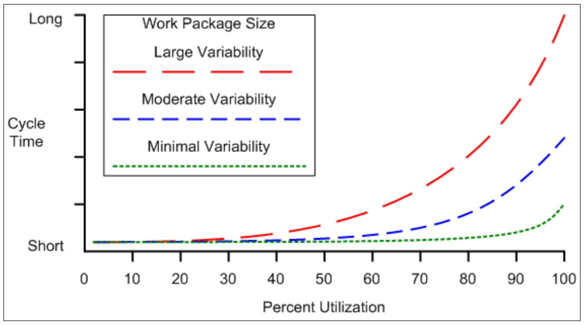
Agile Work Uses Lean Thinking

By Mishkin Berteig
Berteig Consulting Inc.

Lean and Agile are both methods of improving the effectiveness and performance of work processes. Lean comes primarily from manufacturing and in particular the Toyota Production System. Agile comes primarily from Agile Software Development and Project Management. Agile Work has borrowed heavily from Lean thinking and practices. There are three important connections between Lean and Agile: queueing theory, empirical process control, and team self-management.

Queueing Theory

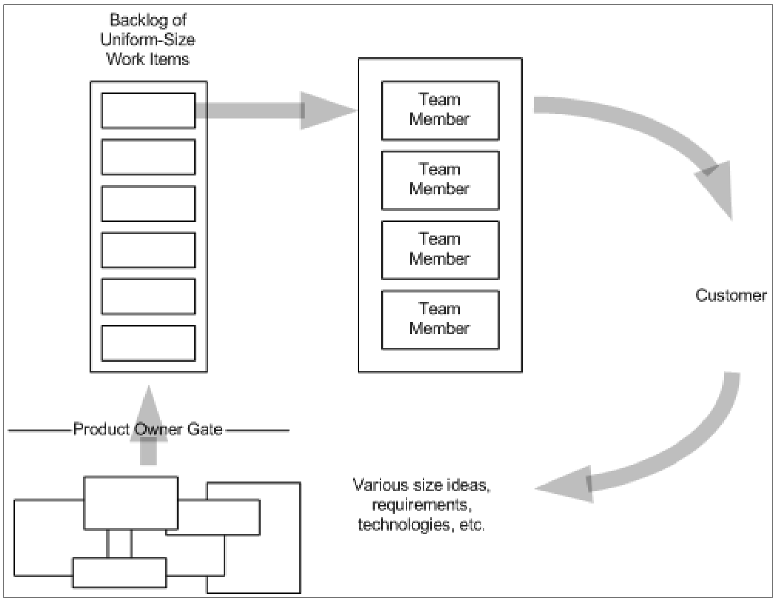
Queueing theory describes the statistical and theoretical behavior of queues. In a queue system, there are two basic parts: work items and worker units. Worker units perform some work upon work items. The streaming of work items through worker units composes a queueing system. The time it takes for a work item to enter the system and then exit the system is the process cycle time. From the study of real queueing systems and from simulation of queueing systems, researchers have shown there are some simple methods for creating an efficient queueing system that minimizes process cycle time.



One method for making a queueing system more efficient relates to the size of work items and how this relates to worker unit utilization levels and process cycle time. Queues behave in a very interesting way in relation to utilization and process cycle time. As utilization of a worker unit approaches 100%, process cycle time goes up exponentially. Not only that, but the more variability there is in the size of the work items, the worse this effect becomes. With a queue with work items that are all the same size, the worker units can maintain a very high level of utilization and the process cycle time is not significantly affected. However, with a queue with work items that are many different sizes, a worker unit will slow down significantly and process cycle times become worse even with relatively low levels of utilization. A simple example of this is a grocery store with a Ten-Items-Or-Less checkout lane.

Lean and Agile both use these properties of queues. Lean applies these ideas to manufacturing and other production processes and Agile applies these ideas to project work done by people.

In Agile, iterations are used to create consistently small sized units of work that are then taken on by a team. In other words, the work items are designed to be exactly the size of an iteration and the worker unit is the Agile team. Since iterations are typically much smaller than the size of the overall project and since each iteration is always the same size, this allows the team to achieve very high levels of utilization while maintaining extremely short cycle times (the length of the iteration or release). Compare this to the waterfall approach to project management where the work is only finally delivered at the very end of a long process and you can see that not only do you have a very long cycle time, but each project will be highly variable in size and therefore it will be hard to get high utilization out of teams (think resource planning and leveling).


Backlog of Uniform-Size Work Items
Team Member
Team Member
Team Member
Team Member
Customer
Product Owner Gate
Various size ideas, requirements, technologies, etc.

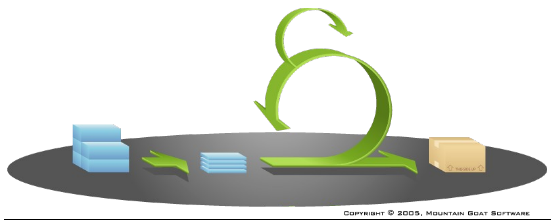
The Theory of Constraints, which is nicely introduced in The Goal by Eli Goldratt, presents some additional basic techniques for making improvements in efficiency of a process. The basic idea is that one can always find a slowest point or constraint in a process by finding out where there is a buildup of unfinished work. For example, if two people are cleaning a kitchen, one washing dishes and the other drying, if the number of wet washed dishes keeps building up, then the constraint for the process is the person drying. In more complex processes either in manufacturing or in creative work such as software development, it is sometimes more difficult to see where the constraint is. Typically, if you find that you have people waiting for work that is coming from someone else, then that someone else is a constraint and means can be found to improve their efficiency. In Agile, the focus on resolving the constraint would be to provide that person extra training or get other members of the team to assist in the work. Agile tends to shy away from a mechanistic perspective on efficiency.

