

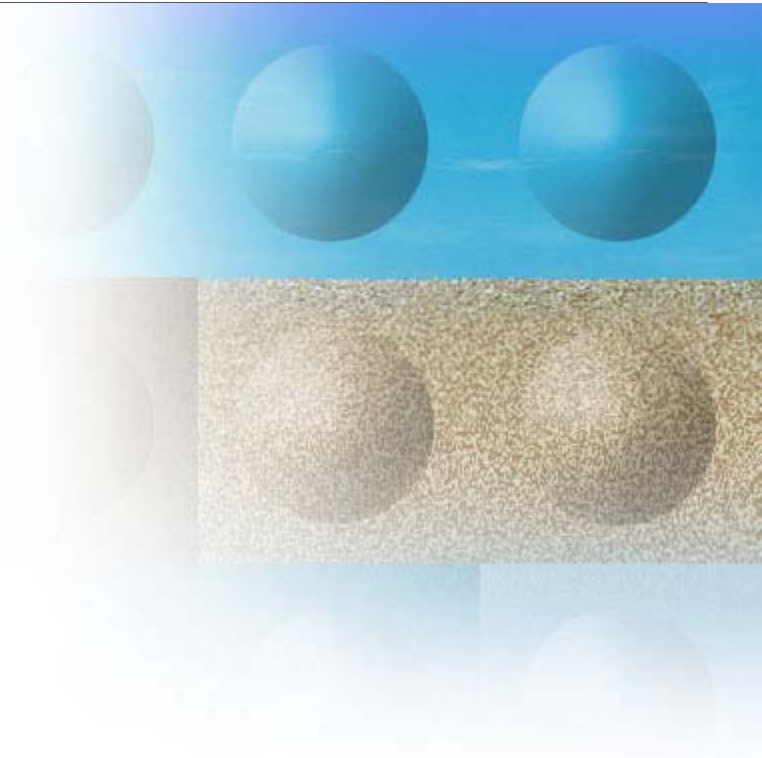
Agility @ Scale in the Real World

Davyd Norris
Senior IT Architect – IBM Rational
dnorris@au1.ibm.com



Agenda

- Agile Software Development
- Agile Adoption Rate
- Scaling Agile
- Successful Adoption



What is Agile?

- An iterative and incremental (evolutionary) approach performed in a highly collaborative and self-organizing manner with just the right amount of ceremony to frequently produce high quality software in a cost effective and timely manner which meets the changing needs of its stakeholders.

- Core principles
 - “Fits just right” process
 - Continuous testing and validation
 - Consistent team collaboration
 - Rapid response to change
 - Ongoing customer involvement
 - Frequent delivery of working software



How Agile is Different

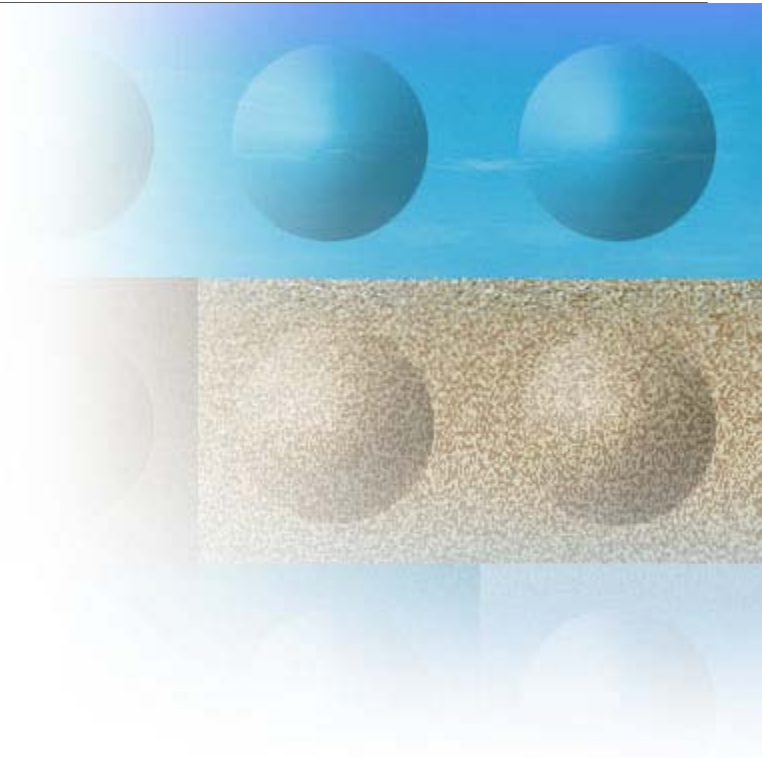
- Focus on collaboration:
 - Less paperwork and more conversation
 - Stakeholders actively involved
- Focus on quality
 - We have a full regression test suite for our systems
 - We develop loosely-coupled, highly cohesive architectures
 - We refactor to keep them this way
- Focus on working software:
 - Greater feedback makes agile projects easier to manage
 - Less documentation is required
 - Less bureaucracy
- Agilists are generalizing specialists:
 - Less hand offs between people
 - Less people required
 - Specialists find it difficult at first to fit into the team
- Agile is based on practice, not theory:
 - This is a significant change from traditional
 - You need to see how agile works in practice to truly understand it

