

Scaling Agile Software Development: Strategies for Applying Agile in Complex Situations

Scott W. Ambler
Practice Leader Agile Development,
IBM Rational

scott_ambler@ca.ibm.com



IBM Rational Software Development Conference 2008

WHERE TEAMS ARE **R-HEROES**



Discussion Topics

- Agile Adoption Rate
- Case Study: Large-Scale Agile Adoption
- Agile@Scale
- Practices for Scaling Agile
- A systematic approach to agile adoptions
- Parting Thoughts



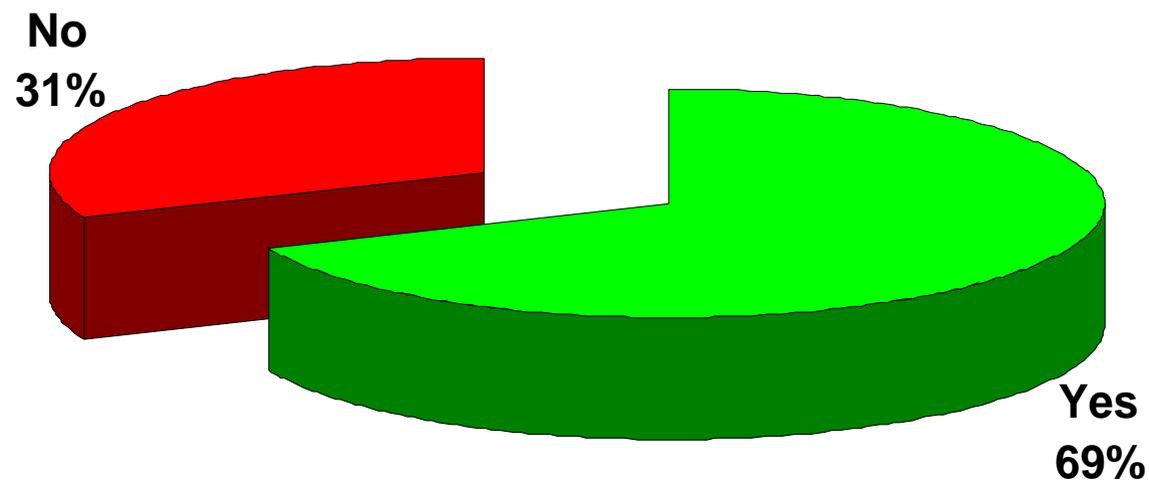
Survey Results

- Agile Adoption Survey
 - February 2008
 - Message sent out to DDJ mailing list
 - 642 respondents:
 - ▶ 54.8% were developers, 29.4% were in management
 - ▶ 41.6% had 10-20 years IT experience, 37.2% had 20+ years
 - ▶ 37.7% worked in orgs of 1000+ people
 - ▶ 71% worked in North America, 17% in Europe, 4.5% in Asia
- Project Success Survey
 - ▶ August 2007
 - ▶ Email to DDJ mailing list
 - ▶ 586 respondents
 - 54% were developers/modelers, 30% were in management
 - 73% had 10+ years in IT
 - 13% worked in orgs of 1000+ IT people
 - 84% worked in commercial firms
 - 69% North American, 18% European

Data, summary, and slides downloadable from www.ambyssoft.com/surveys/



Has Your Organization Adopted One or More Agile Techniques?



18.3% of respondents indicated they're still in the pilot stage

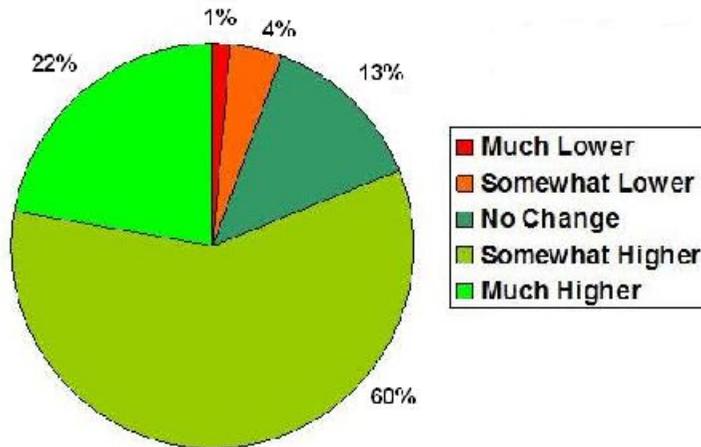
15% of "No" respondents hope to do Agile this year

Source: Dr Dobb's 2008 Agile Adoption Survey

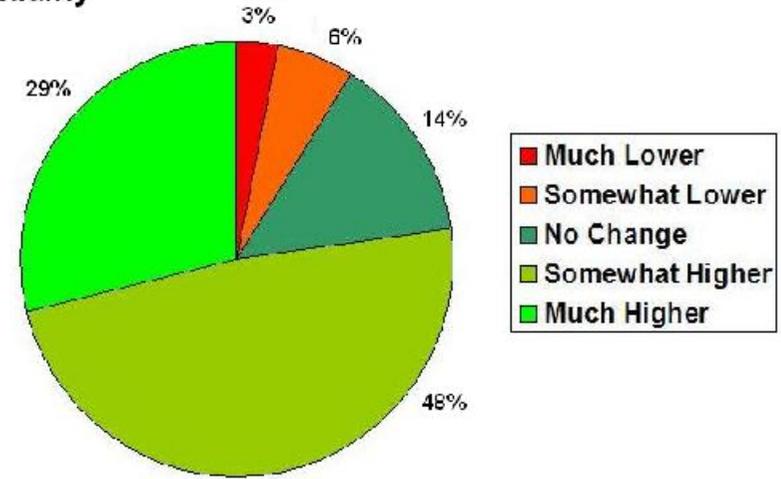


Why Agile? Because it Works!

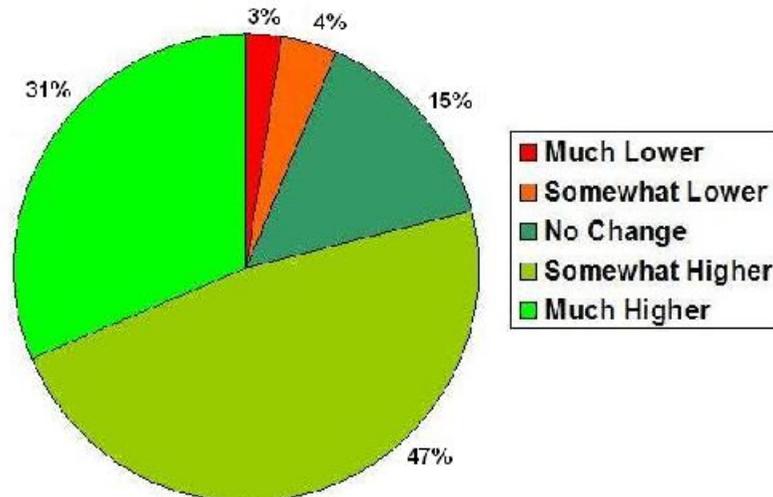
Productivity



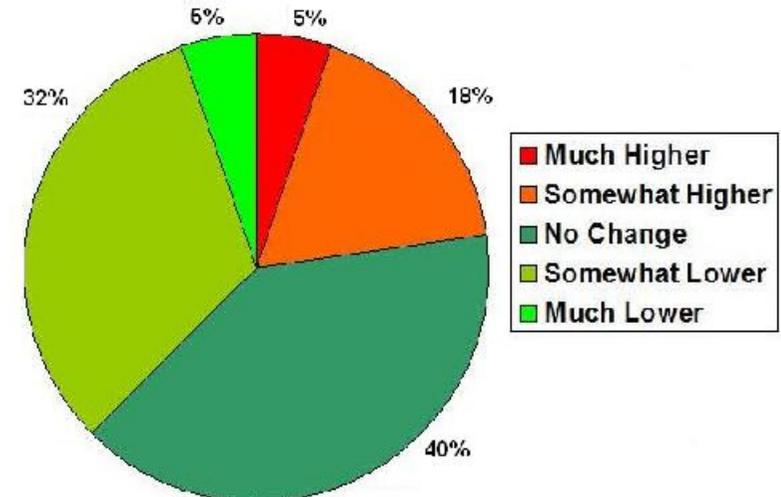
Quality



Business Stakeholder Satisfaction



Cost of System Development



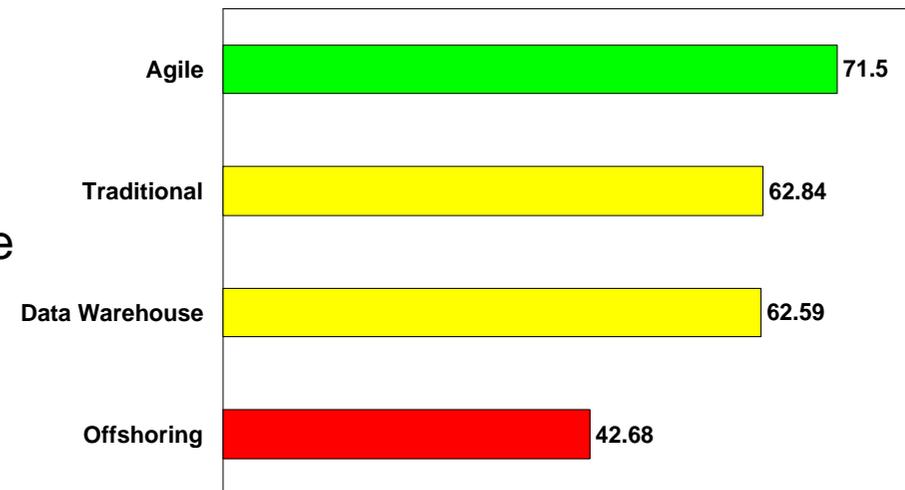
Source: Dr. Dobb's Journal 2008 Agile Adoption Survey

