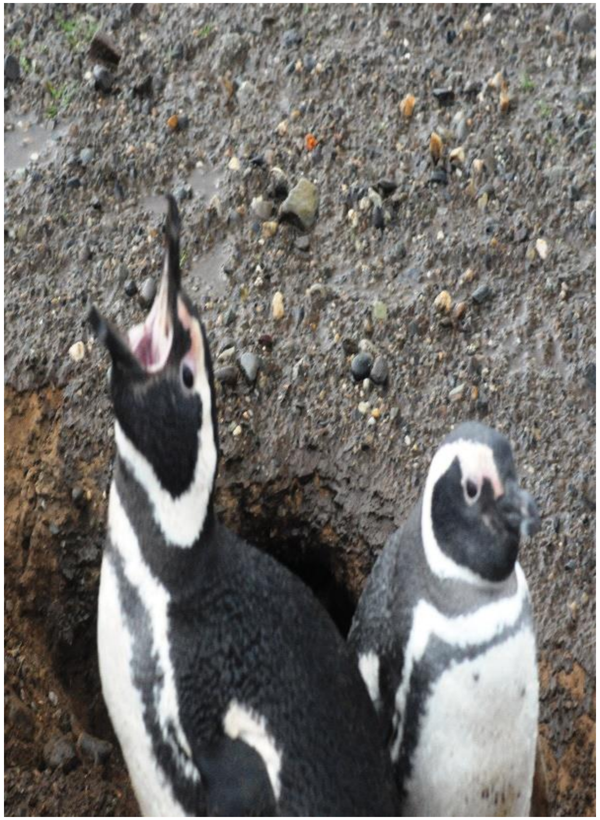
Agile Benchmarks: What Can You Conclude?

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

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



What Is He Saying?

Setting the Stage

Looking at Agile Worldwide

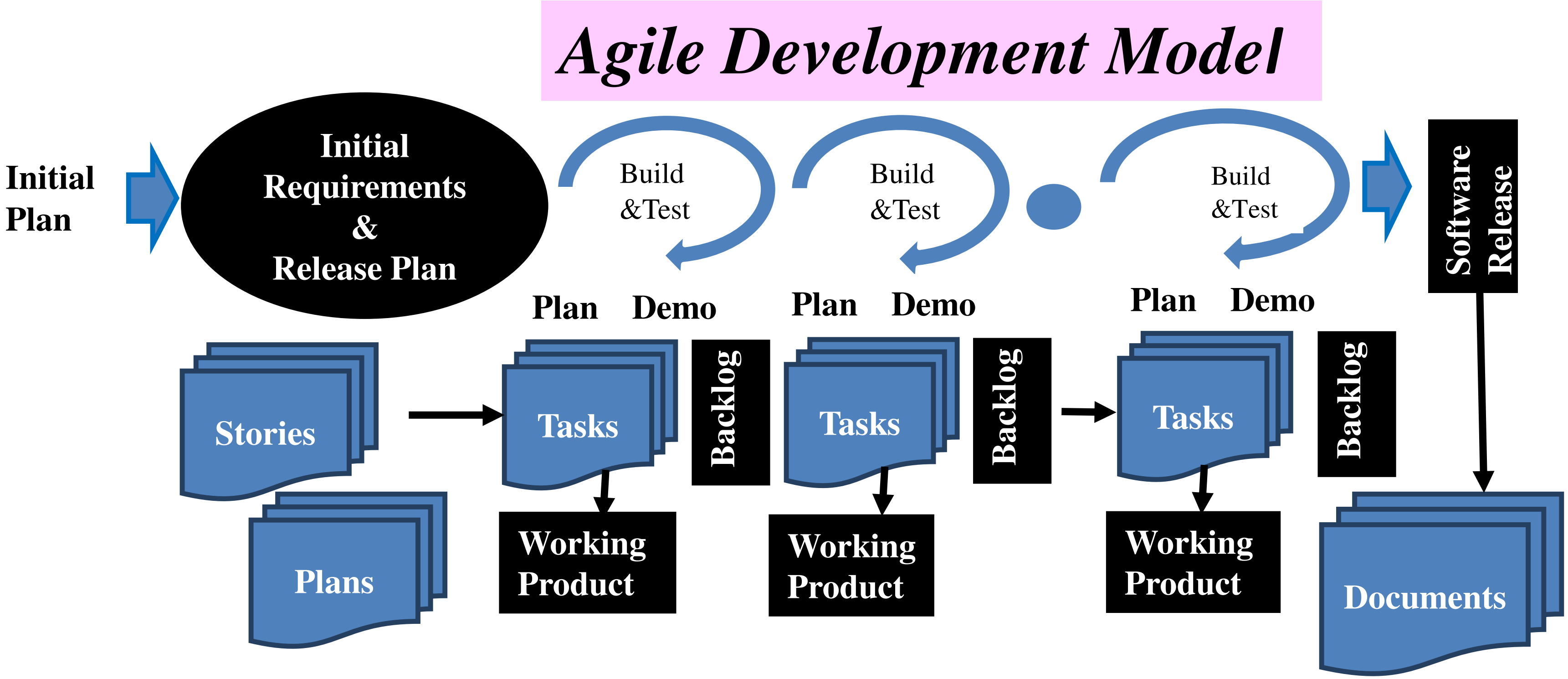
- Polled over 500 organizations to determine state of agile adoption worldwide
  - Response rate of 60% with a little over 300 organizations responding
  - 18 countries responded including:

Australia
Brazil
Canada
China
Finland
France
Germany
India
Israel
Italy
Japan
Mexico
Netherlands
Russia
Sweden
UK
USA
Viet Nam

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

Agile Software Life Cycle



- 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 that 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 adheres 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

