Why Agile, Why Now

By

David Schroeder, PMP, PSM

"Accelerating Corporate Transformation," [i] is the needed approach in business today, according to Harvard Business Review contributor and Corporate Transformation Resources president, Robert Miles. He emphasizes moving fast with a collaborative team. The changing economy is driving the need for new initiatives and business model transformation. Success relies on rapid innovation, business wide collaboration, and ability to respond to changing markets. The tools and technology used must allow fast implementation of ideas and immediate feedback on results, providing the ability try new ideas, fail quickly, learn, innovate and try again.

Traditional Approach

Great ideas can come from any place in the organization and even outside the organization. "Launching a new initiative within a large corporation has been a hit or miss exercise," [ii] states Steve Blank, a Stanford associate professor and National Science Foundation principal. Following a method in use for decades, you start with a great idea, research the idea, develop a business case, present it to the funding body, build a team, and start building.

In many cases, extensive research was performed over a period of many months. Corporations were careful to only invest in those ideas where the return was assured and the return was significant enough to warrant the investment. This approach was certainly appropriate given the often high investment costs and potential risks. Since new initiatives often have a significant software component, either in the product itself or in the product's business model, the cost of large software development and long lead times to first deployment contributed to the high investment cost and risk.

One of the most common software development methodologies, Waterfall, mirrors the traditional product development model with extensive up-front research and evaluation, significant time in planning and design, long development and testing times, and huge deployment efforts. Because of the large investment, there was a strong need to "get it right the first time." Unfortunately, the opposite was true. "The failure rate of large [Waterfall] IT projects with budgets exceeding $1 million was found to be almost 50% higher than for projects with budgets below $350,000," [iii] Given the need for rapid innovation and the important role software plays in innovative success, a new approach is vital.

Innovation Agility

Business schools are adapting to the changing needs for innovation and they have begun teaching "lean start-up" methods for developing new initiatives. Lean methods can be effective in entrepreneurial companies as well as large enterprises. Lean reduces the complexity to improve success. Lean favors experimentation over elaboration, customer value over business objectives, and iterative design over large up-front analysis efforts. Initiative visibility is found in tools like the "business model canvas" developed by Osterwalder and Pigneuer, to visualize the new business model created by the initiative, in a nine box chart. [iv]

Business Model Canvas

Key Partners (8)	Key Activities (6)	Value Proposition (2)	Customer Relationship (4)	Customer Segments (1)
Who are our Key Partners? Who are our key suppliers? Which Key Resources are we acquiring from partners? Which Key Activities do partners perform?	What Key Activities are required for: The Value Propositions? Our Distribution Channels? Customer Relationships? Revenue streams?	What value do we deliver to the customer? Which one of our customer's problems are we helping to solve? What bundles of products and services are we offering to each Customer Segment? Which customer needs are we satisfying? What is the minimum viable product?	What type of relationship does each customer segment expect? Which ones are established? How do we integrate them? How costly are they? How do we get, keep, and grow customers	For whom are we creating value? Who are our most important customers? What are their behaviors? For what are they asking?

	Key Resources (7)		Channels (3)	
	What Key Resources are required for: The Value Propositions? Our Distribution Channels? Customer Relationships? Revenue streams?		Through which Channels do Customer want to be reached? How are we reaching them now? How are our Channels integrated? Which ones work best? Which ones are most cost-efficient? How are we integrating them with customer routines?	

Cost Structure (9)	Revenue (5)
What are the most important costs inherent in our business model? Which Key Resources are most expensive? Which Key Activities are most expensive?	How do we capture value for the enterprise? For what value are our customers really willing to pay? For what do they currently pay? What is the revnue model? What are the pricing tectics?

http://businessmodelgeneration.com/canvas

The tools and technology used must support the lean approach. There must be fast implementation of ideas and immediate feedback on results, providing the ability try new ideas, fail quickly, learn, innovate and try again. The Agile Software Development methodology mirrors the Lean approach. Agile promotes focus on customer value, for both internal and external customers, collaboration amongst all participants, and it delivers rapid iterations of the business initiative and its business model. "It is no longer enough to have a great product," states Osterwalder, "It is the combination of a great product and a great business model that produces success." [v]

