

A Celerity eBook



The Agile Tipping Point: 2018

Changing the World of Work with Iterative Development

celerity

Introduction:

Agile in 2018 & Beyond

The world of work continues to iterate its path forward with Agile methodologies playing a vital role. While Agile is often viewed as the new norm, most organizations have yet to realize its promised potential.

Those that have tipped from adoption to scaling, are now focusing on value creation and supporting Agile teams through enterprise transformation.

In 2018, the word “Agile” has taken on a life of its own, used to describe everything from iterative development to a general sense of flexibility. Its buzzword status notwithstanding, Agile has grown up, and its practitioners are growing in their ability to leverage it effectively. Now seventeen years since *The Agile Manifesto* was first published, many industry

surveys place the Agile adoption rate among development teams at over 95%. Whether your organization has long engaged in the Agile game or is a new entrant, understanding both how we got here and how to rev your Agile engine will ensure your success moving forward.

This book examines in two sections:



Part 1: Agile as the New Normal

What forces have brought Agile to its present zenith of mass adoption?



Part 2: Revving the Agile Engine

What does it take to go beyond scaling Agile and truly transform?



Agile in a Nutshell

The Agile software development methodology began in the mid-1990s but has gained traction due to its suitability for digital projects—a priority for today’s enterprises.

The Agile process includes refining requirements, developing through iterative cycles (“sprints”), and testing with end-users throughout the process—often resulting in a better product.

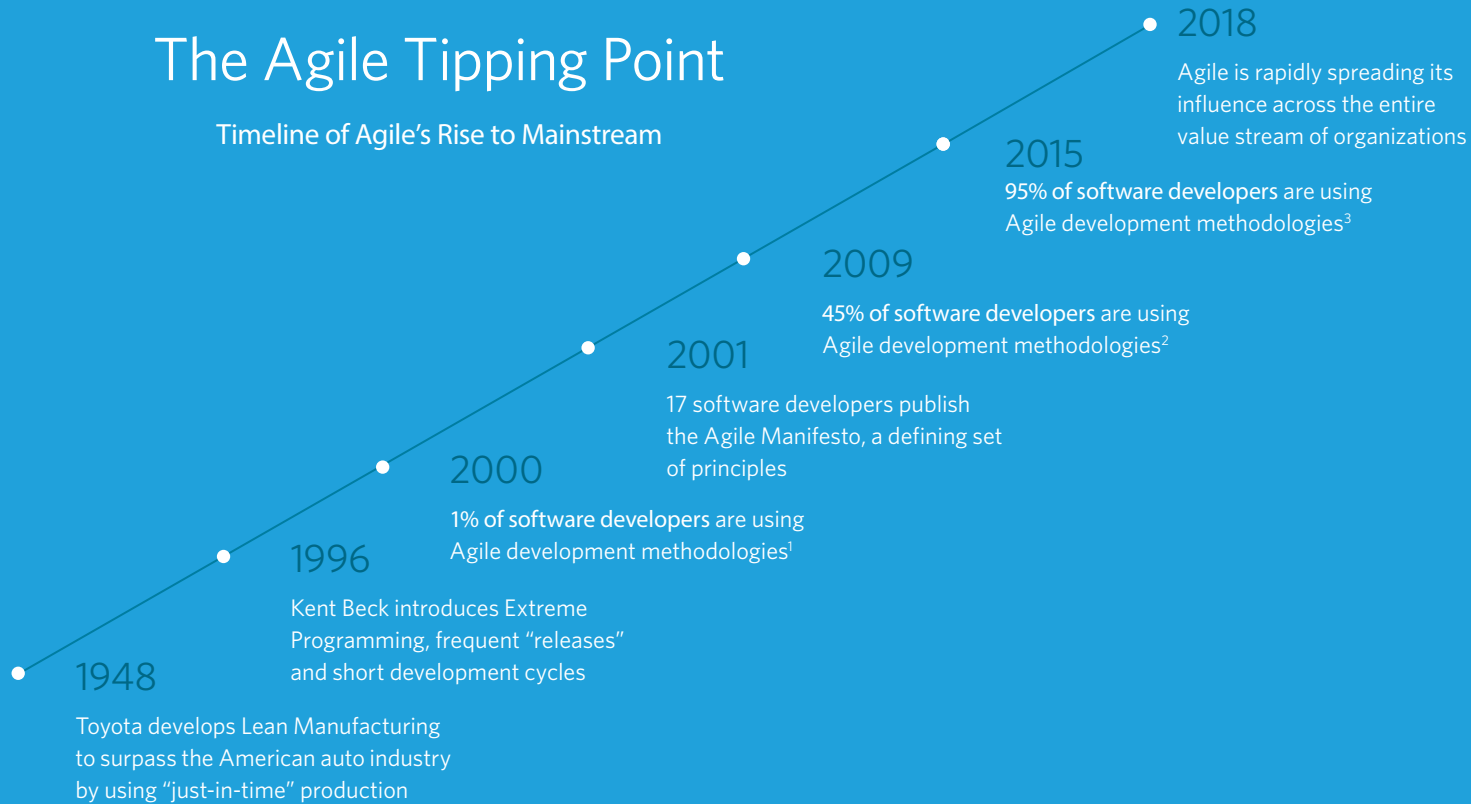
“Change is not for its own sake. It’s the means to the end, not the end.”

