Agility at Scale: Applying Agile Software Development Techniques on Real-World Projects

Scott W. Ambler
Practice Leader Agile Development, IBM Rational
scott_ambler@ca.ibm.com
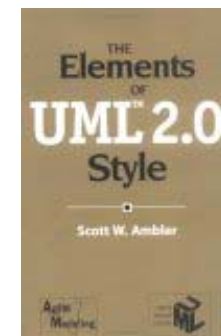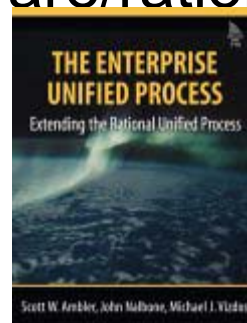
IBM Rational Software Development Conference UK 2007

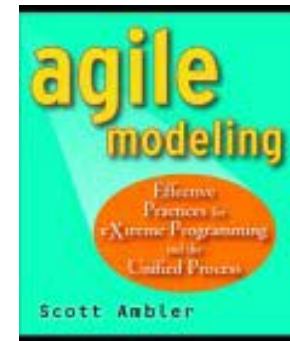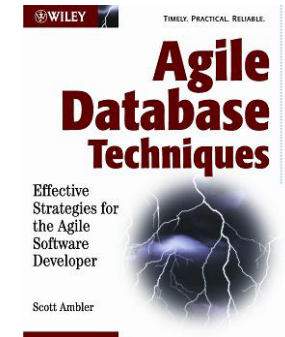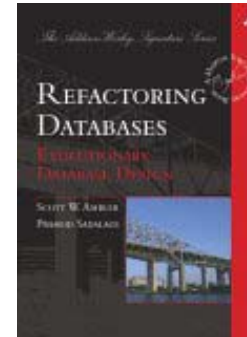What keeps me **Rational**?

Scott Ambler - Background

- **Practice Leader Agile Development**

- **Fellow – International Association of Software Architects**

- **www-306.ibm.com/software/rational/bios/ambler.html**

Discussion Topics

- Warning!

- Agile Software Development

- Agile Adoption Rate

- Scaling Agile

- Parting Thoughts
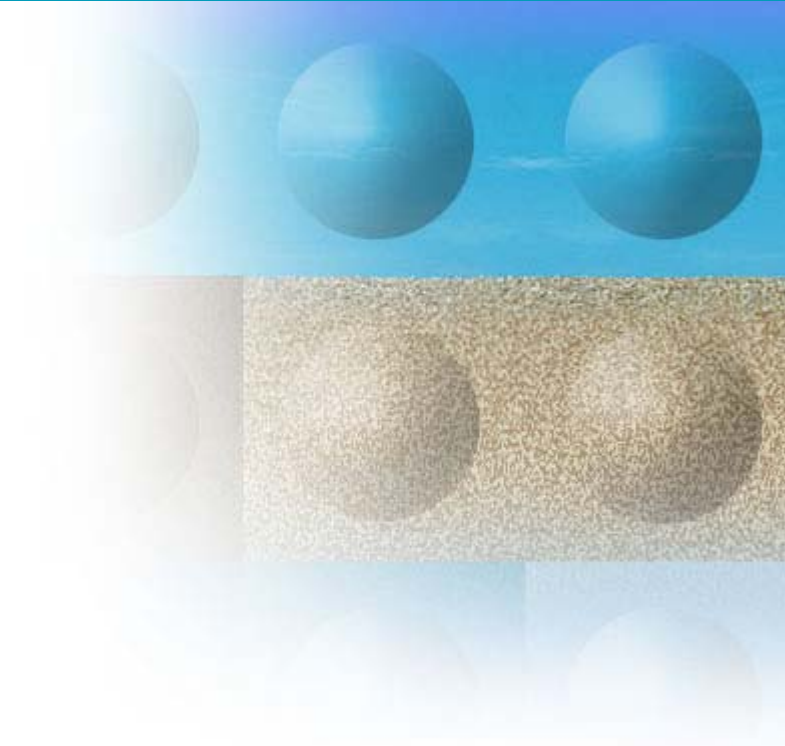
Warning!

- I'm spectacularly blunt at times

- Many new ideas will be presented

- Some may not fit well into your existing environm

- Some will challenge your existing notions about software development

- Some will confirm your unvoiced suspicions

- Don't make any "career-ending moves"

- Be skeptical but open minded

Discussion Topics

- Warning!

- Agile Software Development
  - ▶ What is Agile?
  - ▶ Agile Mythbusters
  - ▶ Agile Practices

- Agile Adoption Rate

- Scaling Agile

- Parting Thoughts
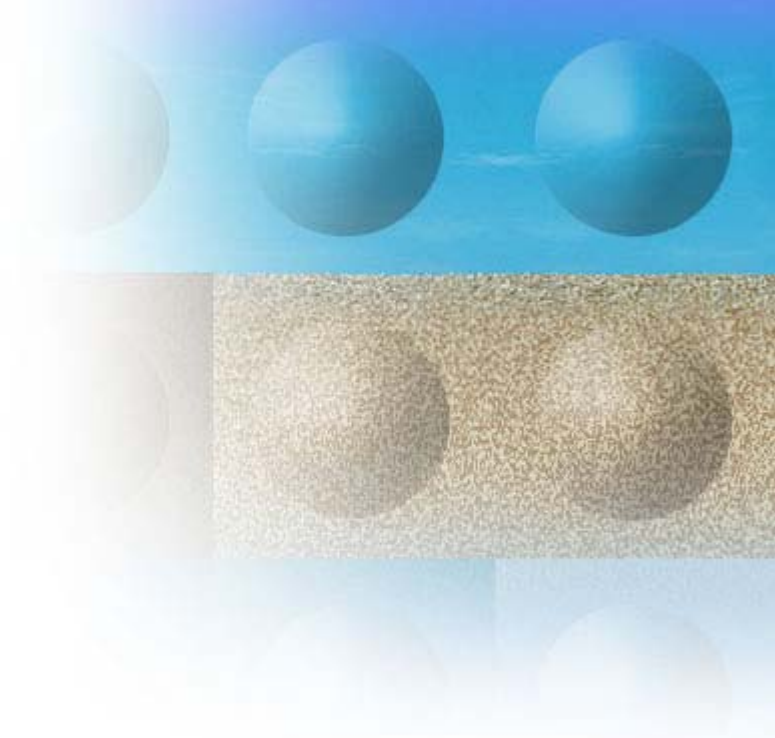
What is Agile?

