

PR-158

Estimating Cost-To-Go Without Stable EVM Data

Peter C. Frederic, Tecolote Research Inc.
Ronald K. Larson, NASA

20 March 2013

2013 Professional Development & Training Workshop

New Orleans, LA

June 18-21, 2013

The opinions expressed herein are solely those of the authors.

Tecolote Research Inc.
415 E. Ocean Ave., Suite H
Lompoc, CA 93436
(805) 588-1330

Estimating Cost-To-Go Without Stable EVM Data

Peter C. Frederic, Tecolote Research Inc.

Ronald K. Larson, NASA

Abstract

Initial independent estimates for development programs are typically created using parametric methods. As a program progresses, it becomes important to separate sunk cost from cost-to-go. In the very early stages of a program, it is acceptable to estimate cost-to-go by simply subtracting the reported sunk cost from the parametrically-estimated total cost. However, as the program continues to progress and more significant progress accumulates, we can no longer assume that dollars spent equate directly to technical progress. It is then necessary to understand progress to-date in order to estimate cost-to-go. If a detailed, stable, trusted, schedule is available, one could potentially estimate costs to go by independently estimating the cost of each of the incomplete activities remaining in the schedule. However, parametric CERs at the detailed schedule activity level are seldom available and independent estimating teams seldom have access to the detailed information required to do build-up estimating at that level. A good solution to this problem is to use earned-value data to make an independent assessment of technical progress, and then apply this information to the parametrically-estimated total cost to determine cost-to-go. This works well if the earned-value data is stable and trusted to represent a stable baseline. If the earned-value data is not stable, a new approach is required. This paper examines a sample program in which the earned value data was obviously not stable, despite the Critical Design Review having been accomplished and a significant amount of money having been spent. In this case, the earned value data indicated that on the average over several years, management's perception of the total scope of the development effort grew as fast as work was accomplished. Although this growth was described by management as "requirements growth", the technical performance parameters to which the parametric CERs responded did not indicate growth in performance requirements. It was apparent that it was not the performance requirements that had grown but management's understanding of the difficulty of the effort required to meet the requirements. As such, the earned-value data could not be trusted as an indicator of relative technical progress. A new approach was required.

This paper describes an approach to determining progress to-date and cost-to-go based on major milestone completion dates. Specifically, we looked at the percent of total actual dollars spent before the CDR milestones for subsystems in each of several analogous programs. We then used the amount of time before and after CDR for each subsystem in the system of interest to determine completion percentages for each subsystem. This percentage, applied to the parametrically-estimated total costs, provided a more credible independent estimate of cost-to-go.

