

viewpoints

Agile Software Development

A new path to software development

Ryan Shriver, *IT Performance Improvement solutions leader*, discuss how Dominion Digital helps clients build next generation software products.

As a process, software development has few if any equals. Extraordinary complexity accompanies every challenging step in a choreography of people, process and technology. Teams are tasked with distinct phases - design, planning, construction and testing - resulting in some to equating it with bridge building or other engineering-type endeavors. Software Engineering itself has emerged as a discipline, with University professors and industry experts worldwide learning and teaching in tandem.

The nature and inherent difficulty of software development has led to efforts to devise new methodologies that diverge from traditional ideas of project development. Extensive up-front planning and design, long construction, testing and implementation phases that conclude after a lengthy sequential cycle is an often-used approach known in the industry as 'the Waterfall Lifecycle.'

In an effort to create a more supple process, better software, and to deliver business value more quickly and efficiently to customers, many software developers are now employing a radically different, customer-centric approach to software development. "Agile" or "iterative" methods of software development are gaining new adherents as stakeholders gain insight into the key benefits that Agile delivers: quicker time-to-value, less exposure to risk, a more adaptive process and better collabora-

tion between product development teams and business people.

Q: What kinds of things are going on in a company prior to your involvement?

AOne is that market pressures are starting to bear down. It's often a time-to-market consideration. Support may be in place on an existing product or product set, but that product may be facing diminishing returns. Internally, the decision is made to build a new product and then you really are looking at a critical time-to-market issue.

Q: What about the internal development process?

AOne thing we notice is that there are not a lot of defined processes in place to build these products. They've been built over the years and the rollout has sometimes depended on individual heroics, which is risky. Sometimes competition is playing havoc with a particular organization and they feel they need to get their product to market to satisfy existing or new potential clients.

Q: What type of companies are these?

AGenerally, we work with companies directly involved in software development, but this isn't always the case. That can mean everything from a small vendor to a much larger company. We work with firms in industries including financial services, defense, publishing and education.

**Q: What are your customers trying to achieve?
What's the goal?**

A The major goal for any custom software development engagement is predictable delivery of business value. For product companies, this typically means bringing new or updated products to market, typically using more leading-edge technologies.

Q: Why does an internal development team need your help?

A The software that we deal with ranges from departmental to enterprise systems. The end-users – the customers of our clients – typically pay substantial amounts for the software and corresponding implementation services for the enterprise systems. Understandably, these clients have a say in how and when the software gets developed, and what features are the most important.

We support the internal development teams by bringing a light-but-effective structure to their processes and by working with all stakeholders to help them get clear requirements, resulting in solid solutions. We also help the team develop their system design and architecture and set up automated development and testing processes. Finally, we help leadership establish the type of team that can succeed.

Q: So this is client driven?

A In some cases, yes. You have important stakeholders who want to see new features added into the next release. The development team calls on us because we have experience across many types and sizes of projects and many enterprise technologies. They need us to help speed up the process. That's just one example. We're often called upon to do a lot more, like helping to build next generation software, which is a more market-driven process.

Q: Are you coming in with your own way of development?

A In a sense, yes with Agile development, but not with a "right way" of doing things with hard-and-

fast rules. We adapt and tailor our processes to the team depending on the company's culture, project and team size and development experience. But yes, we absolutely are doing this differently. Our approach is quite a departure from the rest of the industry, although this method is starting to gain traction. We promote an iterative, or Agile approach to developing software – which is the exact opposite of the Waterfall method that's been around for twenty or thirty years. The Waterfall method is where you gather all the requirements up front – you design, build, test, and then you roll it out. It's one big cycle that can last six to twenty-four months. Twenty-four months before you see any value is an eternity for most organizations.

Q: What's different about the Agile method?

A You see value much, much quicker. Agile is an iterative method of development that moves through the same four steps – requirements, design, construction, testing – but they are implemented in a time-boxed, fixed-budget iteration process. This means that at the end of each iteration, the business sees some real value for the 2% to 5% of budget spent. So they now have a real choice; if they see value and they want to continue, they can choose to continue investing in the project. This is very, very different from the Waterfall approach.

Q: Can you give us a real world example?