- We looked at 2,000 projects from 100 firms
  - 8 applications domains
  - 1,000 agile projects
  - 1,000 traditional projects
  - All data is less than 10 years old
- 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

- Costs are 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
- 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

Demographics of Use

Applications Domain	No. of Firms	No. of Completed Software Projects	Average Experience with Adopted Agile Method			Example Products Generated by Firm
			1 - 2 Years	3 to 5 Years	5+ Years	
Automation	10	100	25%	30%	45%	Pipeline automation
Financial	10	100	20%	30%	50%	Day trading system
Defense	20	250	35%	45%	20%	Weapons system
Info Technology	15	150	15%	30%	55%	ERP applications
Medical	5	100	25%	40%	35%	Pharmacy
Software Tools	10	100	15%	30%	55%	Compiler system
Telecom.	20	100	20%	30%	50%	Switching system
Web Business	10	100	30%	30%	60%	Web site for travel
TOTALS	100	1,000	22%	32%	46%	Weighted Averages

About 3/4 of projects completed with teams with over 3 years of experience

Method Use by Application Domain

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

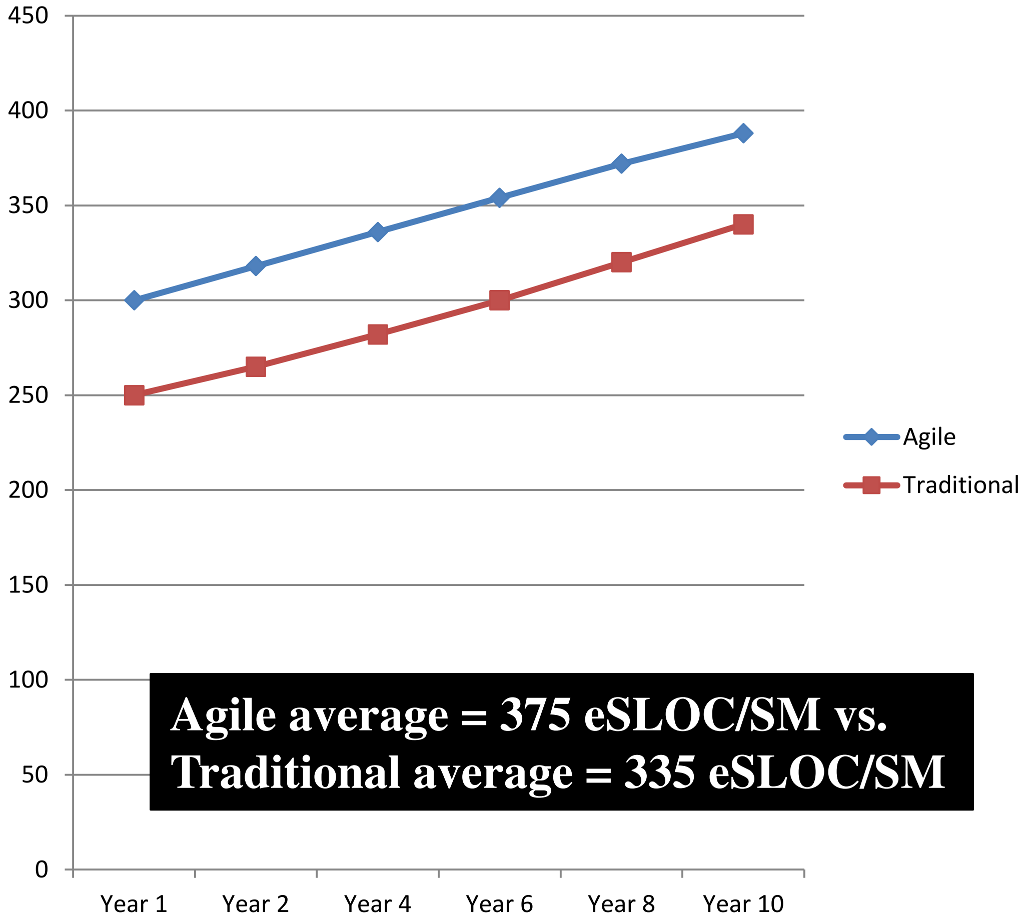
Popularity of methods is a function of project size and scaling – Scrum for small-medium, Hybrid for large

Agile Productivity

How much improvement?

- 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 average = 375 eSLOC/SM vs. Traditional average = 335 eSLOC/SM

Agile vs. Traditional Software Productivity Trends

- 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

Productivity Comparisons – Agile versus Traditional

Applications Domain	No. of Firms	No. of Agile Projects	Agile Productivity (eSLOC/SM)	No. of Non-Agile Projects	Non-Agile Productivity (eSLOC/SM)	% Gain/ Year
Automation	10	100	365	100	310	18
Financial	10	100	[redacted]	100	[redacted]	12.5
Defense (weapons only)	20	250	332	250	289	15
Info Technology	15	150	505	150	455	11
Medical	5	100	[redacted]	100	[redacted]	9
Software Tools	10	100	[redacted]	100	[redacted]	15
Telecommunications	20	100	[redacted]	100	[redacted]	10
Web Business	10	100	[redacted]	100	[redacted]	12
TOTALS	100	1,000	438	1,000	390	12.5