Table of Contents

1	INTRODUCTION	1
2	THE NOMINAL SITUATION	2
3	THE OFF-NOMINAL SITUATION	4
4	THE SOLUTION	5
5	CONCLUSIONS	8

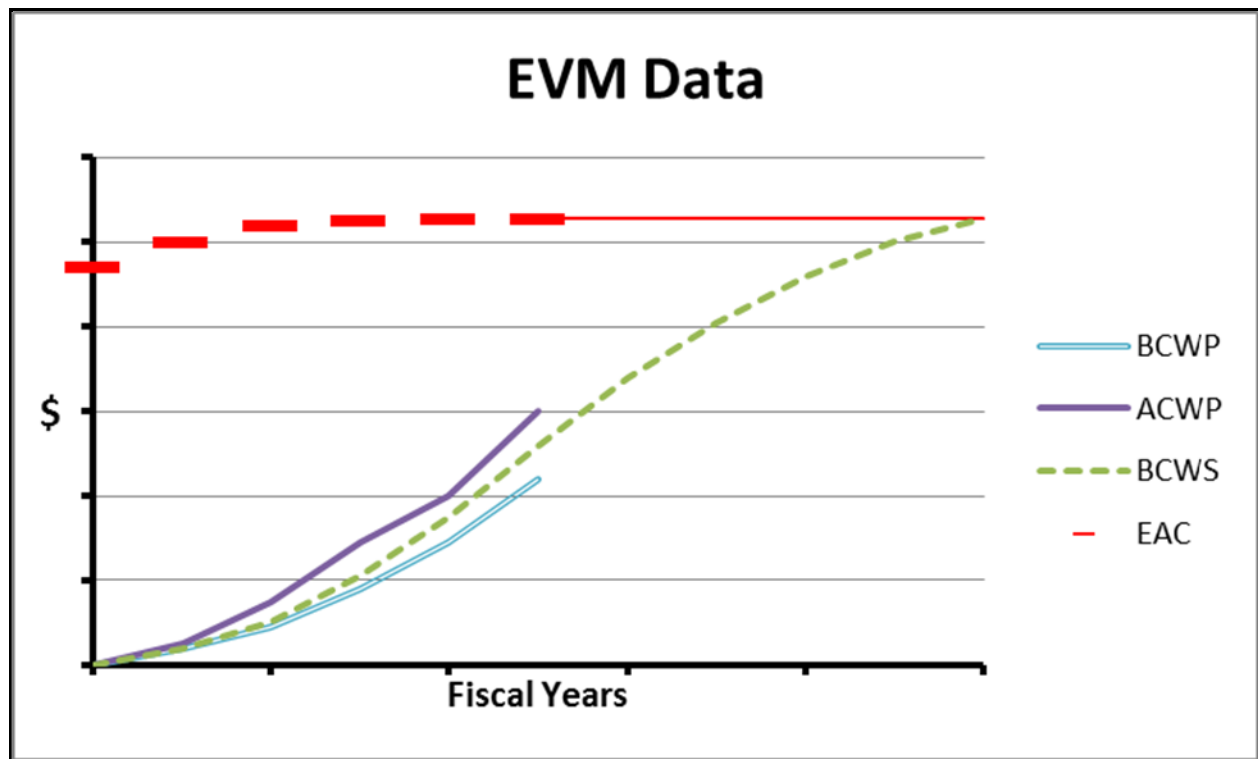
1 INTRODUCTION

Initial independent estimates for development programs are typically created using parametric methods. As a program progresses, it becomes important to separate sunk cost from cost-to-go. In the very early stages of a program, it is acceptable to estimate cost-to-go by simply subtracting the reported sunk cost from the parametrically-estimated total cost. However, as the program continues to progress and more significant progress accumulates, we can no longer assume that dollars spent equate directly to technical progress. It is then necessary to understand progress to-date in order to estimate cost-to-go. If a detailed, stable, trusted, schedule is available, one could potentially estimate costs to go by independently estimating the cost of each of the incomplete activities remaining in the schedule. However, parametric CERs at the detailed schedule activity level are seldom available and independent estimating teams seldom have access to the detailed information required to do build-up estimating at that level. A good solution to this problem is to use Earned-Value Management (EVM) data to make an independent assessment of technical progress, and then apply this information to the parametrically-estimated total cost to determine cost to go. This works well if the EVM data is stable and trusted to represent a stable baseline. If the earned-value data is not stable, a new approach is required. This paper examines a sample program in which the EVM data was obviously not stable, despite the Critical Design Review having been accomplished and a significant amount of money having been spent. In this case, the EVM data indicated that on the average over several years, management's perception of the total scope of the development effort grew as fast as work was accomplished. Although this growth was described by management as "requirements growth", the technical performance parameters to which the parametric CERs responded did not indicate growth in performance requirements. It was apparent that it was not the performance requirements that had grown but management's understanding of the difficulty of the effort required to meet the requirements. As such, the EVM data could not be trusted as an indicator of relative technical progress. A new approach was required.

The sections below describe first (briefly) the normal approach to estimating cost-to-go in a mature program with stable EVM data. Then, a situation is described in which the EVM data is clearly unstable. Finally, an approach is provided for estimating cost-to-go despite the unstable EVM data.

2 THE NOMINAL SITUATION

The figure below illustrates typical EVM data for a mature, stable development program.



In the early years, the program had experienced some cost growth, as evidenced by Estimate At Completion (EAC) increasing slightly in the first three years. However, in the fourth and fifth years the EAC has stabilized indicating that the program baseline is stable and well-understood, which lends credibility to the program plan, represented by the Budgeted Cost of Work Scheduled (BCWS), as well as measures of progress such as the Budgeted Cost of Work Performed (BCWP) and Actual Cost of Work Performed (ACWP). Note that in the case of the program illustrated above, the BCWP has fallen below the BCWS, indicating that less work has been accomplished than planned. Also notice that the ACWP is higher than the BCWP indicating that the work that has been accomplished has cost more than planned.