Copyright 2008 Scott W. Ambler



Why Agile/Lean? It's More Successful

- Quality: 87.3% believe that delivering high quality is more important than delivering on time and on budget
- Scope: 87.3% believe that meeting actual needs of stakeholders is more important than building the system to specification
- Money: 79.6% believe that providing the best ROI is more important than delivering under budget
- Staff: 75.8% believe that having a healthy workplace is more important than delivering on time and on budget
- Schedule: 61.3% believe that delivering when the system is ready to be shipped is more important than delivering on schedule

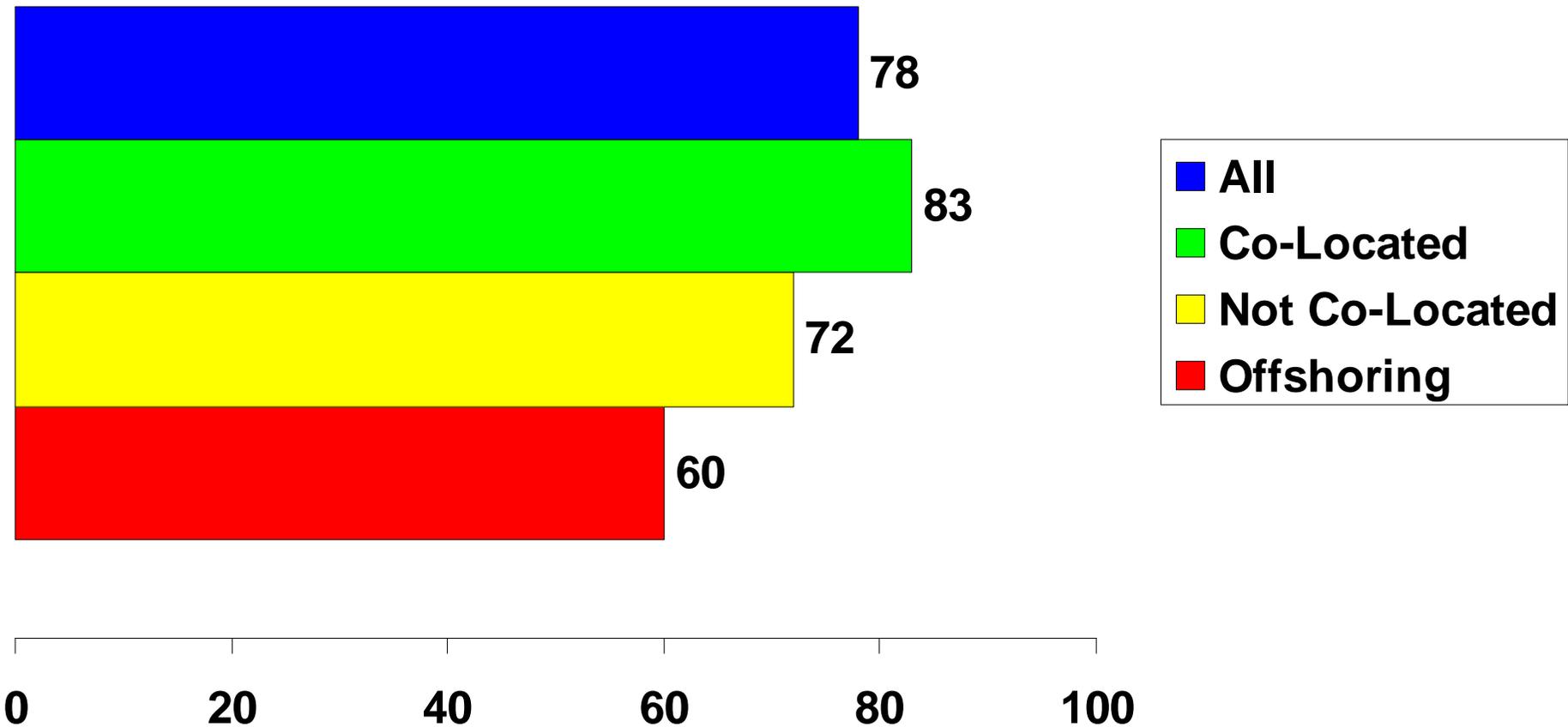


Source: Dr Dobb's 2007 Project Success Survey



Agile Projects Success Rates (%)

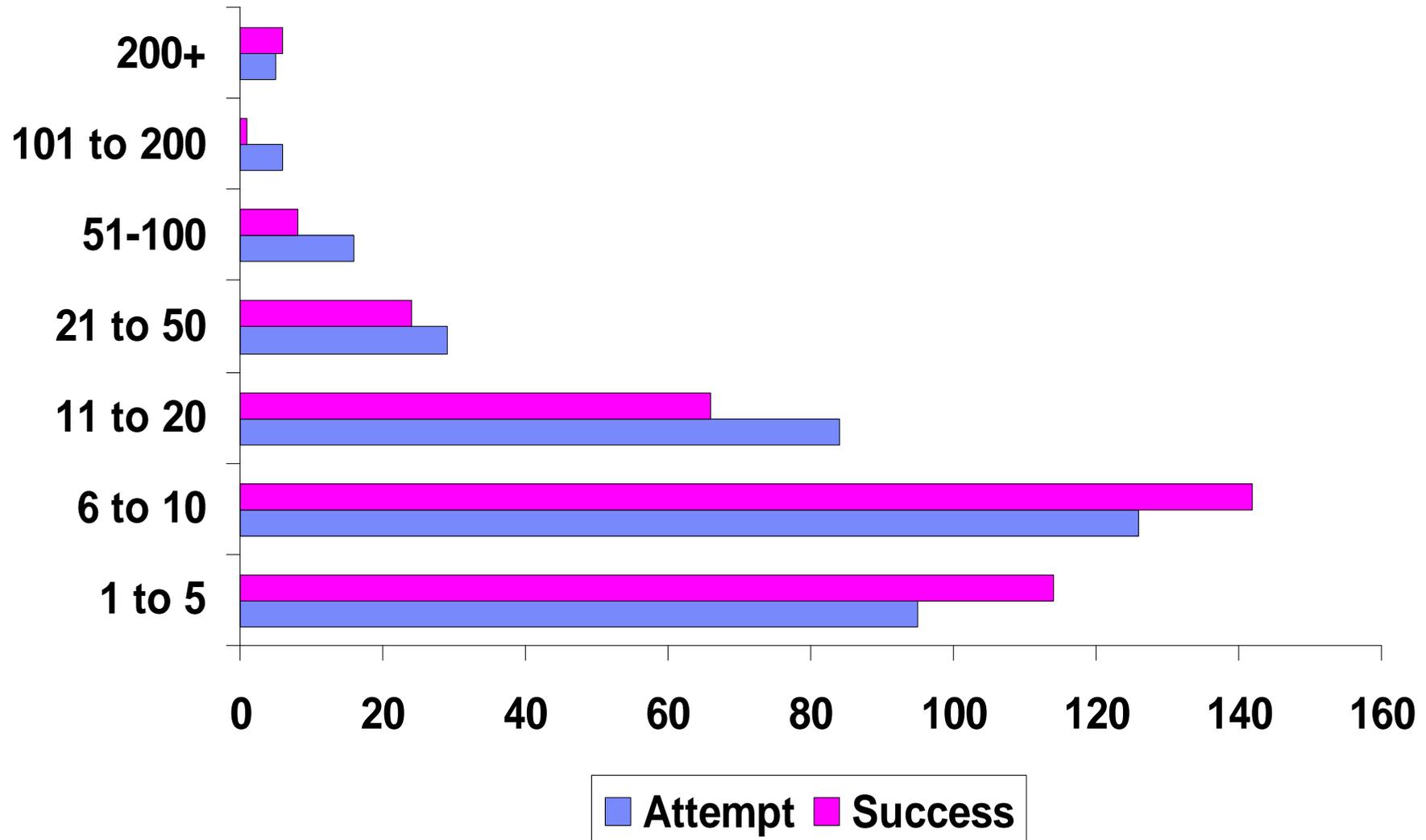
(214 co-located projects, 210 not co-located, 129 offshoring/outsourcing)