Computed using method of weighted averages

Defense Productivity Comparisons – Agile versus Traditional

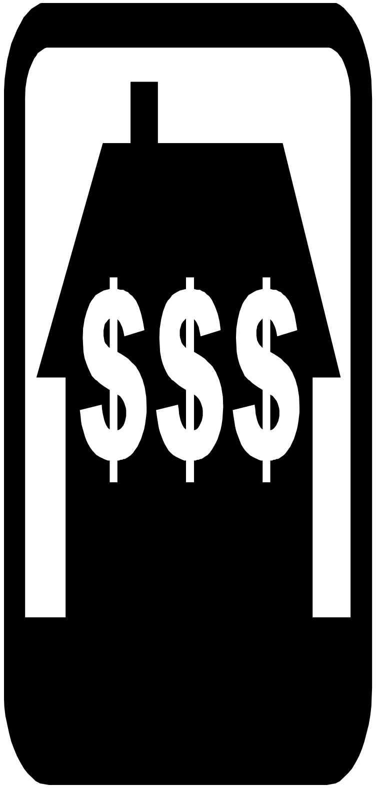
Applications Domain	No. of Firms	No. of Agile Projects	Agile Productivity (eSLOC/SM)	No. of Non-Agile Projects	Non-Agile Productivity (eSLOC/SM)	% Gain/ Year
Avionics	3	10	[redacted]	75	[redacted]	8
Ground	7	50	287	75	265	11
IT	10	50	489	100	435	12
Medical	15	50	[redacted]	50	[redacted]	13
Missile	3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

Computed using method of weighted averages

*** Some firms active in more than one domain**

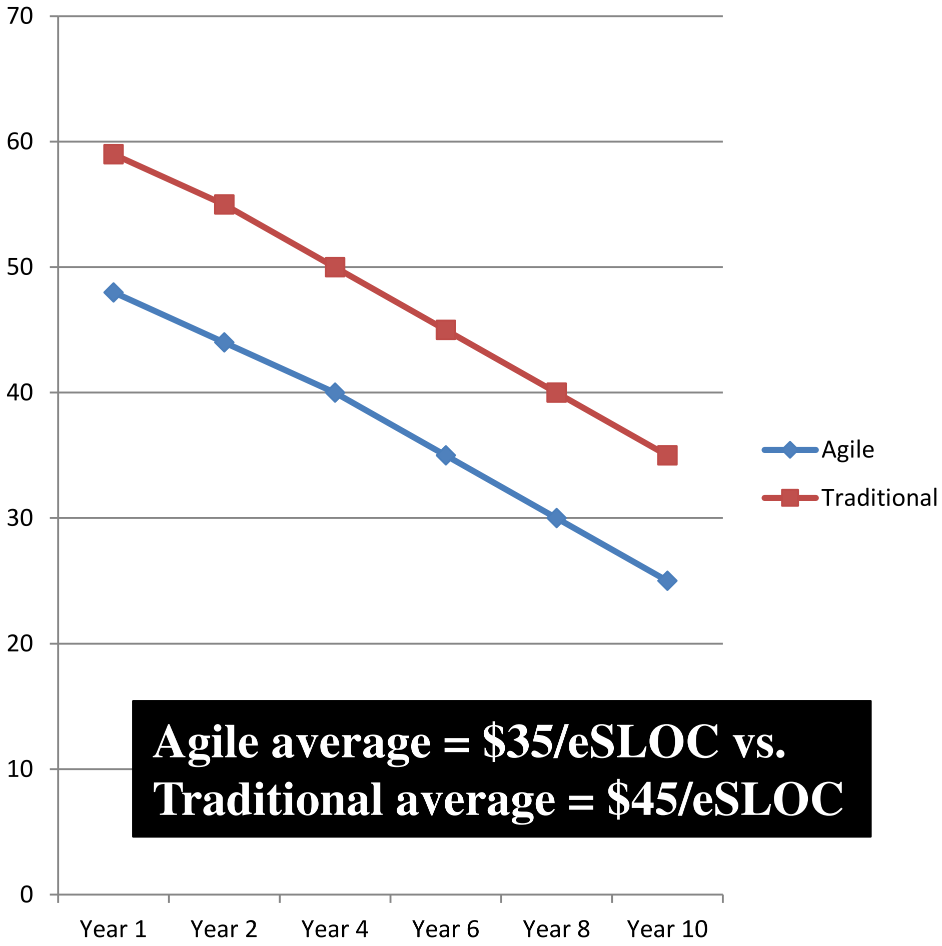
Reifer - Agile Cost

- 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.)



How Much Did you Say?

