Agile Benchmarks: What Can You Conclude?
Purpose of Talk

- **Aim** – summarize our agile benchmarks and studies
- **Agenda**
  - Setting the stage
  - Benchmarking results - 2016
  - Agile study findings
    - Agile adoption
    - Agile and the CMMI
    - Agile scaling
  - Building an agile culture
  - Summary and conclusions
Setting the Stage

- Polled over 500 organizations to determine state of agile adoption worldwide
  - Response rate of 60% with a little over 300 organizations responding
  - 18 countries responded including:

<table>
<thead>
<tr>
<th>Australia</th>
<th>Italy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brazil</td>
<td>Japan</td>
</tr>
<tr>
<td>Canada</td>
<td>Mexico</td>
</tr>
<tr>
<td>China</td>
<td>Netherlands</td>
</tr>
<tr>
<td>Finland</td>
<td>Russia</td>
</tr>
<tr>
<td>France</td>
<td>Sweden</td>
</tr>
<tr>
<td>Germany</td>
<td>UK</td>
</tr>
<tr>
<td>India</td>
<td>USA</td>
</tr>
<tr>
<td>Israel</td>
<td>Viet Nam</td>
</tr>
</tbody>
</table>
What Are Agile Methods?

• Many definitions
  – We define as adhering to the concepts and principles of the Agile Manifesto

• Agile Manifesto
  – Individuals and interactions over processes and tool
  – Working software over comprehensive documentation
  – Customer collaboration over contract negotiations
  – Responding to change over following a plan

• Twelve principles of agile software
  – Highest priority is to satisfy customer via early and continuous delivery of valuable software
  – Welcome changing requirements, even late in development
  – Deliver working software frequently
  – Build projects around motivated individuals
  – Working software is the primary measure of progress
  – More
• There are many types of agile software development life cycle: Agile Unified Process, Extreme Programming, Scrum, hybrids and others.
• Many firms in defense field are using agile in conjunction with CMMI and other process frameworks to tap its benefits at enterprise level.
Many Different Agile Methods

- **Scrum**
  - A disciplined lightweight method where focus is on the frequent delivery of working code
    - Lightweight methods are loosely structured and more flexible than those like MIL-STD-2167A which is considered a heavyweight
    - Customer works with team to identify and prioritize functionality and address issues in real-time
- **Lean and Kanban Software Development**
  - Another lightweight set of methods that is often used in conjunction with agile methods that borrows from manufacturing and focuses on delivering value and eliminating waste
- **Extreme Programming (XP)**
  - A collection of lightweight practices that are focused on developing working code using pair programming and other agile practices
- **Many other methods like Crystal, DaD, DSDM, SAFe, etc.**
  - Including semi-agile and hybrid approaches
  - Many of these methods, including Scrum of Scrum, were developed to address agile-at-scale issues
What Does It Mean to Be Agile?

• Most in the field agree that to be agile:
  – Your approach must adhere to the Manifesto and its guiding principles
  – You view development as a discovery process
  – You develop iteratively placing emphasis on delivering product each iteration
  – You try to minimize waste and overhead
  – Your focus on mostly small projects, i.e., ≤ 50 people
  – You emphasize value

• Many defense firms use agile practices, but are not agile
  – They scale agile to address larger projects by adopting traditional as well as agile concepts (hybrid approach)
  – They continue to view development as a gated process with emphasis on providing documentation
  – Much of what they do is influenced by customer desires to manage cost and schedule performance
  – Emphasis is placed on performance, not value
Why All The Fuss?

