

Agile Development: A Manager's Roadmap for Success

Executive Summary

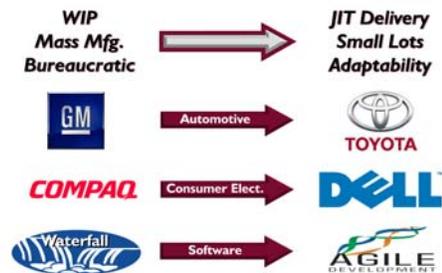
Despite advances in software development technology, just getting software *out the door* continues to be a tremendous challenge. The vast majority of software projects continue to be late, over-budget, and mismatched with market needs by the time they are delivered. Agile Development aims to change this.

Today's rapidly changing business environment demands a more adaptable approach to software delivery. Enabling the same techniques that revolutionized the automotive (Toyota) and consumer electronics (Dell) industries, Agile development is transforming the way software development organizations deliver value and become software *delivery* organizations.

The results from Agile development are overwhelming, and a growing number of successful projects, industry reports, and surveys demonstrate clear results including faster deliver times, lower defect rates, improved team morale, and more satisfied customers. Leading-edge companies such as Fidelity, Siemens, CapitalOne, Lockheed Martin, Motorola, Microsoft, Yahoo, Google, GE, and Cisco Systems are leveraging Agile to improve their business results and to build lasting competitive advantage.

Agile is rapidly becoming the de facto standard software development process for high-performance teams and the real question for software managers and executives is no longer *if* you adopt Agile, but *when* and *how*. Although successfully adopting Agile is not a trivial exercise, Agile done correctly is highly rewarding for organizations and to the individuals involved.

In today's hyper-competitive world, *later* may be too late to adopt Agile development and this *Roadmap for Success* will help you get started. The steps are straightforward: take time to understand the principles of agility, embrace the change that Agile requires, become a champion for Agile projects, foster adoption throughout the organization, and invest in the skills and tools that enable Agile to meet the needs of even the largest development projects. The sooner you start down the path to Agile the faster you will deliver improvements to your bottom line.



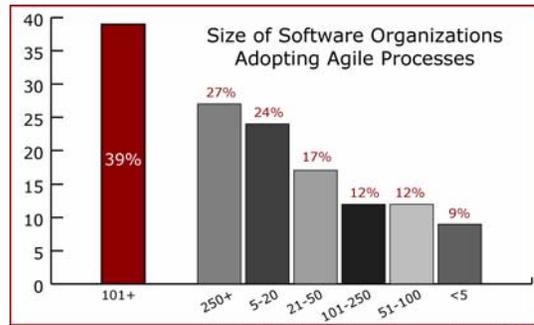
Agile Development: A Manager's Roadmap for Success

The players change but the story seems to remain the same – most software projects are late, over budget, and mis-matched with market needs by the time they are delivered. Regardless of the continuous stream of technology advances in the software field, just getting software *out the door* continues to be our challenge. In a \$250+ billion industry, **one out of every four** dollars is wasted on failed or cancelled projects; with considerably more waste if you include ongoing delays and cost overruns.

Despite mounting evidence that traditional, “big-bang” and ad-hoc approaches to software development do not work, software organizations continue to use the same development processes that have existed for decades. While we may take comfort in consistency, today's rapidly changing business environment fundamentally demands a modern, adaptable approach to software delivery. Similar to the transition to just-in-time (JIT) and lean production processes in both the automotive (Toyota) and consumer electronics (Dell) industries, software development is in the midst of transforming from a rigid, process-driven approach to a more adaptive, business value-driven approach called Agile Development.

Agile development is an umbrella term for a number of iterative and incremental software development methodologies such as Extreme Programming (XP), Scrum, Crystal, Dynamic Systems Development Method (DSDM), Lean Development, and Feature Driven Development (FDD). Especially during the last several years, Agile development has made its way into the software mainstream at companies like

Fidelity, Siemens, CapitalOne, Lockheed Martin, Motorola, Microsoft, Yahoo, Google, GE, and Cisco Systems. These and many other companies are using Agile processes to deliver significant improvements in innovation, quality, productivity, and competitive advantage.