Source: Dr Dobb's 2008 Agile Adoption Survey



Largest Team Size Attempted vs. Successful

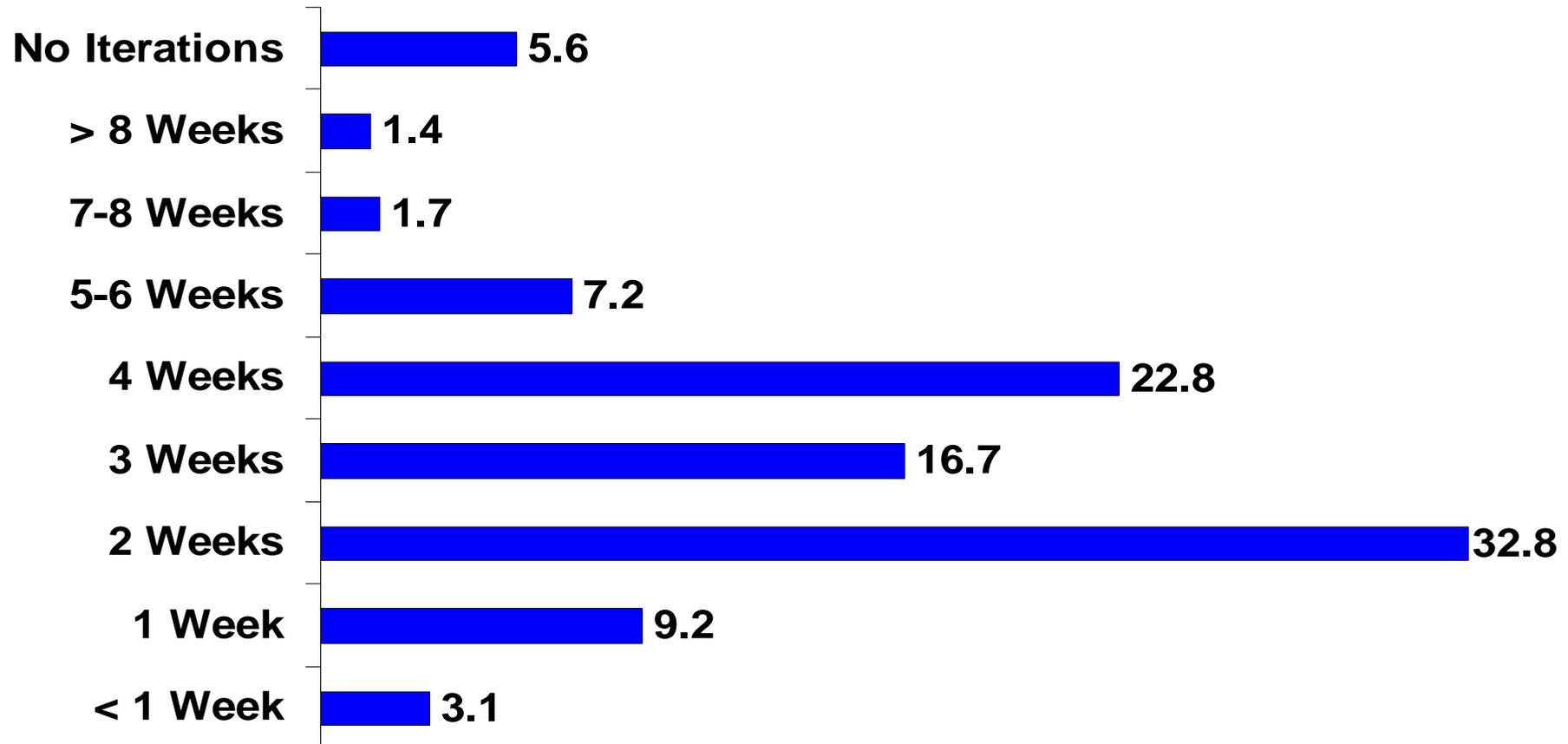


Source: Dr Dobb's 2008 Agile Adoption Survey



Length of Iterations (% respondents)

73% have iterations between 2 and 4 weeks in length



Source: Dr Dobb's 2008 Agile Adoption Survey



Discussion Topics

- Agile Adoption Rate
- Case Study: Large-Scale Agile Adoption
- Agile@Scale
- Practices for Scaling Agile
- A systematic approach to agile adoptions
- Parting Thoughts



IBM is Going Agile

One of the worlds largest agile transformations

- **Agile key to continue “lead the pack”**
 - ▶ Tomorrow’s leaders must adopt appropriate agile techniques
- **Large scale transformation**
 - ▶ ~35,000 developers
- **Very diverse development contexts**
 - ▶ From: New products, short time-to-market, Web 2.0
 - ▶ To: Mature products, risk reduction, older technologies
- **Agility at Scale is key**
 - ▶ Team size, geographical distribution, compliance, application complexity, ...



IBM is Going Agile

Sample data

- +2,000 developers trained in last 12 months
- +100 teams using IBM Rational Team Concert
- +1,000 members of Agile@IBM community
- Yearly IBM Academy of Technology Conference on Agile Development

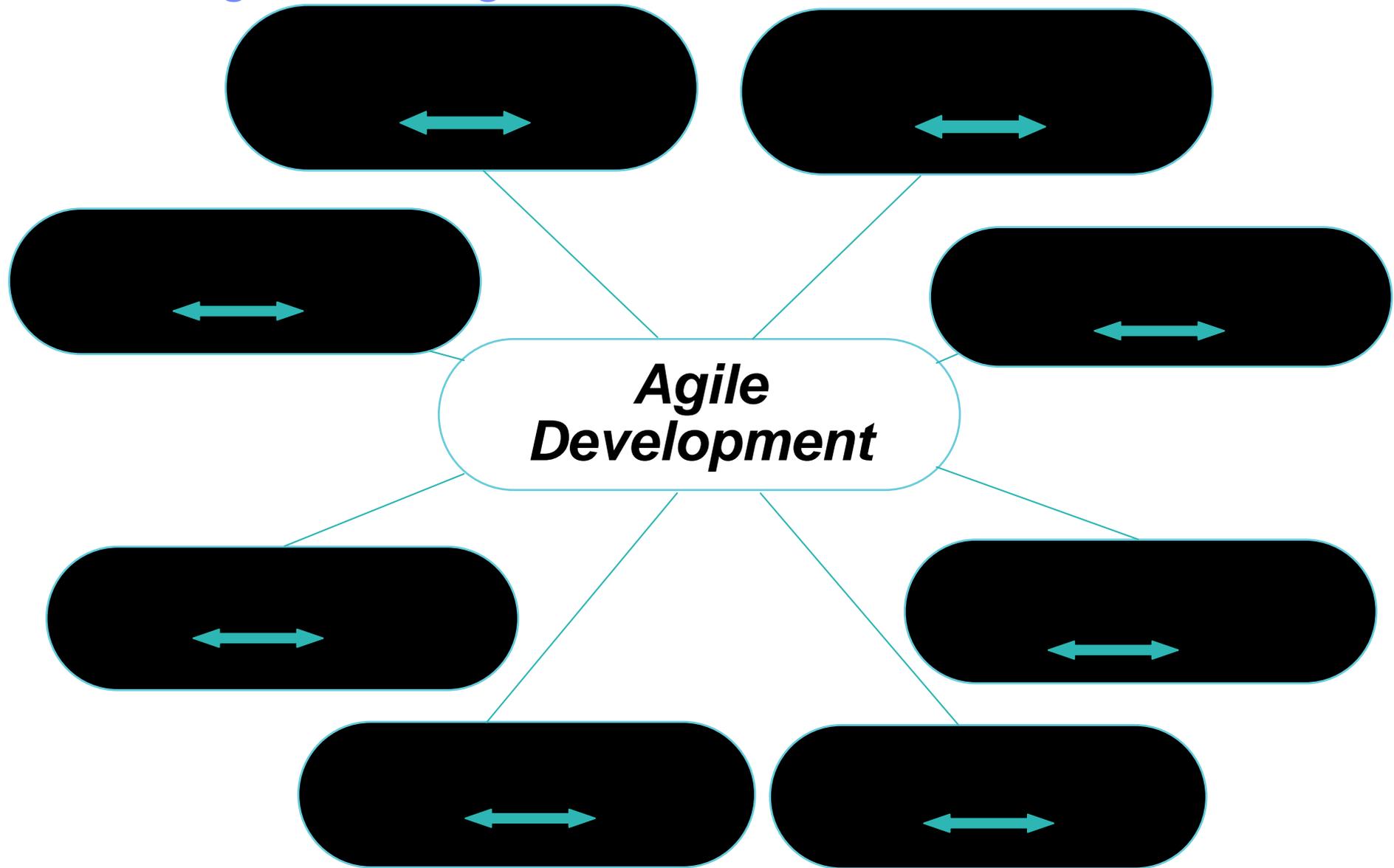


Discussion Topics

- Agile Adoption Rate
- Case Study: Large-Scale Agile Adoption
- Agile@Scale
- Practices for Scaling Agile
- A systematic approach to agile adoptions
- Parting Thoughts