- Agile advocates suggest that their way is a better way
  - Software developers love it
  - Customers do learn to like it
- While there is a lot of hype about agile, it has promise
  - Unsubstantiated claims abound about benefits
  - Numbers cited by reputable sources show advantages
  - Based on benefits, many firms have moved to agile use
  - Agile for IT is mandated in UK and USA governments
- Besides the numbers, there is a groundswell for change
  - Too many failures
  - Too much bureaucracy
  - Too little attention paid to what makes software sense
- Agile comes with good and bad
  - Many firms have tapped the good for their benefit
  - There are many blueprints for success
  - However, there are many issues that have to be worked
Let’s Look at the Numbers

<table>
<thead>
<tr>
<th><strong>We looked at 2,000 projects from 100 firms</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• 8 applications domains</td>
</tr>
<tr>
<td>• 1,000 agile projects</td>
</tr>
<tr>
<td>• 1,000 traditional projects</td>
</tr>
<tr>
<td>• All data is less than 10 years old</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Productivity is better</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Many critics argue results due to Hawthorne effect – they always do when positive</td>
</tr>
<tr>
<td>• However, the data supports the conclusion that agile performance is between 10 to 35% better than traditional norms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Costs are cheaper</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Again many caveats</td>
</tr>
<tr>
<td>• Data supports agile is 10 to 40% better after transition</td>
</tr>
<tr>
<td>• Several databases in addition to ours support this conclusion</td>
</tr>
<tr>
<td>• Some argue that advocates put too much attention on test and not enough on engineering quality into the product</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Quality gets better with age</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Some controversy</td>
</tr>
<tr>
<td>• Data supports agile is 10 to 40% better after transition</td>
</tr>
<tr>
<td>• Several databases in addition to ours support this conclusion</td>
</tr>
<tr>
<td>• Some argue that advocates put too much attention on test and not enough on engineering quality into the product</td>
</tr>
</tbody>
</table>
## Demographics of Use

<table>
<thead>
<tr>
<th>Applications Domain</th>
<th>No. of Firms</th>
<th>No. of Completed Software Projects</th>
<th>Average Experience with Adopted Agile Method</th>
<th>Example Products Generated by Firm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 - 2 Years</td>
<td>3 to 5 Years</td>
</tr>
<tr>
<td>Automation</td>
<td>10</td>
<td>100</td>
<td>25%</td>
<td>30%</td>
</tr>
<tr>
<td>Financial</td>
<td>10</td>
<td>100</td>
<td>20%</td>
<td>30%</td>
</tr>
<tr>
<td>Defense</td>
<td>20</td>
<td>250</td>
<td>35%</td>
<td>45%</td>
</tr>
<tr>
<td>Info Technology</td>
<td>15</td>
<td>150</td>
<td>15%</td>
<td>30%</td>
</tr>
<tr>
<td>Medical</td>
<td>5</td>
<td>100</td>
<td>25%</td>
<td>40%</td>
</tr>
<tr>
<td>Software Tools</td>
<td>10</td>
<td>100</td>
<td>15%</td>
<td>30%</td>
</tr>
<tr>
<td>Telecom.</td>
<td>20</td>
<td>100</td>
<td>20%</td>
<td>30%</td>
</tr>
<tr>
<td>Web Business</td>
<td>10</td>
<td>100</td>
<td>30%</td>
<td>30%</td>
</tr>
<tr>
<td><strong>TOTALS</strong></td>
<td><strong>100</strong></td>
<td><strong>1,000</strong></td>
<td><strong>22%</strong></td>
<td><strong>32%</strong></td>
</tr>
</tbody>
</table>

About 3/4 of projects completed with teams with over 3 years of experience
## Method Use by Application Domain

