



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

**A PROBABILISTIC APPROACH TO JOB-SHOP
MANUFACTURING ESTIMATION AND
PERFORMANCE FORECASTING**

by

Eric V. Kline

September 2023

Thesis Advisor:
Second Reader:

Timothy P. Anderson
Douglas L. Van Bossuyt

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC, 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 2023	3. REPORT TYPE AND DATES COVERED Master's thesis	
4. TITLE AND SUBTITLE A PROBABILISTIC APPROACH TO JOB-SHOP MANUFACTURING ESTIMATION AND PERFORMANCE FORECASTING			5. FUNDING NUMBERS	
6. AUTHOR(S) Eric V. Kline				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) In defense job-shop manufacturing, cost and schedule estimates along with performance to those estimates have a heavy impact on overall project budget and schedule for pre-milestone B activities. Variations at this stage of the project tend to directly influence the availability of the system under development, which results in propagating delays through the rest of the Technology Maturation & Risk Reduction phase. Project stakeholders have a need to understand impacts from uncertainty along with anticipated variances to cost and schedule continuously throughout the manufacturing process. Extending earned value management techniques with probabilistic modeling may be able to provide meaningful insight into manufacturing estimates and performance without requiring project stakeholders to have detailed familiarity with the manufacturing process. This thesis explores if probabilistic estimation methods are sufficient to predict actual performance and compare existing methods of project estimation and performance reporting with combined probabilistic forecasting. This work demonstrates that additional insight can be attained with probabilistic forecasting that is useful to program managers during job-shop manufacturing project execution and can serve as an additional leading indicator of cost and schedule variance.				
14. SUBJECT TERMS probabilistic cost, probabilistic schedule, probabilistic estimation, probabilistic performance, earned value management, critical path, manufacturing estimation			15. NUMBER OF PAGES 167	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

**A PROBABILISTIC APPROACH TO JOB-SHOP MANUFACTURING
ESTIMATION AND PERFORMANCE FORECASTING**

Eric V. Kline
Civilian, Department of the Navy
BSIE, Lehigh University, 2009

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING MANAGEMENT

from the

**NAVAL POSTGRADUATE SCHOOL
September 2023**

Approved by: Timothy P. Anderson
Advisor

Douglas L. Van Bossuyt
Second Reader

Oleg A. Yakimenko
Chair, Department of Systems Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

In defense job-shop manufacturing, cost and schedule estimates along with performance to those estimates have a heavy impact on overall project budget and schedule for pre-milestone B activities. Variations at this stage of the project tend to directly influence the availability of the system under development, which results in propagating delays through the rest of the Technology Maturation & Risk Reduction phase. Project stakeholders have a need to understand impacts from uncertainty along with anticipated variances to cost and schedule continuously throughout the manufacturing process. Extending earned value management techniques with probabilistic modeling may be able to provide meaningful insight into manufacturing estimates and performance without requiring project stakeholders to have detailed familiarity with the manufacturing process. This thesis explores if probabilistic estimation methods are sufficient to predict actual performance and compare existing methods of project estimation and performance reporting with combined probabilistic forecasting. This work demonstrates that additional insight can be attained with probabilistic forecasting that is useful to program managers during job-shop manufacturing project execution and can serve as an additional leading indicator of cost and schedule variance.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	MOTIVATION	2
B.	RESEARCH QUESTIONS.....	3
C.	BENEFITS OF THIS STUDY	3
II.	RELATED WORKS.....	5
A.	GAO COST ESTIMATING AND ASSESSMENT GUIDE	5
	1. Initiation and Research Stage.....	7
	2. Assessment Stage.....	8
	3. Analysis Stage.....	11
	4. Presentation Stage.....	14
B.	DOD COST ESTIMATING GUIDE.....	16
	1. Policy	17
	2. Program Definition	17
	3. Cost Estimate Basis.....	17
	4. Select Cost/Schedule Estimating Methods.....	17
	5. Build Cost Estimate Model	19
	6. Review the Initial Results.....	20
	7. Final Results and Documentation.....	20
C.	JOINT AGENCY COST SCHEDULE RISK AND UNCERTAINTY HANDBOOK.....	23
D.	EARNED VALUE MANAGEMENT	25
E.	PLANNING AND FORECASTING USING INTEGRATED S- CURVES AND STOCHASTIC S-CURVES	26
F.	CALCULATING TOTAL PROBABILISTIC ESTIMATES	28
III.	RESEARCH METHODS.....	31
IV.	BOTTOM-UP ESTIMATION AND FORECASTING FRAMEWORK	33
A.	DEVELOPMENT OF THE WORK BREAKDOWN STRUCTURE.....	33
B.	POPULATION OF INDIVIDUAL WBS ELEMENTS WITH OPERATIONS	34
C.	FORMULATION OF THE ACTIVITY NETWORK DIAGRAM.....	35
D.	CALCULATION OF THE CRITICAL PATH AND FLOAT	36

1.	Enumeration of Paths	37
2.	Calculation of CPM Properties for Each Node.....	38
3.	Calculating Early, Late, and Mid Hours for CPM.....	41
E.	REVISING CPM ESTIMATES WITH TIME-SERIES ACTUAL COSTS.....	45
V.	USE OF EARNED VALUE MANAGEMENT	49
A.	DEVELOPMENT OF EARNED VALUE MANAGEMENT CHART	49
B.	CALCULATING EVM VARIANCES AND EFFICIENCIES	51
VI.	DISCRETE PROBABILISTIC ESTIMATION FOR MANUFACTURING OPERATIONS	55
A.	IDENTIFYING PLANNED VARIANCES IN ACTIVITIES	55
B.	ADJUSTING SME INPUTS TO ACCOUNT FOR UNCERTAINTY.....	56
C.	UTILIZING PROBABILISTIC ESTIMATES TO CALCULATE EXPECTED OUTCOME.....	58
D.	CRITICAL PATH OF CUMULATIVE SCHEDULE WITH PLANNED VARIANCES	59
VII.	COMPARISON BETWEEN EARNED VALUE MANAGEMENT AND PROBABILISTIC MONTE CARLO SIMULATION.....	67
A.	PMD PROJECT AT T=11/04/2022 (PROJECT START)	67
B.	PMD PROJECT AT T=12/01/2022	68
C.	PMD PROJECT AT T=01/03/2023	70
D.	PMD PROJECT AT T=02/01/2023	71
E.	PMD PROJECT AT T=03/01/2023	73
F.	PMD PROJECT AT T=03/31/2023	75
G.	PMD PROJECT AT T=05/24/2023 (PROJECT END)	76
H.	COST AND SCHEDULE CONFIDENCE INTERVALS OVER EXECUTION	77
VIII.	CONCLUSION AND RECOMMENDATIONS.....	81
	APPENDIX. VISUAL BASIC FOR APPLICATIONS (VBA) CODE	83
A.	FORMS	83
1.	MainFrm.....	83
B.	MODULES	92

1.	LoadModule.....	92
2.	MainModule	104
3.	ReportModule	105
4.	TrialModule.....	124
C.	CLASS MODULES	127
1.	JO_Class	127
2.	JO_Estimation.....	131
3.	JO_Eval.....	135
4.	JO_Eval_Bin.....	136
5.	JOT_Class.....	138
6.	TO_Class.....	138
7.	WBS_Class.....	140
	LIST OF REFERENCES.....	141
	INITIAL DISTRIBUTION LIST	143

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

Figure 1.	The Cost Estimating Process. Source: Persons (2020, fig. 5).....	6
Figure 2.	A Cumulative Distribution Function, or S-Curve. Source: Persons (2020, fig. 14).	13
Figure 3.	The DOD Cost Estimating Process. Source: CAPE (2020, fig. 1).	16
Figure 4.	Estimating Method Applicability. Source: CAPE (2020, fig. 2).	18
Figure 5.	Total Ownership Cost Composition. Source: CAPE (2020, fig. 5).	19
Figure 6.	Sand Chart (Stacked Bar) (Notional). Source: CAPE (2020, fig. 8).	21
Figure 7.	Pareto Chart (Notional). Source: CAPE (2020, fig. 9).	22
Figure 8.	S-Curve Example (Notional). Source: CAPE (2020, fig. 14).	23
Figure 9.	Cost of Work Report (left). Source: DD (1963, fig. 15). Earned Value Management “Gold Card” (right). Source: DAU (2020).	25
Figure 10.	Probabilistic Monitoring Using Integrated S-Curves. Source: Barraza and Back (2000, fig. 4).	27
Figure 11.	Probabilistic Performance Forecasting Using SS-Curves. Source: Barraza, Back, and Mata (2004, fig. 2).	28
Figure 12.	Framework for Predicting the Change in the Critical Path. Source: Hammad et al. (2020, fig. 1).	30
Figure 13.	WBS for Example Project.....	34
Figure 14.	Example Sequence of Operations within a WBS	34
Figure 15.	Activity Network Diagram for Example Project	36
Figure 16.	WBS Rollup Activity Network Diagram for Example Project.....	36
Figure 17.	CPM Early, Late, and Mid Hours over Time for Example Project	43
Figure 18.	CPM Early, Late, and Mid Hours over Time for Example Project with Overall Schedule Float.....	44
Figure 19.	CPM Early, Late and, Mid Hours over Time for Example Project with Overall Schedule Shortfall.....	45

Figure 20.	CPM Early, Late, and Mid Hours for Example Project with Revised Expectations Based on Actual Costs at T=08/06/2021	47
Figure 21.	CPM Early, Late, and Mid Hours for Example Project with Revised Expectations Based on Actual Costs at T=09/13/2021	48
Figure 22.	Earned Value Management Chart for Example Project at T=09/13/2021	50
Figure 23.	Cost and Schedule Variances (CV% & SV%) over Time for Example Project at T=09/13/2021	52
Figure 24.	Cost and Schedule Efficiencies (CPI & SPI) over Time for Example Project at T=09/13/2021	53
Figure 25.	Probability Density Function of the Triangular Distribution.....	55
Figure 26.	Probability Density Function of the Triangular Distribution with 50% and 75% Intervals	56
Figure 27.	Amended Probability Distribution Function of Triangular Distribution Accounting for Uncertainty	58
Figure 28.	Histogram of Baseline Schedule and Cost Outcomes from Monte Carlo Simulation	62
Figure 29.	Cumulative Distribution Function (CDF) of Baseline Schedule and Cost Outcomes from Monte Carlo Simulation	63
Figure 30.	Two-Dimensional Histogram of Cost and Schedule from Monte Carlo Simulation	64
Figure 31.	Three-Dimensional Histogram of Cost and Schedule from Monte Carlo Simulation	64
Figure 32.	Three-Dimensional CDF of Cost and Schedule from Monte Carlo Simulation	65
Figure 33.	Two-Dimensional Histogram of Cost and Schedule from Monte Carlo Simulation where Remaining Activities are on the Critical Path	66
Figure 34.	EVM Charts of PMD Project at T=11/04/2022	67
Figure 35.	CPM and Trial Results of PMD Project at T=11/04/2022.....	68
Figure 36.	EVM Charts of PMD Project at T=12/01/2022	69

Figure 37.	CPM and Trial Results of PMD Project at T=12/01/2022.....	70
Figure 38.	EVM Charts of PMD Project at T=01/03/2023	70
Figure 39.	CPM and Trial Results of PMD Project at T=01/03/2023.....	71
Figure 40.	EVM Charts of PMD Project at T=02/01/2023	72
Figure 41.	CPM and Trial Results of PMD Project at T=02/01/2023.....	73
Figure 42.	EVM Charts of PMD Project at T=03/01/2023	74
Figure 43.	CPM and Trial Results of PMD Project at T=03/01/2023.....	74
Figure 44.	EVM Charts of PMD Project at T=03/31/2023	75
Figure 45.	CPM and Trial Results of PMD Project at T=03/31/2023.....	76
Figure 46.	EVM Charts of PMD Project at T=05/24/2023	76
Figure 47.	CPM of PMD Project at T=05/24/2023	77
Figure 48.	Cost Expectation and Trial Confidence over PMD Project Execution.....	78
Figure 49.	Schedule Expectation and Trial Confidence over PMD Project Execution	79
Figure 50.	MainFrm Layout	83
Figure 51.	MainFrm after Run Results.....	84

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	Recommended Uncertainty Distributions. Source: NCCA (2014, tbl. 2–2).....	24
Table 2.	Comparison of Terms Used by PERT Cost (1963) and EVM Gold Card (2020)	26
Table 3.	Paths Resulting from Example Project	37
Table 4.	Critical Path Method (CPM) Properties.....	39
Table 5.	CPM Calculations for Example Project.....	40
Table 6.	Expected Value and Variance for Distributions. Adapted from Hayter (2013).....	59
Table 7.	Trial Durations for Various Distributions.....	61

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ACWP	Actual Cost of Work Performed
ALRE	Aircraft Launch and Recovery Equipment
BCWP	Budgeted Cost of Work Performed
BCWS	Budgeted Cost of Work Scheduled
CAPE	Cost Assessment and Program Evaluation
CARD	Cost Analysis Requirements Description
CDF	Cumulative Distribution Function
CPI	Cost Efficiency
CPM	Critical Path Method
CPMC	Critical Path / Monte Carlo
CV	Cost Variance
DOD	Department of Defense
EAC	Estimate at Completion
EF	Early Finish
EH	Early Hours
ES	Early Start
Est	Estimated Time
ET	End Time
EVM	Earned Value Management
GAO	U.S. Government Accountability Office
ICE	Independent Cost Estimate
JA CSRUH	Joint Agency Cost Schedule Risk and Uncertainty Handbook
LF	Late Finish
LH	Late Hours
LS	Late Start
MH	Mid Hours
MSA	Material Solution Analysis

NAWCAD	Naval Air Warfare Center – Aircraft Division
OSD	Office of Secretary of Defense
PDF	Probability Density Function
PMD	Prototype and Manufacturing Division
QA	Quality Assurance
SE	Support Equipment
SME	Subject Matter Expert
SPI	Schedule Efficiency
SS-Curves	Stochastic S-Curves
ST	Start Time
SV	Schedule Variance
TDP	Technical Data Package
TE	Time Entries
TR	Time Remaining
TW	Time Worked
USC	United States Code
VBA	Visual Basic for Applications
WBS	Work Breakdown Structure

EXECUTIVE SUMMARY

Well informed cost and schedule estimation and performance forecasting is an essential activity when conducting job-shop manufacturing operations. In job-shop manufacturing, parts and systems are typically manufactured in very low quantities. The parts are highly customized to meet specific functional requirements. Unlike large quantity manufacturing, job-shop manufacturing is often performed without analogous data to serve as a baseline for estimation and performance monitoring. Therefore, bottom-up estimates, otherwise known as engineering build-up estimates, are used to estimate cost and schedule drivers at lower levels of the work breakdown structure (WBS) (Persons 2020, 107). These estimates are aggregated to form the total cost and schedule budgets for the project.

In the current paradigm at the Prototype and Manufacturing Division (PMD) of Naval Air Warfare Center – Aircraft Division, Lakehurst (NAWCAD-LKE), forming a job-shop manufacturing cost budget can be straightforward when utilizing single-value expectation estimates from subject matter experts (SMEs). To form the project's cost budget, an estimator can sum all the individual cost estimates from the lower levels of the WBS. In a slightly more refined model, adjustments can be made to capture risk and uncertainty based on management's tolerance for variation. Schedule estimates are often developed by assessing the magnitude of work to be performed, evaluating the long lead material items, and performing a subjective assessment based on a lead production controller's experience.

This thesis demonstrates that improved analytical efforts utilizing a combination of earned value management (EVM) techniques alongside critical path / Monte Carlo (CPMC) techniques can provide more information to decision makers who are responsible for determining a quoted cost and schedule to customer representatives.

Job-shop manufacturing efforts often consist of subassemblies and components which can be executed in parallel. This parallel activity throughout the activity network drives a need to calculate the critical path, or the longest series of sequential activities throughout the network, to form an estimate for the schedule budget.

To calculate the critical path, each operation within the network requires a labor hour estimate. These low-level estimates are subject to their own risk and uncertainty. SMEs can provide labor hour estimates for the best case, worst case, and most likely case to objectively quantify the impacts of risk and uncertainty. Estimators can use that input from the SMEs to calculate a probability density function (PDF) in the form of a triangular distribution representing the estimated labor hours for each activity in the network.

From a cost budget perspective, these PDFs help decision makers understand the possible range of values that may be realized during actual execution. To utilize this information to better inform a schedule estimate, estimators must perform a Monte Carlo analysis to determine how sensitive the critical path is to the potential labor hour variation. The result of the CPMC analysis can provide decision makers with insight into the possible durations that could occur given the expected labor hour variation.

During execution, recomputing the CPMC analysis with actuals from execution can help program managers make prioritization decisions when determining which projects should be accelerated when competing projects require the same resources. Evaluating the float of each project's schedule and determining if the specific operation is likely to affect the critical path of the project if it is delayed can aid managers to make informed production decisions that reduce the impact when attempting to meet overall schedule requirements.

One of the outcomes of this work is a tool, written as an extension to Microsoft Excel™ utilizing Visual Basic for Applications (VBA) code, that can be used to perform Monte Carlo analysis on cost and schedule for job-shop manufacturing efforts. Additionally, this research concludes that earned value management (EVM) metrics can be useful and effective for reporting on historical performance of an effort when a baseline schedule can be established, but EVM does not address expected variation. CPMC analysis helps to bridge this gap in information, providing estimators and decision makers with tools to anticipate variances and manage collective customer requirements and expectations.

References

Persons, Timothy M. 2020. *Cost Estimating and Assessment Guide*. GAO-20-195G. Washington, DC: Government Accountability Office. <https://www.gao.gov/assets/gao-20-195g.pdf>.

ACKNOWLEDGMENTS

I would like to extend my deepest gratitude to my advisor, Timothy Anderson, for their unwavering guidance, expertise, and mentorship throughout this developmental process. Your insights, constructive feedback, and encouragement have been invaluable in shaping the direction and quality of this work.

I would also like to extend my heartfelt appreciation to my wife, Debi, whose unwavering love and understanding have been instrumental to my success. Your belief in me has been a constant source of strength. Your sacrifices and unwavering support have allowed me to focus on my research and strive for excellence. I could not have done this without you.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

In defense job-shop manufacturing, the budget and schedule of a project is heavily influenced by early cost and schedule estimates. In many areas, these initial estimates are often developed and communicated as point estimates without integrated risk and uncertainty. Furthermore, the communicated point estimates often consist of many decomposed cost and schedule point estimates that are aggregated across the work breakdown structure (WBS) (Naval Center for Cost Analysis [NCCA] 2014, sec. 2.2).

Engineering build-up estimation, otherwise known as the bottom-up estimation method, is used in job-shop manufacturing to gather estimates from the various subject matter experts (SMEs) (Office of the Secretary of Defense Cost Assessment and Program Evaluation [CAPE] 2020, sec. 6.1.2). SME point estimates may be skewed by motivational and cognitive biases (NCCA 2014, sec. 2.5.2). SMEs may also be responsible for ensuring work is performed against cost and time estimates at lower levels of the WBS. If the SME is motivated to ensure work is completed *on time* and *on budget*, the SME point estimate may be generally higher than an unbiased estimate. This upwardly biased estimate can help to ensure that the performance variation of resources assigned to perform the work does not result in low-level budget shortfalls or schedule overruns however, in the aggregate, this can cause overall budget inflation.

Variances from point estimates occur during project execution. When program managers are limited to baseline point estimates, their ability to set meaningful tripwires and other condition-based interventions at the lower levels of the WBS is hindered. Utilizing methods reflected in the *DAU Earned Value Management “Gold Card”* provides a view on overall project execution (Defense Acquisition University [DAU] 2020), but this process can be extended to capture risk and uncertainty and allow for lower WBS level element monitoring.

This thesis explores the usage of stochastic S-Curves (SS-Curves) in the estimation and execution of efforts performed in a job-shop manufacturing environment. This chapter

discusses the motivation for the thesis, the thesis research question, the scope of the thesis, and the benefits of this study.

A. MOTIVATION

Cost and schedule analyses are performed progressively throughout the systems engineering life cycle. As requirements are initially formed and then refined, cost and schedule are used as either objectives or constraints that support analysis of alternatives. As a program transitions to initial prototyping and low-rate production in the implementation and integration phase, costs and schedules need to be generated to provide program managers budgetary estimates for initial provisioning and serve as a framework for monitoring the performance of the effort.

At the Prototype and Manufacturing Division (PMD) of the Aircraft Launch and Recovery Equipment (ALRE) and Support Equipment (SE) Group, Naval Air Warfare Center – Aircraft Division (NAWCAD), projects are often estimated using decomposed point estimates. During the estimation process, a project lead will establish a WBS, which is often reflective of the project's parts list. For each element of the WBS, a routing list will be created that represents the flow of materials through the various work centers of the job-shop. Work centers are a collection of resources that perform similar work (e.g., Assembly, Metal Fabrication, Welding). SMEs that are aligned to each work center will provide individual point estimates for the scope of work being performed in their area. When all SMEs have provided input, the project lead will add all estimates together, taking into consideration allowable concurrency, to provide a point estimate for cost and schedule to the sponsor.

PMD also uses those estimates as the basis for monitoring the performance of projects as they are executed. The data reported during execution at the individual routing line are:

- Time Worked – For each routing line, time worked (TW) at time (t) is the difference between the end time (ET) and the start time (ST) summed across the set of time entries (TE) that have occurred before time (t).

$$TW_t = \sum_{i \in TE} (ET_i - ST_i)$$

- Time Remaining – For each routing line, time remaining (TR) at time (t) is the non-negative difference between the estimated time (Est) and the time worked (TW) performed against that level of effort before time (t).

$$TR_t = \max(Est - TW_t, 0)$$

- Percent Complete – Subjective input from the artisan or inspector performing the work collected at the end of each shift.

Communication of expected variances is essential to fully inform decision makers of the possible outcomes that may arise during project execution. When using a bottom-up cost and schedule estimation method without information on expected variances, program managers will often set a project budget based on the SME-derived point estimates and then apply subjective adjustments in the form of Management Reserves to attempt to capture anticipated variations. When budgets for variances are allocated at higher level WBS elements, it becomes more difficult to pinpoint specific activities that are experiencing performance issues.

B. RESEARCH QUESTIONS

Can a probabilistic estimation and performance reporting approach to job-shop manufacturing provide higher fidelity information to external project stakeholders?

- Can a job-shop manufacturing effort use probabilistic bottom-up estimation techniques to communicate anticipated variation or level of cost / schedule risk associated with a proposal?
- Can earned value management (EVM) projections be improved using activity-level probability distributions?

C. BENEFITS OF THIS STUDY

A bottom-up probabilistic estimation approach for job-shop manufacturing activities can increase the accuracy of estimates and ensure that risks/opportunities that

affect cost and schedule are objectively captured as potential variation. This approach can also help to inform program managers of realized risk/opportunities and discrete variation that exceeds expected values to combat the masking effects of cumulative performance reporting. A well-informed project management team can leverage the additional data provided by probabilistic sensitive reporting to make decisions relevant to funding, resource management, and schedule.

II. RELATED WORKS

This chapter reviews existing work related to this thesis. It presents a summary of related literature associated with cost and schedule estimation, earned value management, and reporting methods to plan and forecast job execution.

A. GAO COST ESTIMATING AND ASSESSMENT GUIDE

The U.S. Government Accountability Office (GAO) recommends the use of the *Cost Estimating and Assessment Guide* (GAO Guide), first published in 2009 and revised in 2020. In this guide, Persons (2020) outlines the steps in the estimating process without restriction on program size or complexity. Generally, the GAO Guide describes the four (4) stage, twelve (12) step process to developing a cost estimate as shown in Figure 1.

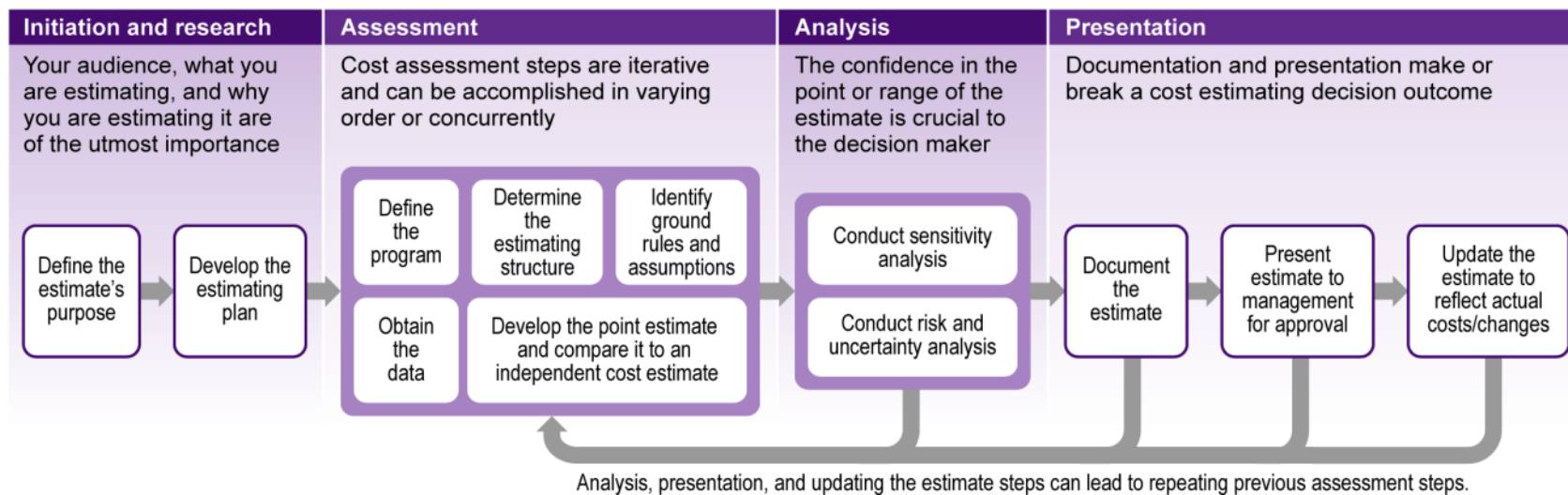


Figure 1. The Cost Estimating Process. Source: Persons (2020, fig. 5).

1. Initiation and Research Stage

The first stage of the cost estimating process within the GAO Guide is the “Initiation and research” stage (Persons 2020). This stage is largely centered around establishing expectations for the estimate. This is achieved by two steps established in the GAO Guide, “Define the estimate’s purpose” and “Develop the estimating plan.” The outcome of these two steps is a well-defined set of expectations for the estimating team and leadership.

a. Define the Estimate’s Purpose

As Persons (2020) explains in the GAO Guide, in the first step, the estimate’s purpose is defined such that cost estimators are informed of the intended use of the estimates and what types of decisions the estimate is intended to support (38). It also codifies the scope of the estimate to establish items that will be included or excluded from the resultant estimate (38–39). In this stage, cost estimators also assess the maturity of the effort (38). The maturity of the effort will have an impact on the level of detail that needs to be included in the estimate (38). The GAO Guide indicates that in situations where the technical maturity of the system is low, “more detail [does] not provide greater accuracy,” and that “if a detailed technical description of the system being analyzed is lacking information, analysts will find it difficult to identify an estimate all of the cost elements” (Persons 2020, 38). For these scenarios, the guide indicates that “it may be better to develop the estimate at a relatively higher [WBS] level” (Persons 2020, 39).

b. Develop the Estimating Plan

Persons (2020) indicates that in the next step, the estimate plan is developed (41). In this step, the estimating team is formulated from personnel from multiple disciplines (41). The plan should outline the roles and responsibilities of each member of the team (41–44). Additionally, the plan should consist of “a detailed schedule to complete the cost estimate that includes realistic key decision points or milestones, and that provides margins for unforeseen, but not unexpected, delays” (Persons 2020, 44).

2. Assessment Stage

Pearsons (2020) continues to the assessment stage, which consists of defining the program and determining the estimating structure (48–70). Additionally, in this stage, ground rules and assumptions are documented, relevant data is gathered, and a point estimate is developed and validated (71–129). This stage is the first stage where a feedback loop is introduced. The estimation team may need to revisit some of these steps as more information is uncovered throughout the estimation process.

a. Define the Program

Pearsons (2020) establishes that the first step of the assessment stage is to define the program (48). In this stage, the initial technical baseline is defined (48–49). The technical baseline includes “information on development, testing, production, operations and maintenance, planned upgrades, and disposal” (Persons 2020, 49). Many technical baseline elements are enumerated within Table 4 of the GAO Guide. Some examples of technical baseline elements include the system purpose, the work breakdown structure, the acquisition plan or strategy, the safety plan, and many others (table 4). The elements of the technical baseline will vary and should be tailored to meet the system’s needs (50). Beyond the technical baseline elements, key system characteristics and performance parameters will also need to be identified (52). Table 5 of the GAO Guide lists many example system characteristics, and they are also expected to be tailored based on the needs of the system.

b. Determine the Estimating Structure

Persons (2020) defines the second step of the assessment stage as determining the estimating structure, or more commonly referred to as the work breakdown structure (56). The WBS is a hierarchical structure of effort (56). There are two different types of WBS structures—a functionally aligned structure where elements of the WBS are organized by the category of the activity, and a product aligned structure where elements of the WBS are organized by segments of the physical architecture (Persons 2020, 59).

Persons (2020) advocates for breaking work into smaller and more manageable work elements to support earned value management (EVM) (66). The GAO Guide

indicates that “a WBS is an essential part of EVM cost, schedule, and technical monitoring because it provides a consistent framework from which to measure progress.” It continues to say “the WBS also provides a common thread between EVM and the schedule that allows for further understanding of program costs and schedule variances.” (Persons 2020, 66). The impact that the WBS has throughout the estimation and execution of an effort makes it an essential element of the cost estimate (67–68).

c. Identify Ground Rules and Assumptions

Persons (2020) identifies the next step of the assessment stage as identification of ground rules and assumptions (71). Ground rules are “a common set of agreed upon estimating standards” (Persons 2020, 71). These standards codify the constraints under which the estimate is developed and can include budgetary constraints, or multi-year execution planning methods (73–75). In job-shop manufacturing, common ground rules include the identification of boundaries between partnering organizations, quantities for production and spare/repair parts, and items omitted from the estimate.

In the absence of comprehensive ground rules that cover all elements of the cost estimate, assumptions must be made (71). Assumptions should be made by experienced estimators who can ensure their assumptions are realistic and by those who understand the effects of their assumptions (71). As more information comes out about an effort over time, these assumptions may need to be adjusted. When assumptions are adjusted, it is important to have traceability to the parameters influenced by the assumption to allow for reformulation of the estimate.

d. Obtain the Data

Persons (2020) indicates that once the program and structure are defined and when the ground rules and assumptions are established, the initial data collection step can begin (81). There are generally three main types of data that support estimation: cost data, schedule or program data, and technical data (81). Cost data is often limited to hours of resources needed by skill level, as other costs associated with the operation of facilities are often not available (84). For the purposes of cost estimating, schedule or program data is used to inform on aspects of historical execution that may have been affected by schedule

pressures or other program choices (84). Technical data is essential to have a full understanding of the activities or drivers to the cost data (85).

One of the biggest challenges for a cost estimator is assessing the available data and deciding what data would be useful to support the specific cost estimate (89–90). Persons (2020) identifies a few factors to consider when assessing the applicability of available data (89–90). These factors include consistency between the historical work content with the current work content, consistency of performance over time, and the level of detail in the data (89–90).

Once data has been collected, initial analysis of the data can begin (90). Descriptive statistics should be developed to identify correlations in data (91). Calculating metrics like the mean, standard deviation, variance, and others can provide insight into trends that exist in the underlying data (91). Outliers in data should be identified but retained in the data set to prevent claims of bias or improper sanitization of information. (91)

Finally, the data that is collected needs to be documented so that as more information becomes available, the appropriate data elements can be maintained (98). A strongly documented data set can be easily adjusted as early assumptions related to a project are refined into more formal definitions (98). A robust documentation of data can also support subsequent cost estimates (99).

e. Develop the Point Estimate and Compare It to an Independent Cost Estimate

Persons (2020) then indicates that in the development of a point estimate and independent cost estimate comparison step, a point estimate is first developed (102). There are many methods to develop cost estimates. According to the GAO Guide, the most common methods are analogy, engineering build-up, and parametric estimation (102). The analogy method takes one or more similar programs and adjusts to account for differences (103). The engineering build-up method aggregates smaller scoped cost estimates through a fully decomposed WBS (103). The parametric method leverages system attributes or other parameters to extrapolate costs from established relationships (103).

In job-shop manufacturing, the most common method used for point estimating is engineering build-up. Often efforts that are suitable for job-shop manufacturing are relatively small in scope compared to a full program. Additionally, a detailed design is often available to job-shop manufacturing cost estimators. One of the largest disadvantages of performing an engineering build-up estimate is that it is often costly and labor intensive to develop (108). However, the structures developed in the estimating process can be directly utilized during execution to manage execution of an effort through a job-shop facility.

An independent cost estimate (ICE) is typically something prepared by an activity that is not a direct stakeholder in a program's execution (123–124). A separate, nearly unbiased estimate can serve as a method of validation of an offerors' cost estimate (123–124). An independent estimate typically utilizes parametric or analogous estimating methods and there is often considerable effort required to reconcile estimates when comparing an engineering build-up estimate to a higher-level estimate (128).

3. Analysis Stage

The next stage of the cost estimating process developed by Persons (2020) is the analysis stage (34). The analysis stage consists of conducting a sensitivity analysis and performing a risk and uncertainty analysis (34). These analyses are essential to communicate to decision makers the potential sources of variation that could impact the resultant performance of an effort (130). Further assessment and data collection may be needed to progressively narrow the resultant analysis.

a. Conduct Sensitivity Analysis

After the assessment stage, a point estimate is available that was prepared utilizing the ground rules and assumptions that had been established. Persons (2020) then indicates that a sensitivity analysis is conducted to understand the relationship between the assumptions and parameters and the resultant cost estimate (130). To perform a sensitivity analysis, each assumption or parameter is varied independently over a range of values, and the resultant cost is recorded (130–131). This feedback loop to the assessment stage further emphasizes the need for strong data hierarchy and documentation. Relationships between

the varied assumption or parameter and the cost are evaluated to determine which factors have the greatest impact on the overall costs (135–136).

b. Conduct Risk and Uncertainty Analysis

Also, after the assessment stage, Persons (2020) indicates that a risk and uncertainty analysis can be performed. There are many sources of risk that could affect the performance of an effort. According to the GAO Guide, risks can result in impacts to costs, durations, performance parameters, and requirements (139). An objective assessment of risk at the lowest levels of a cost estimate will result in potential variation of costs that can, in the aggregate, significantly affect the confidence that actual costs will be at or below a point estimate (139–140).

In a job-shop manufacturing environment, risks can be in the form of an expected scrap rate, availability of resources (such as equipment failures or labor shortages), poorly defined technical requirements, and many others. These risks can be quantified with a likelihood and a consequence to be factored into an overall cost estimate.

There are many other sources of variation that are much more difficult to anticipate. Uncertainty is present when there are hidden factors that can influence the performance of an effort (141–142). These factors can include varying efficiency and performance of laborers, inconsistent or inaccurate data that served as the basis for the estimate, and biases that may have influenced the estimation process (142).

The GAO Guide describes a process to conduct a cost risk and uncertainty analysis. This process starts with assigning probability distributions to WBS elements that reflect the risk and uncertainty that could apply to that specific WBS element (144). A Monte Carlo simulation can then be performed.