Pearl Zhu, Digital Agility

The higher success rate of Agile projects has resulted in mass adoption among development teams, and non-development functions are jumping on the Agile bandwagon as well. These teams—like Marketing, HR, and Accounting—independently benefit from iterative work cycles, but they are also able to better support Agile's ability to create value for an organization.

The Agile Tipping Point

Timeline of Agile's Rise to Mainstream



67% of Agile development teams are deploying at least once a week.

Sauce Labs

Agile Beyond the Tipping Point

Since the Agile Manifesto introduced the term back in 2001, the Agile movement—with all of its values, principles, methods, practices, and tools—has significantly changed the landscape of how we work. Today, leading companies from Amazon to GE use Agile methods to manage various aspects of their businesses.

Three key market forces are credited for Agile's critical mass adoption:



Force 1 Digital Acceleration

80% reported that Agile helped them achieve faster speed to market than traditional methods.³



Force 2 User is King

End users play a big role in Agile and are demanding greater control over features and the overall experience.



Force 3 Empowered Teams

Control-and-command style of management is dead, and corporate culture is increasingly flat-structured.



Force 1

Digital Acceleration

Companies are racing to differentiate themselves through innovative products and services. And they're looking to Agile for a turbo boost.

The widespread adoption of Agile comes partly from the rapid rate of digital innovation. You know, the same reason why your phone is out of date the minute it's activated. Or why walkmen were replaced by CD players, which were replaced by iPods and then the iPhone... only to be replaced by a newer iPhone.

Unpredictability rules this digital landscape, thanks to swiftly emerging standards, volatile platforms, intermittent connections, and constantly improved devices. To remain

competitive, businesses strive to shorten product development lifecycles and remain more flexible amid each new wave of change.

That's Agile's forte. Agile has gained credibility as digital's wingman due to its ability to constantly incorporate new and changing requirements. Since the cost of fixing something increases exponentially over time, Agile's shortened feedback loops drive down risk and help teams to find problems earlier rather than later.

The Agile Process:



Top Benefits of Agile

Ability to manage changing priorities	<div><div></div></div>	87%
Productivity	<div><div></div></div>	85%
Project Visibility	<div><div></div></div>	84%
Team Morale	<div><div></div></div>	81%
Delivery Predictability	<div><div></div></div>	81%
Faster Time to Market	<div><div></div></div>	80%
Software Quality	<div><div></div></div>	79%
Less Project Risks	<div><div></div></div>	78%
Business/IT Alignment	<div><div></div></div>	77%
Software Maintainability	<div><div></div></div>	70%
Better Manage Distributed Teams	<div><div></div></div>	62%



Force 2

User is King & Agile is Queen

Good user experience design provides a competitive advantage for businesses. Why not insert the user into the development process?

That's exactly what Agile does. Through shortened feedback loops (typically two-week sprints), Agile weaves together insight from employees, customers, and other stakeholders to ensure they have a say in which features they do and don't like.

