Measuring Change in Software Projects Through an Earned Value Lens

Pinar Efe[1](✉), Onur Demirors[2,3], and Boualem Benetallah[3]

[1] Bilgi Grubu Ltd., Ankara, Turkey
pinar.efe@gmail.com
[2] School of Computer Science and Engineering,
University of New South Wales, Sydney, Australia
[3] Department of Computer Engineering,
Izmir Institute of Technology, Izmir, Turkey

Abstract. Earned Value Management (EVM) is a common performance management tool for project management. EVM enables depicting the project progress in terms of scope, cost and schedule and provides future predictions based on trends and patterns. Even though EVM is widely used in various disciplines like manufacturing and construction, it is not common in software industry. One reason for this underutilization is the mismatch of an inherent nature of the software projects and the traditional EVM. Traditional EVM ignores change effort but it is predominant in software projects. We have developed cEVM as an extension to the traditional EVM to incorporate change and subsequent rework and evolution costs to measure earned value in software development projects more accurately. In this study, we focus on two applications of cEVM we performed to explore the usability of cEVM and to compare cEVM with traditional EVM. This paper discusses the results of the case studies as well as benefits and difficulties of cEVM.

Keywords: Earned Value Management · Software project management
Performance Measurement · Reworking · Change management

1 Introduction

Earned Value Management (EVM) is a quantitative performance management tool that is in use for more than 50 years [1]. It objectively measures the project progress and performance in terms of scope, cost and schedule and estimates the future of the project. It basically compares the planned and actual work and calculates the value of the accomplished work. In spite of its wide spread use and success in many industries such as mining and construction, it is largely underutilized in the software industry. We believe it is related with an inherent property of software projects [2]

Software projects face a significant factor of change that is not frequently encountered in traditional fields of engineering [3–5]. In traditional fields such as manufacturing and construction, once the problem is defined, it can be assumed to be stable and change is neither very common nor physically possible. If a task is

I. Stamelos et al. (Eds.): SPICE 2018, CCIS 918, pp. 200–214, 2018.
https://doi.org/10.1007/978-3-030-00623-5_14

completed once, it is assumed that there will be no frequent changes on that but in software projects it is a common practice.

The earlier approach to deal with the problem in the software industry was to avoid changes by making better analysis and better plans. The objective was to stabilize products and processes but it did not produce desired effects [6] and has been never enough to prevent or avoid change [3, 4]. The studies show that software specialists spend about 30% to 50% of their time on rework [5, 7, 8]. In software development projects, a completed task might be redone after sometime during the project execution because of various reasons like defects and improvements.

Once it was clear that change cannot be avoided, various solutions have been proposed for its management. With the emergence of agile methods change is accepted as inevitable and the modern approach become embracing and managing the change instead of avoiding or preventing it. Today, agile approaches are used by a large amount of software organizations in the world [9, 10] but they do not have the variety of the tools that exists for traditional project management [10]. We believe traditional project management methods, tools and techniques need to be adapted or replaced by more effective ones considering the change factor to provide best of both worlds.

EVM is based on the traditional project management approach. It assumes that the plan established at the beginning will be stable during the execution of the project. The initial planning and baselines are very important for EVM, since it fundamentally describes how much the project align with the initial plan. EVM does not offer any special treatment for later changes and rework.

The main drawback of EVM for software projects is the influence of late effort spent for change of completed tasks [2]. The effort and cost spent later do not increase the value earned and the same is value for the change. The accomplished scope is still considered to be the same but costs more. As a result, EVM depicts an incorrect picture to project managers about the progress and the future of the project. The effort spent for change including unpredictable changes, requirement and design changes, software bugs, improvements, technical debts is ignored. If there were no changes and we did it absolutely right in every aspect for the first time, we would not have such a discussion and we would have had the same EV in every calculation.

The change oriented extension, cEVM, has been proposed [11] in order to overcome the change related drawbacks of EVM by introducing measures related with change. The model brings change aspect into the traditional EVM based on any kind of rework and evolution costs and incorporates them into scope, cost and schedule aspects to enable better visualization of software projects progress.

