



© 2018 by the author(s). This is an Open Access article distributed under the terms of the Creative Commons Attribution 4.0 International (CC BY 4.0) License (<https://creativecommons.org/licenses/by/4.0/>), allowing third parties to copy and redistribute the material in any medium or format and to remix, transform, and build upon the material for any purpose, even commercially, provided the original work is properly cited and states its license.

Citation: Forte, F. M. and Kloppenborg, T. J. 2017. The Agile Mindset for Project Management. *International Research Network on Organizing by Projects (IRNOP) 2017*, UTS ePRESS, Sydney: NSW, pp. 1-15. <https://doi.org/10.5130/pmrp.irnop2017.5740>

Published by UTS ePRESS | <http://pmrp.epress.lib.uts.edu.au>

CONFERENCE PAPER

The Agile Mindset for Project Management

Frank M. Forte, CSM, PMP, SPC^{1*}, and Timothy J. Kloppenborg, PhD, PMP²

¹Forte Leadership Technology, LLC, United States. f4forte@gmail.com

²Xavier University, United States. kloppenborgt@xavier.edu

***Corresponding author:** Frank Forte, Forte Leadership Technology LLC. f4forte@gmail.com

Name: International Research Network on Organizing by Projects (IRNOP) 2017

Location: Boston University, United States

Dates: 11-14 June 2017

Host Organisation: Metropolitan College at Boston University

DOI: <https://doi.org/10.5130/pmrp.irnop2017.5740>

Published: 07/06/2018

Synopsis

We describe Agile in three ideas: satisfy the customer, engage the team, and simplify everything. All ideas from a recently updated *Agile Manifesto* fit in our three-idea framework. We also considered all ideas from the *PMI Agile Certification Outline* and had Agile experts prioritize them. The resulting important ideas all fit in the three-idea framework. Collectively, these explain why Agile is useful and how to perform it. We conclude by explaining values, benefits and actions of Agile.

Research Design

There were multiple parts to this research design. First was extensive practice using Agile while studying for and passing several Agile certification exams. As an outgrowth of this practice and study, in 2016 an update to the *Agile Manifesto* was written with an eye towards current practice in both IT projects and a variety of other contexts to update and broaden the relevance. The updated manifesto contains four values and 13 processes, just as the original manifesto did. Then we brainstormed and created the simple three-part model dealing with customer, team and simplification.

To further validate, we used the *PMI Agile Certification Outline* as a standard that had input from many people and is widely accepted. We de-duplicated a few items that occurred

DECLARATION OF CONFLICTING INTEREST The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article. **FUNDING** The author(s) received no financial support for the research, authorship, and/or publication of this article.

repeatedly and used the remaining 37 items as survey items. We asked a number of experts who each have at least 20 years of project management experience, including both traditional and Agile to prioritize the de-duplicated items. We ended up with 23 items that were considered to be important by at least half of the experts.

Relevance to Practice and Education

These findings are relevant to practice, as Agile is often described in various ways and may be poorly understood. Many practitioners who attempt Agile focus on one or a few aspects, rather than the totality, and are likely to be disappointed in their results. Agile may be at a point, like TQM was a generation ago, with great popularity, but poor understanding. It could suffer from the same fate, with better organizations receiving substantial benefits, but many other organizations finding poor results unless it is easily understood. This is an attempt to help Agile to be better understood.

The findings are relevant for teaching as this is a simple organizing framework that has been shown to incorporate all of the more important Agile ideas. This can be taught as a primer on Agile or easily incorporated into a broader course on project management.

Main Findings

All four values and 13 principles from the updated *Agile Manifesto* fit neatly into one of the three categories of customer, team and simplification.

All 23 prioritized items from the *PMI Agile Certification Outline* map directly into one of the three categories of customer, team and simplification.

The items from the updated manifesto largely explain why Agile is useful, while the prioritized items from the PMI outline largely explain how to perform Agile.

Research Implications

This was a small-scale pilot study. A larger study might bring out more nuances of how to perform Agile and why certain items are more valuable than others. Although this concisely summarizes the most important aspects of Agile into a simple framework, further study may indicate that certain items are more important to perform quickly when adapting to Agile or might be better suited to projects in a particular industry or other context.

