The Effectiveness Formula:  
Key to Next Generation Development
Latest Acquisition Shift

- Recent acquisition headlines
  - Air Force One -- $4B Costs too much, should be cancelled
  - Lockheed F35 – Costs too much
  - Trump claims $600M cut from F35 program

- New acquisition world ahead
  - Development cost and productivity improvement are targets
  - “Agile” is latest technology silver bullet
Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- **Individuals and interactions** over processes and tools.
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation.
- **Responding to change** over following a plan.

That is, while we value the items on the right, we value the items on the left more.
Software Productivity Gains

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

Productivity Workshop
Presented at the ICEAA Southern California Chapter Workshop
March 27, 2017
Plato’s Cave

- Perception limited to image on wall
- Traditional mentality
- Reliance on silver bullets
- Resistance to change
- Minimum improvement
- No out-of-box thinking
Process Evolution

- Code and Repair
- Waterfall 2167A (1960s)
- Prototyping
- Incremental
- Spiral Development (1970s)
- Rapid Application Development (RAD)
- Capability Maturity Model (CMM) (1990s)
- CMM Integrated (CMMI)
- Agile
Traditional Capability Rating

- Programming ability
  - Education
  - IQ
  - Problem solving skills
- Efficiency
- Communication ability
- Cooperation

Traditional data does not work for agile development environments
Our job is to escape the cave, look around, then come back and tell others what we have seen . . .

Of course, they won’t believe us.

Dennis L. McKiernan, 1995
Software engineering is:
(1) the systematic design and development of software products 
(2) the management of the software process.
Effectiveness Formula

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

where
- \( E \) = Effectiveness
- \( C \) = Communication skills (0 – 1)
- \( M \) = Management concept awareness (0 – 1)
- \( CS \) = Computer science technical ability (0 – 1)


*Initial Goal: Development environment model*
History Contributions

- Low productivity improvement over the last 3 decades has been primarily driven by technology.
- Productivity/Effectiveness is a function of 3 attributes: communications, management, and technology.
- Communications and management are KEY productivity drivers.
- Effectiveness Formula applies to almost all development environments.
The Ctb Calibration Surprise

- Same application, same organization, same facilities, same people
- Two projects far outside grouping
- Traditional definitions could not explain outliers
- Management effectiveness was the differentiator

Industry Avg = 8635 = highest rating to date
Hawthorne Effect

- 1924 Productivity Experiment
  - *Western Electric Company, Hawthorne, Illinois*
  - *Conducted by Harvard Business School*

- Initially designed to study illumination productivity effects

- Illustrated importance of worker’s attitudes and feelings on performance

- People driven by esteem and self-actualization, not by physiological and safety needs
Lockheed Skunk Works™

- Unofficial name given to Lockheed Advanced Development Projects Unit managed by Kelly Johnson – dates back to the period around WWII.
- Dispenses with both physical and nonphysical walls
- Known for producing the P-80 fighter aircraft prototype (XP-80) in only 137 days.
- Johnson: “We are defined, not by the technologies we create, but by the process in which we create them”
Generic Skunk Works Defined

- A small group of experts who move outside an organization’s mainstream operations in order to develop a new technology or application as quickly as possible without the organization’s bureaucracy or strict process application.
- Creative free thinkers without conventional boundaries
- Physically open workspace environment that encourages intra-team access and communications
- Tools and processes tailored to the project’s requirements.
- Minimize development risk while maintaining the greatest possible agility and creativity.
Cube Farm

- Counterexample to Skunk Works environment
- Violates all rules for a productive environment in terms of both communications and collaboration
- Raises all barriers that block effective communication
- Devolved from the Action Office originally used by the Lockheed Skunk Works
- Most widely used software development environment today.
Management Research Focus

- General management concepts, 1911
  Management philosophy: Plan, Organize, Command, Coordinate, Control
- Hawthorne experiment (1924-1932)
- People impacts, Mayo, 1933
- Lockheed Skunk Works, 1940s
- Theory X/ Theory Y, McGregor, 1960
- CMM, Humphrey, 1989
## Theory X / Theory Y Concept