In a program such as the one illustrated above, EVM data would be viewed as a good indicator of the performing contractor's progress towards completion of the overall effort required. Typically, we would calculate an independent estimate at completion (IEAC) based the actual cost of work performed (ACWP), the budgeted cost of work performed (BCWP or earned value), and the budget at completion (BAC) according to the following formula:

$$\text{IEAC} = \text{ACWP}_{\text{CUM}} + (\text{BAC} - \text{BCWP}_{\text{CUM}}) / \text{CPI} \quad (1)$$

or

$$\text{IEAC} = \text{ACWP}_{\text{CUM}} + (\text{BAC} - \text{BCWP}_{\text{CUM}}) / (\text{CPI} * \text{SPI}) \quad (2)$$

Where

IEAC	=	Independent Estimate At Completion
ACWP _{CUM}	=	Actual Cost of Work Performed (Cumulative)
BAC	=	Budget At Completion
BCWP _{CUM}	=	Budgeted Cost of Work Performed (Cumulative)
CPI	=	BCWP _p / ACWP _p = Cost Performance Index over some period of time “p”
SPI	=	BCWP _p / BCWS _p = Schedule Performance Index over some period of time “p”
BCWS	=	Budgeted Cost of Work Scheduled

This formula (1) estimates project cost by applying cost growth, represented by the Cost Performance Index (CPI) to the remaining budgeted cost-to-go. In the second form of the equation (2), the Schedule Performance Index is introduced in the denominator, presumably to represent further cost growth due to schedule slippage. For a high-fidelity IEAC, these equations should be applied at the lowest WBS level for which EVM data is available.

For a *more* independent EAC, the following approach can be used:

$$\text{IEAC} = \text{ACWP}_{\text{CUM}} + (1 - \text{BCWP}_{\text{CUM}} / \text{BAC}) * \text{TC}_I \quad (3)$$

Where

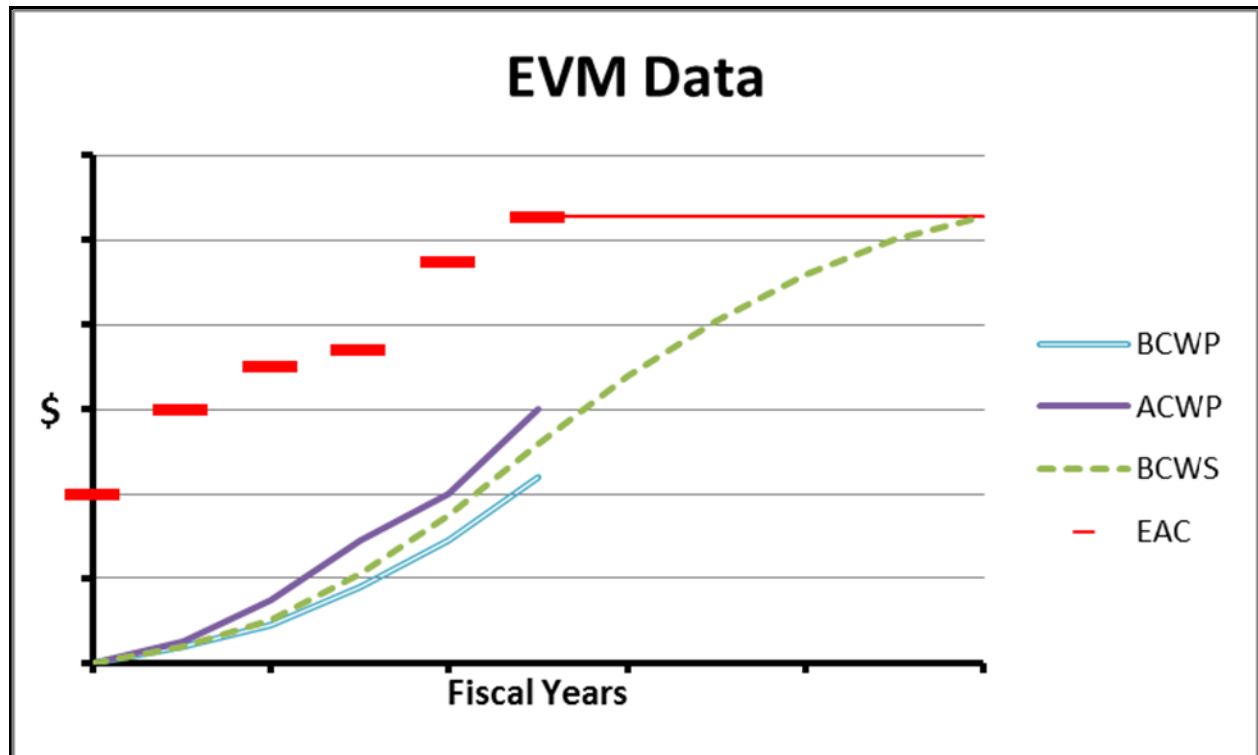
IEAC	=	more Independent Estimate At Completion
ACWP _{CUM}	=	Actual Cost of Work Performed (Cumulative)
BAC	=	Budget At Completion
BCWP _{CUM}	=	Budgeted Cost of Work Performed (Cumulative)
TC _I	=	Independently-estimated Total Cost