In a Monte Carlo simulation, software can perform a series of trials against the data set that result in the random assignment of specific values from the various distributions (144). The results of each trial are aggregated, and the possible outcomes are recorded (144). The distribution of outcomes can be used to communicate the range and likelihood

of the actual cost (144). Often this communication is done with a graphical representation of data in the form of a cumulative distribution function, or S-Curve (144).

Figure 2 shows an example of an S-Curve from the GAO Guide. Decision makers can review the curve and set a budget that would result in the probability that the cost would be less than or equal to the budget selected (145). In the example in Figure 2, If a budget of \$1,096 thousand was selected, there is a 70% chance that the actual costs will be less than or equal to \$1,096 thousand. The final budget selected by decision makers should be based on mission or business factors that would impact their tolerance to risk acceptance (145).

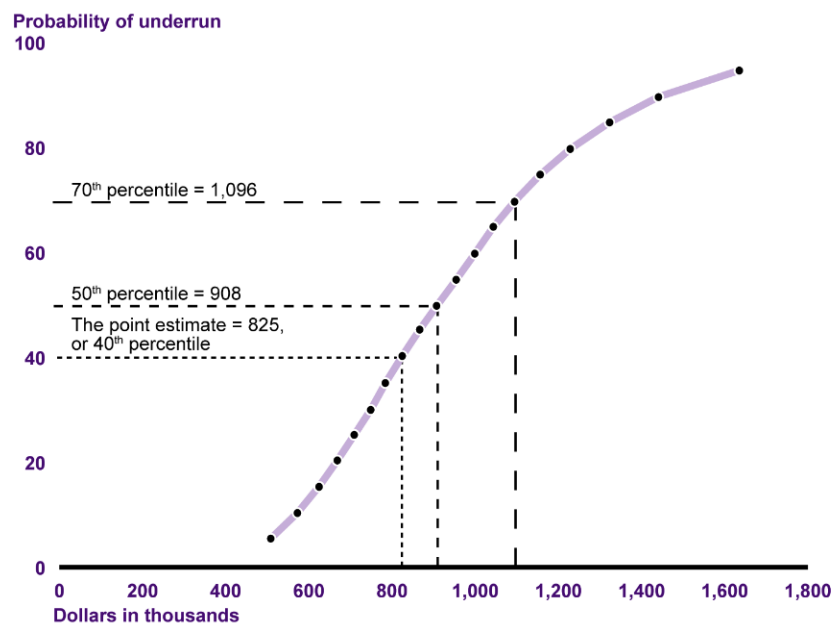


Figure 2. A Cumulative Distribution Function, or S-Curve.
Source: Persons (2020, fig. 14).

As an effort is executed, the risk and uncertainty analysis should be repeated to capture the results of realized risk and to recalculate the new distribution of possible outcomes given the performance of the project to date (163). Correlational factors should also be considered in this evaluation, as early cost variation may be indicative of future variation (155–158).

4. Presentation Stage

After the analysis stage, Persons (2020) indicates that cost estimators can begin the presentation stage (34). In this stage, the estimate is documented and presented to management for approval (or for further refinement) (167). This stage also consists of the continual updates needed to reflect actual costs or changes that occur during execution (167).

a. Document the Estimate

Persons (2020) establishes that documentation of the cost estimate should be performed after analysis is completed (167). The GAO Guide provides Table 15, which indicates the types of information that should be included in this documentation. The GAO Guide recommends that the documentation should consist of the following sections (table 15):

- Cover page and table of contents
- Executive summary
- Introduction
- System description
- Program inputs
- Estimating method and data by WBS cost element
- Sensitivity analysis
- Risk and uncertainty analysis
- Management approval
- Updates reflecting actual costs and changes

All the information in the documentation is necessary to ensure that the cost estimate can be validated and defended (167). Having high fidelity information contained

in the estimate's documentation will help inspire confidence in the estimate and demonstrate that due diligence had been performed (167). The GAO Guide also stresses the importance of having a well-documented estimate to ensure that others can understand the methodology of the estimate and can either adjust or repurpose the estimate in the future (167). It also supports the independent validation of the estimate and gives validators information to support reconciliation between an independent cost estimate and the prepared estimate (167).

b. Present Estimate to Management for Approval

After documentation of the estimate is completed, Persons (2020) states that the estimate may be presented to management or other decision makers for approval (176). The reporting requirements for an estimate may vary depending on the expectations set forth by the decision maker but generally, the presentation should be clear and provide sufficient detail to inspire confidence in the estimate (176–177).

The result of the presentation should be a determination of whether management accepts the cost estimate (177). Additionally, management should set the confidence level to resolve the probability distribution into a point estimate that can be communicated to the requesting activity (177).

In this stage, a possible outcome is a rejection of the estimate by the decision maker. Iteration may be necessary to refine the estimate to capture adjustments to the overall scope, assumptions, or ground rules so that an estimate that meets the refined expectation can be provided. This further emphasizes the importance of robust documentation and traceability in the estimate so that adjustments can be incorporated without having to completely re-perform the estimation effort.

c. Update the Estimate to Reflect Actual Costs/Changes

Persons (2020) indicates that during execution of the effort, actual cost and performance data becomes available (180). Estimates should be regularly updated with actual data to refine projections for the estimate at completion (EAC) (180–181). This information should be reported regularly utilizing EVM or other reporting methods to

communicate variances (181). The assumptions that were made early in the estimating process may ultimately prove to be inaccurate (181). As a result, the baseline estimate may also need to be iterated to accommodate significant changes in scope (181).

B. DOD COST ESTIMATING GUIDE

In 2020, the DOD issued the *DOD Cost Estimating Guide* (DOD Guide) which generally embraces the GAO perspective on cost estimating while also including provisions from component guides, handbooks, and manuals (CAPE 2020, 20). The DOD Guide focuses on Figure 3 as a basic framework which predominantly emphasizes the use of “Data” throughout the estimating process and further stresses the iterative nature of the cost estimating process.

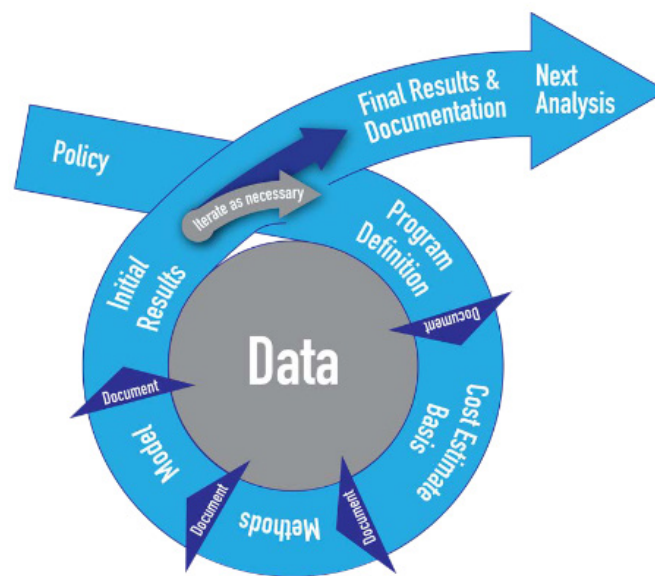


Figure 3. The DOD Cost Estimating Process. Source: CAPE (2020, fig. 1).

One of the largest differences between the GAO Guide and the DOD Guide is that the DOD Guide relies on incremental review, approval, and refinement of process artifacts to achieve a more sequential process flow (CAPE 2020, 20). Conversely, the GAO Guide supports continuous iteration and interaction between stages as requirements are refined and as information becomes available (Persons 2020, 34). This high-level procedural

difference may be driven by DOD’s experiences interacting with its Prime Contractors as more frequent iteration can drive extensive estimation costs.

1. Policy

In the CAPE (2020) DOD Guide, the cost estimating process starts with established policies (7). The DOD Office of Secretary of Defense (OSD) Cost Assessment and Program Evaluation (CAPE) maintains policies and procedures for cost analysis (7). These policies and procedures are used in concert with statutes and directives from the United States Code (USC) as set by the United States Congress (7).

2. Program Definition

CAPE (2020) indicates that the Program Definition is largely captured in the cost analysis requirements description (CARD) (25). The CARD is a document that captures a snapshot of the program that includes programmatic, performance, technical, operational, sustainment, and schedule characteristics of a program (27). Additionally, in this phase of the estimating process, many similar steps from the GAO Guide’s Assessment Stage are performed including the initial development of a WBS and an overall cost estimating model (27–29).

3. Cost Estimate Basis

Estimate planning, defining ground rules and assumptions, and documentation are all part of the CAPE (2020) DOD Guide Cost Estimate Basis phase (31–36). It includes program management approval of the program WBS and further refinement of that WBS to support the cost estimate (33). It also includes cost estimate scheduling to establish key dates to support program decision milestones (33–34).

4. Select Cost/Schedule Estimating Methods

CAPE (2020) indicates that the selection of the estimating method is largely dependent on “the maturity of the program, the nature of a particular element of the estimate structure, the usefulness of available data, and the time available to develop the

estimate.” (55). The DOD Guide recommends the use of analogy, engineering build-up, extrapolation of actuals, and parametric methods (55).

The DOD Guide also provides guidance on when each of the methods are most applicable depending on the current progress of the program through the system’s life cycle (55–60). Figure 4 shows the generic system life cycle, where progress through the life cycle is represented in the X-axis. The Y-axis represents the percentage where an estimating method might be suitable in that phase of the life cycle. As an example, in the material solution analysis (MSA) phase, approximately 85% of the time, the analogy method would be most applicable, with the remaining 15% of the time being performed with the parametric method. This figure is provided as a “rule-of-thumb,” but it does provide additional insight into how estimation can become more accurate as the system becomes more well-defined (60).

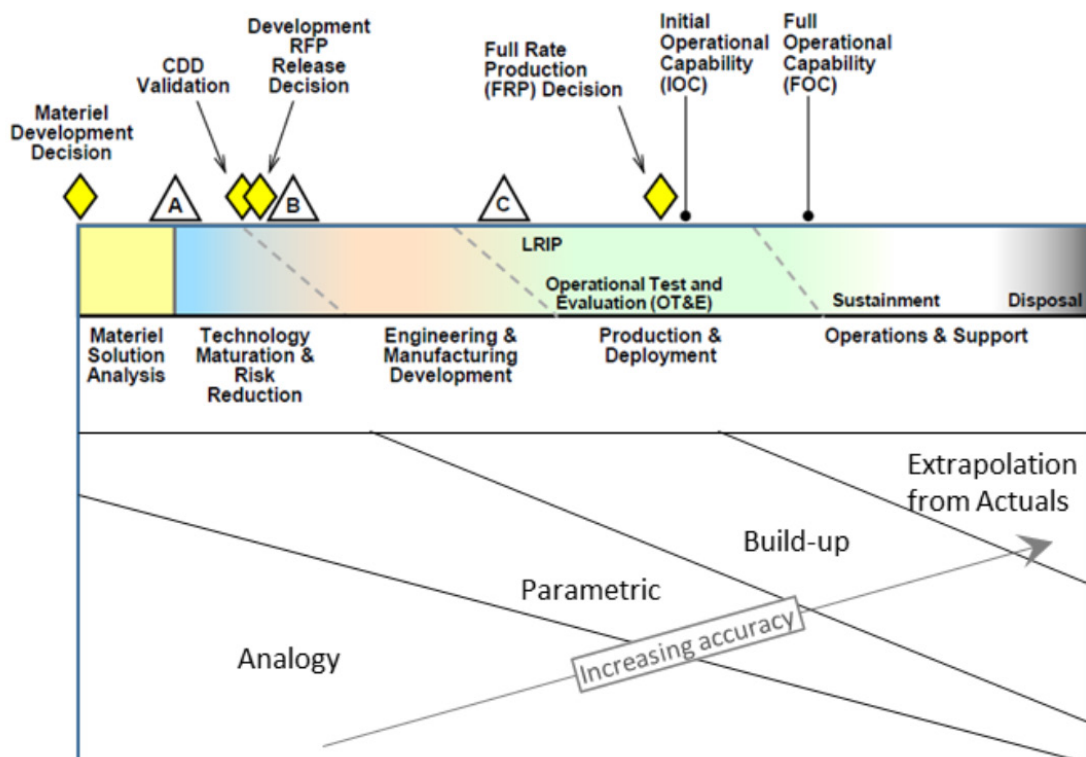


Figure 4. Estimating Method Applicability. Source: CAPE (2020, fig. 2).

5. Build Cost Estimate Model

CAPE (2020) identifies the cost estimate model as the tool that brings all the data together to start forming an aggregated estimate (68). The DOD Guide is not prescriptive on which tools to use to perform this activity, however it does provide guidance on what elements should be included to ensure a comprehensive assessment (73). The DOD Guide also stresses the importance of building mathematical links within the model so that items that are driven, or otherwise related, to other elements of the model are directly linked to allow for proper propagation of changes (73).

The DOD Guide also includes guidance for capturing the phasing of costs throughout a project's life cycle (70–71). Capturing cost and schedule expectations for obligations and expenses is important for DOD programs as those values are needed to request the right types of funding in the correct fiscal year (70–71). These costs should also be organizable by the “Common Cost Metrics” as shown in Figure 5. These metrics are useful to high level decision makers when assessing alternatives (74).

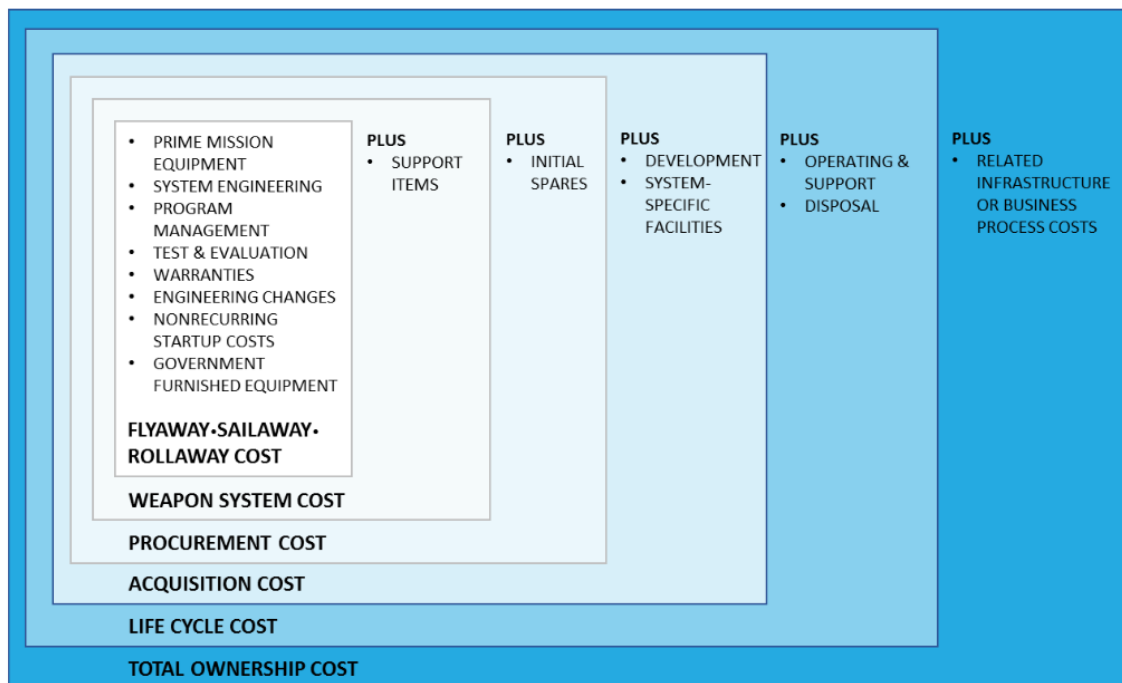


Figure 5. Total Ownership Cost Composition. Source: CAPE (2020, fig. 5).

6. Review the Initial Results

According to CAPE (2020), reviewing the initial results of the estimate is a critical step to ensure the integrity of the estimate prior to further dissemination of the result (78). The first step of this process is to perform a crosscheck, or sanity check, on the outcome of the estimate (78–79). Generally, this step is to ensure that the methods and model have produced a reasonable value and the model parameters drive the appropriate behavior in the estimate (78–79).

This phase also addresses the execution of sensitivity analysis and what-if analysis (79). An estimate model may include many different model parameters such as physical and performance characteristics, cost and schedule constraints, among others (79). Each of these parameters will influence the result of the model (79). A sensitivity analysis evaluates the magnitude of overall change that would result from an independent incremental change to each model parameter (79).

Within the DOD Guide, sensitivity analysis also includes the execution of uncertainty analysis (82–83). The DOD Guide lists the common methods of uncertainty analysis which includes case-based risk, method of moments, simulation (inputs-based), and simulation (outputs-based) (81). This thesis focuses on the performance of inputs-based simulation to perform trials to simulate uncertainty in the lowest-level estimated values.

7. Final Results and Documentation

CAPE (2020) indicates that the results and documentation of the estimate are critical to communicate the results of the effort (85). The general intent of the documentation is to ensure that others can understand, validate, or replicate the results (85). The DOD Guide describes the recommended contents of the results which includes all the input information that was used to formulate the estimate and the analysis that was performed (85–86).

Several charts and graphs are proposed in the DOD Guide to aid in the communication of results. A sand chart, as shown in Figure 6, can demonstrate the amount

of funding needed by fiscal year, categorized by the different phases of acquisition (88–89).

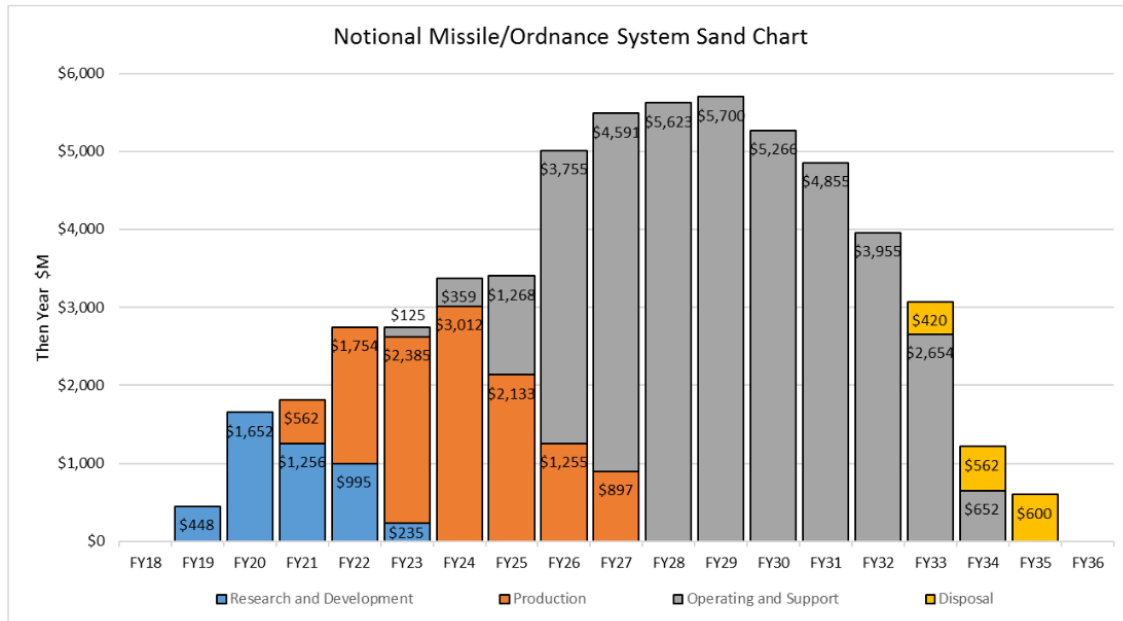


Figure 6. Sand Chart (Stacked Bar) (Notional). Source: CAPE (2020, fig. 8).

A Pareto Chart, as shown in Figure 7, can show the largest contributors to cost (90). This information can be useful to identify which components of the system are driving the overall cost (90).

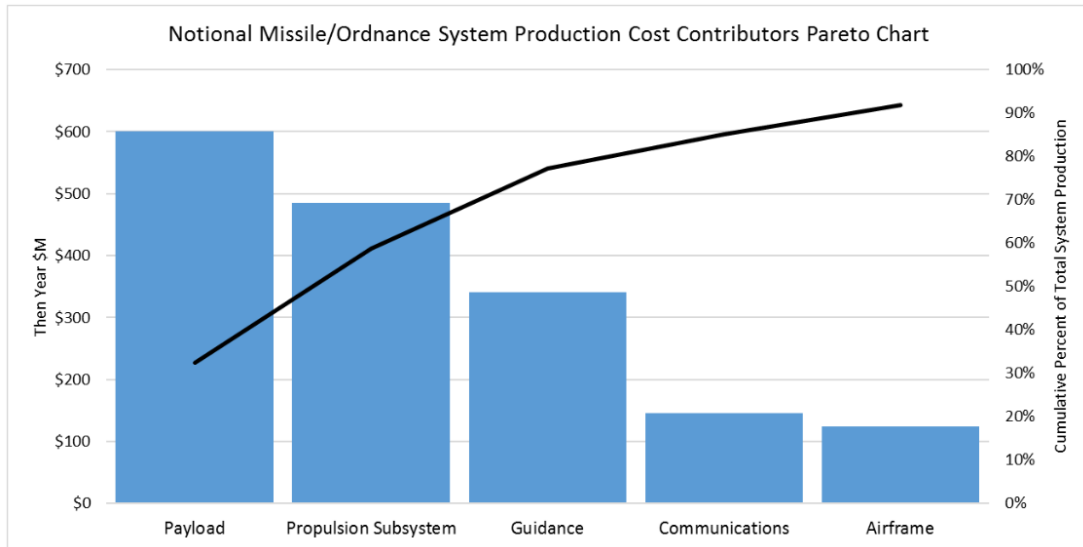


Figure 7. Pareto Chart (Notional). Source: CAPE (2020, fig. 9).

A S-Curve Chart, see Figure 8, is also included in the DOD Guide. This type of chart is very useful to communicate the range of costs that correspond with the probability of an outcome (94). The X-axis represents an estimated value for a variable parameter such as cost or schedule and the Y-axis represents the probability that the value will take a resulting value less than or equal to the estimated value.

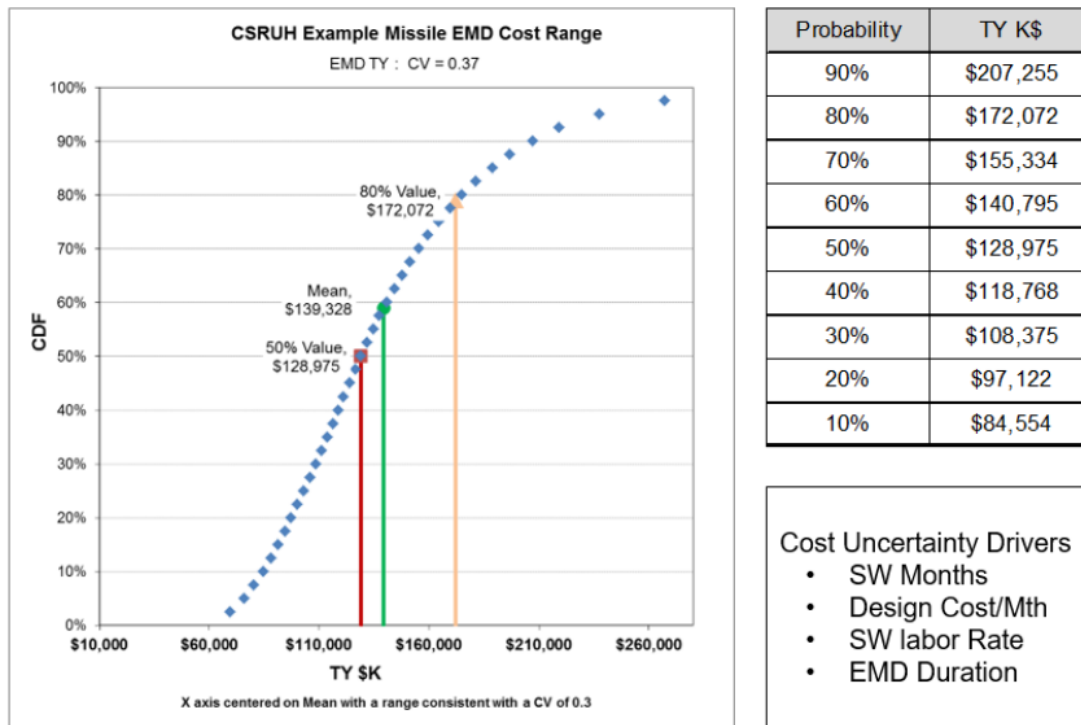


Figure 8. S-Curve Example (Notional). Source: CAPE (2020, fig. 14).

C. JOINT AGENCY COST SCHEDULE RISK AND UNCERTAINTY HANDBOOK

Both the Persons (2020) GAO Guide and the CAPE (2020) DOD Guide make considerable references to the *Joint Agency Cost Schedule Risk and Uncertainty Handbook* (JA CSRUH) published in 2014 by the Naval Center for Cost Analysis (NCCA). This handbook expands on the “Risk and Uncertainty Analysis” phase of the GAO Guide and is generally referenced in the DOD Guide in sections describing cost estimation methods and the collection, validation, normalization, and analysis of data. The NCCA (2014) notably describes the uncertainty that needs to be captured in the generation of a cost estimate (8). It describes the need to develop a point estimate while concurrently estimating the combined uncertainty of every element based upon cost, schedule, technical, and risk register influences to provide a basis for estimating the probability of exceeding a specific budget (8). The JA CSRUH also provides recommendations (see Table 1) for the types and degree of uncertainty distributions to leverage based on the application. As PMD presently

relies on SME opinion for the basis of cost estimates, triangular distributions will largely be used for analysis in this thesis.

Table 1. Recommended Uncertainty Distributions.
Source: NCCA (2014, tbl. 2–2)

Distribution	Typical Application	Knowledge of Mode	Number of Parameters Required	Recommended Parameters
Lognormal	Default when no better info. Probability skewed right. Replicate another model result. Power OLS CER uncertainty.	Mean or median known better than the mode	2	Median, high
Triangular	Expert opinion. Finite min/max. Probability reduces towards endpoints. Skew possible. Labor rates, labor rate adjustments, factor methods	Good idea	3	Low, mode, and high
BetaPert	Like triangular, but mode is 4 times more important than min or max.	Very good idea	3	Low, mode, and high
Beta	Like triangular, but min/max region known better than mode.	Not sure	4	Min, low, high, and max
Normal	Equal chance low/high. Unbounded in either direction. Linear OLS CER uncertainty.	Good idea, but unbounded in either direction	2	Mean/Median/Mode and high value
Uniform	Equal chance over uncertainty range. Finite min/max.	No idea	2	Low and High

The JA CSRUH also extensively discusses the subjective uncertainty associated with the elicitation of estimate bounds (high and low values) from SMEs (28–30). Numerous motivational and cognitive biases are identified in the handbook that could

influence the range of values received from a SME (28–29). The handbook describes some methods to augment the values provided by the SME to adjust them to capture a historically realistic range of values along with adjustments to the bounds for skew (30–32). This method is demonstrated in Section VI.B.

D. EARNED VALUE MANAGEMENT

Earned value management (EVM) was introduced in the DOD and NASA in the early 1960s and codified under the original name “PERT Cost” (Department of Defense [DD] 1963). The general concepts have remained the same, but some of the terminology has changed. In Figure 9, the Cost of Work Report circa 1962 is shown next to the modern Earned Value Management “Gold Card.” The general concepts established by PERT Cost have remained largely unchanged for the last 60 years.

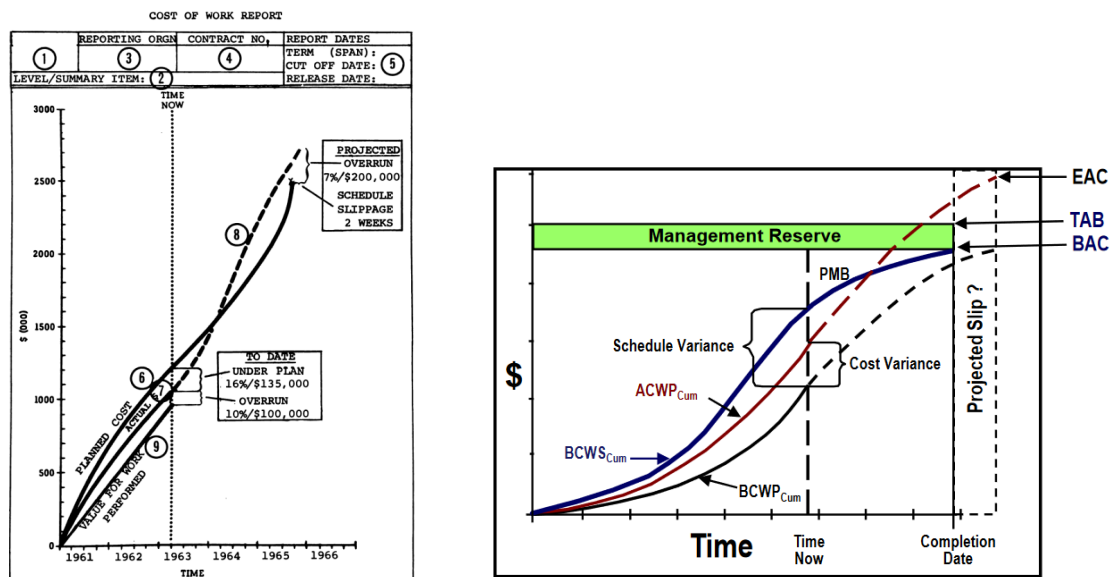


Figure 9. Cost of Work Report (left). Source: DD (1963, fig. 15). Earned Value Management “Gold Card” (right). Source: DAU (2020).

The terms used by these two documents are largely synonymous and a partial cross reference is shown in Table 2.

Table 2. Comparison of Terms Used by PERT Cost (1963) and EVM Gold Card (2020)

PERT Cost Term	Earned Value Management Gold Card Term
Planned Cost	Budgeted Cost of Work Scheduled (BCWS)
Actual Cost	Actual Cost of Work Performed (ACWP)
Value for Work Performed	Budgeted Cost of Work Performed (BCWP)
Latest Revised Estimate	Estimate at Completion (EAC)
Projected (Overrun) Underrun	Cost Variance

Standard EVM plotting is useful when solely considering the expected values of the various parameters of EVM. In many cases, the expected variances of activities under the WBS are known. When those values are known, other plotting methodologies can be employed to provide a comprehensive picture of the range of outcomes that could happen during project execution.

E. PLANNING AND FORECASTING USING INTEGRATED S-CURVES AND STOCHASTIC S-CURVES

When activity variances are known, the EVM plot can be partially redrawn with range values showing cost and duration as distributions around the expected value over the time of the project as shown in Figure 10. The envelope shows the full range of values if no unplanned scope changes or events occur during execution.

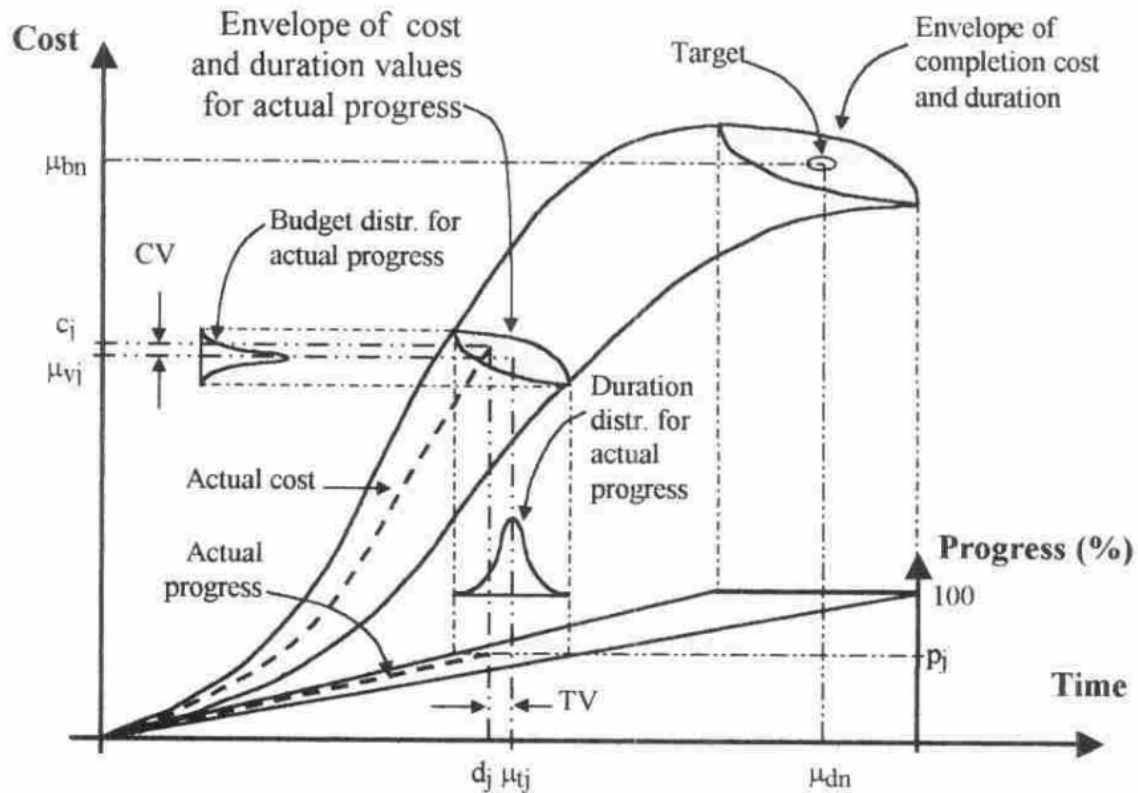


Figure 10. Probabilistic Monitoring Using Integrated S-Curves.
Source: Barraza and Back (2000, fig. 4).

The concepts introduced by EVM techniques can also be extended to include probabilistic forecasting based on anticipated and realized variances. Prior to any execution, if the individual activity's expected values and variances are known, a cumulative planned expected value and variance can be calculated. As work is performed over time and activity variances are realized, the projected cumulative expected value and variance can be recalculated. With no changes to future activity variances, this recalculation will result in a smaller cumulative variance.