This is a great improvement over more traditional approaches like Waterfall, where users often don't see products until they are completely finished. By that time, technology has evolved, and the key requirements have changed.

"Continuous improvement makes each iteration better than the last and delights customers," said Steve Denning in *The Leader's Guide to Radical Management*. Consider the way Apple's products have evolved over time, from the Macintosh to the iPhone 7 Plus. "Getting to the customer really early in the process and constantly validating are best practices that smart companies have been doing from day one," says John Casey, entrepreneur in residence at George Mason University.

"The essence of the Agile movement, whether in new product development, new service offerings, software applications, or project management, rests on two foundational goals: delivering valuable products to customers and creating working environments in which people look forward to coming to work each day."

Jim Highsmith, *Agile Project Management: Creating Innovative Products*

Agile & Millennials: A Perfect Pairing

Millennials represent 30% of the population and are the largest, most demanding user group.

Millennial Values

1. Instant gratification
2. Transparency
3. Participation
4. Disruptive Brands

How Agile Responds

1. Faster, shorter delivery cycles
2. Continuous feedback loops
3. Full team collaboration
4. Constant innovation and adoption



Check out the Superdesk at Barbarian headquarters in New York. Everyone works at the giant desk together—including the CEO. The Superdesk even has its own Twitter feed.

© Photo by Michael Moran and The Barbarian Group

Organizational Alignment: A Key to Digital Excellence⁴

Increased alignment between IT & Business directly correlates with an increase in digital performance.



Force 3 Empowered Teams

For most of the 20th century, authoritative management ruled. But today, corporate structure is flatter, more collaborative, and more team-based than ever, adding more fuel to Agile's fire.

Organizational silos are morphing into cross-functional teams that must have better communication methods to meet delivery deadlines.

Since Agile is all about empowering people, it thrives in a de-layered, communicative workplace. Agile teams report high levels of employee satisfaction, since they are all heavily engaged in the decision-making process and are collectively working towards a common goal. This reinforces Agile's key tenet: "One Fails, We All Fail".

Now that IT is recognized as a revenue-driver for the Business, both sides can move past the traditional issues that stemmed from IT delivering products the Business didn't want. Agile development is a great way for the two sides to improve trust through transparency.

⁴ Apigee, "Digital Transformation"

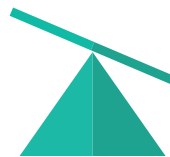
Part 2: Revving the Agile Engine

There are several questions to consider when deciding the future of Agile for your teams. These questions deal largely with the work environment and processes in place to transform your organization through successful, scalable Agile development.

To help you best position your teams for Agile success, we explore:



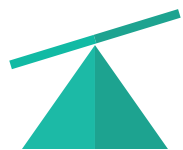
Why Agile Fails
Avoid these common pitfalls



How to Scale Agile Successfully
5 Best Practices



Major Agile Success Stories
5 Use Cases



Why Agile Fails

Doubt, Disillusionment, and Dabbling



Agile isn't the right answer for everyone. And not everyone does Agile the right way. Here are the top reasons why an Agile development project might crash and burn:

Doubt

Agile projects are typically grassroots initiatives spearheaded by IT, but lack of executive buy-in can be fatal. Many companies haven't fully resolved the historic distrust between IT and Business units.

Even if an executive team buys into the idea of Agile, they often doubt that the change to Agile will drive business value, associating the flexibility with "loose" processes that may increase risk.

38%

of developers feel that they lack management support.⁵

Disillusionment

For all its benefits, Agile is still hard to implement. Many forays into Agile end in disaster because unrealistic expectations cause stakeholders to give up on the method easily. Additionally, 25% of organizations report that they don't have clearly identified metrics to measure the success of Agile projects, leaving ROI uncertain and changing perceptions of positive headway.

55%

of companies say company culture made the switch to Agile difficult⁵

Dabbling

Experienced Agile talent is hard to find—and, teams often try to guess their way through the highly nuanced development process. Don't fall into the trap of assuming the reduced processes of Agile mean there are no actual rules. Agile is a prescribed methodology and training/practice is a requirement. You simply can't go about Agile halfcocked. Continuous integration, proper code and library repositories, and automated deployments/testing are all must-haves.

