Integrating Project Management and Organizational Change Management

Improving Project Management Performance Series

Reducing Time to Value Attainment: Using Sequential Analysis, Probability and the Weighing of Evidence

John Aaron, PhD PMP
Bruce Fischer, PhD
James Kulich, PhD

Elmhurst College Department of Business
And School for Professional Studies

The Reducing Time to Value Attainment: The Holy Grail of Project Management on Business Transformation Initiatives

For the last 25 years now the project management community has sought to find ways to deliver solutions to stakeholders more quickly. The source of this pursuit follows a very simple Return on Investment (ROI) calculation using discounted cash flows. Quite simply if the targeted benefit from a project is realized (in whole or in part) more quickly, the Benefit numerator of the ROI equation increases and the Cost denominator of the equation decreases. Many would also argue that project risk decreases. Thus taken altogether this simple truism suggests that reducing project cycle times and time to value will remain powerful motivating forces in the project management community for years to come.

Over the years the project management community has responded with various proposed approaches to shorten project life cycles. These include the following:

- Limiting the scope of project releases to meet only "must have" requirements while simultaneously maintaining rigorous and effective change control of scope.
- Gaining agreement in advance to use standard, off the shelf, prebuilt and pretested functionality for technology solutions that are installed and adopted as part of the scope of the project. Thus reducing the need for time consuming and expensive customization.
- Applying iterative development approaches to flush out requirements more accurately and to provide focus during build and testing.
- Empowering project team members to make decisions (which reduces delays) and establishing project environments that are more trusting of the team.
- Ensuring that projects are adequately staffed with dedicated team members. Also the team members remain protected from being pulled arbitrarily to support operations or other projects.
- Ensuring that executive sponsorship and appropriate project governance exists so that emergent issues can be cleared quickly.
- Making sure that the people side of the project (including user acceptance and user training) is addressed just as rigorously as the technical development side.

Throughout the years the success of the above approaches in shortening project life cycles has been quite variable. Each project situation is unique and organizations vary substantially in their ability to apply these approaches as prescribed. As a result there really is no universal magical idea that will prove effective for all projects and all organizations in reducing project cycles and reducing time to value.

That said, there is an additional approach that has yet to be considered as a means to shorten time to value. The premise is to use sequential analysis as a statistical method to identify and restrict the number of requirements-build-test iterations required in a project life cycle. From a project management perspective the characteristic feature of the method is that the number of required cycles are not determined in advance. The decision to add or eliminate cycles depends, at each stage, on the results of observations previously made. According to Wald (1947), the benefit of the sequential method is that "test procedures can be constructed which require, on average, a substantially smaller number of observations than equally reliable test procedures based upon a predetermined number of observations". Here we equate Wald's use of the term "observations " with project test cycles.

Statistical Mechanics of Sequential Analysis as Used On Business Transformation Projects

The scope of most business transformation projects includes the installation and deployment of a new technology within an organization. As such this scope generally includes the following elements in the project work breakdown structure (WBS) that must be executed throughout the project life cycle.

- Flushing out requirements and validating scope
- Validating the installation and build of the new technology
- Testing the technology through use cases and resolving identified problems
- Training the users and securing adoption of the new technology and changes to business processes
- Validating the value propositions

Generally, each of the above is conducted in one or more cycles within multiple project phases. The key feature of sequential analysis is that we do not predetermine how many cycles are necessary. Rather, we base our decision statistically on the outcomes of the observations as they are made. This is accomplished by performing statistical hypothesis testing using probability ratios with acceptance or rejection tied to the level of confidence required.

Certain obvious advantages exist when using such an approach. First, it is an evidence based approach that is tied to facts, probability, logic and the scientific method. (Note: for those new to the field of project management, Jiminy Cricket was wrong; wishing will <u>not</u> make it so!). Secondly, it is an adaptive approach and fits naturally with other adaptive approach such as the Agile methodology.

The term "Sequential Analysis" was first coined at the Statistical Research Group at Columbia University and published by Abraham Wald in 1947. Independently and previous to that Alan Turing developed and used a similar approach based upon Bayesian statistics in cryptographic work at Bletchley Park in the early 1940's. In his 1950 book titled *Probability and the Weighing of Evidence,* I.J. Good (Turing's statistical assistant) credited both Harold Jefferys and Alan Turing with independently identifying probability as "Bayes Factors" that could be used as evidence weights for testing hypotheses statistically. In his book Good also referenced the work performed by Wald and used similar examples to show that the Bayesian approach would lead to similar conclusions and reductions in required sample sizes. In this paper we will borrow heavily from Good's writings on the subject from a Bayesian perspective.

A discussion of mechanics begin with Bayes' Theorem with the probability P of a hypothesis given evidence is:

$$\frac{P(H\,|\,E)}{P(H)} \propto P(E\,|\,H)$$

Where:

- H is a variable and refers to a hypothesis to be determined either true or false.
- E is fixed and refers to evidence that either tends to support or refute the hypothesis H or its negation (H bar).
- the symbol $\propto$ means "proportionate to" .

P(E|H) is called the likelihood of H given E.

The ratio of likelihoods of H and its negation $\overline{H}$ (H-bar) given evidence E is stated as:

$$\frac{P(E\,|\,H)}{P(E\,|\,\overline{H})}$$

Good uses the simplest case when there are only two hypotheses (H and H bar), and the following relationship holds:

$$\frac{O(H|E)}{O(H)} = \frac{P(E|H)}{P(E|\bar{H})}$$

Where O(H|E) is defined as P(H|E)/(1-P(H|E) which is referred to as the odds of H given E. And O(H) is referred to as the initial odds; whereas O(H|E) is referred to as the final odds. If p is any probability, the corresponding odds are defined as o=p/1-p, so that p=o/1+o.

The factor (i.e. the Bayes' factor) in favor of hypothesis H given the evidence E is derived from the likelihood ratio. Good wrote that both Turing and Jefferys found it convenient to take the logarithm of the likelihood ratio which allowed probabilities to become additive. In doing so Turing adopted the bels and decibels (db) notation from acoustics and electrical engineering where the bel is the logarithm to base 10 of the ratio of two intensities of sound.

For our project management purposes evidence can either be favorable (supportive) or unfavorable (non supportive) to a hypothesis. For instance, the question of whether or not testing is finished on a project would be a rather typical hypothesis. And evidence based on testing performance would either support or fail to support that hypothesis.

Further, in the application of these tools we start with some baseline probability about H which may be based upon subjective or objective calculation. Then given some evidence, if *f* is the resulting factor in favor of a hypothesis, we say that the hypothesis has gained $\log_{10} f$ bels or ($10 \log_{10} f$) db. Turing referred to this metric as the *weight of evidence* or the amount of information for H given E or the *Plausibility Gained* for the hypothesis.

