Tracking Software Development Progress with Earned Value and Use Case Point

Guixiang Liu
School of Computer Engineering, Qingdao Technological University, Qingdao, China
Email: liuguixiang@qtech.edu.cn

Abstract—The earned value method is an internationally recognized project management tool for tracking progress on projects. It periodically calculates the earned value of their completed project tasks, discovers the schedule and cost variances of the projects from the baseline, and thus helps to timely take measures to adjust and control the projects. Despite its popularity, it has not been widely applied to the software development projects. After a brief introduction to the earned value method, the key issues facing its applications in software development projects were analyzed. Those how to use the function point to quantitatively represent the software project baseline and objectively, accurately and consistently measure the costs, progress and completed work together into a simple indicator to monitor and control the project progress. The earned value method combines the accomplishment of work, progress and costs together into a simple indicator to monitor and control the project progress. The earned value method is the technique to measure the project performance, which incorporated scope, time and cost data. Given executing the baseline plan of projects, project managers and its team, the actual project information are inputted into and compared with the baseline process, which can indicate at which scale of satisfaction of scope, progress and costs are. The baseline plan is made at the beginning of projects. Actual project information mainly includes whether a certain task is complete or not, how much it completed, its actual start and ending dates, its actual expenses, etc.

Earned value method need to calculate three values of each project: Budgeted Cost of Work Scheduled (BCWS), Actual Cost of Work Performed (ACWP) and Budgeted Cost of Work Performed (BCWP).

BCWS is also a planned value, which is the funds which need to input to complete a certain work at a certain time based on the approved and accepted progress plan and budget. It is a ruler or standard to measure the project progress and its cost.

ACWP is the funds actually consumed for the accomplishment of work at a certain time of implementing a project, it is also called the actual value.

BCWP is the funds supposed to input to complete the certain work at a certain time of implementing a project according to its budget. It indicates the actual work progress of satisfaction of quality standard, realizes the transformation of results from inputs. In general cases, project owners make payments to project contractors for the accomplishment of work based on this value, that is the amount what the contractors earned, therefore also called Earned Value (EV). Earned value is the only data in earned value method to monitor and control the process of software development.

Index Terms—earned value, use case point, software development, project management

I. INTRODUCTION

In order to manage and control projects, it must to measure their performances throughout the entire life cycles, such as costs, progress and completed work. The earned value method is an internationally recognized project management technique to measure the performance of engineering projects. However, there are short of applications of this method in the area of software development, due to key factors that how to objectively, accurately and consistently measure the costs, time and completed work.

This article provided a software progress management technique, which combined the earned value method and function point technique based on Unified Modeling Language (UML), quantitatively track, monitor and control the process of software development.

II. EARNED VALUE METHOD

A. Application of Earned Value Method in Project Management

Progress, costs and quality are three basic objectives of project management. However those objectives are separated, individually controlled and managed by the common project management practices, which caused many problems. For example, the shortage of fund causes progress delay or poor quality, or when discovering severely over budget, it is too late to recover it.

Earned value method combines the accomplishment of work, progress and costs together into a simple indicator to monitor and control the project progress. The earned value method is the technique to measure the project performance, which incorporated scope, time and cost data. Given executing the baseline plan of projects, project managers and its team, the actual project information are inputted into and compared with the baseline process, which can indicate at which scale of satisfaction of scope, progress and costs are. The baseline plan is made at the beginning of projects. Actual project information mainly includes whether a certain task is complete or not, how much it completed, its actual start and ending dates, its actual expenses, etc.

Earned value method need to calculate three values of each project: Budgeted Cost of Work Scheduled (BCWS), Actual Cost of Work Performed (ACWP) and Budgeted Cost of Work Performed (BCWP).

BCWS is also a planned value, which is the funds which need to input to complete a certain work at a certain time based on the approved and accepted progress plan and budget. It is a ruler or standard to measure the project progress and its cost.

ACWP is the funds actually consumed for the accomplishment of work at a certain time of implementing a project, it is also called the actual value.

BCWP is the funds supposed to input to complete the certain work at a certain time of implementing a project according to its budget. It indicates the actual work progress of satisfaction of quality standard, realizes the transformation of results from inputs. In general cases, project owners make payments to project contractors for the accomplishment of work based on this value, that is the amount what the contractors earned, therefore also called Earned Value (EV). Earned value is the only data or information in earned value method to monitor and control the project progress. It is not the actual amount of payments or a payment standard. Other indicators include as following:

Cost Variance (CV) = BCWP – ACWP, CV>0, indicates expenses under budget; CV=0, indicates expenses equal to the plan; CV<0, indicates over budget.