<table>
<thead>
<tr>
<th>Applications Domain</th>
<th>Crystal</th>
<th>DSDM</th>
<th>Lean/Kanban</th>
<th>Scrum</th>
<th>SAFe</th>
<th>Hybrid</th>
<th>Methods Used</th>
<th>% of Total Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automation</td>
<td>1</td>
<td>15</td>
<td>33</td>
<td>2</td>
<td>10</td>
<td>61</td>
<td>8%</td>
<td></td>
</tr>
<tr>
<td>Financial</td>
<td>4</td>
<td>4</td>
<td>10</td>
<td>47</td>
<td>18</td>
<td>14</td>
<td>97</td>
<td>12%</td>
</tr>
<tr>
<td>Defense</td>
<td>4</td>
<td>8</td>
<td>32</td>
<td>6</td>
<td>77</td>
<td>127</td>
<td>17%</td>
<td></td>
</tr>
<tr>
<td>Info Technology</td>
<td>5</td>
<td>5</td>
<td>15</td>
<td>42</td>
<td>32</td>
<td>20</td>
<td>119</td>
<td>16%</td>
</tr>
<tr>
<td>Medical</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>23</td>
<td>3</td>
<td>11</td>
<td>48</td>
<td>6%</td>
</tr>
<tr>
<td>Software Tools</td>
<td>3</td>
<td>5</td>
<td>12</td>
<td>45</td>
<td>18</td>
<td>15</td>
<td>98</td>
<td>13%</td>
</tr>
<tr>
<td>Telecom</td>
<td>1</td>
<td>4</td>
<td>21</td>
<td>64</td>
<td>9</td>
<td>32</td>
<td>131</td>
<td>17%</td>
</tr>
<tr>
<td>Web Business</td>
<td>2</td>
<td>2</td>
<td>14</td>
<td>34</td>
<td>2</td>
<td>32</td>
<td>86</td>
<td>11%</td>
</tr>
<tr>
<td>TOTALS</td>
<td>17</td>
<td>29</td>
<td>100</td>
<td>320</td>
<td>90</td>
<td>211</td>
<td>767</td>
<td>100%</td>
</tr>
</tbody>
</table>

Popularity of methods is a function of project size and scaling – Scrum for small-medium, Hybrid for large.
Agile Productivity

• Agile productivity as measured in eSLOC/SM
  – Major difficulty is determining size
    • Most agile advocates use user stories or story points as their size measure
    • Must convert and then normalize to a standard measure which you can relate to for comparison purposes
    • Must take different types of work into account (new, modified, reused, generated and carried forward [software used as-is from one release to another])
  – Effort must be related to the work done during agile life cycle which differs from traditional allocations
Agile Vs. Traditional Productivity

- Agile productivity seems higher by factors of 10 to 35% depending on domain
- True even after adoption has taken place and buzz dissipates
- Time to market is improved due to frequent deliveries
- Average gain during past three years averaged 10 to 15% annually

Agile vs. Traditional Software Productivity Trends

Agile average = 375 eSLOC/SM vs. Traditional average = 335 eSLOC/SM
# Productivity Comparisons – Agile versus Traditional

<table>
<thead>
<tr>
<th>Applications Domain</th>
<th>No. of Firms</th>
<th>No. of Agile Projects</th>
<th>Agile Productivity (eSLOC/SM)</th>
<th>No. of Non-Agile Projects</th>
<th>Non-Agile Productivity (eSLOC/SM)</th>
<th>% Gain/Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automation</td>
<td>10</td>
<td>100</td>
<td>365</td>
<td>100</td>
<td>310</td>
<td>18</td>
</tr>
<tr>
<td>Financial</td>
<td>10</td>
<td>100</td>
<td>12.5</td>
<td>100</td>
<td>289</td>
<td>12.5</td>
</tr>
<tr>
<td>Defense (weapons only)</td>
<td>20</td>
<td>250</td>
<td>15</td>
<td>250</td>
<td>289</td>
<td>15</td>
</tr>
<tr>
<td>Info Technology</td>
<td>15</td>
<td>150</td>
<td>11</td>
<td>150</td>
<td>455</td>
<td>11</td>
</tr>
<tr>
<td>Medical</td>
<td>5</td>
<td>100</td>
<td>9</td>
<td>100</td>
<td>455</td>
<td>9</td>
</tr>
<tr>
<td>Software Tools</td>
<td>10</td>
<td>100</td>
<td>15</td>
<td>100</td>
<td>455</td>
<td>15</td>
</tr>
<tr>
<td>Telecommunications</td>
<td>20</td>
<td>100</td>
<td>10</td>
<td>100</td>
<td>455</td>
<td>10</td>
</tr>
<tr>
<td>Web Business</td>
<td>10</td>
<td>100</td>
<td>12</td>
<td>100</td>
<td>455</td>
<td>12</td>
</tr>
<tr>
<td><strong>TOTALS</strong></td>
<td><strong>100</strong></td>
<td><strong>1,000</strong></td>
<td><strong>438</strong></td>
<td><strong>1,000</strong></td>
<td><strong>390</strong></td>
<td><strong>12.5</strong></td>
</tr>
</tbody>
</table>