Keywords

Agile, Project, Project Management, Customer Satisfaction, Team, Team Engagement, Simplification, Leadership, Value, Principle

The Agile Mindset for Project Management

Agile is a form of adaptive or change-driven project management in which PM is largely reacting to what has happened in the early stages of a project rather than planning everything in detail from the start. Documentation is minimal early in the project but becomes progressively more complete. To understand Agile, one needs to know both the methods and the mindset of Agile practice. For the methods, a project vision is developed and shared early. Project teams plan in short bursts (generally of one to four weeks) often called sprints or

iterations. The details are planned for the upcoming iteration, and very little change is allowed during it. Products are defined and delivered one iteration at a time, with an output that has business value successfully finished in each iteration. Then the next iteration is planned. The mindset is empowering, engaging and enables open communication, as detailed below.

The word *Agile* has become synonymous with change in technology organizations large and small; however, there is an emerging misconception of how Agile can be used to make a significant change in complex endeavours. Most of the books on this subject focus on identifying the mechanics of a particular practice within the Agile space, but fail to define the overarching thinking that makes them so effective. These include, but are not limited to, the use of Scrum, XP or Kanban. Although these are necessary works and lay the foundation for Agile practices, they do not address the mindset shift needed to take full advantage of these practices. The misconception is further extended by the use of traditional project management words to explain similar concepts within Agile without drawing clear distinctions. It is left to leaders to figure out what the difference is and how they need to think differently in order to get the outsized gains attributed to Agile as reported in numerous case studies. Without the firm and deliberate transition to a new way of thinking about teams and how teams work differently with Agile practices, Agile will quickly devolve into a semantic equivalency that will leave executives more frustrated by the failure of another approach that degrades their ability to unlock the performance potential of their organization and their teams, and capitalize on the individual knowledge of team members.

Agile has grown tremendously in popularity in the last few years. Although it was originally conceived and described as being useful for software development projects, many people involved in other types of development projects have begun to experiment with and often embrace it.

Our concern is that Agile might follow the path of total quality management (TQM) from the 1990s. TQM also emerged from a frustration in current management techniques. The difference is that TQM was oriented towards managing an ongoing organization, whereas Agile is aimed at managing a software project. The old way of managing for quality was to develop or produce a product and then test it. It was deemed to be either acceptable or not. Adherents of TQM correctly saw this as a wasteful and sloppy way of doing things. Therefore, they proposed a dramatically different approach that emphasized stakeholder satisfaction, empowered performance, process improvement and management by facts.² This new approach helped many organizations vastly improve their quality and competitiveness. Unfortunately, there was no broad consensus of exactly what constituted TQM. Some advocates emphasized management by facts driven largely by statistical analysis; others emphasized the importance of front-line workers making decisions. The result was that although many of the better-managed organizations reaped substantial benefits from incorporating TQM ideas and techniques into their management systems, many other organizations wasted time and effort before giving up. Are we at such a crossroads on Agile? Like TQM, Agile has multiple aspects, and one cannot cherry-pick the few they like. For it to work best, leaders need to understand the whole of it.

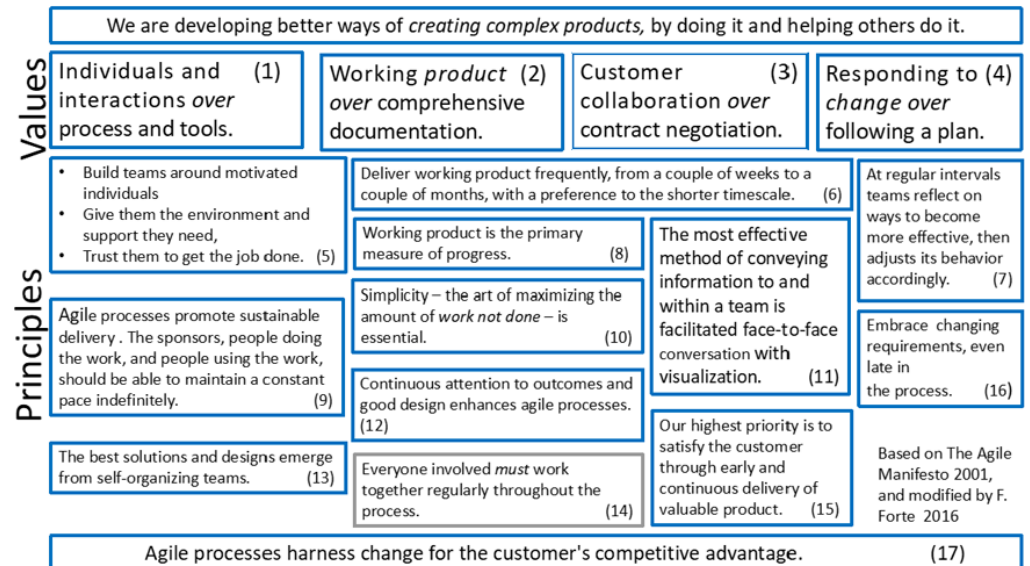
Therefore, we decided to describe Agile as three simple ideas. Each of these ideas can be expanded upon, but having an easy way to describe and remember the various aspects of Agile may help to promote common understanding and acceptance.

