Agile Introduction: Are You A Laggard?

Reifer Consultants LLC
14820 N Dragons Breath Ln
Prescott, AZ 86305-5644
Phone: (928)237-9060
Web: www.reifer.com
Purpose of Talk

- **Aim** – summarize the results of a recent survey documenting the state of agile adoption worldwide

- **Agenda**
  - Setting the stage
  - Moore’s model – crossing the chasm
  - Survey results
  - Playing catch-up
  - Building an agile engineering culture
  - Summary and conclusions
Author Background

- 40+ years of software government, industry and academia
- Specialist in estimation, metrics and management
- Co-developer of COCOMO cost model
- Developing and delivering benchmarks for over two decades
- Working with agile methods since 2002
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:

| Australia | Italy |
| Brazil | Japan |
| Canada | Mexico |
| China | Netherlands |
| Finland | Russia |
| France | Sweden |
| Germany | UK |
| India | USA |
| Israel | Viet Nam |
What Are Agile Methods?

• 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
  – Business people and developers must work together daily
  – Build projects around motivated individuals
  – The most efficient way of conveying information is via face to face conversation
  – Working software is the primary measure of progress
  – The best architectures, requirements and designs emerge from self-organizing teams
  – Four 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
  – 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 is focused on developing working code using pair programming and other tenants of agile manifesto

• **Many other methods like Crystal, DaD, DSDM, etc. including semi-agile and hybrid approaches**
  – Many of these methods, including Scrum of Scrum, were developed to address agile-at-scale issues
Why All The Fuss?

- Agile advocates suggest there way is a better way
  - Software developers love it
  - Customers learn to like it
- There is a lot of hype about agile methods
  - Unsubstantiated claims abound about benefits
  - Numbers cited by reputable sources show promise
  - Based on results, 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
What Does the Survey Say?

- Agile is here to stay
  - It is the primary way software is being developed worldwide
- Scrum is the primary method used for small to medium software projects
  - Fundamental principles followed, some practices omitted
- Larger projects use some hybrid approach
  - Large organizations try to harmonize agile with their existing processes
  - Other engineering and some manufacturing disciplines moving to agile approaches
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.
Worldwide Agile Methodology Usage (Number of Organizations)

<table>
<thead>
<tr>
<th>Size</th>
<th>Description</th>
<th>Location</th>
<th>EA</th>
<th>EM</th>
<th>LM</th>
<th>Laggards</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>≤ 20 KSLOC</td>
<td>America (125)</td>
<td>13</td>
<td>67</td>
<td>28</td>
<td>17</td>
</tr>
<tr>
<td>Medium</td>
<td>20K &lt; x ≤ 150K</td>
<td>Asia (64)</td>
<td>8</td>
<td>22</td>
<td>20</td>
<td>14</td>
</tr>
<tr>
<td>Large</td>
<td>150K &lt; x ≤ 750K</td>
<td>Australia (30)</td>
<td>3</td>
<td>10</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>Very Large</td>
<td>&gt; 750 KSLOC</td>
<td>Europe (101)</td>
<td>7</td>
<td>53</td>
<td>26</td>
<td>15</td>
</tr>
</tbody>
</table>

November 2015 © 2015, Reifer Consultants LLC
Scrum is the Leader of the Pack

Agile Method Usage by Size of Project

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>
Some More Observations

- The 20% other methods was made up of:
  - AUP
  - XP
  - Crystal
  - Other

- We estimate about 30 to 45% of those who claimed to be agile were stretching things
  - Self-organizing and managed teams the biggest controversy

- Most large projects had teams that were geographically distributed
  - Most of firms using these have instituted some form of project management

- Many of the larger organizations still used CMMI and ISO process frameworks
  - Many who did so for governance or compliance reasons
Feature council collaboratively makes decisions relative to product vision, features, project staffing, process, issues, risks and other related areas effecting delivery.
### Government Lags Industry by 5 to 7 Years Outside of the UK*

<table>
<thead>
<tr>
<th></th>
<th>EA*</th>
<th>EM*</th>
<th>LM*</th>
<th>Laggards*</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Europe (44)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Industry (36)</td>
<td>22%</td>
<td>44%</td>
<td>25%</td>
<td>9%</td>
</tr>
<tr>
<td>Government (8)</td>
<td>30%</td>
<td>33%</td>
<td>3%</td>
<td>34%</td>
</tr>
<tr>
<td><strong>United States (108)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Industry (95)</td>
<td>19%</td>
<td>39%</td>
<td>25%</td>
<td>17%</td>
</tr>
<tr>
<td>Government (11)</td>
<td>36%</td>
<td>27%</td>
<td>2%</td>
<td>25%</td>
</tr>
<tr>
<td><strong>Asia (48)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Industry (41)</td>
<td>16%</td>
<td>45%</td>
<td>20%</td>
<td>19%</td>
</tr>
<tr>
<td>Government (7)</td>
<td>31%</td>
<td>32%</td>
<td>4%</td>
<td>33%</td>
</tr>
</tbody>
</table>

**Legend**

* Numbers include those organizations for which we have verified data last month

**EA** – Early Adopters

**EM** – Early Majority

**LM** – Late Majority

**Laggards** – considering agile

* UK government mandated the use of agile methods in 2010 for Info Tech (IT)
## Why So Much Fuss about Agile?

