Making sense of Schedule Risk Analysis

John Owen - Barbecana Inc.
John Owen - jowen@barbecana.com

• 5 Years managing project controls software in the Oil and Gas industry
• 28 years developing and supporting project management software
• 8 years as product manager at Deltek
  • Responsible for Schedule and Risk Tools
• Moved to Texas in 2003
• Married a Texan in 2006
• Became a US Citizen in January 2013
• Joined Barbecana in February 2014.
• Became a Cowboys fan October 2013.
Let’s assume that we plan a project and are very good at estimating how long work will take.

We then execute the project and, on average, the actual time taken to perform work is right on the estimates, or maybe even a little better.

How likely is the project to finish on time or perhaps even a little early?
Projects and Models

Project

A project is a defined scope of work to achieve a desired objective. A project results in ‘change’. Where there was no bridge there is now a bridge...

Critical Path Method (CPM)

Critical Path Method is a technique for constructing a model of a project. This will include a complete list of the tasks/activities/things to be done to deliver/completed the project, the time required for each task, and logical dependencies to show the order tasks should be performed.

The model is then used to calculate the start and end dates of the individual tasks and predict the expected project completion date.
The trouble with Critical Path Method

Critical Path Method calculates a single deterministic expected finish date for your project.

Every task(activity), no matter how similar to tasks before it, is always subject to some uncertainty.

This means that the only sure fact, for the expected project end date calculated by CPM, is that it will almost certainly be wrong.

Worse, the end date calculated by CPM is usually overly optimistic.

Let’s find out why.
A simple Critical Path Method (CPM) model

Assuming Task A starts on Day 1 of our project, Task A will finish on Day 5. Because of the Finish-to-Start relationship between Task A and B, the earliest Task B can start is Day 6. Adding the 5 days duration for Task B means that the project will complete on Day 10.

A delay to Task A will cause a delay to the start of Task B but we have an opportunity to make up the lost time by completing Task B in less than the estimated time, so that the project can still be completed on time.
In this example, Tasks A and B can both start when the project starts. However, the project will only be complete when both Tasks A and B have been completed. A delay for Task A will cause a delay for the project, even if Task B is not delayed or finishes early (and vice versa).
Let’s tabulate the possible outcomes for Task A and B (and to simplify the table we’ll count an on-time finish as early).

<table>
<thead>
<tr>
<th>Task A</th>
<th>Task B</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>Early</td>
<td>Early</td>
<td>Early</td>
</tr>
<tr>
<td>Late</td>
<td>Early</td>
<td>Late</td>
</tr>
<tr>
<td>Early</td>
<td>Late</td>
<td>Late</td>
</tr>
<tr>
<td>Late</td>
<td>Late</td>
<td>Late</td>
</tr>
</tbody>
</table>

We know that the project will only complete on time if both Task A and B finish early or on time. Of our four possible outcomes we can see this only happens one time. Any of the other three possible outcomes results in a late project completion.
One reason projects fail

As the number of predecessors for any given activity increases, it becomes less likely that it will start on time. This effect is called Merge Bias.

Merge Bias is the single biggest reason that project models, built using Critical Path Method (CPM), produce an unrealistic estimate for project completion.

As the complexity of the project model increases, and the number of activities with multiple predecessors grows, the probability of attaining the deliverable dates calculated by CPM decreases.

So project failure may not be caused by poor execution, but simply by the fact that the plan was never realistic or achievable in the first place.
Should we add a contingency to estimates?

Task duration estimates can be made using a variety of techniques (expert judgment, parametric estimating etc.). The duration represents the time you expect the task to take. A good estimate equates to a 50/50 chance of finishing the work in the time allotted.

So why not add a contingency to each task duration. Wouldn’t that improve our chance of finishing each task on time?

The problem is Parkinson’s law. “Work expands so as to fill the time available for its completion”.

Never plan work using padded durations that include contingency.
Improving the Model

So if a standard Critical Path Method (CPM) project model results in an unrealistically optimistic ‘deterministic’ deliverable date and adding contingency to task durations is a bad idea, what can we do to improve matters?