Computed using method of weighted averages
### Defense Productivity Comparisons – Agile versus Traditional

<table>
<thead>
<tr>
<th>Applications Domain</th>
<th>No. of Firms</th>
<th>No. of Agile Projects</th>
<th>Agile Productivity (eSLOC/SM)</th>
<th>No. of Non-Agile Projects</th>
<th>Non-Agile Productivity (eSLOC/SM)</th>
<th>% Gain/Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avionics</td>
<td>3</td>
<td>10</td>
<td></td>
<td>75</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>Ground</td>
<td>7</td>
<td>50</td>
<td>287</td>
<td>75</td>
<td>265</td>
<td>11</td>
</tr>
<tr>
<td>IT</td>
<td>10</td>
<td>50</td>
<td>489</td>
<td>100</td>
<td>435</td>
<td>12</td>
</tr>
<tr>
<td>Medical</td>
<td>15</td>
<td>50</td>
<td></td>
<td>50</td>
<td></td>
<td>13</td>
</tr>
<tr>
<td>Missile</td>
<td>3</td>
<td>10</td>
<td></td>
<td>50</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>Shipboard</td>
<td>10</td>
<td>50</td>
<td></td>
<td>75</td>
<td></td>
<td>7</td>
</tr>
<tr>
<td>Space</td>
<td>7</td>
<td>15</td>
<td>172</td>
<td>50</td>
<td>166</td>
<td>3</td>
</tr>
<tr>
<td>Trainers</td>
<td>5</td>
<td>15</td>
<td></td>
<td>25</td>
<td></td>
<td>11</td>
</tr>
<tr>
<td><strong>TOTALS</strong></td>
<td><strong>20</strong>*</td>
<td><strong>250</strong></td>
<td><strong>332</strong></td>
<td><strong>500</strong></td>
<td><strong>289</strong></td>
<td><strong>9.3</strong></td>
</tr>
</tbody>
</table>

* Some firms active in more than one domain

Computed using method of weighted averages
Reifer - Agile Cost

• Agile cost measured in $/eSLOC
  – Easily converted using factors we developed to $/story or story point, $/UML point, $/function/feature point, etc.
  – Measure sensitive to labor rates including where and how they were calculated
    • Including both out- and in-sourcing work
  – The following major costs are not included as they were funded separately
    • Process reengineering (especially those related to processes that support agile; i.e., CM/DM, metrics and SQA)
    • Facilities costs (agile tools, war room, etc.)
    • Change management (education, etc.)
Agile methods are between 10 and 25% cheaper even when labor rates are normalized across domains.

As noted, many of the costs related to managing the transition to agile are not accounted for.

Average reduction during past three years averaged 8 to 12% annually.

Seems to be many issues related to agile supplier management.

**Agile average = $35/eSLOC vs. Traditional average = $45/eSLOC**
## Cost Comparisons – Agile versus Traditional

<table>
<thead>
<tr>
<th>Applications Domain</th>
<th>No. of Firms</th>
<th>No. of Agile Projects</th>
<th>Agile Cost ($/eSLOC)</th>
<th>No. of Non-Agile Projects</th>
<th>Non-Agile Cost ($/eSLOC)</th>
<th>% Reduction Year*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automation</td>
<td>10</td>
<td>100</td>
<td>27</td>
<td>100</td>
<td>30</td>
<td>11</td>
</tr>
<tr>
<td>Financial</td>
<td>10</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>102</td>
<td>9</td>
</tr>
<tr>
<td>Defense (weapons only)</td>
<td>20</td>
<td>250</td>
<td>88</td>
<td>250</td>
<td>102</td>
<td>14</td>
</tr>
<tr>
<td>Info Technology</td>
<td>15</td>
<td>150</td>
<td>20</td>
<td>150</td>
<td>22</td>
<td>12</td>
</tr>
<tr>
<td>Medical</td>
<td>5</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>10</td>
</tr>
<tr>
<td>Software Tools</td>
<td>10</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>16</td>
</tr>
<tr>
<td>Telecommunications</td>
<td>20</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>8</td>
</tr>
<tr>
<td>Web Business</td>
<td>10</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>11</td>
</tr>
<tr>
<td>TOTALS</td>
<td>100</td>
<td>1,000</td>
<td>$37.70</td>
<td>1,000</td>
<td>$42.80</td>
<td>11.9</td>
</tr>
</tbody>
</table>

