Dr. Thomas Liedtke
Alcatel
D 70435 Stuttgart
(Germany)

e_mail: TLiedtke@alcatel.de
phone: +49 711 821 40346
fax.: +49 711 821 42230

Peter Paetzold
Alcatel
D 70435 Stuttgart
(Germany)

e_mail: Peter.Paetzold@alcatel.de
phone: +49 711 821 49183
fax.: +49 711 821 422 30

Three Numbers to Measure Project Performance

accepted paper for ASM 2001,
the International Conference on Applications of Software Measurement

Abstract:

We present a method which produces at any time during the execution of a big software development project a reliable prediction of the total duration and of the total cost to expect at project completion. Based on this, appropriate steering measures can be taken: from recognition of success over recovery actions up to project cancellation.

Having a powerful **Project Performance Measuring** in place, potential occurrence of schedule overrun and cost overrun in software projects can be managed and controlled. Although particular cost tracking and several progress metrics may be applied during the execution of projects, reliable knowledge of the total duration and the total cost tends to be not available.

The basic idea presented in our paper is to correlate cumulative cost consumed to current completion reached, and to learn out of this about the future of the project. Prerequisites are a cost consumption plan and a deliverables completion plan. The approach is presented both theoretically and on hand of a real life case. Special attention is paid to project management techniques related to the method.

0 Table of Contents

1 The Reality

A potential risk of Software development projects is to produce schedule overrun and cost overrun. Schedule overrun tends to occur in form of surprises: close to milestones to reach, it turns out that they can not be met. Cost overrun tends to occur after the facts: after completion of only a part of the project, it turns out that the total project budget has been consumed. Although cost tracking and progress metrics are applied during the execution of projects, reliable knowledge of the total duration and the total cost may not be available. That is why projects which are found in the middle of execution with schedule overrun or cost overrun tend to be continued until completion, just because their final schedule overrun and cost overrun are not known at the moment when the decision to stop should be taken.

2 The Ideal

In an ideal world, at each point in time of project execution, a reliable prediction is available, what the total project duration will be and what the total project cost consumption will be.

3 The Promise

We present a method which produces at any time during the execution of a project a reliable prediction of the total project duration and of the total project cost to expect at project completion. This method is called **Project Performance Prediction**, **(P3)**.

The input needed for **P3** is nothing extraordinary: project budget with planned consumption over time, project schedule with intermediate milestones, and information about the current project status reached at the moment when **P3** is applied. The good idea with **P3** is to correlate cumulative cost consumed to current completion reached, and to learn out of this about the future of the project.

P3 can be applied to a project as a whole, how big or small the project may ever be, or recursively to subprojects out of which a project is composed.

P3 can be introduced at any point in time between project start and project completion. Obviously, the earlier it is introduced, the earlier its output can be used for project steering. The later in the project **P3** is applied, the more reliable and the less interesting are its results: closer to project completion, less unexpected things can happen. **P3** can be applied arbitrarily often during project execution.

P3 is presented both theoretically and on hand of a real life case.

The theory behind **P3** has an easy and a more difficult part. The easy part is pure arithmetic: the **Earned Value Analysis**. The more difficult part is related to project management: how to define precisely the Degree of Completion reached in a big software development project.

4 The Method In Theory Part I: Pure Arithmetic

4.1 Earned Value, How to Define It, And What to Learn Out of It

The key concept of all we talk about is the Earned Value. Roughly speaking, the Earned Value is the budgeted cost of the work actually performed, or the value of intermediate work products actually produced. The Earned Value is independent of the actual cost of work performed.

The basic idea of **P3** is to take at a point in time during project execution the Earned Value produced so far, the cost planned to consume so far, and the cost actually consumed so far, and to calculate out of these three the total duration and the total cost to expect at project completion. During the calculation, a few intermediate results (indices) are produced.

One basic assumption of **P3** is: the total project cost, i.e. the project budget, is fixed and is revised only exceptionally. We will show later how to quantitatively define when such an exceptional case occurs.

To enable easy understanding and to make some definitions clear, we will give calculated numbers for a *example* project after each definition. The requested project task of the *example* is like following:

10 documents have to be written within 10 weeks. 10 persons are planned to do the job. The completion of each document will cost 10 person weeks (pweeks).