Schedule Variance (SV) = BCWP – BCWS, SV>0, indicates ahead of schedule; SV=0, indicates schedule as planned; SV<0, indicates behind schedule.

Similar indicators, such as Cost Performance Index (CPI) = BCWP / ACWP and Schedule Performance Index (SPI) = BCWP / BCWS.

© 2009 ACADEMY PUBLISHER
AP-PROC-CS-09CN004 596
CPI is used to calculate Estimate At Completion (EAC), which uses EV to forecast the funds spent for completion of project based on the implementation of current project. For the similar token, SPI is used to estimate the time needed to finish a project. Budget At Completion (BAC) is the budget cost for all tasks of a project, that is total budget of the project. Variance At Completion (VAC) = EAC – BAC, BAC indicates completion budget, that is the sum of BCWS for all workloads of the project.

When actually applying the earned value method, it is generally shown BCWS, BCWP and ACWP, as well as EAC and BAC, etc of the project on earned value line chart.

Figure 1 shows the EV line on the time scale of 10, including SV based on time calculation and SV based on cost calculation. Cost SV indicates the difference between the actual time and EV time spent under the condition of accomplishment of the same amount of tasks. Time SV indicates the difference of the actual time and EV time used under the condition of the same amount of funds consumed. The chart illustrates the forecast of project progress (showed as dotted line) and EAC as well as variance of BAC, VAC.

EV method is used to establish the aggregated baseline of project activities, periodically calculates BCWP and ACWP and as well as SV and CV during the course of a project to forecast project completion time and final cost, and scientifically directs us to timely adjust project management to realize the project objectives.

B. Key Technological Issues of EV

There are two key issues which have to resolve when applying EV. First one is how to breakdown all works of a project into a list of work packages including tasks, schedules and funds, generally expressed as a tree Work Breakdown Structure (WBS). The other one is how to use an exact same indicator to show the plan of each work package, to measure schedule and expenditure of a project, i.e. corresponding EV BCWP and actual cost ACWP in order to finish the work package.

Work package is an independent and smallest unit which can be planned, managed and controlled, which has definite techniques and management, etc of project contents, and distributing work duration and budget BCWS. When all work packages are done, the entire project is completed.

The calculation of EV and the size of work package in WBS, the confirmation of task completion and the cycle of recording completion of work are intimately related. If a project breaks down too rough, gradation of work package too large, the cycle of data collection too short, work package probably unfinished, therefore it is impossible to calculate EV; if the cycle is too long, data obtained have no value to managerial measures.

The method to set EV of completing a task is get the ratio of a completed task and planned quota value. For example, confirming the price per square meter before painting wall, calculating EV by the area of completion of painted, or timely using the completed wall as a milestone and making budget, once the milestone is achieved, related EV is obtained.

During last few years, there are researches on how to apply this technique in software development, mainly focusing on how to define WBS in the specific software procedure, such as spiral development, Scrum method of agile development and improvement of software procedure. However, no single united measurement standard of EV to confirm completion of software development task is provided. This standard not only is able to show the project scale, but also to show the scale of task of WBS; it can indicate the project plan, become the project baseline, as well as measure the project schedule and reflect the project progress.

Due to the human resource cost is the largest part of software projects, man month workload may be used to record the actual value of project expenditure; however, there are no cause-result relation between the time and progress of a project. Furthermore, this measurement unit is overkill for small and mid size projects. Therefore, it is not suitable to use it to plan tasks and to calculate EV.

The lines of source code are very appropriate to estimate the workload of a project and record the amount of work accomplished. Nevertheless, there are two severe shortcomings using lines of source code to represent EV. First one is that it is hard to precisely estimate the lines of source code for the software to be developed at the earlier stage of analysis and design of a project. The other one is that in the task of software development, besides of coding there are analysis, design, verification and other many works. Furthermore, those works take far large proportion than coding task. Therefore, the lines of source code of a system are not appropriate either to express EV to indicate project budge as well as performance of work completed.

Function points, being capable to estimate in the earlier stage of software development, and expressing as the small amount workload, is an appropriate measurement to monitor and control a project in applying EV.
III. EV BASED ON FUNCTION POINT

A. Function Point Method

Function point, based on users’ point of view, is a standard method to measure software development project or to improve the scale of the project.

Function point measures the scale of software from function point of view, being independent of development languages. Once obtaining needs of users’ functions, it can be used to measure the scale. Thus it is able to quantify the scale at the earlier stage of the project.

With the advancement of software development, there are various variations and improvements of the function point method, such as Mark II function point, overall function point, as well as object-orientated technique function point and use case model based use case point.