In this case, a totally independent method (a parametric CER, for example) is used to estimate the total cost of the WBS item, assuming industry-average cost and schedule performance. The EVM data is used to determine the sunk cost (ACWP_{CUM}) and the value of work to date in terms of completion percentage (BCWP_{CUM} / BAC).

However, this type of estimate still depends on a certain degree of trust in the EVM system, and is only useful if the baseline work content, as measured by the BAC is stable.

3 THE OFF-NOMINAL SITUATION

The figure below illustrates a case in which the EVM baseline for a program has clearly not stabilized, despite several years of technical progress.



In this case, management's understanding of the scope of the program has grown continuously over the course of the program. In fact, the program baseline has grown as fast as work has been accomplished. Most disturbingly, there has been no leveling off in the baseline growth trend. There is nothing in this data to indicate that the people behind the earned value data have any idea how much the program will ultimately cost.

There are several possible causes for this situation:

Performance requirements growth – Performance requirements growth is usually a result of external stakeholders increasing the performance requirements on the system. For example, military planners might decide that an aircraft needs to carry a heavier payload than originally required. These requirement changes should be reflected in changes in cost-driving performance parameters or design parameters to which parametric CERs respond.

Design requirements growth – Design requirements growth occurs when designers discover that the preliminary designs on which the initial costs estimates are based are inadequate to meet the performance requirements. This might be caused by performance requirements growth,

optimistic technology assumptions, or simply inadequate preliminary design practices. The result is growth in mass, software lines of code, or other design parameters. These requirement changes should be reflected in changes in cost-driving design parameters to which parametric CERs respond.

Execution requirements growth – This refers to unforeseen additional tasks required to complete the development program. For example, more system testing than planned might be required to prove that the system meets performance requirements. These requirements often represent a level of detail not captured in typical cost-driving parameters for parametric CERs.

Really bad early cost estimating – This could be due to incomplete or inaccurate technical and programmatic information provided to the program cost estimators, political pressure to make the program appear affordable, or simply poor estimating practices.

In cases such as the one illustrated above, regardless of the cause of the program's cost growth, independent methods must be used to estimate total cost. Furthermore, even the progress-to-date metrics from the EVM system can't be trusted without a stable baseline, so an independent method must be used to determine the value of progress to date. Fortunately, even when the EVM data is completely unstable, ACWP is an auditable historical fact and can still be used reliably to establish sunk cost.

