CORADMO in 2001: A RAD Odyssey

Cyrus Fakharzadeh
fakharza@usc.edu

16th International Forum on COCOMO and Software Cost Modeling
Introduction

RAD (Rapid Application Development)

• an application of any of a number of techniques or strategies to reduce software development cycle time

CORADMO

• COCOMO II model extension
• Focuses on software development schedules and costs using RAD techniques
Constructive Rapid Application Development Model

• Calculates/predicts
  – schedule (months, M)
  – personnel (P)
  – adjusted effort (person-months, PM)

• Based on
  – Effort and schedule distribution to the various phases
  – Selected schedule driver ratings impacts on the M, P, and PM of each phase.
Six Classes of Strategies for RAD

- Reuse, Very High-level Languages (RVHL)
- Development Process Reengineering (DPRS)
- Collaboration Support (CLAB)
- Architecture, Risk Resolution (RESL)
- Prepositioning Assets (PPOS)
- RAD Capability of Personnel (RCAP)
RAD Opportunity Tree

Eliminating Tasks
- Business process reengineering - O
- Development process reengineering - DPRS
- Reusing assets - RVHL
- Applications generation - RVHL
- Design-to-schedule - O
- Tools and automation - O
- Work streamlining (80-20) - O
- Increasing parallelism - RESL

Reducing Time Per Task
- Reducing failures - RESL
- Reducing their effects - RESL
- Early error elimination - RESL
- Process anchor points - RESL
- Improving process maturity - O
- Collaboration support - CLAB

Reducing Single-Point Failure Risks
- Minimizing task dependencies - DPRS
- Avoiding high fan-in, fan-out - DPRS
- Reducing task variance - DPRS
- Removing tasks from critical path - DPRS

Reducing Backtracking
- Prepositioning resources - PPOS
- Nightly builds, testing - PPOS
- Weekend warriors, 24x7 development - PPOS

Activity Network Streamlining
- More RAD experience - RCAP
- O: covered by classic cube root model

Increasing Effective Workweek

Better People and Incentives

Transition to Learning Organization
Background

COCOMO II Schedule shortfalls:
• Reflects projects optimized for minimum cost
• Model does not address RAD strategies

COCOMO II.2000 Duration Calculation

Cube Root Law: Months \sim 3.67 \ (Person-Months)^f

where \ 0.28 \leq f \leq 0.34

CORADMO differs from COCOMO:
• A square root instead in computing the number of months needed to complete a small project
• Square root law (i.e. f = 0.5)
COPSEMO

Constructive Phased Schedule and Effort Model

Inputs: the baseline effort and schedule from COCOMO II

Outputs: the effort and schedule by phase needed for CORADMO.

Phases: Inception, Elaboration, Construction, and Transition

Source: MBASE/RUP (Model-Based Architecting & Software Engineering/Rational Unified Process) life-cycle model
COPSEMO Months as F(PM)

- Previous RAD/COCOMO
  - Bridge for large projects

- ~3\(^{\text{cube-root}}\)
  - COCOMO II
    - Minimum cost

- Square root
  - Minimum schedule
    - Small projects

- Previous RAD/COCOMO
- New COPSEMO-M [Square Root]
- Original COPSEMO-M
- CII-M [Cube Root]
Physical Model

COCOMO II cost drivers

COCOMO II.2000 via COCOMO_charts.xls

RESL; Baseline effort, schedule

Phase Distributions (COPSEMO Extension)

RESL; Baseline effort, schedule

Effort, schedule by phase; No SCED

RAD Extension (CoRADMo.xls)

RAD effort, schedule by phase

RVHL
DPRS
CLAB
PPOS
RCAP

Schedule calculated; SCED removed; PM & M distributed per phase

Baseline Effort & Sched.
Results

• Delphi Exercise Forms distributed
• Experts from Academia, Industry and Government
  – Affiliates, Professors, and Researchers
• EMR (Effort Multiplier Range)
  – Highest divided by Lowest across the rating scale for effort
• SMR (Schedule Multiplier Range)
  – Highest divided by Lowest across the rating scale.
## % Effort per phase

<table>
<thead>
<tr>
<th>Phase</th>
<th>Original</th>
<th>Delphi Mean</th>
<th>Delphi Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inception – I</td>
<td>6.0</td>
<td>10.29</td>
<td>4.75</td>
</tr>
<tr>
<td>Elaboration – E</td>
<td>24.0</td>
<td>23.71</td>
<td>5.38</td>
</tr>
<tr>
<td>Construction – C</td>
<td>76.0</td>
<td>71.29</td>
<td>12.00</td>
</tr>
<tr>
<td>Total I, E, &amp; C</td>
<td>106.0</td>
<td>105.29</td>
<td>4.24</td>
</tr>
</tbody>
</table>
% Schedule per phase

<table>
<thead>
<tr>
<th></th>
<th>Original</th>
<th>Delphi Mean</th>
<th>Delphi Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inception – I</td>
<td>12.5</td>
<td>15.71</td>
<td>4.99</td>
</tr>
<tr>
<td>Elaboration – E</td>
<td>37.5</td>
<td>29.86</td>
<td>5.64</td>
</tr>
<tr>
<td>Construction – C</td>
<td>62.5</td>
<td>63.14</td>
<td>11.94</td>
</tr>
<tr>
<td>Total I, E, &amp; C</td>
<td>112.5</td>
<td>108.71</td>
<td>6.94</td>
</tr>
</tbody>
</table>
Reuse, Very High-level Languages