Challenges with Agile in the Mainstream



Agility is Relative – It Depends on Project Dynamics

Organizational Drivers

Team Size
Geographical Distribution
Organization Distribution
Entrenched process, people, policy

- Mature or existing projects
- Many developers
- Complex, multi-platform applications
- Distributed teams
- Need for scalability, reproducibility, and traceability

- Maturing projects
- Multi-platform
- Growing in complexity
- Remote or offshore work
- Greater need for coordination and handoffs

- Small team
- New projects
- Simple application
- Co-located
- Minimal need for documentation

Technical and Regulatory Drivers

Compliance
Governance
Application complexity



Jazz Platform

Expanding Collaborative Application Lifecycle Management

Rational Team Concert

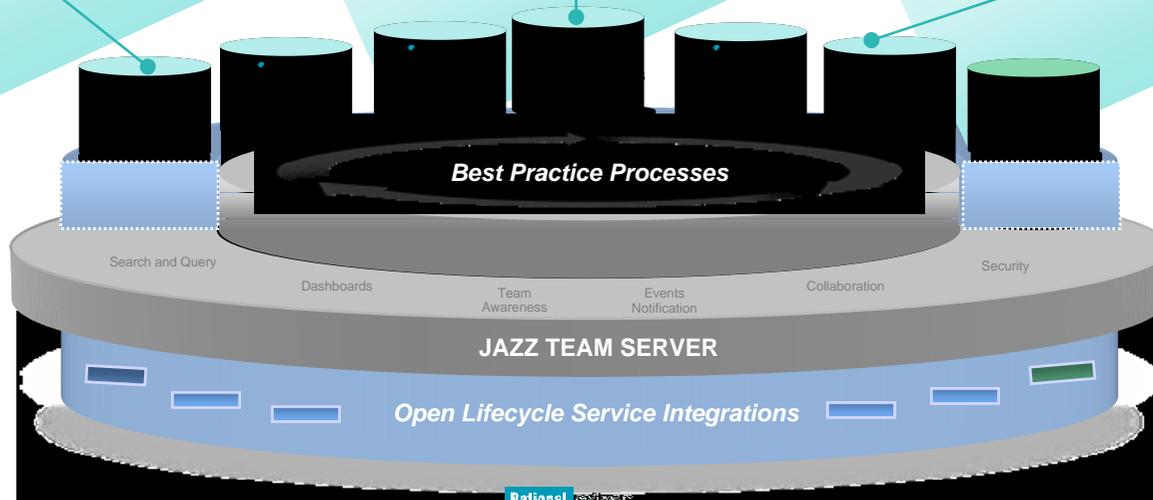
Innovation Through Collaboration
Unify by "thinking & working" in unison with real-time project health

Rational Requirements Composer

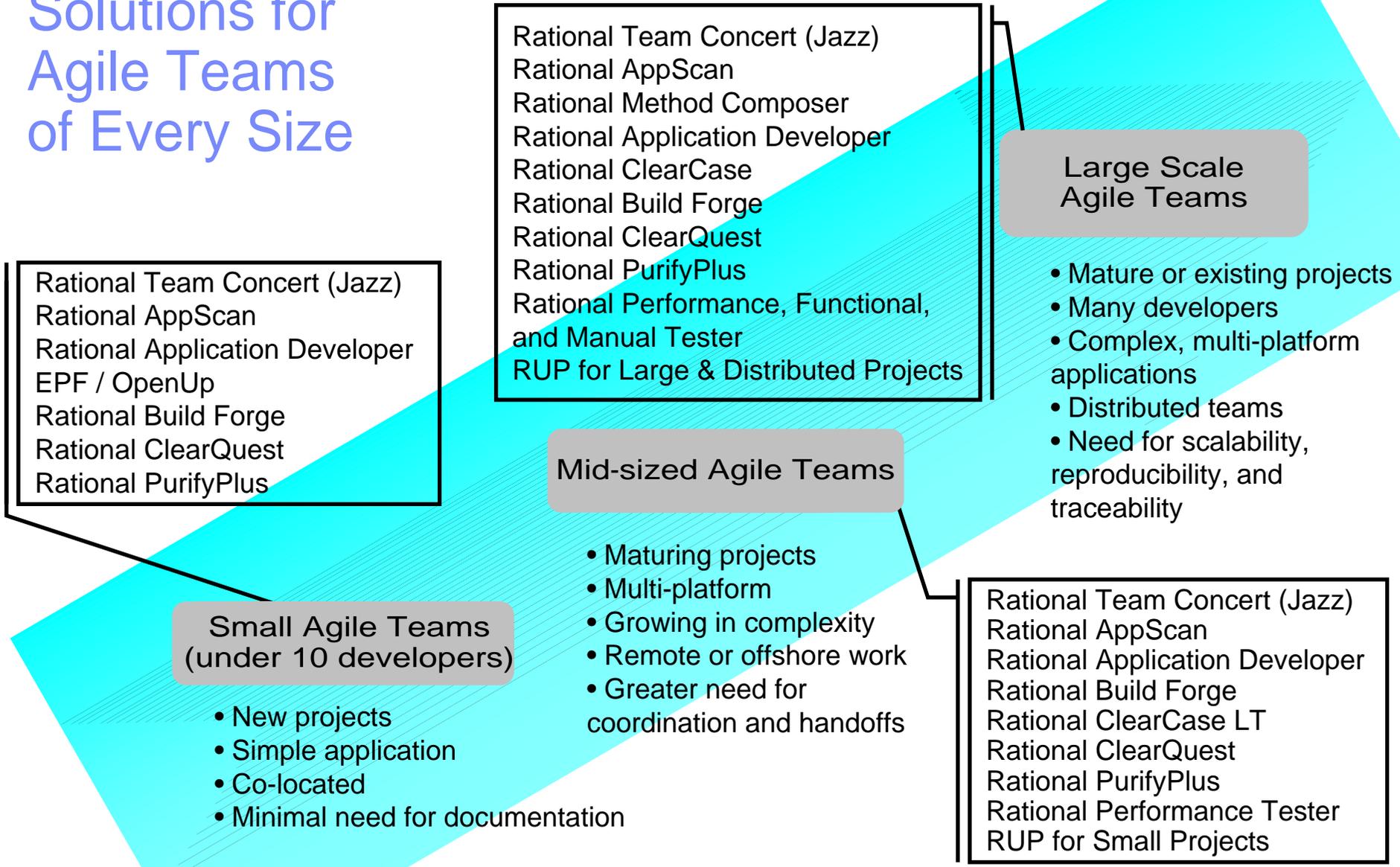
Business Expert Collaboration
Elicit, capture, elaborate, discuss and review requirements

Rational Quality Manager

Collaborative Business-driven Quality
Coordinate quality assurance plans, processes and resources