Agile Vs. Traditional Cost



**Agile average = $35/eSLOC vs.
Traditional average = $45/eSLOC**

Agile vs. Traditional Software Cost Trends

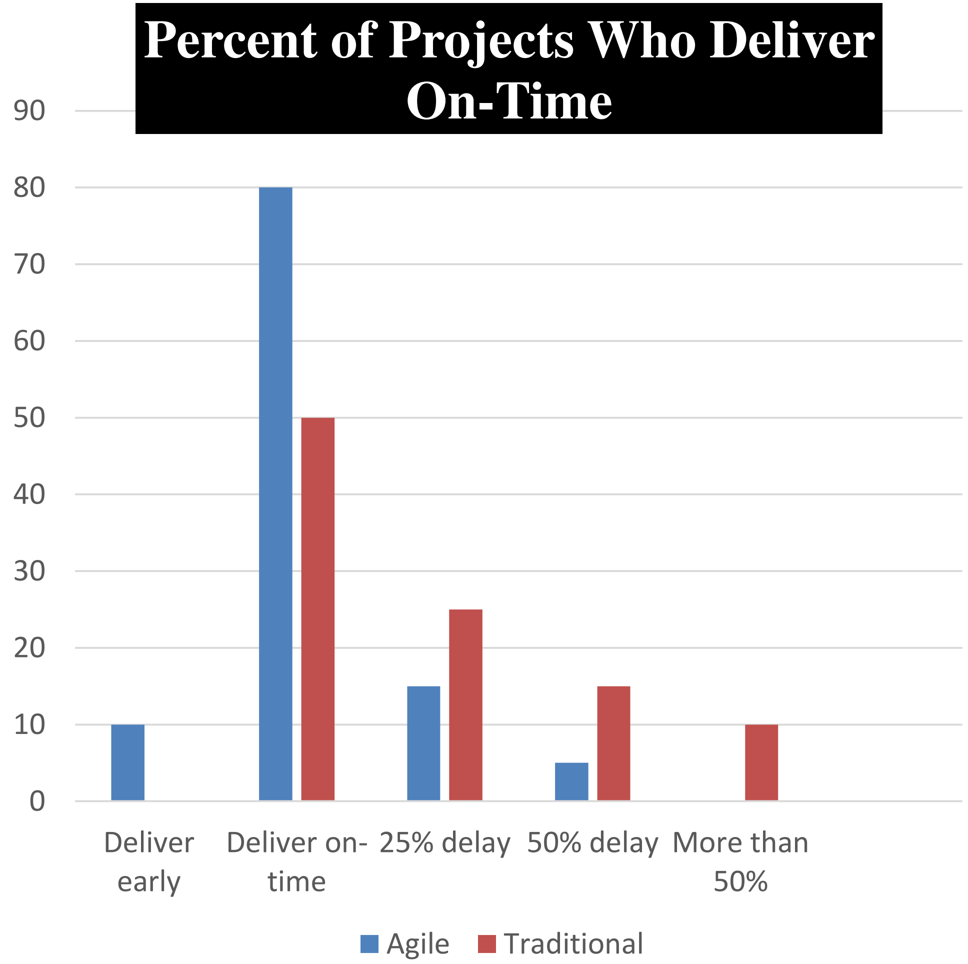
- Agile methods are between 10 and 25% cheaper even when labor rates are normalized across domains
- As noted, many of the costs related to managing the transition to agile are not accounted for
- Average reduction during past three years averaged 8 to 12% annually
- Seems to be many issues related to agile supplier management