The weight of evidence calculations are particularly useful for multiple observations (multiple pieces of evidence) because they are additive. As I.J. Good describes:

> Suppose that a series of experiments are performed with the results E_1, E_2........E_n and that these are independent given H and H bar. Then the resulting factor is equal to the product of the individual factors, and therefore the resulting weight of evidence is equal to the sum of the individual weights of evidence. For

$$\frac{P(E_1, E_2, \ldots\ldots E_n | H)}{P(E_1, E_2, \ldots\ldots E_n | \bar{H})} = \frac{P(E_1 | H)}{P(E_2 | \bar{H})} \cdots\cdots\cdots \frac{P(E_n | H)}{P(E_n | \bar{H})}$$

To explain this approach Good provides the following example which closely parallels that provided by Wald:

> A die is selected from a hat containing 10 homogenous dice and one loaded one. The loaded one is assumed of having a chance of 1/3 of yielding a 6. The selected die is thrown nine times and comes down 6 eight times. What is the final probability that the selected die is the loaded one?

In this example we intuitively know that the initial probability of selecting the loaded die is rather low– 1 out of 11 or .0909. We also know that new information becomes available every time we roll the selected die and get a result. Every time we rolled a 6, it tended to support the hypothesis that we selected the loaded die. Every time it was not a 6 on the roll, it tended to refute the hypothesis that we selected the loaded die.

Figure 1 shows the results as a table and Figure 2 as a graph. We assume the die came down with a 6 on rolls 1,3,4,5,6,7,8,9. Whereas it came down with a 2 on roll number 2. The graph illustrates how each roll of a 6 resulted in the accumulation of new, additional evidence that supported or refuted the hypothesis that we in fact had selected the loaded die. Indeed we went from less than a 10% probability initially to 95% certainty after the ninth roll. The graph shows the accumulated steady support for the hypothesis except for roll #2 where we saw a dip.

From the table in Figure 1 we see that the initial plausibility for the selected die's being loaded is $10 \log_{10}$= -10db.

		Initial Plausibility of Choosing Loaded Die	<-------------------------------Plausibility Gained or Lost From Evidence----------------------------------->								
	Action	**Choose Die from Hat**	**Roll 1**	**Roll 2**	**Roll 3**	**Roll 4**	**Roll 5**	**Roll 6**	**Roll 7**	**Roll 8**	**Roll 9**
	Result	**1 Die is Chosen Out of 11**	**For**	**Against**	**For**	**For**	**For**	**For**	**For**	**For**	**For**
Gained if Roll a 6	$P(E\|H)/P(E\|\bar{H})$	0.0909	2.0000		2.0000	2.0000	2.0000	2.0000	2.0000	2.0000	2.0000
Lost if Not Roll a 6	$1-P(E\|H)/1-P(E\|\bar{H})$	0.9091		0.8000							
	db	-10.0000	3.0103	-0.9691	3.0103	3.0103	3.0103	3.0103	3.0103	3.0103	3.0103
		Starting Point	1	2	3	4	5	6	7	8	9
	Cumulative db	-10.0000	-6.9897	-7.9588	-4.9485	-1.9382	1.0721	4.0824	7.0927	10.1030	13.1133
	Cumulative Odds	0.1000	0.2000	0.1600	0.3200	0.6400	1.2800	2.5600	5.1200	10.2400	20.4800
	Cumulative Probability	0.0909	0.1667	0.1379	0.2424	0.3902	0.5614	0.7191	0.8366	0.9110	0.9534