Solutions for Agile Teams of Every Size



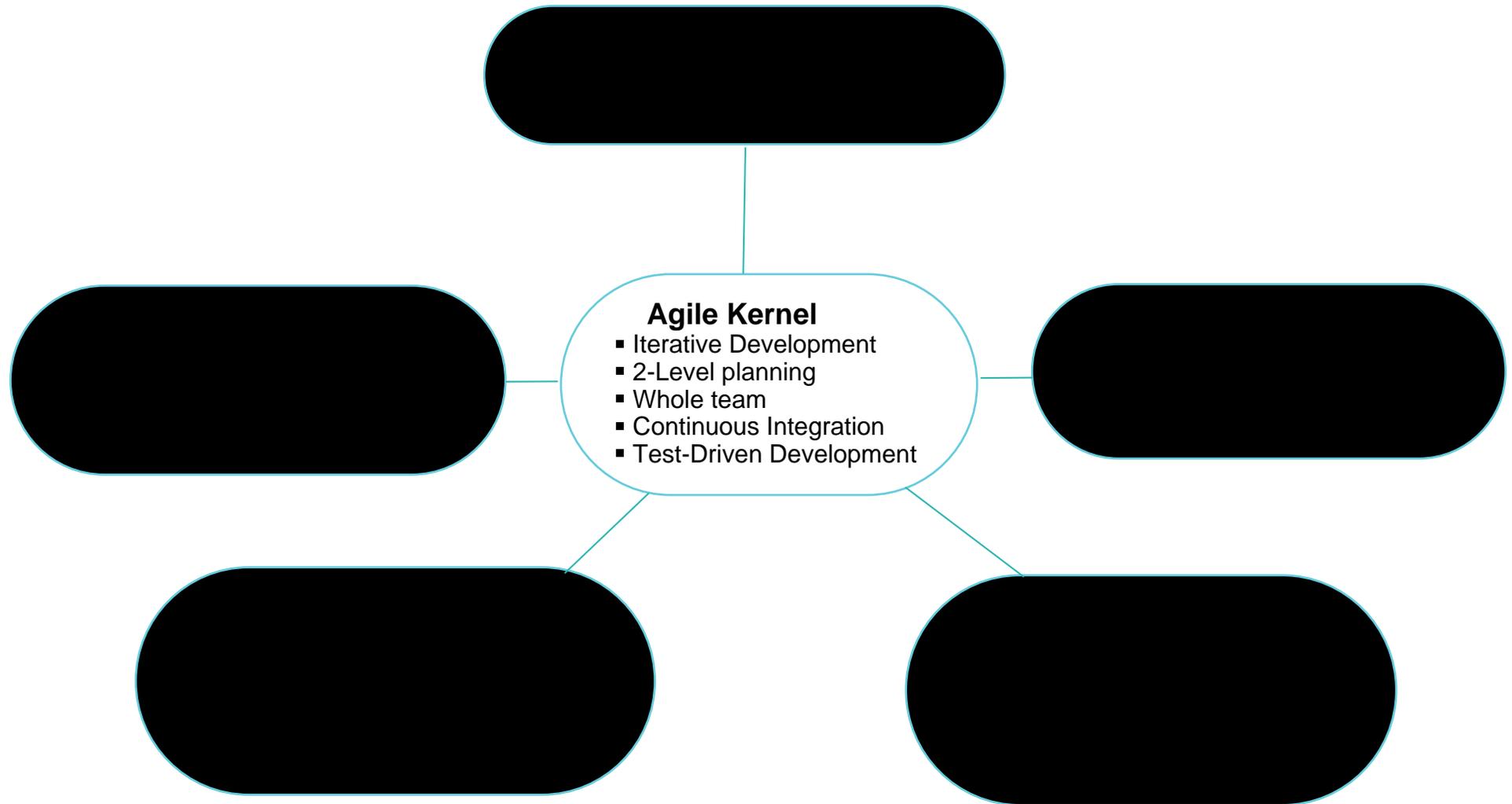
Discussion Topics

- Agile Adoption Rate
- Case Study: Large-Scale Agile Adoption
- Agile@Scale
- Practices for Scaling Agile
- A systematic approach to agile adoptions
- Parting Thoughts

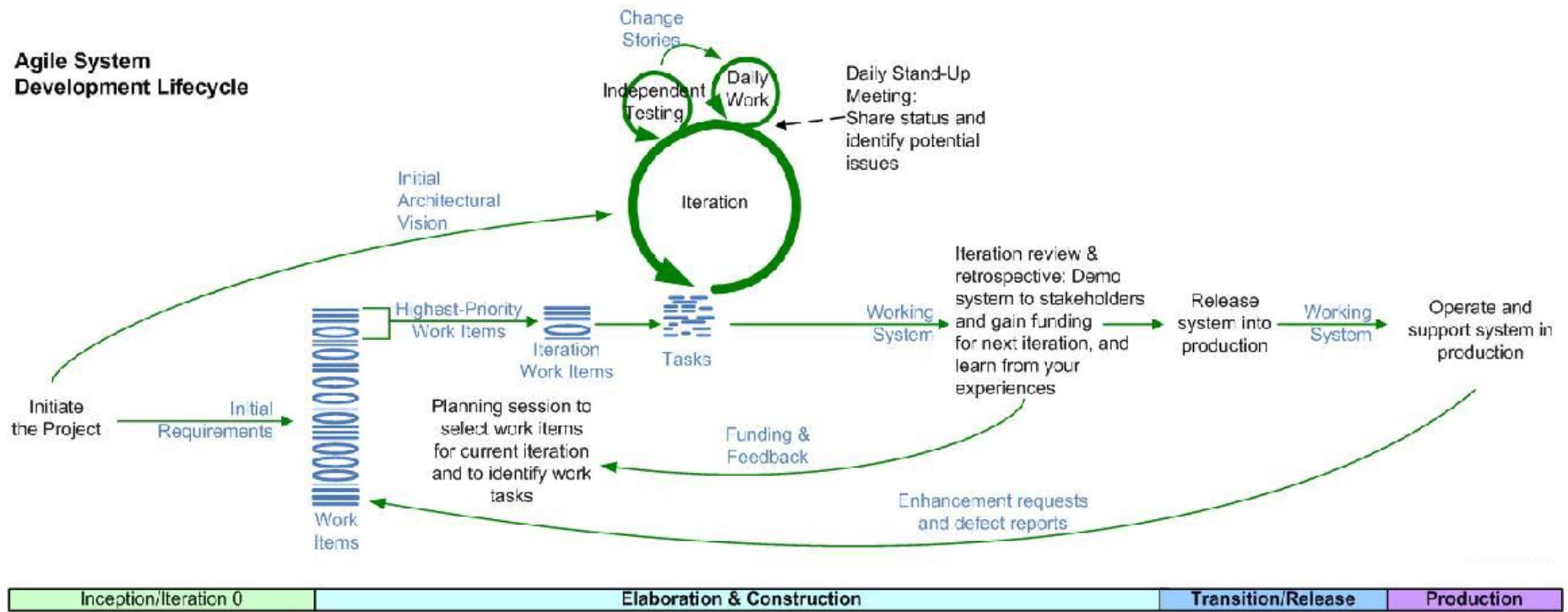


Planned Practices for RMC 7.5

Start Small - Add Practices When Needed to Scale



The Agile System Development Lifecycle (SDLC)



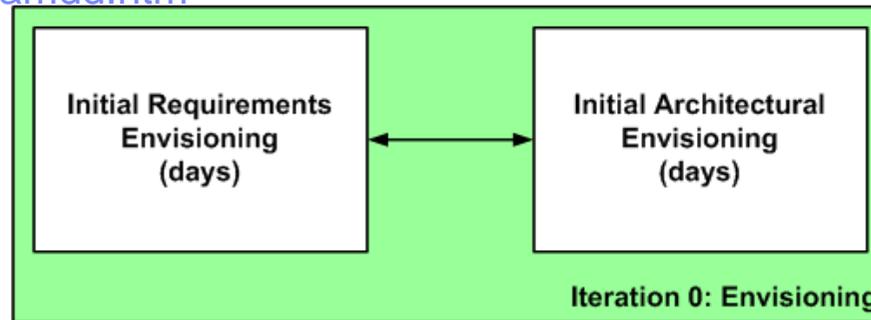
Practices: Iterative Development, 2-level Planning, Risk-Value Lifecycle



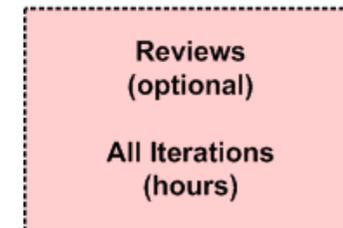
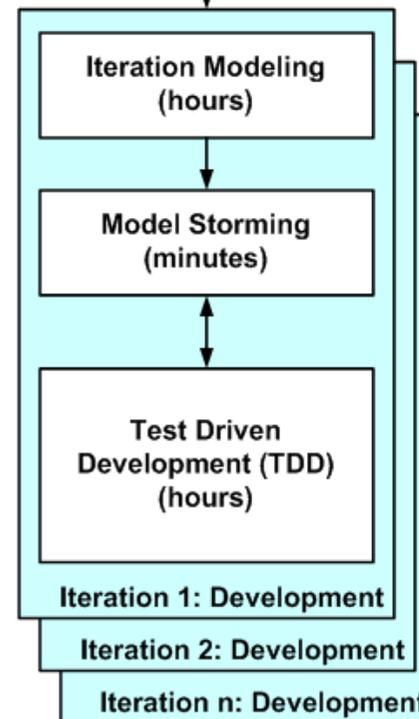
Scaling TDD: Agile Model Driven Development (AMDD)

www.agilemodeling.com/essays/amdd.htm

- Identify the high-level scope
- Identify initial "requirements stack"
- Identify an architectural vision