Cost Comparisons – Agile versus Traditional

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

Computed using method of weighted averages

Agile Time-to-Market

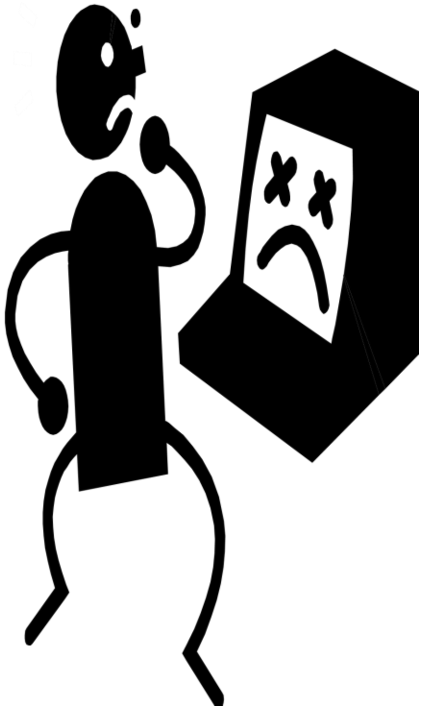


Percent of Projects Who Deliver On-Time

- 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%

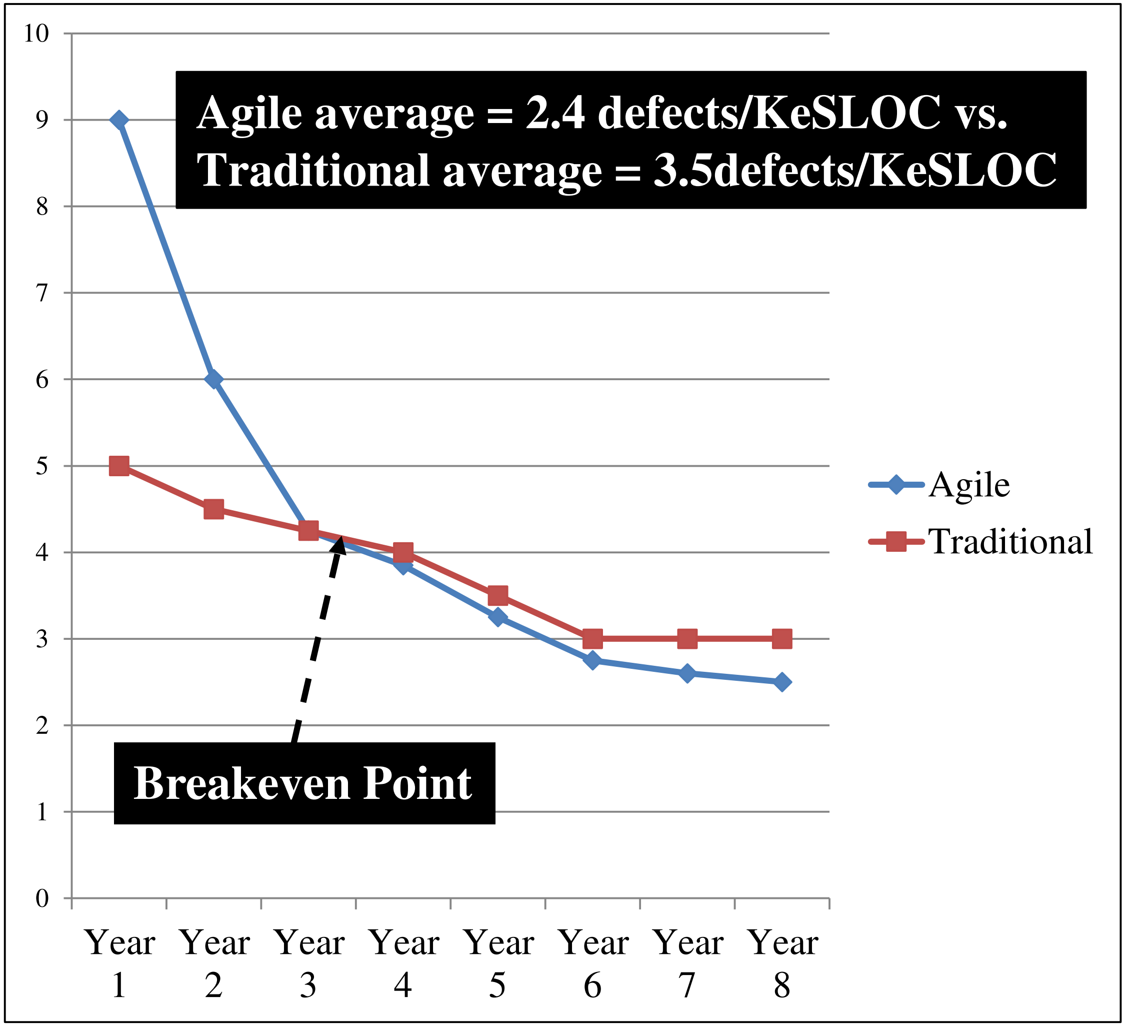
Reifer - Agile Quality

- 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



Bugs, What Bugs?

Agile Vs. Traditional Quality



Agile average = 2.4 defects/KeSLOC vs. Traditional average = 3.5defects/KeSLOC

Breakeven Point

- 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

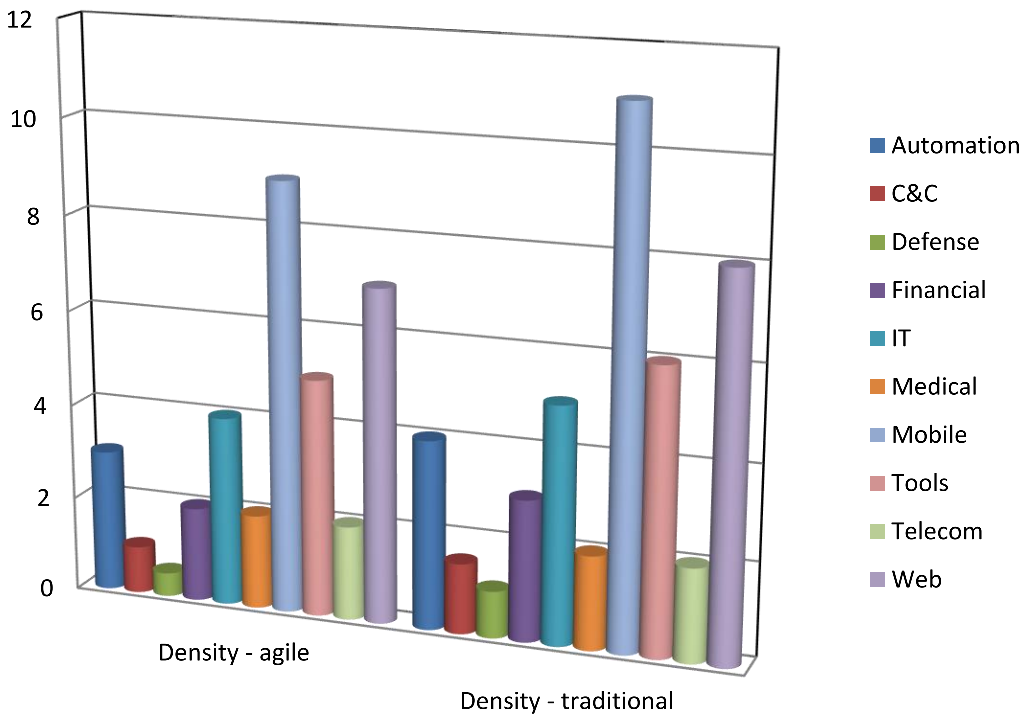
Quality Comparisons – Agile versus Traditional

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

Computed using weighted average number of defects

Quality Survey- Defect Density

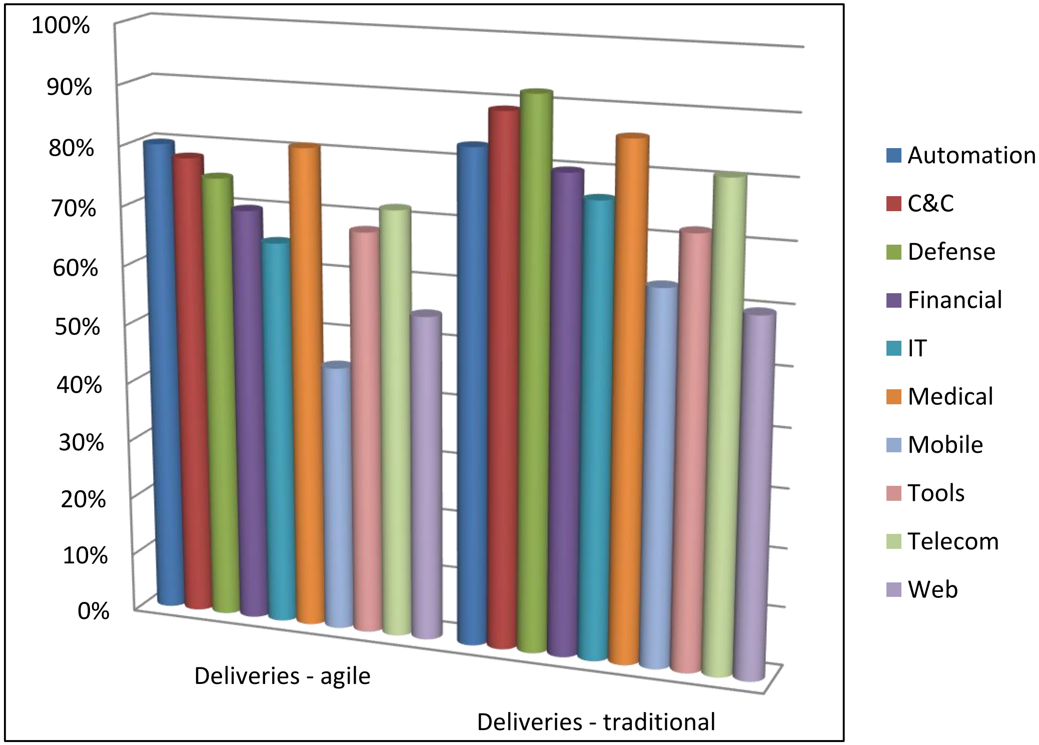
Figure 1 - Reliability by Domain Measured by Defect Density (Defects/1,000 Unadjusted Function Points [UFP])