Addressing Misconceptions about Agile

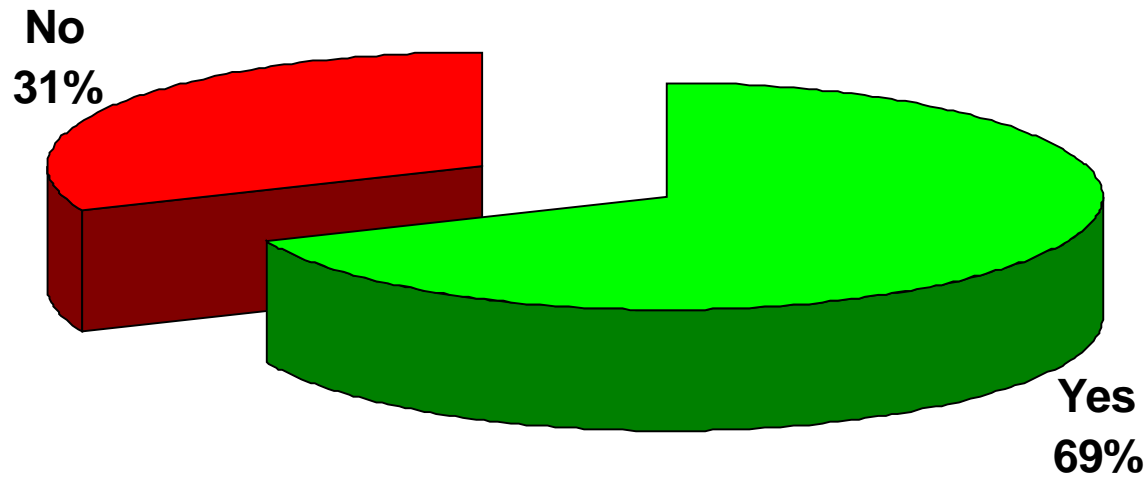
1. Agile teams write documentation
2. Agile teams model
3. Agile requires greater discipline than traditional approaches
4. Agile teams do more planning than traditional teams, but it's just in time (JIT)
5. Agile is more predictable than traditional
6. Agile scales very well
7. RUP can be as agile as you want to make it
8. Agile is not a fad, it is being adopted by the majority of organizations
9. Agile can do fixed price, but it's still poor practice to do so

Agenda

- Agile Software Development
- Agile Adoption Rate
- Scaling Agile
- Successful Adoption



Has Your Organisation Adopted One or More Agile Techniques?



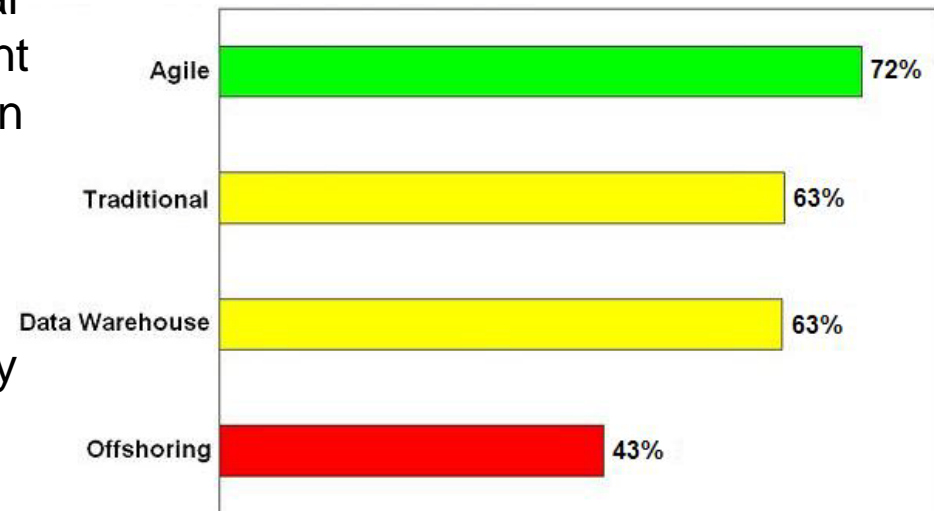
18% of respondents indicated they're still in the pilot stage

