The Effectiveness Formula

A Quantitative measure of Organization Capability
Software Development isn’t Easy

Programming a computer does require intelligence. Indeed, it requires so much intelligence that nobody really does it very well. Sure, some programmers are better than others, but we all bump and crash around like overgrown infants. Why? Because programming computers is by far the hardest intellectual task that human beings have ever tried to do. Ever

Chaos 2013 Software Project Survey

Success Rate

- Success: 39
- Failed: 18
- Challenged: 43
Software Problems

- Unreliable
- Late delivery
- Modification costs prohibitive
- Impossible to maintain
- Inadequate performance
- Product exceeds budget costs

1968 CS Conference, Munich, Germany
Productivity Gains -- Technology

- Structured Analysis
- Structured Design
- PWB
- Process Maturity
- Ada
- OOD
- Structured Programming
- 3rd Generation Languages
Effectiveness Formula

\[ E = C[M(CS)] \]

where
\[ E = \text{Effectiveness} \]
\[ C = \text{Communication skills (0 – 1)} \]
\[ M = \text{Management concept awareness (0 – 1)} \]
\[ CS = \text{Computer science technical ability (0 – 1)} \]

Effectiveness Measures

- Organization capability (0 – 1)
- Source lines per person month
- Milestones per day
- Dollars per source line
- Basic Technology Constant
- COCOMO cost drivers
Could not achieve correlation with all data until PEOPLE (communications and management) included in organization capability definitions.
Traditional Capability Rating

MAJOR

CS

• Programming ability
  – Education
  – IQ
  – Problem solving skills

MINOR

C & M

• Efficiency
• Communication ability
• Cooperation

Traditional data does not work for agile development environments
Poor management can increase software costs more rapidly than any other factor…..

Despite this cost variation, COCOMO does not include a factor for management quality, but instead provides estimates which assume that the project will be well managed…..

Weinberg’s Analysis of COCOMO Drivers

Management Research Highlights

- General management concepts, 1911
  - Planning
  - Organizing
  - Commanding
  - Coordinating
  - Controlling
- Hawthorne experiment (1924-1932)
- People impacts, Mayo, 1933
- Lockheed Skunk Works, Johnson, 1940s
- Theory X/ Theory Y, McGregor, 1960
- CMM, Humphrey, 1989
While there are many unique characteristics to software, they all require more management discipline, not less. Managers should thus demand detailed plans, tracking systems, and periodic technical and management reviews. **Software management should be entirely traditional, only more so.** Unfortunately, many managers who insist on these items for hardware let their software teams get by without them.

Productivity Impact of Motivation

Area affected by motivation

EMPLOYEE

Percent of Ability

80 to 90 percent

20 to 30 percent
Extreme Capability Rating

- Motivation
- Management style
- Use of team methods
  - Communication ability
  - Cooperation
- Hawthorne Effect
- Working environment
  - Noise level
  - Individual working (thinking) space
  - Proximity of team members
- Problem solving skills
- Programming ability
Components of Communication

How Much Communication Here?

(0 – 1) scale

= 0.07!
Is This Arrangement Any Better?

Not much, but if one used the phone or Skype?
Pulse of a Project

- Rate at which a project makes progress
  - Time for information to move from one mind to another (Communication)
    - Elapsed time between issue and decision (Management)
  - Three important factors
    - Quality and sufficiency of information
      - Distance to decision makers
      - Degradation of information
- P-80 fighter prototype in 247 days?
### Effective Capability Examples: Controlled Experiments

<table>
<thead>
<tr>
<th>Project Type (date)</th>
<th>Productivity Gain (%)</th>
<th>Error Rate Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pair Programming* (1975)</td>
<td>136</td>
<td>$10^{-3}$</td>
</tr>
<tr>
<td>War Room (1998)</td>
<td>98**</td>
<td>$10^{-1}$</td>
</tr>
</tbody>
</table>

* Two programmers/ one workstation
** Development teams first use of Ada programming language
Theory X Management

1. Work is inherently distasteful to most people.
2. Most people are not ambitious, have little desire for responsibility, and prefer to be directed.
3. Most people have little capacity for creativity in solving organizational problems.
4. Motivation occurs only at the physiological and safety levels.
5. Management philosophy: Plan, Organize, Command, Coordinate, Control
Theory Y Management

1. Work is as natural as play, if conditions are favorable.
2. Self-control is often indispensable in achieving organizational goals.
3. The capacity for creativity in solving organizational problems is widely distributed in the population.
4. Motivation occurs at the social, esteem, and self-actualization levels, as well as physiological and security levels.
Basic Technology Constant Factors

C & M
- Analyst Capability
- Programmer Capability

CS
- Application Experience
- Modern Practices
- Modern Tools
- Terminal Response Time
- Hardcopy Turnaround Time
Effectiveness Formula Redux

\[ E = CM (CS) \]

where

\( E \) = Effectiveness

\( CM \) = Communications /Management ability (0 – 1)

\( CS \) = Computer science technical ability (0 – 1)