- An iterative and incremental (evolutionary) approach performed in a highly collaborative and self-organizing manner with **just the right amount of ceremony** to 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

Mythbusters

1. Agile provides stakeholders control over the budget, schedule, and scope

2. Agilists document – Google "Agile Documentation"

3. Agile requires great discipline

4. Agilists plan on a just-in-time (JIT) basis

5. Agile is far more predictable – Greater quality, better economics, increased governance

6. Agile scales – The Eclipse Way is agile

7. Agile is quickly becoming the norm

8. Agile requires skilled people

9. RUP is as agile as you make it

Mainstream Agile Practices

- Regular Deployment of Working Software

- Non-Solo Development

- Refactoring

- Continuous Integration

- Configuration Management

- Test Driven Development (TDD)

- Agile Testing

- Agile Documentation

High Priority

Each iteration implement the highest-priority work items

High-priority work items should be well-defined

Each new work item is prioritized and added to the stack

Work items may be reprioritized at any time

Low-priority work items can be vague

Work items may be removed at any time

Low Priority

Work Item List
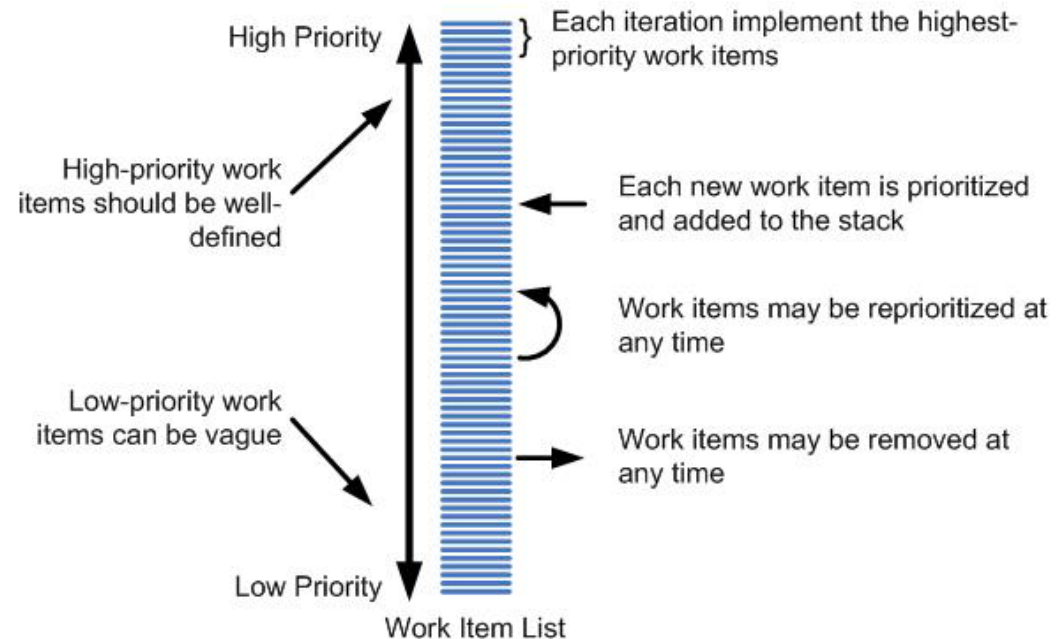
Discussion Topics

- Warning!

- Agile Software Development

- Agile Adoption Rate

  ▶ Survey overview

  ▶ Adoption rates

  ▶ Success rates

- Scaling Agile

- Parting Thoughts
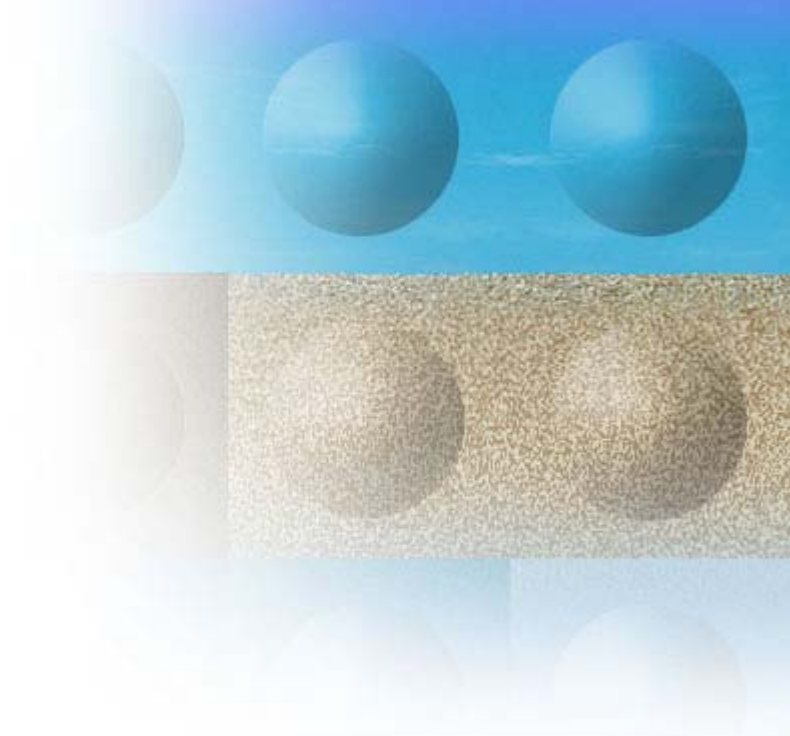
Survey Results

- **Agile Adoption Survey**
  - ▶ March 2007
  - ▶ Aug 2007 Column
  - ▶ Advertised in Editor's blog on www.ddj.com
  - ▶ 781 respondents
  - ▶ 52% were developers, 22% were in management
  - ▶ 73% had 10+ years in IT
  - ▶ 33% worked in orgs of 1000+ people
  - ▶ 85% worked in commercial firms
  - ▶ 95% North American

- **Project Success Survey**
  - ▶ August 2007
  - ▶ Dec 2007 Column
  - ▶ Email to DDJ mailing list
  - ▶ 535 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

- Details posted at www.ambysoft.com/surveys/

Has Your Organization Adopted One or More Agile Techniques?

No 31%

Yes 69%
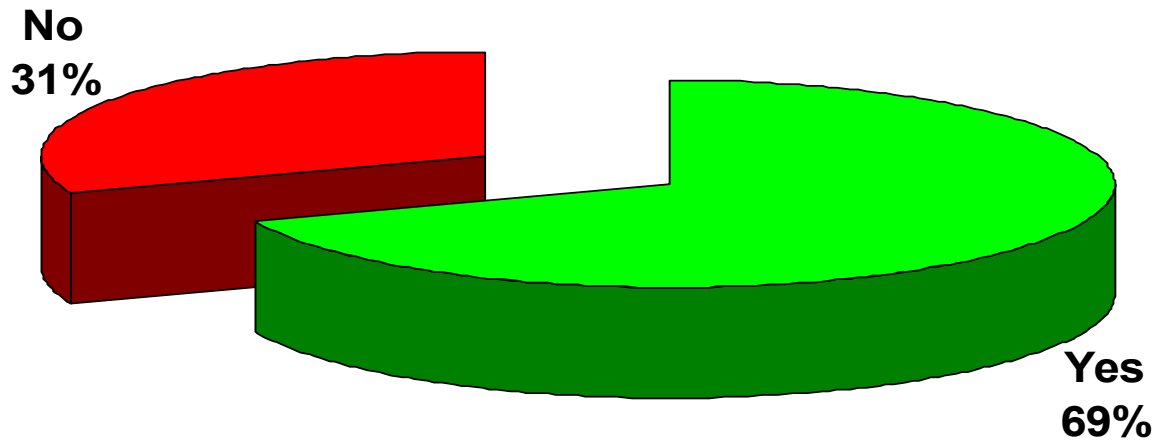
When Will You Adopt Agile?
(241 said they hadn't yet)