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

Source: Dr Dobb's 2008 Agile Adoption Survey

Why Agile/Lean? It's More Successful

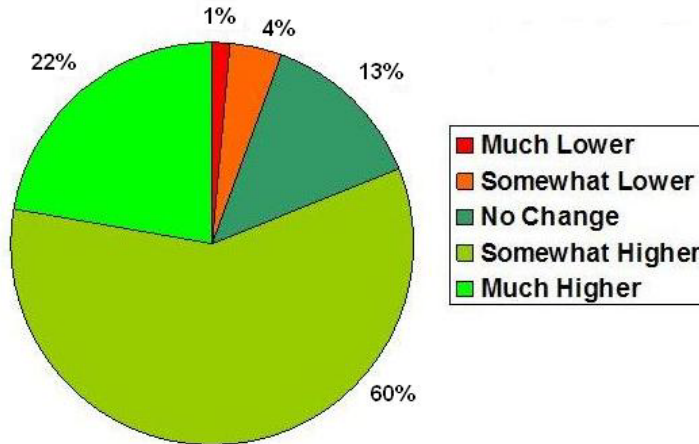
- Quality: 87% believe that delivering high quality is more important than delivering on time and on budget
- Scope: 87% believe that meeting actual needs of stakeholders is more important than building the system to specification
- Money: 80% believe that providing the best ROI is more important than delivering under budget
- Staff: 76% believe that having a healthy workplace is more important than delivering on time and on budget
- Schedule: 62% believe that delivering when the system is ready to be shipped is more important than delivering on schedule



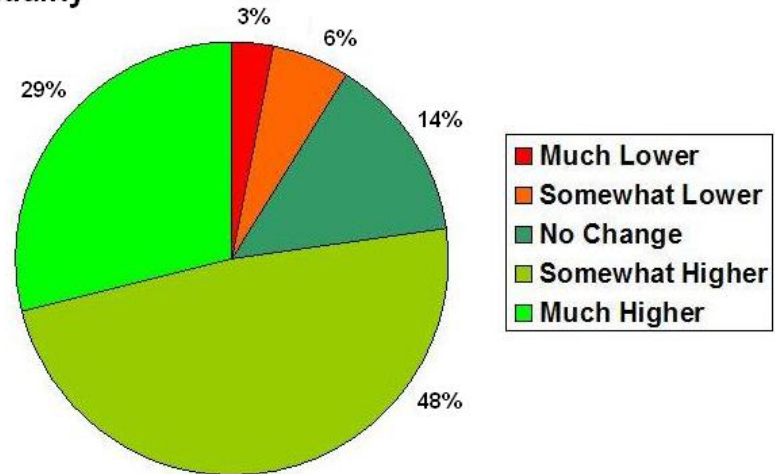
Source: Dr Dobb's 2007 Project Success Survey

Why Agile? Because it Works!

