Calibrating the Project Planning Process

7th Annual CMMI Technology Conference & User Group
Denver, Colorado
Abstract No. 5757

November 12-15, 2007

Don Corpron
Division Manager - Six Sigma
Northrop Grumman Corporation
All Clipart is from Microsoft® Online and is used consistent with the End User License Agreement
A leading integrator of complex, mission-enabling systems

2006 Sales - ~$5.5B

17,500 employees in 47 states, 18 countries

2500 active contracts

Deep, legacy domain expertise in priority, high-growth segments

Trusted provider of mission critical end-to-end solutions

Focused on Program Performance

Data reflects 2006 results realigned for 2007 organization
Today we’ll discuss...

- The problem of protecting due-date performance
- How to “calibrate” your Project Planning Process, that is, determine the systematic error
- Mechanically, how to get data from Microsoft® Project to Minitab®
Often projects have difficulty finishing on time

Often projects have difficulty staying within budget

Often agreed to scope or specifications are cut from a project to maintain cost or schedule

All result in Customer irritation and perhaps undesirable consequences for the Project Manager
Project Manager’s Dilemma

Objective
Be a good Project Manager

Requirement
Complete Projects on Time (to be realistic)

Prerequisite
Add protection to tasks

Requirement
Respond to Customer’s need for Short Lead Time (quick response)

Prerequisite
Don’t add protection to tasks
Many reasons why projects may underperform:

- Inadequate or poorly defined requirements
- Competing priorities
- Clients changing their mind
- Unforeseen events (Murphy)
- Poor communications
- Unsatisfactory means of measuring progress
- Key people not available when needed

Need “protection” from these factors:

- Pressure to meet unrealistic due dates
- Factors outside our control
- Essential information not available on-time (designs, specifications, materials and authorizations)
- Too much re-work
- Lack of leadership or good management
Eliyahu Goldratt in his Critical Chain theory suggests that projects create buffers to protect due-date performance.

Project buffers are “zero-resource-tasks” in schedules that absorb the risks inherent in planning.

The issue for the Project Planning Process is to determine how big, and where to place these buffers in a project schedule.

This is not intended to be a dissertation on Critical Chain, just borrowing some ideas.
Some differences in project management

CRITICAL PATH

- Places high value on the completion of tasks on time,
- Employs techniques to minimize slack or float,
- Uses the amount of slack or float to set priorities

CRITICAL CHAIN

- Places low value on tasks being completed on time,
- Inserts buffers even on paths that are critical,
- Manages buffers to minimize unplanned expediting, overtime and other costly deviations from schedule
- Track both the baseline and actual durations of project tasks
- “Chunk” the project plan so tasks aren’t too different in size
- Analyze plan error with Xbar charts and Capability Charts
- Use information to develop “zero-resource-buffers” that protect due-date performance (by WBS or functional area)
“Toolbox” that you will need

- Microsoft® Project
- Microsoft® Excel
- Minitab®
Some methodology considerations

- Task durations are relatively independent; Start/Finish dates are not
- Task durations seldom are normally distributed which always presents analytic challenges
- Charting the averages of averages tends to produce normally distributed data even where the underlying data are not
- Large projects often have hundreds of tasks - presenting all the data points overwhelms the audience
Walkthrough example of a simple software project

Shows an approach to get data from Microsoft® Project to Mintab® to analyze schedule performance.
## Project Plan

### Summary Tasks
(aggregate subtasks)

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Duration</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Scope</td>
<td>4.5 days?</td>
<td>8/7/2007</td>
<td>8/13/2007</td>
</tr>
<tr>
<td>27 Identify modular/termed design parameters</td>
<td>1 day?</td>
<td>9/25/2007</td>
<td></td>
</tr>
<tr>
<td>28 Assign development staff</td>
<td>1 day?</td>
<td>9/26/2007</td>
<td></td>
</tr>
<tr>
<td>29 Develop code</td>
<td>15 days?</td>
<td>9/27/2007</td>
<td></td>
</tr>
<tr>
<td>31 Development complete</td>
<td>0 days?</td>
<td>10/23/2007</td>
<td></td>
</tr>
</tbody>
</table>

### Task Durations

- **Developer**
- **Developer**
- **Developer**
- **Developer**

### Milestones (Zero time and resource tasks that mark completion events)
Establish a baseline and monitor actual durations.