The three simple ideas are as follows:

1. Satisfy the customer.
2. Engage the team.
3. Simplify everything.

Then we went in two directions to see how they converge. First, we rewrote the Agile Manifesto, updating it according to effective practice and describing it in such a manner to be relevant to many types of projects rather than just software development. Our updated Manifesto is shown in Exhibit 1.

Exhibit 1 Agile manifesto updated



With that done, we map the updated Manifesto on the three main topic categories to see how they fit. This forms the values and principles behind Agile, organized in a simple manner. Our updated Manifesto mapped to the customer, team and simplicity is shown in Exhibit 2.

Exhibit 2 Updated Agile Manifesto Mapped to Three Ideas

1. Satisfy the customer:

- Value customer collaboration *more than* contract negotiation.
- Our highest priority is to satisfy the customer through early and continuous delivery of valuable *product*.
- Embrace *changing* requirements, even late in the process.
- Agile processes harness *change* for the customer's competitive advantage.

2. Engage the team:

- Value Individuals and interactions *more than* process and tools.
- Build *teams* around motivated individuals.
- Give them the environment and support they need.
- Trust them to get the job done.
- Agile processes promote sustainable *delivery*. The sponsors, people doing the work and people using the work, should be able to maintain a constant pace indefinitely.
- The most effective method of conveying information to and within a team is *facilitated face-to-face conversation with visualization*.
- The best *solutions* and designs emerge from self-organizing teams.
- Everyone* involved *must* work together regularly throughout the process.

3. Simplify everything:

- Value working *product more than* comprehensive documentation.
- Value responding to *change* more than following a plan.
- Deliver working *product* frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- At regular intervals, *teams* reflect on ways to become more effective and then adjust their behaviour accordingly.
- Working *product* is the primary measure of progress.
- Simplicity – the art of maximizing the amount of *work not done* – is essential.
- Continuous attention to outcomes and good design enhances Agile processes.

Next we looked at the *PMI Agile Certification Outline* and asked leading practitioners and writers to prioritize among the many ideas. Which are truly important? Then we group those ideas into the same three topic areas. These are more action-oriented and form the methods that can be used to implement Agile, again explained in a simple manner. We started with all of the tasks identified in the *PMI Agile Certification Outline*. We used our judgment to de-duplicate similar terms, reducing the number of tasks to 37, as shown in Exhibit 3. These 37 items are listed in the order in which they appeared in the *PMI Agile Certification Outline*.

Exhibit 3 Agile tasks paraphrased and de-duplicated from PMI-ACP Exam content outline

1. Maintain highly visible information registers.
2. Allow people to experiment.
3. Encourage collaboration.
4. Encourage emergent leadership.
5. Practise servant leadership.
6. Define and deliver products incrementally.
7. Gain consensus on acceptance criteria on a JIT basis.
8. Tailor the project and team processes.
9. Solicit stakeholder feedback early and often.
10. Prioritize collaboratively.
11. Reprioritize as conditions change.
12. Engage an empowered business stakeholder.
13. Promote knowledge sharing.
14. Form working agreements.
15. Engage new stakeholders as needed.
16. Foster group decision-making and conflict resolution.
17. Establish a shared vision.
18. Ensure common understanding of success criteria.
19. Balance need for certainty with adaptability.
20. Cooperatively devise ground rules.
21. Encourage team members to become generalized specialists.
22. Discover team and individual motivators.
23. Co-locate or use collaborative tools.
24. Reduce distractions.
25. Align project and team goals through sharing vision.
26. Measure velocity in order to better predict.
27. Plan at multiple levels.

28. Include stakeholders in transparent planning.
29. Make increasingly specific commitments.
30. Use retrospectives to create better plans.
31. Adapt plans to reflect changes.
32. Initially create high-level estimates.
33. Refine estimates as your understanding increases.
34. Have appropriate team members resolve issues.
35. Maintain a visible, monitored, prioritized risk list.
36. Conduct frequent retrospectives.
37. Perform value-stream analysis¹

Then we sent a survey to experienced project professionals, three of whom had written or are writing books on Agile, with most of the others having had significant experience facilitating Agile projects. All have had at least 20 years of project leadership-related experience, generally running PMOs or consulting. We asked the survey respondents to identify up to 25 tasks they felt were most important. Eight of them responded. Of the 37 initial items, 12 were prioritized by more than half of the respondents, and another 11 were prioritized by exactly half of the respondents. Those items are listed in Exhibit 4. Once again, within their prioritization group of half or more than half, they are in the order listed in the *PMI Agile Certification Outline*.