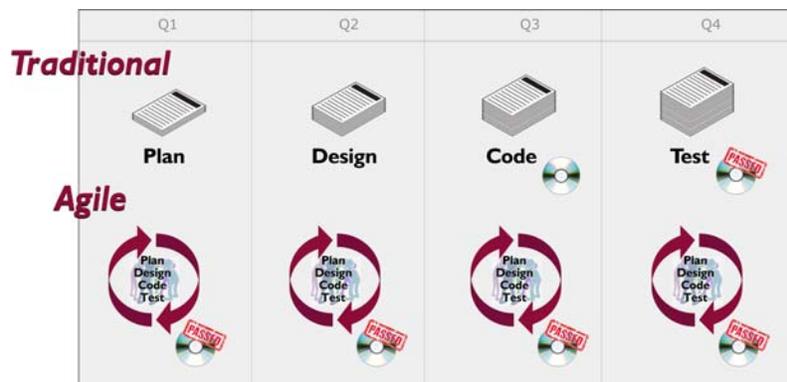
source: VersionOne/Agile Alliance Survey August 2006

Understand Agility

It is important to first understand what Agile Development is and what it is not. Regardless of what you may read or hear from a few over zealous early adopters, Agile development is neither a silver bullet nor a religious software faction. For the vast majority of practitioners, Agile development is simply the natural evolution of software process to support today's accelerated, rapidly changing business environment. Through a lightweight, low-ceremony approach to software development, Agile incorporates a highly disciplined set of management and engineering best practices for accelerating and improving the delivery process. The key tenets associated with Agile development have been around for decades and are based on a series of simple, proven steps.

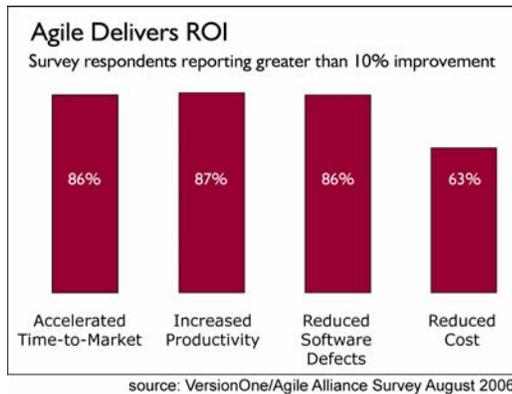
In Agile projects:

- Organizations break large initiatives down into smaller projects or releases.
- Cross-functional teams break projects down into small pieces of functionality.
- Features are worked on collaboratively and delivered in the order of business value.
- Software is planned and delivered in short, frequent iterations that incorporate all aspects of software development – e.g., planning, analysis, design, development, testing, and integration.
- All stakeholders (executives, managers, customers, developers, testers, etc.) are involved throughout the delivery cycle to ensure ongoing alignment with evolving market needs.
- Working, tested software serves as the primary measure of progress.



In Agile development:

- High-level, feature-driven plans evolve over time - replacing speculative, task-based details.
- Continuous, JIT planning methods are substituted for detailed, upfront plans.
- Visibility into project status and progress is based on the undeniable truth of working software.
- Highly simplified and effective metrics based on team productivity trends and historical results help drive much more predictable, reliable project forecasting.



While the benefits associated with accelerated delivery of business value, improved visibility, increased adaptability, and reduced risk are apparent - what is less apparent is how Agile development specifically impacts managers. On a day to day basis, Agile's greatest impact is in the style of management required to be successful.

Agile thrives under a "lead and adapt" management style in which project teams are empowered to communicate and deliver. The freedom to transition away from the traditional task-master role and into a more strategic, collaborative role will appeal to many managers. While the early champions of Agile were primarily developers, recent

evidence shows that the individuals driving adoption are now primarily project managers, development directors, and senior executives. This upward movement within the organizational chart is in direct correlation to Agile transitioning from small teams to large enterprises.