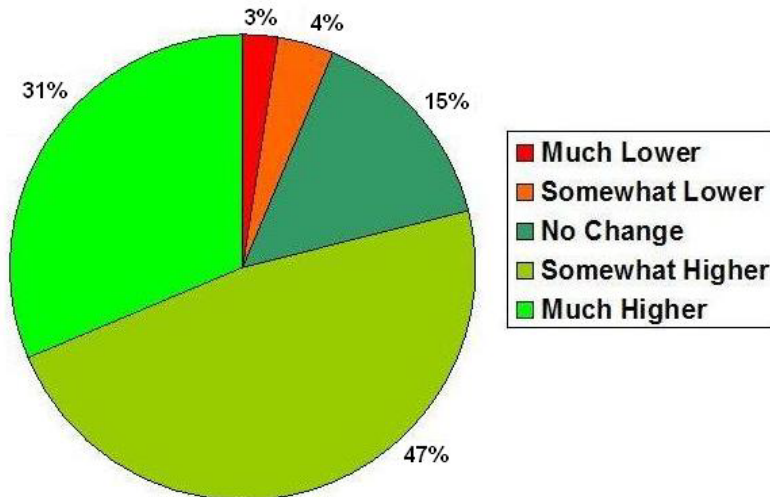
Productivity



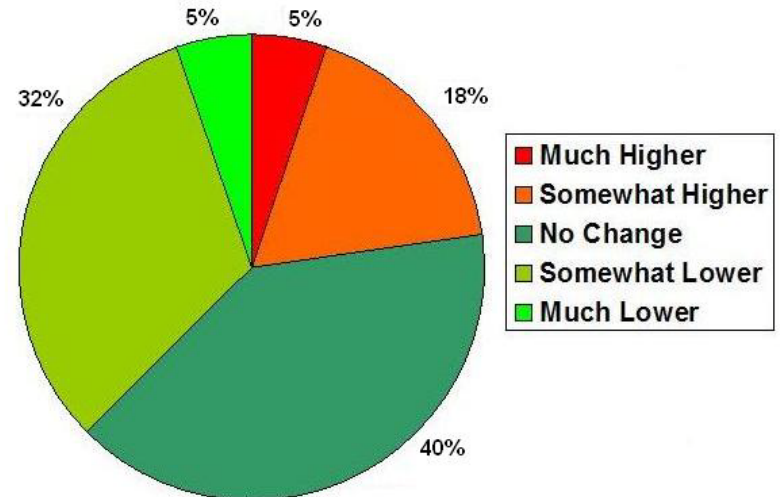
Quality



Business Stakeholder Satisfaction



Cost of System Development

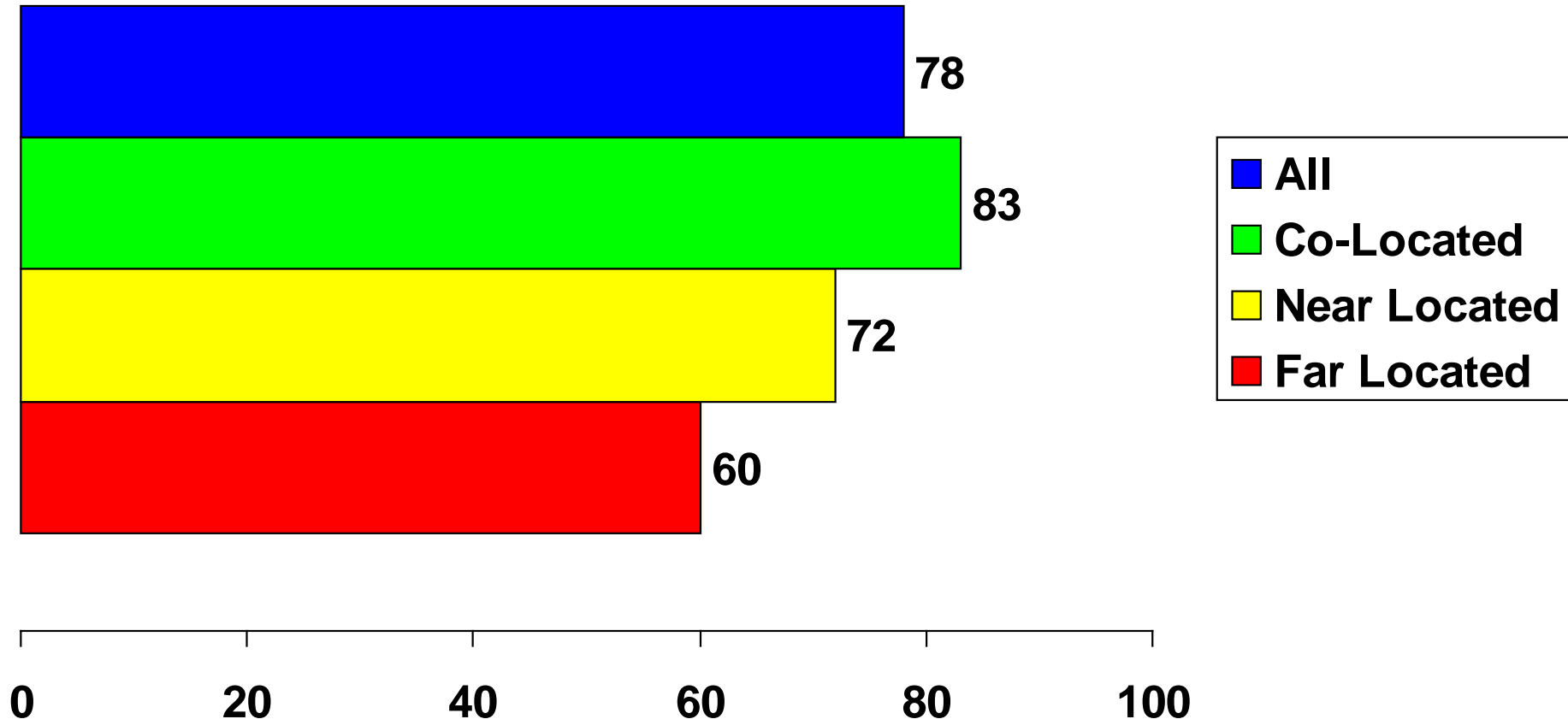


Source: Dr. Dobb's Journal 2008 Agile Adoption Survey