<table>
<thead>
<tr>
<th><strong>Productivity is better</strong></th>
<th><strong>Cost is cheaper</strong></th>
</tr>
</thead>
</table>
| • We looked at over 1,500 projects from 100 firms  
  – 10 applications domains  
  – 750 agile projects  
  – 750 traditional projects  
  – All data is less than 10 years old | • Cost is cheaper  
  – Again many caveats  
  – Data localized to USA for this analysis  
  – Data again supports the conclusions agile is between 10 to 25% cheaper than norms after currency issues resolved |
| • Productivity is better  
  – Many caveats associated with the statement  
  – Many critics argue results due to Hawthorne effect – they always do when positive  
  – However, the data supports the conclusion that agile performance is between 10 to 35% better than traditional norms | • Quality gets better with age  
  – Some controversy  
  – Data supports agile is 10 to 40% better after transition  
  – Several databases in addition to ours support this conclusion  
  – Some argue that advocates put too much attention on test and not enough on engineering quality into the product |
What Does the Other Data Say?

• ISBSG – Australia
  – Data from 10+ nations
  – Mostly IT projects analyzed
    • 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 surveys
  – Sources include Agile Alliance and variance agile SIGs
  – Shows much higher gains
  – Data not validated

• QSM Data
  – Faster, cheaper and better quality

• David Rico
  – Summarizes other’s studies
  – Faster, cheaper and better quality
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 average = 375 eSLOC/SM vs. Traditional average = 335 eSLOC/SM

Agile vs. Traditional Software Productivity Trends
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 and how they were calculated
  – The following major costs are not factored into the computation 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
• 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)
  – 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 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%.
- Traditional projects that deliver 100% functionality often exceed promised deadlines and/or budgets by 40 to 50%.
Agile vs. Traditional Quality

- At first, agile quality is not as good as that computed for traditional projects.
- Break-even is in about 3 years.
- Root cause seems to be that too much attention is placed on testing and not enough on engineering quality into the product.

Agile average = 3.5 defects/KeSLOC vs. Traditional average = 3 defects/KeSLOC

Agile Adoption & Quality as Measured by Defect Rates
Quality Survey- Defect Density

• Confirms earlier findings
  – Reifer, Capers Jones and ISBSG
• Results favorable to agile
  – Agile averages about 20% better
  – Defense was 33% better
  – Telecom and mobile the same
• Findings based on 662 projects
Quality Survey – Defect Rates

Figure 2 - Reliability by Domain Measured by Defect Rate [(Defects Found)/Month]

- Defect find rates were lower overall for agile except for defense
  - Agile averages about 30% lower
  - Defense averages about 25% higher (surprising – viewed as positive)
- This was an metric captured by another survey
- Findings based on 555 projects
Quality Survey – Fix Response Time

- Defect repair times were better for agile except in the medical and mobile domains
  - Averaged between 30 to 40% quicker
  - Defense was 60% better
  - Medical and mobile were same for both methodologies
- Findings based on 425 projects
Quality Survey – Defects Introduced by Fixes (Escapes)

- Results are domain specific
  - Defense, telecom and medical the same
  - Financial and C&C fairied best using traditional
  - Other domains did best using agile methods
- Findings based on 433 projects

Figure 4 - Maintainability by Domain Measured by Number of Defects Introduced during Fixes
Quality Survey – Fitness for Use

- 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
Lessons Learned from Agile Adopters

During Early Adoption Stage

• Define transformation goals and expectations – issue vision paper
• Tie your initiatives to your firm’s business goals and tailor your measures of success accordingly
• Educate your management, engage them frequently via demos, and keep them informed throughout the transformation
• Educate everyone else on staff including your customers so that they know and agree to what you are trying to accomplish
• Select the agile method that makes sense for you, train your people in its use and tailor it by using it on one or more pilot projects
• Have the pilot define the rules, metrics and processes as they need them in a just-in-time manner
• Develop a fan-out plan as pilots near their end using opportunities
• Plan to use actual stuff from the pilots in your training materials
• Publish a retrospective to capture the lessons learned during pilots
More Lessons Learned

Fan-out to Early Majority

- Start small, build big – “Many small wins equal a big one”
- Fan out experienced people from your pilots to lead the next projects
- Use the people from your pilots to train others in a just-in-time manner
- If using Scrum, build a cadre of Scrum Masters and then pool and use them as needed on your teams
- Coach the product owners so that they understand their roles – act as voice of the customer, not project managers
  - If using IPTs, make product owner the chair instead of systems engineering
- Embrace test-first concepts and automate your regression testing using a standard toolset selected for that purpose (Bonfire, VersionOne, etc.)
- In addition to issues, look for risks at your daily standups, list them and take action to mitigate them based on their priorities and consequences
- Stress building team rhythm, i.e., increasing delivery rate using velocity and other metrics as measures
Even More Lessons Learned