Figure 1-Weight of Evidence Dice Rolling Example

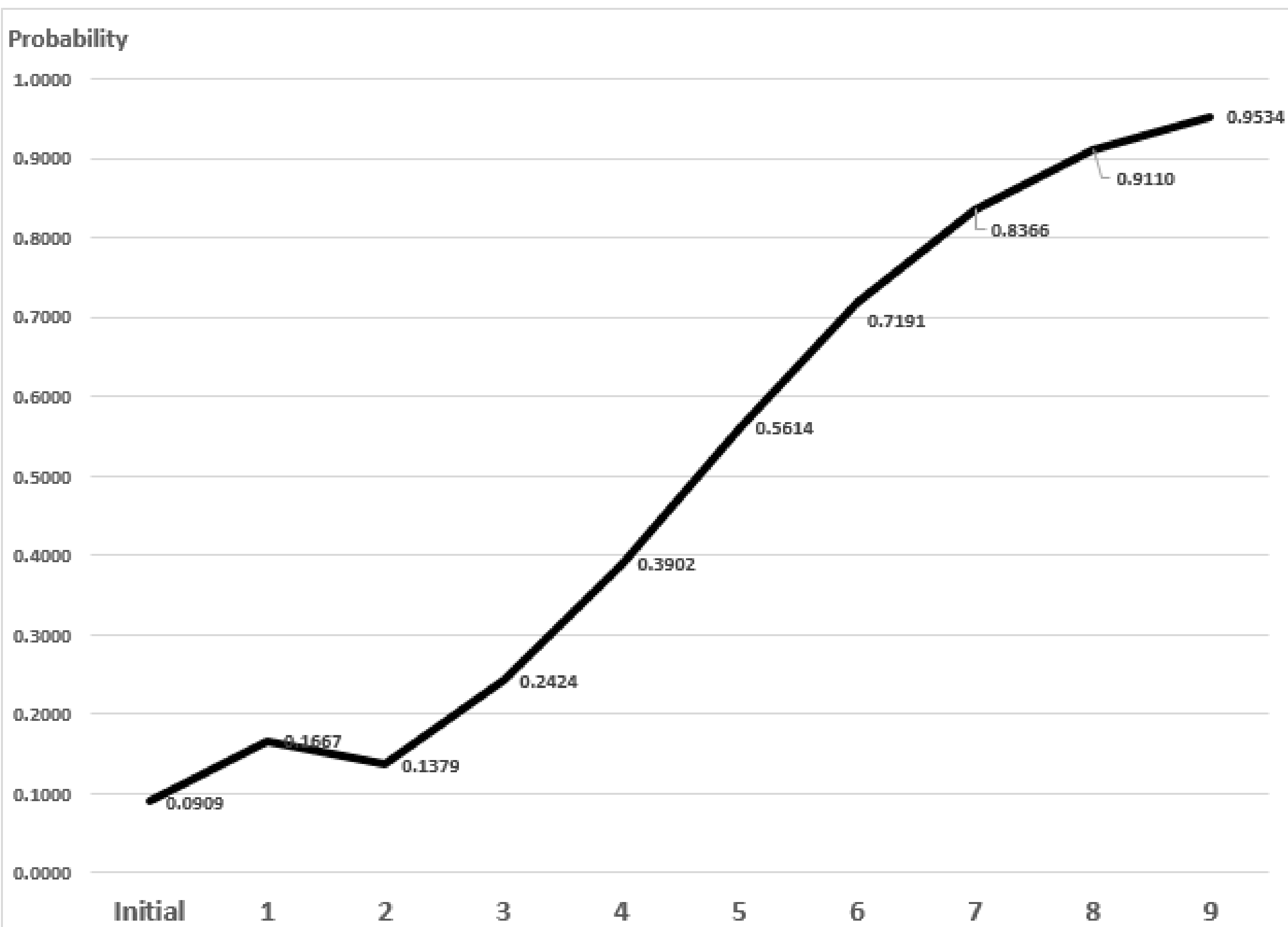


Figure 2-Cumulative Probability Dice Rolling Example

For each six rolled the hypothesis that the selected die is the loaded one gains a factor of 1/3÷1/6 or 3.01 db. For each non six rolled it loses a of 5/6÷2/3 or nearly 1 db (0.9691 db). In this example the cumulative net gain from the nine rolls is 23 db, the final plausibility is 13 db and the final odds are 20 to 1 in favor of the selected die being the loaded one. The 20 to 1 odds convert to a probability of 0.95 given that probability=odds/(1+odds). The graph shown in Figure 2 illustrates how the sequence of evidence has cumulatively built up to support the conclusion that the hypothesis is true. Certainly, if more confidence was required, more rolls could be performed. In this case we may feel that 95% confidence is adequate; so we would cease conducting further experiments.

In this paper we argue that this same weight of evidence approach is a useful decision making tool for determining the minimal number of requirements-design-build-test cycles on business transformation projects. In doing this the approach gives us a framework for minimizing project cycle time without sacrificing the quality of the project. We will walk through two examples using build-test cycles and value auditing as our examples.

Project Management Example #1- Using Sequential Analysis and Weight of Evidence for Determining the Number of Build-Test Cycles.

It is quite common for developers to use requirements-build-test cycles iteratively to progressively elaborate until the required project deliverable is ready to be handed over to the customer. This adaptive approach has been used to some degree over the last 30 years, and it is especially prominent today with Agile based methodologies where we call these cycles "sprints".

The concept of iterative progressive elaboration is perhaps best illustrated by the diagram of the "spiral model" initially developed by Barry Boehm in the mid 1980's. Every methodology has its own nuances, but Figure 3 will serve as useful model for general discussion. Figure 3 illustrates how the developed product evolves in cycles from concept to various prototypes to a released product. The question immediately becomes: how many cycles are really needed? Is two the right number of prototypes needed? These are important questions that developers and project managers face commonly. In order to save effort and cost we want to conduct the fewest cycles possible, but we must conduct enough cycles to ensure adequate system quality and user acceptance. All too often these decisions are made arbitrarily without systematic consideration of the available evidence as it accumulates. The disciplined plan-control paradigm of project management is necessary, but this approach suggests that we do need to remain adequately flexible to adjust the plan based upon the evidence gathered throughout the project life cycle. We will apply the weight of evidence approach using sequential analysis to maintain that flexibility.

As shown in Figure 3 during the first cycle we identify the initial requirements and scope. This first cycle could actually be done during the Discovery phase prior to the ramp up of the project team. We assume that this exercise yields the identification of 25 best practice solutions (i.e. processes) that need to be delivered as the project deliverable for the first release. We further assume that the team believes 100 business scenarios (use cases) will be identified that the new system will need to support and that users will need to be ready to execute properly before going live. Using only modal salient scenarios identified thus far (see Figure 4) the team estimates that probability of the system being ready (out of the box) using standard functionality is 25%. Hence P(H)=0.25 and P (H bar)=0.75. The team also works to gather and close all identified open issues related to the identification of project scope.

Next the project manager and consultants collaborate to calculate odds and probabilities for the second cycle (Prototype 1) by considering past experiences on prior projects as well as the unique aspects of the project at hand. The team determines that the first prototype will be capable of demonstrating all 25 best practice solutions and they identify 68 out of the total estimated 100 (1σ) of the required use cases. After building and demonstrating the first prototype and closing open issues, the team estimates P(E|H) = 0.95 and the P(E|H bar)=0.50. Here P(E|H) refers to the probability of "passing the review" given that the Hypothesis (i.e. system completeness) is true. And P(E|H bar)

