

Agile Alliance Survey: Are We There Yet?

Posted by [Diana Larsen](#) on Sep 13, 2006 01:00 AM

Community [Agile](#) Topics [Agile in the Enterprise](#) , [Methodologies](#) Tags [Agile Alliance](#) , [Statistics](#)



Agile, Five Years Later

To start with a short history: Five years ago, in the spring of 2001, seventeen free-thinking software engineers got together to uncover commonalities in their thinking about software development methods. Individually, through the 1980's and 1990's, they had developed a variety of low ceremony approaches to writing software that had become grouped under the designation, "Lightweight Methods." During this pivotal meeting they re-named their suite of approaches, "Agile." They wrote the Agile Manifesto*, signed it, published it online and invited others to sign on to it too. Soon, they and other enthusiasts formed a non-profit organization—The Agile Alliance, conferences bloomed, some of the signers wrote more books, and Agile was launched.

Related Vendor Content

[5 Ways to Ensure Application Performance](#)

[The Agile Business Analyst: Skills and Techniques needed for Agile](#)

[The Future of Software Delivery According to visionaries Grady Booch & Erich Gamma](#)

[Agile Development: A Manager's Roadmap for Success](#)

[The Agile Project Manager](#)

Related Sponsor

[VersionOne](#) is recognized by Agile practitioners as the leader in Agile project management tools. Companies such as Adobe, BBC, CNN, Dow, HP, IBM, Sony and 3M have turned to VersionOne to help deliver greater value to their customers.



Looking back over the last five years and forward to the next five, where is Agile headed now? I thought about this question myself and circulated a survey to the other Agile Alliance board members for their input. Together we formed a picture of the trends we're seeing in the Agile world. The trends include:

- Awareness & Propagation
- Leadership & Scaling
- Commodification & Dilution
- Emerging Challenges

Awareness & Propagation: The spread of "Agile".

By now, most people in the software industry have heard about Agile software development, whether or not they are clear about what it means. Ron Jeffries wrote, "Agile approaches are being applied effectively in essentially every kind of enterprise, and with good results. It is certainly possible to fail using Agile, but the situations where this happens seem to begin by not doing anything that would be recognized as truly applying the principles and practices."

Agile, and the recognition of the need for methods that respond to market changes and deliver faster, is showing up in places where no one expected it, like on mission critical projects (NASA and military applications) and other government-regulated processes (e.g., the V-Model XT in Germany).

We're no longer debating whether agile works to develop software, we're debating how to do it in a specific company/context. Agile methods are no longer considered the approach of last resort in a crisis, but more often turned to as the first choice for rapid, responsive, accurate development.

In places where Scrum is adopted as a management practice, XP is often not far behind as the development practice. Yet when people think of Scrum, XP, Crystal or DSDM, they still don't automatically think, "Agile." They think of the methods as techniques, or about the related tools, rather than about the values Agile represents.

Agile has become a "buzzword" in the software development world, often without a clear understanding of what it means. Agile practices may be used but not recognized as such. Rachel Davies wrote, "I have just been working with a team who adopted Lean Software Development from Mary & Tom Poppendieck's book. They don't know XP or Scrum but are working in one-week iterations within 6-week stages with small cross-functional teams of business, developers and testers. I have introduced them to estimating in points rather than guessing."

What are some of the ways we find evidence for the spread of Agile?

1) Evidence in Publications

Five years ago, Ole Jepsen noted, there were no books about Agile; only books describing the separate methods. Three years ago, you'd find only a few Agile books on the bookstore shelf, and still very little with the specifics of how to go Agile. In the last 12-18 months, books with detailed advice are starting to appear:

- Sanjiv Augustine's [Managing Agile Projects](#)
- Mike Cohn's [Agile Estimation and Planning and User Stories Applied](#)
- Jutta Eckstein's [Agile Development in the Large](#)
- Ward Cunningham and Rick Mugridge's [Fit for Developing Software : Framework for Integrated Tests](#)
- Ken Schwaber's [Agile Project Management with Scrum](#)
- Michael Feather's [Working Effectively with Legacy Code](#)
- and others

2) Evidence from Conferences

When the two U.S.-based conferences on Agile merged last year, the organizers expected that much the same crowd of 150–250 people had been attending both. Allowing for that overlap and some growth, they planned for about 300–400 attendees at the Agile 2005 conference. More than 700 people came. Many people came, not because they were part of Agile projects, but to satisfy their curiosity and discover whether Agile might be a fit for them. Conference Director Todd Little reports that the 2006 organizing committee expects another 30%+ growth in attendance. They plan a conference that will accommodate 800–1200 people.