In this study, we conducted two case studies in two different companies to explore the usability of cEVM compared to the traditional EVM and an iterative project. We have explored if it helps to manage software development projects better comparing to traditional EVM. The first project is rather suitable to target of EVM since it applies traditional project management approach and waterfall life cycle model. The second project utilize an iterative development approach. We selected these projects to observe if/how cEVM could be used or extended for iterative and agile projects. We obtained promising results in both cases that we were able to calculate the progress as well as future estimates more precisely using cEVM. We have also faced some challenges that require further studies.

The paper is structured as follows. Section 2 summarizes the background on EVM. Section 3 presents cEVM with new measures briefly. Section 4 presents the application studies, with the background of the projects and results of the application and discusses the findings. Section 5 draws conclusion.

2 Background

This section presents the relevant background on EVM including the history and overview of the method.

The Earned Value concept in its most fundamental form has been used in industrial manufacturing in the early American factories since the late 1800s [1]. EVM formally introduced as a project management tool by the US Navy as part of the PERT/Cost methodology in 1962. Later in 1967, the US Department of Defense (DoD) formally issued Cost/Schedule Control Systems Criteria, which incorporates the EV concept with thirty-five criteria and mandated their use on systems developed for DoD. The private industry did not utilize EVM in the industrial projects except governmental contracts due to the complexity till the mid of the 90s.

After simplification of EVM in 1997, it has been evolved and was formally issued as a standard by American National Standards Institute (ANSI) [12]. The usage has been spread out to the other governmental agencies and private industry and the other nations e.g. Australia, Canada, and Sweden [13]. Project Management Institute (PMI) involved an overview of EVM in the first version of the PMBOK and broadened in subsequent versions [14, 15]. In 2007, PMI published a separate guideline "Practice Standard of Earned Value Management" to empower its role [16].

Furthermore, Agile EVM has been proposed as a light-weight adaption of EVM for agile project management [17–19]. Agile EVM does not aim to replace current agile metrics. Instead, it is just an additional one to existing others to increase the visibility of the project status and to support decision making. It particularly utilizes the terminology defined in Scrum and contains a simplified set of EV calculations adapted from traditional EVM.

EVM is all about planning, which results a Performance Measurement Baseline (PMB) and then controlling/measuring progress and performance according to this plan/baseline [20]. It has three key data elements:

- **Planned Value (PV)** is the sum of all the budgets for all planned work at any given time in the project schedule, corresponds to established PMB. The performance is measured against PV, typically plotted cost versus time with S-shaped curve.
- **Earned Value (EV)** is the value of the work progress at a given point in time, expressed in terms of PV.
- **Actual Cost (AC)** is the summation of the resources spent in accomplishing all work performed for the time period.

The key data elements are the basics of EVM (see Fig. 1). The other EVM metrics involving variances, indices, forecasts are originated from these elements (see Table 1).

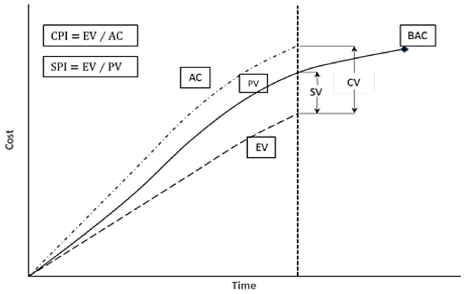

Fig. 1. EVM measures

Table 1. EVM metrics

Metric	Equation	Description
Schedule Variance (SV)	EV – PV	The difference between the planned value of the work scheduled and the value of the work accomplished
Schedule Performance Index (SPI)	EV – AC	Index showing the efficiency of the time utilized
Cost Variance (CV)	EV – AC	The difference between the values of the work accomplished and the actual cost incurred
Cost Performance Index (CPI)	EV/AC	Index showing the efficiency of the utilization of the resources allocated to the project
Budget at Completion (BAC)	Total PV	Cost of total estimated work in the plan
Estimate to Complete (ETC)	(BAC – EV)/CPI	Estimated cost required to finish all the remaining work
Estimate at Completion (EAC)	AC + ((BAC – EV)/CPI), AC + ETC, BAC/CPI	Projected final cost required to finish complete work and based on a statistical prediction using the performance indexes
Variance at Completion (VAC)	BAC – EAC	The difference between what the project was originally baselined to cost, versus what it is now estimated to cost

The variances reflect the project current status comparing the planned and actual data elements. The indices depict how efficiently cost and schedule used and also show the trends of the progress. They also enable predicting the future of the project based on the fundamental principle that trends and patterns in the past determine the future.