Agile replaces the long and documentation heavy phases and deliverables with simple tools and lightweight documentation such as the Business Model Canvas. To start the development, analysts can quickly translate the business initiative into User Stories that describe customer and enterprise value, the need for resources and partners, and cost structures in a group of statements. For example, "As a college student, I would like be able to easily share experiences with my friends and family so I can stay connected to their lives while I am away at school," and "As an entrepreneur, I would like to deliver on-line experience sharing capability to college students with targeted advertising so I can serve their need and realize sufficient value to cover development, hosting partner costs, and earn a profit," could have been the first User Stories for Facebook.

Agile Approach

Image if you would that you are building your "dream home" and you are having friends and family over for a holiday dinner. You prepare the dinner, serve it, and entertain your guests in your new kitchen, dining room, and family room, while the bedrooms, den and other parts of the house are still being designed. Image also that after the dinner, you decide to make changes to the kitchen and your cost and schedule did not change. All you had to do was delay the game room in the basement to a later project. That is the power of Agile.

Every project has a beginning and an end. Projects require a Charter, a deliverables plan, a schedule, a budget, and resources. Projects must manage risks, lead stakeholders, and provide the end product described by the Charter.   Agile projects are no different.   The team and process used to deliver the product, however, are very different.

Agile is iterative and incremental software development, where requirements and solutions evolve through collaboration between cross-functional, self-organizing teams. It promotes adaptive planning and encourages rapid and flexible response to change. [vi] It is a lightweight framework that allows the team to utilize their collective skills to produce. Scrum is an implementation of the Agile Software Development methodology and the Scrum team has three roles, Developer, Product Owner and Scrum Master.   The Developers design and build the software product. The Product Owner provides business-functional guidance, and the Scrum Master facilitates the process. [vii]

Rather than project phases, Agile has iterations of activity focused on the deliverable goal of an increment of the working software product. The activities are grouped into Sprints and each Sprint is completed in a time box of typically two weeks. The Sprint consists of Planning, Development, Review, and Retrospection.

The chartered deliverable plan, "Product Backlog," is described in a set of high level User Stories and the project schedule becomes an ordering of the backlog described in the "Release Plan." A Program Release Plan is single Product Backlog for multiple Agile projects that are linked to a common initiative. The program or project is chartered in a time box and the budget is the resource cost plus external partners and tools needed. Estimating becomes the following questions, "How much can we accomplish within the time box and who or what will we need to accomplish it?" At the end of the chartered development time, the project has a Review and Retrospective. Essentially, the project is guided similarly as is the Sprint. By creating a project time box of 90 days, less than $350,000, small projects are delivered without becoming large projects and large projects are tailored every 90 days to realign with business innovation needs.

The Scrum team resembles a matrix project team model, in that each team member reports to their functional manager, and they are responsible to their team for deliverables.[viii] The team's skillsets will vary. Developers will have software engineering, architecture, QA, DBA, server management, business analysis, and business line skills. Product Owners will have enterprise expertise and a stakeholder network, business knowledge, communication, and leadership skills. The Scrum Master will have Scrum framework expertise, enterprise knowledge, communication, and leadership skills.   Project Management skills will exist in both the Product Owner and Scrum Master, although the Scrum Master tends to perform more of the Project Management activities.

Each sprint begins with planning.   The team chooses the items from the Product Backlog to complete this sprint and defines the meaning of "done."   In simple projects, "done" could mean that an increment of the final product is released into a production environment for the end consumers to use. In more complex projects, "done" might be defined as an increment released to a controlled test environment for the consumers to try. The planning segment of the sprint quickly gives way to development. During development, increment components are produced and assembled. Backlog for future sprints is more clearly defined.   Issues from previous sprints are resolved. And, anything else that the team determines is needed to meet the sprint's goal as defined during planning.   This is a lot to accomplish in two weeks, so focus is paramount. Each day there is a "Stand-up" meeting, time boxed at 15 minutes, to discuss plans and progress and to seek support for impediment removal.