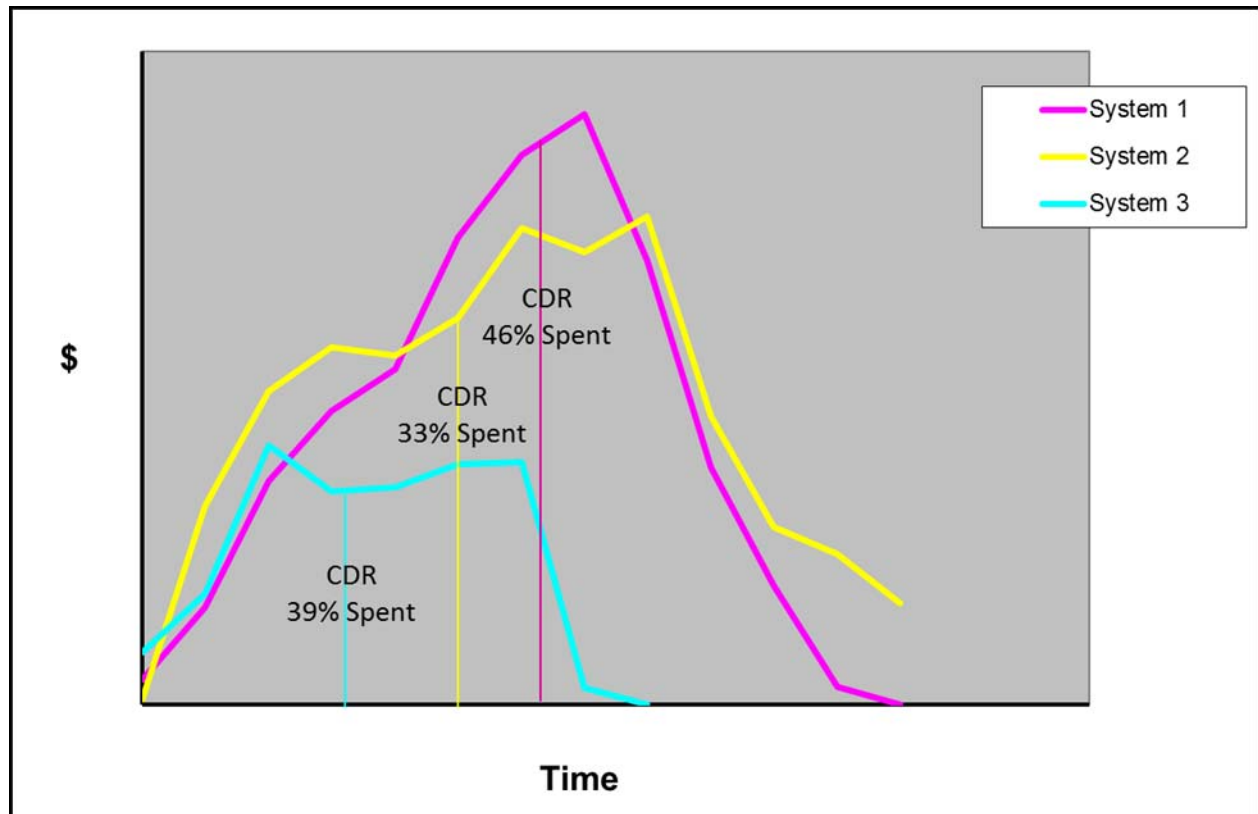
4 THE SOLUTION

The solution to the problem of estimating cost-to-go can be found by looking at historical cost and schedule information for analogous programs. By definition, the purpose of schedule milestones is to provide a means of measuring progress. The program management community has developed consistent definitions of key development program milestones. In particular, the Critical Design Review (CDR) is defined as the point in a development program when system or subsystem design is finalized, and fabrication of full-scale hardware can proceed. If definitions of key milestones such as CDR dates are consistent with historical projects, then key milestones can be assumed to represent a consistent portion of overall work. Estimating cost-to-go in the absence of reliable EVM data is a three-step process:

1. Look at historical data for similar projects and determine the typical portion of work (in dollar terms) represented by the key milestones.
2. Look at the schedule for the current project and calculate completion factors based on when current milestones were or will be accomplished.
3. Apply *incompletion* factors ($1.0 - \text{completion factor}$) to independently estimated totals to estimate cost-to-go.

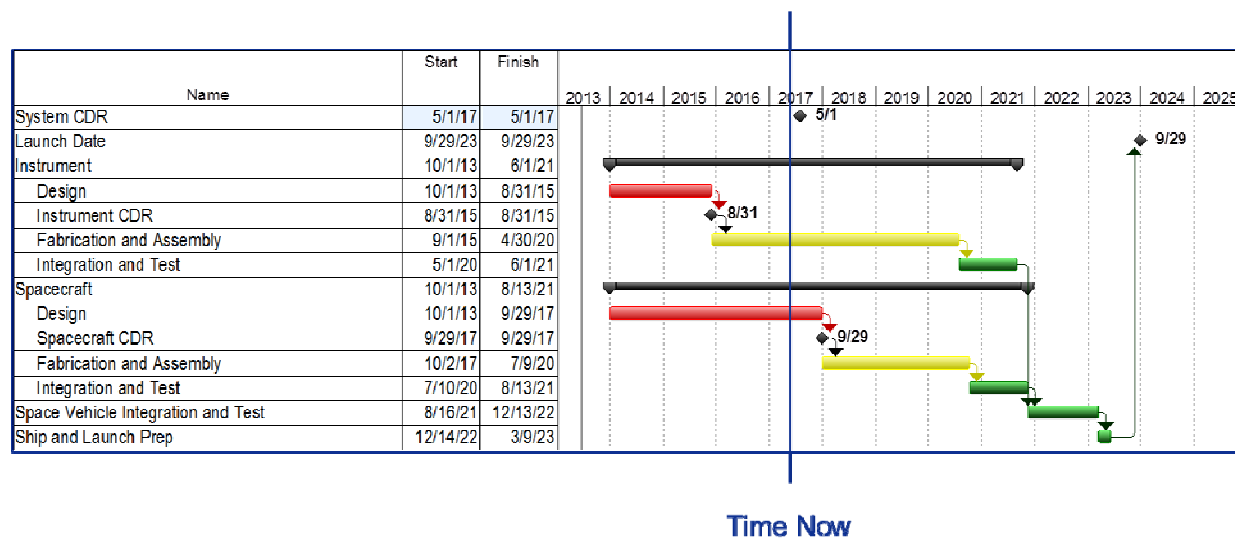
This process should be carried out at the lowest level of detail possible. Consider a much-simplified space vehicle development program. If the EVM data was clearly unreliable, the first