- 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

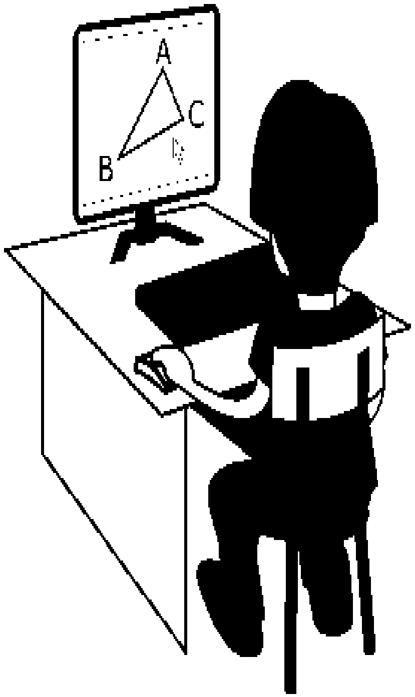
Quality Survey – Fitness for Use

Figure 5 - Fitness for Use by Domain Measured by Percentage of Stories or Features/Functions Delivered as Promised



- 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

Agile Quality During Development



When Will I Finish?

- 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

Agile Challenge - Software Quality

- 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?

- 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

Example Results - QSM

Demographics

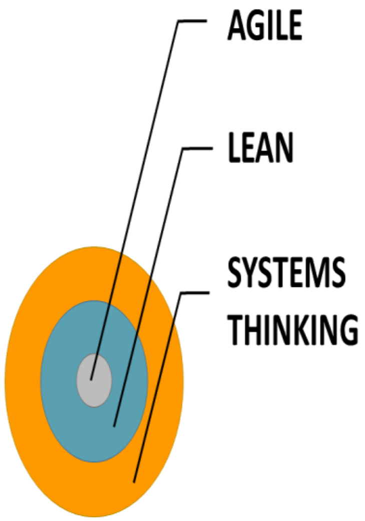
- 54 recently completed Agile projects/total not disclosed
- 12 different companies
- 87% business, 7% scientific applications, 6% system software
- Team size clustered in 5-10 and 20-50 ranges
- Median size 42.9k lines of code
- Median effort 47 staff months
- Median staff 7.5
- Median duration 6.1 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.

