Agility At Scale and Distance

A Video Brief by Frost & Sullivan

Good day and welcome. I'm Susan McNeice with global research and consulting firm Frost & Sullivan, here to discuss the business of software delivery using the Agile methodology. We'll consider Agile in the context of today's demanding environment, even with large, decentralized and even geographically dispersed teams.

(visual) So just what is Agile? Created 10 years ago by a small group of technology professionals, Agile was born from the idea that software could be delivered incrementally, improving one's business in as little as two weeks. These professionals believed there were alternatives to the waterfall methodology without compromising quality, creativity or business value.

Agile starts with the idea of a multi-disciplinary team of design engineers, software developers, test experts and of course, business stakeholders. The team culls a small set of requirements from a larger inventory of known business needs, and working together as a unit through all the phases of the development lifecycle, they create a layer of commercially usable software in just a couple of weeks.

The Agile approach is regarded by many as a way to successfully post incremental gains to one's business. It is inherently less risky than the traditional methods because the business

stakeholders are always involved, so the product is more likely to be accepted by the end-user community.

Now, popular belief has it that Agile is best implemented in small, co-located teams of just 10-15 members. But what if your team doesn't look like this? Can you still have employ Agile methods? What do you do when the group is large, say more than 75 or even 100 members, spread out across the country or around the world? How do you keep the communication and the artifacts synchronized, and allow the business owners and software delivery team to make informed decisions?

Lastly, how do you seamlessly fold in a third-party provider to the overall team and still deliver according to the needs of a given project as well as company development standards?

As we'll see, it is possible to have an Agile environment, even with these challenges. With today's expanded set of integrated software delivery tools for requirements collection, quality management, build, test and team collaboration, such as those offered by IBM Rational, these challenges can be reduced and even minimized. Your business can employ disciplined Agile methods, even in large, complex projects, using "just enough" process, and 'just enough' methodology and tooling to make an Agile approach efficient and effective, without compromise.

Let's look at an example. Our team is charged with modernizing a customer service application, adding e-commerce and self-service features. In Brussels, there are 20 team members responsible for business requirements, interface development, end-user acceptance testing and overall program management. In Boston, another 25 members perform independent

integration tests as well as some program management. And another 50 developers employed by a third-party firm work in Bangalore. They maintain the original application, and will develop the new software, including the testing of new and changed modules.

The Chief Product Owner in Brussels interacts closely with the product owners in each location to capture both 'big picture' and individual feature discussions in a requirements management application. The delivery collaboration platform escorts the requirements through review and approval cycles, keeping the team and their project artifacts synchronized at all times. The team members in all three cities have real-time, iterative discussions directly with anyone else on the team as needed, answering questions and clarifying the operation of the UI in words, drawings, photographs and other artifacts.

The Program Manager in Boston reports to the overall program executive in Brussels, working closely with the Chief Product Owner, the informal subteam leaders, called Scrum Masters, and with the outsourcing firm to manage the contract. The Scrum Masters of each subteam facilitate 'daily stand up' meetings via a voice or video conference call. And throughout the day, the Program Manager uses the delivery collaboration platform to check on the status of individual items, visualizing each team's activities and advising them as necessary.

For the offshore service provider, this same delivery collaboration platform allows immediate feedback on questions, helping them to keep to that two week schedule. The capture of key metrics in the platform allows the offshore partner to tightly manage resource allocation, and

supply progress data to the program office in a transparent, well documented manner, making the overall relationship easier to manage.

The development teams make their working builds available to the testing team in Boston. The Boston team then runs integration and investigative tests to validate the overall quality of the system, and ensure the development teams are conforming to corporate coding standards. Just enough process, supported by automated, integrated workflow and tooling allows these independent testers to manage their work and to report and prioritize any defects back to the development subteams. Then the subteams treat these defects as requirements, returning them into the overall backlog where they are inserted into the next iteration and tracked to successful completion.

(visual) Does it work? Consider these two customers using IBM Rational software. Fujitsu New Zealand has improved customer satisfaction while enjoying a 20 percent savings on time needed for projects. And a project manager for North America's Emerging Health I.T. says her team has reduced the time to delivery by more than 85 percent, and with a third fewer people.

Today's software environment is driven by the mandate to deliver value at the speed of today's businesses, and there really is no "one size fits all" method for implementing agile software delivery processes. But today's integrated, collaborative software delivery platforms give you the means to operate in a lightweight framework in less time and with better results, allowing you to make Agile Software delivery a reality for your business, even when your teams are large, complex or geographically dispersed.

Thank you for joining us today in this exploration of an Agile approach to software delivery. For

Frost & Sullivan, I'm Susan McNeice. Good day.