Copyright 2008 Scott W. Ambler

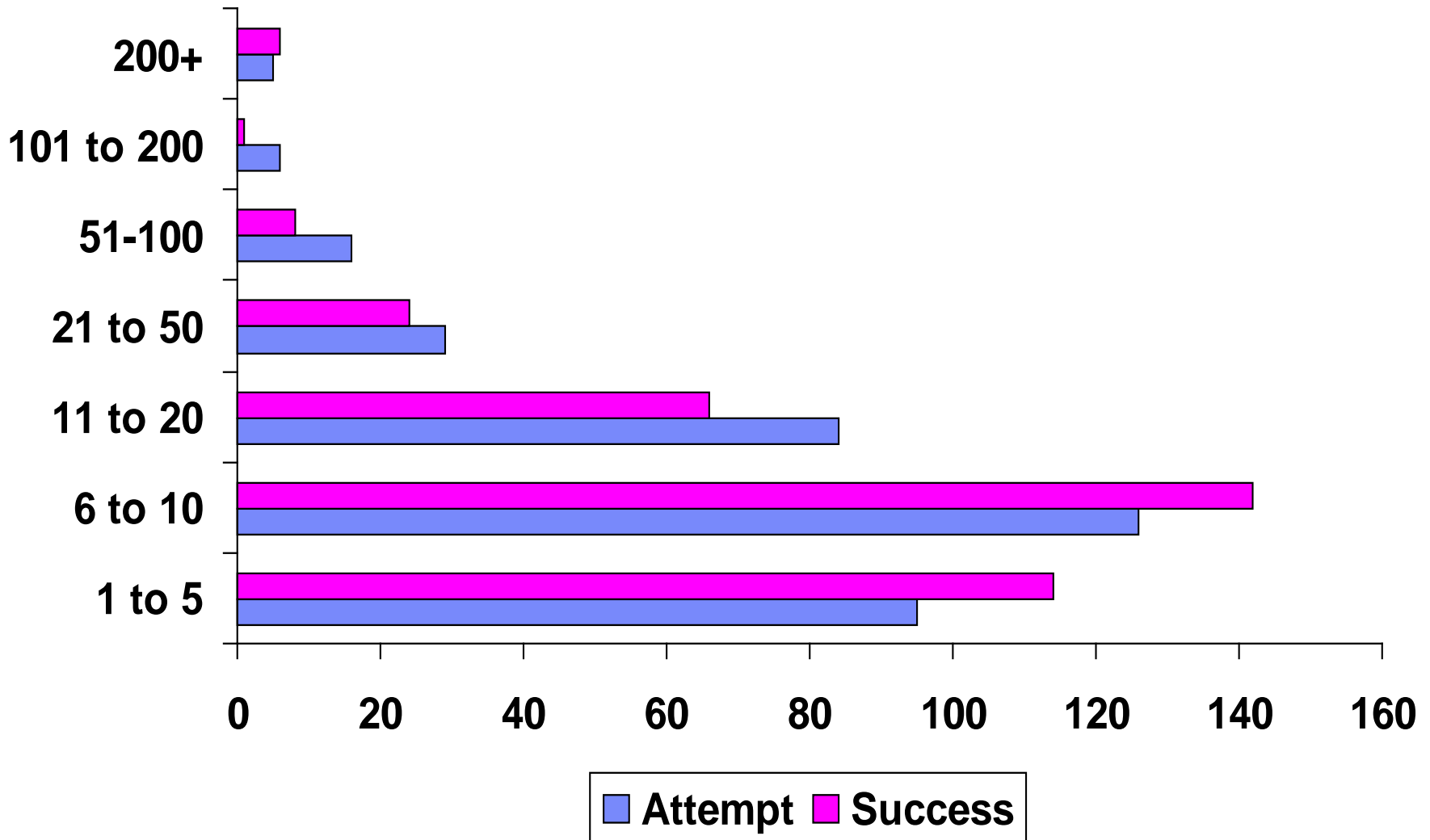
Agile Projects Success Rates (%)

(214 co-located projects, 210 near located, 129 far located)



Source: Dr Dobb's 2008 Agile Adoption Survey

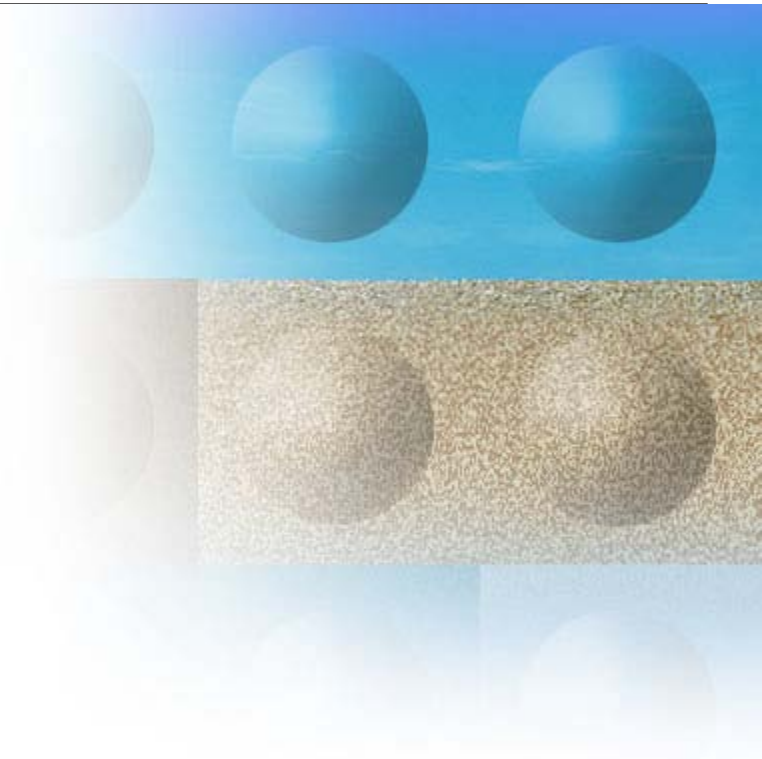
Largest Team Size Attempted vs. Successful