4.2 Three Numbers to Measure Project Performance

Due to the fact, that the Earned Value Method discriminates budget, cost, and schedule, three main parameters have to be tracked separately:

1. **Budgeted cost of work scheduled (*BCWS*):** the sum of approved cost estimates for activities scheduled to be performed during a given period.
2. **Budgeted cost of work performed (*BCWP*):** the sum of approved cost estimates for activities completed during a given period (often called 'Earned Value').
3. **Actual cost of work performed (*ACWP*):** the sum of actual cost spent to accomplish work during a given time period.

After 5 weeks the follwing picture for our example project appears:

- *BCWS = 50 pweeks: it was planned to have all 10 contributors for entire 10 weeks allocated. Therefore after 5 weeks, 10 persons à5 weeks were scheduled.*
- *BCWP = 40 pweeks. 4 documents are ready after 5 weeks: each document completed has the value of 10 pweeks. The earned value is therefore 40 pweeks.*
- *ACWP = 45 pweeks. In average 9 people were reporting on this project for the first 5 weeks on this project. Therefore the sum of actual cost spent is 45 pweeks.*

4.3 Prediction for Total Cost

Following cost and budget numbers are defined:

1. **Budgeted cost at completion (*BAC*):** At the beginning of the project *BAC* once has to be provided: the planned total cost of the project.
2. **Estimated cost to complete (*ETC*):** *ETC = (BAC - BCWP)x ACWP/ BCWP*: In a first step the remaining effort which has to be spent till completion is the difference between budgeted cost at completion and the earned value. The multiplication factor is a cost index by which the remaining effort is multiplied, assuming that the same factor as observed up to now will be achieved for the remaining activities.

3. **Estimated cost at completion (*EAC*):** *EAC = ACWP + ETC*. At the end of the project the actual cost spent so far and the estimated cost to complete will be spent. At the end of the project *EAC* will be *ACWP* (*ETC = 0* at the end).
4. **At completion cost variance (*ACV*):** *ACV = BAC - EAC*. This number is an estimate of the variance of the original estimated total cost of the project and the estimated cost at completion.

For our example following numbers can be calculated:

- *BAC = 100 pweeks (10 documents à10 pweeks).*
- *ETC = (100 – 40)x 45 / 40 = 67,5 pweeks which have to be spent for the remaining project. The assumption is, that same efficiency (45 pweeks for 4 documents = 11,3 pweeks per document) will be consumed for the 6 remaining documents.*
- *EAC = 45 + 67,5 = 112,5 pweeks is reflecting the estimated cost at completion. 45 pweeks were already spent, further 67,5 pweeks have to be spent to complete the remaining 6 documents.*
- *ACV = 100 – 112,5 = -12,5 pweeks cost variance (in this example this means an overrun) are expected if project is running continuously with same indices.*

4.4 Degree of Completion

The Degree of Completion reached at any point in time during the execution of a project is essential to define the Earned Value produced. In big software development projects, it appears not trivial how to precisely define the Degree of Completion reached. Imagine you are project leader of a $ 10 million software development project, and you shall say each week again at which percentage your project is completed.

We present two alternative methods for this: One based on **countable work products**, the other based on **re-estimated cost to complete**.

Assuming Degree of Completion (*DOC*) would be known, *ETC* can be calculated out of *ACWP* and *DOC*:

1. **Estimated cost at completion (*EAC*):** $\frac{ACWP}{DOC}$. The formula simply extrapolates actual cost of work performed for 100% completion degree.
2. **Estimated cost to complete (*ETC*)**:
3. **Degree of Completion (*DOC*):** if (re-) ETC and ACWP is known:

According to countable work products our example project are ready by 40%:

- *DOC = 40%.*
- *EAC = ACWP/DOC = 45/0.4 = 112.5 pweeks.*
- *ETC = 45*(1-0.4)/0.4=67.5 pweeks.*

Due to the fact, that the same cost index for 6 remaining documents will appear, DOC calculated based on ETC and ACWP by formula above is again 40%.

4.5 Variances Expressed In Monetary Units

1. **Schedule Variance (*SV*):** *SV = BCWP – BCWS* is reflecting if schedule in terms of work product completion compared with planned forecast is in line. Numbers greater than 0 are indicating that completion degree is ahead of schedule.
2. **Cost Variance (*CV*):** *CV = BCWP – ACWP* is reflecting if all completed workproducts are accomplished within their costs estimated. Number lower than 0 are reflecting, that the actual cost of work performed is higher as the budgeted cost of work performed -> project is getting more expensive.
3. **Budget Variance (*BV*):** *BV = BCWS – ACWP* is showing if planned resources are on board of project. A number less than 0 is indicating an overrun, more people as planned were reporting on project.