Exhibit 4 Prioritized Agile Tasks

TASKS PRIORITIZED BY MORE THAN HALF OF THE RESPONDENTS

- Encourage collaboration.
- Practise servant leadership.
- Define and deliver products incrementally.
- Solicit stakeholder feedback early and often.
- Reprioritize as conditions change.
- Engage an empowered business stakeholder,
- Promote knowledge sharing.
- Cooperatively devise ground rules.
- Co-locate or use collaborative tools.
- Refine estimates as your understanding increases.
- Maintain a visible, monitored, prioritized risk list.
- Conduct frequent retrospectives.

TASKS PRIORITIZED BY EXACTLY HALF OF THE RESPONDENTS

- Maintain highly visible information registers.
- Tailor the project and team process.
- Prioritize collaboratively.
- Form working agreements.
- Establish a shared vision.
- Ensure common understanding of the success criteria.
- Discover team and individual motivators.
- Reduce distractions.
- Align project and team goals through vision sharing.
- Adapt plans to reflect changes.
- Have appropriate team members resolve issues.

To make these long lists actionable, we grouped those into these three logical groups, as shown in Exhibit 5.

Exhibit 5 Key Agile Tasks from PMI-ACP Outline Mapped onto Three Key Agile Mindset Ideas

Satisfy customers by placing emphasis on outputs that satisfy their needs,

- Engage an empowered business stakeholder.
- Establish a shared vision.
- Ensure common understanding of success criteria.
- Solicit stakeholder feedback early and often.

Engage the team through empowerment, cooperation, knowledge sharing, servant leadership and visible and continual communication.

- Cooperatively devise ground rules.
- Discover team and individual motivators.
- Prioritize collaboratively.
- Have appropriate team members resolve issues.
- Practise servant leadership.
- Co-locate or use collaborative tools.
- Form working agreements.
- Reduce distractions.
- Encourage collaboration.
- Promote knowledge sharing.
- Maintain a visible, monitored, prioritized risk list.
- Maintain highly visible information registers.

Simplify everything with a sustainable cadence and emphasis on process improvement.

- Tailor the project and team process.
- Define and deliver products incrementally.
- Refine estimates as your understanding increases.
- Reprioritize as conditions change.
- Adapt plans to reflect changes.
- Conduct frequent retrospectives.

Now we tie it all together in Exhibit 6 by showing both the values and beliefs that answer the question of why we do Agile and the tasks that show how, mapped together on the three simple ideas of customers, teams, and simplicity.

Exhibit 6: The Why and How of Agile

1. Satisfy the Customer

Why	How
<ul style="list-style-type: none"> Value customer collaboration <i>over</i> contract negotiation. Our highest priority is to satisfy the customer through early and continuous delivery of valuable <i>product</i>. <i>Embrace changing</i> requirements, even late in the process. Agile <i>processes</i> harness <i>change</i> for the customer's competitive advantage. 	<ul style="list-style-type: none"> Solicit stakeholder feedback early and often Engage an empowered business stakeholder Establish a shared vision Ensure common understanding of success criteria

2. Engage the Team

Why	How
<ul style="list-style-type: none"> Value Individuals and interactions over process and tools. Build teams around motivated individuals. Give them the environment and support they need. Trust them to get the job done. Agile processes promote sustainable delivery. The sponsors, people doing the work, and people using the work, should be able to maintain a constant pace indefinitely. The most effective method of conveying information to and within a team is facilitated face-to-face conversation with visualization. The best solutions and designs emerge from self-organizing teams. Everyone involved must work together regularly throughout the process. 	<ul style="list-style-type: none"> Maintain highly visible information registers Encourage collaboration Practice servant leadership Prioritize collaboratively Promote knowledge sharing Form working agreements Cooperatively devise ground rules Discover team and individual motivators Co-locate or use collaborative tools Reduce distractions Have appropriate team members resolve issues Maintain a visible, monitored, prioritized risk list

3. Simplify Everything

Why	How
<ul style="list-style-type: none"> • Value Working product over comprehensive documentation. • Value Responding to change over following a plan. • Deliver working product frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale. • At regular intervals teams reflect on ways to become more effective, then adjusts its behavior accordingly. • Working product is the primary measure of progress. • Simplicity – the art of maximizing the amount of work not done – is essential. • Continuous attention to outcomes and good design enhances agile processes. 	<ul style="list-style-type: none"> • Define and deliver products incrementally • Tailor the project and team processes • Refine estimates as your understanding increases • Reprioritize as conditions change • Adapt plans to reflect changes • Conduct frequent retrospectives

Finally, we offer some explanations of the values, beliefs and actions of Agile.