Baseline the Task Durations

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Baseline Duration</th>
<th>Duration</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verification Test</td>
<td>40 hrs</td>
<td>40 hrs</td>
<td>12/22/2007</td>
<td>1/1/2008</td>
</tr>
<tr>
<td>Deployment</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>1/2/2008</td>
<td>1/4/2008</td>
</tr>
</tbody>
</table>

Record the actual time (Performer’s logs)
Project file to Minitab® via Excel

- Save the Project file as an Excel workbook¹
- Export the Duration, Baseline Duration, and Finish Date²
- Strip out summary tasks and milestones³
- Fire up Minitab and read the Excel file

Notes:
1. Will put you into the export wizard
2. Time units need to be the same
3. Milestones are usually zero time and resources tasks that mark a completion
#### Compute the Plan error of each task

\[ \text{Error} = 100 \times \frac{\text{Duration} - \text{Baseline\_Duration}}{\text{Baseline\_Duration}} \]

<table>
<thead>
<tr>
<th>C3</th>
<th>C4-T</th>
<th>C5-D</th>
<th>C6-D</th>
<th>C7</th>
<th>C8 Error</th>
<th>C9</th>
<th>C10-T</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Start_Date</td>
<td>Finish_Date</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 day</td>
<td>4 hrs?</td>
<td>8/7/2007</td>
<td>08/2007</td>
<td></td>
<td>-75.000</td>
<td>08/2007</td>
<td></td>
</tr>
<tr>
<td>1 wk</td>
<td>1 day?</td>
<td>8/8/2007</td>
<td>08/2007</td>
<td></td>
<td>0.000</td>
<td>08/2007</td>
<td></td>
</tr>
<tr>
<td>1 day</td>
<td>1 day?</td>
<td>8/15/2007</td>
<td>08/2007</td>
<td></td>
<td>0.000</td>
<td>08/2007</td>
<td></td>
</tr>
<tr>
<td>1 day</td>
<td>4 hrs?</td>
<td>9/7/2007</td>
<td>09/2007</td>
<td></td>
<td>-75.000</td>
<td>09/2007</td>
<td></td>
</tr>
<tr>
<td>1 day</td>
<td>1 day?</td>
<td>9/10/2007</td>
<td>09/2007</td>
<td></td>
<td>100.000</td>
<td>09/2007</td>
<td></td>
</tr>
<tr>
<td>1 day</td>
<td>9/12/2007</td>
<td>09/2007</td>
<td></td>
<td>0.000</td>
<td>09/2007</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 day</td>
<td>9/14/2007</td>
<td>09/2007</td>
<td></td>
<td>200.000</td>
<td>09/2007</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 days</td>
<td>9/25/2007</td>
<td>10/2007</td>
<td></td>
<td>0.000</td>
<td>10/2007</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 day</td>
<td>10/14/2007</td>
<td>10/2007</td>
<td></td>
<td>100.000</td>
<td>10/2007</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Alternatively could do this with ±Days early or late
Construct an Xbar Chart in Minitab

- Sort the data by finish date
- Unstack and transpose data so they are in columns by month
- Select the common tests for special causes
- Interpret the results
The chart shows that the overall error is about 3%.

Control limits vary because of a differing number of tasks each month.

Tests performed with unequal sample sizes.
One example from a real project

Systematic 15% underestimate of duration

Xbar Chart of Planning Error
Work Package S

Tests performed with unequal sample sizes

Within the ±20% error range 56% of the time

Process Capability of Work Package S
Calculations Based on Lognormal Distribution Model

Overall Capability
Z.Bench 0.54
Z.LSL 0.47
Z.USL 0.29
Ppk 0.10

Exp. Overall Performance
% < LSL 31.93
% > USL 38.65
% Total 70.59

©2007 Northrop Grumman Space and Mission Systems Corporation. All Rights Reserved
Systematic 15% underestimate of duration

Within the ±20% error range 55% of the time

Xbar Chart of Planning Error
Work Package K

Process Capability of Work Package K
Calculations Based on Lognormal Distribution Model

Tests performed with unequal sample sizes
Parkinson’s Law

“WORK EXPANDS TO FILL THE TIME AVAILABLE”

Cyril Northcote Parkinson (1909-1993)

Naval historian and author of some sixty books, the most famous of which was his best seller Parkinson's Law, which led him to be also considered as an important scholar within the field of public administration.
Calibrating your projects can improve due-date performance

Source data already is available in many projects; don’t need to collect new data

Can compare differences among WBS’s, workgroups, or functional groups to determine inherent planning error
Questions?