3 cEVM Overview

cEVM provides integrating the change aspect into traditional EVM by defining change related concepts, measures and metrics [11]. Change results in reworking and/or evolution costs and the distinguishing feature of cEVM is to incorporate rework and evolution costs into actual costs and to calibrate EV based on this total cost and its trends in time.

cEVM defines the following key measures in addition to PV, EV and AC: Reworking and Evolution Cost (REC), Total Cost (TC), Cost Factor (cf), First-time Completion Efficiency (ftce), Expected Reworking and Evolution Cost (RECexp), and Estimated EV (EVest) (see Table 2). Plus, cEVM improves the performance metrics of traditional EVM (see Table 3).

Table 2. cEVM measures

Metric	Equation	Description
Reworking and Evolution Cost (REC)	EV − PV	Rework and evolution cost occurring after once a task completed, including bugs, defects, improvements
Total Cost (TC)	REC + AC	Total cost of project summing up rework and evolution cost with actual cost
Change Factor (cf)	REC/ACt − 1	Index showing the change ratio of the task, phase or project
First-time Completion Efficiency (ftce)	1 − cf	Index showing what percentage of the task, phase or project done right at first time
Expected Reworking and Evolution Cost (RECexp)	AC * cf	Total expected rework and evolution cost for the completed tasks according to the change factor
Estimated Earned Value (EVest)	EV * TC/ (AC + RECexp)	Calibrated EV according to the change trends

4 cEVM Applications

Two applications of cEVM have been conducted to explore the usability of cEVM and to compare the benefits regarding the traditional EVM.

Two projects applying different software development approaches have been selected. The first project is the software development part of a large-scale integration project with waterfall approach and the second one is a maintenance project with iterative approach using Scrum.

We used written documents, which are mainly the project plans, progress reports, sprint backlogs, error reports, and semi-structured interview methods to collect data. Two interviews with project managers have been performed to get the brief project info, to gather data and to clarify the issues.

Table 3. cEVM performance metrics

Metric	Equation
Estimated Schedule Variance (SVest)	EVest – PV
Estimated Schedule Performance Index (SPIest)	EVest/PV
Estimated Cost Variance (CVest)	EVest – TC
Estimated Cost Performance Index (CPIest)	EVest/TC
Estimated Variance at Completion (VACest)	BAC – EACest
Estimate To Complete - Estimated (ETCest)	(BAC – EVest)/CPIest
Estimate at Completion - Estimated (EACest)	TC + ((BAC – EVest)/CPIest), TC + ETCest, BAC/CPIest
Estimated Total Reworking and Evolution Cost (ETREC)	cf * (BAC – EVest) + REC
Estimate To Complete Reworking and Evolution Cost (TCRECest)	cf *(BAC – EVest)

These two case applications initially performed around 2014–2015 in the scope of the PhD thesis of the author [21], which mainly focus on quality dimension for EVM. Afterwards, this study has been revised for cEVM in 2018.

4.1 Application I

The organization is the Turkish subsidiary of a global company serving consultancy and systems integration services on various business sectors including financial services, health, public sector, retail, telecommunications and transportation. It employs nearly seven hundred engineers and holds ISO 9001:2008 and ISO/IEC 27001 certifications.

The project is the development of command and control system that integrates emergency management solution with fifteen applications. The project started in March 2011 and planned completion date was September 2011. It was completed with two months delay. It is a sub-project of a large scaled integration project. We focus on the software development project since our aim is to apply cEVM on software projects.

The project follows waterfall development methodology which is tailored according to their project needs. The development phases, including detailed analysis, design and testing activities, follow initial analysis and design phase. At the end, deployment & training phase located.

The project team includes 15 full-time software developers and a project manager. There are no specific analyst, developer or test engineer roles in the team, all engineers are doing all the tasks depending on the needs.

Java technologies together with Oracle Fusion and TCL/TK scripting have been used during development. MSSQL was used for database management system. The requirements and test cases were stored on MS Excel. MS Project and MS Excel were project management tools while Bugzilla was for managing the errors and changes.