Computed using method of weighted averages
Agile Time-to-Market

Percent of Projects Who Deliver On-Time

- Agile methods realize hard-deadlines 80 to 90% of the time versus a 40 to 60% average for plan-driven projects
- Typical goodness-of-fit is 80 to 90% (percent features delivered vs. that obliged)
- Traditional projects that deliver 100% functionality often exceed promised deadlines and/or budgets by 40 to 50%
Reifer - Agile Quality

• Quality measured in defects/KeSLOC computed post-delivery
  – Again, selected so we could compare against traditional project performance
  – Different measures are used to compute quality during development
  – Quality measures should include more of the “ilities,” but hard to quantify softer factors like “maintainability” and “fitness of use”

• Major influence factors include:
  – Degree to which release was tested (latent defects) – unit as well as integration testing
  – When during the life cycle you decide to measure it (development or maintenance)

• Other measures used as well to give a more well-rounded view of quality
Agile average = 2.4 defects/KeSLOC vs. Traditional average = 3.5 defects/KeSLOC

- At first, agile quality is not as good as that computed for traditional projects
- Breakeven in return is between 2 to 3 years
- Root cause seems to be that too much attention placed on testing and not enough on engineering quality into the product
### Quality Comparisons – Agile versus Traditional

<table>
<thead>
<tr>
<th>Applications Domain</th>
<th>No. of Firms</th>
<th>No. of Agile Projects</th>
<th>Agile Quality (d/KeSLOC)</th>
<th>No. of Traditional Projects</th>
<th>Traditional Quality* (d/KeSLOC)</th>
<th>% Reduction Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automation</td>
<td>10</td>
<td>100</td>
<td>2.5</td>
<td>100</td>
<td>2.5</td>
<td>0</td>
</tr>
<tr>
<td>Financial</td>
<td>10</td>
<td>100</td>
<td></td>
<td>100</td>
<td></td>
<td>-12.5</td>
</tr>
<tr>
<td>Defense (weapons)</td>
<td>20</td>
<td>250</td>
<td>0.85</td>
<td>250</td>
<td>0.85</td>
<td>0</td>
</tr>
<tr>
<td>Info Technology</td>
<td>15</td>
<td>150</td>
<td>3.5</td>
<td>150</td>
<td>4</td>
<td>-14</td>
</tr>
<tr>
<td>Medical</td>
<td>5</td>
<td>100</td>
<td></td>
<td>100</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Software Tools</td>
<td>10</td>
<td>100</td>
<td></td>
<td>100</td>
<td></td>
<td>-20</td>
</tr>
<tr>
<td>Telecom</td>
<td>20</td>
<td>100</td>
<td></td>
<td>100</td>
<td></td>
<td>-6</td>
</tr>
<tr>
<td>Web Business</td>
<td>10</td>
<td>100</td>
<td></td>
<td>100</td>
<td></td>
<td>-7</td>
</tr>
<tr>
<td><strong>TOTALS</strong></td>
<td>100</td>
<td>1,000</td>
<td>2.4</td>
<td>1,000</td>
<td>3.5</td>
<td>-6</td>
</tr>
</tbody>
</table>

Computed using weighted average number of defects
Quality Survey - Defect Density

- Confirms earlier findings
  - Reifer, Capers Jones and ISBSG
- Results favorable to agile
  - Agile averages about 22% better
  - Defense was 33% better
  - Telecom and mobile the same
- Findings based on 662 projects

Figure 1 - Reliability by Domain Measured by Defect Density (Defects/1,000 Unadjusted Function Points [UFP])
Fitness using traditional methods was better across all domains:
- 89% vs. 75% for defense
- Averages about 83% vs. 70%

Not surprising as agile puts priority on time-to-market.

