative cost of the project. (This cost calculates from the summation of operation cost, transportation expense <sup>1</sup> , stationary and document preparation expense)	Spreadsheet Software
	Cost per Module	To measure the cost of application development per application module.	

### 6. Conclusion

Measurement can perform with several project attributes aspects e.g. work effort, cost, defect, and productivity. The result of measurement leads to apply in both assessment and prediction. In our research, we use this evaluation like a ruler to

determine approach for continuous improvement. The proposed information just would like to present what we do for project measurement. However, in the measurement world, there are many techniques that suit for applying in different purposes.

<sup>1</sup> There are the project workshops for system requirement gathering sometimes held on customer site and the activities through the system integration test (SIT) phase and user acceptance test (UAT) phase also operate on customer site.

## 7. References

- [1] B. Curtis, M. Kellner, and J. Over, *TMProcess Modeling*,<sup>o</sup> Comm.ACM, pp. 75-90, Sept. 1992.
- [2] B. Hughes and M. Cotterell, *Software Project Management*.: McGraw-Hill, 3<sup>rd</sup> Edition, 2002.
- [3] Buglione, L.; Abran, A.; Meli, R.: How *functional size measurement supports the balanced scorecard framework for ICT*. Proc. of the 4th European Conference on Software Measurement and ICT Control, FESMA-DASMA 2001, May 2001, Heidelberg, Germany, pp. 259-272
- [4] Fabrizio R., "*A Survey of Software Metrics*", Technical Report no. DEIS-LIA-96-010 LIA series no. 17. 1996.
- [5] Fenton N.E., "*Software measurement: a necessary scientific basis*", Journal - IEEE Trans Software Engineering, vol. 20(3), pp. 199-206, 1994.
- [6] G. Pall, *Quality Process Management*. Englewood Cliffs, N.J.: Prentice Hall, 1987.
- [7] Ian S., *Software Engineering: Challenges for the 21st Century*, pp.
- [8] M.L. Russ and J. D. McGregor, "*A Software Development Process for Small Projects*", Journal - IEEE Software, Sep/Oct, pp. 96-101, 2000.
- [9] W.S. Humphrey, "*Managing the Software Process*". Mass.: Addison-Wesley, 1989.