Applying EVM and cEVM

The case study first conducted in January 2015 and revised in March 2018. At the beginning, we contacted the project manager via e-mail. Then we explained the study and discuss the needs in a semi-structured interview. Afterwards, he delivered us the project data in MS Project and Excel sheets.

The documents include the released project plan and realization data of the plan as well as the error reports exported from Bugzilla to an excel sheet. All the necessary data for the EVM and cEVM application were gathered from these documents. We resolve the conflicts and get more project details by means of second semi-structure interview in a face-to-face meeting and resolve the problems.

We applied EVM and cEVM every four weeks, so the month is used as the time unit during applications. The effort, in person-hour, is used as cost unit.

First, EVM application has been conducted based on the project plan that the project manager provided. Table 4 shows EVM application results and Fig. 2 presents the EVM graph.

Table 4. Case I EVM application results

	Month	PV	AC	EV	SPI	SV	CPI	CV	EAC	VAC	ETC
1	Mar	2034	2340	1836	0,90	−198	0,78	−504	16369	−3526	14029
2	Apr	4284	5490	3726	0,87	−558	0,68	−1764	18923	−6080	13433
3	May	6174	7740	5436	0,88	−738	0,70	−2304	18286	−5443	10546
4	Jun	7974	10053	6984	0,88	−990	0,69	−3069	18487	−5644	8434
5	July	10134	12618	8829	0,87	−1305	0,70	−3789	18355	−5512	5737
6	Aug	12474	15210	10719	0,86	−1755	0,70	−4491	18224	−5381	3014
7	Sep	12843	15705	11088	0,86	−1755	0,71	−4617	18191	−5348	2486
8	Oct		17523	12843	1,00	0	0,73	−4680	17523	−4680	0

cEVM application has been started with collecting RECs from the error reports. We calculated TC and cf, ftce, RECexp and EVest sequentially based on that (see Table 5 and Fig. 3). Afterwards, the performance metrics have been calculated (see Table 6).

Results

Initial EVM application shows that the project has a cost overrun and a delay in the schedule from the beginning to the end of the project as both seen in EVM graph and performance metrics.

cEVM firstly underlines reworking and evolution costs. At the beginning of the project, there is rather low REC, 45 person-hours in the second period of cEVM after the analysis and design phase. It increases considerably starting from the third phase and keeps its rising trend around 4000 person-hours through the end of September, when the project is planned to be completed. The project has total 5535 person-hours costs spent for reworking and evolution when it is completed with almost two months delay.

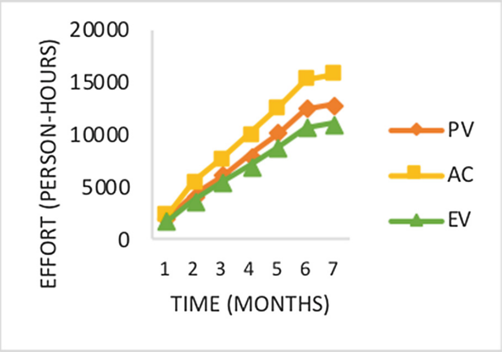

Fig. 2. Case I EVM graph

Table 5. Case I cEVM application results

	PV	AC	EV	REC	TC	cf	ftce	RECexp	EVest
1	2034	2340	1836	0	2340	0,00	1,00	0	1836
2	4284	5490	3726	45	5535	0,02	0,98	106	3686
3	6174	7740	5436	684	8424	0,12	0,88	964	5261
4	7974	10053	6984	1242	11295	0,16	0,84	1613	6762
5	10134	12618	8829	2196	14814	0,22	0,78	2756	8507
6	12474	15210	10719	3249	18459	0,26	0,74	3916	10345
7	12843	15705	11088	4077	19782	0,27	0,73	4210	11014
8		17523	12843	4698	22221	0,30	0,70	5242	12536
9				5535	23058	0,32	0,68		