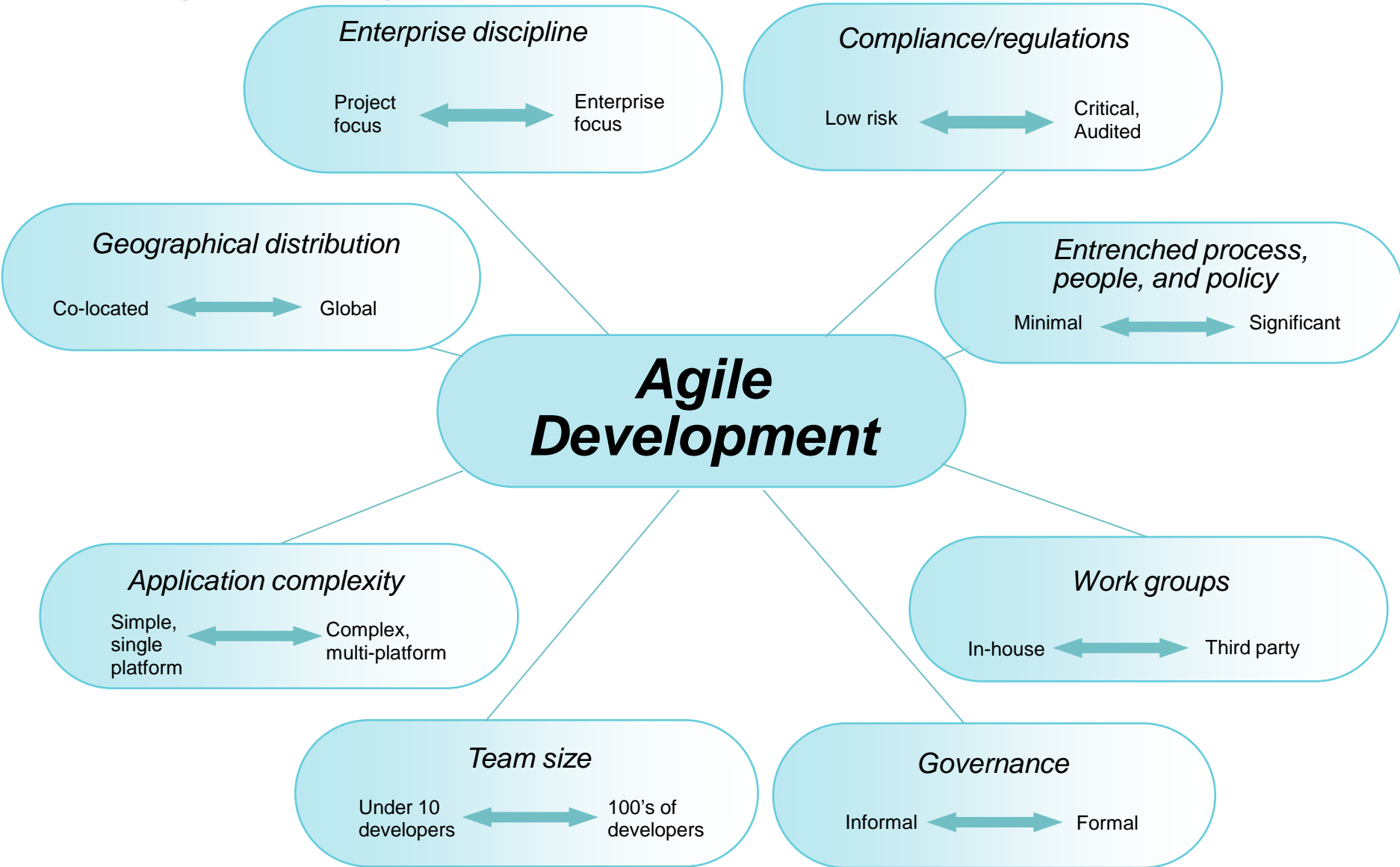
Source: Dr Dobb's 2008 Agile Adoption Survey

Agenda

- Agile Software Development
- Agile Adoption Rate
- **Scaling Agile**
 - Challenges with Agile in the Mainstream
 - Agility is Relative
 - AMDD Practices
 - Agile Testing
 - RUP Agile
 - Agile Data Practices
- Successful Adoption