- Modeling is part of iteration planning effort
- Need to model enough to give good estimates
- Need to plan the work for the iteration
- Work through specific issues on a JIT manner
- Stakeholders actively participate
- Requirements evolve throughout project
- Model just enough for now, you can always come back later
- Develop working software via a test-first approach
- Details captured in the form of executable specifications



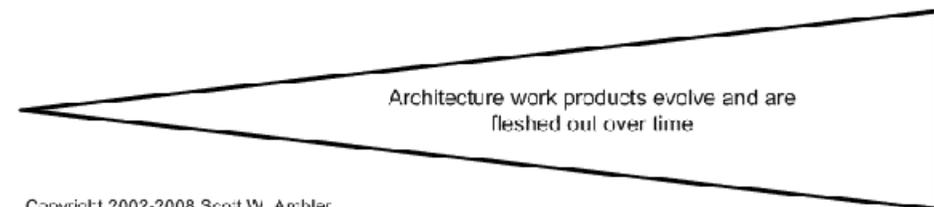
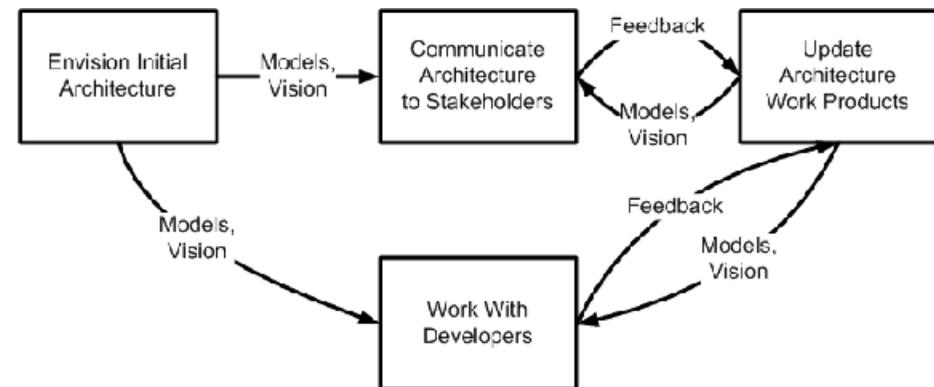
Practices: Evolutionary Architecture, Evolutionary Design, Component-Based Software Architecture

Copyright 2003-2007
Scott W. Ambler



Enterprise/Program Architecture

- Provide technical vision to the enterprise
- Promote reuse and common infrastructure
- Develop reference architectures
- Develop guidance
- Work closely with development teams
- www.agiledata.org/essays/enterpriseArchitecture.html



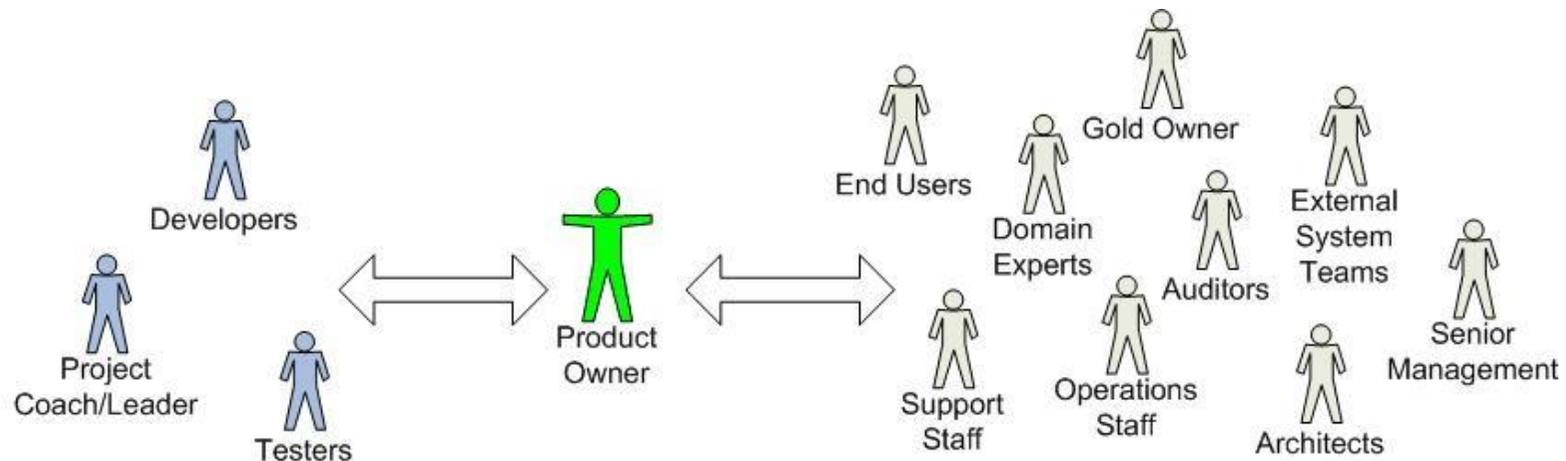
Copyright 2002-2008 Scott W. Ambler

Practices: Evolutionary Architecture,
Evolutionary Design, Component-
Based Software Architecture



Scaling XP's On-Site Customer and Scrum's Product Owner

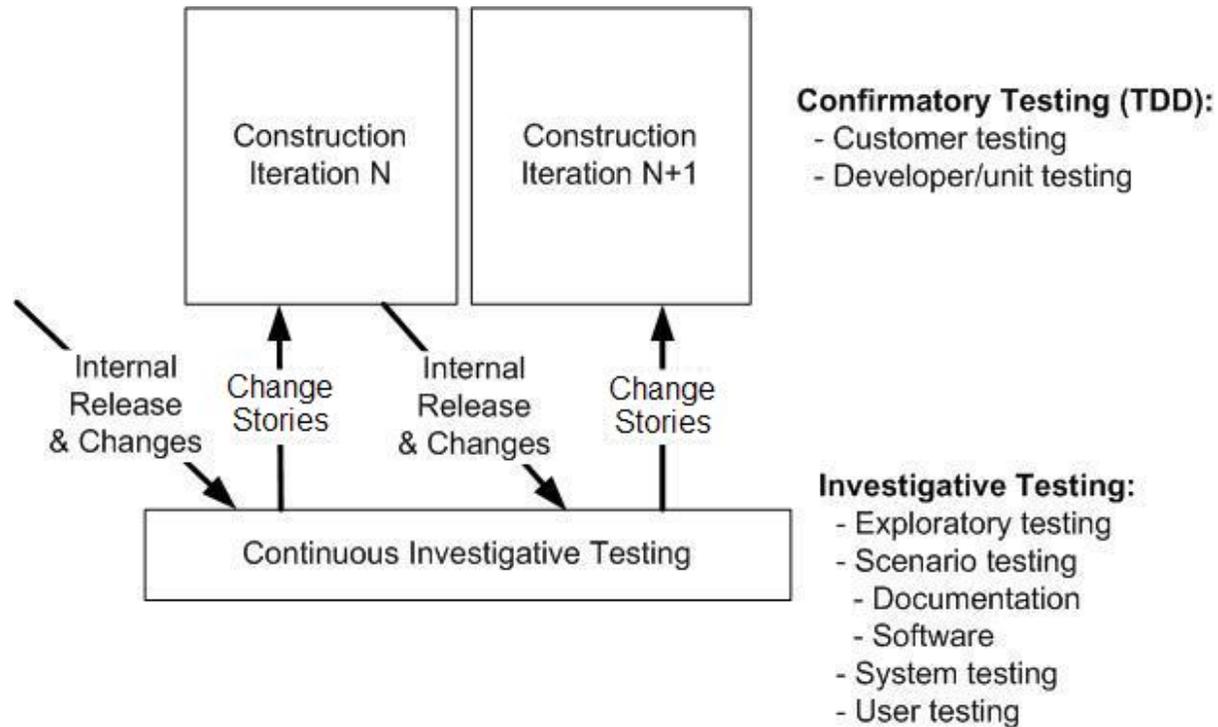
- On-site customer is nice, so put them to work
 - ▶ Stakeholders can be active participants in modeling
- Product owner is really a communication conduit between the team and stakeholders
 - ▶ Must have agile business analysis skills
 - ▶ PO gets the team access to the relevant stakeholders just in time
 - ▶ Negotiate, negotiate, negotiate
- Dr. Dobb's Journal, January 2008



Practice: Whole Team



Scaling TDD: Comprehensive Agile Testing



TDD is a form of confirmatory testing

TDD is a great start, but it's not the full testing picture

Effective agile teams push their working builds to an independent test team on a regular basis for investigative testing