Initiating Agility

The decision to adopt Agile development requires vision, fortitude, and an understanding of the business and organizational benefits that accompany Agile. Managers that champion Agile internally have a strong desire to accelerate the delivery of business value within their companies and to improve the manner in which this value is delivered. The time required to fully deploy Agile processes will vary significantly depending on the size of an organization, its complexity, and the willingness to change. While teams will certainly hit obstacles along the way, the business and team benefits help an organization remain focused on the ultimate goal.

For Agile to be successful, it is important that the use of Agile development not be mandated in a purely top-down manner. It is very important for teams to **want** to use Agile processes to be more successful. To enable this success, it is always advisable to find an enthusiastic, open-minded team and to use them to initiate a pilot project. This initial project will be used to work out many of the internal kinks of doing things a different way.

If Agile expertise exists internally, leverage it to the fullest extent on this team. If not, acquire the right skills by hiring experienced Agile personnel or by engaging well-respected Agile consultants and coaches. In many situations, it is advisable to bring in outside expertise to help introduce change. Regardless of whether true or not, insiders are often perceived as less objective. It will be important to take this opportunity to educate key management personnel and stakeholders on both the *value* and *values* of Agile development. You may also want to help set initial expectations by introducing simplified metrics such as velocity charts and burndown graphs.

Lend the project your undivided attention and give it time to demonstrate the benefits of Agile processes. It is during this formative period the team will learn how to span the chasm that once separated different parts of the organization. Agile teams work towards a common goal as a single unit. There should be little of the us-versus-them thinking that is commonplace on traditional projects. Although the project stakeholders may belong on different branches of the org chart, when united on an Agile team they must work collaboratively to achieve success.

Keep in mind the common pitfalls many teams face when introducing any type of change and ensure you have a team with the appropriate support, experience and resources available to them. The commitment shown to this initial team will establish a foundation for success going forward.

Your efforts as an Agile champion will be rewarded in due course. The success of an Agile pilot project is sure to garner the interest of others. Soon a steady stream of questions will come: How did you get



going so fast with Agile? Why doesn't the team work weekends or pull all-nighters anymore? Why is the defect rate so low? And, just why does everyone on the team look so happy?

Embrace Organizational Change

Yes, Agile development involves change but it is change for the better. Agile requires letting go of the past and embracing new ways of doing things. In the interest of maximizing productivity and quality, departmental silos are retired in favor of empowered, cross-functional teams. Teams are then able to share a single, common vision, relying on one another throughout the process to deliver the greatest business value possible in the allotted time. You should feel confident that everyone who is motivated to change and improve can find a new home in an Agile environment.

Take the role of development manager. In traditional projects the development manager spends days each week maintaining schedules, reviewing Gantt and PERT charts, assigning work to team members and following up on tasks. Agile teams are more self-directed entities that are empowered to detail and deliver their own tasks. As a result, this frees up more time for the development manager to focus on higher value activities such as removing project roadblocks, facilitating collaboration among project stakeholders, coordinating organization-wide activities, communicating goals and progress against those goals, and enabling a healthy, productive working environment.

Better Metrics: Clear Visibility into Real-Time Progress and Status



Agile offers similar benefits to quality assurance (QA) personnel. In Agile projects QA is brought into the process immediately where its benefit is greatest. Test scenarios are fleshed out at same time, or even before, software is written. All the way down at the code level, test-driven development follows hand-in-hand with Agile's commitment to technical excellence and quality. Don't forget that Agile teams must collaborate to deliver tested, working software at the end of each iteration. This forges a strong reliance between developers and testers as they work together closely to achieve the common goal.