The biggest flaw with CPM is that a single estimated duration is captured for each task. This is very unrealistic. Even for tasks that have been performed before there will be some uncertainty, even if it is completely external (for example high absenteeism when the USA played in the 2014 World Cup).

So a better model would be based on a range of estimates for each task.
The Performance Evaluation and Review Technique (PERT) captures an Optimistic, Most Likely, and Pessimistic duration for each task.

This is a big improvement in our estimating technique. We can then use the three points to calculate an expected duration for each task.

Expected Duration = (Optimistic + (4 x Most Likely) + Pessimistic) / 6

The expected duration is then used with a regular CPM algorithm to calculate an expected project completion date.

PERT also calculates a Standard Deviation (Error) around the expected finish date so practitioners can select a possible completion date based on a desired level of confidence (probability).
The trouble with PERT

Because the range of values for each task are distilled into a single expected duration for each task, followed by a regular CPM style calculation, PERT does not model the effect of Merge Bias.

PERT also calculates a single deterministic Critical Path (just like CPM) whereas, referring back to our simple parallel Task A and B example earlier, we can see that in reality the Critical Path may vary and that would be good to know from a management perspective.

So how can we improve on PERT?
Monte Carlo Simulation

Deterministic solutions like CPM and PERT cannot model the interaction of uncertainty on the various tasks in the project.

But what if we could simulate the execution of the project thousands of times to see how the uncertainty inherent to each task interacts?

This is what Monte Carlo Simulation (aka Schedule Risk Analysis) allows us to do.

The technique is named after Monte Carlo, a European city that is famous for games of chance.
Simulation?

Can’t we just ‘calculate’ the correct answer?

Unfortunately no. Modelling the interaction of many ‘random’ variables can’t be handled by an equation.

Even apparently straightforward tasks, like predicting the future position of the Moon relative to the Sun and Earth, have to be handled by simulation (Three Body Problem).

A Monte Carlo Risk Analysis will consist of many individual simulations (sometimes called trials or iterations) and the results of each individual simulation will be tallied for reporting.
Estimate Uncertainty (Epistemic)

Like PERT, Monte Carlo simulation typically captures an Optimistic, Most Likely and Pessimistic duration estimate for each task - a **Three Point Estimate**. Actual duration values are expected to be closer to the Most Likely value while there is some (usually smaller) chance they may approach the Optimistic and Pessimistic values.

While PERT used a calculation to derive an Expected Duration for each task, Monte Carlo simulation will pick a new ‘sample’ expected duration for each task, from within the ranges specified, for each iteration of the simulation.

The sample duration can be weighted using a probability distribution within the three point estimate.
A distribution curve defines how likely a specific duration will be sampled from the range specified by the optimistic/most likely/pessimistic values.

In this simplistic example, if asked to select any random $x$, it is more likely to be closer to the center of the distribution simply because there are more $x$'s under the peak of the curve.
Different distribution curves can be used to change the likelihood that values will be closer to the most likely or extreme values.

- **Optimistic**
- **Most Likely**
- **Pessimistic**

**Normal Distribution**

**Triangular Distribution**
Skewed Distributions

Some distributions allow for the Most Likely value to be skewed toward either the Optimistic or Pessimistic values.
Confidence Intervals allow us to model conditional estimates like ‘I’m 80% confident the duration will be between 8 and 10 days’.
Distribution Types
Selecting a Distribution Type

General Guidance

• Use historical data to determine an appropriate distribution
• Unless there is a compelling reason, do not use Uniform
• In the absence of specific guidance, use Lognormal
• Use Beta or Triangular if you need to specify the degree of skew
• Use Confidence Limits if the estimator wants to hedge

It really doesn’t make a lot of difference...
Distribution Types (again)
Event Uncertainty (Aleatoric)

Risks and Opportunities...