A One of my clients is a software product company that's been in business for over 35 years. They make a software solution that is used in the financial services industry. Another client is a subsidiary of a large defense company that develops systems for the marine and defense industries. In both cases, we're helping them build their next generation product.

Q: Is this a full redesign?

A In my case, yes. From the ground up, we're redesigning their systems to work with relational databases, web browsers and rules engines. The new systems support Service Oriented Architecture, a newer approach to enterprise integration.

Q: Can you walk us through the early stages?

A We sit down with all the stakeholders and ask them: “What are the top ten most important things that this system has to do?” So as we work through each iteration, we measure and report back. We can tell our customer, “We’ve improved this metric that you deemed important by this percentage.” Typically, when we can clearly show results like that, the business will agree with us that they are getting value.

Q: Who are the stakeholders?

A There are stakeholders inside and outside the company. Inside, it’s really the C-levels (CEO, COO, CIO, etc.) and executive business sponsors, the people who fund and support the project. Externally, you’re talking about the company’s customers. If a customer is a Fortune 500 company, that customer may have a lot to say about what order we build in new features and so on. Other stakeholders include product managers, sales, marketing, project managers and real, live users of the system.

Q: Are they trying to change a series of products or just one?

A In one example, they have multiple products developed over their history using different technologies. The insight that they’ve had, and it’s an important insight, is that across these different products, there are vastly different implementations for similar concepts. So you have different security models and system controls in each product, but they were all produced by the same company.

Our client realized that having these on a common platform made business sense so they could cross-sell clients other products in the product line and ensure the client had solutions for their processing needs.

Q: So the goal is what exactly? What do they hope to have at the end of the day?

A In this case, the goal is to have a world-class integrated solution. So instead of having separate code bases for all the product lines, they want multiple prod-

uct lines to exist on a single code base. So that way, the common things like customer service transactions and security models all work the same, no matter which of the products you license.

Q: Are there additional upsides to Agile methods?

A Joint collaboration is a big plus. The Agile method inherently promotes a great sense of collaboration between all teams. Traditionally, you’d have the requirements team working in isolation, working out all the requirements a product should have. By the time the requirements team sits down to talk to the developers, the developers have less understanding how they arrived where they did and why. Our approach is to have these teams working together every day of the project. In fact, they often sit side-by-side with developers. Our process values face-to-face communication among all stakeholders.

Q: What other gains are your customers looking for?

A A product that sets the industry on fire and can be delivered in a rapid fashion. In addition, our clients also want to build a team that can turn out new releases of a product on a timely schedule. That way they can continue to easily add and integrate features to support internal innovations or to respond to competitive pressures. They want a mature software development process. They don’t want, or shouldn’t want anyway, to depend on individual heroics. And, of course, they want a strong return on their investment.

Q: So what’s an iteration? How long do they last?

A An iteration is typically one to four weeks. One estimate that I like is that each iteration should cost around 2-5% of the overall budget. What you’re basically saying is that we’re going to break the overall budget down, and either by time or by budget or both, and at the end of each iteration, we are going to deliver value back to the stakeholders. We’re going to give you something that shows we are delivering value.

Q: Any other benefits to iterations?

A There are many, but one important advantage is that the process is repetitive. Your teams are working together over and over again on these iterations and they naturally become more cohesive and effective. At the end of each iteration we do a post-mortem and talk about lessons learned and develop an ongoing process of continuous improvement.

Q: So in one sense you're closing the feedback loop.

A Absolutely. You're able to receive feedback from your stakeholders and correct your course much more quickly. You're getting the software into the hands of the people who are going to use it so they can then say, "Yes, this meets my needs, or no, it doesn't."

Q: So what would a world-class outcome look like?

A The client is seeing the system working in production and adding business value. They can see this value in a matter of weeks or months. Our approach means the team, as a whole, rallies together. You don't have separate camps.

Q: Does the company have to change the way they build software?

A Most likely, yes. A client may have a suite of products that have been developed over a number of years with a small team. They now have to compress that and do what used to take many years in just a few years. This brings up a number of issues like the size of the company in terms of people and infrastructure.

Q: Let's say an executive or a manager in a software development company needs outside help to develop a product. What forces shape his or her choices around methodology and approach?

A No one is ever going to get fired for initiating a Waterfall project, especially on a large, mission-critical project. The truth is that most software development projects are Waterfall in nature. It's seen as safer and more traditional. Big companies often mandate the approach.

