Process Maturity Estimation Model Validation

Quantifying the Value of Improving Software Process Maturity

Stuart Glickman
Quality Analysis & Registration
September 24, 1997

BELLCORE PROPRIETARY - INTERNAL USE ONLY
This document contains proprietary information that shall be distributed, copied, or made available only within Bellcore, except with written permission of Bellcore.

Process Maturity Estimation Model Validation

Agenda

- Goals and Objectives
- Why Estimate?
- PML to PMAT to Process Maturity Estimation Model
- PML, ISO 9000, and TR-179 Mapping Results
- Findings
- Validation Methodology
- Validation Results:
  - Schedule
  - Delivered Defects
  - Headcount
- How Easy is it to Collect?
- What Does it all Mean?
- Next Steps

BELLCORE PROPRIETARY - INTERNAL USE ONLY. See proprietary restrictions on this page.
Process Maturity Estimation Model Validation

**Goals and Objectives**

- Develop a software project estimation model that is easily usable by Customers
  - Base model on the fact that process improvement is used to reduce delivered defects to Customers
  - Quantify the models of process improvement: PML, ISO, and TR179 to remove the subjectivity (1996)
- Validate model developed using Value Added Cases
- Determine if model can be used to analyze a supplier's future release for
  - Delivered Defects
  - Development Schedule
  - Development Effort or Cost

**Why Estimate?**

- How Customer can use estimates
  - Early Determination:
    - realistic delivery date (support deployment strategy)
    - expected price of a release (price = cost x mark-up)
    - expected delivered software defects (support customer site maintenance strategy)
  - General Benefit of Estimation
    - Can determine what factors to address timely and effectively if cost, schedule, or quality are a problem
    - May need to perform tradeoff analysis between feature content and price or delivery
Software Quality Standards and Tools Used on this Project

- TR-NWT-179: Bellcore's Software Quality Program Generic Requirements
  - Defines highest level of quality management system
- ISO 9000-3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software
- PML: Bellcore's Process Maturity Level (QPS 88.001)
  - Measures coverage of TR179's major requirements
  - 116 questions
  - Used to quantify maturity

Software Quality Standards and Tools Used on this Project

- Checkpoint: SPR Software Estimation tool (SPR, Capers Jones)
  - Used to estimate defects, effort, and schedule
  - Uses various personnel experience factors, sizing, and complexity
  - Large variable with hundreds of variables
- COCOMO 2: Software Estimation tool and algorithms (Prof. Barry Boehm, USC)
  - Published algorithms to estimate project effort and schedule
  - Uses project personnel experience, sizing, and complexity
  - Maps to the SEI CMM maturity level (PMAT variable)
PML, ISO 9000, and TR-179 Mapping

Relationship among Quality System Standards

Going from PML to Checkpoint to Process Maturity Estimation Model
Going from PML to PMAT to Process Maturity Estimation Model

<table>
<thead>
<tr>
<th>PML 6</th>
<th>COCOMO2</th>
</tr>
</thead>
<tbody>
<tr>
<td>PML 5</td>
<td>PMAT 0</td>
</tr>
<tr>
<td>PML 4</td>
<td>PMAT 1</td>
</tr>
<tr>
<td>PML 3</td>
<td>PMAT 2</td>
</tr>
<tr>
<td>PML 2</td>
<td>PMAT 3</td>
</tr>
<tr>
<td>PML 1</td>
<td>PMAT 4</td>
</tr>
<tr>
<td></td>
<td>PMAT 5</td>
</tr>
</tbody>
</table>

Delivered Defects Schedule Cost

Process Maturity Estimation Model

Quality System Tools: COCOMO2's PMAT Variable

- Estimates software development project effort based on SEI CMM process maturity level
- Assign value to 18 SEI CMM Key Process Areas
- Algorithm easily applied to several Bellcore Projects
- Bellcore PML levels correspond to PMAT levels
### Model Conformance Levels of Quality System Standards

<table>
<thead>
<tr>
<th>Increasing Process Maturity</th>
<th>Checkpoint Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>PML 6 World Class</td>
<td></td>
</tr>
<tr>
<td>PML 5 Optimizing</td>
<td>TR179</td>
</tr>
<tr>
<td>PML 4 Managed</td>
<td>ISO 9000-3</td>
</tr>
<tr>
<td>PML 3 Defined</td>
<td></td>
</tr>
<tr>
<td>PML 2 Repeatable</td>
<td></td>
</tr>
<tr>
<td>PML 1 Initial</td>
<td>ISO 9000-3</td>
</tr>
</tbody>
</table>

**1996 Findings**

- Customers receive the most benefit by a supplier progressing to PML level 3 (significant decrease in delivered defects).
  - After PML level 3, Customers continue to get moderate improvement in delivered defects.
- Suppliers receive the most benefit by progressing to PML level 6 (maximum development cost reduction).
- Significant development cost and delivered defect improvements by going from ISO to TR-179 compliant.
1996 Work Program: PML Mapping Result

**Effect of PML on Delivered Defects**

[Graph showing the relationship between PML and delivered defects.]