However, all priority stories were delivered.
Total number of defects found exceed those predicted using software reliability models (IEEE recommended)
  - Not surprising when you are performing testing continuously over many sprints

Defect rates and densities are similar to plan-driven developments

Distribution of defects changes as well as does their criticality
  - Fewer requirements and design defects make it to the final release
  - Fewer critical defects are fielded because criticality along with value are used to prioritize agile backlogs
Many agile proponents unfortunately equate quality with testing
   – Their focus is placed on test-first concepts and test automation
   – Use defect backlogs to track open problems by priority and date found

Many in agile community view independent quality assurance groups as wasteful and unnecessary efforts
   – Argue that you can’t inspect quality into products

Forward-leaning agile firms engineer quality into products
   – Redefine quality’s role to being teachers and refocus their goals accordingly
   – Quality personnel work as members of teams rather than in an independent capability

Emphasis of many agile proponents is placed on test because they need to frequently revalidate their releases
   – Regression test baselines created and delivered with products
   – Automated test tools used in conjunction with version control

**Bottom line** - quality assurance as an independent organization is eliminated and its role is absorbed by the engineering team
### Numbers - What Do Others Say?

<table>
<thead>
<tr>
<th>Source</th>
<th>Details</th>
</tr>
</thead>
</table>
| **ISBSG – Australia** | - Data from 10+ nations  
- Mostly IT projects analyzed  
  - Mostly small to medium jobs  
  - Some defense  
- Data **“fully supports”** our findings  
- Not enough data to fully assess quality |
| **Capers Jones data** | - Data **“fully supports”** our findings including that related to quality  
- Shows that some practices like pair programming should be avoided |
| **Agile community data** | - Mostly soft data taken from ‘feel good” surveys  
- Sources include Agile Alliance, Rally Software, Version One and others  
- Shows much higher gains  
- Data has not been validated |
| **QSM Data** | - Shows agile is faster, cheaper and better quality |
| **David Rico** | - Summarizes other’s studies  
- Also show agile is faster, cheaper and better quality |
### Demographics
- 54 recently completed Agile projects/total not disclosed
- 12 different companies
- 87% business, 7% scientific applications, 6% system software
- Team size clustered in 5-10 and 20-50 ranges
- Median size 42.9k lines of code
- Median effort 47 staff months
- Median staff 7.5
- Median duration 6.1 months
- Principally new development

### Findings
- Agile projects outperform conventional development in productivity, cost and quality
  - 11% higher productivity
  - 30% less time
  - 38% fewer defects
- Agile projects complete more quickly than traditional projects but with about the same amount of effort
  - Staffing levels are higher, but overall staff hours expended are less

*Beyond the Hype, 2011.*
More Interesting Findings

• Agile is here to stay
  – It is the primary way software is being developed across the board worldwide

• Scrum is the primary method used for small to medium software projects
  – Fundamental principles followed

• Process frameworks like the CMMI are on the decline commercially
  – High cost of SCAMPI-A and marginal value turn firms off
  – Defense firms view CMMI as a requirement

• Larger projects use a scaled, hybrid approach
  – Large organizations try to harmonize agile with their existing processes
  – However, 8 to 42% of those trying to scale agile methods abandon them or revert back to traditional approaches after 7 years
Worldwide Agile Methodology Usage (Number of Organizations)

Size of Organizations

<table>
<thead>
<tr>
<th>Size</th>
<th>No. Employees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>&lt; 500</td>
</tr>
<tr>
<td>Medium</td>
<td>500 to 5,000</td>
</tr>
<tr>
<td>Large</td>
<td>5,000 to 50,000</td>
</tr>
<tr>
<td>Very Large</td>
<td>&gt; 50,000</td>
</tr>
</tbody>
</table>