– Enlist the aid of your quality assurance personnel as you look to teach teams how to engineer quality into your products
– Use the metrics and measures to help manage the effort
  • See our metrics guide to help define what makes sense for your project
– Build momentum for change through people by spreading the word via networking, newsletters and publicity events
– Anticipate the issues during fan-out and work them in parallel so that solutions are available when needed by the Late Majority
  • Issues for agile-at-scale projects are different than those for small to medium developments
  • Biggest issues for agile-at-scale projects include process harmonization, project versus product management roles, job/career paths, openness versus control, collaboration and teamwork, and contracting philosophy
  • Remember, most organizations use hybrid approaches to address such issues; i.e., need to harmonize with their processes and infrastructure
Final Lessons Learned

Mature Operations

• Continue to refine and reinvent yourself – go lean as a next step
• Modify your infrastructure and institutionalize your processes
  – Goal is to have a single engineering process for all disciplines to use
  – However, must do more, i.e., must modify how finance, HR and other non-engineering disciplines do business so as to support the agile principles
• Put in place a management structure that assures your management and customers that both you and they are in control
  – Project and program management concepts still need to be applied, but in an agile rather than traditional manner (agile PMI certification)
  – Discipline and agility must be balanced especially in regulated environments where safety and security issues dominate (banking, defense, etc.)
• The philosophy that work is to do the right thing at the right time for the right reasons
  – If it makes good technical and business sense, you can sell it especially if you have developed the sponsorship and a good business justification

Laggards can catch up quickly via strategies outlined in my recent paper on the topic
Process Harmonization

- At first look, the process frameworks like the CMMI® and agile 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 have been singled out as the biggest issue in process improvement
  - Many firms believe that they are a waste of time
  - Others believe that they force you to implement agile in unnatural ways
  - Majority of commercial firms have moved to self-assessments as an alternative
- 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
Many agile proponents equate quality with testing
  – Their focus is therefore 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 quality assurance as wasteful effort
  – Argue that you can’t inspect quality into products
More forward leaning agile firms focus on engineering quality into products
  – Redefine quality’s role to being teachers and refocus their work
  – Quality personnel work as members of teams rather than in an independent capability
Emphasis of many is placed on test because of need to revalidate many iterations
  – Regression test baselines created and delivered with products
  – Automated test tools used in conjunction with version control to manage tests and test data
One Engineering Process

- At the enterprise level, many firms have defined a single engineering process
  - Commercial – Adobe, Fidelity, etc.
  - Defense – all of the big boys say they do it
- For Info Tech (IT), it is relatively easy
  - Lots of models exist and help is available
- For systems 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
  - Products that result are sort of agile
  - Many retain their old ways and don’t realize the full benefit from the change
Summary of Agile Strengths

• Besides real advantages in time-to-market, productivity and cost, other strengths include:
  – Ability to react quickly to changing priorities
  – Less documentation (+ & -)
  – Increased and continual dialog with customers/users
  – Emphasis on teams, team building and teamwork
  – Increased openness as working code is made available to everyone
  – Emphasis on continuous integration and automated regression testing
  – Increased willingness to try something new
• Besides the identified quality issues, other agile weaknesses include:
  – Contracting/subcontracting issues abound for agile
    • How do you deal with liability issues?
    • What are good contract terms and conditions?
    • What are the deliverables?
  – Oversight management and IV&V practices
  – Agile QA needs to be viewed as much more than just testing
  – Systems engineering mismatches
  – Progress measurement and reporting issues
  – Scalability
    • Addressing global projects
  – Project level risk management practices for agile
  – The jury is still out on agile maintenance

Of Course, Some Negatives
Most Important Positive - Business Value is Emphasized

- Agile pays attention to providing value to the customer
  - Everything done is prioritized and tied to measures of value (time to market, etc.)
  - The customer is engaged throughout the development (not just at reviews)
- Base choices provide maximum business value to the customer
  - Especially between tradeoffs made between cost-time-scope (the iron triangle)
- Benefits derived are quantified and reviewed
  - ROI, competitiveness, capability, capacity along with cost, time-to-market and quality gains
Summary - Beware the Hype

• Agile is the primary way used to develop software worldwide
  – Scrum is the primary approach for small to medium projects
  – Hybrid approaches are used for agile-at-scale jobs

• The agile pluses are:
  – Higher productivity and quality
  – Lower costs
  – Lots of softer factors that lead to higher morale and motivation

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

Want to Buy a Used Car?
Conclusions – Be Successful

• We believe that use of agile methods has merit and is worth pursuing
  – Tie adoption to your business goals and a solid business case

• When making the move, do so with both your eyes and ears wide open
  – Do what makes sense in your firm and applications domain
  – Listen to the practitioners, not the zealots

• Post-introduction
  – Fan-out the technology and develop the infrastructure needed to address issues that can impede its adoption

• Our success strategy - make a positive contribution
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?
When eating an elephant take one bite at a time.

Creighton Adams

When you are finished changing, you are finished.

Benjamin Franklin

If you want to make enemies, try to change something.

Woodrow Wilson

Change before you have to.

Jack Welch

Let’s Keep Things Simple