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

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

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

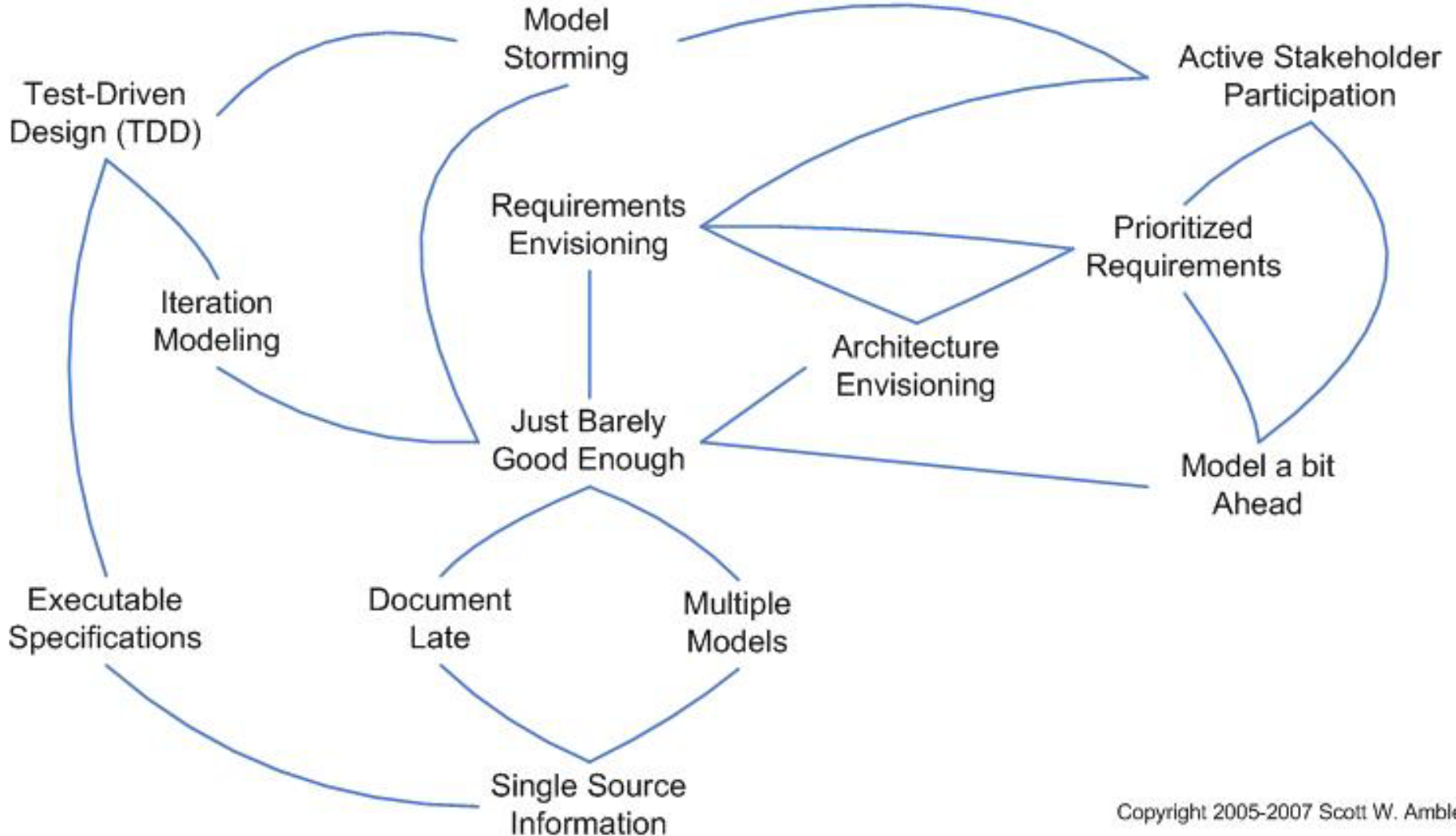
- Mature or existing projects
- Large teams
- Complex, multi-platform applications
- Distributed teams
- Need for scalability, reproducibility, and traceability