For all 3 numbers it's preferrable to get positive values.

For our example project following numbers are calculatable:

- *SV = 40 – 50 = -10 pweeks. This means, that the example project is 10 pweeks behind schedule.*
- *CV = 40 – 45 = -5 pweeks are indicating, that project is 5 pweeks more expensive than expected, 5 pweeks were spent in addition to achieve completion reached.*
- *BV = 50 – 45 = 5 pweeks are reflecting, that 5 pweeks were not spent.*

4.6 Performance Indicators

Instead of variances leading to absolute monetary units, performance indicators can be defined:

1. **Schedule Performance Indicator (*SPI*):** is giving an indication how budgeted cost of work performed is related to budgeted cost of work scheduled. Numbers greater than 1 are indicating that the project is ahead of schedule.
2. **Cost Performance Indicator (*CPI*):** is putting into relation budgeted cost of work performed and actual cost of work performed. A quotient greater than 1 indicates a cost overrun.

3. **Budget Usage Indicator (*BUI*):** is relating actual cost of work performed and budgeted cost of work scheduled. Numbers greater than 1 are observable if cost is spent faster than scheduled

Please observe, that *EAC* is independent of *CPI* (s. definitions in section 4.4).

For our example project we can conclude as follows:

- *SPI = 40/50 = 80% schedule performance (lower 1 => project is behind schedule).*
- *CPI = 45/40 = 113% cost performance index, that means, that the completion was more expensive as planned by 13%.*
- *BUI = 45/50 = 90% budget usage which leads to the conclusion, that only 90% of the manpower planned was really working for the project.*

Bottom line after 5 weeks: considering all numbers calculated, the project status can be summarized as follows:

Although in total an underrun is observable (5 pweeks), the project tends to become more expensive than expected by 13%. The schedule variance of -10 pweeks (4 i.s.o. 5 documents completed) is caused by 2 reasons:

- *Only 90% of planned resources were available (= 5 pweeks missing).*
- *The effort for documentation completion was underestimated. To complete the first 4 documents additional 5 pweeks had to be spent.*

In best case the project can be completed within 105 pweeks (ACWP+ETC= 45+60) if cost index will come back to 100%. That means an overrun of 5% compared with original planning. In worst case the project will be completed with an overrun of 13%.

The schedule can be kept only if more resources (people, Saturday work, overtime, ...) will be put on the project and/or contents of project can be decreased. For example the creation of one document will be skipped.

4.7 Necessity For Re-Budgeting

In normal circumstances, the budgeted cost at completion shall not be revised. If however *EAC* and *BAC* deviate too much from each other, exceptionally re-definition of the project budget may be needed.

An example for the quantitative definition of such abnormal circumstances is:

"if for at least four weeks in sequence, and if the latest re-definition of the project budget was done at least two months ago, then the project budget has to be revised."

Similar rules can be defined for rescheduling.

5 The Method In Theory Part II: Project Management

The administrative overhead to apply the Earned Value Method with high granularity will grow with the size of a project. Therefore it must be ensured, that the goals and targets are defined carefully upfront, before forcing people to report numbers which are not usable in a reasonable manner. We present typical subjects which should be discussed before introducing the Earned Value Method to a project.

5.1 Consideration of Individual Project Parameters

If a deviation from project baseline occurs, potentially available overall project numbers will neither be sufficient to explain observed deviation, nor to trigger needed corrective actions. A split of the project into several 'subprojects' has to be defined. Following project parameters (hopefully known before) have to be considered in order to define subprojects most reasonably:

- **Size** and **criticality** of subprojects: It makes simply no sense to track uncritical or "cheap" tasks with high granularity. In addition the number of 20 subprojects should not be exceeded from managerial point of view. Criteria to group a subset of tasks defining one subproject should consider its size and criticality for the whole project.
- **Measurable completeness**: due to the fact, that for every subproject effort, budget, and completeness of related workproducts has to be tracked, it has to be ensured that all three numbers can be monitored during life-time of the subproject. An exception could be accepted for tasks if *ECT* can be re-estimated regularly.
- **Existing reporting lines of stakeholders:** Contribution of several organizational areas to one single subtask has to be considered accordingly. It could be, that subprojects have to be splitted once more in order to reflect separate contribution.
- Complete sum of **administrative tasks** has to be considered. It has to be decided case by case, on which subprojects the administrative tasks have to be counted. Example: project team could become own subproject, department leaders can be counted on related subprojects representing a phase (e.g. System Design, Top Level Design, ...)
- **Project phases:** To reflect already well-known tasks from existing master plans and to make an easy understanding of subprojects, for each phase of the project a separate subproject should be defined, like for instance: Top Level Design, Integration, ...
- **External contribution** to project. In order to evaluate external contributions, own subprojects should be defined.