<table>
<thead>
<tr>
<th>Theory X</th>
<th>Theory Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Work is inherently distasteful to most people.</td>
<td>1. Work is as natural as play, if conditions are favorable.</td>
</tr>
<tr>
<td>2. Most people are not ambitious, have little desire for responsibility, and prefer to be directed.</td>
<td>2. Self-control is often indispensable in achieving organizational goals.</td>
</tr>
<tr>
<td>3. Most people have little capacity for creativity in solving organizational problems.</td>
<td>3. The capacity for creativity in solving organizational problems is widely distributed in the population.</td>
</tr>
<tr>
<td>4. Motivation occurs only at the physiological and safety levels.</td>
<td>4. Motivation occurs at the social, esteem, and self-actualization levels, as well as physiological and security levels.</td>
</tr>
<tr>
<td>5. Management philosophy: Plan, Organize, Command, Coordinate, Control</td>
<td></td>
</tr>
</tbody>
</table>
Relative Project Impact

3. Tools
11. People
17. Systems
64. Management

Source: G. Weinberg, Quality Software Management, Vol. 3
1975 Pair Programming Study

- Real time system executive
- 30,000 Fortran source lines
- 6 software components
- 5 two-person teams plus manager
- Average productivity prior to study = 77 LPPM
- Pilot study productivity = 175 LPPM
- Error rate <0.1% of norm
Effective Team Definition

- Team members:
  1. Operate with **high degree** of **interdependence**, 
  2. Share authority/responsibility for **self-management**, 
  3. Are **accountable** for collective **performance**, 
  4. Work toward **common** goal and shared rewards.

- A team has a strong sense of mutual commitment that creates **synergy**, thus generating performance greater than the sum of performance of its individual members.

  Source: businessdictionary.com
Cross Functional Team Study

- Real Time System
- 57,500 Ada Source Lines
- 1 Software Component
- Cross Functional Team Organization
- Average Productivity Prior to Study = ??
- Pilot Study Productivity = 218 LPPM
- Basic Technology Constant = 8635 (highest recorded in aerospace industry)
Effective Team Environment

- Physically open environment that encourages intra-team access and communication
- Tools and processes adapted to project requirements
- Maintain free flow of information
- Minimize outside interference
- Maximize collaboration and communication
Culture Issues

- Organization Culture
- Comfort Zone
- Confirmation Bias
- K-12 Phenomenon
Organization Culture

- Development organizations have a lot in common
- Technologies
- Physical environment
- Management
  - Hierarchical pyramid
  - Classical 20\textsuperscript{th} century approach
    - \textit{Plan, organize, staff, control, direct}
- Basic Technology Constant?

Presented at the ICEAA Southern California Chapter Workshop
March 27, 2017
Comfort Zone

- Definition: *a place or situation where one feels safe or at ease and without stress.*
- Mental boundary or behavioral state
- Provides an *unfounded* sense of security
- Person tends to stay within the “Zone”
- Stepping outside boundary raises anxiety level and stress response, a danger zone
Confirmation Bias Examples

- Structured programming (1970s)
- GOTO-less programming
- Ada
- Object-oriented development
- Cubicles
- CMMI
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.

K-12 Contributions

- K-12 ingrains inefficiency and waste
- Discourages learning and personal improvement
- Suppresses communication skills
- Workers are inherently lazy and must be closely supervised
- Impedes team formation and behavior
- Ideal conditioning for cubicle behavior
Status Quo Pressure

- Cultural resistance to change is still strong
  - Comfort zone
  - Confirmation bias
  - K-12 phenomenon persists
- Reliance on “Silver bullets”
- Management school support of only *plan, organize, command, coordinate,* and *control* functions
- CMMI (1985)
Relative Project Impact

- Tools: 3
- People: 11
- Systems: 17
- Management: 64

Source: G. Weinberg, Quality Software Management, Vol. 3
Traditional Capability Rating

- Programming ability
- Education
- IQ
- Problem solving skills
- Efficiency
- Communication ability
- Cooperation

MAJOR

MINOR

Traditional data does not work for agile development environments
Effective 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
Effectiveness Formula Redux

\[ E = CM (CS) \]

where

- \( E \) = Effectiveness
- \( CM \) = Communications/Management ability (0 – 1)
- \( CS \) = Software Engineering technical ability (0 – 1)

- Forty five years of validation data !!
- Data does not permit communication – management split
Software Productivity Gains

- Agile DOD
- Structured Analysis
- Structured Design
- Structured Programming
- Ada
- PWB
- OOD
- 3rd Generation Languages
- Process Maturity
### Development Component:

