



IBM Software Group

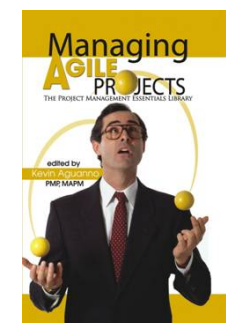
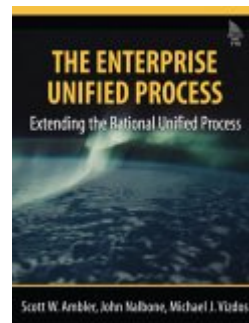
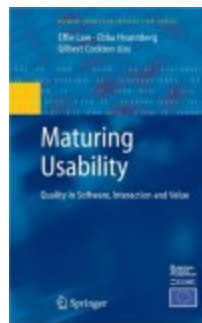
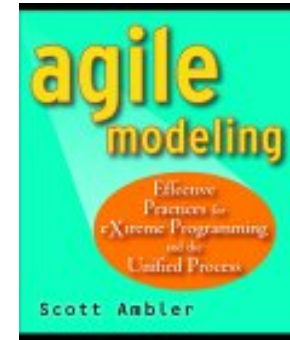
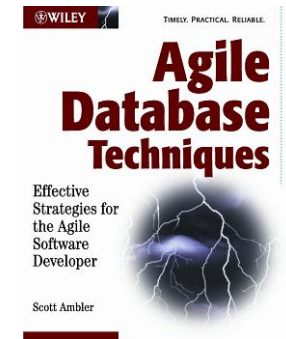
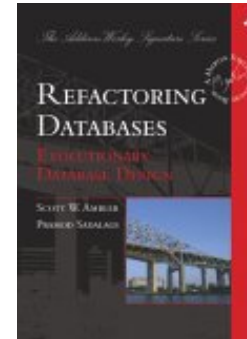
Scaling Agile Software Development for the Real World

Rational. software

[Go to IBM](#)

Scott Ambler - Background

- Practice Leader Agile Development
- Senior Contributing Editor, DDJ
- Fellow – International Association of Software Architects
- www.ibm.com/rational/bios/ambler.html
- www.ibm.com/developerworks/blogs/page/ambler



Agenda

- Agile Software Development
- Agile Adoption and Success Rates
- Scaling Agile Development
- Conclusion



Agenda

- Agile Software Development
 - ▶ What have you heard?
 - ▶ Defining Agile
 - ▶ How Agile is Different
- Agile Adoption and Success Rates
- Scaling Agile Development
- Conclusion



Discussion: What Have You Heard About Agile?

- What is Agile?
- What are your concerns?
- When doesn't Agile apply?
- Are you doing Agile?
- Are you thinking about doing Agile?



Defining 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

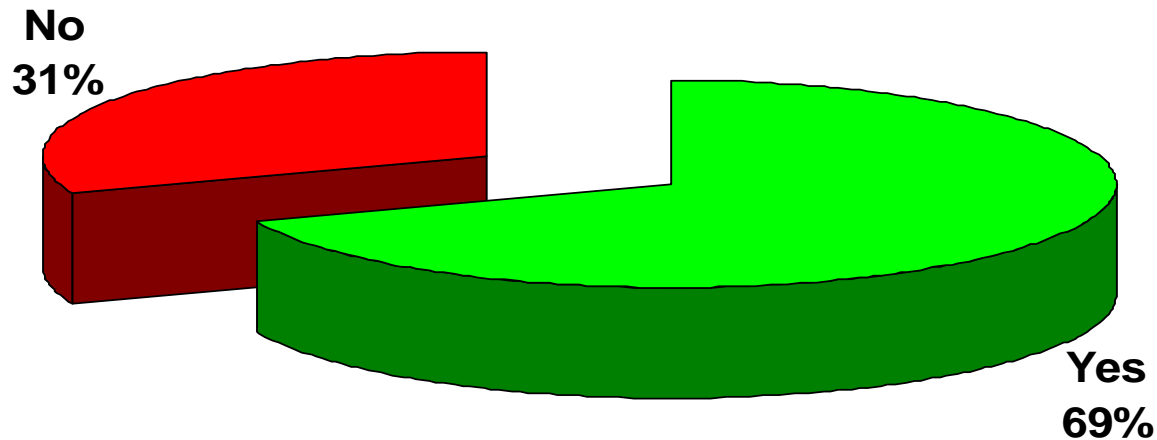


Agenda

- Agile Software Development
- Agile Adoption and Success Rates
 - ▶ Adoption rates
 - ▶ Success rates
- Scaling Agile Development
- Conclusion



Has Your Organization Adopted One or More Agile Techniques?