5.2 Ensuring Fast Feedback

People working on the project will be recommended strongly to report as accurately as possible what they are doing. **P3** is introduced in order to be able to trigger needed corrective actions as soon as deviations occur. Therefore, a fast feedback of reported figures has to be installed. If people have to report in a different way compared to the past or if people contribute to more than one project the overhead for the people can become an issue.

The team for analyzing the reported figures has to be installed in front of the project. To introduce a jour fix is recommended. It should be defined as well how the result figures are collected, aggregated and presented.

5.3 How to Define Completeness

For each subproject a completeness figure has to be defined enabling the link per sub-project between budget planned, effort spent and Earned Value of completed workproducts. To make it more visible, different kinds of workproducts have to be monitored separately. Of course for one subproject there can be more than one kind of workproduct. For example Top Level Design may capture creating change requests, writing a number of baseline documents, performing reviews, inspections, workshops etc. For each kind of workproduct the completeness has to be planned in terms of numbers per time frame. Reflecting the necessary effort to complete, the effort for the subproject has to be broken down to each kind of workproduct.

Table 1 is reflecting example definition for phases and workproducts. **Table 2** is splitting Top Level Design into different completeness figures, weighting their earned value be effort splitt.

Name of subproject	Workproducts	Completeness
Top Level Design	Change Requests, Documents, Reviews, Inspections	Percentages of change requests approved; documents released, Reviews inspected, ...
Coding	Coded statements	Percentage of completely coded modules
Integration Test	Test cases, Fault reports	Weighted percentages of Test cases achieved; fault reports solved
Project team	Assignment of project leader and team	Completion Degree in terms of *BAC-ETC*

Table 1: example definition for phases and workproducts

Activity	Planned items	Factor	Week 1	Week 2	Week 3	Week 4	Week 5
Change requests planned	100	0,40	3	14	38	100	100
Change requests com-			2	16	35	98	100

pleted							
Documents planned	20	0,30	1	5	10	20	20
Documents completed			0	6	11	19	20
Reviews planned	100	0,15	0	3	14	38	100
Reviews completed			0	2	16	40	100
Inspections planned	50	0,15	0	5	20	25	50
Inspections completed			0	7	16	30	50
Completion planned cumulative			**2,7%**	**15%**	**38%**	**83%**	**100%**
Completion reached cumulative			**0,8%**	**18%**	**37%**	**81%**	**100%**

Table 2: break down of Top Level Design completeness figure

Different possibilities are existing, if no workproducts in terms of countable items are existing, like for example for a project team. In this case the total effort to contribute for the project has to be estimated upfront the project. During life-time spent effort will be compared against original planned effort. Completeness can be measured in percentage of spent effort or according *DOC* definition based on *ACWP* and *ETC* (s. above).

It has to be avoided, that the completeness is focussing on the end of the (sub-) project. If the completion is defined only by final delivery of workproducts at the end of a project we will see at any time low schedule indices during life time of the project. Nothing completed so far (as planned), but the actual effort increases. During execution of project, we're always more expensive than expected (more effort spent than earned).

6 The Method In Practice

In real life, we use **P3** for a large software development project within the area of telecommunication. We demonstrate how **P3** is applied to this project, what was necessary to introduce it, and what it brings.

6.1 At Project Start

The project has been divided in 17 subprojects, based on the project development lifecycle model we use and on particularities of this project.

The total project budget has been distributed completely to the identified subprojects, based on the project budgeting rules and on experience from former projects.

The budget of each subproject has been stretched out over time, based on the project development lifecycle again and on the milestone plan of this project. This gave the planned cost consumption *BCWS* for each point in time of the project execution.