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Cte Calculation</th>
<th>Basic Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complexity, D</td>
<td>12</td>
<td>Development Effort: 586.5PM</td>
</tr>
<tr>
<td>Effective Size</td>
<td></td>
<td>Productivity: 140.3 SLOC/PM</td>
</tr>
<tr>
<td>SLOC, New</td>
<td>80,000</td>
<td>Requirements Level: 8.0 Percent</td>
</tr>
<tr>
<td>SLOC, Modified</td>
<td>2,000</td>
<td>Requirements Effort: 46.3PM</td>
</tr>
<tr>
<td>SLOC, Reused</td>
<td>33,000</td>
<td>Integration Level: 22.0 Percent</td>
</tr>
<tr>
<td>Size, Total</td>
<td>115,000</td>
<td>Integration Effort: 129.0PM</td>
</tr>
<tr>
<td>Size, Effective</td>
<td>82,650</td>
<td>Total Effort: 762.5PM</td>
</tr>
</tbody>
</table>

### Ctb Calculation

<table>
<thead>
<tr>
<th>Environment</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ACAP</td>
<td>1</td>
</tr>
<tr>
<td>AEXP</td>
<td>1.29</td>
</tr>
<tr>
<td>MODP</td>
<td>1.1</td>
</tr>
<tr>
<td>PCAP</td>
<td>1</td>
</tr>
<tr>
<td>RESP</td>
<td>1</td>
</tr>
<tr>
<td>TOOL</td>
<td>1</td>
</tr>
<tr>
<td>TURN</td>
<td>0.87</td>
</tr>
<tr>
<td>Capability Score</td>
<td>0.19</td>
</tr>
<tr>
<td>Percentile Rating</td>
<td>29%</td>
</tr>
<tr>
<td>Ctb</td>
<td>4,953</td>
</tr>
</tbody>
</table>

### Management

<table>
<thead>
<tr>
<th>Management</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ACAP</td>
<td>1</td>
</tr>
<tr>
<td>AEXP</td>
<td>1.29</td>
</tr>
<tr>
<td>MODP</td>
<td>1.1</td>
</tr>
<tr>
<td>PCAP</td>
<td>1</td>
</tr>
<tr>
<td>RESP</td>
<td>1</td>
</tr>
<tr>
<td>TOOL</td>
<td>1</td>
</tr>
<tr>
<td>TURN</td>
<td>0.87</td>
</tr>
<tr>
<td>Capability Score</td>
<td>0.19</td>
</tr>
<tr>
<td>Percentile Rating</td>
<td>29%</td>
</tr>
<tr>
<td>Ctb</td>
<td>4,953</td>
</tr>
<tr>
<td>Templates</td>
<td>3</td>
</tr>
</tbody>
</table>

### Product

<table>
<thead>
<tr>
<th>Product</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ACAP</td>
<td>1</td>
</tr>
<tr>
<td>AEXP</td>
<td>1.29</td>
</tr>
<tr>
<td>MODP</td>
<td>1.1</td>
</tr>
<tr>
<td>PCAP</td>
<td>1</td>
</tr>
<tr>
<td>RESP</td>
<td>1</td>
</tr>
<tr>
<td>TOOL</td>
<td>1</td>
</tr>
<tr>
<td>TURN</td>
<td>0.87</td>
</tr>
<tr>
<td>Capability Score</td>
<td>0.19</td>
</tr>
<tr>
<td>Percentile Rating</td>
<td>29%</td>
</tr>
<tr>
<td>Ctb</td>
<td>4,953</td>
</tr>
<tr>
<td>Templates</td>
<td>3</td>
</tr>
<tr>
<td>Spec</td>
<td>1.12</td>
</tr>
<tr>
<td>Test</td>
<td>1.05</td>
</tr>
<tr>
<td>Timc</td>
<td>1.00</td>
</tr>
<tr>
<td>Tvol</td>
<td>1.00</td>
</tr>
<tr>
<td>Cte</td>
<td>3,402</td>
</tr>
</tbody>
</table>