4%
8%
12%
46%
9%
9%
12%

| ■ <3 Months | ■ 3-6 Months | ☐ 6-12 Months | ☐ 12-24 Months |
| ■ > 24 Months | ■ Never | ☐ Don' Know | |
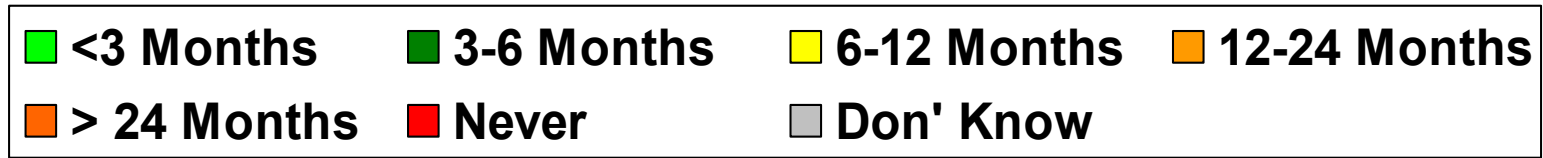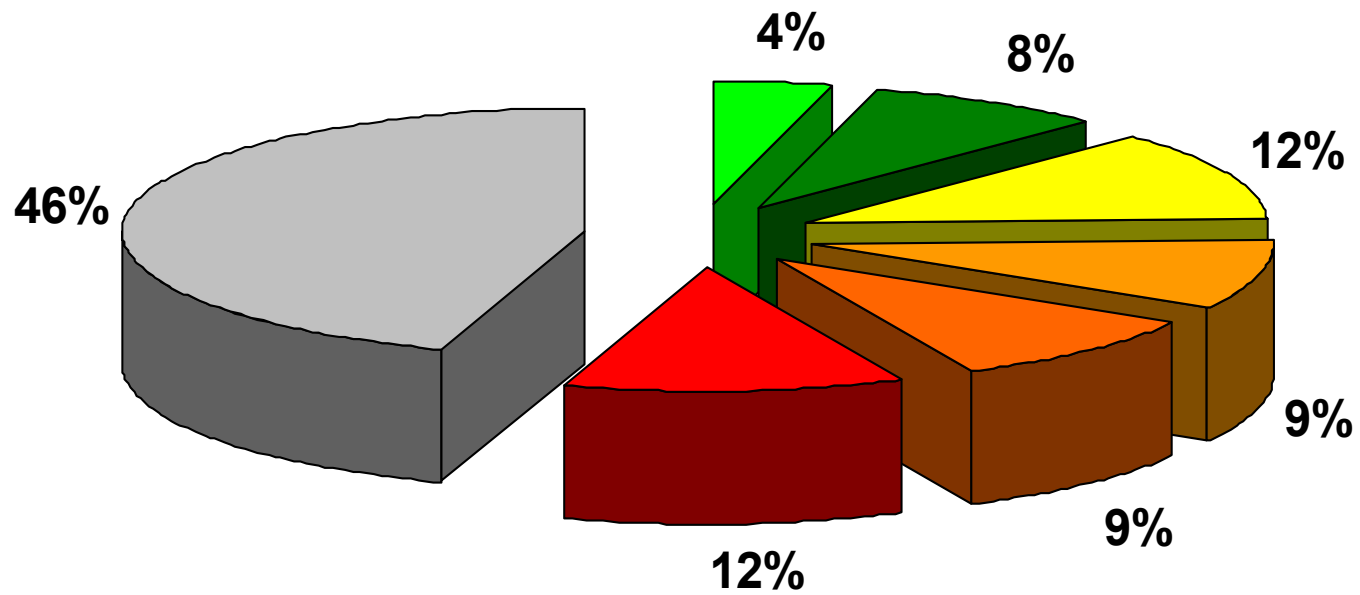
Number of Agile Projects Run
85% have run multiple agile projects

Category	Value
20+	57
10 to 20	30
5 to 10	82
2 to 5	194
1	64

Project Success Rate (%)

