Outline

• Issues
• Data Relationships
• Predicting Quality, a Case Study
• Observations
• Questions?
Can Quality be Predicted?

“Perfection is the enemy of the possible”
- Voltaire (paraphrased)

“Precision is not accuracy”
- William Horton
• What is Acceptable Quality?
  • Cost of Quality vs. Cost of Lack of Quality
• Different Standards and Definitions
  • How Many Severity Levels?
  • What about Changes?
• Lack of Relevant (or any) History
• Variability
Issues

• Cost of Quality
  • Microsoft Desktop Software vs. NORAD Missile Defense System Software

• Microsoft Business Model
  • Optimize Profit, Maximize Market Penetration, Planned Obsolescence, Increased Expectations

• NORAD “Business” Model
  • Avoid at all Costs False Positives and Negatives

• Cost of Lack of Quality
  • The Unfortunate Fate of Ashton Tate
Issues

Policies & Procedures Guide
Behavior (sometimes poorly)

What is Measured & Monitored
is Optimized
Issues: Variability

Defect Variability for a 100k SLOC Command & Control System
Issues: Variability

Defect Variability for a 100k SLOC Business System
Data Relationships: Time, Effort, Defects

- Time
- Effort
- Defects

SIZE

The Intelligence behind Successful Software Projects
Time/Quality Trade-off

Cum Defects Total

SOLUTION PANEL - <QSM Default Solution>

<table>
<thead>
<tr>
<th>C&amp;T</th>
<th>Life Cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Duration</td>
</tr>
<tr>
<td></td>
<td>Effort</td>
</tr>
<tr>
<td></td>
<td>Cost</td>
</tr>
<tr>
<td></td>
<td>Peak Staff</td>
</tr>
<tr>
<td></td>
<td>MTD</td>
</tr>
</tbody>
</table>


P1=15.8 MBI=2.6 EFF S LOC=100,000
### Schedule/Quality Trade-off

<table>
<thead>
<tr>
<th></th>
<th>Default</th>
<th>10% Compression</th>
<th>20% Compression</th>
<th>10% Extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duration Mths</td>
<td>25.9</td>
<td>23.3</td>
<td>20.7</td>
<td>28.5</td>
</tr>
<tr>
<td>Defect Count</td>
<td>1,033</td>
<td>1,316</td>
<td>1,715</td>
<td>849</td>
</tr>
<tr>
<td>% Change</td>
<td></td>
<td>27.4%</td>
<td>66.0%</td>
<td>-17.8%</td>
</tr>
</tbody>
</table>

100,000 Lines of Code Command & Control Project

Schedule Compression Comes at the Expense of Quality (and Cost)
Staff/ Quality Trade-off

<table>
<thead>
<tr>
<th>Staff Size / Quality Trade-off</th>
<th>Peak Staff 16</th>
<th>Peak Staff 32</th>
<th>% Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defects</td>
<td>1043</td>
<td>1411</td>
<td>35%</td>
</tr>
<tr>
<td>Effort Mths.</td>
<td>225</td>
<td>392</td>
<td>74%</td>
</tr>
</tbody>
</table>

100,000 Lines of Code Command & Control Project

The Further a Project Deviates from Optimal Staffing and/or Schedule, the More Pronounced the Impact on Cost and Quality
Defect Comparison

- 20% compression
- Peak staff 32
- Default
- 10% compression
- 10% extension

Defects - System Integration to Delivery

- Logged Solutions
- QSM 2008 G&C
- Avg. Line Style
- 1 Sign. Line Style
- Project Quality Bench
Case Study

- Huge Multi-year Development Project
  - Around 2.1 Million SLOC
- Hardware & Software Components
- 6 Increments with Significant Inter-Dependencies and Overlap
- Size, Schedule, Staffing Provided from 1st Three (Completed) Increments
- Defect Data Thru Dec., 2008 Provided
- Quality at Deployment Most Critical Factor
- Desired Implementation Date July, 2009
Case Study

- Completed Increments Modeled in Estimating Tool
- Remaining Increments Modeled Based on Demonstrated Productivity & Projected Staffing
- Defects in Models Tuned to Reflect Defects Discovered Thru Dec., 2008
- Increments Combined to Provide Program Level View
Case Study

Recreation of Completed Increment
Case Study

Defects Modeled from Actual Performance
Case Study

By Increment

Cumulative
Case Study Calibrated Defect Model

390 Defects in First Month of Operation (Aug – 09)
Case Study Observations

- Last Increment Estimated to Complete 10 Months Late
  - Productivity Modeled on Completed Increments
- 1,175 Projected Defects Remaining at Desired Implementation Date (July, 2009)
  - Approximately 1 Defect Encountered for Every 2 Hours of Operation during First Month in Operation
  - Defect Rate Unacceptable
Case Study Conclusion

• Go-Live Date Postponed to Oct., 2010
  • Model Predicts 4 Remaining Defects
  • Initial Defect Rate in Production Around 1 per Month

• Decision to Postpone Implementation May be Result of Schedule Slippage
  • Quality Model Provided Support for Decision
Case Study

**Strengths**
- Good Project Metrics (Schedule, Size, Defects, Staffing) Make for Better Modeling
- Defect Rate Close to Industry Average (90%)

**Weaknesses**
- Models for Last 3 Increments are Estimates
  - Assume They Will Behave According to Plan (Schedules for Increments 4 & 5 Did Not)
  - No Way to Determine Potential Impact of Remaining Defects
- Difficult to Model Interdependencies
Conclusion

• Modeling is a Useful Tool, when Properly Calibrated, to Predict Residual Defects and Defect Discovery Rates

• Information Valuable in Determining when a Product is Sufficiently Stable to Go-Live
• Organizations Have Quality Profiles that are Strongly Influenced by how they Develop Software
  • Business Model May Play a Key Role
• Historical Performance is a Sound Basis for Predicting Future Performance
• Schedule and Staffing Levels Affect Defect Creation
  • Influence More Pronounced the Greater the Deviation from the Norm
• Modeling can Provide Answers
Questions?