Satisfy customers by placing emphasis on outputs that satisfy their needs:

This is a broad view of what Agile would call delivering value. It is further defined by what value means. In the context of complex products, this normally means software. So, the idea here is to deliver working software with every iteration. This departs from the idea that a document or a test plan would be of value to the customer if the product of the effort is software. Although other items may be considered necessary by the customer for regulatory reasons, these could be considered the cost of doing business and should be minimized whenever possible, with the primary focus being the working software every two weeks or so. This prioritizes the customer's needs based on what the product being built is supposed to be.

Think about this in the context of a physical building. Although there is value in the models and drawings of the building, they are not in and of themselves the building. They are cost to getting a building. If you spend more time and effort on the drawing and models than is required, this surfeit becomes waste, no longer adding value to the product. This is a fairly easy concept to understand within the physical realm; however, for the world of complex products, it is not as clear. This is often because we do not understand how we get to good software. The Standish survey on software product value indicates that up to 80% of the software functions created are never used. That is a large number of hours spent designing functions that are not adding value. The way Agile approaches this problem is by asking the customer what would add value next, in short intervals. At some point along this journey, the customer will probably need to pause and digest what she or he has and get feedback from other users before requesting additional features. You will notice how different this approach is from the traditional requirements-driven model used in traditional software development.

What tends to make this Agile approach unique to some and revolutionary to others is that we have used the physical world of project management and applied it to the virtual

world of software products. This metaphor was useful in the early days of software when we were constrained by physics, whether those physics were in the form of real-time physical systems, like missile guidance systems, or whether those constraints involved the limitations of the computing power, memory or disk space. As we have moved away from the physical constraints and limitations, we have had a proliferation of software products. This proliferation has spawned increasing complexity. It has not necessarily driven a better understanding of how to deliver working product cheaper, better, faster.

Now that we are experimenting with alternative ways of delivering value to the customer, we have an opportunity to think differently about what that means.

To summarize, we need to deliver value in very short intervals and get feedback before we build more. This allows us, as creators of complex products, to pivot along the product life cycle.

Engage all participants through empowerment, cooperation, knowledge sharing, servant leadership and visible and continual communication:

Without servant leadership, there cannot be empowered teams. In reality we cannot empower anyone; they must empower themselves, and they will not do that as long as the risk of empowerment is larger than the risk of not doing so. The environment must create the structure, set the behaviours taught and learned, and then the servant leaders can act as guides and coaches on how each team and team member can become more self-organized. This is the journey that is “being agile.” Empowered teams come out of the process; we cannot start with teams that are empowered, especially if they have been other than empowered in the past. So, although this is one of the first things on our list and grouping, it is the result or benefit of the other areas discussed.

Cooperation is key to the development of self-organizing teams that behave in an empowered way over time. It is not easy to get cooperation within the corporate structures of today’s organizations. We have tried for years to have matrixed organizations that come together in an ideal way for short periods of time. However, we often do not allow enough time for the team development process to take place, and then if we do, we break up the team up as soon as the work is completed instead of having that high-performing team focus on the product and allowing the team members to grow with one another and the product. Cooperation does not come easily; there must be trust before there can be cooperation, and for true cooperation there is going to need to be space for conflict. So the servant leader will need to be comfortable facilitating conflict and at times help to draw it out. This is the way of progress and maturity for the team, without which there will be little improvement at being agile.

Sharing becomes organic in high-performing, cross-functional teams. There is a desire to break down barriers and to become expert on the product. This will only happen when the team feels a sense of ownership with the product and a sense of commitment to the team. There is a need to create a team identity, have ways of working and be able to exhibit good-natured fun and to give one another a hard time to make sure everyone knows that the team’s on the hook for the commitment. The knowledge sharing within Agile happens for the product, the process and the team. The more knowledge there is in these three areas, the more Agile can be practised, and the more likely it is for the team to truly become agile. As they become more agile, they develop less of a need for process.