Category	Success Rate
Agile	71.5
Traditional	62.84
Data Warehouse	62.59
Offshoring	42.68
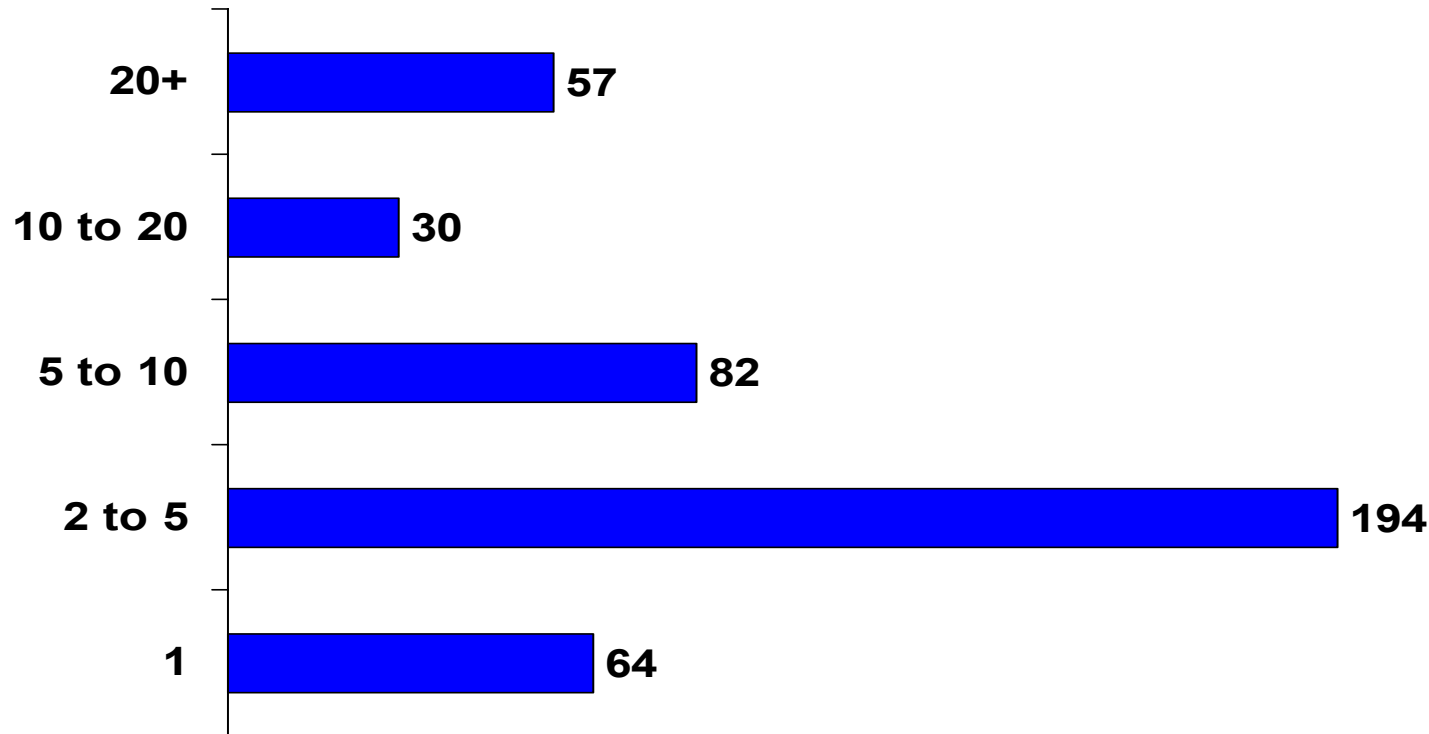
% of Successful Agile Projects
(296 co-located, 251 not co-location, 130 offshoring): Agile Adoption Survey
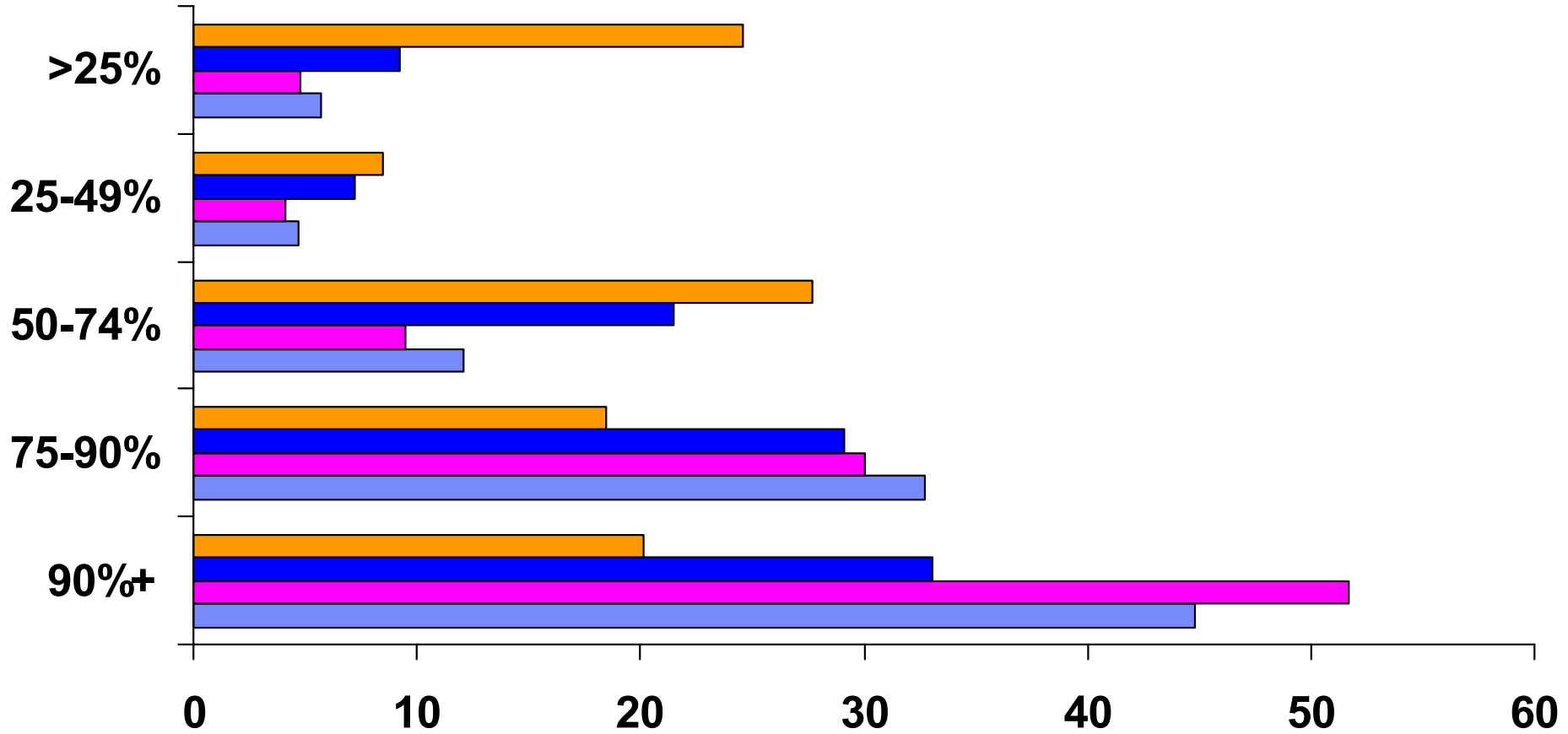
Legend: ■ All ■ Co-Located ■ Not Co-Located ■ Offshoring

Largest Team Size Attempted vs. Successful
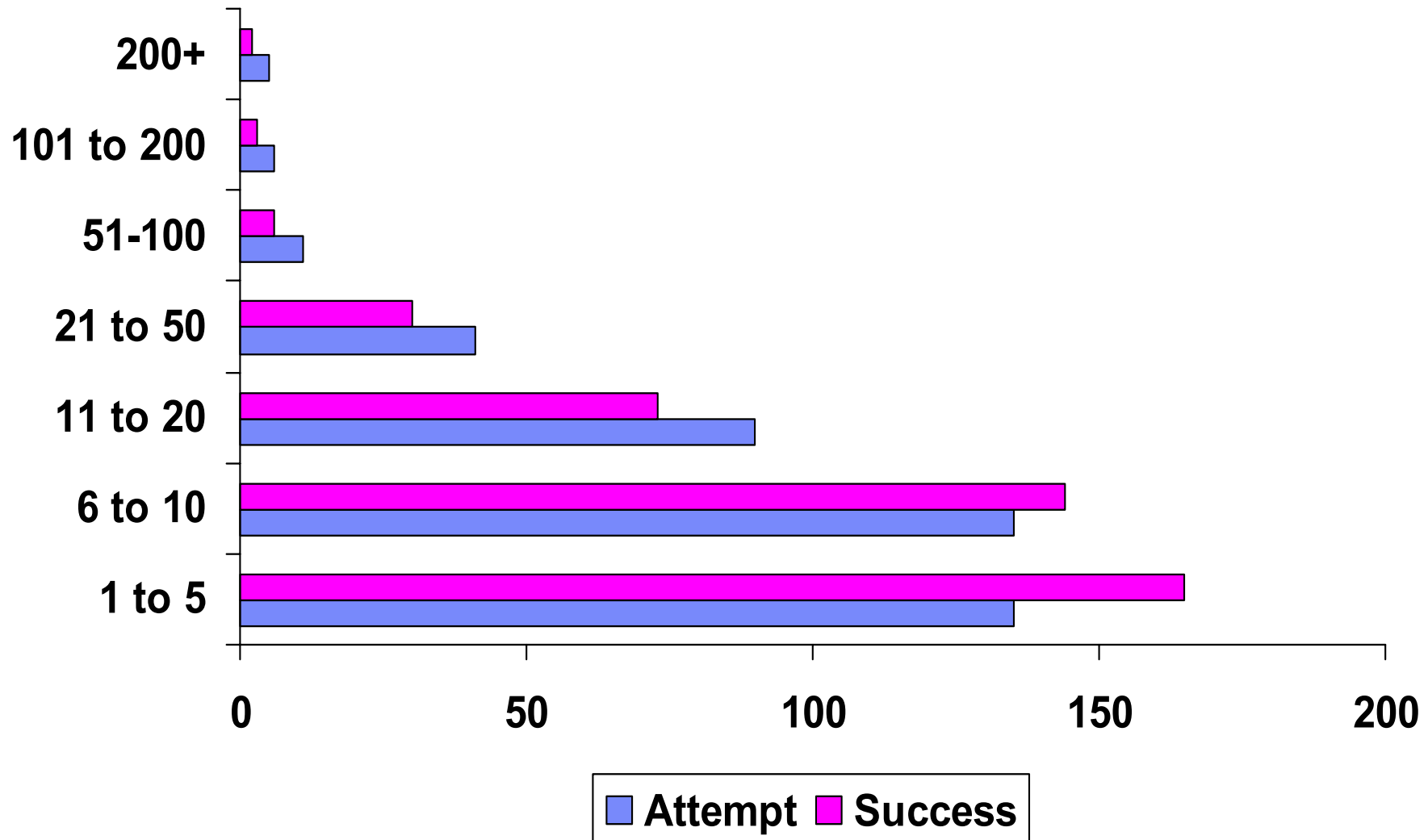
Discussion Topics

- Warning!