The three point estimate allows us to model our lack of knowledge regarding the final expected duration of a task but what about ‘things’ that may or may not happen (random events)?

• A module fails testing
• Required equipment isn’t available at the right time

These scenarios cannot be modelled simply using estimate uncertainty but they can be modelled using conditional and/or probabilistic branching.
Probabilistic Branching

Modelling random events that may impact our project is achieved using Probabilistic branching.

The Rework task will only be considered 20% of the time during the simulation.
Conditional branching lets us model alternatives based on dates.

Provided Fabrication is completed before 30Oct14 we can transport the equipment using route 1 otherwise we will have to use route 2.
The end result

The histogram shows the probability of the project completing on a specific date.

The S-Curve shows the probability of completion by a specific date.

CPM expected finish Jan 23
- 6% chance by Jan 23
- 50% chance by Feb 27
- 80% chance by Mar 18
- 100% chance by May 28

Most likely date Feb 19 (12.5%) although only 36% chance of finishing by then.
Using the results

If you have the luxury of telling your client when you can deliver, pick a date from the S-Curve that matches your appetite for risk. From the previous slide we might commit to March 18 because it gives us an 80% chance of completing by that date.

On the other hand, if you have to meet a date imposed by the client (for example January 31) and the level of confidence for that date is low (SRA is suggesting we only have a 12% chance of completing by January 31) then we’ll have to rework the schedule. More on that later.
More is better

As with any statistical sampling method, the larger the set of samples (simulations), the higher the confidence in the results.
Bi-Modal Distributions

A probability histogram may not necessarily exhibit a classic distribution curve.

The example shown here shows the range of likely finish dates for a project that contains branching.

This model contains a 25% chance of realizing an ‘opportunity’ that will bring in the project significantly earlier - but don’t bet the business...
Standard Deviation is a measure of the variability around the mean result.

1 Standard Deviation spans ≈ 68% of values

2 standard deviations spans ≈ 95% of values

3 standard deviations spans ≈ 99.5% of values

For example, if the standard deviation is 22 days then 99.5% of results will fall within ± 66 days of the mean date (a span of 132 days)

A larger value for standard deviation means a higher degree of variability in the outcome.
In the following schedule, the project completion (calculated by CPM) is being driven by the hardware tasks.
Adding uncertainty

While hardware development is estimated to take longer, the uncertainty for software development is greater.

We have applied a LogNormal distribution to both hardware and software tasks but increased the pessimistic duration for software tasks to 150%.

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Remaining Duration</th>
<th>Duration Distribution Type</th>
<th>Duration Optimistic</th>
<th>Duration Most Likely</th>
<th>Duration Pessimistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware vs Software (20)</td>
<td>560 hrs</td>
<td>(None)</td>
<td>90%</td>
<td>100%</td>
<td>110%</td>
</tr>
<tr>
<td>Initiate</td>
<td>40 hrs</td>
<td>(None)</td>
<td>90%</td>
<td>100%</td>
<td>110%</td>
</tr>
<tr>
<td>Development</td>
<td>480 hrs</td>
<td>(None)</td>
<td>90%</td>
<td>100%</td>
<td>110%</td>
</tr>
<tr>
<td>Hardware</td>
<td>480 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
</tr>
<tr>
<td>HW Task 1</td>
<td>160 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
</tr>
<tr>
<td>HW Task 2</td>
<td>160 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
</tr>
<tr>
<td>HW Task 3</td>
<td>160 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
</tr>
<tr>
<td>HW Task 4</td>
<td>160 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
</tr>
<tr>
<td>Software</td>
<td>432 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.35%</td>
<td>150%</td>
</tr>
<tr>
<td>SW Task 1</td>
<td>144 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.35%</td>
<td>150%</td>
</tr>
<tr>
<td>SW Task 2</td>
<td>144 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.35%</td>
<td>150%</td>
</tr>
<tr>
<td>SW Task 3</td>
<td>144 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.35%</td>
<td>150%</td>
</tr>
<tr>
<td>SW Task 4</td>
<td>144 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.35%</td>
<td>150%</td>
</tr>
<tr>
<td>Integration</td>
<td>40 hrs</td>
<td>(None)</td>
<td>90%</td>
<td>100%</td>
<td>110%</td>
</tr>
<tr>
<td>Delivery</td>
<td>0</td>
<td>(None)</td>
<td>90%</td>
<td>100%</td>
<td>110%</td>
</tr>
</tbody>
</table>
And the result is...