All leadership, if done properly, is servant leadership. Now that is not to say that there are not leaders who want to be served. That is not the leadership style we refer to within the context of Agile principles. It does not mean that the leaders within the Agile movement are just there to do the bidding of the team either. Often people give the pat answer of the servant leader within Agile as being there to remove impediments, sometimes called blockers, in order for the team to meet its commitments. Although this is an example of how a leader may serve the needs of the team, it is but a trivial example. We see this example as a worthwhile activity for a servant leader only until that leader can help team members figure out how to remove their own impediments. Many traditional leaders are actually just managers, and the distinction I draw is that we manage *things* but lead *people*. We are only leading people as long as they are following. If they are following us, then we are leading. Following is something one does out of one's own volition. So there is a higher-level form of leadership in the Agile model than in traditional project management. Historically, in my experience the leaders within traditional projects were really managers, managing tasks and tracking progress through reporting.

The other key in the Agile organization is that there is a defused leadership model where there is not one servant leader. The traditional roles are now split into focus areas – process, product, the how and so on. Because there are more opportunities for leadership, there is a need for disciples to “stay within the swim lane” they are assigned to and refrain from commenting outside their remit. If this can be done and there are people in the servant leadership roles, we begin to see the benefits of this safety net of servant leadership designed to support and empower the team. We then begin to shift the focus from the leader onto the led.

To create, we must have conversation if we are creating together. The level of conversation becomes of higher quality as we add visual aids to that communication. We try to bring the physical realm into all team communication, whether that involves only moving a story card on a kanban board or using whiteboards or other visual means to communicate ideas, design or concepts. Let's face it: we have been communicating as a species much longer with drawings than with words. Visual communication supports a key concept of Agile, and that is transparency. With transparency we expose the lack of understanding and lack of clarity. Once exposed, we are able to address real issues as a team. The more we drive transparency, the better we can collaborate and create. It seems that the lack of transparency comes from a lack of trust, and that comes from how teams are formed and developed. If we can at least do benign things, for example document the work on a 3×5 card and talk about the card every morning, and if we are willing to have people ask questions, we have begun the process. Although this sounds trivial, we are amazed at how much better teams perform after they begin to do this. Many of the diagnostic tools we use with teams are by observation. If we see team members looking at the kanban board instead of looking at one another when talking about their work, we understand there is a lack of trust and openness. This is another aspect of visual within Agile: the value is in the seeing the state of the information, the work, the people or anything else that will impact the product. Once visibility begins to happen, we can start understanding what we may want to experiment with to improve our effectiveness. More on this is explained in a later section.

When we try to be efficient with our conversation, we set ourselves up for failure as a team. We can have short conversations when we work mostly independently of others. If we were able to deterministically break very complex creative efforts into discrete tasks and know we had addressed all the potential problems, we could be independent of the other creators. This is not the case with emerging creative works. How could it be? The person who is asking

us to create does not yet know what “good” looks like, so how could we, the team of people being asked to create, understand what “good” looks like? It is through continual conversation that we explore and learn what each team member is thinking and creating. It is through the ceremonies of Agile techniques like scrum that the team plans and reflects on what is working and what to improve next. It is through the system demonstration that we learn what the customer thinks of our work and understand what the customer would like us to apply our creative talents towards next. In parallel, the team also reflects on how the work was done for the prior iteration, typically two weeks, and considers how it will improve. This, again, is a conversation.

Keep things simple with a sustainable pace or (and) cadence and emphasis on process improvement.

Minimal process is a hallmark of Agile thinking. “Lightweight process” was how the first draft of the Agile manifesto described the thinking. It was later determined that the signatories did not want to be known as lightweights. It is one of the principles of Agile to figure out what *not* to do, termed *work not done*. We need to focus our efforts in that direction when we evaluate our ways of working. If we are trying to enable people not to have to communicate, we are unnecessarily complicating the process. So the more we communicate face-to-face, another Agile principle, the less process we need. This is obvious in our personal lives, but somehow in our business lives we have eliminated face-to-face communication in the name of efficiency.

In particular, for products that are complex and where the product cannot be well described, we actually need less efficiency and more conversation. The model within Agile practices is to start with a few principles and values and see what else is needed for it to work, looking towards reducing the need for process as the team matures. If a team that is self-organizing is able to stay together for longer periods of time and work on the same long-lived product, less process is needed. The process tree must be continually pruned to ensure that the processes being used by the team are as lightweight as possible for any given team at any given time. Without this relentless quest for the minimal viable process, we are doing the customer a disservice by slowing down the value delivery with unnecessary process.

One example of waste is how often within Agile practices teams try to get better with estimating their velocity instead of eliminating the need to estimate at all. If we could assure a customer that our teams will produce what they say they will when they commit and that they will continue to get better over time, the obsession with estimating would be waste and eliminated. A process once outlived becomes an impediment to producing more value more quickly for the customer.