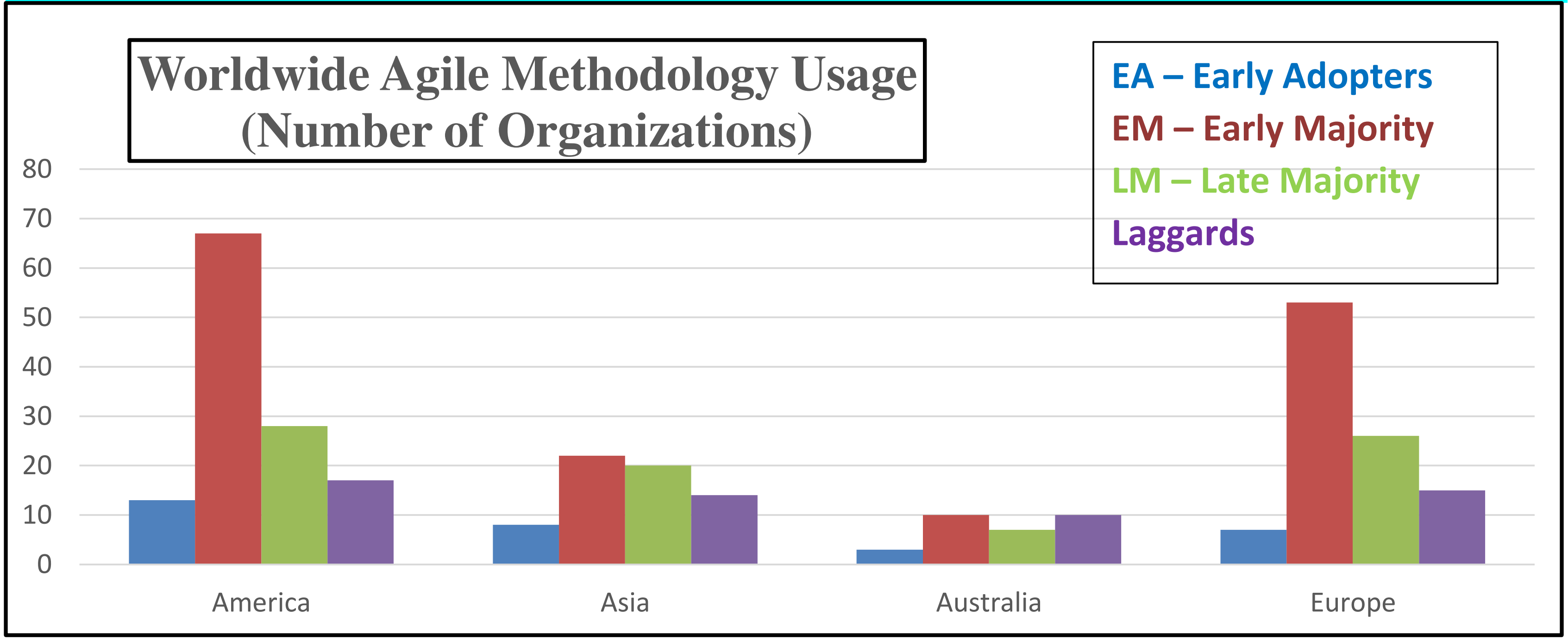
More Interesting Findings



Agile may be Here to Stay

- 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

Agile Is Here To Stay



Size of Organizations

Size	No. Employees
Small	< 500
Medium	500 to 5,000
Large	5,000 to 50,000
Very Large	> 50,000

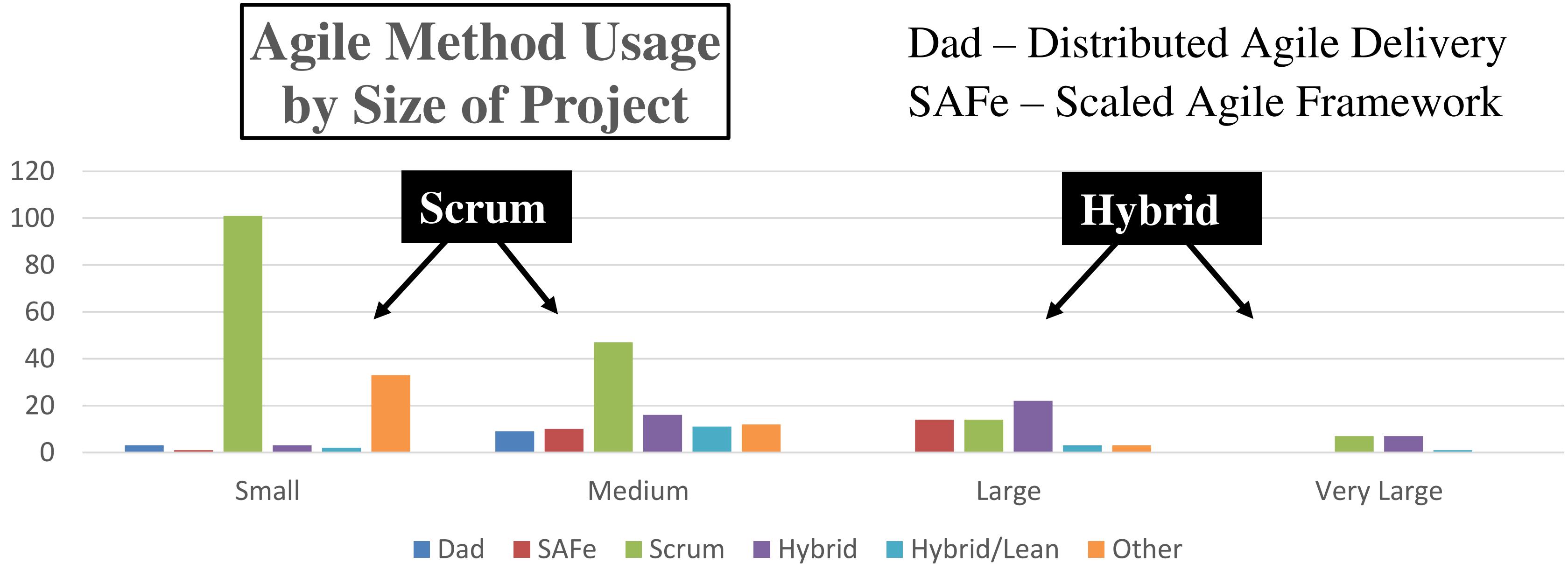
No. of Organizations/Transition Stage

Location	EA	EM	LM	Laggards
America (125)	13	67	28	17
Asia (64)	8	22	20	14
Australia (30)	3	10	7	10
Europe (101)	7	53	26	15

Scrum is the Leader of the Pack

Dad – Distributed Agile Delivery

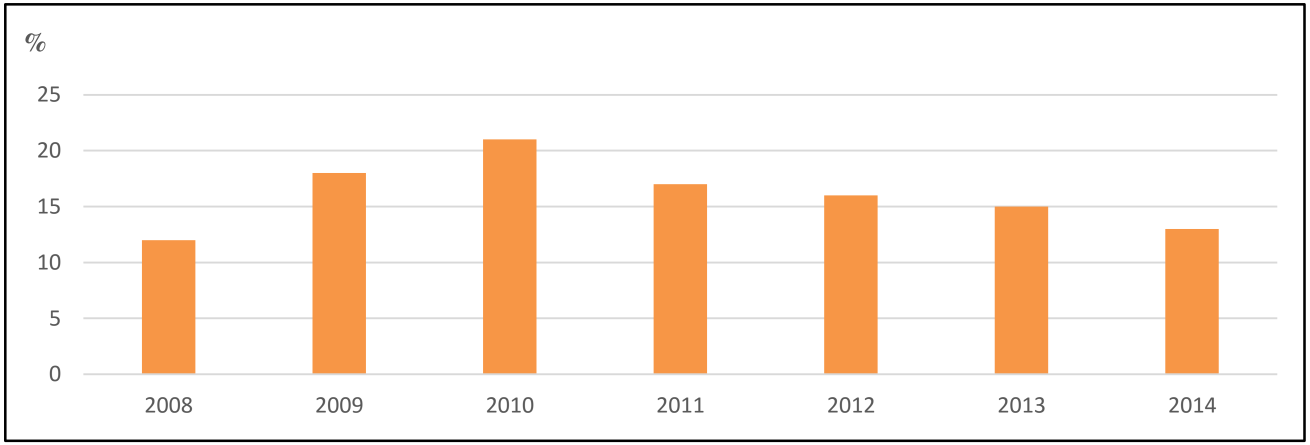
SAFe – Scaled Agile Framework



Agile Method Usage by Number of Projects

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

Agile and the CMMI®



Percentage Agile as a % of those rated via CMMI®

Process Maturity by SDLC & Sector

Sector	SDLC	Number of Organizations Rated for SDLC			
		CMMI	ISO	OMG BPMM[21]	Not Reporting
Commercial (70)	Traditional	3	16	3	12
	Agile		36		
Defense (30)	Traditional	15			4
	Agile	11			
TOTAL		29	52	3	16