- Agile Software Development

- Agile Adoption Rate

- Scaling Agile

  ▶ Challenges with Agile in the Mainstream

  ▶ Agility is Relative

  ▶ Process and Tooling

  ▶ AMDD Practices

  ▶ RUP Agile

  ▶ Agile Data Practices

  ▶ Lean Governance

- Parting Thoughts

Challenges with Agile in the Mainstream

Compliance requirement
Low risk ⟷ Critical, Audited

Geographical distribution
Co-located ⟷ Global

Entrenched process, people, and policy
Minimal ⟷ Significant

Agile Development

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
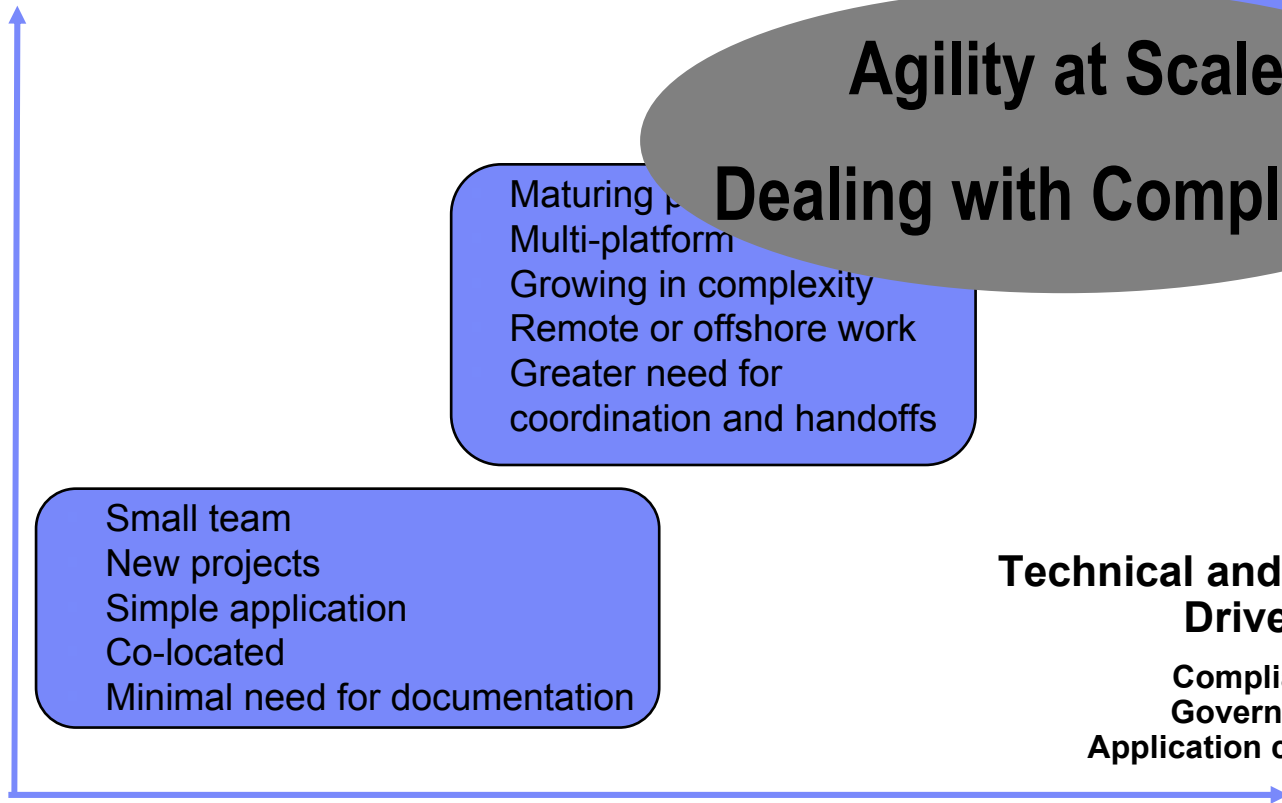
Mature or existing projects
50+ developers
Complex. multi-platform applications

Agility at Scale

Dealing with Complexity

Maturing p...
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

The Agile Process Continuum

Organizational Drivers

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

Rational Unified Process

Eclipse Way

XP, Scrum,
OpenUP, DSDM
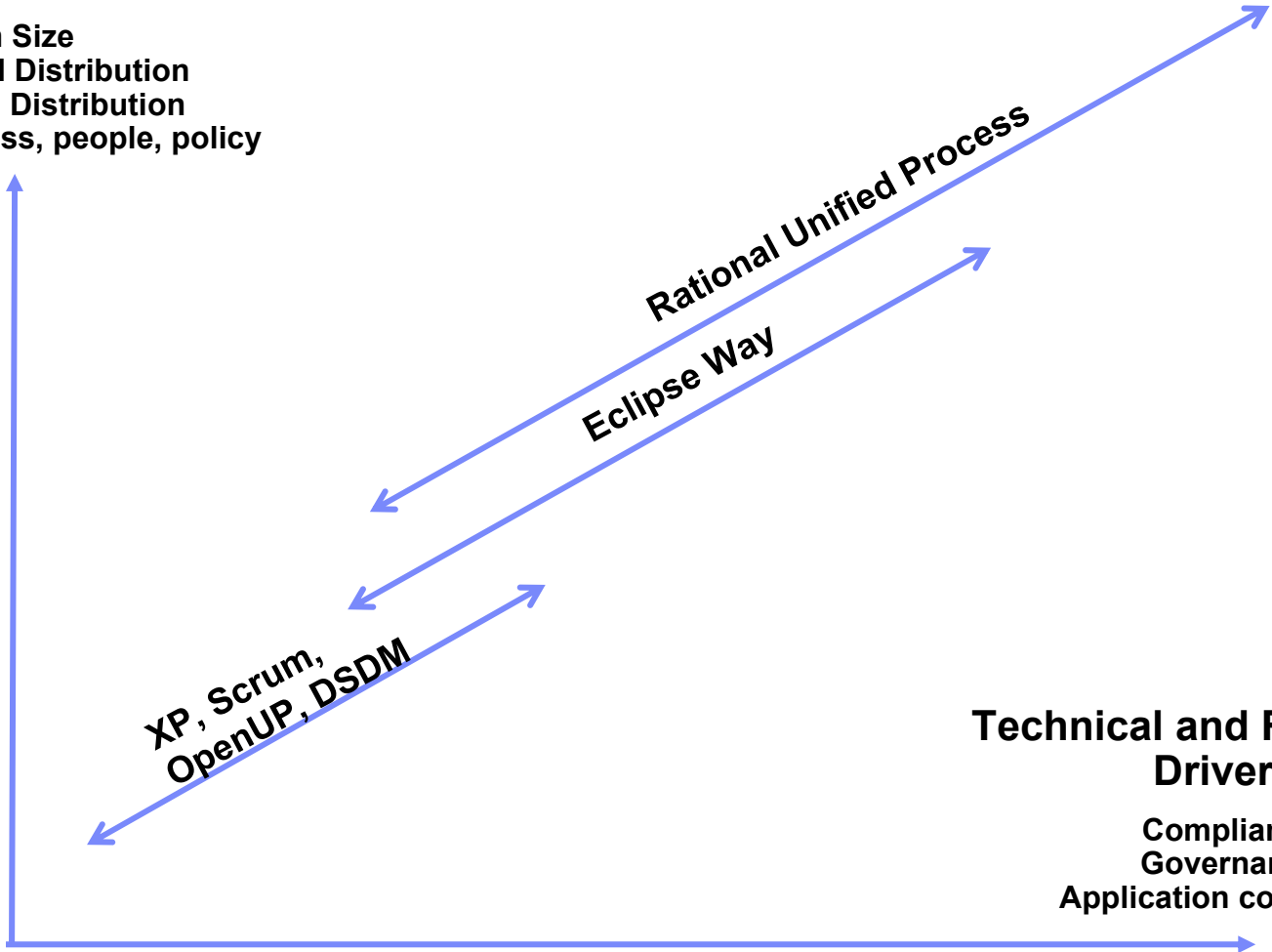
Technical and Regulatory Drivers