The pace of creation must be measured and metred to be optimal. If we provide too much time between idea conception and delivery, we allow the work to expand and we bump up against the principle of work not done. If we allow that to happen because we do not understand the work that needs to be done, we will surely do too much. The “too much” will not be valued by the customer and therefore becomes waste. Pace plays a critical role in preventing waste by allowing the team to determine its pace and by not telling the team how much to get done in a time interval, but allowing it to continually ask itself how to get better through process improvement. Once we have our minimum time interval we need to ensure we pace ourselves to be able to create at a level of acceptable quality, and to have that pace for a long period of time.

This is the opposite of the legendary death marches software projects have historically undertaken to complete work on time. Once we establish our pace, we again reflect periodically on how we can improve upon that pace. It may be by investing in tools or by a new process, or even by the elimination of process. It may be through cross-training the team or other things the team may look at and experiment with. So, once a sustainable pace is created, it is then exposed to inspection, reflection and improvement, as are all the other areas of thinking in an Agile way.

Whereas pace speaks to how fast we are going, cadence speaks to how often we coordinate our activities. When we scale to larger efforts, we will have dependencies on other cross-functional self-organizing teams. These teams, like our team, are working through the groupings above, and although that is happening, we need to have a common cadence to ensure we are all building in the same direction when we have dependencies and need other teams to help us deliver value to our common larger customer. If we can do that coordination at predictable, predetermined times, we then are able to focus on our own team commitments without constant conversation.

Now, here is where some apparent contradiction may be seen. We want constant conversation at the team level, but we do not want constant conversation outside the team level. Correct. The creative unit is the team, ideally seven members, plus or minus two. That is where the magic happens. As communication channels get larger or when they are not continual, instead of continual conversation we need to have highly coordinated communication. Everyone being on cadence does this. Think of a marching band: if there were no cadence, there would be no music. The music is the creative process, and cadence allows that to happen in a synchronized manner. The mistake we often see is that we do not know well enough what the dependencies are so we expose all information at the higher levels, expecting someone else to identify our dependencies. This is not useful or productive. The mechanism used to prevent this behaviour is commitment. After these cadence events, teams commit to producing working product and demonstrating this product to the customer and eliciting feedback. This commitment, if taken seriously, will drive transparency and the opportunities to get better over time.

Woven throughout these three areas, we see the ideas of process improvement and the ultimate desire to eliminate the need for process. It is the combination of *kaizen*, meaning “continuous improvement,” and the idea of relentless reflection that helps us get to the ultimate utility of only delivering value to the customer. We are accustomed to hearing about continuous improvement, but the other side of the same coin is *hansei* – “relentless reflection.” This is the principle of stopping and reflecting on how to get better as a team. Within Agile, this is done with the ceremony of the retrospective; and at the larger scales, with inspecting and adapting. Although these are not new concepts within the software creation process, the discipline around these concepts is.

Again, we take well-known concepts and wrap them within a cadence and pace that allows the team to make changes at a very small batch size. Think how small a change we can experiment with to see a discernible difference, and then make that change and see what results. Instead of trying to optimize a creative process, just focus on improvement. We will see quickly with our experiments whether or not we are heading in a useful direction. Then we can pivot in the next iteration without an enormous amount of time, effort or reputation on the line. As we experiment, there be will times when there is a concerted focus on one aspect of a value or principle, or there may be a realization that some of our processes are obsolete.

There are certain constraints to the utility of our three groups as we begin to generalize for all process and the creation of all products, so we need to keep this discussion within the context of creating complex products that do not have a deterministic representation of good.

In this paper, we define how to shift an organization's mindset in order to capitalize on what Scrum and Kanban have suggested with regards to organizational constructs and how to shift the terminology to yield the promise of what it means to "be agile." We explain how to effectively implement the priorities and principles that continue to guide teams doing complex projects in a new way of thinking, as outlined in the Agile Manifesto. We caution that the longer the manifesto is preached in an organization without incorporating a shifted mindset, the more likely the organization is to lose the geometric benefits of the new organizational constructs and unique approach.

If an organization is truly able to understand and implement the principles of Agile in the context in which they were intended, the benefits will be notable, including the organization's increased value delivery by self-organizing teams. By making these adjustments, organizations will be able to unlock the innate ability of the combined knowledge of each individual. How long this process will take for well-established organizations is not well defined, which makes the transition largely based on how quickly leaders at all levels can understand and embrace Agile thinking.

Notes

1. PMI Agile Certified Practitioner (PMI-ACP) Examination Content Outline, Newtown Square, PA: 2014.
2. Kloppenborg, T.J., *Contemporary Project Management 3e*, 2015, Mason, OH: Cengage Learning, pp.296-303.