39%

of developers say there isn't enough personnel with Agile experience⁶



Five Ways to Scale Agile Successfully

Clearly Agile is more than just a buzzword. Deployed correctly, it has the potential to transform and accelerate entire organizations. Here are five ways to rev your Agile engine and ensure your success.

Here are five best practices for scaling Agile development to get you started:

1. Make sure everyone's on board.

On both the Business and IT side, organize a "critical mass" of advocates and early adopters, and institute an Agile mandate for IT projects to ensure adoption. Conduct trainings and workshops to socialize the method.

2. Keep each team as small as possible.

Often, five really good developers can do the work of 20. Determine the team's size by the nature of the problem being addressed, not management expectations.

3. Emphasize portfolio management.

Organize with subteams, and then have the leaders from each subteam coordinate daily during a "scrum of scrums." Use an online tool to manage deadlines and responsibilities.

4. Do modeling up front.

Establish a common business vision (based on high-level requirements) and a common technical vision (based on common architecture). Use an Informational Radiator to make results visible in a common space. This helps teams focus on short-term goals while keeping sight of the vision.

5. Obsess over delivering to the business through IT.

A sense of urgency and a healthy partnership between the business, and IT is the glue that keeps this method together. Success should be constantly evaluated through sprint retrospectives so that both sides can adapt and maximize value.

"Don't expect Agile to happen overnight; 40% of companies using Agile for 100% of their projects took over four years to get there."

Forrester

✓ Scaled Agile Success Stories

Scaled Agile Frameworks consist of three levels: team, program, and portfolio, all of which connect lean business strategies to agile development execution. Many companies have implemented this model, or some version of it, to achieve groundbreaking results and innovation:

Spotify

Spotify scaled Agile methods to over 30 teams across three cities since inception, and now organizes hundreds of developers through Agile “squads” and “tribes.” This matrix, which is weighted towards delivery, groups people “into collocated squads, where people with different skill sets collaborate and self-organize to deliver a great product,” says CEO Henrik Kniberg.

Netflix

Netflix excels at putting the user experience first, as evidenced by its ability to stream 4 billion hours of content nearly ubiquitously across nearly 250 different devices. Instead of becoming bureaucratic and hierarchical as it has grown larger, Netflix says that it aspires to grow fast through flexibility and freedom, without becoming complex or chaotic.

John Deere

John Deere, a 179-year-old midwestern manufacturer, was able to rebrand itself from being an old-school tractor company to an innovative technology disrupter because of Agile methodologies. Through Agile coaching and the task to “discover potentially disruptive technology,” the team cut down their production time in half.

GE Healthcare

GE’s CIO reported “This year, our target is 60% of our program spend to be done using Agile.” To help speed adoption, GE hosted internal Agile conferences that brought together business representatives and Agile industry leaders.

One of the world’s largest manufacturers embarked on a large-scale Agile transformation that grew a group 100 developers practicing Agile to over 1,200 and counting. The company reports that time to market is 20% faster, time to production is down 20%, and employee engagement is up 10%.



Agile Outside of IT

“73% of respondents at companies using agile processes and 77% of those at organizations deploying DevOps said these efforts improve employee recruitment and retention.”

Baseline Magazine

Conclusion:

Accelerate Your Agile Results with the Right Partner

At Celerity, we believe the most flexible companies win. Building in a system of adaptable, rapid delivery enables companies to engage users early and learn quickly from failures. These are the keys to breakthrough innovation.

While Agile has established itself, there are certain factors in many corporate environments that make it difficult to implement Agile on an enterprise-wide scale. If you're ready to get started in Agile, or to scale your existing Agile processes to an enterprise level, Celerity can help you plan for success.

Our own Agile Development Teams work alongside yours to execute projects rapidly and properly, while teaching them the ins and outs of Agile best practices.

The Rapid Rate of Innovation

"I want [a Scrum team] going so fast they are just on the verge of spinning out of control, yet are able to keep it together and deliver something classic and powerful."

Mike Cohn, Agile expert

Agile Dictionary

Agile

An umbrella term for iterative, incremental software development methodologies. Includes Extreme Programming, Scrum, Crystal, Dynamic Systems Development Method, Lean, and Feature-Driven Development. Agile methodologies emphasize small teams delivering small increments of working software with great frequency while working in close collaboration with the customer and adapting to changing requirements.