18% are still in the “pilot phase”

15% of “No” respondents hope to do Agile this year

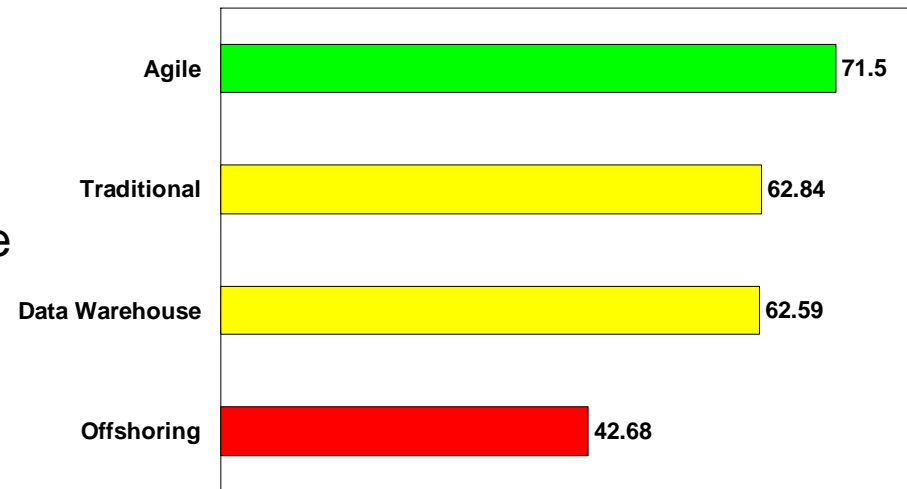
Source: Interim results from Dr Dobb’s 2008 Agile Adoption Survey,

Updated results to be posted at www.ambyssoft.com/surveys/ in May



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



Agenda

- Agile Software Development
- Agile Adoption and Success Rates
- **Scaling Agile Development**
- Conclusion



Discussion: Does Agile Scale?

- What are the issues that concern you?
- What are your experiences with Agile in practice?
- Have you tried to scale agile?



Challenges with Agile in the Mainstream

Agile Development

Compliance requirement

Low risk ↔ Critical, Audited

Enterprise discipline

Project focus ↔ Enterprise focus

Geographical distribution

Co-located ↔ Global

Entrenched process, people, and policy

Minimal ↔ Significant

Application complexity

Simple, single platform ↔ Complex, multi-platform

Organization distribution (outsourcing, partnerships)

In-house ↔ Third party

Team size

Under 10 developers ↔ 100's of developers

Degree of Governance

Informal ↔ Formal



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
- 50+ developers
- Complex, multi-platform applications
- Distributed teams
- Need for scalability, reproducibility, and traceability

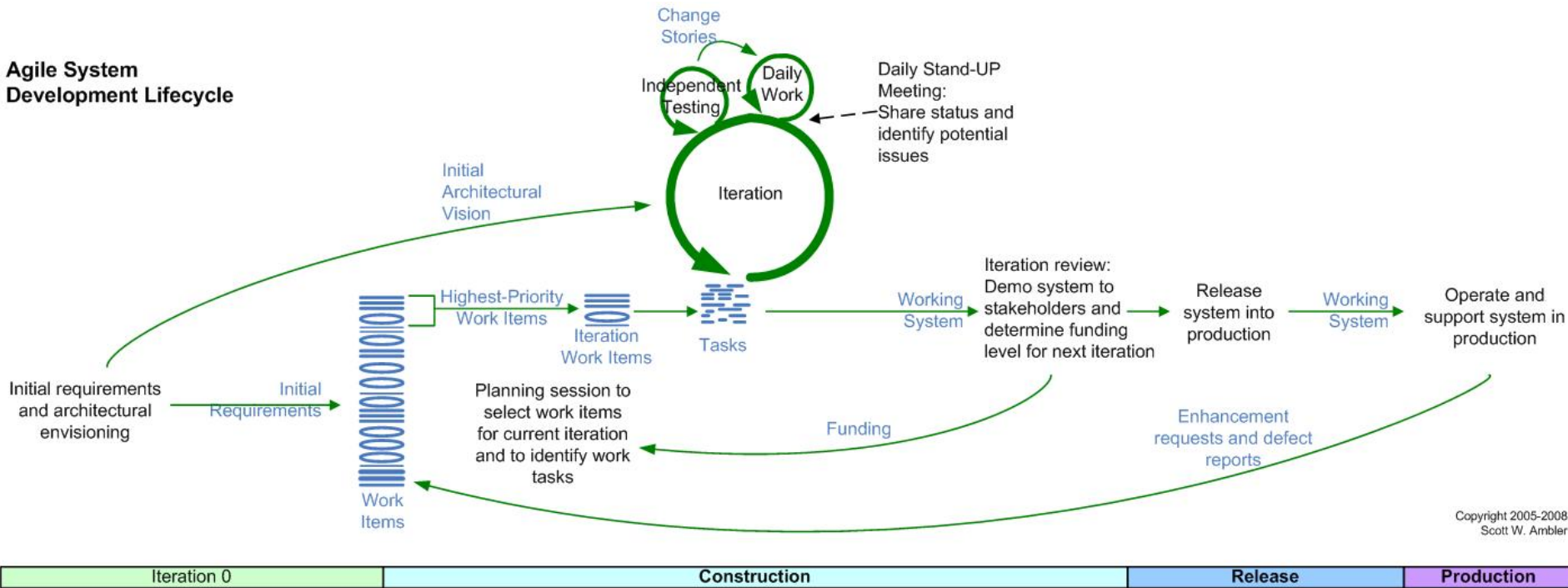
Technical and Regulatory Drivers

Compliance
 Governance
 Application complexity



The Agile System Development Lifecycle (SDLC)

Agile System Development Lifecycle



Copyright 2005-2008
Scott W. Ambler



Agenda

- Agile Software Development
- Agile Adoption and Success Rates
- Scaling Agile Development
- Conclusion
 - ▶ Parting Thoughts?
 - ▶ Critical Resources



Discussion: Parting Thoughts?

- Can you make Agile work in your organizations?
- Is it worth trying?
- What impediments do you think there will be?



Critical IBM Agile Resources

www.jazz.net

www.ibm.com/rational/agile/

www.ibm.com/developerworks/

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