The role of the software developer also transitions to a focus on delivering value vs. coding to a "spec". Agile programmers are committed to improving their craft in terms of design, quality and technical excellence. The focus falls squarely on delivering software and less on process and tasks. This results in a much higher degree of job satisfaction. Professionally, developers are now part of a unified team and must collaborate with others (customers, QA, tech writers, etc.) in an effort to maximize the value their teams deliver. Feedback loops are accelerated and accountability is high. Gone are the days of developers working behind closed doors for months on end with a high risk of delivering software that no longer meets market demands.

Generally speaking, communicating how people and departments fit together going forward will give teams a much greater sense of belonging. Given Agile development is as much about people as anything else, taking extra time to treat this change in roles and responsibilities is especially critical for managers.

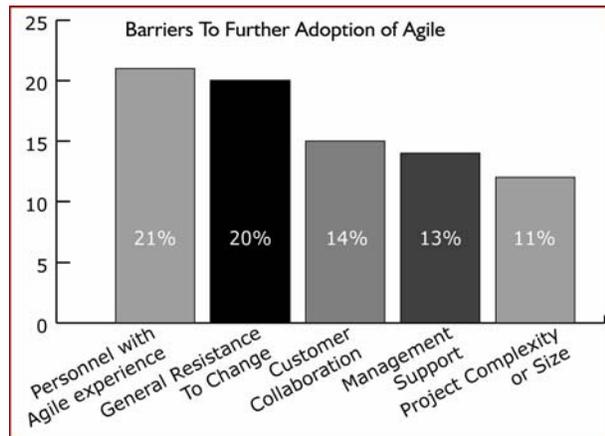
Promote Results Early and Often

Project success begets success, but it takes time to convince an organization that Agile offers a path to broad, longer-term success. Begin generating support by highlighting results within your immediate circle of influence such as your department, division, or larger team. Whether through formal or informal gatherings, focus on both the tangible and intangible value delivered by your Agile projects. For tangible results, communicate simple measures such as delivery dates, defect counts, customer satisfaction, and the reliability of Agile planning metrics to drive home the point. For less tangible results, focus on



communicating improvements in areas such as team morale and the ability to more easily respond to changing priorities.

Remain steady in your conviction. Resistance to change is natural so don't expect overnight acceptance. You may even be able to take advantage of a healthy skepticism by allowing it to help point out less obvious areas of concern. With this type of ammunition, you can focus on communicating successes and dispelling myths in these specific areas.



source: VersionOne/Agile Alliance Survey August 2006

Broadening the adoption of Agile will certainly require additional executive management support. Take special care to educate management on the benefits and results, as well as the challenges associated with Agile development. Setting realistic expectations is critical – Agile should not be perceived as a silver bullet, but instead simply a more modern, disciplined approach to software delivery which is better able to handle accelerated delivery cycles and changing priorities. Rely on the successes of early projects to help illustrate the value of Agile processes.

Scaling Up and Out

Following these initial successes, the enthusiasm for Agile will typically be running high yet it is advisable to continue a rollout strategy that incrementally deploys Agile within the organization – not necessarily one project at a time, but in planned chunks. Aligning change with the speed that your organization is capable of accommodating change is important. Some organizations are able to quickly adapt and change course, others are not quite so nimble.

Throughout the rollout process, continue to assess projects and opportunities which are open to or in need of change and have the most to gain from Agile. Do your best to align these opportunities with the overall goals of the organization and that of key stakeholders. By doing so you are likely to garner additional support as it becomes apparent that Agile positively impacts the bottom line. Have a publicized, open-door policy to daily stand-ups and planning sessions for personnel on non-agile teams. This will give them a non-threatening introduction and the ability to observe the Agile process.

Continue building credibility and facilitating success by seeding teams with the necessary expertise. You will now have experienced personnel from within earlier projects to choose from and utilizing the expertise of a coach or trainer experienced in scaling Agile would be of tremendous benefit to the organization. Ensure that adequate training is provided and that teams *believe* they have been provided the skills required to succeed. As projects progress, broadly share lessons learned and regularly communicate wins as they occur. As important, quickly learn from failures and adjust course accordingly.