Technical and Regulatory Drivers

Compliance
Governance
Application complexity

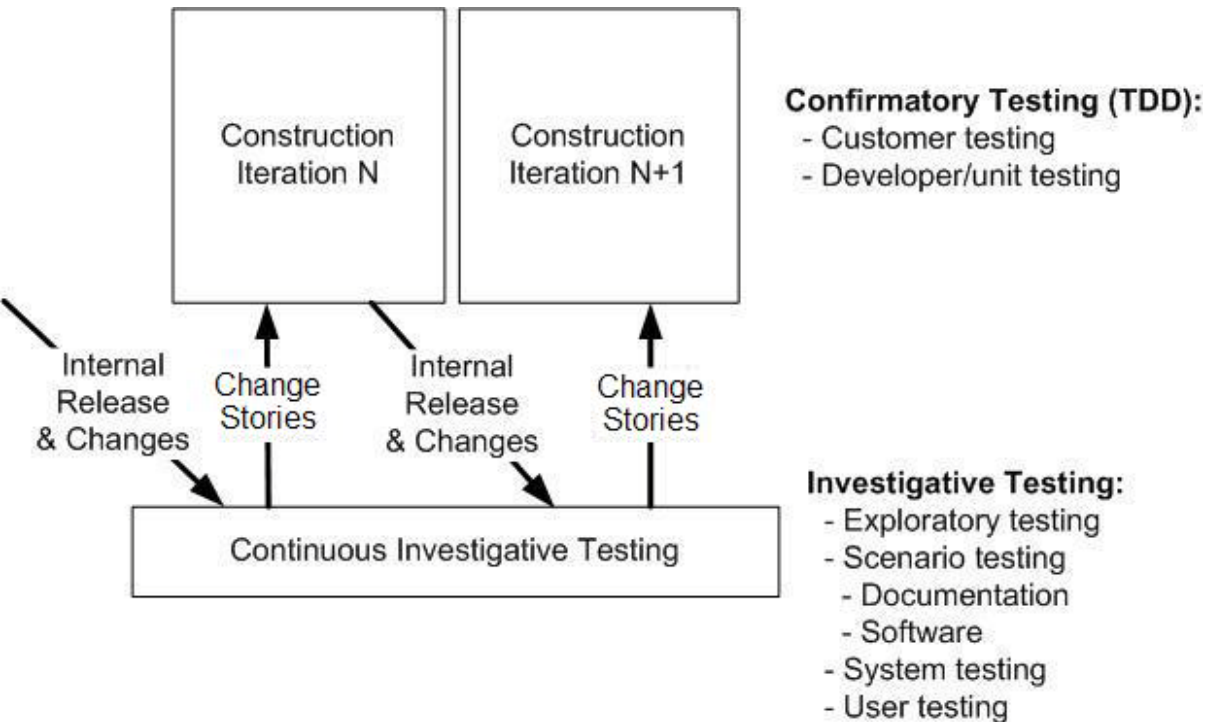
Agile Modeling Best Practices

www.agilemodeling.com/essays/bestPractices.htm



Copyright 2005-2007 Scott W. Ambler

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

Source: January 2007 Dr. Dobb's Magazine

(www.ddj.com/dept/debug/196603549)

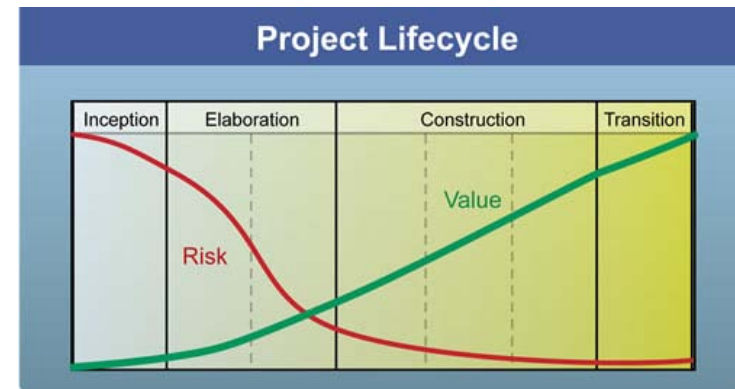
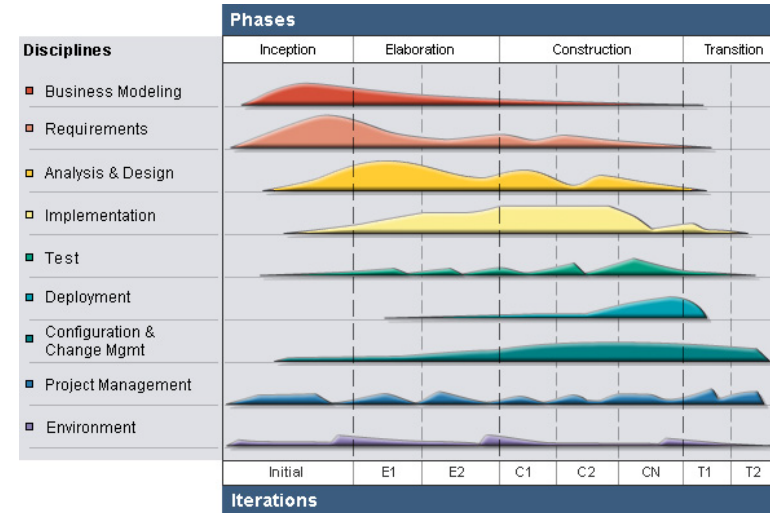
Agile Database Practices

- Database Refactoring
 - www.agiledata.org/essays/databaseRefactoring.html
- Database Regression Testing
 - www.agiledata.org/essays/databaseTesting.html
- Continuous Database Integration
 - www.martinfowler.com/articles/evodb.html



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



Agenda

- Agile Software Development
- Agile Adoption Rate
- Scaling Agile
- Successful Adoption
 - Measured Capability Improvement Framework (MCIF)
 - Critical IBM Agile Resources

Target: Phase 1

Already implemented

Outside scope

Measured capability improvement framework

Map business value to software delivery practices

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

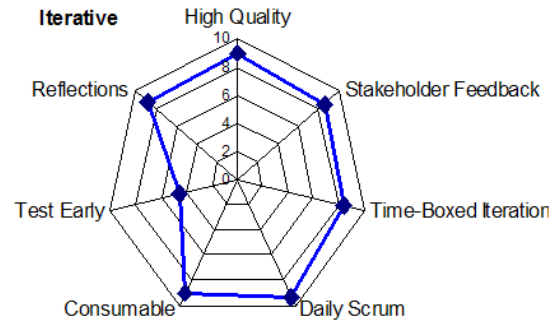
Software Delivery Best Practices

- 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



Critical IBM Agile Resources

www.ibm.com/rational/agile/

www.ibm.com/developerworks/

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



Questions