1996 Work Program: PML Mapping Result

**Effect of PML on Development Cost**

[Graph showing the percentage difference from PML 1 cost.]

BELLCORE PROPRIETARY - INTERNAL USE ONLY. See proprietary restrictions on this page.
1997 Work Program: Process Maturity Estimation Model

Validation Methodology

- Supplier reaches a higher PML level.
- Bellcore surveillance engineer generates Value Added case
  » Create pre and post Checkpoint estimation model runs reflecting supplier's improvement.
  » Compare Checkpoint outputs with RQMS (Supplier's quality data)
  » Share Checkpoint model settings and outputs with Supplier
- Use difference between the two Checkpoint model run estimates for delivered defects to calculate Value Added. Value Added are defects prevented in the field times cost of a defect.
- Transfer a few of the variables from the Checkpoint run and put them into appropriate Process Maturity Estimation Model. All other Checkpoint variables are pre-assigned by knowing the Supplier's PML level.
- Compare the Checkpoint and Process Maturity Estimation Model runs, are they in reasonable agreement?

\[
\text{Value Added} = f(PML_N) - f(PML_M)
\]

BELLCORE PROPRIETARY - INTERNAL USE ONLY. See proprietary restrictions on this page.

BELLCORE
### Process Maturity Estimation Model Validation

**Data Sampling Statistics**

- PML levels: A data point was available for every PML level
- Number companies: 5 companies were involved in the validation spanning switch, transport, and voice messaging

<table>
<thead>
<tr>
<th>Variables needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Programming language type and lines of code (New and/or Change, Base)</td>
</tr>
<tr>
<td>- Overlap constraints (How accelerated is the development)</td>
</tr>
<tr>
<td>- How often is the application used by the customer</td>
</tr>
<tr>
<td>- New Problem and data complexity</td>
</tr>
<tr>
<td>- Project Goals (What level of quality is the software development project managed to)</td>
</tr>
</tbody>
</table>
Validation Results

**Process Maturity Estimation Model**

<table>
<thead>
<tr>
<th>Schedule (months)</th>
<th>Model</th>
<th>Actual</th>
<th>Percent Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1A</td>
<td>18.8</td>
<td>17.7</td>
<td>6.0%</td>
</tr>
<tr>
<td>1B</td>
<td>24.8</td>
<td>25.7</td>
<td>3.6%</td>
</tr>
<tr>
<td>2A</td>
<td>17.8</td>
<td>17.6</td>
<td>1.2%</td>
</tr>
<tr>
<td>2B</td>
<td>17.7</td>
<td>17.5</td>
<td>1.2%</td>
</tr>
<tr>
<td>3A</td>
<td>49.2</td>
<td>57</td>
<td>16.4%</td>
</tr>
<tr>
<td>3B</td>
<td>45</td>
<td>55</td>
<td>22.2%</td>
</tr>
<tr>
<td>4A</td>
<td>11.9</td>
<td>12.7</td>
<td>6.8%</td>
</tr>
<tr>
<td>4B</td>
<td>13.3</td>
<td>12.3</td>
<td>8.0%</td>
</tr>
<tr>
<td>5A</td>
<td>23.3</td>
<td>21.3</td>
<td>9.0%</td>
</tr>
<tr>
<td>5B</td>
<td>22.3</td>
<td>26.5</td>
<td>18.7%</td>
</tr>
<tr>
<td>5C</td>
<td>64.9</td>
<td>65.1</td>
<td>1.8%</td>
</tr>
<tr>
<td>Totals</td>
<td>384.3</td>
<td>379.2</td>
<td>1.3%</td>
</tr>
</tbody>
</table>