The analysis is suggesting that we only have a 4% chance of completing by the Aug 26 date suggested by CPM.

This is probably not a surprise given the level of uncertainty we added to the schedule but:-

As managers where should we focus our effort? The critical path?
Percent Critical / Criticality Index

This table shows both the Criticality calculated by CPM and also a Percent Critical Index calculated by the Monte Carlo simulation.

Observe that the hardware tasks (Critical according to CPM) were on the critical path less than 12% of the time.

We should focus our energy on the Software Tasks!
Sensitivity analysis also helps us focus our management effort, often portrayed using a ‘Tornado’ chart as shown below.

The chart clearly highlights the tasks contributing the most uncertainty to the project (or milestone) outcome.
Another output from Schedule Risk Analysis is a ‘Risk Adjusted Schedule’. This is a schedule with the task durations and dates adjusted to reflect a specific level of confidence.
Using Risk Adjusted Schedules

A Risk Adjusted Schedule is based on a specific level of confidence from the Schedule Risk Analysis, say 80%.

Ideally, your contract will be based on the risk adjusted schedule.

Never manage the project based on the risk adjusted schedule. The project is managed against the original CPM plan. The difference between the original CPM plan and the risk adjusted schedule agreed with the client is your contingency (buffer/margin).

This is exactly the same as how cost contingencies are built into contracts.
Schedule Margin is a technique used to manage schedule contingency.

In our example, CPM has an expected finish date of August 26. Our 80% confidence date from SRA is September 4. This is a difference of 7 working days.
Correlation allows us to model the impact of shared influencing factors such as:

- Task were estimated by the same person
- A common management team
- Tasks are performed by the same subcontractor
- Economic factors affecting the entire project

For example, if a task executed by a subcontractor performs well then it is possible that other tasks performed by the subcontractor will also perform well. If, on another simulation, a task performs badly, then other tasks performed by the same subcontractor may also perform badly.
### Example Correlation Data