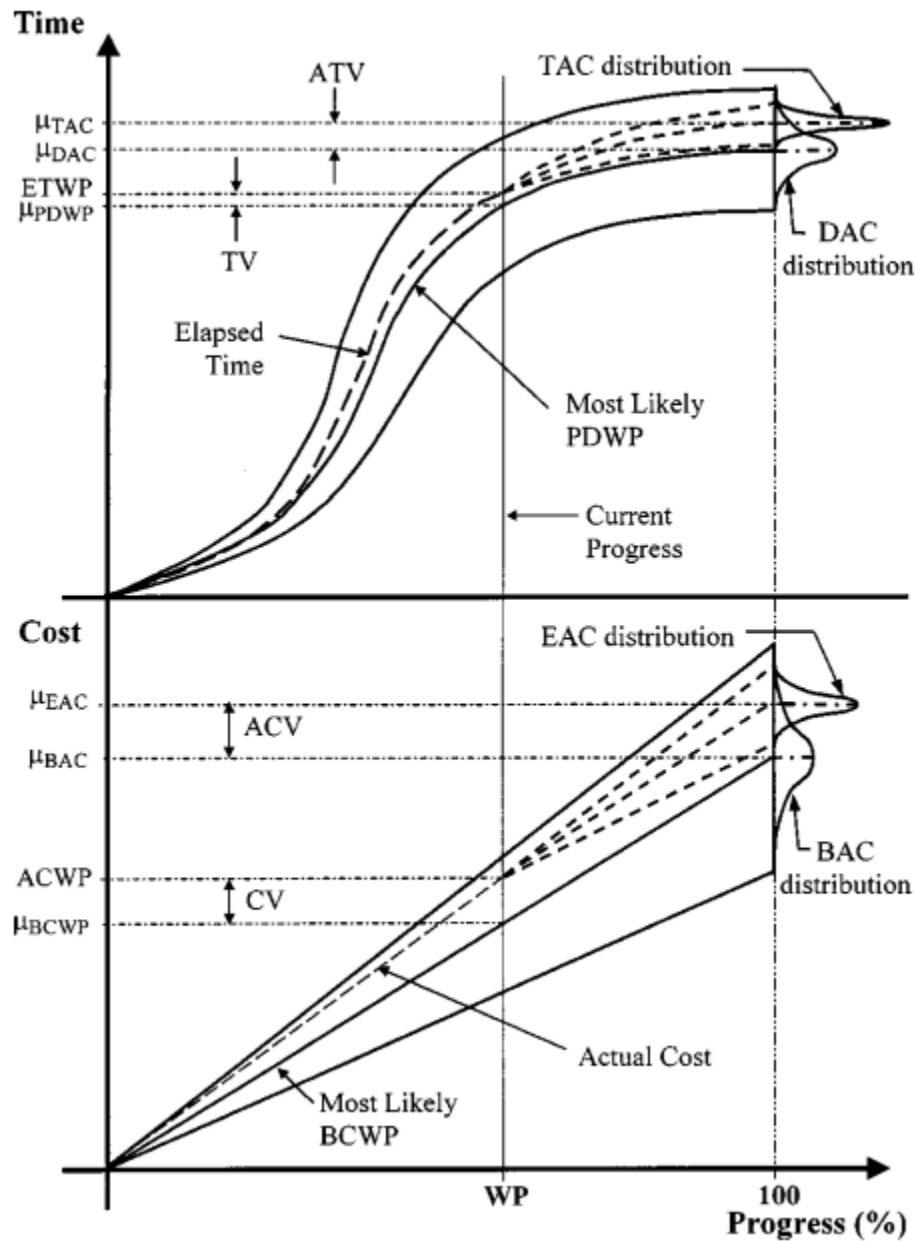


Figure 11. Probabilistic Performance Forecasting Using SS-Curves.
Source: Barraza, Back, and Mata (2004, fig. 2).

F. CALCULATING TOTAL PROBABILISTIC ESTIMATES

Random variables, such as the individual probabilistic cost estimates for operations within a WBS can be added together to calculate the properties of a cumulative random variable, or the total cost. Hayter (2013) indicates that the expected value or mean of the

cumulative random variable (μ_{Total}) is the sum of all individual operation expected values (μ_i) (132). Additionally, the variance of the cumulative random variable (σ_{Total}^2) is the sum of all individual operation variances (σ_i^2), assuming the random variables are independent (132).

$$\mu_{Total} = \sum_{i \in Ops} \mu_i$$

$$\sigma_{Total}^2 = \sum_{i \in Ops} \sigma_i^2$$

Calculating the probabilistic schedule estimate for the entire project is considerably more complex. A simplification of schedule estimating is to determine the single expected critical path through the network of the WBS. Activities on the critical path could be aggregated to determine an incomplete schedule estimate and variance. This method does not account for variation that results in changes to the critical path.

A delay analysis framework to predict critical path changes using sensitivity analysis demonstrates a method to predict the critical path and include the alternate potential critical paths into a comprehensive estimate for schedule duration along with its variance (Hammad et al. 2020).

As shown in Figure 12, the delay analysis framework is executed in three sections. The first phase is to capture the estimates and performance of individual activities. The second phase calculates the impact to an activity by its own performance as well as other activities in the network. The last phase is to calculate the float, or slack, in the network.

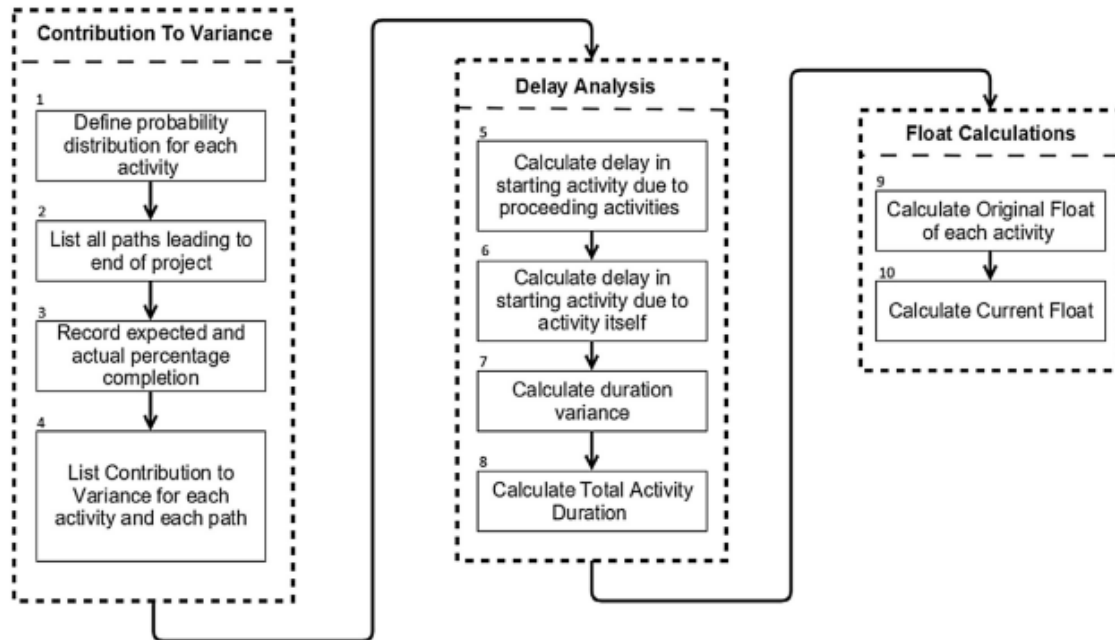


Figure 12. Framework for Predicting the Change in the Critical Path.
Source: Hammad et al. (2020, fig. 1).

III. RESEARCH METHODS

This thesis will utilize the experimental approach (Giachetti 2016). Existing estimation and performance data from PMD will be used as a baseline for estimation accuracy. Historical estimation data from PMD is a single value representing the expected number of hours an operation would take to perform. SMEs will provide new probabilistic estimates for previously performed operations (best case, worst case, and most-likely case). The accuracy of the new estimates will be assessed over time based on the historical performance of the original completed effort. The result of this experimental analysis will be used to determine if a probabilistic approach sufficiently predicts actual performance variation at the discrete operation level. It will also be used to simulate how project reporting could have been prepared with the new baseline throughout project execution.

This thesis will also result in the generation of a software tool as an extension of Microsoft Excel. Visual Basic for Applications (VBA) code will be developed to perform cost and schedule estimation, Monte Carlo simulation, and reporting at various stages of a project's execution cycle.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. BOTTOM-UP ESTIMATION AND FORECASTING FRAMEWORK

A. DEVELOPMENT OF THE WORK BREAKDOWN STRUCTURE

Bottom-up estimation of an effort to produce some quantity of an end item (American Society of Mechanical Engineers [ASME] 2020) first starts with a decomposition of the customer requirement. In job-shop manufacturing, the customer requirement is often provided as a technical data package (TDP) in accordance with MIL-STD-31000 (US Army Research Development & Engineering Command [ARDEC] 2018). The TDP consists of product definition information, which includes models, drawings, associated lists, and other information necessary for a manufacturing activity to produce the end item which meets all functional requirements.

Typically, end items are assemblies, or a group of items joined together to satisfy a functional requirement. Some of the items needed for final assembly may be assemblies themselves. Project planners will evaluate the TDP of an end item and recursively evaluate all the components needed to perform the assembly through the drawing hierarchy. Each branch of the hierarchy is satisfied once the planner reaches the lowest inseparably procurable or manufacturable item for all lower-level branches. The resultant WBS, example shown in Figure 13, is used as a framework to further decompose the customer requirement into actionable activity.

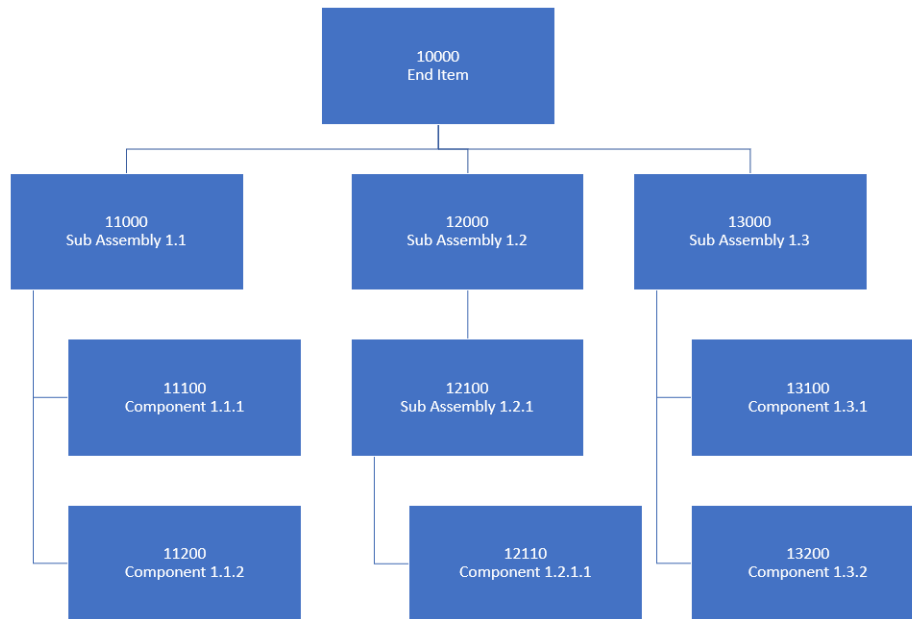


Figure 13. WBS for Example Project

B. POPULATION OF INDIVIDUAL WBS ELEMENTS WITH OPERATIONS

Once the work breakdown structure is established, each element of the structure can be evaluated independently to determine the steps necessary to transform input materials into the required output. Operations within a WBS are defined in series and specify the flow of materials through the various work centers of a shop. An example of operations within a WBS is shown in Figure 14.



Figure 14. Example Sequence of Operations within a WBS

In the example shown in Figure 14, raw material is inspected by Quality Assurance (QA) to ensure the materials conform to the product requirements. Once that step is completed, the Metal Fabrication shop will then saw cut the material to an approximate size. Saw cutting is commonly used as a roughing operation to make larger pieces of material more manageable during subsequent operations. After saw cutting, the material is

moved to the Machine Shop for the removal of excess material to form its final shape. Once the item is fully machined, the QA inspectors will perform dimensional inspection to verify the sizes and form achieved conform with the specifications on the TDP. Next the parts will go to the Paint Shop to receive a coat of primer and then be painted. Finally, the paint is inspected by the QA inspectors and the item is ready to be incorporated in the next higher assembly.

During the planning and estimation process, each operation is assigned an estimated number of billable hours, or the estimated labor costs associated with the execution of the operation. Estimates have typically been in the form of a single scalar value (e.g., 6 hours).

Chapter VI, “Discrete Probabilistic Estimation for Manufacturing Operations,” explores the implementation of a probabilistic estimation approach to capture the planned variation in labor resulting from many factors (e.g., varying artisan skill or the influence of available equipment). These operations will also serve as the billing elements that actual labor hours are assigned to during project execution.

C. FORMULATION OF THE ACTIVITY NETWORK DIAGRAM

Once all WBS elements have established operations, an activity network diagram can be generated. This network shows all the steps necessary to produce a product. Figure 15 shows an example of an activity network diagram where activity is performed on the node. Each node represents an operation where the text within the circle is the operation sequence number. Each operation connects to its next operation within the WBS element. At the end of the specific WBS element, the last operation connects to the first operation of the next higher WBS element.

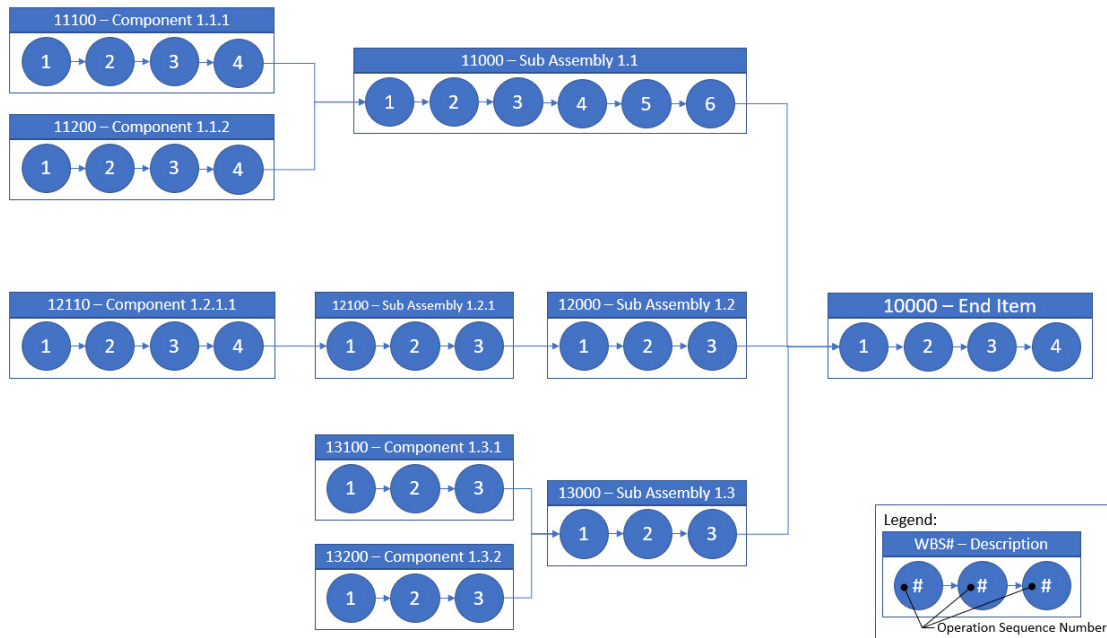


Figure 15. Activity Network Diagram for Example Project

The diagram shown in Figure 15 can also be logically reduced to the diagram shown in Figure 16. In this example, the activity within each WBS element can be additively combined such that the durations and expected variances of the WBS element are the sums of the individual operations that occur within that WBS element.

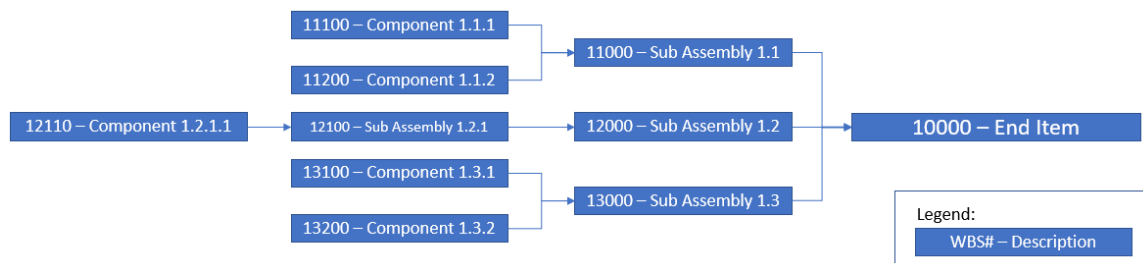


Figure 16. WBS Rollup Activity Network Diagram for Example Project

D. CALCULATION OF THE CRITICAL PATH AND FLOAT

The critical path of an activity network diagram is the path through the network which has the greatest cumulative duration. Any path with a duration less than the critical

path has an amount of float. Float is the amount of time which a path could be delayed without delaying the overall schedule. To calculate the critical path and the float for a project, the different paths through the network need to be enumerated. Once paths have been enumerated, critical path method (CPM) properties need to be calculated for each node. Once CPM properties are known, the data can be used to plot varying execution scenarios as a function of schedule and cost.

1. Enumeration of Paths

An activity network diagram with expected durations of activities can be used to calculate the network's critical path. The examples in Figure 15 and Figure 16 have five (5) distinct paths through the network (shown in Table 3) and any one of them could be the critical path.

Table 3. Paths Resulting from Example Project

Path Number	Path Routing (Rollup)	Path Routing (Expanded)
1	111 → 11 → 1	111-1 → 111-2 → 111-3 → 111-4 → 11-1 → 11-2 → 11-3 → 11-4 → 11-5 → 11-6 → 1-1 → 1-2 → 1-3 → 1-4
2	112 → 11 → 1	112-1 → 112-2 → 112-3 → 112-4 → 11-1 → 11-2 → 11-3 → 11-4 → 11-5 → 11-6 → 1-1 → 1-2 → 1-3 → 1-4
3	1211 → 121 → 12 → 1	1211-1 → 1211-2 → 1211-3 → 1211-4 → 121-1 → 121-2 → 121-3 → 12-1 → 12-2 → 12-3 → 1-1 → 1-2 → 1-3 → 1-4
4	131 → 13 → 1	131-1 → 131-2 → 131-3 → 13-1 → 13-2 → 13-3 → 1-1 → 1-2 → 1-3 → 1-4
5	132 → 13 → 1	132-1 → 132-2 → 132-3 → 13-1 → 13-2 → 13-3 → 1-1 → 1-2 → 1-3 → 1-4

Trailing zeros from WBS Element Numbers have been omitted for clarity.

2. Calculation of CPM Properties for Each Node

To determine the critical path, five properties—Early Start, Early Finish, Late Start, Late Finish, and Float—are calculated for each node in the activity network diagram. As described in Table 4, Early Start (ES) and Early Finish (EF) properties are calculated starting from the earliest nodes in the network and progressively calculating the properties for all successor nodes. Once the ES and EF properties are calculated for all nodes including the final node in the network, the Late Start (LS) and Late Finish (LF) properties can be calculated. These properties are calculated by starting from the final node and progressively calculating the properties for all predecessor nodes. Float can be calculated by comparing the difference between the LS and ES, or the LF and ES properties.

Table 4. Critical Path Method (CPM) Properties

Property	Description	Calculation
Early Start (ES)	The earliest time that this node could begin work. This assumes that all previous activities complete at their earliest possible time.	The maximum EF of all directly preceding nodes. If there are no preceding nodes, the value is zero.
Early Finish (EF)	The earliest time that this node could complete work. This assumes that all previous activities, along with this activity complete at their earliest possible time.	The ES of the node plus the duration of the activity.
Late Start (LS)	The latest time that this activity could start work without driving additional changes to the overall completion schedule. This assumes that the activity of all successor nodes as well as this node completes on time.	The LF of the node minus the duration of the activity.
Late Finish (LF)	The latest time that this activity could complete work without driving additional changes to the overall completion schedule. This assumes that the activity of all successor nodes completes on time.	The minimum LS of all directly succeeding nodes. If there are no directly succeeding nodes, the value is the EF of the node. *
Float	The amount of delay a single node could experience without affecting the overall completion schedule.	The difference between the LF and the EF of the node.

*If a targeted delivery date, or customer promised date exists, the LF of the final node should be set to the delivery date prior to calculating LS, LF, and Float of all other nodes.

Once all node calculations are complete, the path in the network that has the minimal amount of float is the critical path through the network.

In the example shown in Table 5, the WBS#, Op# and Durations are given. No promised date has been provided. The CPM properties described by Table 4 are calculated. The calculated critical path is the path consisting of WBS Elements 11100, 11000, and 10000 (Path #1 in Table 3).

Table 5. CPM Calculations for Example Project

WBS#	Op#	Duration	ES	EF	LS	LF	Float
10000	1	2.0	205.0	207.0	205.0	207.0	0.0
10000	2	51.5	207.0	258.5	207.0	258.5	0.0
10000	3	5.0	258.5	263.5	258.5	263.5	0.0
10000	4	36.0	263.5	299.5	263.5	299.5	0.0
11000	1	90.0	21.5	111.5	21.5	111.5	0.0
11000	2	10.0	111.5	121.5	111.5	121.5	0.0
11000	3	35.0	121.5	156.5	121.5	156.5	0.0
11000	4	40.0	156.5	196.5	156.5	196.5	0.0
11000	5	7.5	196.5	204.0	196.5	204.0	0.0
11000	6	1.0	204.0	205.0	204.0	205.0	0.0
11100	1	1.0	0.0	1.0	0.0	1.0	0.0
11100	2	2.0	1.0	3.0	1.0	3.0	0.0
11100	3	16.5	3.0	19.5	3.0	19.5	0.0
11100	4	2.0	19.5	21.5	19.5	21.5	0.0
11200	1	1.0	0.0	1.0	10.5	11.5	10.5
11200	2	2.0	1.0	3.0	11.5	13.5	10.5
11200	3	6.0	3.0	9.0	13.5	19.5	10.5
11200	4	2.0	9.0	11.0	19.5	21.5	10.5
12000	1	1.0	49.5	50.5	69.0	70.0	19.5
12000	2	120.0	50.5	170.5	70.0	190.0	19.5
12000	3	15.0	170.5	185.5	190.0	205.0	19.5
12100	1	1.0	42.5	43.5	62.0	63.0	19.5
12100	2	5.0	43.5	48.5	63.0	68.0	19.5
12100	3	1.0	48.5	49.5	68.0	69.0	19.5
12110	1	1.0	0.0	1.0	19.5	20.5	19.5
12110	2	17.0	1.0	18.0	20.5	37.5	19.5
12110	3	22.5	18.0	40.5	37.5	60.0	19.5
12110	4	2.0	40.5	42.5	60.0	62.0	19.5
13000	1	1.0	17.0	18.0	187.0	188.0	170.0
13000	2	16.0	18.0	34.0	188.0	204.0	170.0
13000	3	1.0	34.0	35.0	204.0	205.0	170.0
13100	1	1.0	0.0	1.0	170.0	171.0	170.0
13100	2	13.0	1.0	14.0	171.0	184.0	170.0
13100	3	3.0	14.0	17.0	184.0	187.0	170.0
13200	1	1.0	0.0	1.0	179.0	180.0	179.0
13200	2	6.0	1.0	7.0	180.0	186.0	179.0
13200	3	1.0	7.0	8.0	186.0	187.0	179.0

3. Calculating Early, Late, and Mid Hours for CPM

Once the CPM properties have been calculated, an assessment can be performed on the availability of float for the duration of the project. The total duration of the project (p_d) is the EF/LF of the last activity node in the network. In the example shown in Table 5, it is 299.5 hours from the start of the project (WBS# 10000, Op# 4).

For any time (t) between 0 and p_d , values for Early Hours and Late Hours (EH_t and LH_t) can be calculated. Early Hours is the number of billable hours that could possibly be executed before time, t , if all activities can start immediately when their predecessor activities have been completed. Late Hours is the number of billable hours that must be executed before time, t , to prevent slippage of the overall schedule.

$n := \{\text{All Nodes In Activity Network Diagram}\}$

$n_a \subset n \mid \text{active}$

$$EH_t = \sum_{i \in n_a} (d_i * (t \geq EF_i)) + ((d_i - (EF_i - t)) * (t \geq ES_i) * (t < EF_i)) \quad \forall t \in [0, p_d]$$

$$LH_t = \sum_{i \in n_a} (d_i * (t \geq LF_i)) + ((d_i - (LF_i - t)) * (t \geq LS_i) * (t < LF_i)) \quad \forall t \in [0, p_d]$$

The equations above assume that the rate of work being performed for any given node (i) in the set of nodes in the network (n) is linearly distributed throughout the duration of the activity.

Another term, Mid Hours (MH_t) is a useful parameter defined as the average of EH_t and LH_t for any valid time, t . Given some flexibility in scheduling non-critical path activities, MH_t serves as an indicator of an approximate discrete baseline schedule.

Plotting EH, LH and MH values over time can be useful during the execution phase of a project. Top-level execution metrics, such as Schedule Variance, refers to a firm BCWS which does not consider the variation that can occur with non-critical path activities without affecting the overall completion date. For projects that have a higher level of concurrent activity across the WBS, the magnitude of float can become much greater. The float should be considered in any top-level analysis. It is important for program managers

to understand the trade space that exists in the schedule and float can be representative of the internal flexibilities that exist within the activity network without compromising the critical path or the overall delivery schedule. Calculating the EH, LH, and MH values for the example in Table 5 results in the chart shown in Figure 17.

This chart describes the cost incurred throughout the period of performance if all activities were performed without variation from the plan. The Early Hours series (blue line) is the expected execution if all activities were started as soon as its predecessor's had been completed. The Late Hours series (orange line) is the execution if all activities were started just in time to meet the earliest achievable schedule. The difference between these two series is the available float in the system at any given time. As the project starts, there is some room in the schedule for some activities to be delayed without affecting the overall delivery date. As the project nears completion, and the series converges into a single line, no float remains in the system and delays to activities will result in a delay to the delivery.

In practice, it is expected that some activities will be delayed for many possible reasons such as conflicting priorities with other efforts that utilize the same resources, unplanned material shortages, unplanned labor shortages. A successful project execution will track somewhere between the Early and Late series, so the Mid Hours series (black line) is useful to gauge the consumption of float in execution.

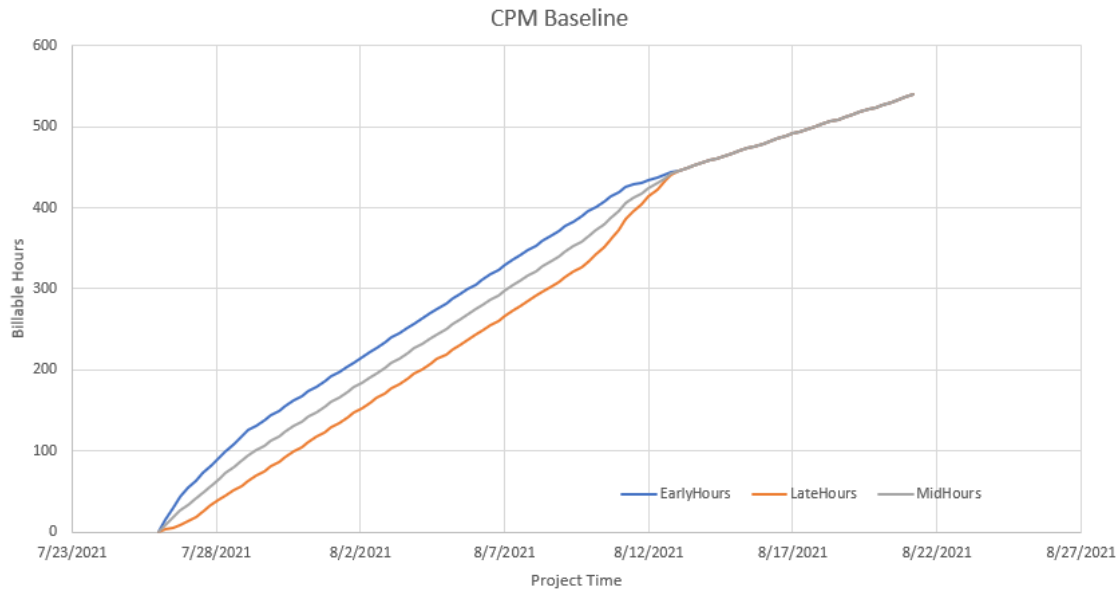


Figure 17. CPM Early, Late, and Mid Hours over Time for Example Project

If a target delivery date is provided that is beyond the calculated EF of a project, a positive float will be present for all activities in the project. The difference between the calculated EF and the target delivery date may be considered “Schedule Management Reserve.” In Figure 18, the CPM properties were calculated including a targeted delivery date beyond the calculated EF. The initial flat section of Late Hours from T=7/26 through 7/31 represents the initial delay allowable before the targeted delivery date is unattainable.

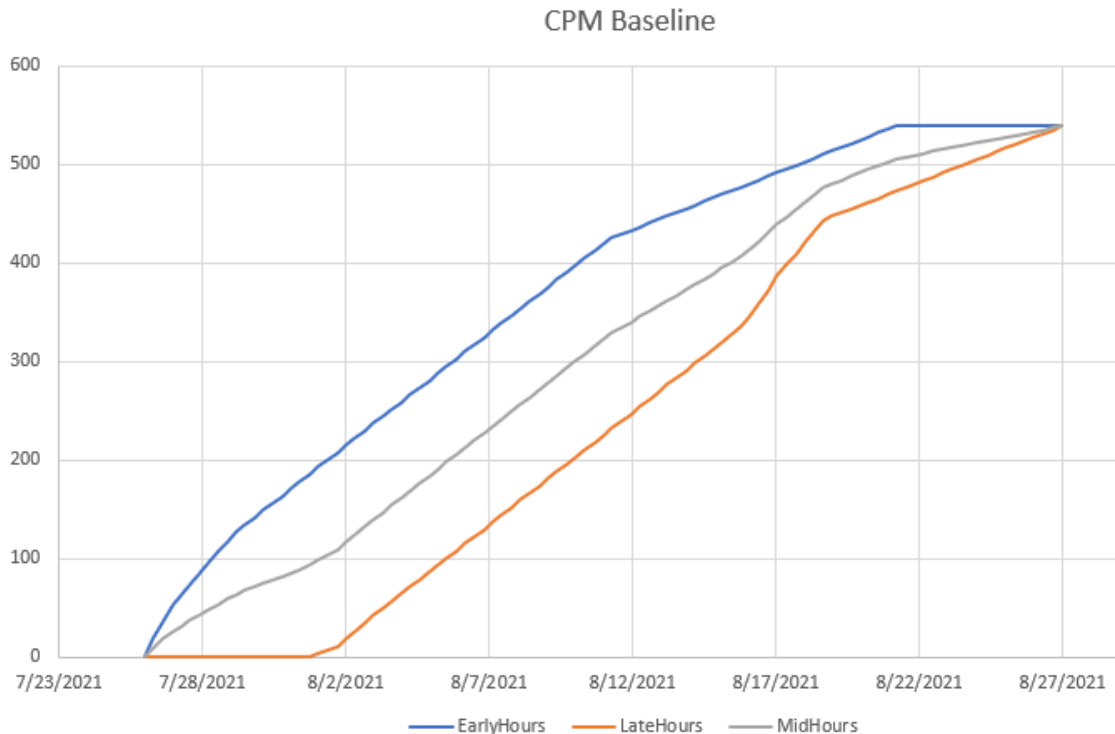


Figure 18. CPM Early, Late, and Mid Hours over Time for Example Project with Overall Schedule Float

If a targeted delivery date is provided that is prior to the calculated EF of a project, negative float will be present for at least some of the activities in the project. The difference between the calculated EF and the targeted delivery date may be considered an available schedule shortfall. In Figure 19, the CPM properties were calculated including a targeted delivery date prior to the calculated EF. The initial data point for Late Hours represents the minimum number of billable hours that should have been completed by this time to achieve the targeted delivery date. The horizontal section of Late Hours from T=8/19 to 8/21 represents the period that the project will be overdue if all goes as expected.

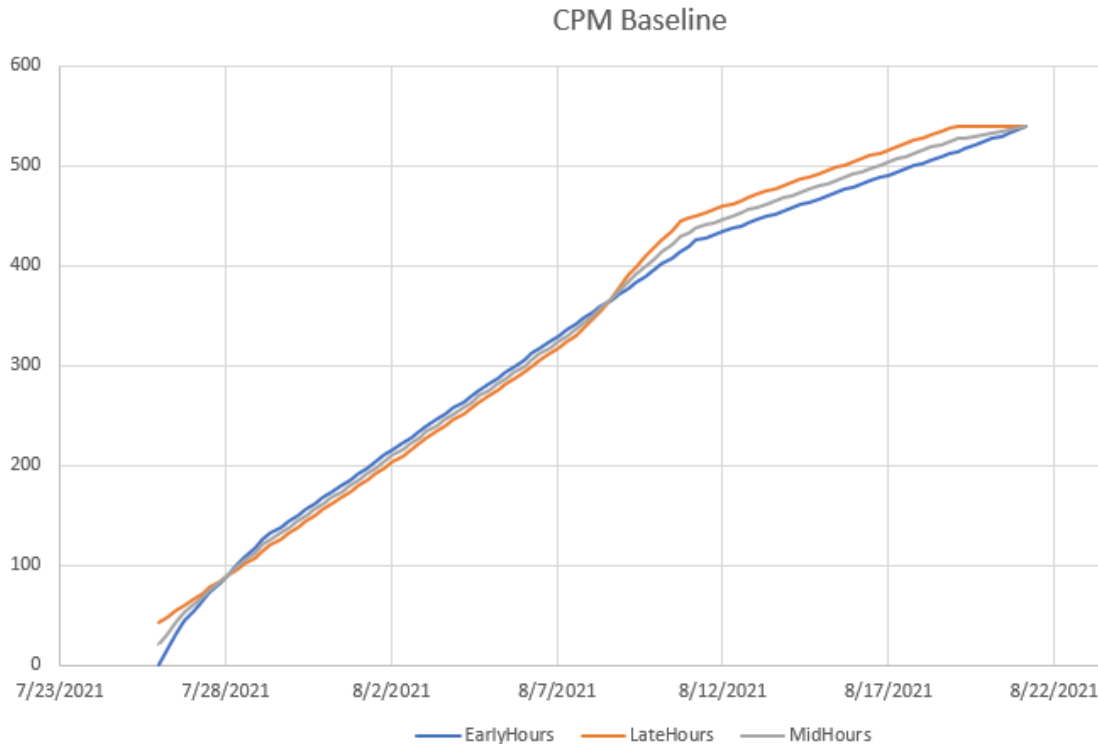


Figure 19. CPM Early, Late and, Mid Hours over Time for Example Project with Overall Schedule Shortfall

E. REVISING CPM ESTIMATES WITH TIME-SERIES ACTUAL COSTS

Once a project begins, variances from the baseline plan should be expected. As actual cost data ($a_{i,t}$) becomes available, CPM calculations can be recomputed by first calculating the remaining unexecuted hours for each activity within the activity network. The remaining unexecuted hours for each activity (r_i) is the original planned duration (d_i) less the sum of all actual entries for that activity. In the absence of additional information, such as percentage completion, where actual costs exceed the planned duration, the remaining unexecuted hours take a value of zero.

Additionally, if any of an activity's successor activities have accrued any actual costs (s_i), we consider the activity to be complete and set the remaining unexecuted hours to zero. This is applicable to the special case where materials will travel from one work center to another, preventing concurrent execution on multiple nodes of the path. This

calculation could be substituted with an adequate indication of when activities have been completed.

$$m \subset n | \text{succeeds } i$$

$$s_i = \sum_{i \in m} \sum_{t \in T} a_{i,t} \quad \forall i \in n$$

$$r_i = \max \left\{ \left(d_i - \sum_{t \in T} a_{i,t} \right) * (s_i = 0), 0 \right\} \quad \forall i \in n$$

In the example shown in Figure 20, after about one week of execution, the revised CPM properties show some project execution delay along with unplanned variation. The earliest completion date for the project has shifted from 8/21 to 8/29. It is also apparent that some of the activities associated with this project were completed in less time than was originally expected as the projected overall cost has been reduced. The expected total cost has shifted from 539 to approximately 517 hours.

