

AGILE SUCCESS AT THE ENTERPRISE LEVEL



BY ADAM HERNDON

IF YOU'VE SUCCESSFULLY USED Agile methodology at the project level, you know its unique benefits:

- Smaller, more frequent releases
- Higher quality product
- Better ability to respond quickly to changing internal and external needs
- Improved return on investment

It's tempting to wonder whether you might enjoy these benefits elsewhere in your organization. Is it really possible to think beyond agility in a single project and grow an Agile enterprise? Yes ... sometimes. But before taking that leap, consider what Agile is not:

- A magic bullet that will solve all software delivery woes
- A simple change to make in a large organization

One of the key words used to describe organizations that successfully implement Agile across the board is "different." It's true. Agile is applied differently at the enterprise level than at the single team level. As with any type of delivery methodology, scaling increases risk, complexity, and size. So before you make the decision to commit your organization to Agile on a large scale, take into account these 10 key considerations.

10 KEY CONSIDERATIONS

1. AGILE: THE RIGHT FIT?

Take a hard, honest look at your organization. One of the first steps to scaling Agile at the enterprise level is determining whether Agile is the correct methodology in the first place.

Large development teams dealing with mission critical systems that rely on strict regulation and high predictability are probably better suited for traditional methodologies. Heavier focus on plans, documentation, and processes allow traditional methodologies to coordinate and communicate the necessary processes/deliverables across large teams. On the other hand, if your organization needs to accommodate rapid change, has small/medium teams, and focuses on developing in quick iterations, Agile may be a great fit.

2. SOLUTION ARCHITECTURE (SA)

Given that Agile teams move quickly to deliver shippable code, it is critically important that some person or group is able to see the big picture of an integrated solution across all teams. This is not a new concept for integrated enterprises, and it is absolutely essential to an Agile one. SAs look at the organization from an all-encompassing view that shows how each part relates to others and to the whole, ensuring that priorities and dependencies are followed. They help IT teams deliver business goals. Without an SA, separate teams often end up creating solutions that do not meet the dependencies of other integrated teams; and ultimately the integrated solution has major flaws.

3. UNDERSTANDING CAPACITY, REMOVING BOTTLENECKS

A successful Agile team's capacity is not based solely on development resource needs, but includes resource needs for every project phase. Understanding the capacity of the entire team – business analysis and quality assurance included – makes it possible to truly measure and plan for the throughput of the team as a whole. And understanding each team, in turn, enables

an accurate determination of the entire enterprise's capacity. Only then can a reliable decision be made as to the ability to take on more, integrated work. Leaving out the testing phase, for example, when planning capacity, will effectively defeat the purpose of Agile, creating a QA backlog that leads to costly delays and risks.

Successful enterprises often use a demand management office to help determine capacity, set priorities, and define the schedule of an IT organization's ability to take on work. This team knows how many people will be needed for each phase and can place the right resources to support Agile all the way through each project. With this view of the entire company's resource population, it can plan more accurately by identifying and removing any bottlenecks.

4. RELEASE MANAGEMENT

A piece of software is not complete until it is released into the client's production environment and in the hands of the intended user. While it may be a (relatively) simple process for a single team to develop, test, and release a product, releasing becomes increasingly difficult when operating on a scale that involves integration between teams and increased team size, project size, and problem being addressed.

Release management is important to Agile on any scale, but never more so than when scaling Agile to the enterprise level. Because Agile is all about delivering value quickly through smaller, more frequent releases, it takes tight coordination among all enterprise teams to ensure satisfactory delivery to the client. This means processes and communications must be in sync among development, testing, operations, support, marketing, and sales to be ready for each release and make sure the client is, too.

5. CROSS-TEAM COLLABORATION

You already know it, but it bears repeating: When teams collaborate, integrated solutions are far more likely to succeed. That goes for any methodology, and

Agile is no exception. For the Agile enterprise, release management and solution architects can help ensure good collaboration. To strengthen collaboration even further, Agile enterprises may also tailor cross-team scrums to the project and organization, based on functional groups (BA scrums, development lead scrums, scrum-master scrums, etc.). These meetings may or may not happen every day; the rule of thumb is that they happen as often as they are needed. And remember that for integrated solutions, dependencies cannot be over-communicated and should be clearly stated in all stories, iteration plans, testing plans, and release documents.