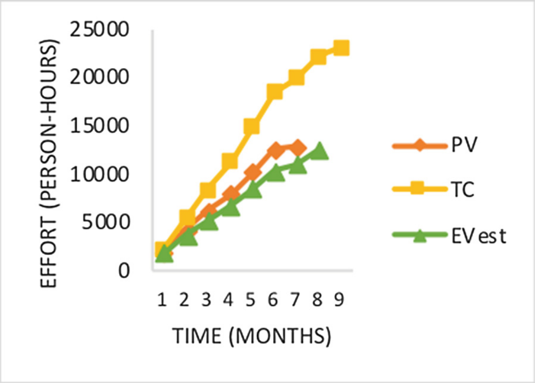

Fig. 3. Case I cEVM graph

Table 6. Case I cEVM performance metrics

	SPIest	SVest	CPIest	CVest	EACest	VACest	ETCest
1	0,90	−198	0,78	−504	16369	−3526	14029
2	0,86	−598	0,67	−1849	19287	−6444	13752
3	0,85	−913	0,62	−3163	20565	−7722	12141
4	0,85	−1212	0,60	−4533	21453	−8610	10158
5	0,84	−1627	0,57	−6307	22364	−9521	7550
6	0,83	−2129	0,56	−8114	22916	−10073	4457
7	0,86	−1829	0,56	−8768	23067	−10224	3285
8	0,98	−307	0,56	−9685	22765	−9922	544

RECs constitute the significant part of the total costs. cEVM presents the actual cost more accurately. TC of cEVM shows that the final cost is almost double of the planned one, 12843 vs 23058 person-hours.

Change factor, cf, highlights that there is an increasing change in the project, almost the percentage of 30%. Hence, ftce indicates that around 70% of the tasks completed right first time in the project.

RECexp, changing between 106 to 5242, gives a clue about how much total REC cost is expected for the completed part based on the past change trends. EVest of cEVM presents the current earned value more accurately considering expected RECs of implemented features.

By means of EVM, the cost overrun starts with 500 person-hour and increases later till 4680 person-hour with CPI changing between 0,78 to 0,69. This low-cost performance index gives an alarm about the cost problem of the project. Subsequently, the value of EAC is calculated around more than 18000 person-hours, which costs more than 5000 person-hours than planned value.

cEVM results spot serious cost problems considering TC and EVest and so expect more cost overrun than EVM. CPI is changing between and CV is 0.78 to 0.54. The cost variance exceeds 9000 person-hours. Based on the improved CPI, cEVM estimates completion budget more accurately. The project manager expects the final budget, EAC, around 22000 person hours during the project execution and variance nearby 10000 person hours.

According to EVM application, the delay in the schedule is also considerable starting from the second phase. SPI is changing between 0.90 to 0.86. The tasks are not implemented on time and the project is behind the schedule with the percentage of 86%. cEVM has relatively better SPI, from 0.90 to 0.83, and SV values, from −200 to −2000, than EVM.

Even though EVM reflects the latency and cost overrun in the project, cEVM presents significantly more accurate numbers and much better future estimates by revealing significant but hidden reworking and evolution costs.

4.2 Application II

The organization is a software development company, developing various e-government projects for a specific government organization. It employs approximately 60 software engineers and holds ISO/IEC 20000 and ISO/IEC 27001 certifications.

The project is the maintenance of a web-based procurement tool that provides managing complex tenders, bids and contracts for a large amount of audiences. The maintenance project was started in February 2013 and planned completion date was November 2013. It has been completed at the beginning of January 2014.

The project follows iterative development approach with Scrum practices. The team includes 7 staff, which are part-time project manager, scrum master also working as software engineer, two software engineers, a senior test engineer and a part-time quality manager. The project team was not fully dedicated to this project, a team member might have also some other responsibilities in another project in some sprints.

The project was developed using .NET Framework. The new features, changes, improvements and errors were stored in Microsoft Team Foundation Server (TFS). MS Project was utilized for project management and MS Word and MS Excel were used for documentation of requirements and testing.

Applying EVM and cEVM

The study initially conducted in May 2014 and revised in April 2018. Firstly, we sent the case study statement to the project manager via e-mail and then conducted an initial meeting with the project manager and quality manager to discuss the details of the study. Afterwards, we scheduled an additional meeting as a semi-structured interview with the quality manager and obtain the brief overview of the project as well as explain the needs of a case study in detail. Finally, the quality manager provided the project data, basically sprint plans, resource utilization reports, error reports and exported all to the excel sheets. We arranged another meeting to discuss the inconsistencies.