Q: If Agile is so good for the customer, why isn't there more of it?

A It's catching on. But right now there's a scarcity of published case studies that show this process works on a large scale. Imagine a project with five developers and a \$50,000 budget, that's one level of risk. But if you have a \$10 million project and you're dealing with half the enterprise, advocating an Agile approach may meet a lot of resistance. It takes courage to lead initiatives on new ways of doing things, especially on large projects.

Q: Where is the resistance coming from?

A Some managers or directors may want to take this approach but don't have the autonomy. Others will be convinced this cannot scale up to a large project. In other situations, the feeling is that Agile won't fit in a cultural sense. The company or development team has been doing things the same way since time immemorial and isn't willing to change.

Q: From a cultural or change management perspective, how do you counter that?

A Well, even though there aren't a lot of case studies, there are some. There's one that involves British Telecom where they rolled this out to 8,000 IT employees. In this study, once people got inside of this process and saw how much collaboration and team unity it fostered, they decided they loved it and didn't want to go back to the old way of doing things. That's a pretty convincing study. Jutta Eckstein has a new book "Agile Software Development in the Large" and she's publishing some results of doing Agile on larger projects and implementing it in larger organizations. Adapting Agile to larger teams is an area of interest for me right now, based upon my experiences over the past few years.

Q: How about autonomy? Someone in an organization gets passionate about this but doesn't have the authority to initiate it. What happens then?

A Start with a small, low-risk project that you can control. Build some familiarity with the process and some success with it. Then you have a track record.

Companies that haven't worked with this approach are just much better off working through some smaller projects and building on that. If it's going to be very large project, the same approach can be taken, but you have to scale up to the large project by starting small and growing the teams, processes and software along the way. This gets back to the 2% - 5% of budget. If the smaller teams aren't delivering value early, the chance of a much larger team delivering value later is not good. The sponsors may want to re-evaluate the situation in this case. Better to know early and be in the position to change things, than to know too late and learn some hard lessons, after investing a lot of money and precious time.

Q: So let's say you have a stakeholder that wants to use an Agile approach. How does he or she sell it?

A They are going to sell the time-to-market issue. They'll say, "We're going to get you solutions sooner. And you're going to be able to use these solutions sooner. We're going to deliver value incrementally and solicit your feedback and guidance iteratively." Too many businesses have been burned in the past by 12 to 24-month cycles. They get to the end of a long project and they find problems and those problems create a mindset that seriously erodes confidence in the ability of IT to deliver quality software products.

Q: So who is using Agile?

A It's the more progressive companies that are willing to use an Agile approach. Some larger companies are experimenting with it on lower-risk projects and having success, but most enterprise projects are still done using Waterfall. But Agile is rapidly gaining adherents. To some degree, Agile still goes against the grain in terms of how people normally think of this process.

Q: Is risk mitigation a selling point?

A No question. Anyone selling this can say definitively, "We're not going to find ourselves fifty percent of the way through, only to have to say, we can't implement this." The Agile method encourages you to tackle high-risk issues sooner. So if something doesn't work

out, you've only exposed a relatively small portion of the budget.

Q: So leaving aside Agile methods, how does Dominion Digital differ in terms of delivering custom software development?

A You can't leave out the Agile method we use. It's integral to what we do and who we are as a company. In everything we do, we are focused on delivering rapid and dramatic results and Agile is an exciting and relatively new method that helps us do this. From a competitive standpoint, many regional firms are still using Waterfall methods and those that have adopted Agile, have only worked on small projects. Dominion Digital has experience delivering this solution to very large firms on high-stakes projects.

In addition to helping with the software development processes, we also have leading technical consultants with many years of experience that provide technical leadership on a variety of platforms. Some of our consultants are industry leaders, authors and speakers on Java and .NET technologies.

Q: What's the last word on this, Ryan?

A Agile delivers business results early and often throughout the process. Dominion Digital uses the most current and up-to-date processes, Agile methods and tools to build software. Finally, when you engage us, it's like hiring a mountain guide. You're getting technical leadership and guidance that helps get you to the places you want to go.

For more information on our solutions, contact us at 1.877.334.4266 or solutions@dominiondigital.com. Or visit our web site www.dominiondigital.com.