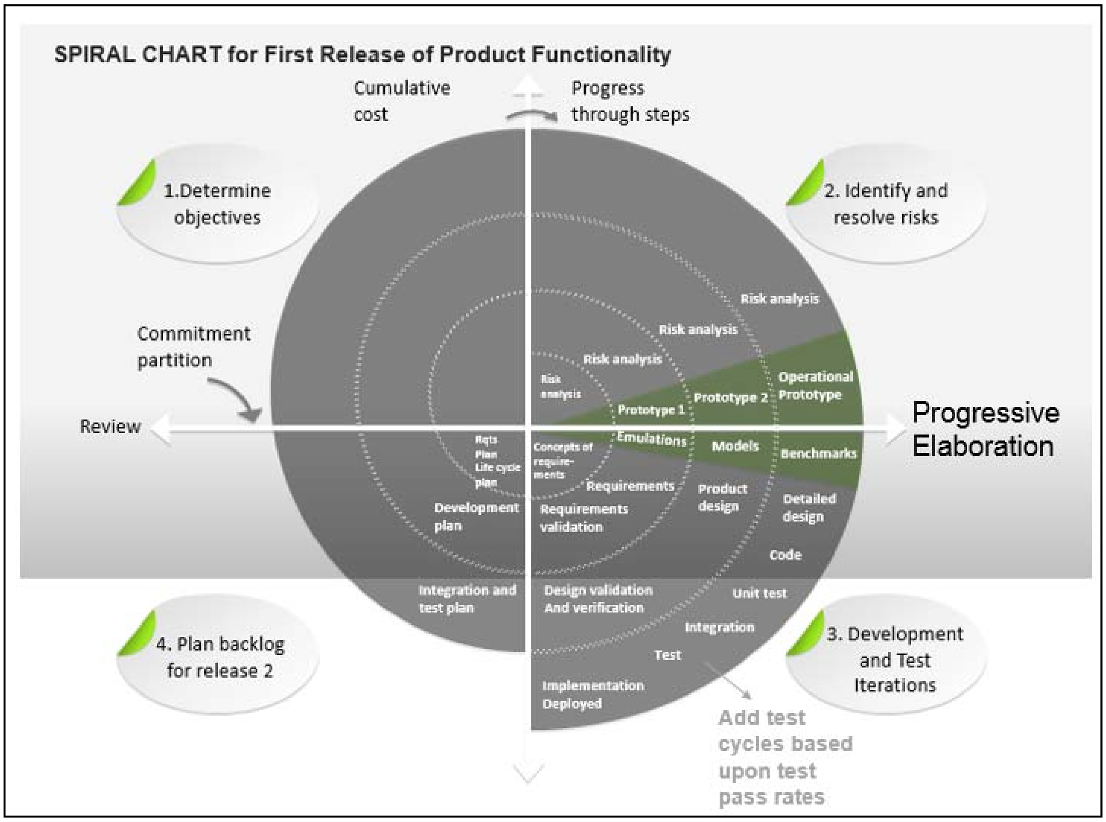


Figure 3-The Spiral model as an Example of an Iterative Development Life Cycle

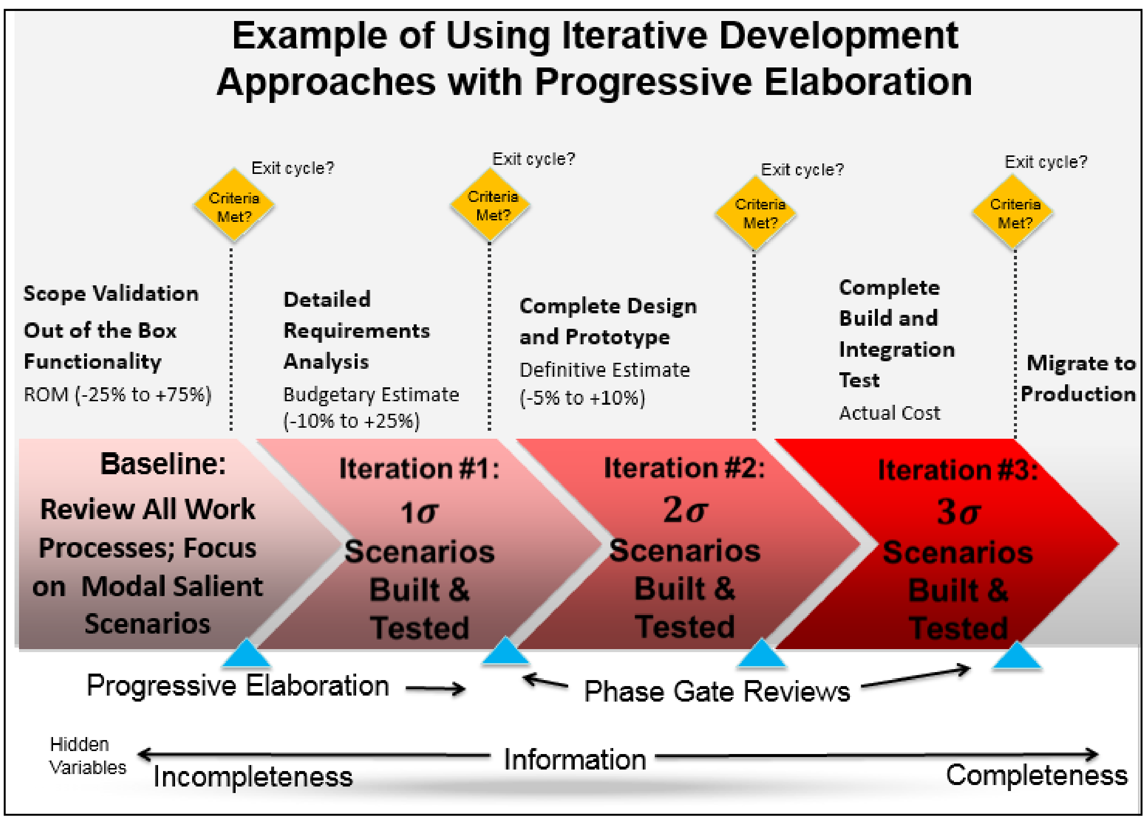


Figure 4: An Example of Progressive Elaboration

refers to the probability of "passing the review" given that the negation H bar (i.e. system incompletion) is true.

A few key points need to be highlighted in regard to the probability estimates. The value assigned to P(E|H bar)= 0.50 drives home a subtle but crucial point. It is saying that even though we have built and successfully demonstrated all 25 best practice solutions using 68 out of estimated 100 use cases, the team believes that there is still a 50% chance that the system is not yet completely ready. Undoubtedly this 0.50 probability is due to the team's belief that that they still have not flushed out all of the required use cases. Hence, they are dealing with uncertainty due to incompleteness of understanding which is always a risk with progressive elaboration. It is also critical to point out that all issues identified in the cycle must be closed. This allows the team to accept the entire "lot" and move on to the next cycle.

As the project advances through the next cycle the project manager and consultants collaborate to calculate odds and probabilities for the third cycle (Prototype 2) following the same approach. The team determines that the first prototype is capable of demonstrating all 25 best practice solutions and they can now identify 95 out of the total estimated 100 (2σ) of the expected use cases. After building and demonstrating the second prototype and closing open issues, the team estimates P(E|H) = 0.97 and the P(E|H bar)=0.15. Here again P(E|H) refers to the probability of "passing the review" given that the Hypothesis (i.e. system completeness) is actually true. And P(E|H bar) refers to the probability of "passing the review" given that the negation of the hypothesis (i.e. system incompletion) is true.

For the third cycle the project manager and consultants collaborate to calculate odds and probabilities (System Integration test) following the same approach. The team determines that the build is capable of demonstrating all 25 best practice solutions and they identify 99 out of the total estimated 100 (3σ) of the expected use cases. After building and demonstrating the second prototype and closing open issues, the team estimates P(E|H) = 0.98 and the P(E|H bar) =0.05.

Figures 5 and 6 provide tabular and graphical overviews of the weight of evidence calculations from the above example. Here we see that the final plausibility is 19 db, and the net gain in plausibility from the three cycles is nearly 24 db. The final odds are 80 to 1 in favor of completeness with a probability of 98%. Our acceptance of the hypothesis of system completeness depends upon the level of confidence required. If 95% confidence was used as our acceptance criteria, we would be set to go. If greater than 99% confidence was absolutely required, we might consider adding another integration test cycle. The benefits versus costs of that decision would have to be considered carefully.

The Weight of Evidence approach combined with Sequential Analysis concept provides several useful insights regarding how project life cycles might be shortened without sacrificing quality. For instance, the starting point P(H) has a significant impact on how many cycles of evidence must be gathered to obtain a final plausibility. By beginning the project with a high P(H), fewer cycles are required. This may explain why companies such as SAP offer pre-configured, rapid deployment solutions that can be demonstrated during the earliest phases of the project—including the Discovery phase. This also explains why implementers try to avoid customization and use standard, out-of-the-box functionality wherever possible. Understandably, the more complete the system is, the greater the weight of evidence supporting readiness from each cycle because P(E|H bar) becomes substantially lower.

The approach also points out that limits may exist to cycle time reduction. As we have seen a certain number of review cycles will always be necessary in order for the project team and stakeholders to gain an adequate level of confidence in the system before going live. It should be noted that measurement gains in db are very non linear and subject to diminishing returns. Adding a cycle of testing may increase costs by 30% yet add only a small gain in db.

Finally, it is essential to re-emphasize the importance of having a formal issue management and defect management process to ensure that open items are resolved before moving on to additional cycles. The failure to remove faults leads to a rejection of the "lot" which causes the weight of evidence for the hypothesis to decline. Consequently the disposition of issues, risks, defects, action items, etc. must be tracked and managed very diligently.