Change stories must be prioritized and put back on the team's work stack

Defects == Requirements

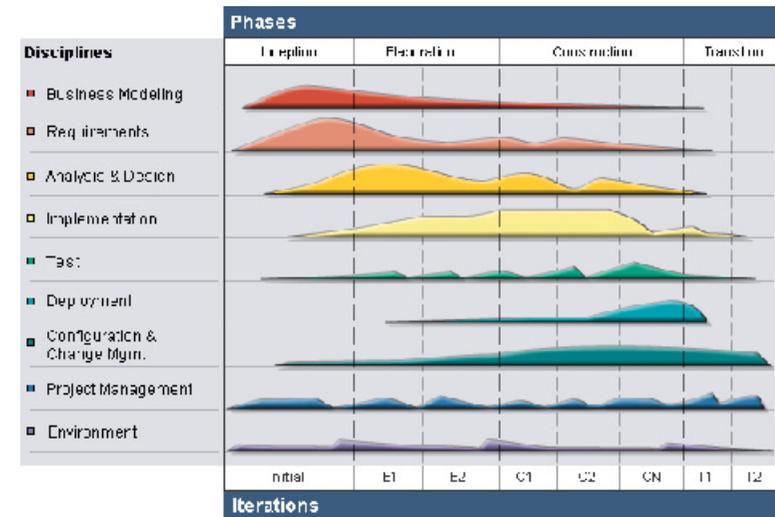
Practices: Test-driven Development, Concurrent Testing, Test Management, Vulnerability Testing, Performance Testing

Source: January 2007 Dr. Dobb's Magazine

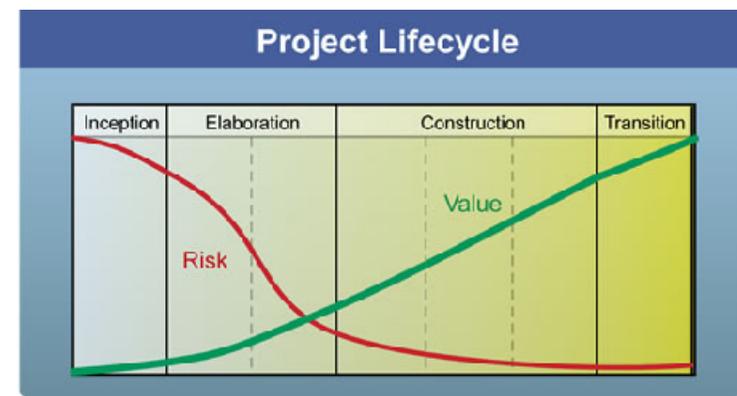


Scale Agile via Rational Unified Process (RUP)

- Organizations have instantiated RUP to be very agile
- Scaling strengths:
 - ▶ Risk-driven milestones
 - ▶ Explicit “go/no-go” decision points
 - ▶ Stakeholder concurrence gained during Inception
 - ▶ Architecture proven via working software during Elaboration
 - ▶ Managed deployment during Transition

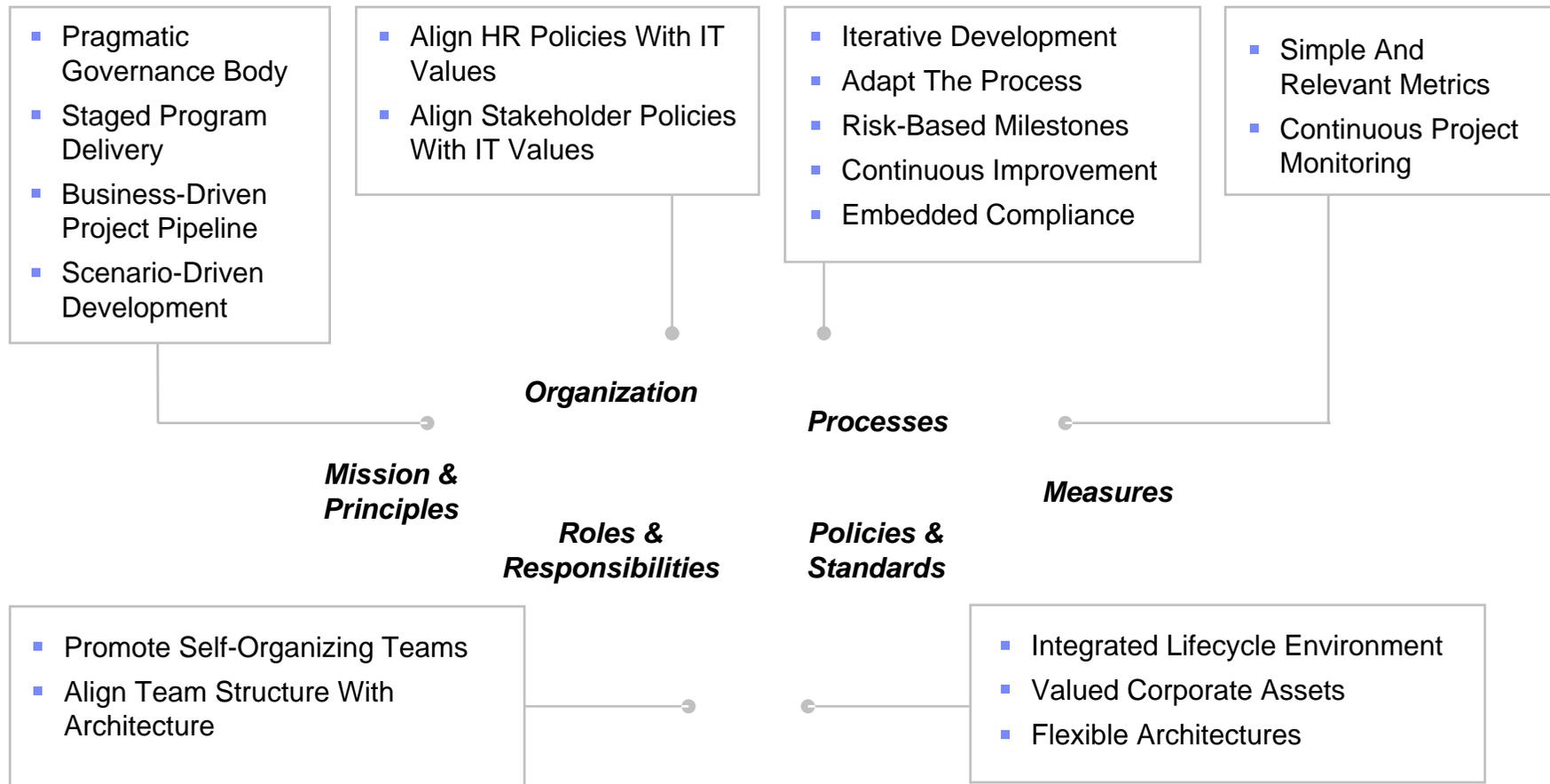


Practices: Iterative Development, 2-level Planning, Risk-Value Lifecycle



Lean Development Governance

www.ibm.com/developerworks/



Discussion Topics

- Agile Adoption Rate
- Case Study: Large-Scale Agile Adoption
- Agile@Scale
- Practices for Scaling Agile
- A systematic approach to agile adoptions
- Parting Thoughts



Measured capability improvement framework

Map business value to software delivery practices

Target: Phase 1
 Already implemented
 Outside scope

Example: Financial Service Company

Customer Business Challenges

- Create financial products more quickly
- Functionality of customer web falling behind competition
- Inconsistencies with integrated financial reporting
- Recent SOX audit failure

Operational Objectives

- Reduce time-to-market
- Improve productivity
- Increase innovation
- Improve consistency/predictability
- Improve oversight
- Enable flexible/global resourcing
- Satisfy compliance mandate

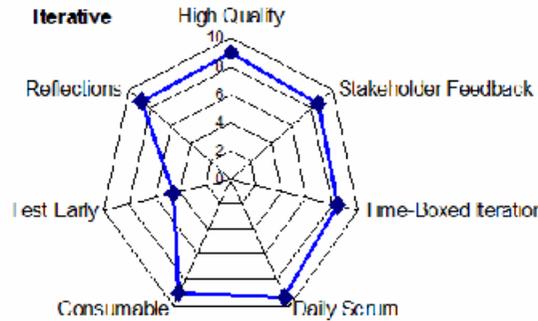
Software Delivery Best Practices

- Use-case driven development
- Continuous integration
- Shared vision
- Whole team
- Risk-value lifecycle
- 2-level project planning
- Test-driven development
- Asset-based development
- Asset governance
- Iterative development
- SOA modeling

Business Metrics

Project	Time to Market (M)	Quality (Defect Density)	Innovation (Cust. Sat.)
A	22	2.3	7
B	14	1.4	4
C	18	1.6	6
D	9	0.3	10
E	6	0.4	8

Ongoing Adoption Assessment



Adopt Practice



Service Offering: Health Assessment for Software Delivery

Coming Soon!