Thirty five years of validation data !!
# Effectiveness vs Industry Rating

<table>
<thead>
<tr>
<th>Effectiveness Value (OCR)</th>
<th>Percentile Rating</th>
<th>Basic Technology Constant</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>100</td>
<td>20,000</td>
</tr>
<tr>
<td>0.9</td>
<td>99</td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>98</td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>96</td>
<td></td>
</tr>
<tr>
<td>0.6</td>
<td>94</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>88</td>
<td></td>
</tr>
<tr>
<td>0.4</td>
<td>80</td>
<td>9580 (OCR=0.45)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8635 (OCR=0.40)</td>
</tr>
<tr>
<td>0.3</td>
<td>65</td>
<td>7350 (OCR=0.32)</td>
</tr>
<tr>
<td>0.2</td>
<td>30</td>
<td>6735 (OCR=0.28)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5707 (OCR=0.24)</td>
</tr>
<tr>
<td>0.1</td>
<td>10</td>
<td>4570 OCR= (0.18)</td>
</tr>
</tbody>
</table>

Industry Avg. = 0.25
## 1975 Experiment

<table>
<thead>
<tr>
<th></th>
<th>Traditional</th>
<th>Pair Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyst Capability</td>
<td>1.0</td>
<td>0.72</td>
</tr>
<tr>
<td>Domain Experience</td>
<td>1.29</td>
<td>1.0</td>
</tr>
<tr>
<td>Programmer Capability</td>
<td>1.0</td>
<td>0.70</td>
</tr>
<tr>
<td>Development Time (Mo)</td>
<td>29</td>
<td>21</td>
</tr>
<tr>
<td>Productivity (SLOC/PM)</td>
<td>202</td>
<td>536 [ X2.36]</td>
</tr>
<tr>
<td>Capability Rating</td>
<td>0.2</td>
<td>0.53</td>
</tr>
<tr>
<td>Percentile Rating (%)</td>
<td>38</td>
<td>89</td>
</tr>
<tr>
<td>Ctb</td>
<td>5,373</td>
<td>12,118</td>
</tr>
</tbody>
</table>
## Capability Calculator

### Complexity, D

<table>
<thead>
<tr>
<th>Cte Calculation</th>
<th>Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complexity, D</td>
<td>8</td>
</tr>
<tr>
<td>Effective Size</td>
<td></td>
</tr>
<tr>
<td>SLOC, New</td>
<td>50,000</td>
</tr>
<tr>
<td>SLOC, Modified</td>
<td>0</td>
</tr>
<tr>
<td>SLOC, Reused</td>
<td>0</td>
</tr>
<tr>
<td>Size, Total</td>
<td>50,000</td>
</tr>
<tr>
<td>Size, Effective</td>
<td>50,000</td>
</tr>
</tbody>
</table>

### Environment

<table>
<thead>
<tr>
<th>Cte Calculation</th>
<th>Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size, Total</td>
<td></td>
</tr>
<tr>
<td>Size, Effective</td>
<td></td>
</tr>
<tr>
<td>SLOC, New</td>
<td></td>
</tr>
<tr>
<td>SLOC, Modified</td>
<td></td>
</tr>
<tr>
<td>SLOC, Reused</td>
<td></td>
</tr>
<tr>
<td>Capability Score</td>
<td>0.20</td>
</tr>
<tr>
<td>Percentile Rating</td>
<td>38%</td>
</tr>
<tr>
<td>Ctb</td>
<td>5,373</td>
</tr>
</tbody>
</table>

### Productivity

<table>
<thead>
<tr>
<th>Cte Calculation</th>
<th>Development Schedule</th>
</tr>
</thead>
<tbody>
<tr>
<td>RLOC</td>
<td>1.00</td>
</tr>
<tr>
<td>Product</td>
<td></td>
</tr>
<tr>
<td>PVOL</td>
<td>1.00</td>
</tr>
<tr>
<td>RLOC</td>
<td>1.00</td>
</tr>
<tr>
<td>Product</td>
<td></td>
</tr>
<tr>
<td>PVOL</td>
<td>1.00</td>
</tr>
<tr>
<td>RLOC</td>
<td>1.00</td>
</tr>
</tbody>
</table>

### Basic Estimate

- **Development Effort**: 247.5PM
- **Productivity**: 202.0SLOC/PM
- **Requirements Level**: 8.0Percent
- **Requirements Effort**: 19.8PM
- **Integration Level**: 22.0Percent
- **Integration Effort**: 54.4PM
- **Total Effort**: 321.7PM

### Maintenance

- **Upgrades (ACT)**: 10 %/Yr
- **Upgrade Effort**: 24.7PM/Yr
- **Knowledge Retention**: 57.5PM/Yr
- **Support level**: 57.5Pers/Yr
Conclusions

- Effectiveness is a function of 3 attributes: communications, management, and technology.
- Low productivity improvement over the last 3 decades has been primarily driven by technology.
- Communications and management are KEY productivity drivers.
- Estimates for non-classic environments must account for all 3 effectiveness attributes for application to agile development.