References

- Agile Faqs Managed chaos, *self-organized vs. self-managed vs. self-directed...What's the difference?*, accessed 13042018. <https://blogs.agilefaqs.com/2014/10/29/self-organised-vs-self-managed-vs-self-directed-whats-the-difference/>
- Allen, W., 2008, *Throughput vs. Velocity*, accessed 13042018. <https://weblogs.asp.net/wallen/throughput-vs-velocity>
- FourSquareviews, 2016, *Agile PM process grid-6.6 Agile tooling (1)*, accessed 13042018. <https://4squareviews.com/2016/01/22/Agile-pm-process-grid-6-6-Agile-tooling-1/>
- Hoogveld, M., 2018, *Agile management: the fast and flexible approach to continuous improvement and innovation in organizations*, Business Expert Press, New York.
- InfoQueue, 2014, *What are self-organizing teams?*, accessed 13042018. <https://www.infoq.com/articles/what-are-self-organising-team>
- Millhollan, C. & Kaarst-Brown, M., 2016, "Lessons for IT project manager efficacy: a review of the literature associated with project success," *Project Management Journal*, vol. 47, no. 5 (October/November 2016), pp.89-106.
- Nearsoft an Indecomm Company, 2014, *Are you ready for a self-managed agile team?*, accessed 13042018. <https://nearsoft.com/blog/are-you-ready-for-a-self-managed-Agile-team/>
- Nicolaas, D., 2018, *Scrum for teams: a guide by practical example*, Business Expert Press, New York.
- Paquette, P. & Frankl, M., 2016, *Agile project management for business transformation success*, Business Expert Press, New York.

Pichler, R., 2016, *10 tips for creating an Agile product roadmap*, accessed 13042018. <http://www.romanpichler.com/blog/10-tips-creating-agile-product-roadmap/>

Pichler, R., 2014, *From Personas to User Stories*, accessed 13042018. <http://www.romanpichler.com/blog/personas-epics-user-stories/>

Principles behind the Agile Manifesto, 2001, Agilemanifesto.org, accessed 13042018. <http://agilemanifesto.org/principles.html>

Project Management Institute, 2014, *Agile Certified Practitioner (PMI-ACP) examination content outline*, Newtown Square, PA.

Project Management Institute, 2017, *Agile practice guide*, Newtown Square, PA.

Scrum Alliance, 2013, *Self-organizing teams: what and how*, accessed 13042018. <https://scrumalliance.org/community/articles/2013/january/self-organizing-teams-what-and-how>

Sitepoint, 2014, *Three powerful estimation techniques for Agile teams*, accessed 13042018. <https://www.sitepoint.com/3-powerful-estimation-techniques-for-Agile-teams/>

Software Development Magazine *Methods and Tools*, accessed 13042018. <<http://www.methodsandtools.com/archive/archive.php?id=61>>

Telerik AD, 2013, *the Importance of timeboxing and iterations for Agile planning*, accessed 13042018. <https://www.telerik.com/blogs/the-importance-of-timeboxing-and-iterations-for-agile-planning>

Twelfth Annual State of Agile Survey, 2018, Version One, accessed 13042018. <https://explore.versionone.com/state-of-agile/versionone-12th-annual-state-of-agile-report>

United States Congress, 2016, *Bill S.1550 – Program Management Improvement Accountability Act*, accessed 13042018. <https://www.congress.gov/bill/114th-congress/senate-bill/1550>

Vanderjack, Brian. 2015, *The Agile edge: managing projects effectively using Agile scrum*, Business Expert Press, New York.

About the Authors



Frank M. Forte

For 30 years, Frank has led, and managed projects and programs around the world for clients such as the U.S. Navy, Air Force, and Army, NASA, CA, CSC, GTE, and Mylan Pharmaceutical. He has worked across many different industries, including Oil & Gas, Software, CPGs, Healthcare, and Construction. Today, Frank speaks, coaches, and consults for individuals and companies on how to affect change. Frank holds a BSCIS and has done graduate work in Software Engineering.



Timothy J. Kloppenborg

Timothy J. Kloppenborg, PhD, PMP is a Professor Emeritus from Xavier University. Tim has over 100 publications including 11 books such as: *Contemporary Project Management*, *Project Management for Archeology*, *Strategic Leadership of Portfolio and Project Management*, *Project Leadership*, and *Managing Project Quality*. Dr. Kloppenborg is the founding collection editor for Business Expert Press's portfolio and project management collection. He has lead thousands of people in consulting, training, and university classes on six continents.