The document that they provide includes the bugs fixed and new features from the feature list as distributed into the monthly releases. We applied EVM and cEVM monthly according to these releases. The effort, in person-hour, was used as cost unit.

Table 7. Case II EVM application results

	Month	PV	AC	EV	SPI	SV	CPI	CV	EAC	VAC	ETC
1	Feb	101	96	86	0,85	−15	0,90	10	4646	−484	4550
2	Mar	750	831	752	1,00	2	0,90	79	4599	−437	3768
3	Apr	1619	1712	1560	0,96	−59	0,91	152	4568	−406	2856
4	May	2207	2392	2185	0,99	−22	0,91	207	4556	−394	2164
5	Jun	2558	2814	2537	0,99	−21	0,90	277	4616	−454	1802
6	Jul	2970	3179	2878	0,97	−92	0,91	301	4597	−435	1418
7	Aug	3203	3480	3160	0,99	−43	0,91	320	4583	−421	1103
8	Sep	3460	3716	3378	0,98	−82	0,91	338	4578	−416	862
9	Oct	3703	4036	3667	0,99	−36	0,91	369	4581	−419	545
10	Nov	4162	4553	4151	1,00	−11	0,91	402	4565	−403	12

We initially applied EVM on the project considering the high-level project plan including planned features and their realizations. Table 7 shows the EVM application results and Fig. 4 is the graphical representation of EVM.

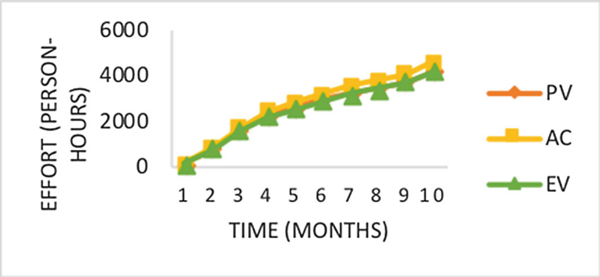

Fig. 4. Case II EVM graph

We applied cEVM just after EVM implementation. The application has been started with the collecting RECs. Next, the cf calculation is accomplished considering TC. Accordingly, RECexp, EVest, ftce are calculated. The application results are given in Table 8. Figure 5 shows the graphical representation of cEVM application. Additionally, Table 9 presents the performance analysis according to cEVM.

Table 8. Case II cEVM application results

	PV	AC	EV	REC	TC	cf	ftce	RECexp	EVest
1	101	96	86	0	96	0,00	1,00	0	86
2	750	831	752	25	856	0,26	0,74	216	615
3	1619	1712	1560	93	1805	0,11	0,89	192	1479
4	2207	2392	2185	341	2733	0,20	0,80	476	2082
5	2558	2814	2537	418	3232	0,17	0,83	492	2480
6	2970	3179	2878	491	3670	0,17	0,83	555	2829
7	3203	3480	3160	550	4030	0,17	0,83	602	3120
8	3460	3716	3378	579	4295	0,17	0,83	618	3347
9	3703	4036	3667	639	4675	0,17	0,83	694	3624
10	4162	4553	4151	758	5311	0,19	0,81	855	4076
				955	5508	0,21	0,79		

Results

The initial EVM application results show that the project has a cost overrun but almost on time.

cEVM makes reworking and evolution costs visible and adds them to the ACs as in the first case study. Initially, it starts with 25 person-hours then it increases till 955 person-hours at the end. The project is completed almost one month delay. cEVM represents all the costs more accurately including hidden RECs, which is around 5508 person-hours.