6. AUTOMATED REGRESSION

As you've noted if you've undertaken a thorough assessment of capacity, quality assurance can be a bottleneck. Many Agile teams fear that scaling to an enterprise level will slow down progress. While this might be true to an extent, automating regression testing will reduce QA time and time to release. Maintained focus and investment in automated regression can not only increase QA velocity, but also enforce better code quality – meaning Agile can deliver on its promise of a higher quality product.

7. CONSISTENT PRIORITIES

As easy as it sounds, this can be tough for an organization. If, rather than following through on a determined set of priorities, the IT organization gets in the habit of reacting to whomever yells loudest for something new, a cycle can be produced in which very little business value is produced. This is not to say that those priorities cannot change. If there's one thing we know about Agile, it's that it adapts well to change! The key is not to get carried away – to figure out when and how often to change priorities. That way, business goals can still be met and the IT organization can deliver complete sets of value.

Demand management is one way to help an organization set and maintain priorities as they relate to resources; and with its high-level view, it can take into account the fact that priorities at the enterprise level are not always the same as the priorities perceived by each team. (Prioritized work does not impact every team all the time.) Interrelated priorities make for complexity in scheduling work for integrated solutions. But that complexity is greatly reduced when each team knows what to work on and feels confident that priorities are set.



“The organization as a whole must adapt how it thinks and operates.”

8. SELF-MANAGING TEAMS, CONSISTENT FRAMEWORK

In an Agile organization, there are multiple teams working at the same time on different phases. And while it is important for each team to be able to self-manage and adjust to its own needs, it's also vital that a framework be established to ensure all teams basically speak the same language. This means release schedules, estimate values, metrics, general processes, etc. should be the same across teams. The risk of operating without this consistency is that teams become more isolated and less transparent. At the enterprise level, a team that is increasingly doing its own thing will collaborate less and ultimately introduce more risk to integrated solutions. With a consistent framework, that risk is reduced and resources can be more flexible. An individual may move from one team to another and be able to get up to speed quickly.

9. WHOLE ENTERPRISE INVOLVEMENT

Commitment: It only really works when it's 100%. Just as teams within a project need a consistent framework, integrated teams in an enterprise need a common methodology. Mixing Agile with more traditional methodologies can create scheduling and dependency nightmares. Imagine if your centralized CRM system used a traditional methodology that released quarterly and your other application teams used an Agile methodology that released every month. Any integrated solution that involves or impacts the CRM system would be forced to deliver quarterly. This would also open up risks for scheduling mistakes, collaboration missteps, and the inability for Agile teams to deliver value quickly.

Going back and forth between the different methodologies will cause conflicts between the amount of required documentation and the schedule of activities within the two different software release lifecycles. Ultimately, the team running in a traditional methodology will force the other Agile teams to conform. So if you're thinking about scaling Agile up, make sure you're ready to go all the way.

10. SCALE SMART

Practices that might have worked for three 8-person teams might not work for ten 20-person teams. Embrace the continual improvement that is Agile and evolve meeting practices as needed. For example, if a daily standup for all teams is needed, does every team member need to attend? Probably not. A scrum-of-scrums is one way to scale intelligently, ensuring checkpoints without bogging down the whole team with constant stopping and starting. As with any Agile teams, retrospectives will allow for the team to reflect and improve. Just make sure retrospectives are cross-team meetings (project retro, development lead retros, etc.), so teams within the enterprise can learn to collaborate more efficiently and effectively.

SO ... ARE YOU READY?

Introducing Agile methodologies to a whole enterprise takes commitment not only to a methodology, but also to a philosophy that extends to teams, communication, training, resource management, leadership, and project governance. The organization as a whole must adapt how it thinks and operates. It's different, for sure. But if it's the right kind of different for your enterprise, it can transform how you serve your clients.

.....
ADAM HERNDON (adam.herndon@jabian.com) is currently a Director at Jabian, where he specializes in IT delivery methodologies. Throughout his career, Adam has been helping organizations leverage “the right” IT delivery methodologies to deliver the highest possible value. He is currently focused on guiding organizations through Agile transformations and scaling Agile to the enterprise level.
.....

© 2013 Jabian, LLC. All rights reserved.
We'd love to hear your feedback on this article.
Please send your comments to journal@jabian.com.