Figure 21 extends the example shown in Figure 20 to a later point in the project's execution timeline. At this stage, the CPM properties show that the earliest completion date for the project is now 9/20. The actual hours shown versus time indicate that there were periods of unplanned delay which likely had a cumulative influence on the overall schedule.

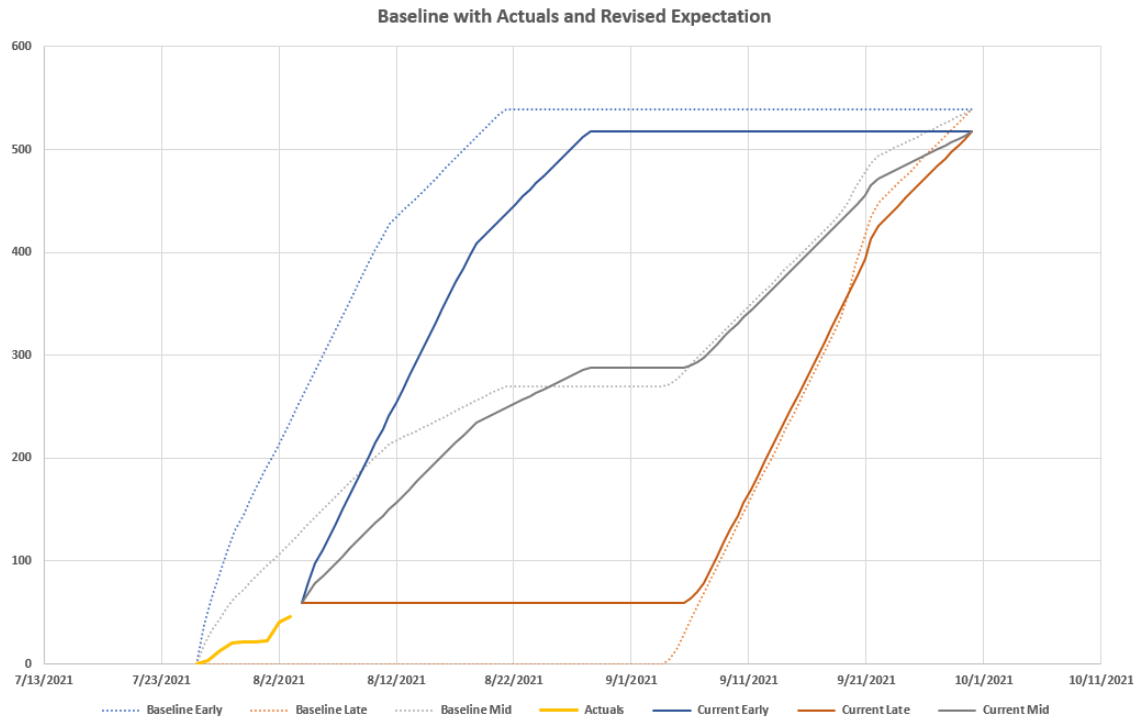


Figure 20. CPM Early, Late, and Mid Hours for Example Project with Revised Expectations Based on Actual Costs at T=08/06/2021

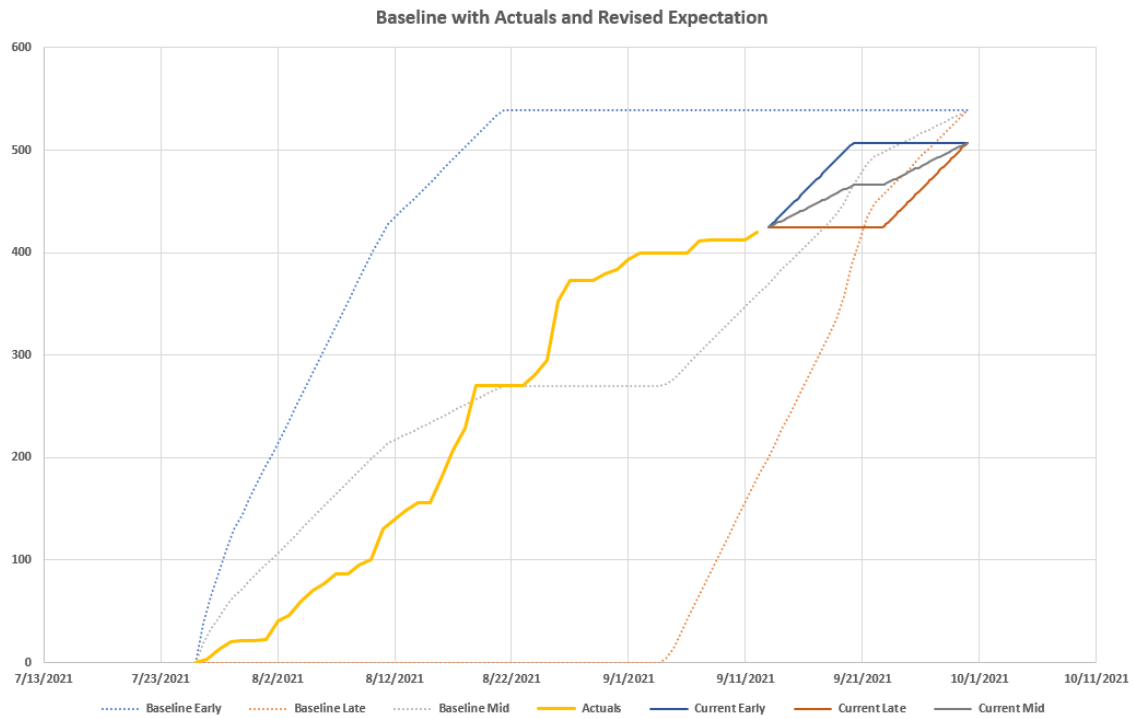


Figure 21. CPM Early, Late, and Mid Hours for Example Project with Revised Expectations Based on Actual Costs at T=09/13/2021

V. USE OF EARNED VALUE MANAGEMENT

A. DEVELOPMENT OF EARNED VALUE MANAGEMENT CHART

Modern Earned Value Management (EVM) charts display time-series data for Budgeted Cost of Work Scheduled (BCWS), Actual Cost of Work Performed (ACWP), and Budgeted Cost of Work Performed (BCWP). The charts also display projected values of ACWP and BCWP for the remaining duration of the project.

BCWS at time (t), can be calculated as the cumulative cost of all activities that are scheduled to be completed prior to or at time (t). Without a prevailing higher-level schedule of events, the Mid Hours (see Section IV.D.3) can be used as an approximation of the baseline schedule.

$$BCWS_t = \frac{EH_t + LH_t}{2} \quad \forall t \in [0, p_d]$$

ACWP at time (t), can be calculated as the cumulative actual cost that is incurred over all activities prior to or at time (t).

$$ACWP_t = \sum_{i \in n, x \in [0, t]} a_{i,x} \quad \forall t \in [0, t_{now}]$$

BCWP at time (t), can be calculated as the cumulative expected costs of all activities that have been completed prior to or at time (t). This metric will lag the ACWP metric, as additional value of partially completed work is not included until the activity has been completed. This metric could be improved with a measurement of “% complete” to provide insight into how much of the budgeted cost has been performed.

$$BCWP_t = \sum_{i \in n, x \in [0, t]} a_{i,x} * (s_i > 0) \quad \forall t \in [0, t_{now}]$$

The projected values of ACWP and BCWP can be calculated by using the revised estimates for MH computed at t_{now} .

$$ACWP_t = ACWP_{t_{now}} + MH_t \forall t \in [t_{now}, p_d]$$

$$BCWP_t = BCWP_{t_{now}} + MH_t \forall t \in [t_{now}, p_d]$$

Utilizing all the calculated metrics, a chart like the chart found on the Earned Value Management Gold Card can be generated. Figure 22 shows the calculated EVM chart for the example project, calculated at $t=9/13$. At this point in the project, it appears that the effort is ahead of schedule as the BCWP is greater than the BCWS. To this point, the project is also enjoying a favorable cost variance.

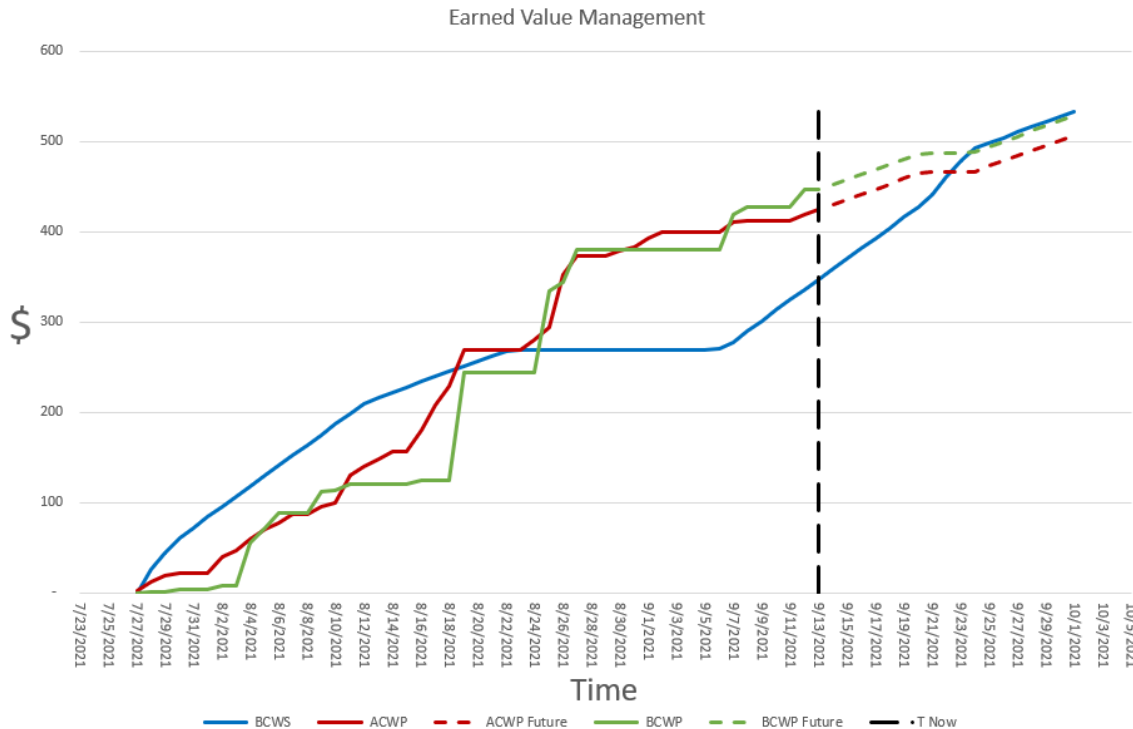


Figure 22. Earned Value Management Chart for Example Project at T=09/13/2021

B. CALCULATING EVM VARIANCES AND EFFICIENCIES

With this information, EVM variances can also be calculated. Cost Variance (CV) is the difference between BCWP and ACWP at time (t). Expressed as a percentage, CV% is the CV over BCWP at time (t). Schedule Variance (SV) is the difference between BCWP and BCWS at time (t). SV% is the SV over BCWS at time (t).

$$CV_t = BCWP_t - ACWP_t, CV\%_t = \frac{CV_t}{BCWP_t}$$
$$SV_t = BCWP_t - BCWS_t, SV\%_t = \frac{SV_t}{BCWS_t}$$

CV is a strong metric that objectively shows the difference in actual costs from planned costs. However, there are some limitations with the effective use of SV in the scenario of Job-Shop Manufacturing. Considering that the total activity network is comprised of many activities that contain float, deviation from a planned schedule does not necessarily indicate that there has been a change to critical path. Additionally, both CV and SV are subject to extreme early values when BCWP and BCWS are initially small therefore, these values are not good early indicators of project performance.

Figure 23 shows the CV% and SV% variances over time for the example project. At $t=9/13$, these variances are 5% and 28% respectively. The projection for CV beyond $t=9/13$ stays consistent, as future values of BCWP and ACWP will increase at the same rate. The projection for SV beyond $t=9/13$ will generally trend towards zero (0), as re-baselined calculations for MH calculate future work performed to be distributed based on an average of best case (working the earliest as possible) and worst case (working the latest as possible without affecting the overall delivery).

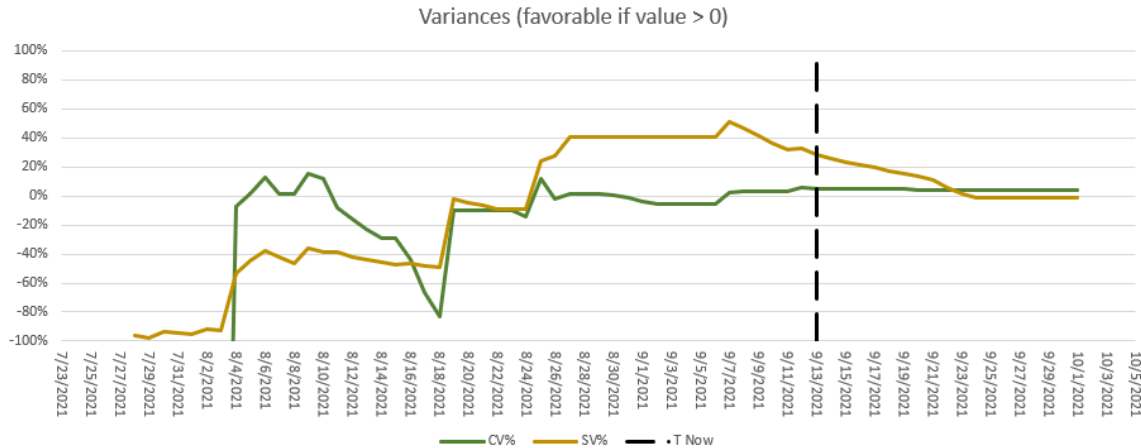


Figure 23. Cost and Schedule Variances (CV% & SV%) over Time for Example Project at T=09/13/2021

EVM Efficiencies can also be calculated. Cost Efficiency (CPI) at time (t) is the ratio of BCWP to ACWP at time t . Schedule Efficiency (SPI) at time (t) is the ratio of BCWP to BCWS at time t . The SPI efficiency is subject to the same limitations of SV in that the BCWS flexibilities are not fully captured in this metric.

$$CPI_t = \frac{BCWP_t}{ACWP_t}, \quad SPI_t = \frac{BCWP_t}{BCWS_t}$$

In Figure 24, CPI and SPI are shown over time for the example project. Like CV, the projection of CPI stays consistent as future values of BCWP and ACWP increase at the same rate. The future value of SPI will drive towards one (1) as the BCWP and BCWS should be equal at the end of the project.

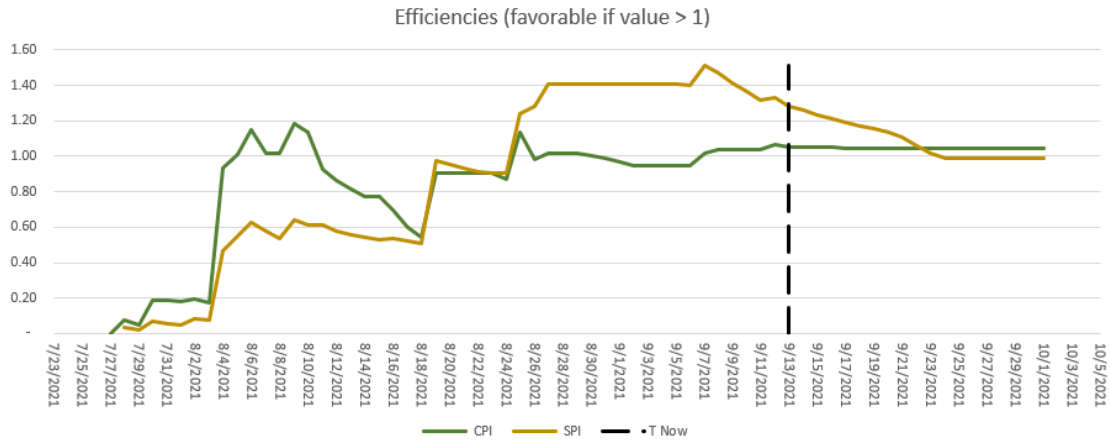


Figure 24. Cost and Schedule Efficiencies (CPI & SPI) over Time for Example Project at T=09/13/2021

THIS PAGE INTENTIONALLY LEFT BLANK

VI. DISCRETE PROBABILISTIC ESTIMATION FOR MANUFACTURING OPERATIONS

A. IDENTIFYING PLANNED VARIANCES IN ACTIVITIES

In job-shop manufacturing operations, the actual duration of any individual activity is influenced by a variety of factors. Some factors include varying artisan skill level, availability of equipment, yield rate, activity complexity, and make quantity. It is extremely difficult to anticipate and account for all the factors that contribute to the actual duration of an activity. To better communicate the influence of these factors, estimates can be reformed from traditional straight expectation values to a probabilistic value. Using random variables with different properties, anticipated variances can be further communicated, rather than assumed as an estimate's expected error.

A probability density function (PDF) shows the distribution of outcomes of a random variable. In the example shown in Figure 25, the random variable can take a value anywhere from 10 to 20. The most likely outcome, or the mode, of the random variable is 17 and the average value, or the mean, of the random variable is 15.7.

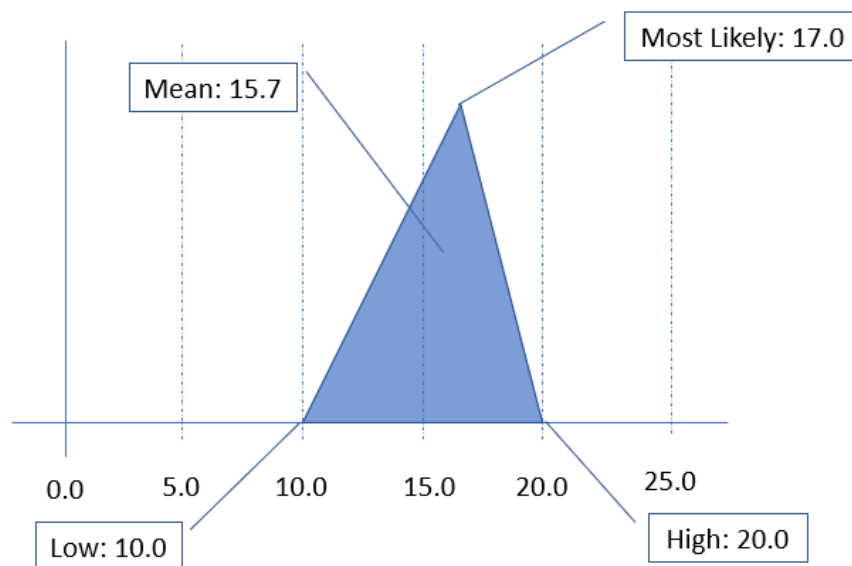


Figure 25. Probability Density Function of the Triangular Distribution

Communication of estimates leveraging a triangular distribution imparts additional knowledge to others about the expected variances that could occur. Consider a saw cutting operation that may be originally planned with 17 hours of labor as a straight expectation. By using a triangular distribution with a low, most likely, and high values of 10, 17, and 20 hours respectively, the expected value can be maintained while more clearly indicating the anticipated variance. With this example, as shown in Figure 26, 50% of the time, the random variable will take a value between 14.2 and 17.3. 75% of the time, the random variable will take a value between 13.0 and 18.1.

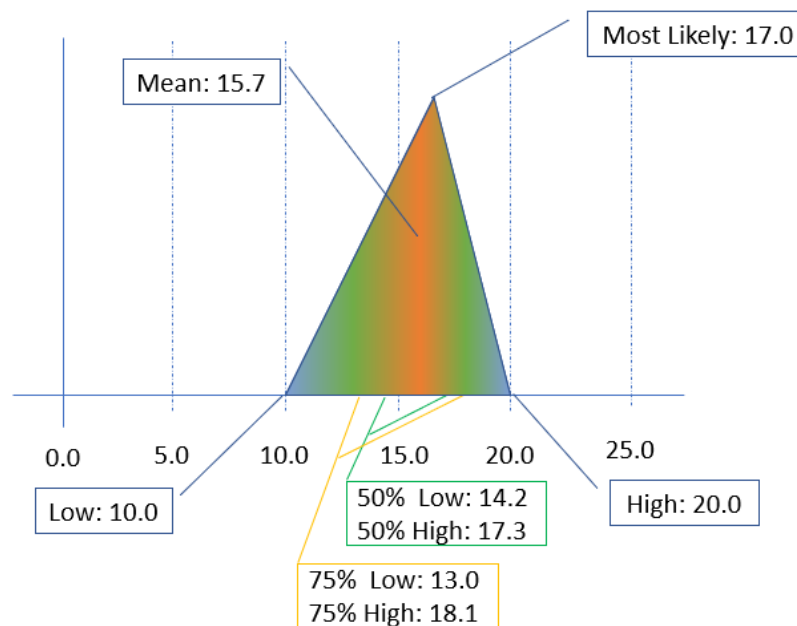


Figure 26. Probability Density Function of the Triangular Distribution with 50% and 75% Intervals

B. ADJUSTING SME INPUTS TO ACCOUNT FOR UNCERTAINTY

The Joint Agency Cost Schedule Risk and Uncertainty Handbook (2014) identifies six (6) uncertainty distributions for analysts to use when considering probabilistic durations of activities (see Table 1). Depending on the information known about a specific operation, one of these distributions can be chosen to model the expected variation in the activity.

Estimation for job-shop manufacturing typically consists of many operations that have limited execution history. In those cases, SMEs will provide subjective estimates based on an assessment of the part requirements along with considerations for available equipment and personnel available. Elicitation of estimates from these SMEs is most easily performed using lower, upper and most likely values (NCCA 2014, sec. 2.5.1) resulting in a triangular distribution. However, JA CSRUH (2014) continues to say “Unless there is evidence to do otherwise, request and interpret subjective bounds (expert opinion) as the 70% range and adjust for skew when using triangular, uniform or betaPERT [distributions]” (NCCA 2014, sec. 2.5.4). To amend the subjective bounds of a triangular distribution, first the skew is calculated from the subjective inputs.

$$\text{Skew} = \frac{\text{Most Likely} - \text{Low}}{\text{High} - \text{Low}}$$

The amended max value can be calculated as

$$\text{Max} = \text{High} + \frac{((\text{High} - \text{Mode}) * \sqrt{1 - \text{Uncert by expert}})}{1 - \sqrt{1 - \text{Uncert by expert}}},$$

where the uncertainty captured by the expert “Uncert by expert” is equal to 70%. The amended min value can then be calculated as

$$\text{Min} = \frac{\text{Mode} - (\text{Skew} * \text{Max})}{1 - \text{Skew}}.$$

As a result, the triangular distribution shown in Figure 26 can be amended as shown in Figure 27. The amended values are what should be used in further analysis unless there is evidence supporting a higher level of certainty from the expert.

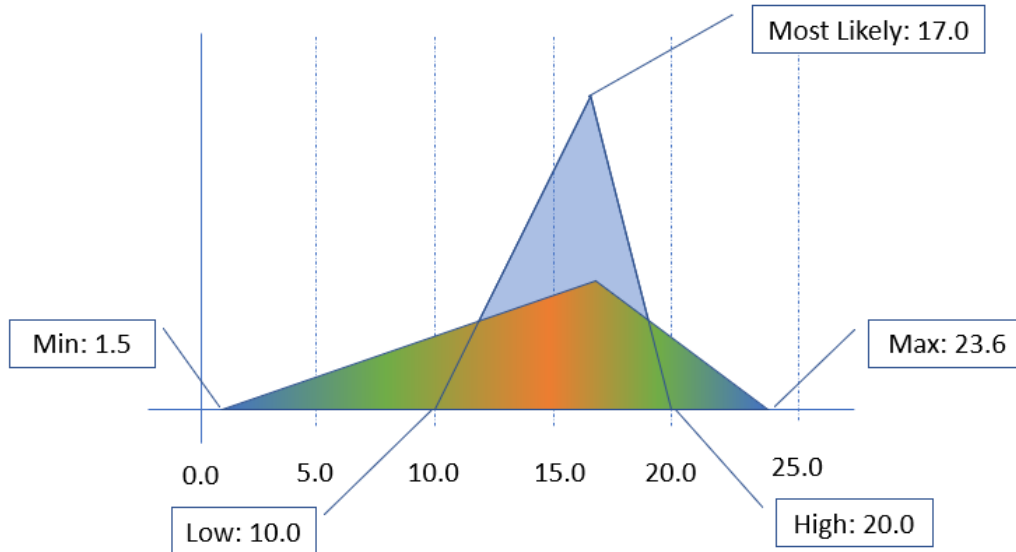


Figure 27. Amended Probability Distribution Function of Triangular Distribution Accounting for Uncertainty

C. UTILIZING PROBABILISTIC ESTIMATES TO CALCULATE EXPECTED OUTCOME

The expected outcome of an activity network is the average cost and schedule that would occur if the scenario were to be run an infinite amount of times. Each operation in the WBS has an expected value and variance as shown in Table 6 based on its applied probability distribution.

Table 6. Expected Value and Variance for Distributions.
Adapted from Hayter (2013)

Distribution	Parameters	Expected Value	Variance
Lognormal	Mean (μ), Variance (σ^2)	$e^{\mu + \frac{\sigma^2}{2}}$	$e^{2\mu + \sigma^2}(e^{\sigma^2} - 1)$
Triangular	Min (a), Max (b), Most Likely (c)	$\frac{a + b + c}{3}$	$\frac{a^2 + b^2 + c^2 - ab - ac - bc}{18}$
Beta Pert	Min (a), Max (b), Most Likely (c)	$\frac{a + 4c + b}{6}$	$\left(\frac{b - a}{6}\right)^2$
Beta	Min (a), Low (α), High (β), Max (b)	$\frac{\alpha}{\alpha + \beta} * (b - a)$	$\frac{\alpha\beta}{(\alpha + \beta)^2 * (\alpha + \beta + 1)}$
Normal	Mean (μ), Variance (σ^2)	μ	σ^2
Uniform	Min (a), Max (b)	$\frac{a + b}{2}$	$\frac{(b - a)^2}{12}$

The expected value can be treated as a reasonable single point estimate to perform critical path calculations and determine an estimated cost and schedule for a project. These estimates are likely to be less susceptible to SME biases as it allows the SME to communicate anticipated variation that would otherwise need to be accounted for by the SME in their single point estimate. An expected cost and schedule derived in this way cannot include cumulative variances and, as a result, does not provide decision makers with enough information to quantify the uncertainty of an overall estimate.

D. CRITICAL PATH OF CUMULATIVE SCHEDULE WITH PLANNED VARIANCES

Depending on the total expected value and variance of each branch of an activity network, the overall critical path of a project may vary during execution. Consider a

scenario where some path, “Path A” is expected to take 16 hours but could take 5-hours more or less, and “Path B” is expected to take 14 hours but could also take 5-hours more or less. Initially, “Path A” is expected to be on the critical path. If during actual execution, “Path A” takes 13 hours and “Path B” takes 17 hours, then “Path B” is the critical path. This relatively simple case of cumulative non-linear probabilistic outcomes drives the need for the use of Monte Carlo simulation as a method to further quantify cumulative variance.

To perform Monte Carlo simulation, a value for the number of trials is initially set (n). JA CSRUH (2014) recommends a default number of trials set to 10,000 but the number of trials should be tested for convergency to verify that it produces a stable result (NCCA 2014, sec. 3.5.1). For each trial, the duration of each activity within the activity network diagram is randomly assigned based on a linear transformation (Table 7) of a uniformly random variable (r_u) between 0 and 1.

Table 7. Trial Durations for Various Distributions

Distribution	Trail Duration
Straight Expectation	$D_t = Exp_{in}$ Where Exp_{in} is the expected value inputted by the user.
Lognormal	$D_t = \text{LogNorm_Inv}\left(r_u, \mu_{in}, \sqrt{\sigma_{in}^2}\right)$ Where μ_{in} and σ_{in}^2 are the mean and variance inputted by the user.
Triangular	$D_t = \begin{cases} L_{in} & \text{for } L_{in} = U_{in} \\ L_{in} + \sqrt{r_u(U_{in} - L_{in})(M_{in} - L_{in})} & \text{for } r_u \leq \frac{M_{in} - L_{in}}{U_{in} - L_{in}} \\ U_{in} - \sqrt{(1 - r_u)(U_{in} - L_{in})(-M_{in} + U_{in})} & \text{for else} \end{cases}$ Where L_{in}, M_{in}, U_{in} are the Low, Most Likely, and Upper values inputted by the user.
BetaPert	$D_t = \text{Beta_Inv}(r_u, \alpha, \beta, L_{in}, U_{in})$ Where $\alpha = \left(\frac{2(U_{in} + 4 * M_{in} - 5 * L_{in})}{3(U_{in} - L_{in})}\right) * \left[1 + \frac{4(M_{in} - L_{in})(U_{in} - M_{in})}{(U_{in} - L_{in})^2}\right]$, and $\beta = \left(\frac{2(5 * U_{in} - 4 * M_{in} - L_{in})}{3(U_{in} - L_{in})}\right) * \left[1 + \frac{4(M_{in} - L_{in})(U_{in} - M_{in})}{(U_{in} - L_{in})^2}\right]$, and L_{in}, M_{in}, U_{in} are the Low, Most Likely, and Upper values inputted by the user.
Beta	$D_t = \text{Beta_Inv}(r_u, \alpha_{in}, \beta_{in}, L_{in}, U_{in})$ Where $\alpha_{in}, \beta_{in}, L_{in}, U_{in}$ are the Alpha, Beta, Low and Upper values inputted by the user.
Normal	$D_t = \text{Norm_Inv}\left(r_u, \mu_{in}, \sqrt{\sigma_{in}^2}\right)$ Where μ_{in}, σ_{in}^2 are the mean and variance inputted by the user.
Uniform	$D_t = (L_{in} * r_u) * (U_{in} - L_{in})$ Where L_{in}, U_{in} are the Low and Upper values inputted by the user.

Once all activity durations are established, the remaining time is calculated for each activity as described in Chapter IV.E “Revising CPM Estimates with Time-Series Actual Costs.” With the remaining time for each activity, the CPM can be employed to calculate the full duration of the project.

Figure 28 shows histograms of outcomes for schedule and cost at the project’s baseline for $n = 5000$ trials. Figure 29 reflects the same data utilizing a Cumulative Distribution Function (CDF). In the CDF, a value of (9/29/2021, 58%) means that given the planned variances, 58% of the time, the final delivery would occur prior to or equal to 9/29/2021.

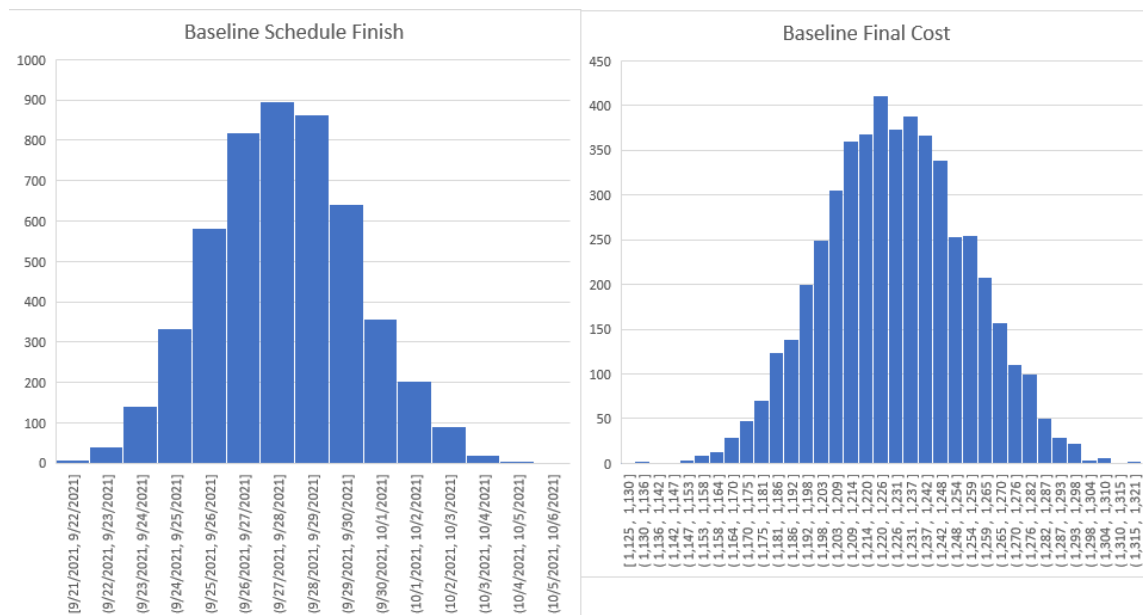


Figure 28. Histogram of Baseline Schedule and Cost Outcomes from Monte Carlo Simulation

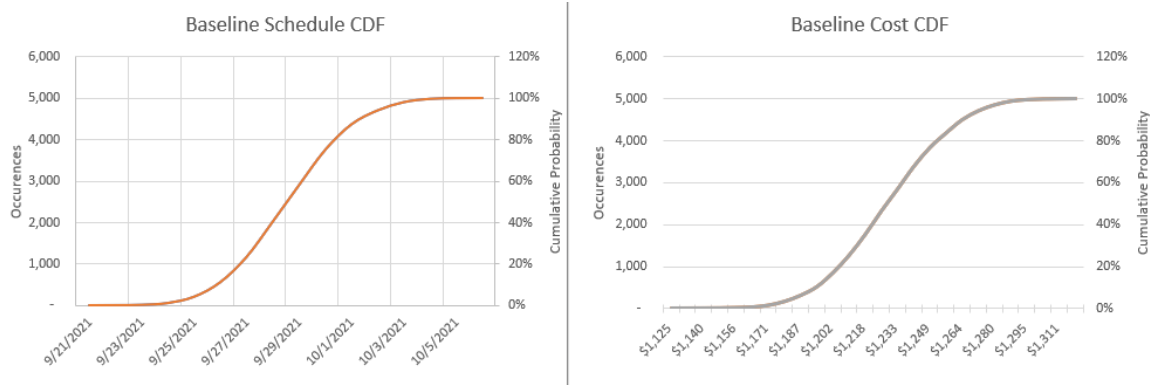


Figure 29. Cumulative Distribution Function (CDF) of Baseline Schedule and Cost Outcomes from Monte Carlo Simulation

It is reasonable to expect that some correlation exists between schedule and cost, where if costs were higher than the expected value, the schedule may respond similarly. This can be evaluated by comparing the outcomes for cost and schedule as a two-dimensional grid, where each section of the grid represents the frequency of outcomes at the intersection.

Figure 30 and Figure 31 shows the 2D-Histogram and 3D-Histogram for the $n = 5000$ trials shown in Figure 28 and Figure 29. In this example, the correlation between cost and schedule is 0.886, which is a very strong correlation indicator. This correlation coefficient is calculated using Pearson's Correlation Coefficient (r_{xy}).

$$r_{xy} = \frac{\sum_{i=1}^n (y_i - \bar{y})(x_i - \bar{x})}{\sqrt{\sum_{i=1}^n (y_i - \bar{y})^2} \sqrt{\sum_{i=1}^n (x_i - \bar{x})^2}}$$

where x_i, y_i are the individual trial data points and $\bar{x}, \bar{y}$ are the average of each set.