			Initial Plausibility of Complete Build	<---------------Plausibility Gained From Evidence------------>		
	Hypotheses		Start	Prototype 1	Prototype 2	Integration Test
H	Complete Build	$P(E \mid H) / P(E \mid \bar{H})$	0.2500	1.9000	6.4667	19.6000
$\bar{H}$	Not Complete Build		0.7500			
		db	-4.7712	2.7875	8.1068	12.9226
			Starting Point	1	2	3
		Cumulative db	-4.7712	-1.9837	6.1231	19.0457
		Cumulative Odds	0.3333	0.6333	4.0956	80.2729
		Cumulative Probability	0.2500	0.3878	0.8038	0.9877

Figure 5– Weight of Evidence Build-Test Example

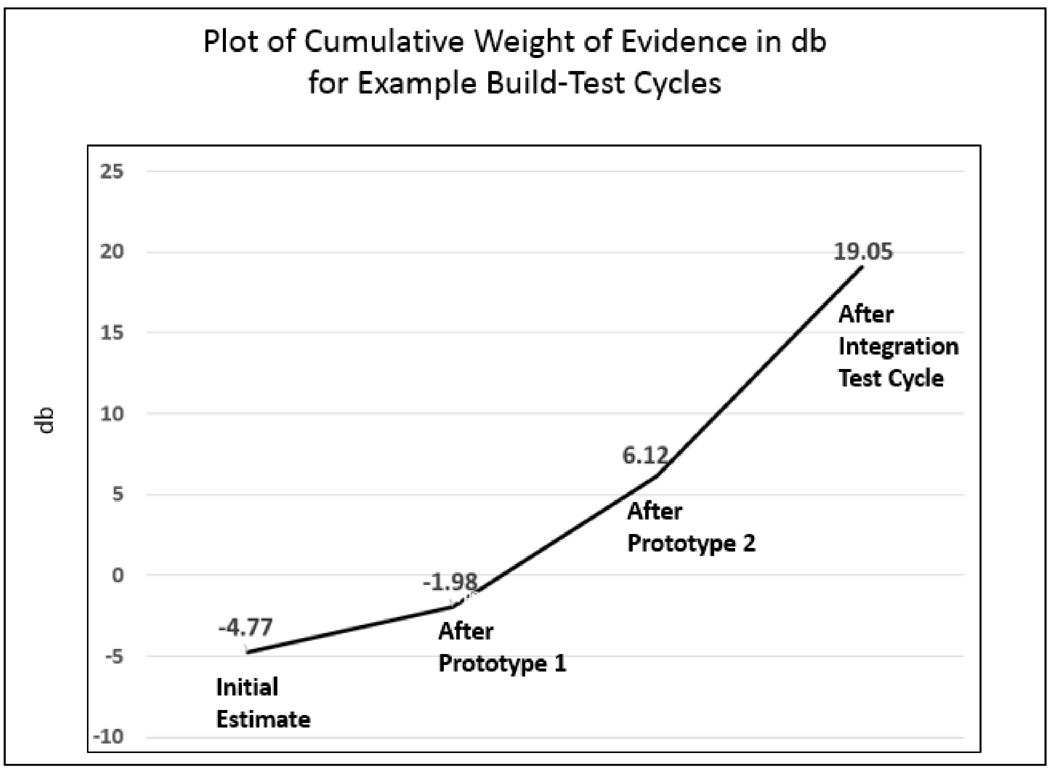


Figure 6-Plot of Cumulative Weight of Evidence Build-Test Example

Project Management Example #2- Using Sequential Analysis and Weight of Evidence for Validating Value Attainment

Increasingly project managers are responsible for verifying that the targeted project benefits will be realized. Historically within the PMBOK framework value realization has been considered a scope verification exercise, but more often than not today's practitioners must distinguish between scope verification and value verification. Scope refers to work items on the project Work Breakdown Structure (WBS). Value refers to the net targeted benefits that should result from the successful execution of the WBS. One might think that a direct mapping between the two would always exist, but this is not the case. Obtaining value does not always result from implementing a new system without consideration of the behavioral and process changes that must accompany the new system. Hence business transformation typically requires changes in technology, people and process simultaneously, and the purview of the project manager must straddle all three domains.

In practice many project managers begin to verify value only after the project deliverables have been handed over to the customer. But by then it is too late to recover if certain value propositions have been skipped or missed. Therefore, we will show a simple example of how sequential analysis and the weighing of evidence can be used to help the project manager avoid this risk by validating and managing value attainment throughout the project life cycle. This approach increases the likelihood that all targeted value propositions will be realized once the system is live.

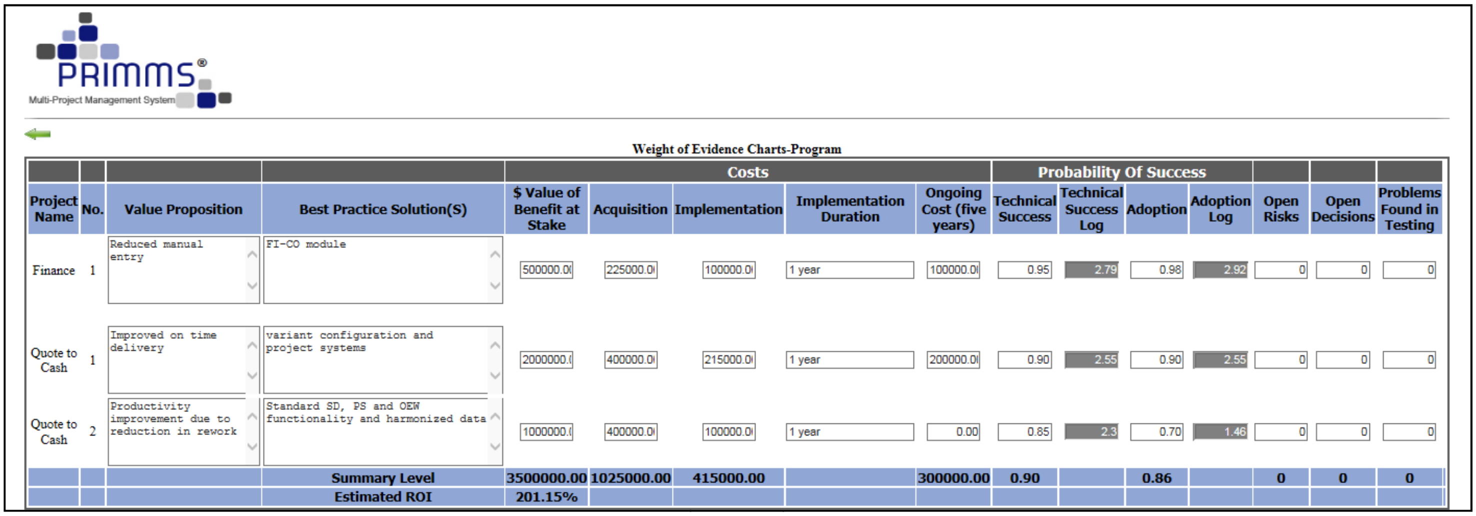
Business transformation projects are investments that are justified based upon a business case. Generally, the benefits expected from the project are articulated in a set of value propositions to be attained. Figure 7 illustrates three sample value propositions totaling $3.5 mil in benefit. In reality hundreds of these kinds of propositions may exist on a project, but we will keep things simple for our example using only three.