And not only that—Agile topics appear more and more often in the programs of more and more non-Agile software conferences. A planning committee for a software testing and quality conference recently discussed how to preserve a balance between Agile and non-Agile topics. The increasing numbers of solid proposals for Agile-related sessions threatened to overshadow more traditional topics at the event.

3) Evidence of Vendors and Tools

More and more vendors are entering the Agile space. In 2001, there were no tools on the market and only a handful of consultants to help start an Agile project or provide skills training in Agile practices.

In 2006, Rally, Version 1, Microsoft, IBM and others, have developed and promoted tools to assist Agile projects. Open Source tools, custom-designed for application in particular projects, have also entered the marketplace.

Dozens of consultants stand ready to assist with practices and support.

4) Evidence from Marketplace Analysts

Gartner, Forrester, IDC and other respected industry analysts have recognized Agile as a phenomenon and mention the influence of Agile in their reports.

Examples from Forrester:

On March 2004, Forrester published a report, "Adopting Agile Development Processes: Improve Time-To-Benefits For Software Projects"

On November 30, 2005 a report "Corporate IT Leads The Second Wave Of Agile Adoption"

On February 7, 2006, a report "Agile Processes Enable SOA Success"

Example from Gartner:

On April 18, 2005, Agile Requirements Definition and Management Will Benefit Application Development

5) Evidence from workplace anecdotes

Agile has shifted its early developer-centric start to focus on products owners, customers, testers, usability experts and others, as needed by the project.

Testers and usability experts have become an expected part of the Agile development team. Discussion lists contain postings about Agile testing & QA, as well as requests from teams looking to hire Agile testers, something unheard of as little as two years ago. The debates about whether Agile testing and usability are possible are subsumed by the number of teams that include both.

Research, journal articles and conference sessions give more attention to the important roles of customer and product owner on Agile projects.

6) Evidence from Signs of Interest

Seventy-eight local user groups have requested a listing on the Agile Alliance website. <http://www.agilealliance.org/resources/usergroups/>

Discussion group members access each other through over 70 yahoo groups alone. New organizations devoted to extending Agile values spring up, including the Agile Project Leadership Network, which has acquired over 150 members in less than a year of existence. <http://www.apln.org>

Enterprise Leadership & Scaling

Folks like Jim Highsmith, Jutta Eckstein, Sanjiv Augustine, Pollyanna Pixton, Bob Schatz and others have shifted the conversation from the individual project level to Agile at the level of the enterprise.

Early discussion on the ideal Agile team size (small), and whether Agile methods could scale up, in scope or complexity, have been overtaken by the number of organizations and large projects that do it. The APLN with its Declaration of Interdependence has chosen to focus on moving Agile into new areas of the business; including leadership, organizational systems and cross-departmental relationships.

In my brief survey of trends, Agile leaders offered the following comments:

Mike Cohn, "Companies are not doing small pilots, they're diving in," and "We're seeing Agile coming top-down now whereas it used to all be bottom up."

Todd Little, "It's a movement toward Leadership over Management, Effectiveness over Efficiency, and Reliability over Repeatability" and "Through Agile, I see companies adapting and scaling the development approach to both meet the needs of the project and stay aligned with the business."

Pollyanna Pixton, "There's an increased focus on, and a wrestling with the meaning of, what it means to lead in an Agile organization, i.e., collaborative leadership."

In reference to the values statement of the Agile Manifesto, Todd Little commented,

"I've noticed something that I broadly call dealing with the right. Project leaders and teams are developing processes and tools (e.g. wikis, collaboration environments, etc.) that support Agile interactions and individuals. Through a focus on documentation as a consumable rather than a deliverable, we create documentation that leads to working software. We create contracts consistent with collaboration and Agile delivery, and, in our planning, we anticipate and expect change."

Scaling Agile.

Agile is scaling along a number of dimensions: geographic distribution and global development, number of collaborations with suppliers, combined hardware/software projects including and beyond embedded software, team size, project size, mission criticality, and involvement with legacy systems.

Applying Agile with teams that are distributed, whether across town or across time zones, has become a daily challenge for many project leaders and team members. The advantages of co-location are meeting the reality of the global marketplace. Teams are seeking and developing new techniques to ensure the healthy communication flow that is central to successful use of Agile methods.

In organizations, Agile is being discussed at the program or product management level, as well as the project level. Jennitta Andrea wrote, "A trend that my company is experiencing is an increase in interest in applying agile techniques on legacy application projects (re-platforming/porting, enhancing, stabilizing, etc). We apply agile planning and management techniques (e.g. story-based, test-first porting) as well as an XP development approach. The 'interesting' part is retrofitting tests onto these complex legacy beasts."