No. of Organizations/Transition Stage

<table>
<thead>
<tr>
<th>Location</th>
<th>EA</th>
<th>EM</th>
<th>LM</th>
<th>Laggards</th>
</tr>
</thead>
<tbody>
<tr>
<td>America (125)</td>
<td>13</td>
<td>67</td>
<td>28</td>
<td>17</td>
</tr>
<tr>
<td>Asia (64)</td>
<td>8</td>
<td>22</td>
<td>20</td>
<td>14</td>
</tr>
<tr>
<td>Australia (30)</td>
<td>3</td>
<td>10</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>Europe (101)</td>
<td>7</td>
<td>53</td>
<td>26</td>
<td>15</td>
</tr>
</tbody>
</table>
Scrum is the Leader of the Pack

Agile Method Usage by Size of Project

Dad – Distributed Agile Delivery
SAFe – Scaled Agile Framework

Agile Method Usage by Number of Projects

<table>
<thead>
<tr>
<th>Location</th>
<th>Dad</th>
<th>SAFe</th>
<th>Scrum</th>
<th>Hybrid</th>
<th>Hybrid/Lean</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>America (125)</td>
<td>9</td>
<td>11</td>
<td>41</td>
<td>33</td>
<td>9</td>
<td>22</td>
</tr>
<tr>
<td>Asia (64)</td>
<td>1</td>
<td>3</td>
<td>33</td>
<td>4</td>
<td>4</td>
<td>19</td>
</tr>
<tr>
<td>Australia (30)</td>
<td>2</td>
<td>3</td>
<td>13</td>
<td>5</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>Europe (101)</td>
<td>0</td>
<td>8</td>
<td>38</td>
<td>6</td>
<td>4</td>
<td>25</td>
</tr>
</tbody>
</table>

September 2016 © 2016, Reifer Consultants LLC
Agile and the CMMI®

Percentage Agile as a % of those rated via CMMI®

<table>
<thead>
<tr>
<th>Year</th>
<th>2008</th>
<th>2009</th>
<th>2010</th>
<th>2011</th>
<th>2012</th>
<th>2013</th>
<th>2014</th>
</tr>
</thead>
<tbody>
<tr>
<td>%</td>
<td>10</td>
<td>15</td>
<td>20</td>
<td>15</td>
<td>10</td>
<td>15</td>
<td>20</td>
</tr>
</tbody>
</table>

Process Maturity by SDLC & Sector

### Number of Organizations Rated for SDLC

<table>
<thead>
<tr>
<th>Sector</th>
<th>SDLC</th>
<th>CMMI</th>
<th>ISO</th>
<th>OMG BPMM</th>
<th>Not Reporting</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Commercial</strong></td>
<td>Traditional</td>
<td>3</td>
<td>16</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Agile</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Defense</strong></td>
<td>Traditional</td>
<td>15</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Agile</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td>29</td>
<td>52</td>
<td>3</td>
<td>16</td>
</tr>
</tbody>
</table>

© 2016, Reifer Consultants LLC
Agile Challenge – Maintaining a Single Engineering Process

- At the enterprise level, many firms have invested in a single engineering process
  - Commercial – Adobe, Fidelity, etc.
  - Defense – all of the big boys say they do it
- For Information Technology (IT) groups, development is relatively easy
  - Lots of models exist and help is available
- For system developers, it is harder because all elements have to be agile
  - Many form an IPT chaired by process group
  - Try to reach consensus on agile approach
  - Systems and hardware engineering must buy-in
  - Many retain their old ways and don’t realize the full benefit from the change
  - Many give up and revert back to their old ways
Process Harmonization

• At first look, the process frameworks like the CMMI® seem at conflict with agile – too much overhead and bureaucracy
  – Many firms have adapted their processes to accommodate agile by mapping practices to their existing frameworks

• Appraisal approaches like SCAMPI-A are expensive and lead appraisers seem to be more comfortable with embracing a business as usual attitude
  – As a result, most of the commercial firms embracing CMMI and agile have moved to self-assessments or via SCAMPI-B appraisals as alternatives