Agile Challenge – Maintaining a Single Engineering Process

An Agile Way of Doing Business

- 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

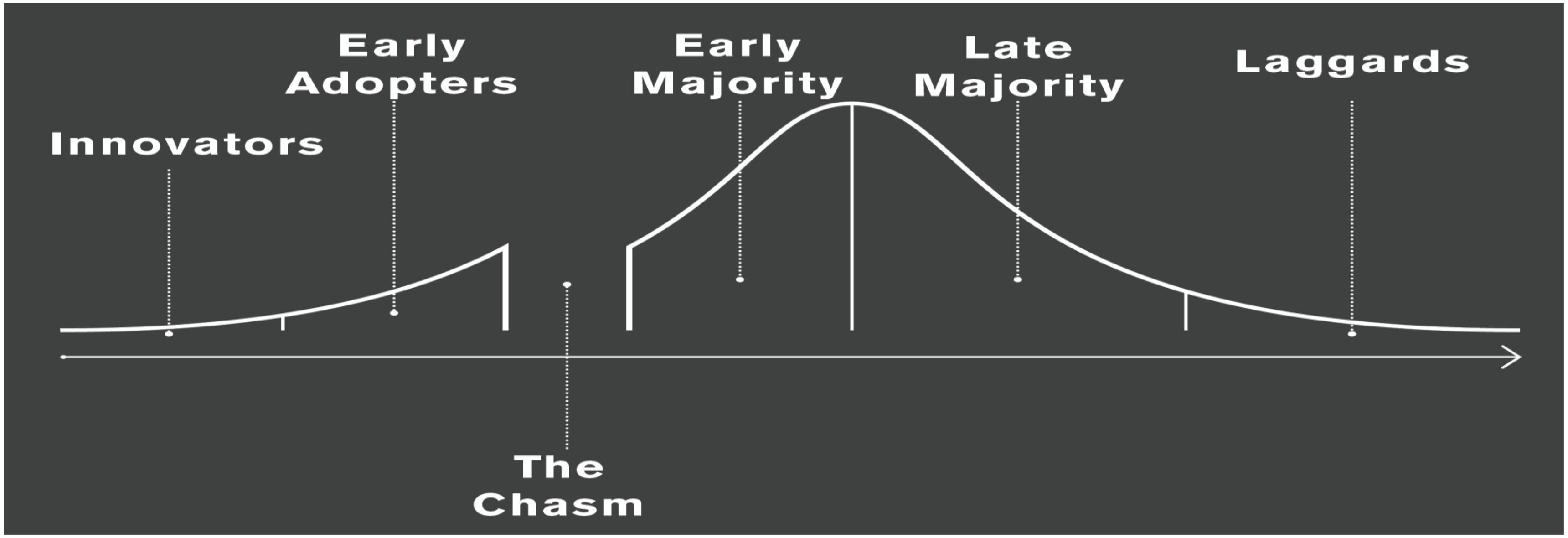
Our other reports which may be of interest to you include:

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

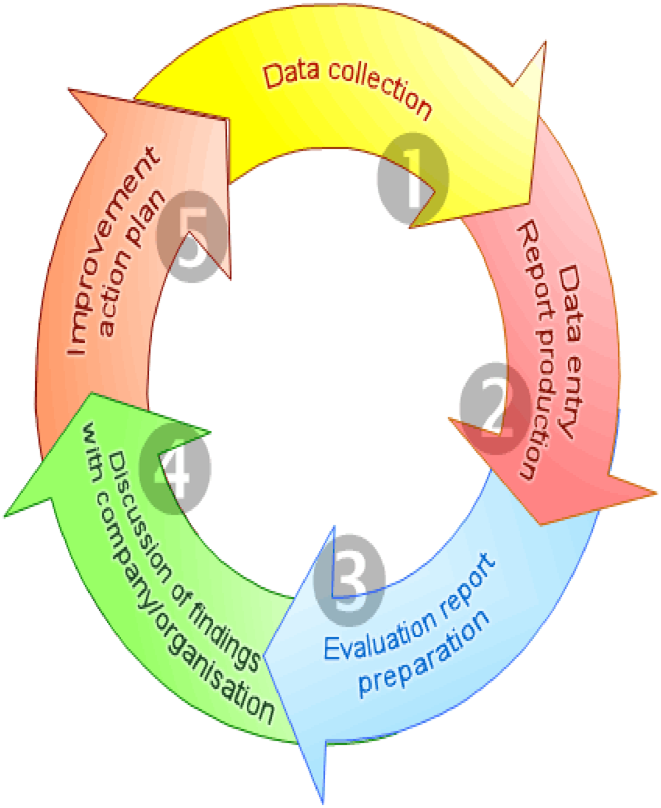
Lots of Agile Resources Available

- 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


Data collection
1
Data entry
Report production
2
Evaluation report
preparation
3
Discussion of findings
with company/organisation
4
Improvement
action plan
5