The Sprint ends with a review of the product increment delivered including a decision on deployment. If the decision is to deploy, then backlog items are added for deployment in the next Sprint. The final piece of the Sprint is the retrospective discussion. The Retrospective is an extremely powerful tool in process improvement and for collaborative team building. Changes are no longer driven from the top down. Everyone has a voice and the voice is heard, acknowledged, and acted upon. At the end of the Sprint, three artifacts are produced, Product Backlog, Burn-down Chart, and notes from the Planning, Review, and Retrospective events.

As an outcome of the review and next planning events, the Product Backlog may change.   New items could be added and existing items either removed or ordered below new items. In Waterfall projects, changes of this type would require steering committee approval, budget and schedule reviews, and amendments to the Charter. In Agile projects, the Charter only describes customer and enterprise value and as long as the altered backlog will deliver the value, the Charter is not changed. In Agile Programs, change reviews are needed to keep dependencies aligned with other teams.

At the end of the Charter time box, the project is ended. The product is reviewed against the value goals. If all the goals are achieved, then the innovation succeeded and the team moves on to the next project. If the innovation goals are only partially satisfied, then a new Charter is developed with a new set of goals and the process continues.   An important concept to note is that a value goal in a Charter can be a component of a larger deliverable, a stage in a larger process improvement, an experiment, or any other goal. The definition of "done" for the goal is defined as "sufficient to deliver the value."   The absence of defects does not determine "done" and the presence of acceptable defects does not preclude "done."

Agile Project Case
The CMO of a loan payment processing company wanted to expand his sales channels into the point of purchase of loans thereby increasing his sales without increasing his internal costs. He contracted with a software development consulting firm to build the point of sale solution.   The consumer focused User Story was, "As a below average credit purchaser of a long term loan, I want to reduce my interest paid so my total cost is more in line with someone with stronger credit."   This User Story was further defined and resulted in an initial product backlog with the following deliverables:
- A rules based point of sale web-site which can be used by lenders and sellers of high cost products
- A set of on-line sales tools to enhance the sales channel's close rate
- A cloud hosted application which can integrate with the CMO's payment processing engine
- A commission application to compensate the new sales channel
- On-line demonstration and training tools to help the sales channel learn to use the product and improve sales

These deliverables were too large to be completed in the first 90 days, so a subset was selected. The subset included:
- A rules based point of sale web-site for high cost products
- A cloud hosted application with semi-automated integration to the payment processing engine
- Commission would be computed with automation, but paid manually

The project was chartered with an end date, target product value, and a target not to exceed cost. The development firm agreed to the parameters providing that only 75% of the target product was required for the target cost. The backlog was groomed further and the CMO and his team determined the order. They wanted to be certain that the 75% they received had the highest value. The commission computations were last, since they could be computed by his staff in the short term.

In the first 2 week sprint, the cloud vendor was chosen, a database was created, the test environment developed, and a lender could access a web page and enter a customer's contact information. By the end of 60 days, the application was able to enroll customers with all the necessary rule based constraints and it was able to integrate with the backend system as prescribed. The CMO's team was generating interest in his product and they found the need to demonstrate the software. He met with the development firm and they adjusted the product backlog to include a scaled down version of the demonstration tools to be completed before the commission module. Since the development team had already completed the 75% required by the agreement, they all felt this change was low risk. By the end of the 90 days, the application was deployed into a cloud based production environment with the rules base point of sale, semi-automated integration, simple demonstration capability, and commission computations. And, three lenders began using the application to sell the CMO'S service.

The first Charter review went very well and a new Charter was developed to add more features to the current production system and to add the training and demonstration tools not deployed in the first release. The Retrospective described what worked well and where improvements were needed.

As improvements, there needed to be more of the CMO's team allocated to the project than anticipated because the project required a significant amount of collaboration between the CMO's team and the developers. The CMO's team initially preferred to work remote from the developer, choosing telephone conversations, email, and remote product demos as their primary mode of communication. They later admitted that being remote led to misunderstanding and possibly caused some of the issues discovered in the product. The developer needed to include a DBA and a system administrator to the team, one day a week, to improve deployments and post deployment support.