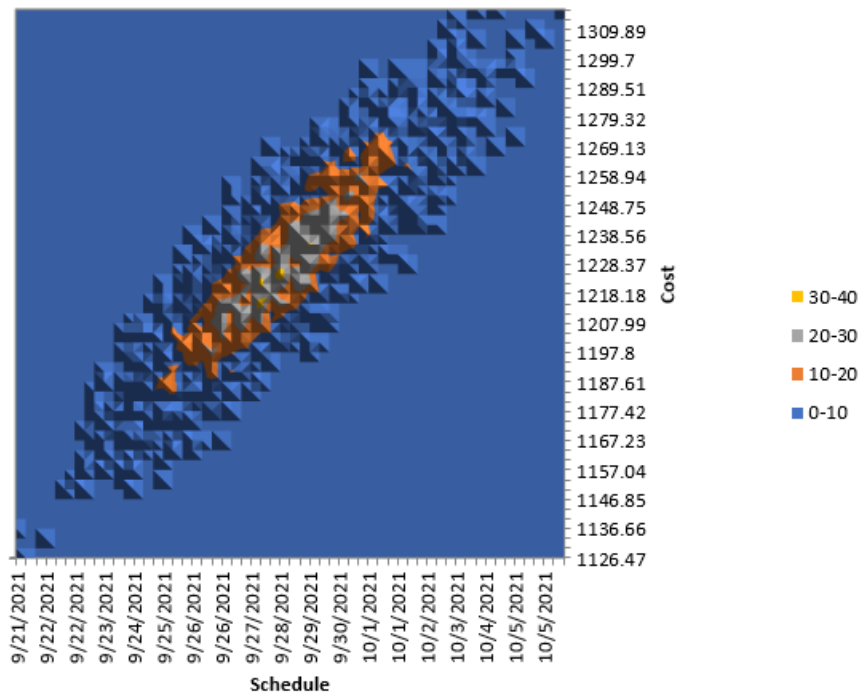


Figure 30. Two-Dimensional Histogram of Cost and Schedule from Monte Carlo Simulation

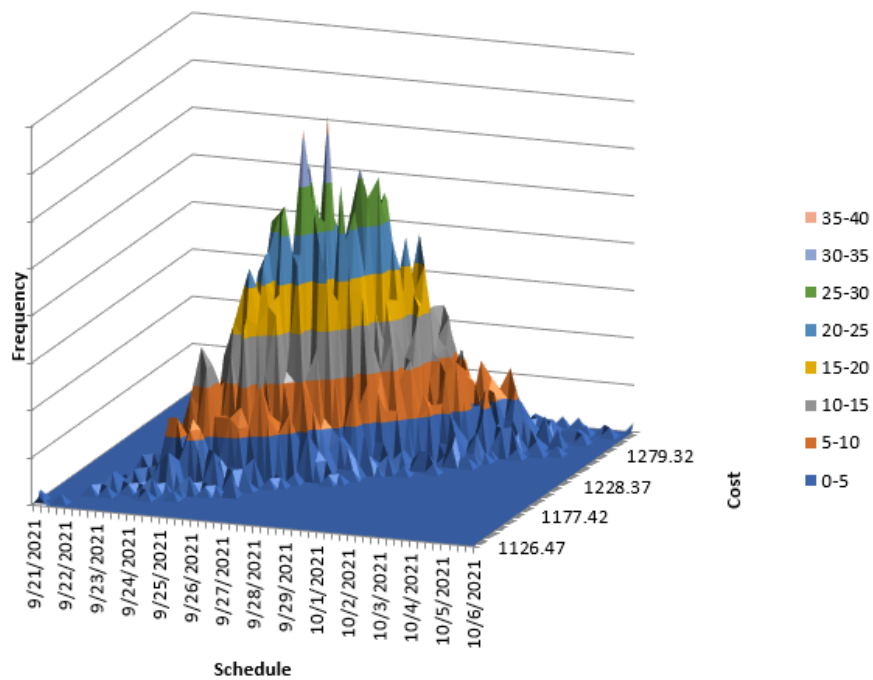


Figure 31. Three-Dimensional Histogram of Cost and Schedule from Monte Carlo Simulation

This information can also be expressed as a Joint Cumulative Distribution Function as shown in X. In this figure, the Z-height represents the probability that cost and schedule would be less than or equal to the X-Y coordinate.

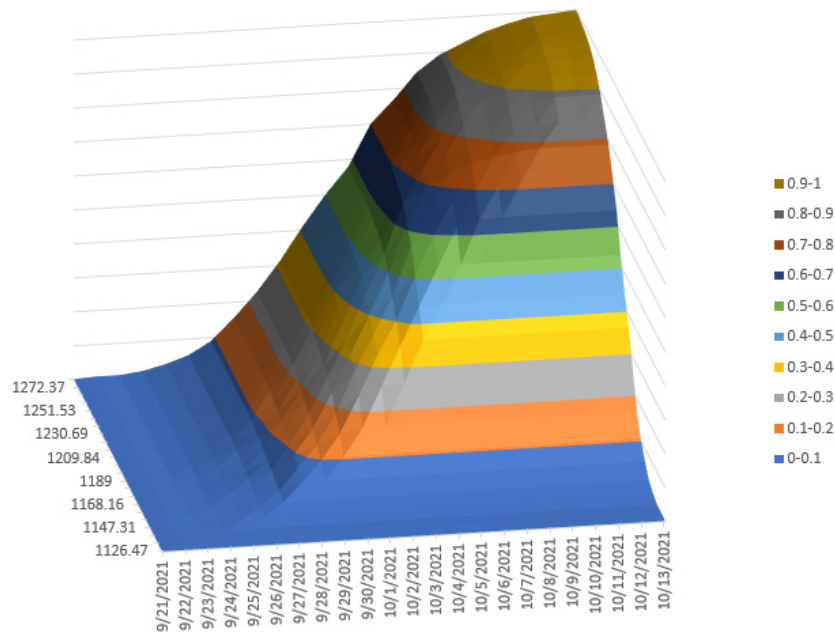


Figure 32. Three-Dimensional CDF of Cost and Schedule from Monte Carlo Simulation

In this project, variations in duration of an individual activity result in a direct impact to the cumulative cost. However, variation in the duration of an activity does not necessarily drive a change to the critical path of a project. The inherent schedule slack on some paths through the activity network results in the correlation coefficient taking values less than one (1).

As the total float of the network reduces to zero (0) over time, the execution projection's correlation will increase and approach a value of one (1) when all remaining activities are on the critical path. At that point, changes to activity durations will have a direct impact, or "day for day slip" on the overall project schedule. Figure 33 shows the revised projections of cost and schedule outcomes at a point in time where all remaining

activities fall on the critical path. In this case, the correlation between cost and schedule is 0.9998.

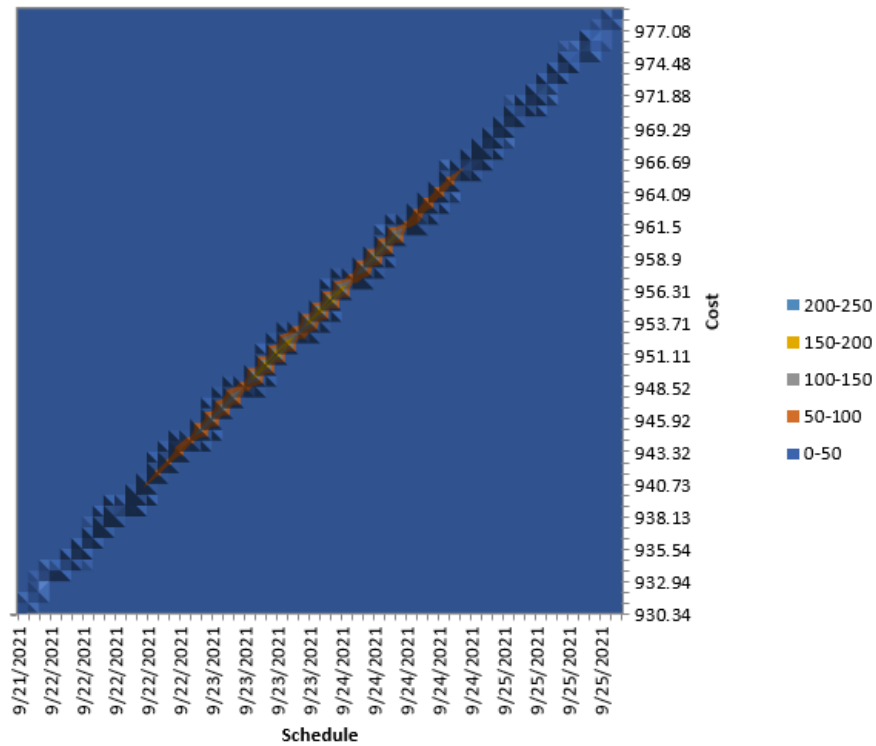


Figure 33. Two-Dimensional Histogram of Cost and Schedule from Monte Carlo Simulation where Remaining Activities are on the Critical Path

VII. COMPARISON BETWEEN EARNED VALUE MANAGEMENT AND PROBABILISTIC MONTE CARLO SIMULATION

In this chapter, the methods described in Chapters IV through VI will be used to track the progress of a project that was executed by the Prototype and Manufacturing Division starting in November of 2022 with an expected delivery date of 3/31/2023. Earned Value Management reporting methods and Probabilistic Monte Carlo Simulation methods will be compared at various points during project execution.

A. PMD PROJECT AT T=11/04/2022 (PROJECT START)

The PMD Project starts on T=11/04/2022. Based on a bottoms-up estimation technique and if all goes according to plan, the project is expected to require 1,582 billable hours of labor. Based on customer negotiations, the schedule completion date is set to 03/31/2023 and the budget has been set to 1740 billable hours. At this stage, the EVM chart in Figure 34 shows the BCWS along with projections for ACWP and BCWP with no variation.

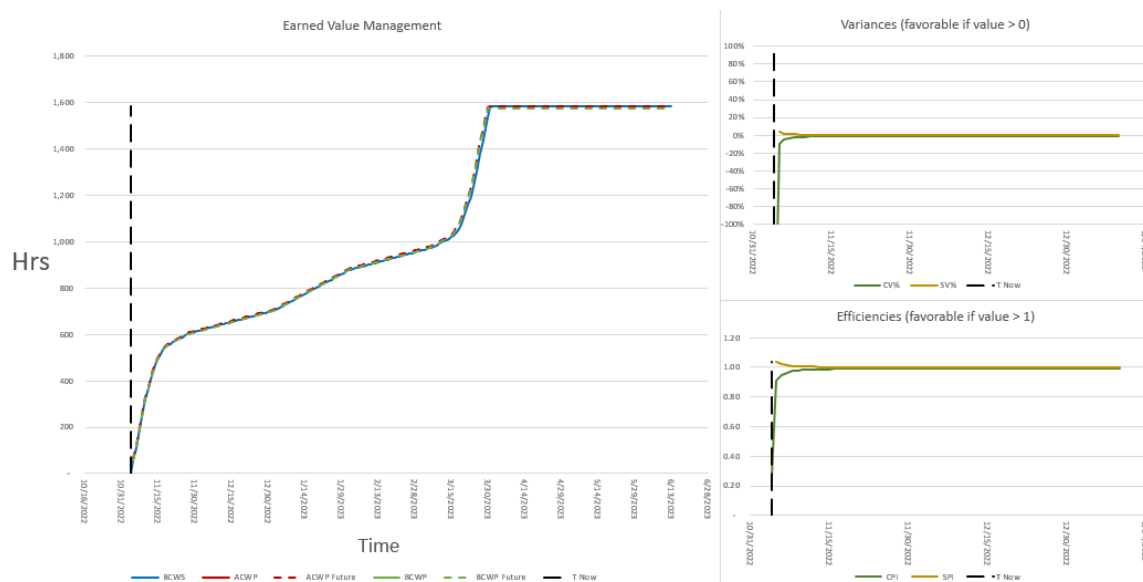


Figure 34. EVM Charts of PMD Project at T=11/04/2022

The CPM Chart and Trial Results in Figure 35 also shows the expected billable hours of labor and the targeted schedule of 3/31/2023. The result of 5000 trials, factoring in the low-level probabilistic estimates, shows that the simulated likelihood of the schedule being completed prior to or at 3/31/2023 is near 100% and executing within the budget of 1,582 hours is around 50%. The likelihood of the budget falling within the negotiated cost of 1,740 hours is around 91%.

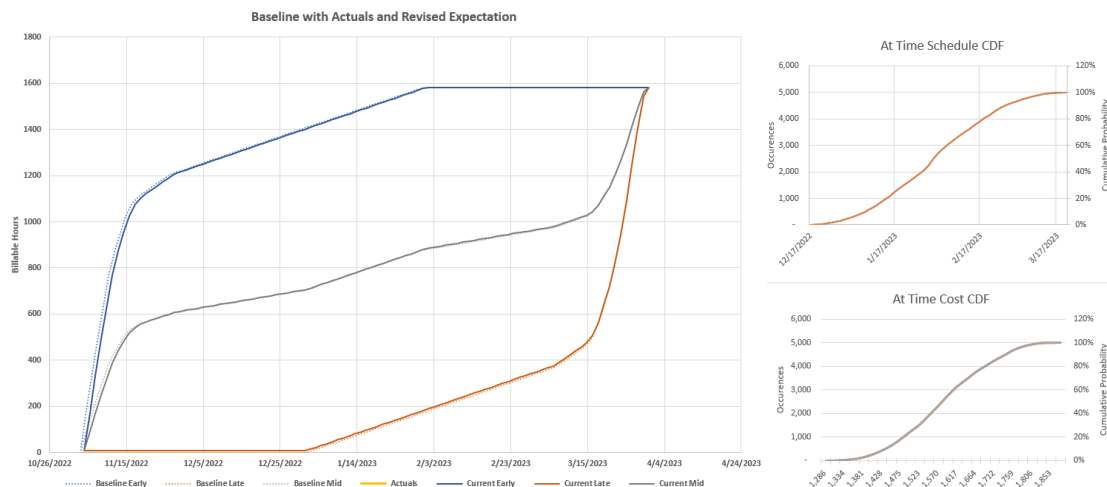


Figure 35. CPM and Trial Results of PMD Project at T=11/04/2022

B. PMD PROJECT AT T=12/01/2022

At T=12/01/2022, the project appears to be working at a slower-than-planned rate. In Figure 36, cost variance is around -73% while schedule variance is around -74%. The magnitude of these numbers should be expected at this stage of a poor performing project as the CV% and SV% are proportional to the relatively low budgeted costs at the time of the metric calculation. At this stage of the project, the EVM chart still predicts a scheduled delivery date of 3/31/2023 as the original schedule consisted of some overall float.

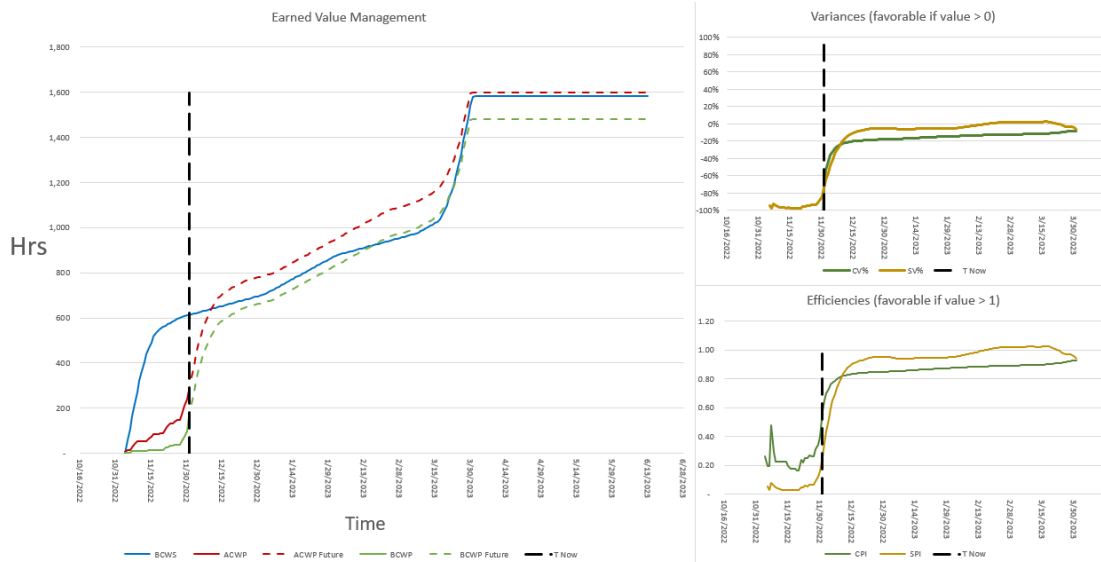


Figure 36. EVM Charts of PMD Project at T=12/01/2022

Switching to the CPM and trial results, the slight increase in expected cost is evident with the total estimate at completion being higher than the original estimate. There does appear to be some consumption of the overall float in the schedule, as the initial point of inflection in the “Current Late” plot has not shifted as much as the elapsed time has shifted. Based on the probabilistic estimates at this stage in the project there is a 97% chance of completing the effort prior to 03/31/2023. There is a 50% chance that the planned budget of 1,582 billable hours is achieved while there is an 87% chance that the budget will fall within the negotiated amount of 1,740 billable hours.

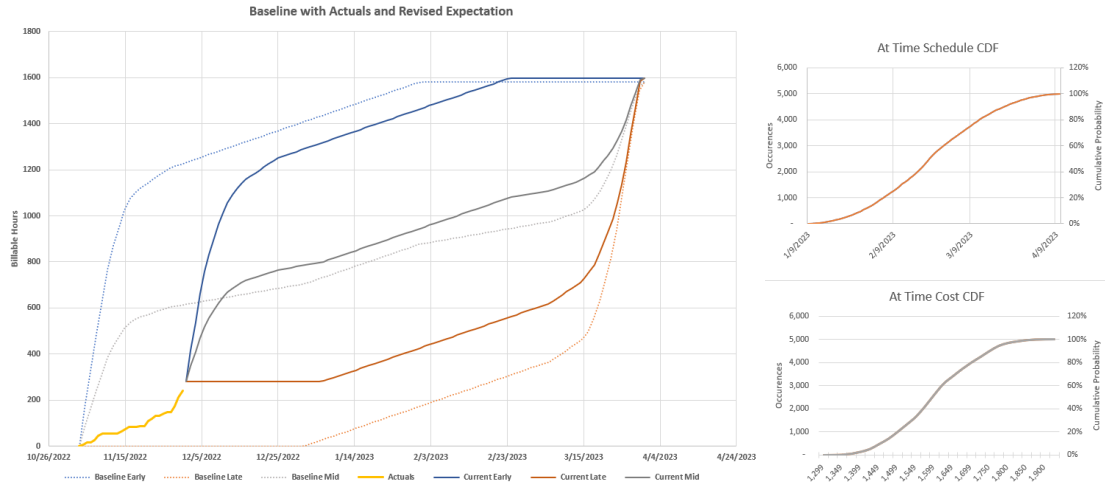


Figure 37. CPM and Trial Results of PMD Project at T=12/01/2022

C. PMD PROJECT AT T=01/03/2023

At T=01/03/2022, the EVM chart in Figure 38 shows that the project is experiencing a negative cost variance at -98%. Schedule variance is mildly unfavorable at -27%. The negative cost variances that have occurred have pushed the expected estimate at completion up to 1,731 billable hours. The expected schedule has not shifted from 03/31/2023.

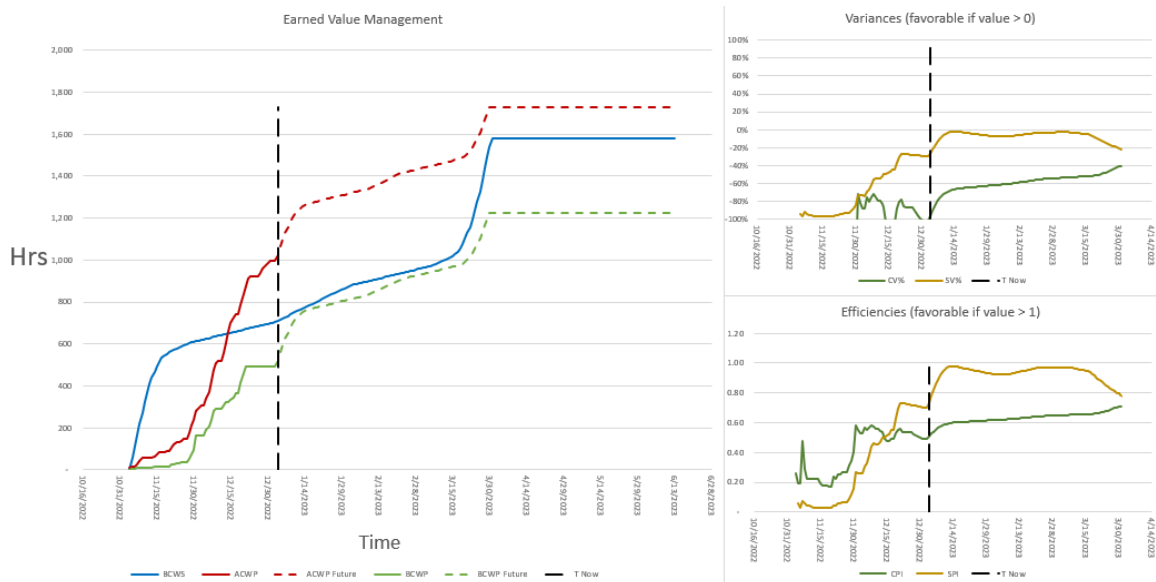


Figure 38. EVM Charts of PMD Project at T=01/03/2023

In the CPM and Trial results charts shown in Figure 39, it is also apparent that the project is experiencing a projected cost overrun while maintaining the original project schedule. There is a 50% chance that the project will ultimately perform at or under the revised expected budget of 1,731 billable hours. At this stage, there is also a 10% chance that the original budget of 1,582 billable hours will be achieved while there is still a 52% chance that the negotiated budget of 1,740 billable hours will be achieved. The likelihood of meeting the 03/31/2023 schedule is reporting at nearly 98%.

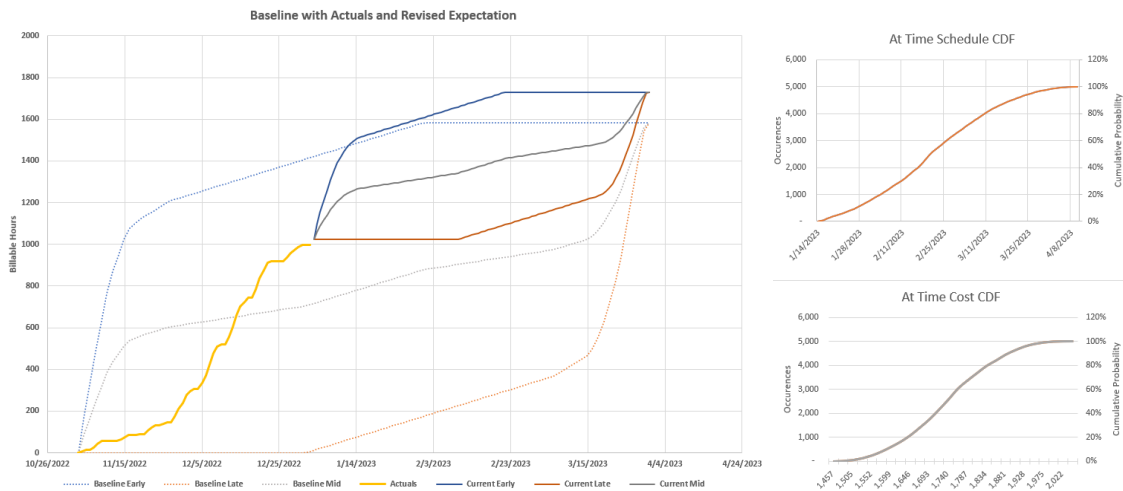


Figure 39. CPM and Trial Results of PMD Project at T=01/03/2023

D. PMD PROJECT AT T=02/01/2023

At T=02/01/2023, considerable project progress has been made as shown in Figure 40. At this stage in the project, the initial budget has been exceeded. The cost variance has become more favorable in the past month of execution, coming in at -25% while schedule variance has improved to a favorable +25%. This positive schedule variance indicates that the project is exceeding its scheduled expectations.

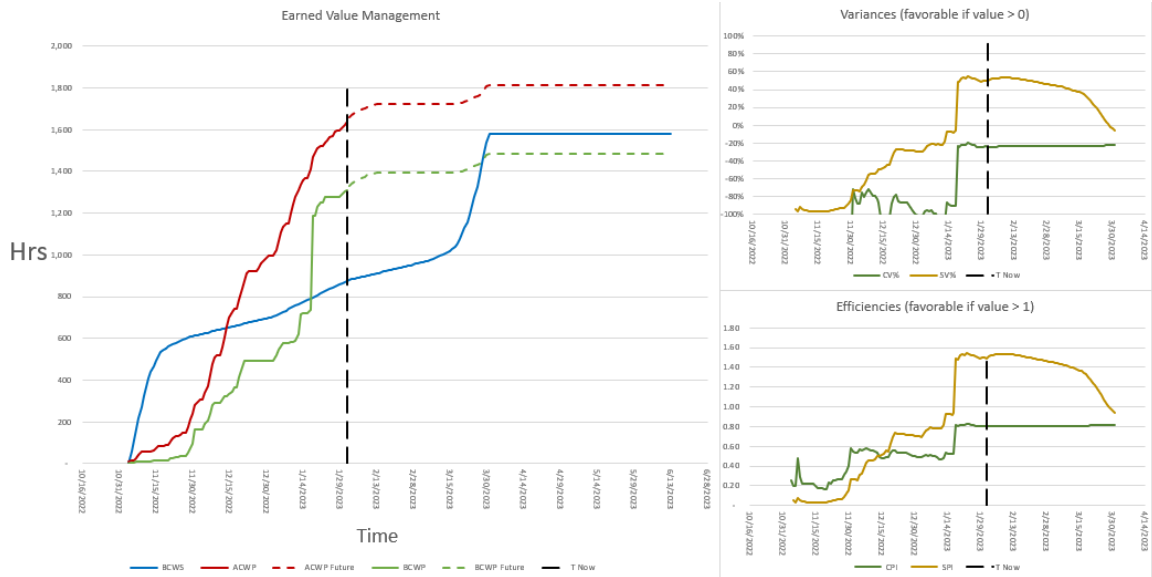


Figure 40. EVM Charts of PMD Project at T=02/01/2023

The CPM and Trial results also show that the expected cost at completion is now expected to be 1,811 billable hours as shown in Figure 41. As shown by the CDF in Figure 41, there is no chance that the billable hours will be less than the original expectation of 1,582 or the negotiated budget of 1,740. Also from the CDF, there is a 47% chance that the budget will meet the revised expectation of 1,811 billable hours. The project is anticipated to be completed within the schedule allocation with near certainty, as all simulated trials completed prior to 02/20/2023.

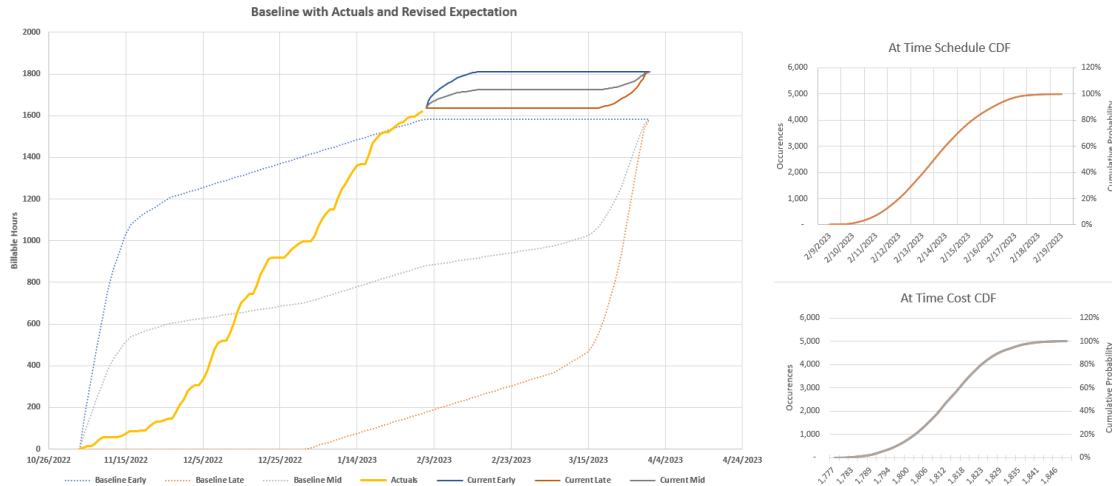


Figure 41. CPM and Trial Results of PMD Project at T=02/01/2023

E. PMD PROJECT AT T=03/01/2023

At 03/01/2023, the project is nearing completion. The BCWP is nearly at the same value as the BCWS as show in Figure 42. ACWP at 03/01/2023 is reported at 1,715 with an expected cost at completion of 1,720. Cost Variance is around -9%, which is still unfavorable based on the initial expectation. The Management Reserve assigned to this effort does cover the cost variance. Schedule Variance is a favorable 65% and it appears that the project will be completed ahead of schedule.

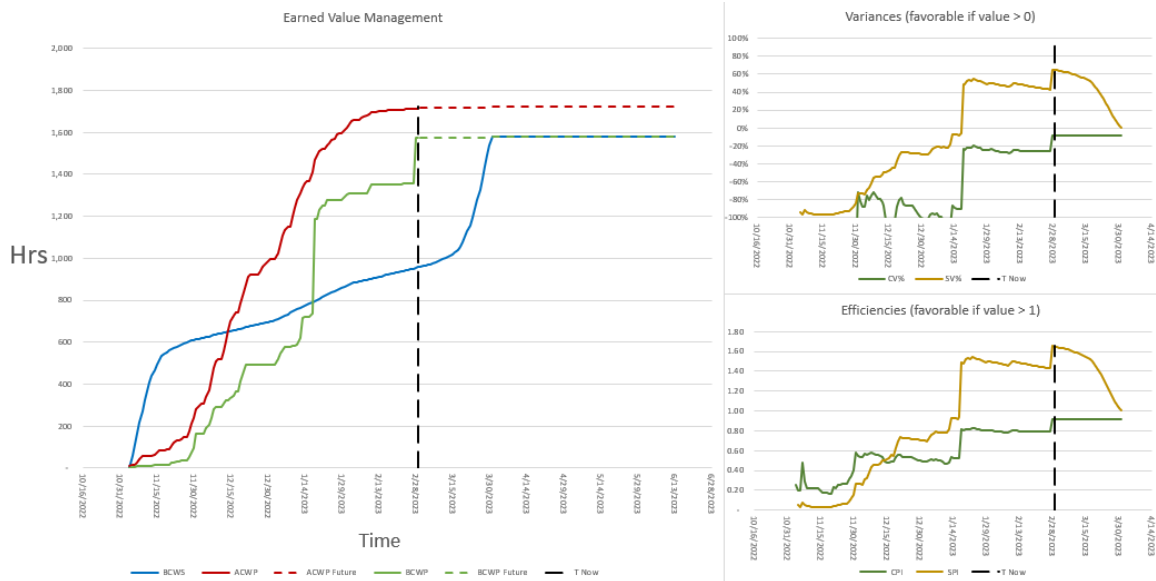


Figure 42. EVM Charts of PMD Project at T=03/01/2023

In Figure 43, the CPM and Trial Results also show the minimal amount of work remaining to complete the project. The project is expected to be completed in one or two days with reasonable certainty as demonstrated by the trial results. The cost is expected to fall between 1,718 and 1,722 billable hours.

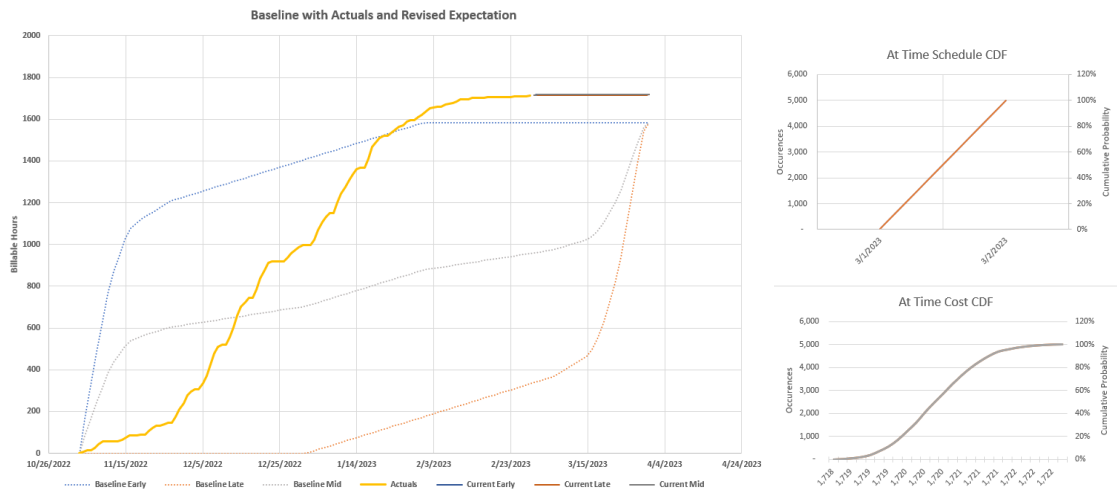


Figure 43. CPM and Trial Results of PMD Project at T=03/01/2023

F. PMD PROJECT AT T=03/31/2023

At T=03/31/2023, the project is still ongoing. As shown in Figure 44, progress over the past month has been minimal and there is still some minor work to be done. Cost Variance is at an unfavorable -10% while Schedule Variance has now reached 0%.

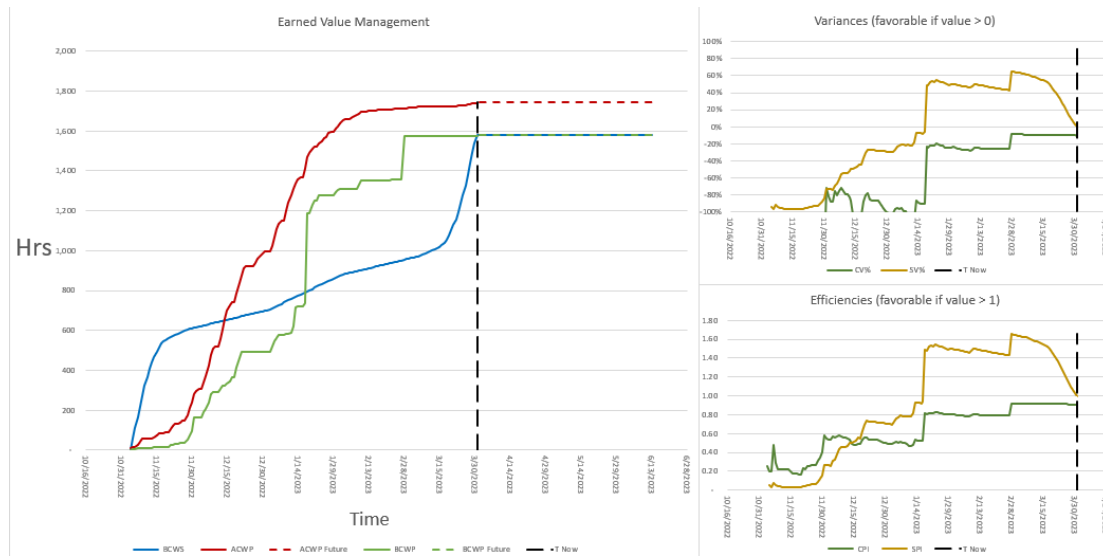


Figure 44. EVM Charts of PMD Project at T=03/31/2023

Figure 45 shows the CPM and Trial results at T=03/31/2023. Here we can clearly see the small amount of work remaining to complete this project as shown by the remaining line segment at the tail end of the actual hours. Additionally, the result of the trials estimates that the project will still be completed in one or two days. The expected costs are somewhere between 1,744 and 1,748.