Compliance
Governance
Application complexity

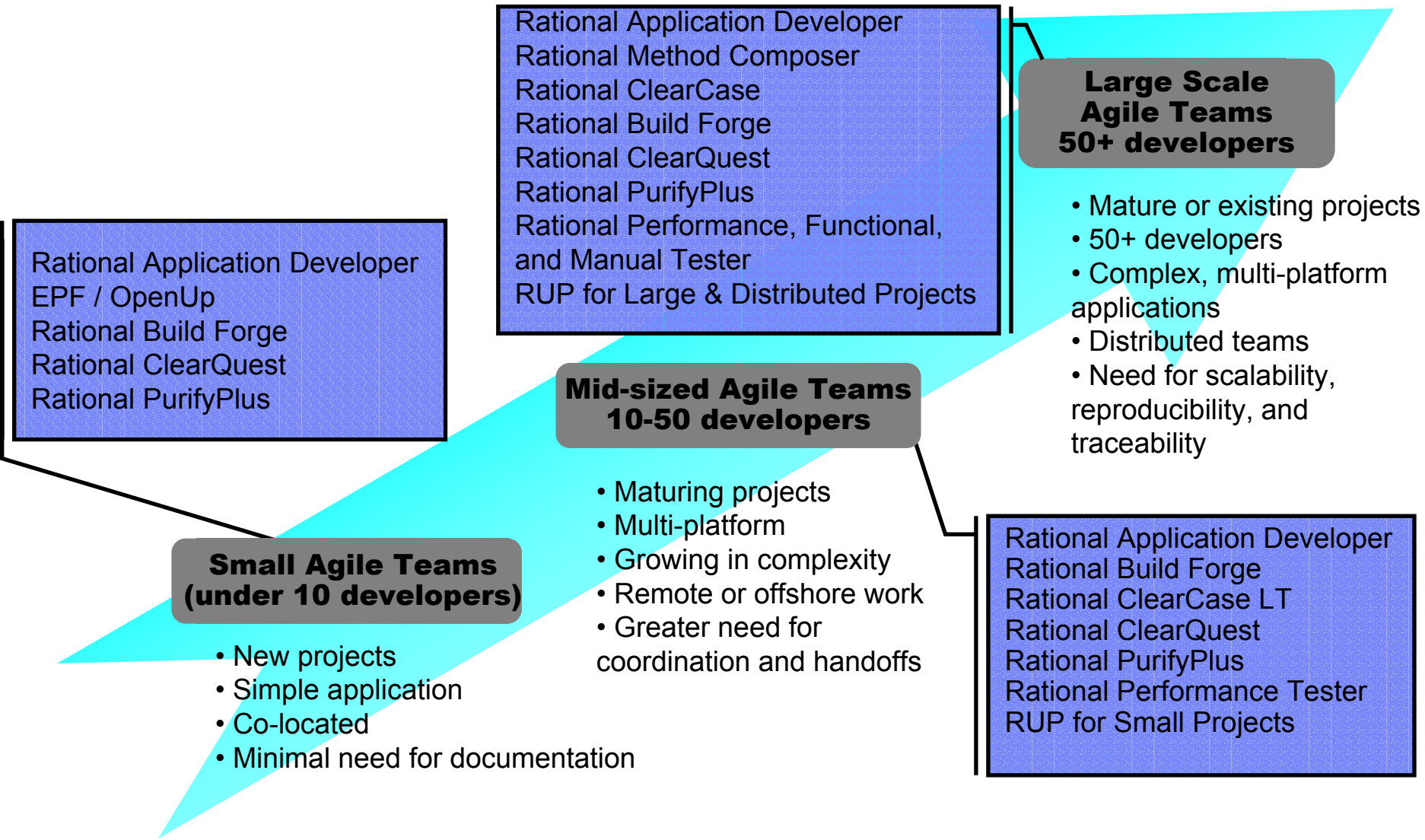# Solutions for Agile Teams of Every Size

Rational Application Developer
Rational Method Composer
Rational ClearCase
Rational Build Forge
Rational ClearQuest
Rational PurifyPlus
Rational Performance, Functional, and Manual Tester
RUP for Large & Distributed Projects

**Large Scale
Agile Teams
50+ developers**

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

Rational Application Developer
EPF / OpenUp
Rational Build Forge
Rational ClearQuest
Rational PurifyPlus

**Mid-sized Agile Teams
10-50 developers**

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

**Small Agile Teams
(under 10 developers)**

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

Rational Application Developer
Rational Build Forge
Rational ClearCase LT
Rational ClearQuest
Rational PurifyPlus
Rational Performance Tester
RUP for Small Projects

What keeps me **Rational**?
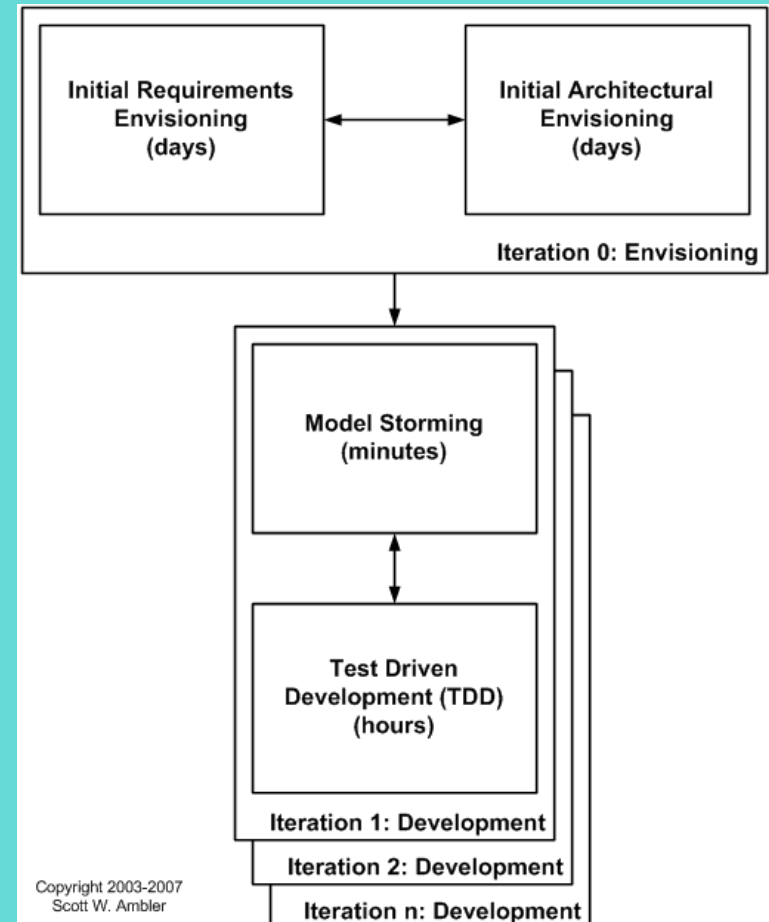
Initial Requirements Modeling