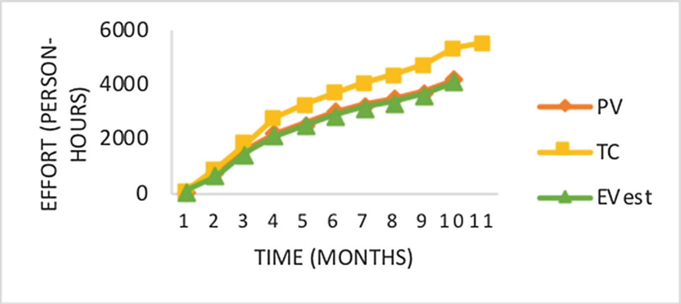

Fig. 5. Case II cEVM graph

Table 9. Case II cEVM performance metrics

	SPIest	SVest	CPIest	CVest	EACest	VACest	ETCest
1	0,85	−15	0,90	10	4646	−484	4550
2	0,82	−135	0,72	241	5797	−1635	4941
3	0,91	−140	0,82	326	5079	−917	3274
4	0,94	−125	0,76	651	5464	−1302	2731
5	0,97	−78	0,77	752	5423	−1261	2191
6	0,95	−141	0,77	841	5399	−1237	1729
7	0,97	−83	0,77	910	5376	−1214	1346
8	0,97	−113	0,78	948	5340	−1178	1045
9	0,98	−79	0,78	1051	5369	−1207	694
10	0,98	−86	0,77	1235	5422	−1260	111

Change factor shows that almost 20% of the tasks are under change. The ftce index, the ratio of the tasks completed right first time, is decreasing around 0,80.

RECexp, changing between 216 to 855 person-hours, shows how much cost is expected for reworking and evolution based on the trends. Using this value, cEVM presents adjusted EVest values, which are much better indication of the gained value comparing to EV.

Based on EVM calculations, the cost overrun starts with 10 person-hours and increases till 400 person-hours with the CPI index over 0,90. CPI already gives an alarm about minor cost problem. The metrics show that EAC will be around 4600 person-hours while variance at completion is nearby 1200 person-hours.

The CPI of cEVM is around 0,77, rather lower than EVM's CPI. cEVM presents more severe cost variance exceeding 1200 person-hours and so provides us to estimate more cost overrun than EVM. cEVM calculates that the project will be completed around 5400 person-hours, which is almost 800 person-hours more than EVM's estimate and much better estimation for 5508 person-hours total cost.

According to EVM application, the schedule seems on track with SPI index between 0.97 to 1 after the initial phase. Based on that, the tasks were mostly implemented on time and the deadlines almost met without any delay. The delay is relatively seen better by means of cEVM, with SPI changing between 0,82 to 0,98.

In summary, during execution of the project, the project manager understands the cost problems more clearly and expects more cost overruns utilizing cEVM. It also gives much better future estimates and allows the project manager to re-plan the activities or budget or scope based on this fact.

4.3 Discussion

We obtained promising results in both cases in regard to cEVM. Change is a critical challenge for software projects applying different life cycle models. In waterfall project the change factor measured around 30% while in iterative one it was around 20%. It is not surprising that waterfall project has more change than iterative one since we plan the project in detail at the beginning in waterfall but iterative project has continuous refinement of plan during execution of project.

Even though there is a huge amount of rework and evolution effort in software projects, these efforts frequently are not well managed. They are mostly perceived as troubles that need to be fixed immediately and fixed by the team members who supposed to perform other tasks. In both cases there were no change related task planned. The change 20%–30% has been realized in an unplanned way. This frequently results in low morale and burnout due to the endless evening and weekend overtime of the development team which in turn causes poor quality that results in more rework later on.

Highlighting these change efforts by making change costs visible by means of cf and REC will help project managers to expose the project's status clearly. The results of both case studies show that the software project's current status is more clearly depicted and project's future is more accurately estimated by cEVM in comparison to EVM.

The case studies show that incorporating change costs into actual costs, cEVM provides much better evaluation of the budget. CPI values are significantly better comparing to the ones that are calculated with traditional EVM. Schedule evaluations are also better due to better EV estimation but are not dramatically improved by cEVM. Additionally, the case study results spot that cEVM provides much better future estimates. Completion budgets of both cases are estimated fairly better by cEVM than EVM.

As observed in both applications, the forecasts of estimated completion budget, EAC, are very close to the actual ones by cEVM.

We also detected that cf as defined in cEVM by itself is a simple and an effective indicator of change. The project managers could track cf trends during project and make root cause analysis when required.

The most significant difficulty we encountered is the availability and the validity of the change related effort data. The effort data is not collected and tracked properly in many software organizations. Organizations to implement cEVM should make issue tracking more systematic and keep effort data. It might bring additional cost of change data collecting to organizations.

5 Conclusion

This study presents the results of applying change oriented EVM on two software projects. The results show that cEVM provides clearer progress information and more accurate future estimates in comparison to traditional EVM.