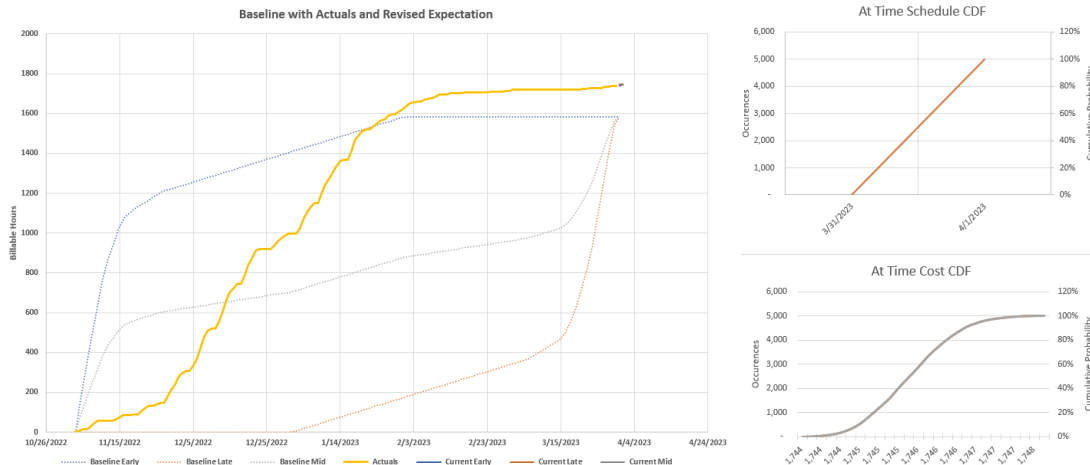


Figure 45. CPM and Trial Results of PMD Project at T=03/31/2023

G. PMD PROJECT AT T=05/24/2023 (PROJECT END)

On 05/24/2023, the project was delivered. The result of this effort was that the project was delivered 54 days late. The project was nearly complete on 03/01/2023 and for 84 days, the project had stalled as shown in Figure 46 and Figure 47. The resultant cost at completion was 1,788 billable hours of labor, which slightly exceeded the management reserves that were placed on this effort at the onset of the project.

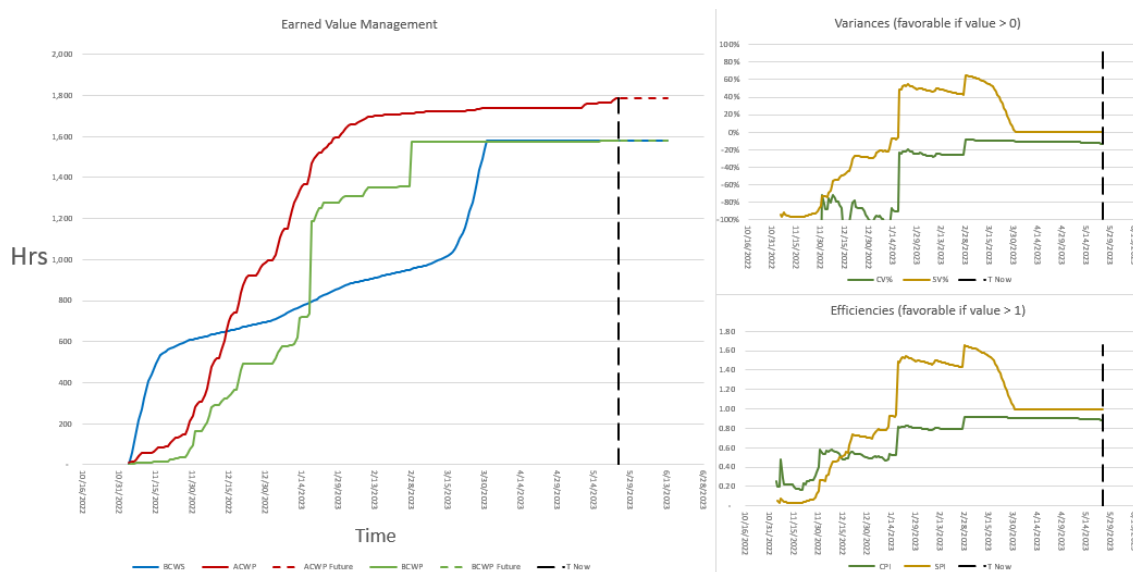


Figure 46. EVM Charts of PMD Project at T=05/24/2023

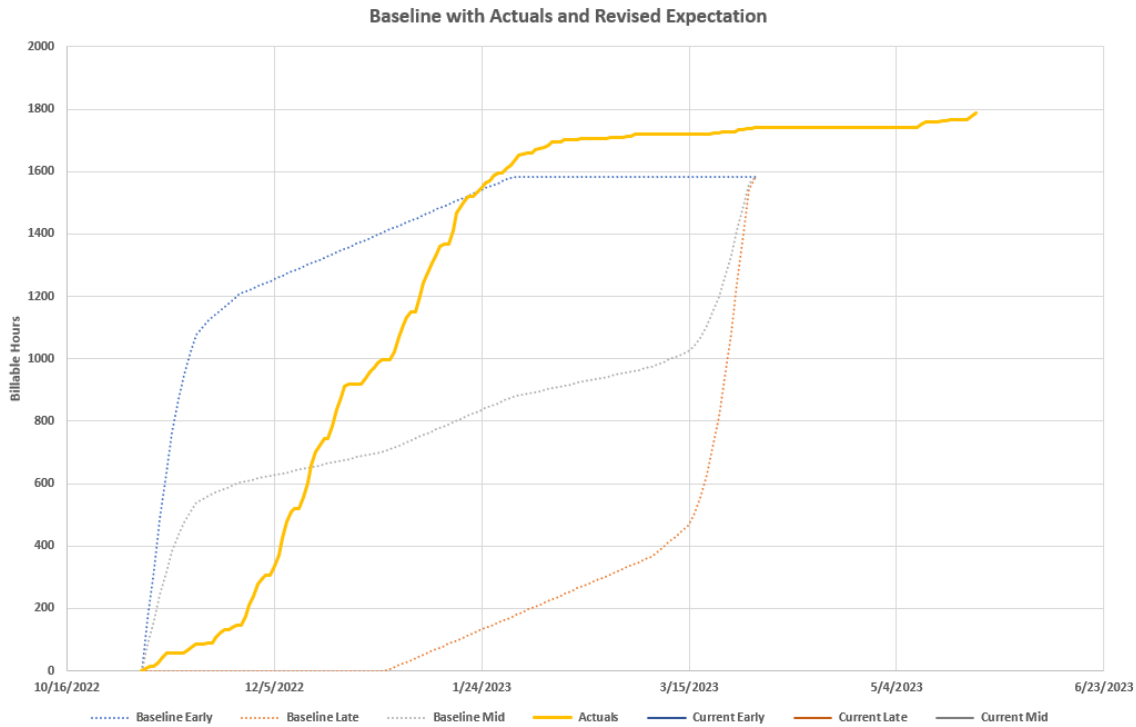


Figure 47. CPM of PMD Project at T=05/24/2023

H. COST AND SCHEDULE CONFIDENCE INTERVALS OVER EXECUTION

Over the course of the PMD Project, trials were conducted, and the likelihood of different outcomes was recorded. In Figure 48, the confidence interval for cost is shown. In this figure, as the total number of open operations are reduced through execution, the range of the confidence interval begins to narrow. Initially, the expected, or planned cost was higher than the trial value expectations with error bars illustrating the confidence interval between 40% and 80% of all trials. From a supplier perspective, this is generally a favorable initial condition, as the available budget exceeds the expected costs.

At around 02/04/2023, the confidence interval of the trial outcomes decreased significantly. This reduction corresponds to the completion of many splintering branches in the network diagram and consolidation of activities into higher levels of the WBS.

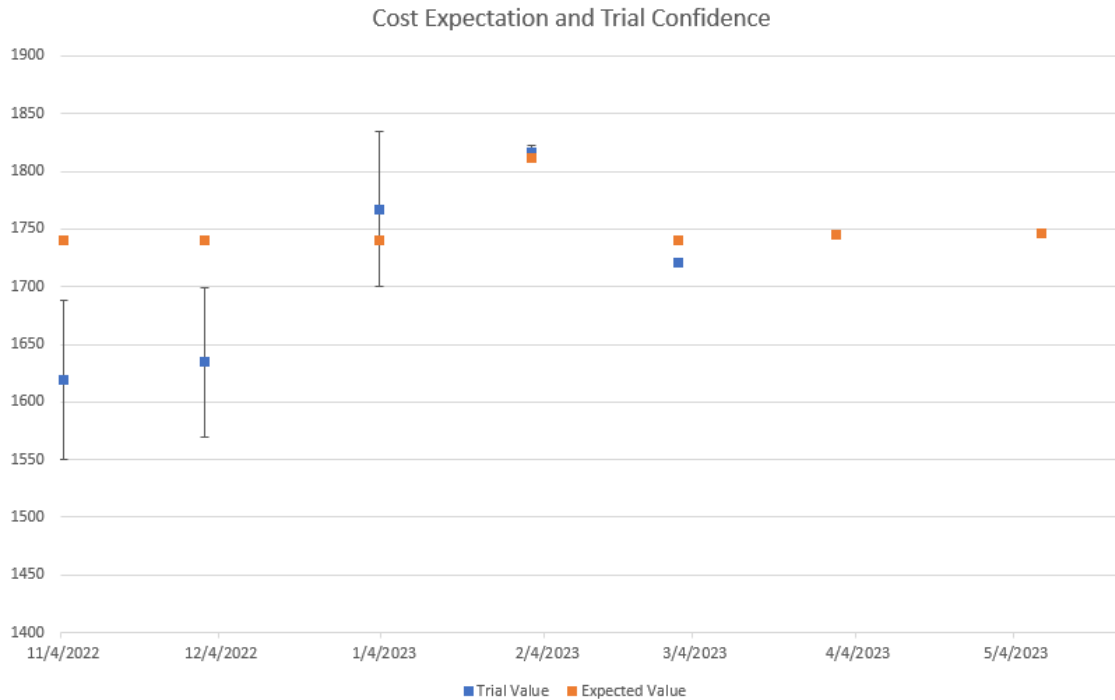


Figure 48. Cost Expectation and Trial Confidence over PMD Project Execution

A similar observation can be made for schedule expectations and trial confidence as shown in Figure 49. At the beginning of the project, the confidence interval for the simulated trials is relatively large and reduces as project execution commences.

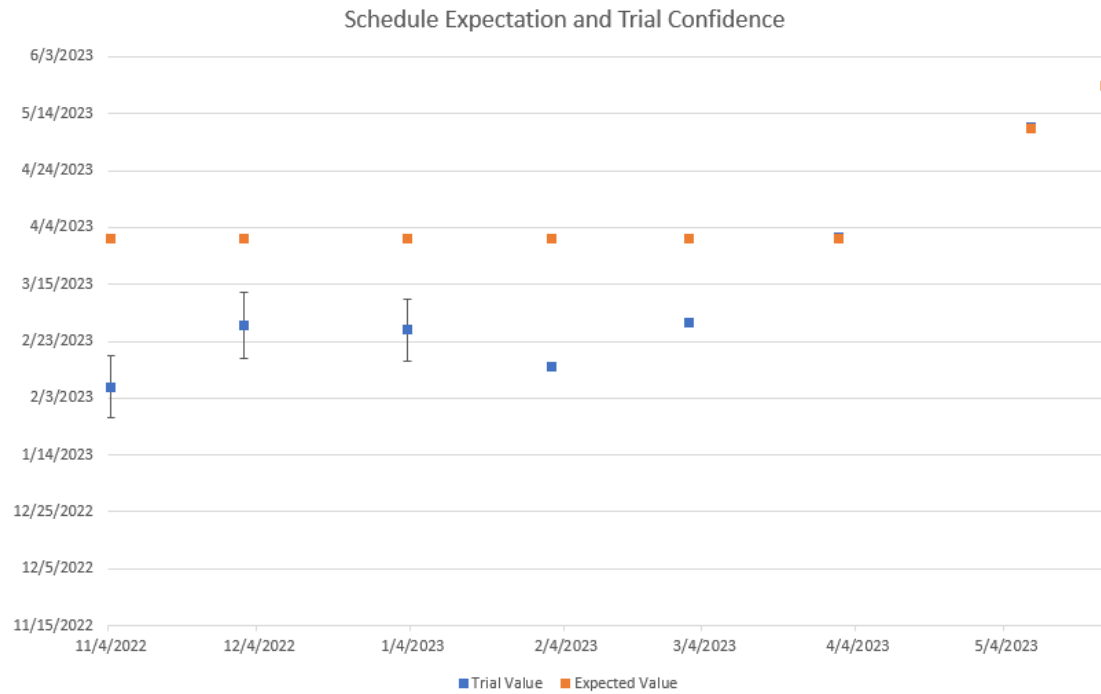


Figure 49. Schedule Expectation and Trial Confidence over PMD Project Execution

THIS PAGE INTENTIONALLY LEFT BLANK

VIII. CONCLUSION AND RECOMMENDATIONS

In Job-Shop Manufacturing, there are several factors that can influence the performance of any specific job. In this thesis, activity planning and performance data was used to evaluate the overall performance of the effort and continually revise forecasts of the outcome as additional information became available.

Earned Value Management reporting techniques are effective for reporting on historical performance of an effort when a baseline schedule can be established. Many insights can be gleaned from the evaluation of an EVM report, and it can serve as a helpful tool for program managers when assessing the performance of projects and determining if additional oversight is needed.

Earned Value Management falls short when considerable expected variation is present in an effort. EVM metrics for schedule rely heavily on the baseline schedule (BCWS) but generally do not have methods to assess the impact of critical path tasking delays at early stages of the project. If critical path activities are delayed, but other progress is made on other paths through the activity network that have float associated with them, the project will still seem to be tracking on pace with the plan. This schedule slip could be undetected with EVM reporting until a much later stage in the project.

Critical Path / Monte Carlo (CPMC) reporting can be a very strong forecasting tool for projects with high levels of expected variation and intrinsic parallelism. In CPMC reporting, early and late bounds are established as extreme threshold values that serve as guides for a program manager as they are assessing the execution of the project.

In CPMC reporting, the updated difference between the early and late bounds throughout execution provides additional insights into the available float that exists in the activity network. Float is a critical metric to manage when prioritizing competing efforts in a Job-Shop. In Job-Shop manufacturing, jobs are often competing for resources (equipment and labor) with other unrelated efforts. A Shop Scheduler must continuously evaluate the prioritization of activities within the shop. If all deliveries are treated as equally important, the most effective way to manage the prioritization across projects is to

evaluate the impact of delay on the critical path. Projects with more schedule float may have greater tolerance for interruptions or delays (for activities that are not currently on the critical path). On the other hand, projects with limited amounts of float may have multiple paths that are potentially able to drive the overall schedule. Delays to paths with limited float could eventually shift that path to be the critical path and ultimately delay a customer delivery.

A limitation to both EVM and CPMC reporting is the availability of data that could serve as a constraint to schedule. In the PMD Project used in Chapter VII, both EVM and CPMC did not anticipate the late-stage delay that the project ultimately experienced. After further investigation, it was determined that the delay was caused by a purchased material item that was incorporated into the top-level assembly. This item was delayed and therefore the project was unable to be delivered to the customer on time. Future iterations of this work should consider expected material availability. They should also associate material requirements with the activity that would first require them so that the Early Start of the activity could not occur prior to material arrival. If material is a part of the critical path, it may drive program managers to cultivate other sources of material or perform some other supplier risk management effort.

If comprehensive data is evaluated with a CPMC methodology, the output reports can be a highly effective tool to assess a project's current and future status. This forward-looking method can be helpful in making prioritization decisions and communicating risk to external stakeholders when variation and other uncertainty is present.

APPENDIX. VISUAL BASIC FOR APPLICATIONS (VBA) CODE

This section contains the Visual Basic for Applications (VBA) code that was developed for this thesis. The code consists of one (1) form, four (4) modules, and seven (7) class modules. All elements of this code are required simultaneously within a macro enabled workbook in Microsoft Excel to perform the functions demonstrated in this thesis.

A. FORMS

Forms are used by Microsoft Products as the primary user interface between the user and the underlying application.

1. MainFrm

The MainFrm or Main Form is the front end for the application. Data is loaded into the input sheets directly in Microsoft Excel and this form is used to trigger all processing activities.

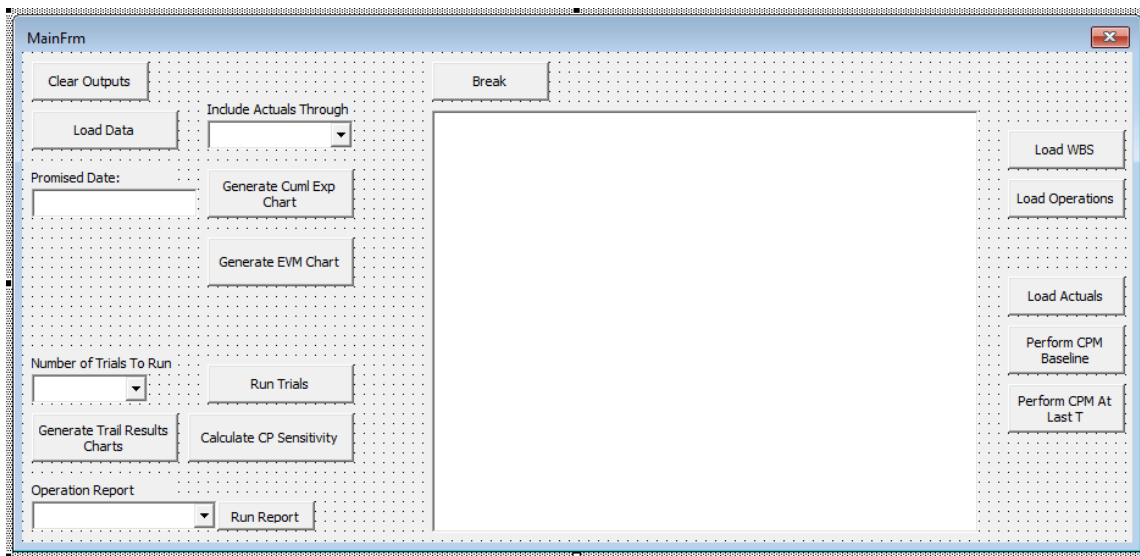
The image shows a screenshot of a Visual Basic for Applications (VBA) form titled "MainFrm". The form is set against a dotted grid background. It features a central large rectangular area, likely for a chart or data display. To the left of this area, there are several controls: a "Clear Outputs" button at the top; a "Load Data" button; a "Promised Date:" label with a text input field; a "Number of Trials To Run" label with a dropdown menu; a "Generate Trail Results Charts" button; and an "Operation Report" label with a dropdown menu. To the right of the central area, there are buttons for "Generate Cuml Exp Chart", "Generate EVM Chart", "Run Trials", "Calculate CP Sensitivity", and "Run Report". Above the central area is a "Break" button. On the far right, there is a vertical stack of buttons: "Load WBS", "Load Operations", "Load Actuals", "Perform CPM Baseline", and "Perform CPM At Last T". The form has a standard Windows-style title bar with a close button (X) in the top right corner.

Figure 50. MainFrm Layout

Clear Outputs

Load Data

Include Actuals Through
9/13/2021

Promised Date:
10/1/2021

Generate Cuml Exp Chart

Generate EVM Chart

Number of Trials To Run
500

Run Trials

Generate Trail Results Charts

Calculate CP Sensitivity

Operation Report

Run Report

Break

7/4/2023 7:27:05 PM - CP Sensitivity Reports completed in: 93.75 mSec
 7/4/2023 7:26:35 PM - Trial Reports completed in: 4546.88 mSec
 7/4/2023 7:26:30 PM - Trials Generated in: 3179.69 mSec
 7/4/2023 7:26:26 PM - EVM Report completed in: 726.56 mSec
 7/4/2023 7:26:24 PM - CPM Calculated in: 179.69 mSec
 7/4/2023 7:26:24 PM - CPM Calculated in: 351.56 mSec
 7/4/2023 7:26:23 PM - Actuals Data Loaded in: 234.38 mSec
 7/4/2023 7:26:13 PM - Actuals Data Read in: 7.81 mSec
 7/4/2023 7:26:13 PM - Actuals Data in rows: 2 to 102
 7/4/2023 7:26:13 PM - Operation Data Loaded in: 39.06 mSec
 7/4/2023 7:26:13 PM - Operation Data in rows: 4 to 40
 7/4/2023 7:26:13 PM - WBS Data Loaded in: 54.69 mSec
 7/4/2023 7:26:13 PM - WBS Data in rows: 4 to 13

Load WBS

Load Operations

Load Actuals

Perform CPM Baseline

Perform CPM At Last T

Figure 51. MainFrm after Run Results

Option Explicit

Private Log As String

Private WBS_Col As Collection

Private JO_Col As Collection

Private ActAll_Col As Collection

Private ActEval_Col As Collection

Private wbsRoot As WBS_Class

Private finalOp As JO_Class

Private Actuals_LowDate As Date

Private Actuals_HighDate As Date

Private OPEval_Col As Collection

Private Trials_Col As Collection

Private PromisedDate As Date

Private Sub BreakBtn_Click()

LogString ("--Break--")

End Sub

Private Sub ClearOutputsBtn_Click()

Sheets("Output-WBSHeirchy").Cells.ClearContents

Sheets("Output-OperationHeirchy").Cells.ClearContents

```
    Sheets("Expected-ActualsAndRemaining").Cells.ClearContents
End Sub
```

```
Public Sub ResetLog()
    Log = ""
    Debug.Print ("Log Reset")
End Sub
```

```
Public Sub LogString(in_str As String)
    Dim newLogLine As String
    newLogLine = Now() & " - " & in_str
    Log = newLogLine & vbCrLf & Log
    Debug.Print (newLogLine)
    LogTextBox.Text = Log
End Sub
```

```
Private Sub CPSensitivityBtn_Click()
    ReportModule.CP_Sensitivity WBS_Col, ActualsCombo.Value
End Sub
```

```
Private Sub CumlExpChartBtn_Click()
    If WBS_Col Is Nothing Then
        LoadWBS
    End If
    If JO_Col Is Nothing Then
        LoadOps
    End If
    If ActAll_Col Is Nothing Then
        ReadActuals
    End If

    If IsDate(TB_PromisedDate.Value) Then
        PromisedDate = CDate(TB_PromisedDate.Value)
    Else
        PromisedDate = CDate("1/1/1700")
    End If

    LoadActuals

    PerformCPMBaseline
```

PerformCPMAtT

```
With Sheets("Expected-BaselineWActualsAndRev")
    .Cells.ClearContents
    .Cells(1, 1) = "Cumulative Output"
    .Cells(2, 1) = "Date"
    .Cells(2, 2) = "Baseline Early"
    .Cells(2, 3) = "Baseline Late"
    .Cells(2, 4) = "Baseline Mid"
    .Cells(2, 5) = "Actuals"
    .Cells(2, 6) = "Current Early"
    .Cells(2, 7) = "Current Late"
    .Cells(2, 8) = "Current Mid"
    Sheets("Expected-CPMBaseline").Range("B3:E" & Sheets("Expected-
CPMBaseline").Cells(3, 5).End(xlDown).Row).Copy .Range("A3")
    Dim actualsStart As Integer
    actualsStart = .Range("A3").End(xlDown).Row + 1
    If Sheets("Expected-ActualsAndRemaining").Range("A4").End(xlDown).Row <
10000 Then
        Sheets("Expected-ActualsAndRemaining").Range("A4:A" & Sheets("Expected-
ActualsAndRemaining").Range("A4").End(xlDown).Row).Copy .Range("A" &
actualsStart)
        Sheets("Expected-ActualsAndRemaining").Range("B4:B" & Sheets("Expected-
ActualsAndRemaining").Range("B4").End(xlDown).Row).Copy .Range("E" &
actualsStart)
    End If
    Dim currentCPMStart As Integer
    currentCPMStart = .Range("A3").End(xlDown).Row + 1
    If Sheets("Expected-CPMCurrent").Range("B3").End(xlDown).Row < 10000 Then
        Sheets("Expected-CPMCurrent").Range("B3:B" & Sheets("Expected-
CPMCurrent").Range("B3").End(xlDown).Row).Copy .Range("A" & currentCPMStart)
        Sheets("Expected-CPMCurrent").Range("C3:E" & Sheets("Expected-
CPMCurrent").Range("C3").End(xlDown).Row).Copy .Range("F" & currentCPMStart)
    End If
    .Activate
End With
End Sub

Private Sub EVMBtn_Click()
```

```
ReportModule.EVMReport OPEval_Col, finalOp, ActualsCombo.Value,  
Actuals_LowDate, Application.WorksheetFunction.Max(Actuals_HighDate,  
PromisedDate)  
End Sub
```

```
Private Sub LoadActualsBtn_Click()  
    Application.ScreenUpdating = False  
  
    Sheets("Expected-ActualsAndRemaining").Cells.ClearContents  
    Sheets("Expected-CPMCurrent").Cells.ClearContents  
  
    If WBS_Col Is Nothing Then  
        LoadWBS  
    End If  
    If JO_Col Is Nothing Then  
        LoadOps  
    End If  
    If ActAll_Col Is Nothing Then  
        ReadActuals  
    End If  
    LoadActuals  
  
    Sheets("Expected-ActualsAndRemaining").Activate  
    Application.ScreenUpdating = True  
End Sub
```

```
Private Sub PerformCPMATBtn_Click()  
    If WBS_Col Is Nothing Then  
        LoadWBS  
    End If  
    If JO_Col Is Nothing Then  
        LoadOps  
    End If  
    If ActAll_Col Is Nothing Then  
        ReadActuals  
    End If  
    If ActEval_Col Is Nothing Then  
        LoadActuals  
    End If
```

```
    PerformCPMAtT
End Sub
```

```
Private Sub PerformCPMBLBtn_Click()
    If WBS_Col Is Nothing Then
        LoadWBS
    End If
    If JO_Col Is Nothing Then
        LoadOps
    End If
    If ActAll_Col Is Nothing Then
        ReadActuals
    End If
    If ActEval_Col Is Nothing Then
        LoadActuals
    End If
End Sub
```

```
    PerformCPMBaseline
End Sub
```

```
Private Sub ReadActualsBtn_Click()
    If WBS_Col Is Nothing Then
        LoadWBS
    End If
    If JO_Col Is Nothing Then
        LoadOps
    End If
End Sub
```

```
    ReadActuals
End Sub
```

```
Private Sub LoadOpsBtn_Click()
    Application.ScreenUpdating = False

    If WBS_Col Is Nothing Then
        LoadWBS
    End If

    LoadOps
    Sheets("Output-OperationHeirchy").Activate
End Sub
```

```
Application.ScreenUpdating = True
End Sub
```

```
Private Sub LoadWBSBtn_Click()
    Application.ScreenUpdating = False
    LoadWBS
    Sheets("Output-WBSHeirchy").Activate
    Application.ScreenUpdating = True
End Sub
```

```
Private Sub LoadWBS()
    Set WBS_Col = LoadModule.LoadWBS(wbsRoot)
End Sub
```

```
Private Sub LoadOps()
    Set JO_Col = LoadModule.LoadOperations(finalOp, WBS_Col)

    OpsCombo.Clear
    Dim comboList() As Variant
    ReDim comboList(0 To JO_Col.Count, 0 To 1)

    Dim workingJO As JO_Class
    Dim i As Integer
    i = 0
    For Each workingJO In JO_Col
        comboList(i, 1) = workingJO.JobOperation
        comboList(i, 0) = workingJO.WBS_String & " - " & workingJO.OperationName

        i = i + 1
    Next workingJO

    OpsCombo.List = comboList
End Sub
```

```
Private Sub ReadActuals()
    Set ActAll_Col = LoadModule.ReadActuals(JO_Col)
    Dim jotBuilder As JOT_Class
    Actuals_LowDate = DateAdd("yyyy", 100, Now())
    Actuals_HighDate = DateAdd("yyyy", -100, Now())
```

```

Dim ActualDates As Collection
Set ActualDates = New Collection
ActualsCombo.Clear

Dim workDates(), tempD As Date
ReDim workDates(1 To ActAll_Col.Count)

Dim i, j As Integer
i = 1

For Each jotBuilder In ActAll_Col
    workDates(i) = jotBuilder.WorkDate
    i = i + 1
Next jotBuilder

For i = LBound(workDates) To UBound(workDates)
    For j = i + 1 To UBound(workDates)
        If workDates(i) > workDates(j) Then
            tempD = workDates(j)
            workDates(j) = workDates(i)
            workDates(i) = tempD
        End If
    Next j
Next i

For i = LBound(workDates) To UBound(workDates)
    On Error Resume Next 'temp supress errors to avoid duplicates
    ActualDates.Add workDates(i), CStr(workDates(i))
    On Error GoTo 0 'resume errors

    If Actuals_LowDate > workDates(i) Then
        Actuals_LowDate = workDates(i)
    End If
    If Actuals_HighDate < workDates(i) Then
        Actuals_HighDate = workDates(i)
    End If
Next i

ActualsCombo.List = ReportModule.CollectionToArray(ActualDates)

```

```

    ActualsCombo.Value = Actuals_HighDate
End Sub

Sub LoadActuals()
    Set ActEval_Col = LoadModule.LoadActuals(JO_Col, ActAll_Col,
ActualsCombo.Value, Actuals_LowDate, finalOp, OPEval_Col)
End Sub

Sub PerformCPMBaseline()
    LoadModule.PerformCPM finalOp, OPEval_Col, DateAdd("d", -1,
Actuals_LowDate), "CPMBaseline", JO_Col, PromisedDate
End Sub

Sub PerformCPMAAtT()
    LoadModule.PerformCPM finalOp, OPEval_Col, ActualsCombo.Value,
"CPMCurrent", JO_Col, PromisedDate
End Sub

Private Sub RunOpReportBtn_Click()
    If ActEval_Col Is Nothing Then
        LoadActuals
    End If

    Dim workingJO As JO_Class
    For Each workingJO In JO_Col
        If workingJO.JobOperation = OpsCombo.Column(1) Then
            ReportModule.OperationReport workingJO
            Exit For
        End If
    Next workingJO

    Sheets("Output-OperationDetailReport").Activate
End Sub

Private Sub RunTrialsBtn_Click()
    If WBS_Col Is Nothing Then
        LoadWBS
    End If
    If JO_Col Is Nothing Then
        LoadOps

```

```

End If
If ActAll_Col Is Nothing Then
    ReadActuals
End If
If ActEval_Col Is Nothing Then
    LoadActuals
End If

Set Trials_Col = TrialModule.LoadTrials(TrialsCombo.Value, finalOp,
Actuals_LowDate, ActualsCombo.Value)
End Sub

Private Sub TrailResultsChsBtn_Click()
    ReportModule.TrialReport Trials_Col, Actuals_LowDate, ActualsCombo.Value
End Sub

Private Sub UserForm_Initialize()
    TrialsCombo.AddItem ("1")
    TrialsCombo.AddItem ("10")
    TrialsCombo.AddItem ("100")
    TrialsCombo.AddItem ("500")
    TrialsCombo.AddItem ("1000")
    TrialsCombo.AddItem ("2000")
    TrialsCombo.AddItem ("5000")
    TrialsCombo.AddItem ("10000")
    TrialsCombo.Value = "500"
End Sub

```

B. MODULES

Modules are generally sequential subroutines that have been broken out to separate files based on their primary purpose.

1. LoadModule

The LoadModule consists of functions for loading input user data from the various workbook sheets. It also has the initial procedure for calculating the Critical Path Method.