**Delivered Defects**

<table>
<thead>
<tr>
<th>Schedule (months)</th>
<th>Model</th>
<th>Actual</th>
<th>Percent Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1A</td>
<td>1158</td>
<td>778</td>
<td>32.5%</td>
</tr>
<tr>
<td>2A</td>
<td>239</td>
<td>59.3</td>
<td>32.6%</td>
</tr>
<tr>
<td>2B</td>
<td>181</td>
<td>59.3</td>
<td>32.6%</td>
</tr>
<tr>
<td>3A</td>
<td>270</td>
<td>27.9</td>
<td>32.6%</td>
</tr>
<tr>
<td>3B</td>
<td>131</td>
<td>29.4</td>
<td>32.6%</td>
</tr>
<tr>
<td>4A</td>
<td>4</td>
<td>4.6</td>
<td>4.6%</td>
</tr>
<tr>
<td>4B</td>
<td>6</td>
<td>6.4</td>
<td>6.4%</td>
</tr>
<tr>
<td>5A</td>
<td>381</td>
<td>148.1</td>
<td>130.7%</td>
</tr>
<tr>
<td>5B</td>
<td>245</td>
<td>143.7</td>
<td>130.7%</td>
</tr>
<tr>
<td>5C</td>
<td>145</td>
<td>66</td>
<td>66.1%</td>
</tr>
<tr>
<td>Totals</td>
<td>2556</td>
<td>824.9</td>
<td>912.8%</td>
</tr>
</tbody>
</table>

**Headcount**

<table>
<thead>
<tr>
<th>Schedule (months)</th>
<th>Model</th>
<th>Actual</th>
<th>Percent Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1A</td>
<td>300+</td>
<td>59.3</td>
<td>32.6%</td>
</tr>
<tr>
<td>2A</td>
<td>59.3</td>
<td>27.9</td>
<td>32.6%</td>
</tr>
<tr>
<td>2B</td>
<td>29.4</td>
<td>29.4</td>
<td>0%</td>
</tr>
<tr>
<td>3A</td>
<td>4.6</td>
<td>4.6</td>
<td>4.6%</td>
</tr>
<tr>
<td>3B</td>
<td>4.6</td>
<td>4.6</td>
<td>4.6%</td>
</tr>
<tr>
<td>4A</td>
<td>66</td>
<td>66.1</td>
<td>66.1%</td>
</tr>
<tr>
<td>4B</td>
<td>66.1</td>
<td>66.1</td>
<td>0%</td>
</tr>
<tr>
<td>5A</td>
<td>20%</td>
<td>9.5%</td>
<td>9.5%</td>
</tr>
<tr>
<td>5B</td>
<td>9.5%</td>
<td>9.5%</td>
<td>9.5%</td>
</tr>
<tr>
<td>5C</td>
<td>9.5%</td>
<td>9.5%</td>
<td>9.5%</td>
</tr>
</tbody>
</table>

**Process Maturity Estimation Model**

400 Estimation Variables to 8

![Process Maturity Estimation Model Diagram](image)

Without Sacrificing Accuracy

BELLCORE PROPRIETARY - INTERNAL USE ONLY. See proprietary restrictions on this page.
Conclusion

- Process Maturity Estimation Model produces excellent estimates of:
  - Schedule
  - Delivered Defects
  - Headcount

Note: Headcount x Schedule x Wage Rate = Cost

How easy is it to collect required variables?

- Programming language type and lines of code (New and/or Change, Base) - Recommend using count function points, size the requirements document.
- Overlap constraints (How accelerated is the development) - Actual data acquired during validation trial: 80% (2), 85% (4), None (5).
- How often is the application used by the customer - Actual data acquired during validation trial: Setting =
  - 4 (Continuous runs, 3),
  - 3.15 (3),
  - 3 (3),
  - 2.5 (2)

Checkpoint settings go from 1 to 5.
How easy is it to collect required variables?

- New data complexity - 3.95 (1), 3.05 (1), 3 (2), 2.85 (1), 2.5 (6)
- Project Goals (What level of quality is the project managed to)
  - Actual data acquired during validation trial:
    5 (Highest Quality, shortest schedule, 4),
    4 (Highest Quality, normal schedule, 3)
    2 (Shortest Development Schedule, extra staff, 4)

How easy is it to collect required variables?

- PML level of the organization needs to be known
- Only a few Process Maturity Estimation Model variables are needed requiring only a few values to choose from.
- Sizing needs to be done in Function Points and software development language needs to be known
- Conclusion - A few variables are now required to obtain accurate software project estimates.
What Does It All Mean?

- Easier to use
- Lower level of knowledge required
- Better results with less effort
- More accurate

Process Maturity Estimation Model

**Next Steps**

- Continue using Process Maturity Estimation Model for Value Added case generation
- Start transfer of Process Maturity Estimation Model to Customers for their use in the customer-supplier relationship