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Remaining Duration</th>
<th>Distribution Type</th>
<th>Duration Optimistic</th>
<th>Duration Most Likely</th>
<th>Duration Pessimistic</th>
<th>Critical</th>
<th>Percent Critical</th>
<th>Correlations</th>
<th>Early Finish Histogram</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware vs Software</td>
<td>0</td>
<td>Normal</td>
<td>90%</td>
<td>100%</td>
<td>110%</td>
<td>True</td>
<td>100%</td>
<td></td>
<td>Graph</td>
</tr>
<tr>
<td>Initiate</td>
<td>40 hrs</td>
<td>Normal</td>
<td>90%</td>
<td>100%</td>
<td>110%</td>
<td>True</td>
<td>100%</td>
<td></td>
<td>Graph</td>
</tr>
<tr>
<td>Development</td>
<td>0</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
<td>True</td>
<td>15%</td>
<td></td>
<td>Graph</td>
</tr>
<tr>
<td>Hardware</td>
<td>0</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
<td>True</td>
<td>15%</td>
<td></td>
<td>Graph</td>
</tr>
<tr>
<td>HW Task 1</td>
<td>160 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
<td>True</td>
<td>15%</td>
<td>Estimator A(40%)</td>
<td>Graph</td>
</tr>
<tr>
<td>HW Task 2</td>
<td>160 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
<td>True</td>
<td>8%</td>
<td>Estimator A(40%)</td>
<td>Graph</td>
</tr>
<tr>
<td>HW Task 3</td>
<td>160 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
<td>True</td>
<td>8%</td>
<td>Estimator A(40%)</td>
<td>Graph</td>
</tr>
<tr>
<td>HW Task 4</td>
<td>160 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>99.39%</td>
<td>110%</td>
<td>True</td>
<td>15%</td>
<td>Estimator A(40%)</td>
<td>Graph</td>
</tr>
<tr>
<td>Software</td>
<td>0</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.5%</td>
<td>150%</td>
<td>False</td>
<td>85%</td>
<td></td>
<td>Graph</td>
</tr>
<tr>
<td>SW Task 1</td>
<td>144 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.5%</td>
<td>150%</td>
<td>False</td>
<td>85%</td>
<td>Estimator B(40%)</td>
<td>Graph</td>
</tr>
<tr>
<td>SW Task 2</td>
<td>144 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.5%</td>
<td>150%</td>
<td>False</td>
<td>42%</td>
<td>Estimator B(40%)</td>
<td>Graph</td>
</tr>
<tr>
<td>SW Task 3</td>
<td>144 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.5%</td>
<td>150%</td>
<td>False</td>
<td>43%</td>
<td>Estimator B(40%)</td>
<td>Graph</td>
</tr>
<tr>
<td>SW Task 4</td>
<td>144 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.5%</td>
<td>150%</td>
<td>False</td>
<td>85%</td>
<td>Estimator B(40%)</td>
<td>Graph</td>
</tr>
<tr>
<td>Integration</td>
<td>144 hrs</td>
<td>LogNormal</td>
<td>90%</td>
<td>115.5%</td>
<td>150%</td>
<td>False</td>
<td>85%</td>
<td></td>
<td>Graph</td>
</tr>
<tr>
<td>Delivery</td>
<td>0</td>
<td>Normal</td>
<td>90%</td>
<td>100%</td>
<td>110%</td>
<td>True</td>
<td>100%</td>
<td></td>
<td>Graph</td>
</tr>
</tbody>
</table>
The effect of Correlation

No Correlation

The mean finish date is roughly the same - but the Standard Deviation (range of uncertainty) has increased. If one task does well - others are also likely to do well (and vice versa).

40% Correlation

Each bar represents 1 day. (Markers show start of interval.)
Preparing for Schedule Risk Analysis

• A good quality schedule
  • Remove Hard Constraints
  • Ensure Level of Effort tasks are not driving the schedule
  • Appropriate Logic
• Software Schedule Validation
  • Barbecana Schedule Inspector
  • Steelray Project Analyzer
  • Deltek Acumen Fuse
  • Oracle Primavera Risk Analysis
That’s a lot of estimating...

Do we need to capture the three point estimates for every task?

Uncertainty is more significant when it affects the critical path.

Consider using a generic SRA to help identify potential critical tasks and consider if those estimates should be enhanced.
I don’t like the answer!

…and my boss will like it even less!!

Unfortunately, with a complex project with many parallel paths, merge bias alone will push the finish date out. Add in the fact that work tends to go worse than it does better and the result may be scary (and hard to believe/communicate).

Use the Sensitivity Tornado chart to see what’s driving the finish date. Is it what you expected?

With the top tasks on the Tornado chart can you?

• Reduce the uncertainty
• Reduce the durations? (apply more resources...)
• Restructure the logic to give those tasks float (slack)
One tip to validate the answer from a Monte Carlo simulation is to simply remove uncertainty and verify the answer is the same as the underlying CPM engine.

Some tools may require at least one task with uncertainty - just add some uncertainty to an isolated task.
Benefits of Schedule Risk Analysis

A much more realistic understanding of the likely completion dates
A better understanding of the tasks that may impact delivery
More appropriate levels of contingency
A greater chance of project success which leads to...

Improved profitability and client satisfaction
Thank you

Solutions for Microsoft Project and Oracle Primavera

info@barbecana.com

Risk Free trial software

www.barbecana.com

Questions about the presentation or Schedule Risk Analysis

John Owen

jowen@barbecana.com

Copyright © 2014 Barbecana Inc.