Degree to which re-use of artifacts other than code and/or very high-level languages are utilized

<table>
<thead>
<tr>
<th>RVHL</th>
<th>EMR</th>
<th>SMR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Original</td>
<td>Delphi Mean</td>
</tr>
<tr>
<td>Inception</td>
<td>1.16</td>
<td>1.25</td>
</tr>
<tr>
<td>Elaboration</td>
<td>1.07</td>
<td>1.25</td>
</tr>
<tr>
<td>Construction</td>
<td>1.00</td>
<td>1.16</td>
</tr>
</tbody>
</table>
**Development Process Reengineering**

Measures the degree to which the project and organization allow and encourage streamlined or reengineered development processes

<table>
<thead>
<tr>
<th>DPRS</th>
<th>EMR</th>
<th>SMR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Original</td>
<td>Delphi Mean</td>
</tr>
<tr>
<td>Inception</td>
<td>1.33</td>
<td>1.32</td>
</tr>
<tr>
<td>Elaboration</td>
<td>1.21</td>
<td>1.24</td>
</tr>
<tr>
<td>Construction</td>
<td>1.21</td>
<td>1.30</td>
</tr>
</tbody>
</table>
Collaboration Support

Accounts for Multisite tool support plus special collaboration tools, yields a reduced effect on schedule and effort

<table>
<thead>
<tr>
<th>CLAB</th>
<th>EMR</th>
<th>SMR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Original</td>
<td>Delphi Mean</td>
</tr>
<tr>
<td>Inception</td>
<td>1.51</td>
<td>1.34</td>
</tr>
<tr>
<td>Elaboration</td>
<td>1.34</td>
<td>1.23</td>
</tr>
<tr>
<td>Construction</td>
<td>1.18</td>
<td>1.23</td>
</tr>
</tbody>
</table>
Architecture, Risk Resolution

Same as COCOMO II RESL

<table>
<thead>
<tr>
<th>RESL</th>
<th>EMR Original</th>
<th>Delphi Mean</th>
<th>Delphi Standard Deviation</th>
<th>SMR Original</th>
<th>Delphi Mean</th>
<th>Delphi Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inception</td>
<td>1.00</td>
<td>1.24</td>
<td>0.34</td>
<td>1.00</td>
<td>1.21</td>
<td>0.35</td>
</tr>
<tr>
<td>Elaboration</td>
<td>1.00</td>
<td>1.24</td>
<td>0.34</td>
<td>1.00</td>
<td>1.23</td>
<td>0.35</td>
</tr>
<tr>
<td>Construction</td>
<td>1.00</td>
<td>1.27</td>
<td>0.33</td>
<td>1.33</td>
<td>1.35</td>
<td>0.29</td>
</tr>
</tbody>
</table>
Prepositioning Assets

Degree to which assets are pre-tailored to a project and furnished to the project for use on demand

<table>
<thead>
<tr>
<th>PPOS</th>
<th>EMR</th>
<th>SMR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Original</td>
<td>Delphi Mean</td>
</tr>
<tr>
<td>Inception</td>
<td>1.10</td>
<td>1.13</td>
</tr>
<tr>
<td>Elaboration</td>
<td>1.10</td>
<td>1.14</td>
</tr>
<tr>
<td>Construction</td>
<td>1.10</td>
<td>1.20</td>
</tr>
</tbody>
</table>
## RAD Capability of Personnel

Accounts for the effects of RAD personnel capability & experience in RAD projects

<table>
<thead>
<tr>
<th>RCAP</th>
<th>EMR</th>
<th>SMR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Original</td>
<td>Delphi Mean</td>
</tr>
<tr>
<td>Inception</td>
<td>1.50</td>
<td>1.48</td>
</tr>
<tr>
<td>Elaboration</td>
<td>1.50</td>
<td>1.44</td>
</tr>
<tr>
<td>Construction</td>
<td>1.50</td>
<td>1.46</td>
</tr>
</tbody>
</table>
Example

With RCAP = Nominal => PM=25, M=5, P=5
Result: The square root law: 5 people for 5 months: 25 PM

With RCAP=XH (Extra High) => PM=20, M=2.8, P=7.1
Result: A super team can put on 7 people and finish in 2.8 months: 20 PM

With RCAP = XL (Extra Low) => PM=30, M=7, P=4
Result: Trying to do RAD with an unqualified team makes them less efficient (30 PM)
RCAP Effort/Schedule Effect

\[ M = \begin{cases} 
3.7 \times \text{(Cube root)} & \text{RCAP = XL} \\
3 \times \text{(Cube root)} & \text{RCAP = XH} \\
\text{Square root} & \text{RCAP = XH} 
\end{cases} \]
Next Steps

- Complete another Delphi Round
- Gather more RAD data
  ➢ Please contact me if you have some
- Analyze Data from RAD projects
- Bayesian Analysis
- Calibrate Model
Issues for Breakout Group

- Treatment of square root, cube root models
- Treatment of RAD drivers
- Relevance to your RAD experience
- Expediting data collection