### Support

<table>
<thead>
<tr>
<th>Support</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PVOL</td>
<td>1.00</td>
</tr>
<tr>
<td>Rloc</td>
<td>1.00</td>
</tr>
<tr>
<td>Environment</td>
<td></td>
</tr>
<tr>
<td>DVOL</td>
<td>1.07</td>
</tr>
<tr>
<td>Rded</td>
<td>1.00</td>
</tr>
<tr>
<td>Management</td>
<td></td>
</tr>
<tr>
<td>Mcls</td>
<td>1.00</td>
</tr>
<tr>
<td>Morg</td>
<td>1.00</td>
</tr>
<tr>
<td>Multi</td>
<td>1.00</td>
</tr>
<tr>
<td>Product</td>
<td></td>
</tr>
<tr>
<td>Disp</td>
<td>1.05</td>
</tr>
<tr>
<td>Host</td>
<td>1.00</td>
</tr>
<tr>
<td>Memc</td>
<td>1.00</td>
</tr>
<tr>
<td>Qual</td>
<td>1.03</td>
</tr>
<tr>
<td>Rtim</td>
<td>1.00</td>
</tr>
<tr>
<td>Rvol</td>
<td>1.07</td>
</tr>
<tr>
<td>Template</td>
<td>1.00</td>
</tr>
<tr>
<td>Secr</td>
<td>1.00</td>
</tr>
<tr>
<td>Spec</td>
<td>1.12</td>
</tr>
<tr>
<td>Test</td>
<td>1.05</td>
</tr>
<tr>
<td>Timc</td>
<td>1.00</td>
</tr>
<tr>
<td>Tvol</td>
<td>1.00</td>
</tr>
<tr>
<td>Cte</td>
<td>3,402</td>
</tr>
</tbody>
</table>

### Development Schedule

| Development Schedule       | 26.2 MO             |
| Requirements Schedule      | 7.4 MO              |

### Requirements Level

| Requirements Effort        | 46.3 PM             |
| Integration Level          | 22.0 Percent        |
| Integration Effort         | 129.0PM             |
| Total Effort               | 762.5PM             |

### Peak development staff

| Peak development staff     | 34.5 Pers           |
| Peak programming staff     | 18.5 Pers           |
| Max staffing rate          | 26.2 Pers/Yr        |

### Maintenance

<table>
<thead>
<tr>
<th>Maintenance</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Upgrades (ACT)</td>
<td>10 %/Yr</td>
</tr>
<tr>
<td>Upgrade Effort</td>
<td>87.2 PM/Yr</td>
</tr>
<tr>
<td>Knowledge Retention</td>
<td>88.2 PM/Yr</td>
</tr>
<tr>
<td>Support level</td>
<td>88.2 Pers/ Yr</td>
</tr>
</tbody>
</table>
## Effectiveness Value Rating

<table>
<thead>
<tr>
<th>Eff Value</th>
<th>Percentile of Industry</th>
<th>Ctb Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>5.0</td>
<td></td>
</tr>
<tr>
<td>0.17</td>
<td>22.4</td>
<td>4570</td>
</tr>
<tr>
<td>0.2</td>
<td>35.2</td>
<td></td>
</tr>
<tr>
<td>0.23</td>
<td>50.0</td>
<td>5707</td>
</tr>
<tr>
<td>0.28</td>
<td>57.9</td>
<td>6735</td>
</tr>
<tr>
<td>0.3</td>
<td>61.0</td>
<td>7000</td>
</tr>
<tr>
<td>0.31</td>
<td>62.5</td>
<td>7350</td>
</tr>
<tr>
<td>0.4</td>
<td>75.2</td>
<td>8630</td>
</tr>
<tr>
<td>0.45</td>
<td>81.0</td>
<td>9587</td>
</tr>
<tr>
<td>0.5</td>
<td>86.0</td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>99.9</td>
<td></td>
</tr>
</tbody>
</table>
The Next Generation?

- The next generation of software development is already here
  - First organization experiments in “modern management in 1910s (Hawthorne Effect)
  - First evidence of an organization shift is traceable to the mid 1940s (Lockheed Skunk Works®)

- Symptoms of next generation shift
  - Agile Manifesto (2001)
  - Herman Miller Co. introduces new office evolution including movable walls and open workplace forms (2013)

- New stress on acquisition cost and productivity
Caverns of Socrates

Our job is to escape the cave, look around, then come back and tell others what we have seen . . .

Of course, they won’t believe us.

Dennis L. McKiernan, 1995
Productivity?

Not my problem!