Burn Down Chart

A key artifact that visualizes a team's progress through units of work that still need to be accomplished (Y axis) vs. Units of time (X axis).

Daily Stand Up or Scrum

A daily, 15-minute meeting with the entire team. The primary purposes are to unearth anything impeding a team member and to promote communication. Stakeholders may attend as silent observers. Three questions everyone should answer are:

- What have you done since yesterday?
- What are you planning to do today?
- Do you have any stumbling blocks?

Epic

A large user story that is broken down into smaller stories prior to implementation. Usually lower priority items.

Extreme Programming (XP)

An Agile method that hinges on customer participation, transparency, testing, and frequent delivery of working software.

Information (or Visual) Radiator

A handwritten, drawn, printed, or electronic display which a team places in a highly visible location, so that all team members can see the latest project status information at a glance.

Iterative Development

A common component of all Agile methods, emphasizing incremental delivery that reduces the risk of project failure by dividing projects into smaller, more manageable pieces of work. This method requires teams to “inspect and adapt” between iterations, as opposed to the Waterfall method of “big design up front” followed by development and testing.

Minimal Marketable Product (MMP)

A fully functioning piece of software. It contains the bare minimum feature set that a user needs. It gives a business the ability to get to the market faster and be empirical about adding future features.

Minimum Viable Product (MVP)

A very early version of software released to see how the market will respond and determine if the concept is worth pursuing.

Product Backlog

An evolving list of product requirements—often expressed in terms of user stories and prioritized by the customer—that tells an Agile team which features to implement first.

Product Owner

Helps the team find its clear, elevating goal by defining the features of the product (with the customers), developing and prioritizing the product backlog, making scope/schedule tradeoff decisions, and adjusting priorities as requirements emerge. Responsible for the “what.”

Agile Dictionary

Scrum Team

A cross-functional team of developers, testers, user experience experts, and business analysts. Typically should be five to nine people.

Scrum Master

A servant-leader that flattens the hierarchy and coaches colleagues to help them improve. Protects the team from distractions and helps them become fully functional and productive. Responsible for the “how.”

Sprint

An iteration or uninterrupted period of time during which an Agile development team designs, codes, and tests the product, usually two to four weeks long. At the sprint's end, the team delivers a “potentially shippable” product increment.

Sprint Planning:

Happens once per sprint at either the beginning or end. The team commits to a set of user stories, prioritized by the Product Owner, that they believe they can accomplish in the sprint. The user stories are broken down into tasks, which are estimated in hours. The result? A sprint backlog.

Sprint Review

Held at the end of each sprint to promote transparency and give stakeholders a chance to provide feedback. The team gives demos of deliverables, which are accepted or rejected by the Product Owner. This helps prevent the “Big Bang” effect at the end of a project where stakeholders are surprised by a deliverable (positively or negatively).

Sprint Retrospective

This happens at the end of a sprint, and allows the team to look inward at what is working and what is not. Teams should focus on areas that will improve cohesiveness and efficiency. Three questions that should be answered are:

- What should we start doing?
- What should we stop doing?
- What should we continue doing?

Story Point Relative Estimation:

A planning method (often called “planning poker”) that helps teams reach consensus around cost and level of effort. The exercise uses a deck of cards to assign difficulty to each high-level feature.

The Agile Manifesto

Published in 2001 by a group of independent thinkers about software development, the manifesto stresses:

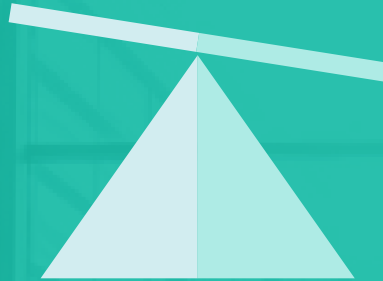
- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Velocity

The average number of points an Agile team accomplishes during an iteration.

Waterfall

Traditional, phase-driven development method, which emphasizes top-down project management, “big design up front,” silos for architecture and design, coding, and testing, and extensive documentation.



The Agile Tipping Point: 2018

Iterative Development and the Changing World of Work