For each subproject, the nature of countable deliverables produced by the subproject has been identified. In eight cases, we found one category of deliverables per subproject sufficient, e.g. test cases passed, fault reports solved. In seven cases, more than one category of deliverables is needed to track the progress of the subproject, e.g. for Detailed Design: change requests approved, design documents released, and document inspections completed. The maximum number of categories of deliverables identified for one subproject is five. If more than one category of deliverables is produced by one subproject, each category has been assigned a weight with which it contributes to the completion of the subproject. The subproject with the lowest number of deliverables produces 32 items. The subproject with the highest number of deliverables delivers 56,000 items.

For each subproject, the total number of deliverables has been defined, and the completion of deliverables over time has been planned. This gave a planned degree of completion for each point in time of the project execution. Although this kind of information is not needed for **P3**, we found it worthwhile to have this.

For two out of the 17 subprojects, it was found that they do not produce anything countable. One of the two is the project management team. We decided to apply progress tracking based on regularly re-estimated cost to complete to these two subprojects.

A simple spreadsheet was set up to calculate the degree of completion of the project in total as weighted mean value of each subproject's degree of completion, with each subproject's share of the total project budget as weights.

6.2 During Project Execution

Since the start of the project, each week the following data is collected:

- **Number of countable deliverables** completed last week in each of the subprojects.
- **Cost consumed** last week by each of the subprojects.

For the subprojects without countable deliverables, the cost to complete is re-estimated every six to ten weeks, based on a revised forecast of who is expected to contribute how much between now and the completion of the subproject.

Out of these inputs, each week the following data is calculated and interpreted, for each subproject and for the project as a whole:

- The **Degree of Completion** reached and its relation to the degree of completion planned.
- The **cumulative cost consumed** so far *ACWP*, the **Earned Value produced** so far *BCWP*, and its relation to the **cumulative cost planned** so far *BCWS.*
- **Cost performance** index, **schedule performance** index, and **budget usage** index.
- **Estimated total cost** at completion *EAC.*

The administrative effort to maintain the **P3** project database has reached meanwhile a stable amount of about two personhours per week. This is possible because we have had in place before both a well developped timesheet system for cost tracking and a comprehensive set of deliverables metrics for progress tracking, upon which **P3** can build. The complete weekly results of **P3** fits on five pages: one with the overall project status, two with curves showing the project history, and two with small-printed detail information about all subprojects.

Table 3: example for overall project status

Table 4: example for project history

Table 5: example for evolution of total effort prediction

6.3 Lessons Learnt So Far

Completeness is crucial: If a project is divided in separately tracked subprojects, **P3** works only if the set of subprojects covers the total project, both budget-wise and progress-wise.

The number of subprojects shall be as small as possible, but big enough to make each subproject so small that it can be understood by one person. Subprojects shall be roughly of the same size.

Subprojects with countable deliverables are easier to track than subprojects with re-estimated cost to complete insofar as subproject progress can be measured objectively and mechanically.

For subprojects with countable deliverables, the number of categories of deliverables shall be as small as possible, preferably one.

Performance indices and predictions of single subprojects are not needed to know status and future of the whole project. They are extremely useful however for project steering and control: you can see immediately, which subprojects deviate from schedule and budget, and focus steering measures on these.

7 The Promise Kept

We have shown how by simple means the schedule performance and cost performance of big software development projects can be predicted from early stages of the project onwards.

The real life example demonstrated here was the first project in our company where **P3** is applied. The expressive results and the easy applicability of **P3** are the reasons why it is applied today (December 2000) to five projects in two different product lines, in three different locations. We expect **P3** to become a standard method to manage big software development projects.

Dr. Thomas Liedtke

- Master's degree in computer science from University of Stuttgart in 1990
- Ph.D. at the computer science department, University of Stuttgart 1993
- 1993 - 1995 member of the quality strategies group at Alcatel SEL AG
- 1995 project quality manager of several large telecommunication projects at Alcatel Telecom
- member and work group leader of several SPI activities
- since 1997 technical project manager of several large telecommunication projects (fixed and mobile) at Alcatel for german market and export markets
- Topics of interest: software testing, software metrics, SW project management.

Peter Paetzold

- Master's degree in mathematics from Humboldt University zu Berlin in 1985
- 1985 – 1990 Institute of Informatics, Academy of Sciences, Berlin
- since 1990 with Alcatel in different positions in Berlin, Paris and Stuttgart
- 1997 – 1999 technical project manager of a big development project
- since 1999 manager operative project controlling