Option Explicit

```

Function LoadWBS(ByRef wbsRoot As WBS_Class) As Collection
    Dim sTime, eTime As Double
    sTime = Timer

    Set LoadWBS = New Collection

    'read the data
    Dim headerRowIndex As Integer

    With Sheets("Input-WBS")
        headerRowIndex = 1
        While .Cells(headerRowIndex, 1) <> "WBS"
            headerRowIndex = headerRowIndex + 1
        Wend

        Dim lastRow As Integer
        lastRow = .Cells(headerRowIndex, 1).End(xlDown).Row 'find the last row in the
data set

        Dim totalRows As Integer
        totalRows = lastRow - headerRowIndex 'calculate how many elements of data we'll
need to iterate over

        MainFrm.LogString ("WBS Data in rows: " & headerRowIndex + 1 & " to " &
lastRow)

        Dim i As Double

        Dim wbsBuilder As WBS_Class

        For i = headerRowIndex + 1 To lastRow
            Set wbsBuilder = New WBS_Class
            wbsBuilder.WBS_String = .Cells(i, 1)
            wbsBuilder.WBSParent_String = .Cells(i, 2)
            Set wbsBuilder.WBS_Children = New Collection
            LoadWBS.Add wbsBuilder
        Next i
    End With

    'Build the heirchy

```

```

Dim wbsBuilder2 As WBS_Class

For Each wbsBuilder In LoadWBS 'looking for this ones parent
    If wbsBuilder.WBSParent_String = "" Then
        Set wbsRoot = wbsBuilder
    Else
        For Each wbsBuilder2 In LoadWBS 'this is the potential parent
            If wbsBuilder2.WBS_String = wbsBuilder.WBSParent_String Then
                wbsBuilder2.WBS_Children.Add wbsBuilder
                Set wbsBuilder.WBS_Parent = wbsBuilder2
                Exit For 'we can exit here because a wbs only has one parent
            End If
        Next wbsBuilder2
    End If
Next wbsBuilder

'write the output
With Sheets("Output-WBSHeirchy")
    .Cells.ClearContents
    .Cells(1, 1) = "WBS Heirchy"
    .Cells(3, 1) = "WBS"
    Dim structureRows() As String
    Dim structureTabs() As String
    Dim j As Integer
    i = 4
    structureRows = Split(wbsRoot.PrintStructure, vbCrLf)
    For j = 0 To UBound(structureRows)
        structureTabs = Split(structureRows(j), vbTab)
        .Cells(i, UBound(structureTabs) + 1) = structureTabs(UBound(structureTabs))
        i = i + 1
    Next j

End With

eTime = Timer
MainFrm.LogString ("WBS Data Loaded in: " & Round((eTime - sTime) * 1000, 2) &
" mSec")
End Function

```

```
Function LoadOperations(ByRef finalOperation As JO_Class, WBS_Col As Collection)
As Collection
```

```
    Dim sTime, eTime As Double
    sTime = Timer
```

```
    Set LoadOperations = New Collection
```

```
    'read the data
    Dim headerRowIndex As Integer
```

```
    With Sheets("Input-Operations")
        headerRowIndex = 1
        While .Cells(headerRowIndex, 1) <> "Job Operation"
            headerRowIndex = headerRowIndex + 1
        Wend
```

```
        Dim lastRow As Integer
        lastRow = .Cells(headerRowIndex, 1).End(xlDown).Row 'find the last row in the
data set
```

```
        Dim totalRows As Integer
        totalRows = lastRow - headerRowIndex 'calculate how many elements of data we'll
need to iterate over
```

```
        MainFrm.LogString ("Operation Data in rows: " & headerRowIndex + 1 & " to " &
lastRow)
```

```
        Dim i As Double
```

```
        Dim opBuilder As JO_Class
```

```
        For i = headerRowIndex + 1 To lastRow
            Set opBuilder = New JO_Class
            opBuilder.JobOperation = .Cells(i, 1)
            opBuilder.WBS_String = .Cells(i, 2)
            opBuilder.Sequence = .Cells(i, 3)
            opBuilder.OperationName = .Cells(i, 4)
            opBuilder.NumResources = .Cells(i, 5)
            Set opBuilder.Op_GlobalPrevs = New Collection
            Set opBuilder.Actuals_All = New Collection
```

```

Set opBuilder.EvalExpPoints = New Collection
Set opBuilder.Trials = New Collection
Set opBuilder.Estimate = New JO_Estimation
If .Cells(i, 6) <> "" Then 'Straight Expectation
    opBuilder.Estimate.mode = 0
    opBuilder.Estimate.p0 = .Cells(i, 6)
ElseIf .Cells(i, 7) <> "" Then 'Lognormal
    opBuilder.Estimate.mode = 1
    opBuilder.Estimate.p0 = .Cells(i, 7)
    opBuilder.Estimate.p1 = .Cells(i, 8)
ElseIf .Cells(i, 9) <> "" Then 'Triangular
    opBuilder.Estimate.mode = 2
    opBuilder.Estimate.p0 = .Cells(i, 9)
    opBuilder.Estimate.p1 = .Cells(i, 10)
    opBuilder.Estimate.p2 = .Cells(i, 11)
ElseIf .Cells(i, 12) <> "" Then 'BetaPert
    opBuilder.Estimate.mode = 3
    opBuilder.Estimate.p0 = .Cells(i, 12)
    opBuilder.Estimate.p1 = .Cells(i, 13)
    opBuilder.Estimate.p2 = .Cells(i, 14)
ElseIf .Cells(i, 15) <> "" Then 'Beta
    opBuilder.Estimate.mode = 4
    opBuilder.Estimate.p0 = .Cells(i, 15)
    opBuilder.Estimate.p1 = .Cells(i, 16)
    opBuilder.Estimate.p2 = .Cells(i, 17)
    opBuilder.Estimate.p3 = .Cells(i, 18)
ElseIf .Cells(i, 19) <> "" Then 'Normal
    opBuilder.Estimate.mode = 5
    opBuilder.Estimate.p0 = .Cells(i, 19)
    opBuilder.Estimate.p1 = .Cells(i, 20)
ElseIf .Cells(i, 21) <> "" Then 'Uniform
    opBuilder.Estimate.mode = 6
    opBuilder.Estimate.p0 = .Cells(i, 21)
    opBuilder.Estimate.p2 = .Cells(i, 22)
Else
    opBuilder.Estimate.mode = -1 'Error
    MainFrm.LogString ("No estimations provided for " &
opBuilder.JobOperation)
End If
LoadOperations.Add opBuilder

```

```

    Next i
End With

'Build the heirchy
Dim opBuilder2 As JO_Class
Dim wbsBuilder As WBS_Class

For Each opBuilder In LoadOperations 'looking for this ones nextOp
    'First connect ops within the same WBS
    For Each opBuilder2 In LoadOperations 'this is the potential nextOp
        If opBuilder2.WBS_String = opBuilder.WBS_String And opBuilder2.Sequence =
opBuilder.Sequence + 1 Then
            Set opBuilder2.Op_ThisPrev = opBuilder
            Set opBuilder.Op_ThisNext = opBuilder2
            Set opBuilder.Op_GlobalNext = opBuilder2
            opBuilder2.Op_GlobalPrevs.Add opBuilder
            Exit For 'we can exit here because a op only has one nextOperation at the op
level
        End If
    Next opBuilder2

    'find the operations WBS
    For Each wbsBuilder In WBS_Col
        If opBuilder.WBS_String = wbsBuilder.WBS_String Then
            Set opBuilder.WBS = wbsBuilder
            Exit For
        End If
    Next wbsBuilder
Next opBuilder

'Next connect last in WBS to first in the next WBS
For Each wbsBuilder In WBS_Col
    For Each opBuilder In LoadOperations
        If wbsBuilder.WBS_String = opBuilder.WBS_String Then
            Set wbsBuilder.Op_First = opBuilder.Op_First
            Set wbsBuilder.Op_Last = opBuilder.Op_Last
            Exit For
        End If
    Next opBuilder
Next wbsBuilder

```

```

'connect first op in wbs to last op in previous wbs
For Each wbsBuilder In WBS_Col
    If Not wbsBuilder.WBS_Parent Is Nothing Then
        wbsBuilder.WBS_Parent.Op_First.Op_GlobalPrevs.Add wbsBuilder.Op_Last
        Set wbsBuilder.Op_Last.Op_GlobalNext = wbsBuilder.WBS_Parent.Op_First
    Else
        Set finalOperation = wbsBuilder.Op_Last
    End If
Next wbsBuilder

'write the output
Dim colLen As Integer
With Sheets("Output-OperationHeirchy")
    .Cells.ClearContents
    .Cells(1, 1) = "Operation Heirchy"
    .Cells(3, 1) = "Operation"
    Dim structureRows() As String
    Dim structureTabs() As String
    Dim j As Integer
    i = 4
    structureRows = Split(finalOperation.PrintStructure, vbCrLf)
    For j = 0 To UBound(structureRows)
        structureTabs = Split(structureRows(j), vbTab)
        .Cells(i, UBound(structureTabs) + 1) = structureTabs(UBound(structureTabs))
        colLen = Application.WorksheetFunction.Max(colLen, UBound(structureTabs))
        i = i + 1
    Next j

End With

eTime = Timer
MainFrm.LogString ("Operation Data Loaded in: " & Round((eTime - sTime) * 1000,
2) & " mSec")
End Function

Function ReadActuals(ByRef JO_Col As Collection) As Collection
    Dim sTime, eTime As Double
    sTime = Timer

```

```

Set ReadActuals = New Collection

'read the data
Dim headerRowIndex As Integer

With Sheets("Input-Actuals")
    headerRowIndex = 1
    While .Cells(headerRowIndex, 1) <> "Job Operation"
        headerRowIndex = headerRowIndex + 1
    Wend

    Dim lastRow As Integer
    lastRow = .Cells(headerRowIndex, 1).End(xlDown).Row 'find the last row in the
data set

    Dim totalRows As Integer
    totalRows = lastRow - headerRowIndex 'calculate how many elements of data we'll
need to iterate over

    MainFrm.LogString ("Actuals Data in rows: " & headerRowIndex + 1 & " to " &
lastRow)

    Dim i As Double

    Dim jotBuilder As JOT_Class

    For i = headerRowIndex + 1 To lastRow
        Set jotBuilder = New JOT_Class
        jotBuilder.JobOperation = .Cells(i, 1)
        jotBuilder.WorkDate = CDate(Format(.Cells(i, 2), "yyyy-mm-dd"))
        jotBuilder.BillableHours = .Cells(i, 3)

        ReadActuals.Add jotBuilder
    Next i
End With

'attach data to JO
Dim opBuilder As JO_Class
For Each jotBuilder In ReadActuals
    For Each opBuilder In JO_Col

```

```

        If opBuilder.JobOperation = jotBuilder.JobOperation Then
            opBuilder.Actuals_All.Add jotBuilder
            Exit For
        End If
    Next opBuilder
Next

eTime = Timer
MainFrm.LogString ("Actuals Data Read in: " & Round((eTime - sTime) * 1000, 2) &
" mSec")

End Function

Function LoadActuals(JO_Col As Collection, Act_Col As Collection, evalDate As Date,
actualsLowDate As Date, finalOp As JO_Class, OPEval_Col As Collection) As
Collection
    Dim sTime, eTime As Double
    sTime = Timer

    Dim jotBuilder As JOT_Class
    Set LoadActuals = New Collection
    Set OPEval_Col = New Collection

    For Each jotBuilder In Act_Col
        If jotBuilder.WorkDate <= evalDate Then
            LoadActuals.Add jotBuilder
        End If
    Next jotBuilder

    Dim opBuilder As JO_Class
    For Each opBuilder In JO_Col
        Set opBuilder.Actuals_Eval = New Collection 'reset on recalc
        Set opBuilder.EvalExpPoints = New Collection 'reset on recalc
        For Each jotBuilder In LoadActuals
            If opBuilder.JobOperation = jotBuilder.JobOperation Then
                opBuilder.Actuals_Eval.Add jotBuilder
                Set jotBuilder.jo = opBuilder
            End If
        Next jotBuilder
    Next opBuilder

```

```

'evaluate each date in the effective range
Dim rDate As Date
rDate = DateAdd("d", -1, actualsLowDate)
Dim joEval As JO_Eval
Dim evalBin As JO_Eval_Bin

While rDate <= evalDate
    Set evalBin = New JO_Eval_Bin
    evalBin.reportDate = rDate
    Set evalBin.Eval_Col = New Collection
    For Each opBuilder In JO_Col
        Set joEval = New JO_Eval
        joEval.reportDate = rDate
        Set joEval.ParentOp = opBuilder
        evalBin.Eval_Col.Add joEval, CStr(opBuilder.JobOperation)
        opBuilder.EvalExpPoints.Add joEval, CStr(rDate)
        opBuilder.DOCALC_ActualHrs rDate
    Next opBuilder
    finalOp.DOCALC_OpActive False, rDate

    OPEval_Col.Add evalBin, CStr(rDate)
    rDate = DateAdd("d", 1, rDate)
Wend

'print actuals over time
With Sheets("Expected-ActualsAndRemaining")
    .Cells.ClearContents
    .Cells(1, 1) = "Cumulative Actuals over Time"
    .Cells(3, 1) = "Date"
    .Cells(3, 2) = "Cuml Actual Hours"
    .Cells(3, 3) = "Cuml Exp Rem Hours"
    Dim rowNum As Integer
    rowNum = 4

    rDate = DateAdd("d", -1, actualsLowDate)
    Dim actualCuml As Double
    Dim remainingCuml As Double

```

```

While rDate < evalDate
    actualCuml = 0
    remainingCuml = 0
    For Each opBuilder In JO_Col
        Set joEval = opBuilder.EvalExpPoints.Item(CStr(rDate))
        actualCuml = actualCuml + joEval.actualHours
        remainingCuml = remainingCuml + joEval.RemainingHours
    Next opBuilder
    .Cells(rowNum, 1) = rDate
    .Cells(rowNum, 2) = actualCuml
    .Cells(rowNum, 3) = remainingCuml
    rowNum = rowNum + 1

    rDate = DateAdd("d", 1, rDate)
Wend
End With

eTime = Timer
MainFrm.LogString ("Actuals Data Loaded in: " & Round((eTime - sTime) * 1000, 2)
& " mSec")

End Function

Function PerformCPM(finalOp As JO_Class, OPEval_Col As Collection, reportDate As
Date, sheetName As String, JO_Col As Collection, PromisedDate As Date) As Collection
    Dim sTime, eTime As Double
    sTime = Timer

    finalOp.DOCALC_CPM_Forwards (reportDate)
    'If finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_EarlyFinish > PromisedDate
    Then 'cant make promised date - revert to earliest date

        'finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_LateFinish =
        finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_EarlyFinish
        'Else
        If PromisedDate = CDate("1/1/1700") Then
            finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_LateFinish =
            finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_EarlyFinish
        Else

```

```

        finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_LateFinish =
MainModule.ConvertDateToDouble(reportDate, PromisedDate)
    End If

```

```

        finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_LateStart =
finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_LateFinish -
finalOp.EvalExpPoints.Item(CStr(reportDate)).RemainingHours
        finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_Float =
finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_LateFinish -
finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_EarlyFinish

```

```

        finalOp.DOCALC_CPM_Backwards (reportDate)

```

```

        Dim myearlyfinish As Double
        myearlyfinish =
Application.WorksheetFunction.Max(finalOp.EvalExpPoints.Item(CStr(reportDate)).CP
M_EarlyFinish, finalOp.EvalExpPoints.Item(CStr(reportDate)).CPM_LateFinish)

```

```

        Dim workingBin As JO_Eval_Bin
        Dim workingJOEval As JO_Eval
        Set workingBin = OPEval_Col.Item(CStr(reportDate))
        workingBin.EarlyFinish = myearlyfinish

```

```

        Dim cpmReportArr() As Double
        ReDim cpmReportArr(0 To 100, 0 To 4)
        cpmReportArr = workingBin.DOCALC_CPMReport

```

```

        With Sheets("Expected-" & sheetName)
            .Cells.ClearContents
            .Cells(1, 1) = sheetName
            .Cells(2, 1) = "T"
            .Cells(2, 2) = "Date"
            .Cells(2, 3) = "EarlyHours"
            .Cells(2, 4) = "LateHours"
            .Cells(2, 5) = "MidHours"
            .Range("A3:E103") = cpmReportArr

            .Cells(2, 7) = "WBS"
            .Cells(2, 8) = "Operation"
            .Cells(2, 9) = "Duration"

```

```

.Cells(2, 10) = "Active"
.Cells(2, 11) = "ES"
.Cells(2, 12) = "EF"
.Cells(2, 13) = "LS"
.Cells(2, 14) = "LF"
.Cells(2, 15) = "Float"

Dim workingJO As JO_Class
Dim i As Integer
i = 3
For Each workingJO In JO_Col
    .Cells(i, 7) = workingJO.WBS_String
    .Cells(i, 8) = workingJO.Sequence
    .Cells(i, 9) = workingJO.EvalExpPoints(CStr(reportDate)).RemainingHours
    .Cells(i, 10) = CStr(workingJO.EvalExpPoints(CStr(reportDate)).OpActive)
    If workingJO.EvalExpPoints(CStr(reportDate)).RemainingHours > 0 Then
        .Cells(i, 11) =
Round(workingJO.EvalExpPoints(CStr(reportDate)).CPM_EarlyStart, 2)
        .Cells(i, 12) =
Round(workingJO.EvalExpPoints(CStr(reportDate)).CPM_EarlyFinish, 2)
        .Cells(i, 13) =
Round(workingJO.EvalExpPoints(CStr(reportDate)).CPM_LateStart, 2)
        .Cells(i, 14) =
Round(workingJO.EvalExpPoints(CStr(reportDate)).CPM_LateFinish, 2)
        .Cells(i, 15) =
Round(workingJO.EvalExpPoints(CStr(reportDate)).CPM_Float, 2)
    End If
    i = i + 1
Next workingJO
End With

eTime = Timer
MainFrm.LogString ("CPM Calculated in: " & Round((eTime - sTime) * 1000, 2) & "
mSec")
End Function

```

2. MainModule

The MainModule contains general use functions that are referenced elsewhere in the program. This is also where some key parameters are hard coded into the tool.

```
Option Explicit
Const hpw As Integer = 80 '80 work hours per week
```

```
Sub MainStart()
    MainFrm.Show
End Sub
```

```
Function ConvertDateToDouble(blDate As Date, convDate As Date) As Double
    ConvertDateToDouble = DateDiff("h", blDate, convDate) * (hpw / (24 * 7))
End Function
```

```
Function ConvertDoubleToDate(blDate As Date, convDouble As Double) As Date
    ConvertDoubleToDate = DateAdd("h", convDouble * ((24 * 7) / hpw), blDate)
End Function
```

3. ReportModule

The ReportModule consists of all the routines to generate the various reports generated by the tool.

```
Option Explicit
```

```
Sub EVMReport(allEvals As Collection, finalOp As JO_Class, tNow As Date, lowDate As Date, highDate As Date)
    Dim sTime, eTime As Double
    sTime = Timer
```

```
    With Sheets("Output-EVM")
        .Cells.ClearContents
        .Cells(1, 1) = "Earned Value Management"

        .Cells(3, 1) = "T"
        .Cells(3, 2) = "Date"
        .Cells(3, 3) = "BCWS"
        .Cells(3, 4) = "ACWP"
        .Cells(3, 5) = "BCWP"
        .Cells(3, 6) = "ACWP Future"
        .Cells(3, 7) = "BCWP Future"
        .Cells(3, 8) = "CV"
        .Cells(3, 9) = "CV%"
        .Cells(3, 10) = "SV"
```

```
.Cells(3, 11) = "SV%"
.Cells(3, 12) = "CPI"
.Cells(3, 13) = "SPI"
```

```
Dim i As Integer
i = 4
```

```
Dim maxHighDate As Date
maxHighDate = Application.WorksheetFunction.Max(highDate,
MainModule.ConvertDateToDouble(lowDate,
finalOp.EvalExpPoints(1).CPM_LateFinish))
```

```
Dim d As Date
Dim jo As JO_Eval
Dim joe As JO_Eval_Bin
Dim bcws, acwp, bcwp, acwp_f, bcwp_f As Double
Dim hasEval As Boolean
Dim maxCPISPI As Double
maxCPISPI = 0
```

```
For d = lowDate To maxHighDate
    hasEval = False
    .Cells(i, 1) = MainModule.ConvertDateToDouble(lowDate, d)
    .Cells(i, 2) = d
    bcws =
    (allEvals(1).BillableHoursAtT(MainModule.ConvertDateToDouble(lowDate, d), False) +
    allEvals(1).BillableHoursAtT(MainModule.ConvertDateToDouble(lowDate, d), True)) /
    2
    .Cells(i, 3) = bcws
```

```
For Each joe In allEvals
    If joe.reportDate = d Then
        acwp = 0
        bcwp = 0
        For Each jo In joe.Eval_Col
            acwp = acwp + jo.actualHours
            If (jo.OpActive = False) Then
                bcwp = bcwp + jo.baseHours
            End If
        Next jo
```

```

        .Cells(i, 4) = acwp
        .Cells(i, 5) = bcwp
        hasEval = True
        Exit For
    End If
Next joe

If hasEval Then
    .Cells(i, 8) = bcwp - acwp
    If bcwp > 0 Then
        .Cells(i, 9) = (bcwp - acwp) / bcwp
    End If
    .Cells(i, 10) = bcwp - bcws
    If (bcws > 0) Then
        .Cells(i, 11) = (bcwp - bcws) / bcws
    End If
    If (acwp > 0) Then
        .Cells(i, 12) = bcwp / acwp
        If maxCPISPI < bcwp / acwp Then maxCPISPI = bcwp / acwp
    End If
    If bcws > 0 Then
        .Cells(i, 13) = bcwp / bcws
        If maxCPISPI < bcwp / bcws Then maxCPISPI = bcwp / bcws
    End If
Else
    acwp_f =
    (allEvals(allEvals.Count).BillableHoursAtT(MainModule.ConvertDateToDouble(tNow,
    d), False) +
    allEvals(allEvals.Count).BillableHoursAtT(MainModule.ConvertDateToDouble(tNow,
    d), True)) / 2
    .Cells(i, 6) = acwp + acwp_f
    .Cells(i, 7) = bcwp + acwp_f
    .Cells(i, 8) = (bcwp + acwp_f) - (acwp + acwp_f)
    If (bcwp + acwp_f) > 0 Then
        .Cells(i, 9) = ((bcwp + acwp_f) - (acwp + acwp_f)) / (bcwp + acwp_f)
    End If
    .Cells(i, 10) = (bcwp + acwp_f) - bcws
    If (bcws > 0) Then
        .Cells(i, 11) = ((bcwp + acwp_f) - bcws) / bcws
    End If

```

```

        If ((acwp + acwp_f) > 0) Then
            .Cells(i, 12) = (bcwp + acwp_f) / (acwp + acwp_f)
            If maxCPISPI < (bcwp + acwp_f) / (acwp + acwp_f) Then maxCPISPI =
(bcwp + acwp_f) / (acwp + acwp_f)
            End If
            If bcws > 0 Then
                .Cells(i, 13) = (bcwp + acwp_f) / bcws
                If maxCPISPI < (bcwp + acwp_f) / bcws Then maxCPISPI = (bcwp +
acwp_f) / bcws
            End If
        End If

```

```

        i = i + 1
    Next d
    .Cells(1, 15) = "T Now"
    .Cells(1, 16) = tNow
    .Cells(2, 16) = tNow
    .Cells(1, 17) = 0
    .Cells(2, 17) = Application.WorksheetFunction.Max(bcwp + acwp_f, acwp +
acwp_f, bcws)
    .Cells(1, 18) = 0
    .Cells(2, 18) = maxCPISPI
    .Cells(1, 19) = -1
    .Cells(2, 19) = 1

    .Activate

```

End With

```

    eTime = Timer
    MainFrm.LogString ("EVM Report completed in: " & Round((eTime - sTime) * 1000,
2) & " mSec")
End Sub

```

```

Sub TrialReport(Trials_Col As Collection, Actuals_LowDate As Date,
Actuals_CurrentDate As Date)
    Dim sTime, eTime As Double

```

sTime = Timer

With Sheets("Output-TrialsHistogram")

.Cells.ClearContents

.Cells(1, 1) = "Trial Summary Output"

.Cells(2, 1) = "Trial"

.Cells(2, 2) = "Baseline Schedule Finish"

.Cells(2, 3) = "Baseline Finish Cost"

.Cells(2, 4) = "AtDate Schedule Finish"

.Cells(2, 5) = "AtDate Finish Cost"

Dim i, j, k As Integer

i = 3

Dim workingTrial As TO_Class

Dim d As Date

Dim blarr(), atarr() As Variant

ReDim blarr(0 To 1, 1 To Trials_Col.Count)

ReDim atarr(0 To 1, 1 To Trials_Col.Count)

j = 1

For Each workingTrial In Trials_Col

.Cells(i, 1) = workingTrial.trialNum

Dim blPoint As JO_Eval

Set blPoint = workingTrial.EvalTrialPoints.Item(CStr(DateAdd("d", -1, Actuals_LowDate)))

d = MainModule.ConvertDoubleToDate(Actuals_LowDate, blPoint.CPM_LateFinish) 'DateAdd("h", blPoint.CPM_LateFinish * ((24 * 7) / 80), Actuals_LowDate) '80 work hours per week

.Cells(i, 2) = Format(d, "yyyy/mm/dd")

blarr(0, j) = d

.Cells(i, 3) = workingTrial.BillableHrsAtD(DateAdd("D", -1, Actuals_LowDate))

blarr(1, j) = .Cells(i, 3)

Dim adPoint As JO_Eval

Set adPoint = workingTrial.EvalTrialPoints.Item(CStr(Actuals_CurrentDate))

d = MainModule.ConvertDoubleToDate(Actuals_CurrentDate, adPoint.CPM_LateFinish) 'DateAdd("h", adPoint.CPM_LateFinish * ((24 * 7) / 80), Actuals_CurrentDate) '80 work hours per week

.Cells(i, 4) = Format(d, "yyyy/mm/dd")

```

    atarr(0, j) = d
    .Cells(i, 5) = workingTrial.BillableHrsAtD(Actuals_CurrentDate)
    atarr(1, j) = .Cells(i, 5)
    i = i + 1
    j = j + 1
Next workingTrial

Dim blhist() As Variant
Dim bl_r As Double
blhist = ReportModule.Generate2DHistogram(blarr, bl_r)

Dim athist() As Variant
Dim at_r As Double
athist = ReportModule.Generate2DHistogram(atarr, at_r)

Dim chtobj As ChartObject
For Each chtobj In .ChartObjects
    Select Case chtobj.Chart.ChartTitle.Text
        Case "Baseline Schedule Finish"
            chtobj.Chart.SetSourceData .Range("B2:B" & i - 1)
        Case "Baseline Final Cost"
            chtobj.Chart.SetSourceData .Range("C2:C" & i - 1)
        Case "AtDate Schedule Finish"
            chtobj.Chart.SetSourceData .Range("D2:D" & i - 1)
        Case "AtDate Final Cost"
            chtobj.Chart.SetSourceData .Range("E2:E" & i - 1)
    End Select
Next chtobj

End With

With Sheets("Output-TrialsCDF")
    .Cells.ClearContents
    .Cells(1, 1) = "Trails CDF"
    .Cells(3, 1) = "Baseline"
    .Cells(4, 1) = "Date"
    .Cells(4, 2) = "Cuml Occurances"
    .Cells(4, 3) = "Cuml Prob"
    .Cells(4, 5) = "Cost"
    .Cells(4, 6) = "Cuml Occurances"

```

```
.Cells(4, 7) = "Cuml Prob"
```

```
.Cells(3, 9) = "At Time"
```

```
.Cells(4, 9) = "Date"
```

```
.Cells(4, 10) = "Cuml Occurances"
```

```
.Cells(4, 11) = "Cuml Prob"
```

```
.Cells(4, 13) = "Cost"
```

```
.Cells(4, 14) = "Cuml Occurances"
```

```
.Cells(4, 15) = "Cuml Prob"
```

```
Dim minCost, maxCost As Double
```

```
Dim minSched, maxSched As Date
```

```
'baseline calcs
```

```
minSched = blarr(0, 1)
```

```
maxSched = blarr(0, 1)
```

```
minCost = blarr(1, 1)
```

```
maxCost = blarr(1, 1)
```

```
Dim trialsI As Integer
```

```
For trialsI = 1 To UBound(blarr, 2) 'calculate the min and max of cost and schedule
```

```
    If minSched > blarr(0, trialsI) Then
```

```
        minSched = blarr(0, trialsI)
```

```
    End If
```

```
    If maxSched < blarr(0, trialsI) Then
```

```
        maxSched = blarr(0, trialsI)
```

```
    End If
```

```
    If minCost > blarr(1, trialsI) Then
```

```
        minCost = blarr(1, trialsI)
```

```
    End If
```

```
    If maxCost < blarr(1, trialsI) Then
```

```
        maxCost = blarr(1, trialsI)
```

```
    End If
```

```
Next trialsI
```

```
Dim trialsD As Date
```

```
trialsD = minSched
```

```
Dim totalCountD As Double
```

```
totalCountD = 0
```

```

Dim rowCountTrials As Double
rowCountTrials = 5
While trialsD <= DateAdd("d", 1, maxSched)
    .Cells(rowCountTrials, 1) = trialsD
    totalCountD = 0
    For trialsI = 1 To UBound(blarr, 2)
        If blarr(0, trialsI) <= trialsD Then
            totalCountD = totalCountD + 1
        End If
    Next trialsI
    .Cells(rowCountTrials, 2) = totalCountD
    .Cells(rowCountTrials, 3) = totalCountD / (UBound(blarr, 2) - LBound(blarr, 2))
    rowCountTrials = rowCountTrials + 1
    trialsD = DateAdd("d", 1, trialsD)
Wend

```

```

Dim trialsC As Double
Dim totalCountC As Double
trialsC = minCost
Dim spanC As Double
spanC = (maxCost - minCost) / 25
rowCountTrials = 5
If spanC > 0 Then
    While trialsC <= maxCost + spanC
        .Cells(rowCountTrials, 5) = trialsC
        totalCountC = 0
        For trialsI = 1 To UBound(blarr, 2)
            If blarr(1, trialsI) <= trialsC Then
                totalCountC = totalCountC + 1
            End If
        Next trialsI
        .Cells(rowCountTrials, 6) = totalCountC
        .Cells(rowCountTrials, 7) = totalCountC / (UBound(blarr, 2) - LBound(blarr,
2))
        rowCountTrials = rowCountTrials + 1
        trialsC = trialsC + spanC
    Wend
Else
    .Cells(rowCountTrials, 5) = trialsC
    totalCountC = UBound(blarr, 2) - LBound(blarr, 2)

```

```

.Cells(rowCountTrials, 6) = totalCountC
.Cells(rowCountTrials, 7) = 1
End If
'at time calcs

minSched = atarr(0, 1)
maxSched = atarr(0, 1)
minCost = atarr(1, 1)
maxCost = atarr(1, 1)

For trialsI = 1 To UBound(atarr, 2) 'calculate the min and max of cost and schedule
    If minSched > atarr(0, trialsI) Then
        minSched = atarr(0, trialsI)
    End If
    If maxSched < atarr(0, trialsI) Then
        maxSched = atarr(0, trialsI)
    End If
    If minCost > atarr(1, trialsI) Then
        minCost = atarr(1, trialsI)
    End If
    If maxCost < atarr(1, trialsI) Then
        maxCost = atarr(1, trialsI)
    End If
Next trialsI

trialsD = minSched

totalCountD = 0

rowCountTrials = 5
While trialsD <= DateAdd("d", 1, maxSched)
    .Cells(rowCountTrials, 9) = trialsD
    totalCountD = 0
    For trialsI = 1 To UBound(atarr, 2)
        If atarr(0, trialsI) <= trialsD Then
            totalCountD = totalCountD + 1
        End If
    Next trialsI
    .Cells(rowCountTrials, 10) = totalCountD
    .Cells(rowCountTrials, 11) = totalCountD / (UBound(atarr, 2) - LBound(atarr, 2))

```

```

        rowCountTrials = rowCountTrials + 1
        trialsD = DateAdd("d", 1, trialsD)
    Wend

    trialsC = minCost
    spanC = (maxCost - minCost) / 25
    rowCountTrials = 5
    If spanC > 0 Then
        While trialsC <= maxCost + spanC
            .Cells(rowCountTrials, 13) = trialsC
            totalCountC = 0
            For trialsI = 1 To UBound(atarr, 2)
                If atarr(1, trialsI) <= trialsC Then
                    totalCountC = totalCountC + 1
                End If
            Next trialsI
            .Cells(rowCountTrials, 14) = totalCountC
            .Cells(rowCountTrials, 15) = totalCountC / (UBound(atarr, 2) - LBound(atarr,
2))

            rowCountTrials = rowCountTrials + 1
            trialsC = trialsC + spanC
        Wend
    Else
        .Cells(rowCountTrials, 5) = trialsC
        totalCountC = UBound(blarr, 2) - LBound(blarr, 2)
        .Cells(rowCountTrials, 6) = totalCountC
        .Cells(rowCountTrials, 7) = 1
    End If
End With

With Sheets("Output-BaselineTrialResults")
    .Cells.ClearContents
    .Cells(1, 1) = "Baseline Histogram"
    .Cells(2, 1) = "Low Bound"
    .Cells(3, 2) = "High Bound"
    .Cells(4, 3) = "Label"
    Dim rn, cn As Integer

    rn = 5
    For j = LBound(blhist, 1) To UBound(blhist, 1)

```

```

.Cells(rn, 1) = Round(blhist(j, 1, 1), 2)
.Cells(rn, 2) = Round(blhist(j, 1, 2), 2)
.Cells(rn, 3) = Round((blhist(j, 1, 1) + blhist(j, 1, 2)) / 2, 2)
rn = rn + 1
Next j

cn = 4
For k = LBound(blhist, 2) To UBound(blhist, 2)
    .Cells(2, cn) = Round(blhist(1, k, 3), 2)
    .Cells(3, cn) = Round(blhist(1, k, 4), 2)
    .Cells(4, cn) = Round((blhist(1, k, 3) + blhist(1, k, 4)) / 2, 2)
    cn = cn + 1
Next k

For j = LBound(blhist, 1) To UBound(blhist, 1)
    For k = LBound(blhist, 2) To UBound(blhist, 2)
        .Cells(4 + j, 3 + k) = blhist(j, k, 5)
    Next k
Next j

For Each chtobj In .ChartObjects
    chtobj.Delete
Next chtobj

.Cells(1, cn + 1) = bl_r
Set chtobj = .ChartObjects.Add(Left:=.Cells(1, cn + 1).Left, Top:=.Cells(2, 1).Top,
Width:=450, Height:=250)

Dim mycht As Chart
Set mycht = chtobj.Chart

mycht.SetSourceData Range(.Cells(5, 4), .Cells(rn - 1, cn - 1)), xlColumns

Dim myaxis As Axis
Set myaxis = mycht.Axes(xlCategory)
myaxis.CategoryNames = Range(.Cells(5, 3), .Cells(rn - 1, 3))

Dim myseries As Series
Dim c As Integer
c = 4

```

```

For Each myseries In mycht.SeriesCollection
    myseries.Name = .Cells(4, c)
    c = c + 1
Next myseries
mycht.ChartType = xlSurfaceTopView
mycht.PlotArea.Format.ThreeD.RotationX = 0
End With

With Sheets("Output-AtTimeTrialResults")
    .Cells.ClearContents
    .Cells(1, 1) = "At Time Histogram"
    .Cells(2, 1) = "Low Bound"
    .Cells(3, 2) = "High Bound"
    .Cells(4, 3) = "Label"

    rn = 5
    For j = LBound(athist, 1) To UBound(athist, 1)
        .Cells(rn, 1) = Round(athist(j, 1, 1), 2)
        .Cells(rn, 2) = Round(athist(j, 1, 2), 2)
        .Cells(rn, 3) = Round((athist(j, 1, 1) + athist(j, 1, 2)) / 2, 2)
        rn = rn + 1
    Next j

    cn = 4
    For k = LBound(athist, 2) To UBound(athist, 2)
        .Cells(2, cn) = Round(athist(1, k, 3), 2)
        .Cells(3, cn) = Round(athist(1, k, 4), 2)
        .Cells(4, cn) = Round((athist(1, k, 3) + athist(1, k, 4)) / 2, 2)
        cn = cn + 1
    Next k

    For j = LBound(athist, 1) To UBound(athist, 1)
        For k = LBound(athist, 2) To UBound(athist, 2)
            .Cells(4 + j, 3 + k) = athist(j, k, 5)
        Next k
    Next j

    For Each chtobj In .ChartObjects
        chtobj.Delete
    Next chtobj

```

```

.Cells(1, cn + 1) = at_r
Set chtobj = .ChartObjects.Add(Left:=.Cells(1, cn + 1).Left, Top:=.Cells(2, 1).Top,
Width:=450, Height:=250)

Set mycht = chtobj.Chart

mycht.SetSourceData Range(.Cells(5, 4), .Cells(rn - 1, cn - 1)), xlColumns

Set myaxis = mycht.Axes(xlCategory)
myaxis.CategoryNames = Range(.Cells(5, 3), .Cells(rn - 1, 3))

c = 4
For Each myseries In mycht.SeriesCollection
    myseries.Name = .Cells(4, c)
    c = c + 1
Next myseries
mycht.ChartType = xlSurfaceTopView
mycht.PlotArea.Format.ThreeD.RotationX = 0
End With

eTime = Timer
MainFrm.LogString ("Trial Reports completed in: " & Round((eTime - sTime) * 1000,
2) & " mSec")
End Sub

Sub OperationReport(op As JO_Class)
    Dim i As Integer
    i = 1
    With Sheets("Output-OperationDetailReport")
        .Cells.ClearContents
        .Cells(i, 1) = "Operation Detail Report"
        i = i + 1
        .Cells(i, 1) = "WBS:"
        .Cells(i, 2) = op.WBS_String
        i = i + 1
        .Cells(i, 1) = "Job Operation:"
        .Cells(i, 2) = op.JobOperation
        i = i + 1
        .Cells(i, 1) = "Estimation Method:"

```

```

.Cells(i, 2) = op.Estimate.estimateMode
i = i + 1
.Cells(i, 1) = "Estimation - " & op.Estimate.parameterName(0) & ":"
.Cells(i, 2) = op.Estimate.p0
i = i + 1
.Cells(i, 1) = "Estimation - " & op.Estimate.parameterName(1) & ":"
.Cells(i, 2) = op.Estimate.p1
i = i + 1
.Cells(i, 1) = "Estimation - " & op.Estimate.parameterName(2) & ":"
.Cells(i, 2) = op.Estimate.p2
i = i + 1
.Cells(i, 1) = "Estimation - " & op.Estimate.parameterName(3) & ":"
.Cells(i, 2) = op.Estimate.p3
i = i + 2
.Cells(i, 1) = "Actual:"
.Cells(i, 2) = op.ActualsEvalVal

i = 1
.Cells(i, 4) = "trial"
.Cells(i, 5) = "value"

i = i + 1
Dim workingTrial As TO_Class
For Each workingTrial In op.Trials
    .Cells(i, 4) = workingTrial.trialNum
    .Cells(i, 5) = workingTrial.trialHrs
    i = i + 1
Next workingTrial

Dim chtobj As ChartObject
For Each chtobj In .ChartObjects
    chtobj.Chart.SetSourceData .Range("E1:E" & i - 1)
    chtobj.Chart.Refresh

Next chtobj
End With
End Sub

Sub CP_Sensitivity(WBS_Col As Collection, rptDate As Date)
    Dim sTime, eTime As Double

```

sTime = Timer

```
Dim workingWBS As WBS_Class
Dim workingOP As JO_Class
Dim workingTrial As TO_Class
Dim workingEval As JO_Eval
Dim maxfloat, minfloat As Double
Dim wbsFloatCol As Collection
Dim i, c_gz, c_ez As Integer
```

i = 1

With Sheets("Output-CPSensitivity")

.Cells.ClearContents

.Cells(i, 1) = "Critical Path Sensitivity as of: " & Format(rptDate, "yyyy/mm/dd")

i = i + 2

.Cells(i, 1) = "WBS"

.Cells(i, 2) = "Max Float"

.Cells(i, 3) = "Min Float"

.Cells(i, 4) = "Average"

.Cells(i, 5) = "Std. Dev."