Agile as a Commodity and the Dilution of Agile

As Agile becomes better known, as more tools are developed to support Agile and more firms provide their own distinctive flavor of Agile to the market, more ideas about Agile become mainstream. Courses and organizations spring up to compete as "Agile" because it's an attractive, emerging business. Many of them are really good; inevitably, some are not. Promoters may pay lip service and give attention to the appearance rather than the substance of Agile.

Many Agile-related ideas are becoming mainstream among programmers because they're carried along with the tools. Microsoft's new MSF specifically supports Agile development. Microsoft's move into the Agile world is a substantial indicator of Agile's acceptance by the market.

The debate continues about how "strictly" to follow the practices of each method. Some say the practices are not important, as long as a project's approach adheres to the Agile values. Mike Cohn commented, "The practices vs. values debate has been won by values." Other leaders in the Agile camp express concern about the move away from applying the practices. Ron Jeffries says, "As we learn more and more about how to apply Agile in various situations, part of that learning is more about how to water down the original concepts about Agile methods and practices. These watered down versions provide a little benefit, but may also delay the real transforming effects, perhaps forever."

Rebecca Wirfs-Brock wrote:

"I know about one local contracting company who tried to interject agile management practices into a state agency, with mixed results. The development team picked up on the 'easier' XP software engineering technical practices (e.g. continuous integration) much more readily than anyone picked up the 'harder' communication, project and team management practices (e.g. pairing, daily meetings) to support Agile. It was not a smashing success.

"On the other hand, I increasingly find developers embracing cool tools (like really nice development environments, build tools, etc.). This enables them to create and test software as they go...and people are doing that. In fact, one trend I've observed over the past year is that people who develop software for the state of Oregon, for example, are re-treading their development practices (and moving to better software dev platforms) out of necessity. Old timers are retiring. So are 40-year-old COBOL programs and 20-year-old legacy systems. Because of constant pressure to do more with less and respond to numerous annual legislative changes, they are making software that is flexible (and not doing lots and lots of big up front designs). They are developing software in a way that allows them to be much more responsive to their customers. This largely means smart developers with great tools, good discipline, and design smarts. To those folks, probably the open source phenomena (including eclipse as well as many Java open source components) has had more impact on their ability to be adaptable and responsive than anything."

New Challenges/Debates in Agile

Is it the quality of the code or the quality of the coders?

Agile methods thrive on communication and sharing of ideas and practices. They thrive in environments where team members self-organize to accomplish delivery. This begins to highlight the need for people who have both collaborative team skills and excellent technical skills. Ole Jepsen wrote, "It's difficult to get

An argument during the emergence of Agile methods was that they required highly skilled, "star" performers to accomplish their iterative outcomes in short bursts. This argument was countered by the idea that all team members increased in skill through the intense communication and transparency that occurs on a team using Agile practices. One team added the practice of giving responsibility for each task to the least qualified individual on the team to accelerate their learning. The rest of the team formed the safety net to ensure the quality of the delivered code.

As methods are adopted in organizations piecemeal, the dilution in practices is causing some folks to take a second look at who is on their teams and what skills they bring to the project.

Do we value individuals or interactions more?

The first value statement of the Manifesto poses a paradox. The idea that individual and interactions are valued more highly than processes and tools is fine, until a conflict arises between the ideas of valuing individuals or team interactions.

Teams adopt an Agile approach for their work and individuals become more productive. Projects exceed expectations, then plateau. They reach higher than before, but may not reach the potential others see is possible. They never achieve the energizing synergy experienced by many of the early adopting agile teams. Team members gain comfort and competence with the practices, but they still face the difficult shift from individual contributor to team member. They need a shift to valuing their interactions and developing greater collaboration skills.

Simple vs. Complex Tools

In a January 2006 blog post, Brian Marick wrote,

"Tools are important. Many of the Agile Manifesto authors had past experience with Smalltalk and other OO languages. That kind of background makes it easier to think of software as something you could readily change. I don't think Agile would have taken off without semi-flexible languages like Java and the fast machines to run them. Moreover, each new tool—JUnit, Cruise Control, refactoring IDEs, FIT—makes it easier for more people to go the Agile route. Without them, Agile would be a niche approach available only to the ridiculously determined."

Rachel Davies reminded us that there is value in keeping tracking low-tech but visible. For Agile2006 conference, we had several experience reports on how moving back to index cards for tracking had increased team productivity.

Summary

World markets and software development will persist in moving as quickly in the next five years as in the past five. (There's no reason to think they won't.) The earliest adopters will accelerate their use of Agile methods to develop software. Agile methods will also continue to spread into new arenas. Leaders in our community contend that Agile methods have "crossed the chasm" to become a respectable alternative for managing and working in software projects.