We explored the usability of cEVM and based on the case study results, we can summarize the main benefits of cEVM as follows:

- reveals hidden change related costs and integrating them into project and performance management
- measures the change status of a project in addition to schedule and cost
- estimates the project progress more precisely comparing to traditional EVM
- estimates project future more accurately in comparison to traditional EVM

cEVM helps to increase project visibility by means of revealing hidden but huge rework and evolution cost. Increased visibility brings more accuracy to the projects. It mainly affects the calculations of the total costs and the earned value and so the project is measured more accurately. Besides, cEVM provides better predictability, which is vital for a project manager to take action as soon as possible. More accurate progress metrics result in more accurate and realistic future estimates that increases the predictability. cEVM provides more accurate actual cost, more precise performance metrics and more accurate future estimates.

We plan to perform further case studies to better comprehend the applicability of cEVM for the different type of projects (e.g. micro services, embedded systems).

Briefly, EVM is a powerful technique to reflect project progress in terms of scope, time and cost. cEVM makes it more usable for software projects incorporating change. cEVM provides significant improvements with change both on measuring the progress clearly and on estimating the future correctly. Change is inevitable for software projects and cEVM is designed to embrace it.

References

1. Anbari, F.: Earned value project management method and extensions. Proj. Manag. J. **34**(4), 12–13 (2003)
2. Efe, P., Demirors, O.: Applying EVM in a software company: benefits and difficulties. In: Proceedings of the 39th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA), Santander, Spain (2013)
3. Cass, A.G., Sutton, S.M., Osterweil, L.J.: Formalizing rework in software processes. In: Oquendo, F. (ed.) EWSPT 2003. LNCS, vol. 2786, pp. 16–31. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-45189-1_3
4. Twentyman, J.: The crippling costs of IT project rework. Inside Knowledge, 15 June 2005
5. Charette, R.N.: Why software fails. IEEE Spectr. **42**(9), 42–49 (2005)
6. Uskarci, A., Demirors, O.: Do staged maturity models result in organization-wide continuous process improvement? Insight from employees. Comput. Stand. Interfaces **52**, 25–40 (2017)
7. Ebert, C., Dumke, R.: Software Measurement: Establish - Extract -Evaluate - Execute. Springer, Berlin (2010). https://doi.org/10.1007/978-3-540-71649-5

8. Laporte, C.Y., Berrhouma, N., Doucet, M., Palza-Vargas, E.: Measuring the cost of software quality of a large software project at bombardier transportation. Softw. Qual. Prof. J. **14**(3), 14–31 (2012)
9. Coskuncay, A., Demirors, O.: Software development in Turkey. IT Prof. **17**(3), 10–13 (2015)
10. VersionOne: 12th Annual State of Agile (2018). https://explore.versionone.com/state-of-agile/versionone-12th-annual-state-of-agile-report
11. Efe, P., Demirors, O.: A change oriented model for earned value management and its application in software development projects. IEEE Trans. Softw. Eng. in review
12. ANSI/EIA -748A: American National Standard Institute/Electronic Industries Alliance/Standard for Earned Value Management Systems (1998)
13. AS 4817-2006: Project performance measurement using Earned Value (2006)
14. Project Management Institute: A Guide to the Project Management Body of Knowledge (PMBOK® Guide), 1st edn. Newtown Square (2000)
15. Project Management Institute: A Guide to the Project Management Body of Knowledge (PMBOK® Guide), 6th edn. Newtown Square (2017)
16. Project Management Institute: Practice Standard for Earned Value Management. Project Management Institute, Newtown Square (2005)
17. Cockburn, A.: Crystal Clear: A Human-Powered Methodology for Small Teams. Pearson Education Inc., Upper Saddle River (2005)
18. Sulaiman, T., Barton, B., Blackburn, T.: AgileEVM – earned value management in scrum projects. In: Proceedings of AGILE 2006 Conference, Minnesota, USA (2006)
19. An Industry Practice Guide for Agile on Earned Value Management Programs, version 1.2, National Defense Industrial Association, 26 March 2018
20. Project Management Institute: Practice Standard for Earned Value Management. Newtown Square (2011)
21. Efe, P.: Quality Integrated Earned Value Management. Ph. D. dissertation, Middle East Technical University (2015). etd.lib.metu.edu.tr/upload/12619220/index.pdf