.Cells(i, 6) = "Count > 0"

.Cells(i, 7) = "Count = 0"

i = i + 1

For Each workingWBS In WBS_Col

Set wbsFloatCol = New Collection

c_gz = 0 'reset count greater than zero

c_ez = 0 'reset count equals zero

Set workingOP = workingWBS.Op_Last

For Each workingTrial In workingOP.Trials

Set workingEval = workingTrial.EvalTrialPoints.Item(CStr(rptDate))

If workingEval.OpActive = True Then

wbsFloatCol.Add workingEval.CPM_Float

If Round(workingEval.CPM_Float, 2) > 0 Then

c_gz = c_gz + 1

Else

c_ez = c_ez + 1

End If

```

        End If
    Next workingTrial
    .Cells(i, 1) = workingWBS.WBS_String
    If wbsFloatCol.Count > 0 Then
        .Cells(i, 2) =
Round(Application.WorksheetFunction.Max(CollectionToArray(wbsFloatCol)), 2)
        .Cells(i, 3) =
Round(Application.WorksheetFunction.Min(CollectionToArray(wbsFloatCol)), 2)
        .Cells(i, 4) =
Round(Application.WorksheetFunction.Average(CollectionToArray(wbsFloatCol)), 2)
        .Cells(i, 5) =
Round(Application.WorksheetFunction.StDev_S(CollectionToArray(wbsFloatCol)), 2)
        .Cells(i, 6) = c_gz
        .Cells(i, 7) = c_ez
    End If
    i = i + 1
Next workingWBS

```

End With

```

eTime = Timer
MainFrm.LogString ("CP Sensitivity Reports completed in: " & Round((eTime -
sTime) * 1000, 2) & " mSec")
End Sub

```

```

Function Generate2DHistogram(valArr() As Variant, ByRef r As Double) As Variant()
    Dim ar1(), ar2() As Double
    ReDim ar1(UBound(valArr, 2) - LBound(valArr, 2))
    ReDim ar2(UBound(valArr, 2) - LBound(valArr, 2))

    Dim i As Integer
    For i = 0 To UBound(valArr, 2) - LBound(valArr, 2)
        ar1(i) = CDBl(valArr(0, i + LBound(valArr, 2)))
        ar2(i) = CDBl(valArr(1, i + LBound(valArr, 2)))
    Next i

    Dim ar1_min, ar1_max, ar2_min, ar2_max As Variant
    ar1_min = Application.WorksheetFunction.Min(ar1)
    ar1_max = Application.WorksheetFunction.Max(ar1)

```

```

If ar1_max = ar1_min Then ar1_max = ar1_min + 1
ar2_min = Application.WorksheetFunction.Min(ar2)
ar2_max = Application.WorksheetFunction.Max(ar2)
If ar2_max = ar2_min Then ar2_max = ar2_min + 1

'determine number of bins
Dim ar1_kd, ar2_kd As Double
Dim ar1_ki, ar2_ki As Integer

ar1_kd = 1.88 * ((UBound(ar1) - LBound(ar1)) ^ 0.4) '1 + (3.322 *
Application.WorksheetFunction.Log(UBound(ar1) - LBound(ar1) + 1))
ar1_ki = CInt(Round(ar1_kd, 0))
ar2_kd = 1.88 * ((UBound(ar2) - LBound(ar2)) ^ 0.4) '1 + (3.322 *
Application.WorksheetFunction.Log(UBound(ar2) - LBound(ar2) + 1))
ar2_ki = CInt(Round(ar2_kd, 0))

'determine bin size
Dim ar1_binsize, ar2_binsize As Double
ar1_binsize = (ar1_max - ar1_min) / ar1_ki
ar2_binsize = (ar2_max - ar2_min) / ar2_ki

'initialize bins
Dim bins() As Variant
ReDim bins(1 To ar1_ki, 1 To ar2_ki, 1 To 5)

Dim j, k As Integer
For j = 1 To ar1_ki
    For k = 1 To ar2_ki
        bins(j, k, 1) = ar1_min + ((j - 1) * ar1_binsize) 'min 1
        bins(j, k, 2) = ar1_min + ((j) * ar1_binsize) 'max 1
        bins(j, k, 3) = ar2_min + ((k - 1) * ar2_binsize) 'min 2
        bins(j, k, 4) = ar2_min + ((k) * ar2_binsize) 'max 2
        bins(j, k, 5) = 0
    Next k
Next j

For i = LBound(valArr, 2) To UBound(valArr, 2)
    j = Round(((CDBl(valArr(0, i)) - ar1_min) / ar1_binsize) + 1, 0)
    If j > ar1_ki Then j = ar1_ki
    k = Round(((CDBl(valArr(1, i)) - ar2_min) / ar2_binsize) + 1, 0)

```

```

        If k > ar2_ki Then k = ar2_ki
        bins(j, k, 5) = bins(j, k, 5) + 1
    Next i

'calculate r value using Pearson's Correlation Coefficient
Dim ar1_mean, ar2_mean As Double
Dim sumTerm1, sumTerm2, sumTerm3 As Double
ar1_mean = Application.WorksheetFunction.Average(ar1)
ar2_mean = Application.WorksheetFunction.Average(ar2)
For i = 0 To UBound(valArr, 2) - LBound(valArr, 2)
    sumTerm1 = sumTerm1 + ((ar1(i) - ar1_mean) * (ar2(i) - ar2_mean))
    sumTerm2 = sumTerm2 + ((ar1(i) - ar1_mean) ^ 2)
    sumTerm3 = sumTerm3 + ((ar2(i) - ar2_mean) ^ 2)
Next i

If sumTerm3 = 0 Or sumTerm2 = 0 Then
    r = 0
Else
    r = sumTerm1 / (Sqr(sumTerm2) * Sqr(sumTerm3))
End If

Generate2DHistogram = bins
End Function

Function GenerateHistogram(valArr() As Double) As Double()
    Dim low_bound, upper_bound As Double
    low_bound = 9999999
    upper_bound = -9999999

'get bounds of array
Dim d As Variant
For Each d In valArr
    If low_bound > d Then
        low_bound = d
    End If
    If upper_bound < d Then
        upper_bound = d
    End If
Next d

```

```

'lower low and raise upper by a small amount
low_bound = low_bound * 0.99
upper_bound = upper_bound * 1.01

'determine number of bins
Dim k_d As Double
k_d = 1 + (3.322 * Application.WorksheetFunction.Log(UBound(valArr) -
LBound(valArr) + 1))
Dim k_i As Integer
k_i = CInt(Round(k_d, 0))

'generate bins
Dim binSize As Double
binSize = (upper_bound - low_bound) / k_i

Dim bins() As Double
ReDim bins(0 To 3, 1 To k_i)

Dim i As Integer
For i = 1 To k_i
    bins(0, i) = low_bound + ((i - 1) * binSize) 'low bound
    bins(1, i) = low_bound + (i * binSize) 'up bound
    bins(2, i) = (bins(0, i) + bins(1, i)) / 2 'x axis
    bins(3, i) = 0 'initialize count
Next i

'populate bins
For Each d In valArr
    For i = 1 To k_i
        If d >= bins(0, i) And d < bins(1, i) Then
            bins(3, i) = bins(3, i) + 1
            Exit For
        End If
    Next i
Next d

GenerateHistogram = bins
End Function

```

Public Function CollectionToArray(myCol As Collection) As Variant

```

Dim result As Variant
Dim cnt As Long

If Not myCol Is Nothing Then
    If myCol.Count > 0 Then
        ReDim result(myCol.Count - 1)
        For cnt = 0 To myCol.Count - 1
            result(cnt) = myCol(cnt + 1)
        Next cnt
        CollectionToArray = result
    End If
End If
End Function

```

4. TrialModule

The TrialModule is where the logic exists to generate and to populate the selected number of trials. In this module, the CPM techniques are also initialized across the trials.

Option Explicit

```

Function LoadTrials(numTrials As Integer, thisOp As JO_Class, firstDate As Date,
lastDate As Date) As Collection
    Dim sTime, eTime As Double
    sTime = Timer

    Set LoadTrials = New Collection

    Dim newTO As TO_Class
    Dim workingJO As JO_Class

    'reset all trial collections
    thisOp.ResetTrials

    'populate Trial data
    Dim i As Integer
    For i = 1 To numTrials
        Set newTO = TrialModule.BuildTrial(i, thisOp, firstDate, lastDate)
        LoadTrials.Add newTO, CStr(i)
    Next i

```

```

Dim workingTrial As TO_Class

'execute baseline CPM
Dim blDate As Date
blDate = DateAdd("d", -1, firstDate)
Dim blEvalPt As JO_Eval
For Each workingTrial In LoadTrials
    'forwards
    workingTrial.DOCALC_CPM_Forwards (blDate)
    'transition
    Set blEvalPt = workingTrial.EvalTrialPoints.Item(CStr(blDate))
    blEvalPt.CPM_LateFinish = blEvalPt.CPM_EarlyFinish
    blEvalPt.CPM_LateStart = blEvalPt.CPM_LateFinish - blEvalPt.RemainingHours
    blEvalPt.CPM_Float = blEvalPt.CPM_LateFinish - blEvalPt.CPM_EarlyFinish
    'backwards
    workingTrial.DOCALC_CPM_Backwards (blDate)
    'end bl eval
Next workingTrial

'execute final CPM
Dim fnDate As Date
fnDate = lastDate
Dim fnEvalPt As JO_Eval
For Each workingTrial In LoadTrials
    'forwards
    workingTrial.DOCALC_CPM_Forwards (fnDate)
    'transition
    Set fnEvalPt = workingTrial.EvalTrialPoints.Item(CStr(fnDate))
    fnEvalPt.CPM_LateFinish = fnEvalPt.CPM_EarlyFinish
    fnEvalPt.CPM_LateStart = fnEvalPt.CPM_LateFinish - fnEvalPt.RemainingHours
    fnEvalPt.CPM_Float = fnEvalPt.CPM_LateFinish - fnEvalPt.CPM_EarlyFinish
    'backwards
    workingTrial.DOCALC_CPM_Backwards (fnDate)
    'end fn eval
Next workingTrial

eTime = Timer
MainFrm.LogString ("Trials Generated in: " & Round((eTime - sTime) * 1000, 2) & "
mSec")
End Function

```

```

Function BuildTrial(trialNum As Integer, thisJO As JO_Class, firstDate As Date,
lastDate As Date) As TO_Class
    'recursive function to build shadow structure to JO for the specific trial

    Dim newTO As TO_Class
    Set newTO = New TO_Class
    Set newTO.ParentJO = thisJO
    newTO.trialNum = trialNum
    newTO.trialHrs = newTO.ParentJO.Estimate.TrialValue
    Set newTO.TO_GlobalPrevs = New Collection
    Set newTO.EvalTrialPoints = New Collection
    Dim rDate As Date
    rDate = DateAdd("d", -1, firstDate)
    Dim evalPt As JO_Eval

    'generate baseline rDate
    Set evalPt = New JO_Eval
    evalPt.reportDate = rDate
    evalPt.baseHours = newTO.trialHrs
    Set evalPt.ParentTO = newTO
    Set evalPt.ParentOp = thisJO
    newTO.EvalTrialPoints.Add evalPt, CStr(rDate)
    'generate lastDate rDate
    rDate = lastDate
    Set evalPt = New JO_Eval
    evalPt.reportDate = rDate
    evalPt.baseHours = newTO.trialHrs
    Set evalPt.ParentTO = newTO
    Set evalPt.ParentOp = thisJO
    newTO.EvalTrialPoints.Add evalPt, CStr(rDate)

    thisJO.Trials.Add newTO, CStr(trialNum)
    'Build Child trials
    Dim workingJO As JO_Class
    For Each workingJO In thisJO.Op_GlobalPrevs
        Dim thisPrevTO As TO_Class
        Set thisPrevTO = TrialModule.BuildTrial(trialNum, workingJO, firstDate, lastDate)
    
```

```
Set thisPrevTO.TO_GlobalNext = newTO
newTO.TO_GlobalPrevs.Add thisPrevTO
Next workingJO
```

```
Set BuildTrial = newTO
End Function
```

```
Function PerformCPM(finalTrial As TO_Class, reportDate As Date, sheetName As
String) As Collection
    Dim sTime, eTime As Double
    sTime = Timer

    finalTrial.DOCALC_CPM_Forwards (reportDate)

    eTime = Timer
    MainFrm.LogString ("Trials CPM Calculated in: " & Round((eTime - sTime) * 1000,
2) & " mSec")
End Function
```

C. CLASS MODULES

Class Modules are instantiable objects that represent different aspects of the data, captured as properties. An instance of a class could contain methods to transform data or have linkages to other classes of information. An example of this is with the WBS Class. The Top-Level WBS is defined as an instance of this class. Within it, there is a property for its name. Additionally, it consists of a collection of other WBS classes which would represent all the Top-Level WBS's directly lower level WBS—each of which has its own name property.

1. JO_Class

The Job Operation Class (JO_Class) is an instantiable class representing a Job Operation (or an activity on the Activity Network Diagram). It consists of numerous properties and links to estimation and WBS classes. It also contains CPM functions and other reporting functions.

Option Explicit

```
Public JobOperation, WBS_String, OperationName As String
Public Sequence As Integer
Public NumResources As Double
```

```
Public Op_ThisNext As JO_Class
Public Op_ThisPrev As JO_Class
Public Op_GlobalNext As JO_Class
Public Op_GlobalPrevs As Collection
Public WBS As WBS_Class
```

```
Public Estimate As JO_Estimation
Public Actuals_All As Collection
Public Actuals_Eval As Collection
```

```
Public EvalExpPoints As Collection
Public Trials As Collection
```

```
Public Property Get Op_First() As JO_Class
    If Op_ThisPrev Is Nothing Then
        Set Op_First = Me
    Else
        Set Op_First = Op_ThisPrev.Op_First
    End If
End Property
```

```
Public Property Get Op_Last() As JO_Class
    If Op_ThisNext Is Nothing Then
        Set Op_Last = Me
    Else
        Set Op_Last = Op_ThisNext.Op_Last
    End If
End Property
```

```
Public Function PrintStructure() As String
    PrintStructure = JobOperation
```

```
    Dim JOChild As JO_Class
    For Each JOChild In Op_GlobalPrevs
```

```

        PrintStructure = PrintStructure & vbCrLf & vbTab &
Replace(JOChild.PrintStructure, vbCrLf, vbCrLf & vbTab)
    Next JOChild
End Function

```

```

Public Sub DOCALC_ActualHrs(rDate As Date)
    Dim myCalcActualHours As Double
    myCalcActualHours = 0
    If Not Actuals_Eval Is Nothing Then
        Dim jotBuilder As JOT_Class
        For Each jotBuilder In Actuals_Eval
            If jotBuilder.WorkDate <= rDate Then
                myCalcActualHours = Round(myCalcActualHours + jotBuilder.BillableHours,
2)
            End If
        Next jotBuilder
    End If
    Dim evalBuilder As JO_Eval
    Set evalBuilder = EvalExpPoints.Item(CStr(rDate))
    evalBuilder.actualHours = myCalcActualHours
End Sub

```

```

Public Function DOCALC_OpActive(nextWithAct As Boolean, rDate As Date) As
Boolean
    Dim myEvalPoint As JO_Eval
    Set myEvalPoint = EvalExpPoints.Item(CStr(rDate))

    myEvalPoint.baseHours = Estimate.expectedValue

    Dim doIHaveAct As Boolean
    doIHaveAct = nextWithAct

    If nextWithAct = True Then
        myEvalPoint.OpActive = False
        myEvalPoint.RemainingHours = 0
    Else
        myEvalPoint.OpActive = True
        If myEvalPoint.actualHours > 0 Then
            doIHaveAct = True
        End If
    End If

```

```

        myEvalPoint.RemainingHours = Round(Application.WorksheetFunction.Max(0,
myEvalPoint.baseHours - myEvalPoint.actualHours), 2)
    End If

```

```

    Dim opBuilder As JO_Class
    For Each opBuilder In Op_GlobalPrevs
        opBuilder.DOCALC_OpActive doIHaveAct, rDate
    Next opBuilder

```

```

End Function

```

```

Public Sub DOCALC_CPM_Forwards(rDate As Date)
    Dim workingJO As JO_Class
    Dim myES As Double
    myES = 0
    If Op_GlobalPrevs.Count > 0 Then
        For Each workingJO In Op_GlobalPrevs
            workingJO.DOCALC_CPM_Forwards (rDate)
            myES = Application.WorksheetFunction.Max(myES,
workingJO.EvalExpPoints.Item(CStr(rDate)).CPM_EarlyFinish)
        Next workingJO
    End If

```

```

    Dim myEP As JO_Eval
    Set myEP = EvalExpPoints.Item(CStr(rDate))
    myEP.CPM_EarlyStart = myES

```

```

    myEP.CPM_EarlyFinish = myEP.CPM_EarlyStart + myEP.RemainingHours
End Sub

```

```

Public Sub DOCALC_CPM_Backwards(rDate As Date)
    Dim workingJO As JO_Class
    If Op_GlobalPrevs.Count > 0 Then
        For Each workingJO In Op_GlobalPrevs
            workingJO.EvalExpPoints.Item(CStr(rDate)).CPM_LateFinish =
EvalExpPoints.Item(CStr(rDate)).CPM_LateStart
            workingJO.EvalExpPoints.Item(CStr(rDate)).CPM_LateStart =
workingJO.EvalExpPoints.Item(CStr(rDate)).CPM_LateFinish -
workingJO.EvalExpPoints.Item(CStr(rDate)).RemainingHours

```

```

        workingJO.EvalExpPoints.Item(CStr(rDate)).CPM_Float =
workingJO.EvalExpPoints.Item(CStr(rDate)).CPM_LateStart -
workingJO.EvalExpPoints.Item(CStr(rDate)).CPM_EarlyStart
        workingJO.DOCALC_CPM_Backwards (rDate)
    Next workingJO
End If
End Sub

```

```

Public Sub DOCALC_CPM_TrialForwards(rDate As Date)
    MsgBox ("Shouldnt be here")
End Sub

```

```

Public Property Get ActualsEvalVal() As Double
    ActualsEvalVal = 0
    Dim workingJOT As JOT_Class
    For Each workingJOT In Actuals_Eval
        ActualsEvalVal = ActualsEvalVal + workingJOT.BillableHours
    Next workingJOT
End Property

```

```

Public Sub ResetTrials()
    Set Trials = New Collection
    Dim workingJO As JO_Class
    For Each workingJO In Op_GlobalPrevs
        workingJO.ResetTrials
    Next workingJO
End Sub

```

2. JO_Estimation

This instantiable class is used to track estimation properties for each operation (JO_Class) and contains methods to calculate a trial value and expected value based on the inputs.

Option Explicit

```

Public mode As Integer
Public p0, p1, p2, p3 As Double

```

```

Property Get TrialValue() As Double

```

```

Randomize
Dim myRand As Single
myRand = Rnd()

Select Case mode
    Case 0 'StraightExpectation
        TrialValue = p0
    Case 1 'Lognormal
        TrialValue = WorksheetFunction.LogNorm_Inv(myRand, p0, Sqr(p1))
    Case 2 'Triangular
        If (p2 - p0 = 0) Then
            TrialValue = p0
        Else
            If (myRand <= (p1 - p0) / (p2 - p0)) Then
                TrialValue = p0 + Sqr(myRand * (p2 - p0) * (p1 - p0))
            Else
                TrialValue = p2 - Sqr((1 - myRand) * (p2 - p0) * ((-1 * p1) + p2))
            End If
        End If
    Case 3 'BetaPert
        Dim myAlpha, myBeta As Double
        myAlpha = (4 * (p1 - p0) / (p2 - p0)) + 1
        myBeta = (4 * (p2 - p1) / (p2 - p0)) + 1
        TrialValue = WorksheetFunction.Beta_Inv(myRand, myAlpha, myBeta, p0, p2)
    Case 4 'Beta
        TrialValue = WorksheetFunction.Beta_Inv(myRand, p0, p1, p2, p3)
    Case 5 'Normal
        TrialValue = WorksheetFunction.Norm_Inv(myRand, p0, Sqr(p1))
    Case 6 'Uniform
        TrialValue = p0 + myRand * (p1 - p0)
End Select
TrialValue = Round(TrialValue, 2)
End Property

Property Get expectedValue() As Double
    Select Case mode
        Case 0 'StraightExpectation
            expectedValue = p0
        Case 1 'Lognormal
            expectedValue = Exp(p0 + (p1 / 2))

```

```

Case 2 'Triangular
    expectedValue = (p0 + p1 + p2) / 3
Case 3 'BetaPert
    expectedValue = (p0 + p1 + (4 * p2)) / 6
Case 4 'Beta
    expectedValue = (p1 / (p0 + p1)) * (p3 - p2)
Case 5 'Normal
    expectedValue = p0
Case 6 'Uniform
    expectedValue = (p0 + p1) / 2
End Select
expectedValue = Round(expectedValue, 2)
End Property

```

```

Property Get estimationMode() As String
    Select Case mode
        Case 0
            estimationMode = "Straight Expectation"
        Case 1
            estimationMode = "Lognormal"
        Case 2
            estimationMode = "Triangular"
        Case 3
            estimationMode = "Beta Pert"
        Case 4
            estimationMode = "Beta"
        Case 5
            estimationMode = "Normal"
        Case 6
            estimationMode = "Uniform"
    End Select
End Property

```

```

Property Get parameterName(i As Integer) As String
    Select Case mode
        Case 0 'Straight Expectation
            Select Case i
                Case 0
                    parameterName = "Value"
            Case Else

```

```

        parameterName = ""
    End Select
Case 1 'Lognormal
    Select Case i
        Case 0
            parameterName = "Mean"
        Case 1
            parameterName = "Variance"
        Case Else
            parameterName = ""
    End Select
Case 2 'Triangular
    Select Case i
        Case 0
            parameterName = "Lower"
        Case 1
            parameterName = "Most Likely"
        Case 2
            parameterName = "Upper"
        Case Else
            parameterName = ""
    End Select
Case 3 'Beta Pert
    Select Case i
        Case 0
            parameterName = "Lower"
        Case 1
            parameterName = "Most Likely"
        Case 2
            parameterName = "Upper"
        Case Else
            parameterName = ""
    End Select
Case 4 'Beta
    Select Case i
        Case 0
            parameterName = "Alpha"
        Case 1
            parameterName = "Beta"
        Case 2

```

```

        parameterName = "Low"
    Case 3
        parameterName = "High"
    Case Else
        parameterName = ""
    End Select
Case 5 'Normal
Select Case i
    Case 0
        parameterName = "Mean"
    Case 1
        parameterName = "Variance"
    Case Else
        parameterName = ""
    End Select
Case 6 'Uniform
Select Case i
    Case 0
        parameterName = "Min"
    Case 1
        parameterName = "Max"
    Case Else
        parameterName = ""
    End Select
End Select
End Property

```

3. JO_Eval

The JO_Eval class is an instantiable class that collects various time-sensitive properties of a Job Operation Class (JO_Class). Some properties include the remaining hours, actual hours, and CPM properties.

Option Explicit

Public reportDate As Date

Public OpActive As Boolean

Public baseHours As Double ' this is either the expected hours or the trial hours

Public actualHours As Double

```
Public RemainingHours As Double
Public ParentOp As JO_Class
Public ParentTO As TO_Class
```

```
Public CPM_EarlyStart, CPM_LateStart, CPM_EarlyFinish, CPM_LateFinish,
CPM_Float As Double
```

4. JO_Eval_Bin

The JO_Eval_Bin class is an instantiable class that performs calculations across all JO_Eval for a given time. In this class there are functions for reporting on the CPM and calculating the total about of billable hours that are expected to occur at a specific time.

Option Explicit

```
Public reportDate As Date
Public Eval_Col As Collection
Public EarlyFinish As Double
```

```
Function DOCALC_CPMReport() As Double()
```

```
    Dim workingJOEval As JO_Eval
```

```
    Dim myActualHours As Double
```

```
    myActualHours = 0
```

```
    For Each workingJOEval In Eval_Col
```

```
        myActualHours = myActualHours + workingJOEval.actualHours
```

```
    Next workingJOEval
```

```
    Dim retArr() As Double
```

```
    ReDim retArr(0 To 100, 0 To 4)
```

```
    Dim i As Integer
```

```
    Dim seg As Double
```

```
    seg = EarlyFinish / 100
```

```
    Dim t As Double
```

```
    Dim d As Date
```

```
    Dim EarlyHours, LateHours As Double
```

```
    For i = 0 To 100
```

```
        t = CDBl(i * seg)
```

```
        d = MainModule.ConvertDoubleToDate(reportDate, t) 'DateAdd("h", t * ((24 * 7) /
80), reportDate) '80 work hours per week
```

```
        EarlyHours = myActualHours + BillableHoursAtT(t, True)
```

```

    LateHours = myActualHours + BillableHoursAtT(t, False)
    retArr(i, 0) = t
    retArr(i, 1) = d
    retArr(i, 2) = EarlyHours
    retArr(i, 3) = LateHours
    retArr(i, 4) = (EarlyHours + LateHours) / 2
Next i

DOCALC_CPMReport = retArr
End Function

Function BillableHoursAtT(t As Double, EarlyOrLate As Boolean) As Double
    Dim workingJOEval As JO_Eval
    BillableHoursAtT = 0
    For Each workingJOEval In Eval_Col
        If workingJOEval.OpActive = True Then
            If EarlyOrLate = True Then
                If workingJOEval.CPM_EarlyFinish <= t Then
                    'add full hours
                    BillableHoursAtT = BillableHoursAtT + workingJOEval.RemainingHours
                ElseIf workingJOEval.CPM_EarlyStart <= t Then
                    'add prorated hours
                    BillableHoursAtT = BillableHoursAtT +
Application.WorksheetFunction.Max(0, workingJOEval.RemainingHours -
(workingJOEval.CPM_EarlyFinish - t))
                End If
            Else
                If workingJOEval.CPM_LateFinish <= t Then
                    'add full hours
                    BillableHoursAtT = BillableHoursAtT + workingJOEval.RemainingHours
                ElseIf workingJOEval.CPM_LateStart <= t Then
                    'add prorated hours
                    BillableHoursAtT = BillableHoursAtT +
Application.WorksheetFunction.Max(0, workingJOEval.RemainingHours -
(workingJOEval.CPM_LateFinish - t))
                End If
            End If
        End If
    Next workingJOEval
End Function

```

5. JOT_Class

The JOT_Class, or “Job Operation Time” Class is an instantiable class representing the actual work hours that are inputted into the tool. Each JOT_Class relates to a JO_Class.

Option Explicit

Public JobOperation As String

Public WorkDate As Date

Public BillableHours As Double

Public jo As JO_Class

6. TO_Class

The TO_Class is an instantiable class that extends a Job Operation class with specific for an individual trial. It consists of trial specific properties and methods for calculating CPM properties and expected hours to be billable prior to an input time.

Option Explicit

Public trialNum As Integer

Public ParentJO As JO_Class

Public TO_GlobalNext As TO_Class

Public TO_GlobalPrevs As Collection

Public trialHrs As Double

Public EvalTrialPoints As Collection

Public Sub DOCALC_CPM_Forwards(rDate As Date)

Dim myES As Double

myES = 0

Dim workingTO As TO_Class

If TO_GlobalPrevs.Count > 0 Then

For Each workingTO In TO_GlobalPrevs

workingTO.DOCALC_CPM_Forwards (rDate)

myES = Application.WorksheetFunction.Max(myES,
workingTO.EvalTrialPoints.Item(CStr(rDate)).CPM_EarlyFinish)

Next workingTO

```

End If

Dim myEP As JO_Eval
Set myEP = EvalTrialPoints.Item(CStr(rDate))
myEP.actualHours = ParentJO.EvalExpPoints.Item(CStr(rDate)).actualHours
myEP.OpActive = ParentJO.EvalExpPoints.Item(CStr(rDate)).OpActive

Dim testHrs As Double
testHrs = myEP.baseHours - myEP.actualHours
If Not myEP.OpActive Then
    testHrs = 0
End If
myEP.RemainingHours = Application.WorksheetFunction.Max(0, testHrs)

myEP.CPM_EarlyStart = myES

myEP.CPM_EarlyFinish = myEP.CPM_EarlyStart + myEP.RemainingHours
End Sub

Public Sub DOCALC_CPM_Backwards(rDate As Date)
    Dim workingTO As TO_Class
    If TO_GlobalPrevs.Count > 0 Then
        For Each workingTO In TO_GlobalPrevs
            workingTO.EvalTrialPoints.Item(CStr(rDate)).CPM_LateFinish =
EvalTrialPoints.Item(CStr(rDate)).CPM_LateStart
            workingTO.EvalTrialPoints.Item(CStr(rDate)).CPM_LateStart =
workingTO.EvalTrialPoints.Item(CStr(rDate)).CPM_LateFinish -
workingTO.EvalTrialPoints.Item(CStr(rDate)).RemainingHours
            workingTO.EvalTrialPoints.Item(CStr(rDate)).CPM_Float =
workingTO.EvalTrialPoints.Item(CStr(rDate)).CPM_LateStart -
workingTO.EvalTrialPoints.Item(CStr(rDate)).CPM_EarlyStart
            workingTO.DOCALC_CPM_Backwards (rDate)
        Next workingTO
    End If
End Sub

Public Property Get BillableHrsAtD(rDate As Date) As Double
    BillableHrsAtD = 0
    Dim workingTO As TO_Class
    If TO_GlobalPrevs.Count > 0 Then

```

```

    For Each workingTO In TO_GlobalPrevs
        BillableHrsAtD = BillableHrsAtD + workingTO.BillableHrsAtD(rDate)
    Next workingTO
End If

Dim myEP As JO_Eval
Set myEP = EvalTrialPoints(CStr(rDate))
BillableHrsAtD = BillableHrsAtD + myEP.actualHours
If myEP.OpActive = True Then
    BillableHrsAtD = BillableHrsAtD + myEP.RemainingHours
End If
End Property

```

7. WBS_Class

The WBS_Class is an instantiable class that primarily stores the various elements of a WBS structure from the input data. It consists of properties specific to the WBS and relational elements to other WBS elements and Job Operation elements.

Option Explicit

```

Public WBS_String, WBSParent_String As String
Public WBS_Parent As WBS_Class
Public WBS_Children As Collection
Public Op_First, Op_Last As JO_Class

Public Function PrintStructure() As String
    PrintStructure = WBS_String

    Dim wbsChild As WBS_Class
    For Each wbsChild In WBS_Children
        PrintStructure = PrintStructure & vbCrLf & vbTab &
        Replace(wbsChild.PrintStructure, vbCrLf, vbCrLf & vbTab)
        'MsgBox (PrintStructure)
    Next wbsChild
End Function

```

LIST OF REFERENCES

- American Society of Mechanical Engineers. 2020. *Types and Applications of Engineering Drawings*. ASME Y14.24. New York, NY: American Society of Mechanical Engineers. <https://doclinkonline.com/376c83f5-56ed-4696-abbc-4fdd1a8be982>.
- Barraza, Gabriel A., and W. Edward Back. 2000. "Probabilistic Monitoring of Project Performance Using SS-Curves." *Journal of Construction Engineering & Management* 126 (2): 142. [https://doi.org/10.1061/\(ASCE\)0733-9364\(2000\)126:2\(142\)](https://doi.org/10.1061/(ASCE)0733-9364(2000)126:2(142)).
- Barraza, Gabriel A., W. Edward Back, and Fernando Mata. 2004. "Probabilistic Forecasting of Project Performance Using Stochastic S Curves." *Journal of Construction Engineering & Management* 130 (1): 25–32. [https://doi.org/10.1061/\(ASCE\)0733-9364\(2004\)130:1\(25\)](https://doi.org/10.1061/(ASCE)0733-9364(2004)130:1(25)).
- Defense Acquisition University. 2020. *Earned Value Management "Gold Card."* Washington, DC: Defense Acquisition University. <https://www.dau.edu/tools/Documents/20200601%20-%20gold%20card%202%20sided.pdf>.
- Department of Defense, National Aeronautics and Space Administration. 1963. *Supplement No. 1 to DOD and NASA Guide PERT Cost Output Reports*. Washington, D.C.: PERT Coordinating Group. <https://catalog.hathitrust.org/Record/001621597>.
- Giachetti, Ronald. 2016. *Systems Engineering Thesis Research Methods*. Monterey, CA: Naval Postgraduate School. <https://nps.edu/documents/106186412/0/ThesisInSE-4.pdf/34166545-3579-454c-9653-14fc26688adb?t=1478629106000>.
- Hammad, Mohammed, Alireza Abbasi, Ripon K. Chakraborty, and Michael J. Ryan. 2020. "Predicting the Critical Path Changes Using Sensitivity Analysis: A Delay Analysis Approach." *International Journal of Managing Projects in Business* 13 (5): 1097–1119. <https://doi.org/10.1108/IJMPB-07-2019-0184>.
- Hayter, Anthony J. 2013. *Probability and Statistics for Engineers and Scientists*. 4th ed. Boston, MA: Cengage.
- Naval Center for Cost Analysis. 2014. *Joint Agency Cost Schedule Risk and Uncertainty Handbook*. Washington, DC: Naval Center for Cost Analysis.
- Office of the Secretary of Defense Cost Assessment and Program Evaluation. 2020. *DOD Cost Estimating Guide*. Washington, DC: Office of the Secretary of Defense. https://www.cape.osd.mil/files/Reports/DOD_CostEstimatingGuidev1.0_Dec2020.pdf.

Persons, Timothy M. 2020. *Cost Estimating and Assessment Guide*. GAO-20-195G. Washington, DC: Government Accountability Office. <https://www.gao.gov/assets/gao-20-195g.pdf>.

U.S. Army Research Development & Engineering Command. 2018. *Department of Defense Standard Practice Technical Data Packages*. MIL-STD-31000. Picatinny Arsenal, NJ: Department of the Army. <https://doclinkonline.com/051f5e82-6437-4316-9c60-f29fc6cfb8f3>.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California



DUDLEY KNOX LIBRARY

NAVAL POSTGRADUATE SCHOOL

WWW.NPS.EDU

WHERE SCIENCE MEETS THE ART OF WARFARE