B. Use Case Point

Use case modeling is a technique vastly used in the field, which describes and captures the function needs of software systems. Use case point, just applying use case modeling and its concept and use case, making the improvements to the calculation method of conventional function point, becomes a new method to estimate the amount of work involved in software development. Since the use case and scenario techniques are considered as standard components of needs of collection and analysis, they can be used to capture the precise description of users’ needs. Therefore, it is extremely meaningful to apply use case and scenarios to estimate the scale and amount of resources needed in software development. Moreover, it is more vigorous and stable than traditional function point and lines of source code.

The calculation procedure of use case point mainly includes following several steps:

1). Actor Weight. There are three types of actors by complexity as shown in Table 1.

<table>
<thead>
<tr>
<th>Actor type</th>
<th>Description</th>
<th>Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>Program interface</td>
<td>1</td>
</tr>
<tr>
<td>Average</td>
<td>Interactive and protocol interface</td>
<td>2</td>
</tr>
<tr>
<td>Complex</td>
<td>Graphic interface</td>
<td>3</td>
</tr>
</tbody>
</table>

2). Use Case Weight. There are three types of use case by complexity as shown in Table 2.

<table>
<thead>
<tr>
<th>Use case type</th>
<th>Description</th>
<th>Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>Max 3 transaction, max 5 analysis classes</td>
<td>1</td>
</tr>
<tr>
<td>Average</td>
<td>4-7 transactions, 5-10 analysis classes</td>
<td>2</td>
</tr>
<tr>
<td>Complex</td>
<td>More than 4 transactions, more than 10 analysis classes</td>
<td>3</td>
</tr>
</tbody>
</table>

3). Combining the sum of use case weights and the sum of actor weights gives Unadjusted Use Case Points (UUCP).

4). Technical Factor Weight. Technical complex factors can be used to calculate the complexity of the project technique. The scale from 0 to 5 is used to give a grade for each factor. Grade 0 means that this factor is not related to the project, and grade 5 represents it the most important. Every factor is multiplied by its weight, and then adding them together.

5). Environmental Factor and Weight of Subgroup. The scale from 0 to 5 is used to give a grade for each factor. Grade 0 means that this factor is not related to the project, and grade 5 represents it the most important.

By weighting Technical Complexity Factor (TCF) and Environmental Factor (EF), Use Case Point (UCP) is finally obtained, and the formula as follow:

\[ UCP = UUCP \times TCF \times EF \]

Attentions, a use case is not equal to a use case point. From one use case, generally several use case points can be calculated, ranging from 1 to 15. By using the function point technique, we consider use case point used in use case calculation as a measurement unit, and referred as UCP.

C. Earned Value Method Based on Use Case Point

Contrasting to function point, use case point is more suitable for earned value method of the modern software project. First, a use case, being system function, relating with design model and realization of codes, is convenient to orderly track the design and source codes by use case; contrastively, it can realize conversely tracking. That enables the measurement of software scale based on use case UCP to represent major products of software development, and to measure the performance of major development activities. Second, experimental research already confirmed that one UCP is equivalent to the scope of amount of work, ranging from 15 to 30 hours, being under a week of amount of work, which is easy to make budget and to record status. Finally, use case technology already became a part of unified modeling language (UML), generally accepted as a needed analyzing tool, and widely used in of modern software development projects.

According to the primary principles of earned value method and function point, we propose a way to monitor and control software development, simply described as fallow.

In the Phase of Project Planning:

1). Choosing a software development process, using one week’s work as an upper limit to defined the tasks of each work package, including technical activities, team members communication, managerial tasks and so on, and establishing the project WBS.

2). Analyzing users’ requirements, constructing system use case, calculating corresponded UCP based on use case within each work package, and becoming an indicator for the project schedule plan and cost budget.

3). Allocating the duration and resources for each work package, completing the project schedule and budget, and establishing the project milestone and baseline.
In the Phase of Project Implementation:

4). Periodically collecting project running data, including the actual cost, task finishing condition, and calculating EV, SV, CV and EAC, etc by UCP of completed tasks.

5). Evaluating the project schedule by analyzing SC, evaluating the project cost by analyzing VC, forecasting the project completion state by analyzing VAC, further directing to how to adopt the management measures in order to proceed the project as planned.

Due to the adoption of use case as a measure indicator of the software project and project task, this method may suit to any software development models or processes. No matter what software development models, they contains activities such as analysis, design, implementation, verification and project management, etc. Because of earned value defined as system use case points, use case can be truly completed only when activating software; further the analysis, design and test activities, although continuously specifying system and gradually producing project ultimate output, but not delivering the final product expressed in use case. In order to set up use case and calculate the earned value of completing tasks at each time period, there is a need to establish the corresponding UCP ratio for the various activities of each use case.