As proponents and stewards of Agile, we have an interesting vantage point to watch how awareness of Agile continues to diffuse; how the growing community addresses issues of size, distribution, and leadership; how proponents monitor the "purity" or adaptation of the methods; and how we ourselves respond to change as new challenges emerge.

* The Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others doing it.

Through this we have come to value:

- Individuals and interactions *over* processes and tools
- Working software *over* comprehensive documentation
- Customer collaboration *over* contract negotiation
- Responding to change *over* following a plan

That is, while there is value in the items on the right, we value items on the left more.

<http://www.agilemanifesto.org>

About the Author

[Diana Larsen](#) partners with leaders of software development projects, IT/IS departments and other technical groups to strengthen their ability to improve project performance, support and sustain change, and build collaborative workplaces. Together she and her clients build workplaces that realize business results while developing and dignifying people. As a specialist in the human side of software development, Larsen serves as a advisor, consultant and facilitator to directors, program and project managers, development teams and others. She has special expertise in using Appreciative Inquiry approaches, Open Space Technology and other large group processes, as well as in leading teams through Project Chartering and Retrospectives.

Larsen serves on the [board](#) of the [Agile Alliance](#) and of the Pacific Northwest Software Quality Conference, participates in planning for the XP 200x and Agile 200x conferences, and speaks at several software conferences every year. She's written articles for Software Development, At Work, Cutter IT Journal, and Cutter's Executive Update and e-Advisor series. Larsen is a founder of the Annual International Retrospective Facilitators Gathering.

The other contributors to this article are all members of the Agile Alliance and APLN Boards: Jennitta Andrea, Mike Cohn, Rachel Davies, Jutta Eckstein, Ron Jeffries, Ole Jepson, Todd Little, Rebecca Wirfs-Brock and Brian Marick; and Pollyanna Pixton.

Photo credit: thanks to Victor Zhang <http://photos.viczhang.com>

Bookmark [digg+](#), [reddit+](#), [del.icio.us+](#), [dzone+](#), [facebook+](#) [slashdot+](#)

26 comments

Watch Thread

Reply

Good pat on the back by Hani Suleiman Posted Sep 14, 2006 8:37 AM

Re: **Good pat on the back** by Deborah Hartmann Posted Sep 14, 2006 9:15 AM

Re: **Good pat on the back** by Hani Suleiman Posted Sep 14, 2006 1:15 PM

Re: **Good pat on the back** by Deborah Hartmann Posted Sep 14, 2006 2:36 PM

Re: **Good pat on the back** by Jason Carreira Posted Sep 16, 2006 10:34 AM

Re: **Good pat on the back** by Floyd Marinescu Posted Sep 17, 2006 10:15 PM

Re: **Good pat on the back** by Hani Suleiman Posted Sep 18, 2006 11:12 AM

Re: **Good pat on the back** by Cedric Beust Posted Sep 18, 2006 11:19 AM

Has Agile Gone Mainstream? by Deborah Hartmann Posted Sep 18, 2006 1:29 PM

Re: **Has Agile Gone Mainstream?** by Cedric Beust Posted Sep 18, 2006 1:47 PM

Re: **Has Agile Gone Mainstream?** by Deborah Hartmann Posted Sep 18, 2006 5:37 PM

Re: **Has Agile Gone Mainstream?** by Cedric Beust Posted Sep 18, 2006 6:10 PM

About News by Deborah Hartmann Posted Sep 19, 2006 7:51 AM

Re: **About News** by Deborah Hartmann Posted Sep 19, 2006 7:52 AM

Re: **Has Agile Gone Mainstream?** by Paul Oldfield Posted Sep 19, 2006 8:21 AM

Re: **Has Agile Gone Mainstream?** by Deborah Hartmann Posted Sep 19, 2006 11:33 AM

Re: **Has Agile Gone Mainstream?** by Hani Suleiman Posted Sep 18, 2006 2:16 PM

Re: **Has Agile Gone Mainstream?** by Deborah Hartmann Posted Sep 18, 2006 5:39 PM

Re: **Has Agile Gone Mainstream?** by Floyd Marinescu Posted Sep 19, 2006 12:32 PM

Re: **Has Agile Gone Mainstream?** by Cedric Beust Posted Sep 19, 2006 1:20 PM

Re: **Has Agile Gone Mainstream?** by Chris Matts Posted Sep 22, 2006 12:26 PM

Re: **Has Agile Gone Mainstream?** by Jason Carreira Posted Sep 18, 2006 3:39 PM

Re: **Good pat on the back** by Floyd Marinescu Posted Sep 19, 2006 12:41 PM