Each value proposition is assigned to a sub-team on the project and is mapped to a best practice solution that goes to the WBS of the project for implementation. As part of the value audits during each phase, the project manager measures two factors for each proposition:

1. The probability of technical success
2. The probability of user adoption

The total probabilities across all value propositions can be averaged (in this case 0.90 and 0.86). In addition a composite probability can be calculated by multiplying the average probability of technical success and the average probability of user adoption (in this case producing an average of 0.77 =0.90*0.86). These probabilities can be checked each phase by performing value audits allowing for preventative and corrective actions as needed. On larger scale projects this accountability would go to the functional team leads representing the various streams of work.

The project manager performs sequential analysis and weight of evidence calculations in nearly the same manner as discussed previously. The only difference is that instead of using discrete probabilities, the PM is now using continuous probability distributions for hypothesis testing. Figure 8 shows that P(E|H) assumes a normal distribution with a mean of 0.8 and standard deviation of 0.2, and P(E|H bar) assumes a normal distribution with a mean of 0.2 and standard deviation of 0.2. Graphs of both distributions are truncated by shading to imply that we constrain probabilities to exist only within the range of 0 to 1.



Multi-Project Management System

Weight of Evidence Charts-Program

				Costs					Probability Of Success						
Project Name	No.	Value Proposition	Best Practice Solution(S)	$ Value of Benefit at Stake	Acquisition	Implementation	Implementation Duration	Ongoing Cost (five years)	Technical Success	Technical Success Log	Adoption	Adoption Log	Open Risks	Open Decisions	Problems Found in Testing
Finance	1	Reduced manual entry	FI-CO module	500000.0	225000.0	100000.0	1 year	100000.0	0.95	2.79	0.98	2.92	0	0	0
Quote to Cash	1	Improved on time delivery	variant configuration and project systems	2000000.	400000.0	215000.0	1 year	200000.0	0.90	2.55	0.90	2.55	0	0	0
Quote to Cash	2	Productivity improvement due to reduction in rework	Standard SD, PS and OEW functionality and harmonized data	1000000.	400000.0	100000.0	1 year	0.00	0.85	2.3	0.70	1.46	0	0	0
			Summary Level	3500000.00	1025000.00	415000.00		300000.00	0.90		0.86		0	0	0
			Estimated ROI	201.15%											

Figure 7-Example of Value Propositions

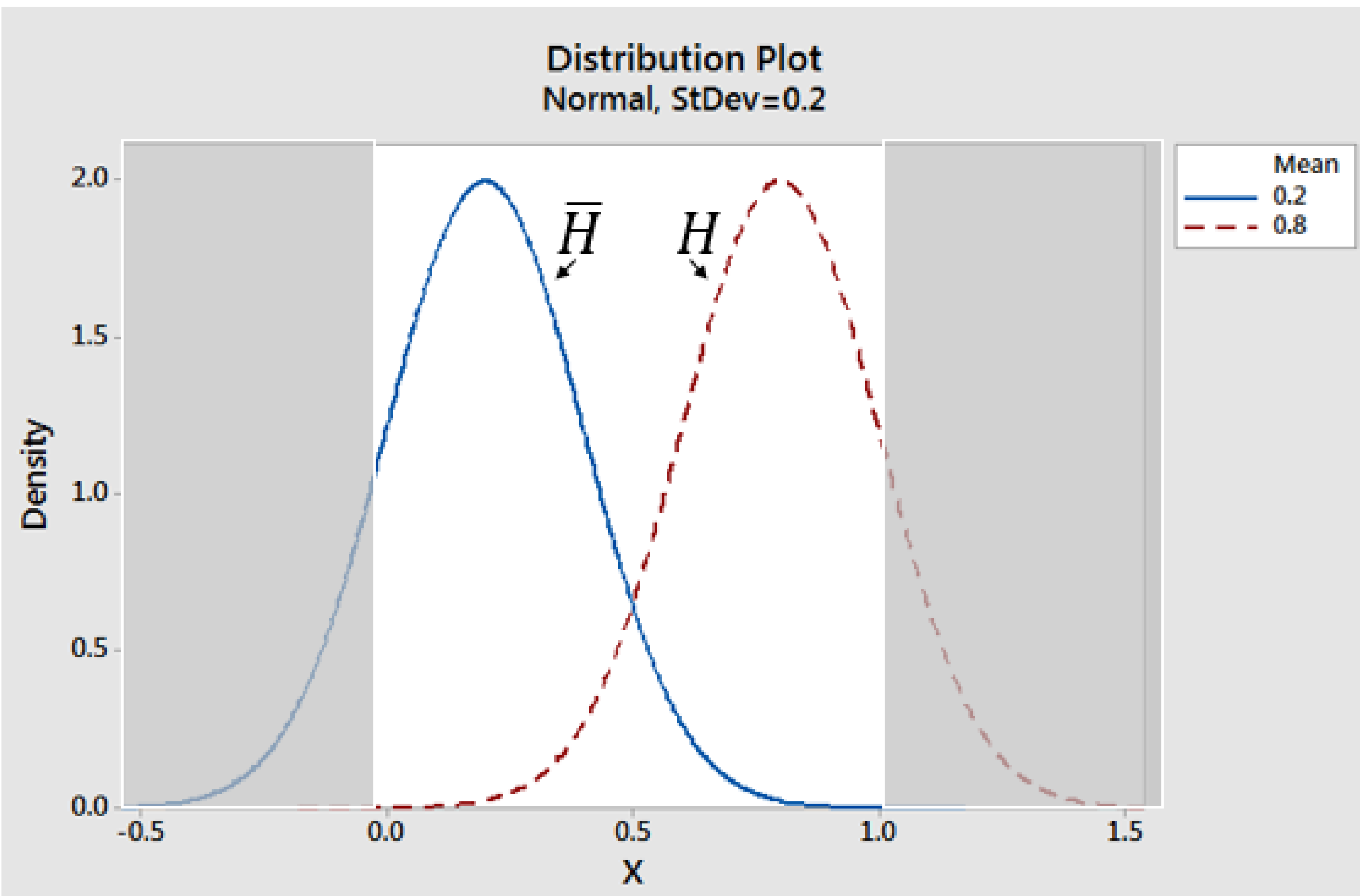


Figure 8-Using Two Truncated Normal Distributions to Compute the Likelihood Ratio P(E|H) ÷ P(E|H bar)

Continuing with our example, the project team can periodically review the project's performance against each value proposition by obtaining evidence that is relevant to the phase of the project. The evidence becomes increasingly specific as the project progresses. Given the evidence the team members can assess the probability of technical success and the probability of user adoption for each value proposition. The table below provides a suggested example of the kinds of evidence that can be obtained.