- Your goals are to:
  1. Identify and agree to the initial scope of your project
  2. Develop the initial stack of requirements
  3. Gather enough information to address initial scheduling and estimating concerns

- Critical models for business application development:
  - Some sort of usage model (use cases, user stories, …)
 - Conceptual/domain model
  - Some UI sketches

Initial Architectural Modeling

- Your goals are to:
  1. Identify and agree to a potential initial architecture of your system
  2. Provide sufficient technical vision for estimating and scheduling concerns

- Critical models for business application development:
  - ▶ Some form of deployment diagram
  - ▶ A free-form "technology stack" diagram

Copyright 2003-2007
Scott W. Ambler
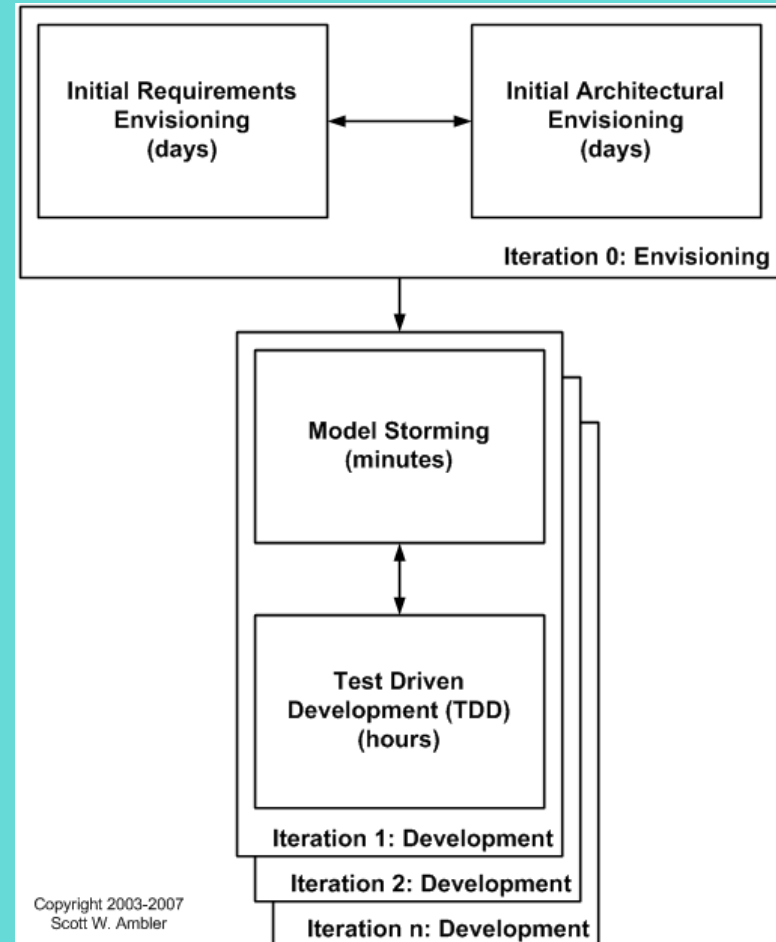
Model Storming

- Just-in-time (JIT) modeling

- Model when you run into "difficulty"
  - ▶ You have a new feature to implement, and need to "analyze" the details
  - ▶ You need to think through a technical issue

- Most modeling on agile projects is done in 10-15 minute bursts followed by hours or days of programming

- Waiting to model is more effective:
  - ▶ You understand the domain better
  - ▶ Stakeholders understand the current solution better
  - ▶ You can focus only on what you need to build

Tooling strategies:

- Choose tools based on skill and need

- Sometimes the "simplest" tool is a complex one

- Whiteboards and paper are great for thinking things though, not so good for capturing information permanently

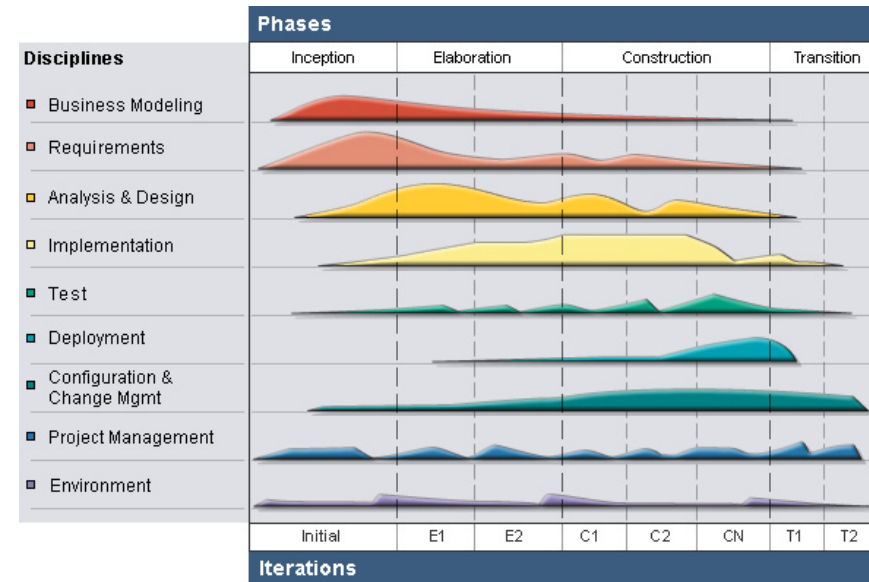- Different teams will have different tooling needs