Table 3 showed the WBS distribution of general software development provided by Royce.

<table>
<thead>
<tr>
<th>WBS element</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project communication and management</td>
<td>10%</td>
</tr>
<tr>
<td>System analysis</td>
<td>15%</td>
</tr>
<tr>
<td>Soft design</td>
<td>20%</td>
</tr>
<tr>
<td>Components implementation</td>
<td>30%</td>
</tr>
<tr>
<td>Construction and testing</td>
<td>25%</td>
</tr>
</tbody>
</table>

IV. EXPERIMENTAL RESEARCH

In order to verify the feasibility and effectiveness of the proposed software project state tracking method, we planned supervising a student project development process and collected relevant data. This student group task is to re-develop a car rental system (ARS) within five months. Its use case is illustrated in Figure 2.

The experimental objective is to verify that during a process of iterative and incremental development, it is capable or not to discover the software development schedule and cost difference with use case point and earned value method. The research focused on examining whether or not use case points can be consistently used as a part of project plan baseline, being as the earned value of the complete project work.

Since we already had developed the ARS software before, the content, duration and staff requirements of the project are basically known to us. The main project data of this experiment are as follow: 4500 yuan (Chinese dollars) project budget, 20 weeks (5 months) of duration; one supervising teacher, at least 50 hours of work for the entire project, 30 yuan per hour; 3 students with mastering software development techniques, 50 yuan per week each one.

The simplified version of Untied Software Development Process (USDP) was adopted in this project, and the main activities of the project were shown on Figure 3. These activities executed more than once by the iterative approaches. We required at least 3 iterations during the entire project.

The earned value calculation is as follow. By the system use cases of Figure 2, we calculated that the system scale was 75 UCP. For the convenience of planning and measuring the development progress, we converted the earned value for completing each UCP into an amount of money, i.e. 40 yuan per UCP.

The work of “project communication and management” was spread throughout the entire project process, and besides of the regular scheduled weekly meeting there was no fixed times. For this we adopted an equally shared calculation approach, and allocated 10% of the earned value for each use case to the entire project cycle. Such the correspondent earned value for each completed UCP is: 40 × 10% / 20 = 0.2 yuan.

According to the convention of the field, the project data were collected once a week. The students submitted their project performance tables (mainly completed works and count of their UCP) on every Friday. The actual number of weeks and cost of every student were used to calculate the actual cost ACWP per week, and calculated the earned value BCWP by the project baseline and Table 1.

For example, in a project performance student A recorded the data for completing the design and test of
the use case “clients confirmation”, equivalent to 5 UCP, and thus the earned value he achieved was $5 \times 20\% \times 40 = 40$ yuan.

By adding the shared earned value of the project management, the earned value for student A was $40 + 0.2 / 3 = 40.07$ yuan.

Each week by adding the earned values of all three students and the supervising teacher together, the work earned value for this week was derived.

Figure 3 is the earned value line chart for the twelfth week of the project implementation. On the foundation of the tenth week (Figure 1), it strengthened the communication and supervision for students, and enabled to slightly raise the earned values for weeks 11 and 12. At the mean time, it lowered the actual students’ expenditure (due to the financial penalty for the students’ uncompleted plan). Although the total SV and CV of this week were larger than that at tenth week, the performance CPI of the twelfth week had improved, and the forecasted completion cost EAC (5050) was slightly lower than the number (5300) of the tenth week.

From the experiment process and the data collected, we found out that the use case point is not only to become an indicator for the project baseline plan and forecast, but also to become to measure the project implementation performance; and the proposed use case point based earned value method was helpful to exactly, timely report the project performance, to track the progress and cost in the software development, and to assist the project manager to apply the effective management practices.

V. CONCLUSIONS AND FUTURE WORKS

This article provides a method to quantitatively monitor and control the software development process, combining with the function point method of software engineering and the earned value method of project management, which is helpful to achieve the scientific, detailed and precise software development management. The experiment research has primarily proved the feasibility and effectiveness.

During calculating the earned value the Royce experience data were used, which had no definite theoretical foundation, and this need to go through the scientific experiment and verification as well as adjustment of the distribution ratio of development activities.

Establishing the better project plan is a fundamental to apply this method. In the future, there are needs to accumulate the data, to which the earned value and use case points are applied, such as various software project types and scale budget, task division and project duration, etc.

REFERENCES