Project Phase	Value Audit Evidence
Discovery	-Mapping of Best Practice Solution to each value prop. -Quantification of each value prop.
Prepare	-Check to ensure that each value prop and best practice solution is mapped to a WBS element on project plan.
Explore	-Review process maps of best practice solutions and verify that value props will be supported. All open issues are recorded and assigned to process owners.
Realization	-During string testing and integration testing, check to ensure that a) process procedures support value props, b) the organizational redesign supports the value props, c) first line supervisors endorse the changes and change management issues are closed .
Deployment	-Check to ensure that user training supports the necessary process changes for each value prop.
Run	-Progress measurements for each value prop are put into place and performance is on target.

The audits produce evidence as probabilities that can be hypotheses tested in aggregate or individually against the H and H bar distributions. Figure 9 illustrates how this would work given our sample evidence where P=0.86. At that probability level the likelihood (height) of the curve for H is substantially higher than the likelihood for the curve on H bar. If the evidence probability was P=0.50, the resultant likelihoods of H and H bar would be equal; and if the evidence probability was below 0.50, the likelihood of H bar would be greater than that for H. Since our weight of evidence calculation is based upon the likelihood ratio of H and H bar, it is clear that in this case an evidence probability for P=0.86 will add to the plausibility of our hypothesis H.

The table in Figure 10 extends this example to illustrate how the weight of evidence might look throughout the entire project life cycle. Also, the graph in Figure 11 on page 13 illustrates how value evidence would accumulate and be displayed for a project throughout its life cycle. The figure also displays graphically how the project manager can communicate the value attainment status as evidence reaches the "acceptance" region. Ideally this would occur by the time of go live. Figure 12 is a screenshot illustrating the use of the approach operates within the web Project Management tool-PRIMMS.

Summary

It is often said that project managers are "keepers of the truth" and in this capacity, they must stay in touch with the facts related to project performance. The tools of sequential analysis, probability and the weighing of evidence are key tools that aid the project manager in this pursuit. In this paper we have discussed the statistical mechanics of these concepts and have shown two examples of their application in the field of project management-one for testing cycle

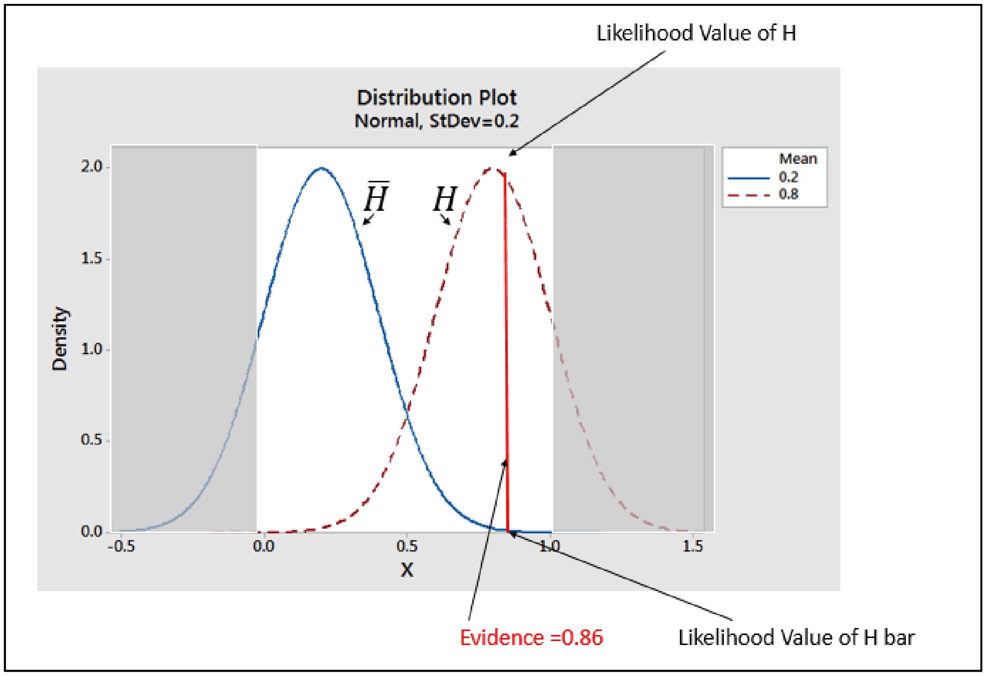
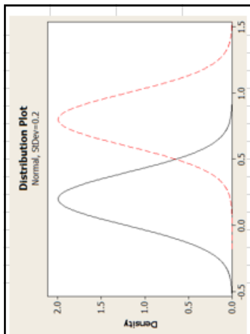


Figure 9



	Discovery	Prepare	Explore	Realize	Deploy	Run
	Initial Plausibility	Likelihood	Likelihood	Likelihood	Likelihood	Likelihood
Acceptance	0.25	1	0.95599748	0.969233234	0.882496903	0.754839602
Evidence		0.8	0.86	0.85	0.9	0.95
Mu		0.8	0.8	0.8	0.8	0.8
sigma		0.2	0.2	0.2	0.2	0.2
z		0	0.3	0.25	0.5	0.75
Rejection	0.75	0.011109	0.00431784	0.005086069	0.002187491	0.000883826
Evidence		0.8	0.86	0.85	0.9	0.95
Mu		0.2	0.2	0.2	0.2	0.2
sigma		0.2	0.2	0.2	0.2	0.2
z		3	3.3	3.25	3.5	3.75
Odds: Acceptance to Rejection	0.333333333	90.0171313	221.406416	190.5662685	403.4287935	854.0587625
10Log Odds	-4.771212547	19.5432517	23.451902	22.8004603	26.05766891	29.31487753

	Plausibility db
Discovery	-4.77
Prepare	19.54
Explore	23.45
Realize	22.80
Deploy	26.06
Run	29.31

	Cumulative Plausibility (db)
Discovery	-4.77
Prepare	14.77
Explore	38.22
Realize	61.02
Deploy	87.08
Run	116.40

Figure 10

management and one for value management. There are many ways that project managers can use Bayesian methods to reduce time to value, and this is just one of them. The interested reader should consult the references provided below.

References:

Good, I.J. (1950). *Probability and the Weighing of Evidence.* New York. Hafner Publishing.

Good, I.J. (1965). *A List of Properties of Bayes-Turing Factors* - Declassified NSA Tech Journal Spring 1965 - Vol. X, No. 2

Jaynes, E.T. (1956). Probability Theory in Science and Engineering, No. 4 in `Colloquium Lectures in Pure and Applied Science,' .