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)
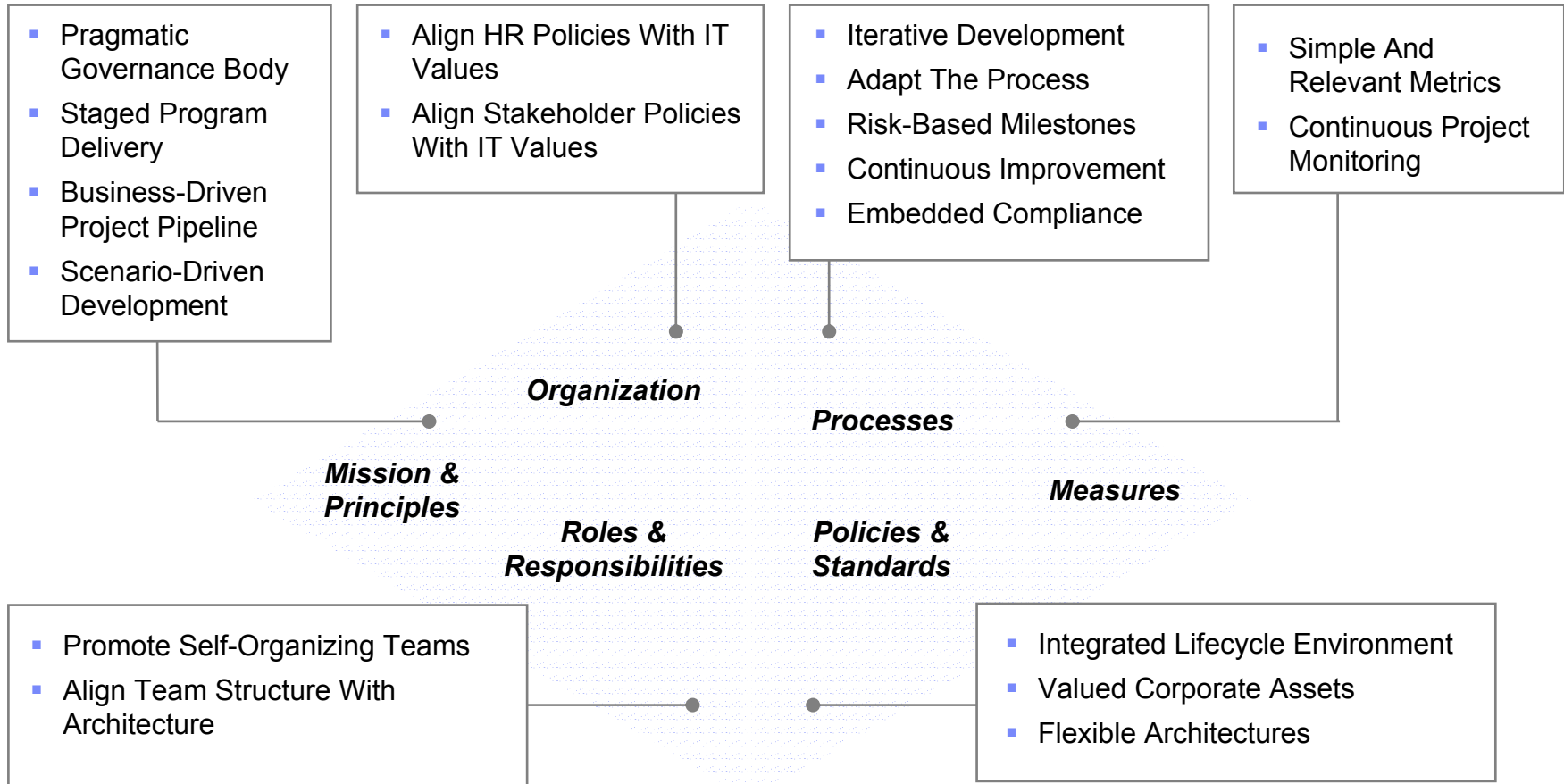
- Organizations have instantiated RUP to be very agile

- Adopt:
  - ▶ Regular delivery of working software
  - ▶ Active stakeholder participation
  - ▶ Daily stand up meetings
  - ▶ Test-Driven Development (TDD)
  - ▶ Continuous builds
  - ▶ Agile documentation techniques

- Scaling strengths:
  - ▶ Business risk addressed in Inception
 - ▶ Technical risk addressed in Elaboration

Lean Development Governance
www.ibm.com/developerworks/

- Pragmatic Governance Body
- Staged Program Delivery
- Business-Driven Project Pipeline
- Scenario-Driven Development

- Align HR Policies With IT Values
- Align Stakeholder Policies With IT Values

- Iterative Development
- Adapt The Process
- Risk-Based Milestones
- Continuous Improvement
- Embedded Compliance

- Simple And Relevant Metrics
- Continuous Project Monitoring

Organization

Processes

Mission & Principles

Measures

Roles & Responsibilities

Policies & Standards

- Promote Self-Organizing Teams
- Align Team Structure With Architecture

- Integrated Lifecycle Environment
- Valued Corporate Assets
- Flexible Architectures

Discussion Topics

- Warning!

- Agile Software Development

- Agile Adoption Rate

- Scaling Agile

- Parting Thoughts
  - ▶ Succeeding with Agile
  - ▶ Critical IBM Agile Resources
  - ▶ A Call to Action
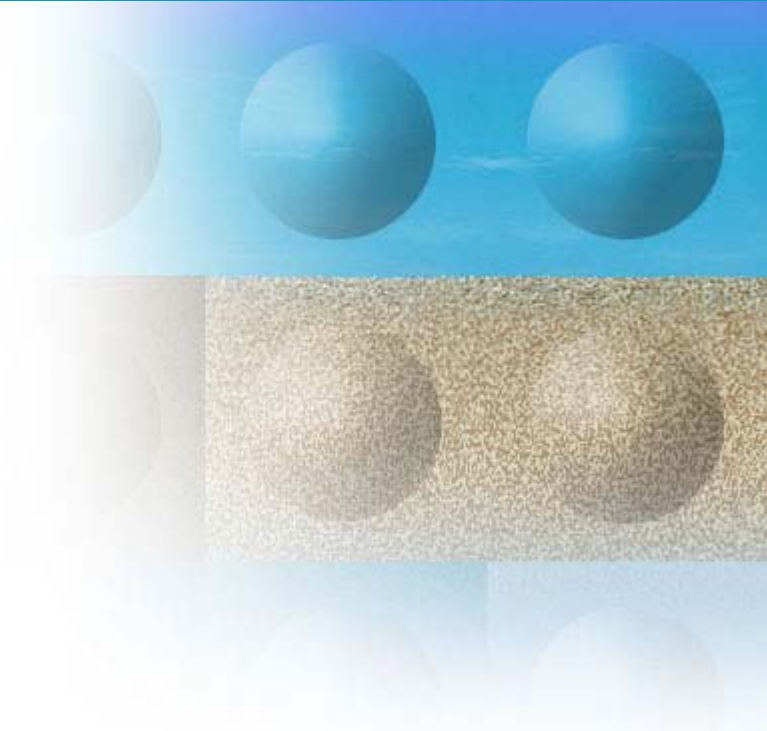
Succeeding with Agile

- **Agile Development is transforming how development is done**
  - ▸ Placing greater demand on processes and tools

- **Agile can scale to accommodate technical and organizational complexity**
  - ▸ Agility at Scale has unique needs
  - ▸ IBM Rational brings a wealth of knowledge to help companies succeed in complex environments

- **IBM Rational is forging new ground in Agile Development**
  - ▸ Bringing technology, services and best practices to help customers reduce the risk of Agile projects
  - ▸ Investing significant resources to improve Agile success

Critical IBM Agile Resources

www.ibm.com/rational/agile/

www.ibm.com/developerworks/

A Call To Action

- **Consider an Agile Pilot Project**
  - ▶ See it work for yourself
  - ▶ Get mentoring help

- **Get some Agile training**
  - ▶ Project management training is critical
  - ▶ Training modelers, developers, … is also critical

- **Adopt an Agile Form of RUP**
  - ▶ RUP done right is agile
  - ▶ RUP provides the control mechanisms and risk-mitigation strategies that other agile methods lack

Thank You

Scott W. Ambler

scott_ambler@ca.ibm.com