As you begin to scale Agile to larger and larger projects and more distributed teams, the challenges obviously amplify yet the core tenets of Agile do not change. Agile methods promote taking large projects and breaking them down into a coordinated series of smaller projects staffed by smaller, cross-functional teams. This workload decomposition places a premium on effective collaboration. Additional complexity is found in the continuous change in plans and priorities associated with Agile projects and must be coordinated across teams.

As with any significant process improvement deployment, consistency across the organization is vital. Reliable, real time visibility into project status is critical for broad-based stakeholder confidence and

acceptance. To provide this type of global insight typically requires a single, consolidated view of plans, priorities, and progress.

Tooling for Success

Traditional lifecycle management tools are unable to meet the needs of Agile organizations because they are based on a fundamentally different approach to planning and measurement. The inherent complexity associated with these tools has further inhibited broad acceptance. To facilitate success in an Agile environment, effective tools must embrace and promote the following characteristics:

- *Iterative, feature-driven development* – Many teams continue to try to force-feed a series of traditional tools that do not facilitate planning and tracking by release or iteration, do not easily enable changes to plans and priorities, and do not use features (e.g. requirements, user stories, product backlog, features, ...) as the primary planning asset. To properly support Agile development, tools must support the actual planning and management practices promoted by Agile.
- *Integrated lifecycle management* – Instead of using different tools for different phases, Agile development follows a tightly integrated process that coordinates high-level feature planning, detailed task and test planning, defect and test measurement, and overall project tracking. Especially in an environment that promotes stakeholder visibility, tracking project information in multiple tools can inhibit accurate, real-time insight into projects.
- *Cross-functional teams* – True support for cross-functional teams means consolidating and facilitating the project planning and tracking needs of customers, product management, project management, programmers, testers, etc. in a single environment for improved collaboration and consistency.
- *Flexible project and team setup* – No two organizations operate in the exact same manner when it comes to organizational structure, terminology, product planning, project scheduling and tracking, and reporting. Any Agile management tool should accommodate organizational requirements and enable them to define unique departmental rollups, product line breakdowns, cross-project reporting, and potentially many teams working on a single project.
- *Agile metrics and reporting* – Agile tools measure a fundamentally different set of metrics than traditional management tools. Business value delivered by the team is the new yardstick, not hours worked per person per task. Burndown graphs, velocity charts, burn-up charts, etc. must be supported out-of-the-box.
- *Stakeholder ease-of-use* – While complex needs may arise, the necessity to provide team members a simple, straightforward planning, tracking, and reporting system is critical for broad-based acceptance. Like Agile development, the more simplified the tool the better. Most importantly, tools should never replace the benefits achieved from daily planning, face-to-face communication, product reviews and retrospectives. An Agile lifecycle management tool is only as good as the process it facilitates and the people that use it.

Just like Toyota and Dell could not deploy traditional manufacturing systems to support their JIT and/or lean production environments, today's organizations cannot haphazardly patch together tools based on outdated planning and tracking methods to succeed in an Agile environment. Agile tools, such as those offered by VersionOne, provide an integrated, out-of-the-box solution for adopting and scaling Agile development. Just as important, they are simple enough for everyone to use and flexible enough to accommodate the rapid change to plans and priorities that are embraced by Agile.



Now is the Time

The question is no longer *if* you move to Agile, but *when* and *how*. The steps required to go Agile, while not easy, are fairly straightforward. Research, investigate, and learn as much as possible about Agile. Identify a project or team in your organization where you believe Agile can thrive. Build the case for a pilot project. Educate executive management about the approach, benefits, and challenges of Agile. Kickoff the pilot project and protect it from the naysayers and skeptics. Invest in your people and build their skills through appropriate tools and training.

You'll soon find that Agile processes result in better quality software delivered in a shorter timeframe than ever before. Your development team will be happier with the sustainable pace of Agile, and your company's stakeholders and shareholders will be convinced by Agile's impact on the bottom line.