• Because of these issues, many of the firms surveyed have abandoned their process improvement efforts
  – About 60% of commercial and less than 10% of defense firms have moved away from use of process frameworks
  – However, those in regulated industries have not because of governance requirements

• Many firms kept their process groups - they lead the charge to agile
  – Put them to use in managing transformations and harmonizing processes
  – Costs to transition are most often absorbed as an overhead expense
Final Agile Challenge - DevOps

- **Definition** – the culture and environment established to facilitate fast and reliable delivery of applications during software development

- **Fundamentals**
  - Spans the entire delivery pipeline including release management, demos, deployment sequencing and related support
  - Typically performed by a group other than development (separate team)
  - Focuses placed on collaboration of teams involved in development (quality assurance, testing, operations, distribution, project and product management, etc.)
  - Big emphasis placed on automation – especially testing
  - Handles the harmonization and cultural change tasks

- **Challenge**
  - How do you budget and pay for this activity which is often thought of as a software maintenance activity
  - How do you address the increased expenses for automation
Defense Insights for Agile

• For defense organizations, agile should be scaled and used in combination with a Level 3+ process
  – Requires frequent milestones and deliverables whose definition may be looser than you may be comfortable with
  – Requires CDRLs to be tailored and requirements waived

• Reporting requirements should be adapted so that they make sense with agile
  – Look at new metrics and different ways of reporting progress (burn-down lists and rate of progress charts) and EVM (work packages via standard WBS may not make sense)

• A demo-based iterative process should be used to engage stakeholder and showoff the progress being made
  – Show me instead of formal reviews and dog-and-pony shows

• Budgets need to be adjusted to accommodate agile challenges
Summary - Beware the Hype

• Software is primarily being developed using agile methods throughout the world
  – Scrum is the primary approach for small to medium projects
  – Scaled, hybrid approaches are being used for agile-at-scale jobs

• The agile pluses are:
  – Higher productivity and quality
  – Lower costs and quicker development
  – Lots of soft factors that lead to higher morale and motivation → recruitment & retention

• The negatives
  – Scaling, contracting, harmonization, risk management and maintenance issues

Want to Buy a Used Car?
Contact Information

• For more information, contact:

Donald J. Reifer, President
Reifer Consultants LLC
14820 N Dragons Breath Ln
Prescott, AZ 86305-5644
Phone: (310) 922-7043
Web: www.reifer.com
Email: don@reifer.com

NELO personnel have access to these documents on the servers

Our other reports which may be of interest to you include:
1. Reifer Cost, Productivity and Quality Benchmarks
2. Quantitative Analysis of Agile Methods
3. Agile Software Quality: A Quantitative Assessment
4. Agile Metrics and Measurement
5. Agile Estimation
6. Agile Visibility and Control
7. Agile Rework, Waste Reduction and Technical Debt
8. Agile Introduction: Are You a Laggard?
Backup
Moore’s Technology Introduction Model – “Crossing the Chasm”

The Question – How Widely Is Agile Being Used Operationally?

- The chasm portrays the difficulty in getting technology adopted by the majority of the organization
- Our data shows that agile methods have crossed the chasm and are being used widely throughout organizations
Lots of Agile Resources Available

• **SEI**
  – This group and the info it provides
  – Reports focused on government issues like acquisition and scaling

• **Agile Alliance**
  – Conferences
  – Reports
  – Access to the experts
  – Focus is commercial, not government

• **Contractors**
  – All have agile, lean and process improvement initiatives

• **Vendors**
  – Good source of info

• **For more on numbers**
  – All the cost modeling firms have agile knowledge bases
  – ISBSG provides benchmarking data
Reifer Benchmarks are Available

• Sold on subscription basis
  – Deep discounts for Government Shops and FFRDCs
• Subscription includes:
  – Software Productivity, Cost and Quality Reports (issued twice annually)
  – Copies of our reports and working papers
  – Newsletter
  – Access to experts to answer relevant questions