step to estimating cost to go would be to look at actual, time-phased expenditures and actual milestone completion dates for similar programs, as shown below.



In this case, cumulative expenditures have been calculated as of the system CDR date for each of three historical programs. The average percent spent at CDR for the three programs is 39.3%. Assuming that true cost performance remained roughly constant over the course of each program (they didn't fire all the bad workers half way through and hire new, much better workers), one can assert that the actual percent spent is equivalent to the actual percentage of effort accomplished. In this case, we assert that for similar programs, CDR completion represents **39.3%** of required effort accomplished. Ideally, this analysis would be performed at the lowest level of WBS detail possible, possibly for multiple milestones (PDR, CDR, etc.). To keep this example simple, it is shown here at the program CDR level only.

Having determined milestone effort equivalence factors, the next step is to apply those factors to the current program. Consider the sample schedule and reported and estimated costs shown below:



Sample Schedule

WBS	CDR Date	Spent Before CDR	ACWP	EAC	TCi
Instrument	8/31/2015	155.0	178.0	203.0	237.0
Spacecraft	9/29/2017	122.0	112.0	170.0	184.0
Space Vehicle Integration and Test	N/A	N/A	10.0	65.0	70.0
Launch	N/A	N/A	0.0	90.0	100.0
Total			300.0	528.0	591.0

Sample Cost Data

In the “Sample Cost Data” figure above, TCi refers to the independently-estimated total cost for each WBS item. The amount “Spent Before CDR” can be an actual amount if the CDR for the WBS item has been accomplished or a projected amount if the CDR has yet to be accomplished. Completion factors would be calculated as shown in box “d” below.

WBS	CDR Date	Spent Before CDR	Percent Complete Based on CPR Date	ACWP	TCi	Independent Cost To-Go	Value of Work Performed
Instrument	8/31/2015	155.0	45.1%	178.0	237.0	130.0	107.0
Spacecraft	9/29/2017	122.0	36.1%	112.0	184.0	117.6	66.4
Space Vehicle Integration and Test	N/A	N/A	N/A	10.0	70.0	60.0	10.0
Launch	N/A	N/A	N/A	0.0	100.0	100.0	0.0
Total		c	a/c	a	b	b * (100% - d)	b * d
			d = 39.3% * a / c	300.0	591.0	407.7	183.3

The “a/c” term in box “d” represents the effort completed relative to CDR, so in the “Spacecraft” row, $a/c = 112.0 / 122.0 = 91.8\%$, so the effort to complete CDR for the Spacecraft is 91.8% complete. Since the historical effort represented by CDR is 39.3% of total effort, then the overall Percent Complete for the Spacecraft is $91.8\% * 39.3\% = 36.1\%$.

Finally, the independent cost-to-go can be determined by multiplying the inverse of Percent Complete times the TCi: $(100\% - 36.1\%) * 184.0 = 117.6$

5 CONCLUSIONS

It is possible to make a sound estimates of costs to-go without stable EVM data, using the method describe herein.

Having higher fidelity historical schedule data allows for higher fidelity estimating of progress based on milestone dates. Having historical schedule data with multiple milestones for numerous programs at the subsystem level or lower would allow accurate completion percentage profiles to be developed and employed. This would provide a useful cross-check, even when stable EVM data is available. **It is important for the cost, schedule, and EVM communities to collect and maintain high-fidelity, historical schedule information.**