- Identify drivers for delivery improvements
- Current state: Showcase practice adoption level for target projects
- Future state: An initial roadmap outlining how you can evolve your delivery approach to address your key business drivers
- List of inhibitors that may prevent successful adoption of target practices
- Identifies potential metrics that help you track progress on the implementation of your roadmap

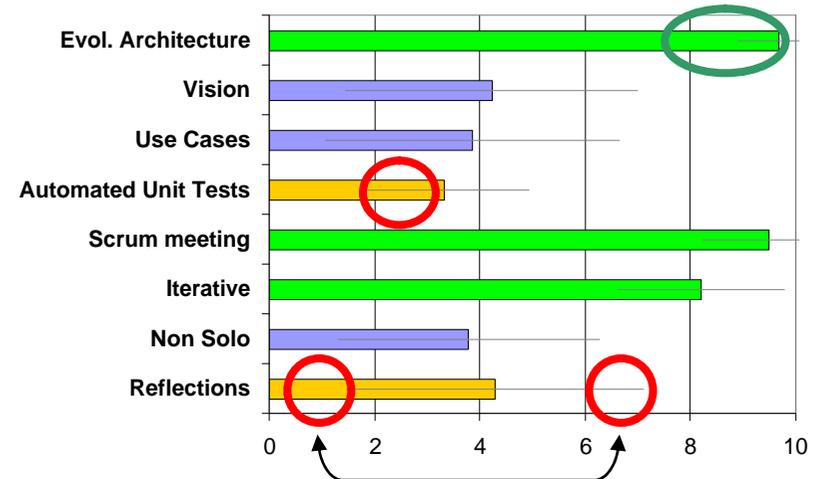


IBM Rational Self Check for Software Teams

The Assessment Tool

Practice	Comments	Average	Deviation
Use Cases	Use Stories instead	7.8	0.50
Unit Test	Not enough, not in build	6.8	1.89
Stakeholder Feedback	Want more	8.5	0.58
Time-Boxed Iterations	Not well specified	8.3	0.50
Daily Scrum	Not Fridays	9.3	0.50
Test Early	Not enough TDD	4.5	1.91
Reflections	Need clear actions	8.8	0.50
Static Analysis	Not automated yet.	3.3	0.96

Big Picture of Practice Adoption



Experience Report

Executive Summary:
The Rational Team Manager project aimed to find an efficient channel of work and release strategies for software to learn from including product-based development, customer-facing, and team self-reflective software to facilitate and.

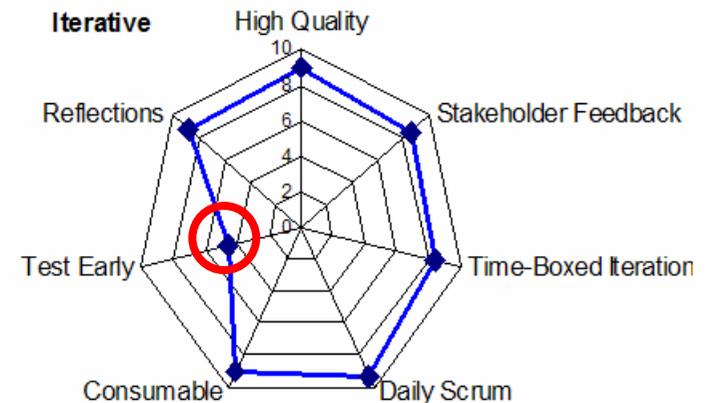
Project Context:

Context Factor	Value	Comments
Project Start	2014-01-01	Project start date
Project Length	3 Months	Project duration
Team	10-12 people (5-6 dev, 3-4 QA, 2-3 PM)	Team composition
Iterations	3 iterations (2 weeks each)	Iteration cycle
Code requirements	None	Code quality requirements
Programming languages	Java, JavaScript, CSS	Technology stack

Comments:
Rational Team Manager was used for 30 days, primarily for project management, with some use for team self-reflective software to facilitate and.

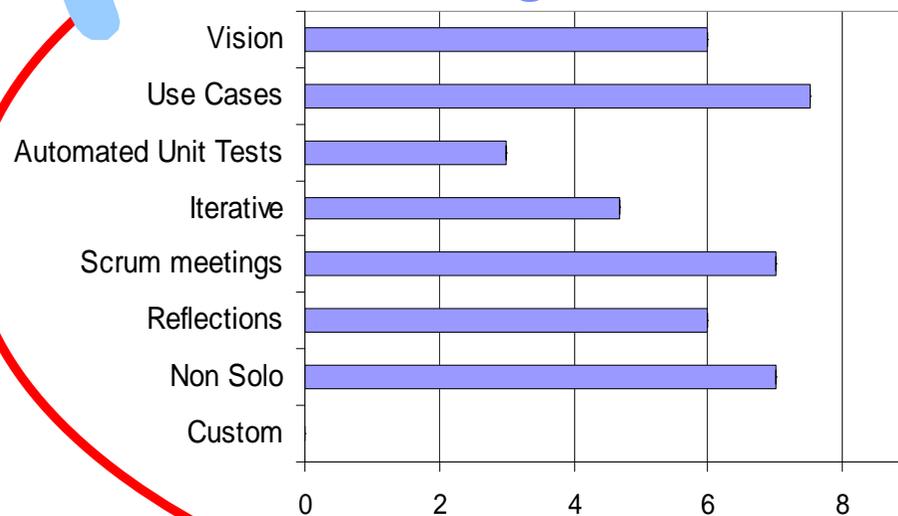
Use of Practices: (Scale of 0 - 10) 10 = 100% usage

Deep Dive on Practice

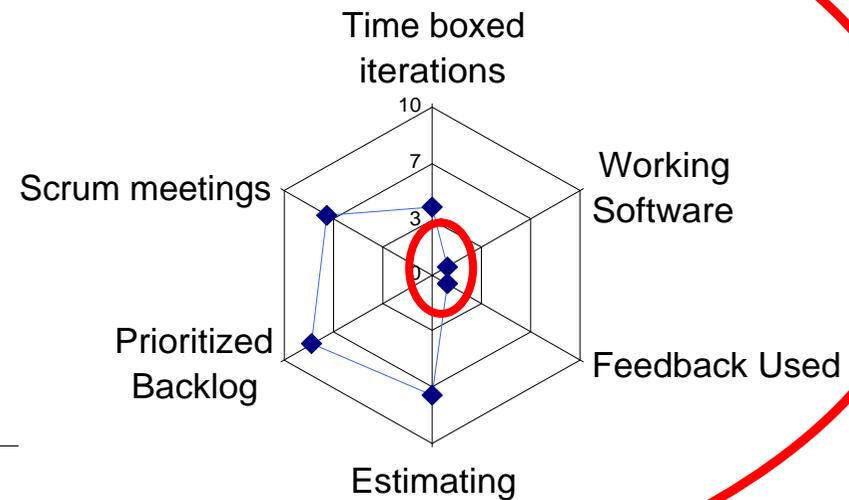


Example: Are we iterative?

The Big Picture



We're iterative, right?



235 people. Multi-shore. Java

8 week "iterations". 24 Month project. New to agile.

Iterations have enabled them to realize the plan was severely overbooked and prioritization of content began early.



Discussion Topics

- Agile Adoption Rate
- Case Study: Large-Scale Agile Adoption
- Agile@Scale
- Practices for Scaling Agile
- A systematic approach to agile adoptions
- Parting Thoughts



A Call To Action

- **Look beyond the Agile rhetoric**
 - ▶ Are you being helped, or being sold consultantware?
- **Agile can scale to accommodate technical and organizational complexity**
 - ▶ Agile@Scale has unique needs
 - ▶ IBM Rational brings a wealth of knowledge to help companies succeed in complex environments
- **Get some Agile mentoring and training**
 - ▶ Project management training is critical
 - ▶ Training modelers, developers, ... is also critical
 - ▶ Use MCIF as a systematic approach to improve
- **Adopt a scalable platform and tools that support agile development**
 - ▶ Jazz platform, IBM Rational Team Concert, IBM Rational Requirements Composer, IBM Rational Quality Manager, BuildForge, ...
- **Adopt an Agile Form of RUP and associated practices**
 - ▶ Extend traditional agile practices with practices for control mechanisms and risk-mitigation strategies



Critical IBM Agile Resources

www.ibm.com/rational/agile/

www.ibm.com/developerworks/

www.ibm.com/developerworks/blogs/page/ambler





QUESTIONS





THANK YOU

Learn more at:

- [IBM Rational software](#)
- [IBM Rational Software Delivery Platform](#)
- [Process and portfolio management](#)
- [Change and release management](#)
- [Quality management](#)
- [Architecture management](#)
- [Rational trial downloads](#)
- [Leading Innovation Web site](#)
- [developerWorks Rational](#)
- [IBM Rational TV](#)
- [IBM Rational Business Partners](#)

© Copyright IBM Corporation 2008. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, the on-demand business logo, Rational, the Rational logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.