As successes, the CMO was ecstatic with the results. Not only did he have a new system in production in 90 days, but he was already seeing positive cash flows from the product. He also appreciated the Retrospectives because his team had a voice into the development process and his team received feedback that helped them improve performance in their roles. And, the CMO very much appreciated the amount of control he had on the outcome with the ability to change course during the project without affecting cost or schedule.

On the next project the CMO's team collocated with the developers for one day each week and they attended the daily stand-ups without fail. The increased collaboration between business SME's and developers produced a higher quality product and reduced rework. The CMO's team reported that the quality using Agile was higher than previous projects where more documentation was needed. They and the CMO had a high degree of confidence in the developer because of the collaboration, the daily stand-ups, the sprint reviews, and the progress they participated in delivering. And, they had fun!

The Bottom Line
Business innovation must be accomplished within the business framework. Innovation must increase revenue disproportionately to increases in cost. Innovation must reach a target customer segment that will have a positive impact on the long term success of the business, and it must be deployed quickly

with the ability to adapt to competition and changing markets.   To achieve sustained agility in business transformation, the executive team will need to support the changes. They will need assurances that the innovations are within the strategic scope, only excellent innovation projects are completed, and the projects are within budgetary and schedule plans so they don't starve the rest of the enterprise. Currently, these assurances are provided by Program and Project Management techniques.

The Project Management Institute's Program and Project Management best practices for risk management, resource allocation, budget and schedule management, and stakeholder leadership are effective tools. Agile addresses these needs within the project, but it does not specifically address these needs for the enterprise. As an adaptation to Agile, the Product Owner or Scrum Master will implement the Program Management best practices to ensure the enterprise level reporting and the organizational leadership that executive teams require, to support new initiatives.

The project's artifacts and the personal knowledge of the project team form a significant amount of intellectual capital. The artifacts created during the project, including the program code and event notes, must be preserved in a way that is searchable, shareable, and safe.   Effective Program Management practices will ensure the appropriate handling of historical artifacts.

Agile with Scrum is the best framework this author has experienced in some 30 years of Program, Project, Engineering, and Business / Product Development experience.   It requires skilled team members with a collaborative spirit and predominately located where they can all breathe the same air. It requires focus and dedication to success of the goal.   As Harry Truman said, "It is amazing what you can accomplish if you don't care who gets the credit." Success with Agile requires *everyone* to check their ego at the door.

As the CMO's team said, "it is fun" and fun is contagious. Once you start to deliver a few successes, Agile's use will grow naturally and it will deliver new products and innovative transformations that will propel your company to become a market leader, earn more market share, increase shareholder value, and grow.

Reference

[i] Miles, Robert, H. "Accelerating Corporate Transformations," *Harvard Business Review*, January-February 2010.
[ii] Blank, Steve. "Why Lean Start-Up Changes Everything," *Harvard Business Review*, May 2013.
[iii] Lars Mieritz. "Gartner Survey Shows Why Projects Fail," June 1, 2012, <Gartner.com>.
[iv] Osterwilder, Pigneuer. "The Business Model Canvas," May 10, 2012, <www.businessmodelgeneration.com>.
[v] Osterwilder, "The Business Model Canvas," February 6, 2012, <youtube.com>.
[vi] Beck, Kent; et al. "Manifesto for Agile Software Development". Agile Alliance. Retrieved 14 June 2010.
[vii] Schwaber, Ken. *Agile Project Management with Scrum*, Redmond: Microsoft Press, 2004.
[viii] *A Guide To The Project Management Body Of Knowledge – Fourth Edition*, Newton Square: Project Management Institute, Inc. 2008: 28-32.

Christensen and Raynor. *The Innovator's Solution*, Boston: Harvard Business School Publishing Corporation. 2003.
Collins, Jim. *From Good To Great*, New York: HarperCollins Publishing, Inc. 2001.
Drucker, Peter F. *The Effective Executive*, New York: HarperCollins Publishing, Inc. 2002.
Kim and Mauborgne. *Blue Ocean Strategy*, Boston: Harvard Business School Publishing Corporation. 2005.