Jaynes, E.T. (2003). *Probability Theory: The Logic of Science.* Cambridge, UK: Cambridge University Press.

Wald, A. (1947). *Sequential Analysis.* New York. Wiley.

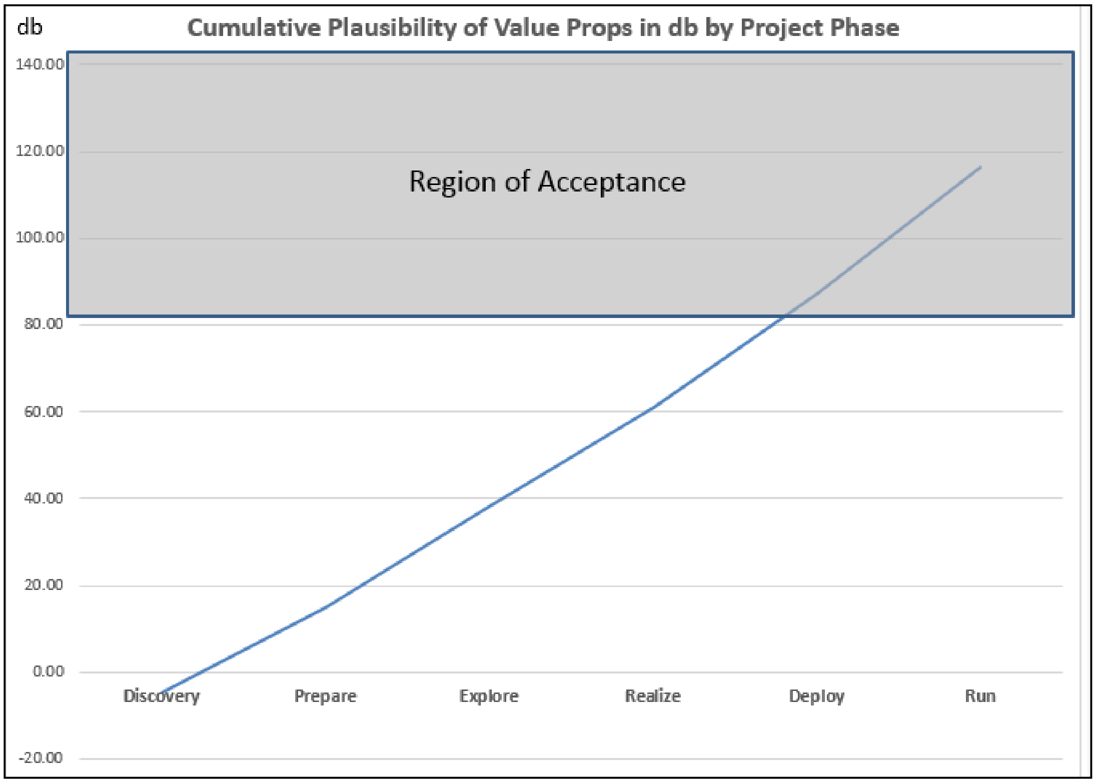
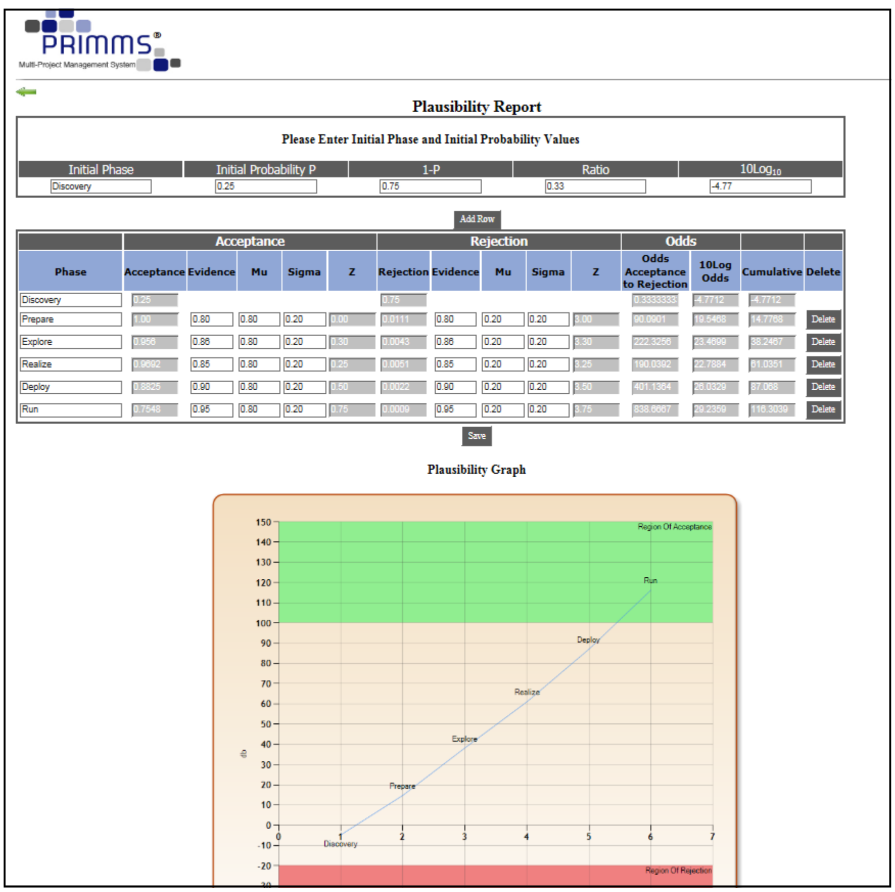


Figure 11



Initial Phase	Initial Probability P	1-P	Ratio	$10Log_{10}$
Discovery	0.25	0.75	0.33	-4.77

	Acceptance					Rejection					Odds			
Phase	Acceptance	Evidence	Mu	Sigma	Z	Rejection	Evidence	Mu	Sigma	Z	Odds Acceptance to Rejection	10Log Odds	Cumulative	Delete
Discovery	0.25					0.75					0.3333333	-4.7712	-4.7712	
Prepare	1.00	0.80	0.80	0.20	0.00	0.0111	0.80	0.20	0.20	3.00	90.0901	19.5468	14.7768	Delete
Explore	0.956	0.86	0.80	0.20	0.30	0.0043	0.86	0.20	0.20	3.30	222.3256	23.4699	38.2467	Delete
Realize	0.9692	0.85	0.80	0.20	0.25	0.0051	0.85	0.20	0.20	3.25	190.0392	22.7884	61.0351	Delete
Deploy	0.8825	0.90	0.80	0.20	0.50	0.0022	0.90	0.20	0.20	3.50	401.1364	26.0329	87.068	Delete
Run	0.7548	0.95	0.80	0.20	0.75	0.0009	0.95	0.20	0.20	3.75	838.6667	29.2359	116.3039	Delete

Figure 12