Finally, in any queueing system there is some point at which work enters the system. This point of entry is very important because it can be used to control the utilization levels of worker units in a queue. In Agile Work, this control is accomplished through backlog management, iterative delivery and adaptive planning. All possible requests, features, constraints, improvements for a project are put into a master Work Item List. This list is strictly prioritized in descending order by a person empowered with this responsibility (called the Product Owner). This list of work items becomes the basis for deciding what work the team will do each iteration. The process of managing this list and how the work is chosen for this iteration allows the customer/client to prioritize important things to be delivered quickly and allows the team to work on consistently sized work units (iterations) and therefore achieve very high levels of utilization. In queueing theory this process is referred to as the gating function in that it provides a gate that lets work items into the system. All work must go through this gate or work item list management process in order for the team to function effectively. If the team is interrupted with work that has somehow skipped the gate it will seriously reduce the efficiency of the team (e.g. a senior sales person comes to the team and declares that it must work on X, it'll only take a day, a potential client really needs this, surely that won't hurt?).

Empirical Process Control

Some work processes cannot be perfectly controlled nor perfectly defined. There may be non-linear interactions between steps in a process or there may be creative input from people required. Processes with these qualities require empirical process control.

The basic attribute of empirical process control constitutes a continuous cycle of inspecting the process for correct operation and results and adapting the process as needed. A simple example of this is detecting impending failure of equipment by constantly monitoring the operation of that equipment.



Copyright © 2005, Mountain Goat Software

In human processes like those to which Agile Work applies, the frequency of inspecting and adapting must match the needs of the process. Many projects occur in the context of constant change. This constant change makes long-term planning a wasteful effort. Rather, short-term planning with constant feedback provides a simple inspect and adapt cycle. This cycle can play out at different levels: daily for a team, monthly for a client of the team. The team inspects and adapts daily at the level of the tasks that it is performing. The client inspects and adapts monthly at the level of the team's actual delivered results.

Lean and agile methods claim to increase both speed and quality. There are four constraints in a system that can be controlled: speed (or schedule, or time to market, or cycle time), quality (number of defects), scope (how much functionality), and cost savings (how much to spend on the work). Frequently, we believe that one has to trade off between these four constraints; spend more money, get more scope; lower quality, go faster. Lean and agile strongly support the idea that as you increase quality, you also increase speed... you just have to do it right.

In Agile Work, increasing speed and quality is done in three ways. First, increase the frequency and quality of communication among team members so that errors are detected early or avoided altogether. Second, drive the work with the creation and execution of automated testing. No work is done without a test in place to check if it is done correctly. This constant testing means that work is always defect-free and therefore very little time/money is spent on fixing defects. Third, eliminate wasteful work steps or obstacles to performance of work. This last one is difficult to do an bears closer examination.

Wasteful work is done in every process, no matter how efficient. Lean tells us that there are several types of waste in a manufacturing process. Those types of waste have analogies in Agile Work. For example, documenting something you plan to do instead of just doing it is wasteful. Another example is waiting while someone completes work that you depend upon. Any step or task that does not add value to the final product of an effort is waste. This standard is very high

SIDEBAR: Waste?

CVA (Customer Value Added - or just VA for Value Added): adding form fit or function to a product or service, an activity that the customer would be willing to pay for in isolation if they knew it was being done – e.g. Creating code, implementing functionality.

BVA (Business Value Added – non-negotiable waste): an activity that is required to operate the business but the customer is unwilling to pay for – e.g. Budget tracking, code documentation.

NVA (Non-Value Added – pure waste): an activity that is not required by the business nor is the customer willing to pay for – e.g. Waiting for resource allocation, requirements documents.

and most organizations have about 80% of their efforts going into wasteful tasks. An organization that has done an initial cut of wasteful work might stand at about 50% waste. The leanest organizations, such as Toyota, stand at about 20% waste.

Agile work eliminates waste in the form of barriers or obstacles that come up when a team is trying to go fast. Sometimes this is in the form of waiting for another group to do something for the agile team... an outsourced request for service. Sometimes waste is in the form of corporate standards or policies around documentation of work. The Process Facilitator role in an agile team has responsibility for working with the team and others to help overcome these obstacles.

Team Self-Organization

The people who actually do work on a day to day basis know that work intimately... more intimately than anyone else. Anyone else who knows the work either knows it historically or in theory, but not in the same intimate manner. This intimate knowledge carries a great deal of potential. In order to release that potential, people must accept individual responsibility and collective responsibility. Management and organizational culture must support that responsibility. Lean and agile have a very similar approach to supporting this self-organization.

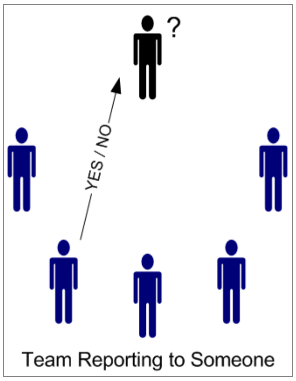
The first aspect of this support is just a simple recognition of this intimate knowledge. The people doing the work must have their expertise acknowledged both individually and collectively.

The second aspect of this support is to share with these people the strategic and tactical "why" of what they are doing. This can include sharing financial models, strategic assessments, etc.

The third aspect of this support is in allowing individuals and teams to create and implement their own process improvements. In lean this focuses on "identifying and eliminating waste" and in agile this focuses on "identifying and removing obstacles".

The fourth aspect of this support, and perhaps the most important, is that of self-organization: teams and individuals organize themselves around the work. Managers no longer have the authority to tell people how to do their jobs, nor do team members report status to their managers. Rather team member report status to the rest of the team by answering three essential questions:

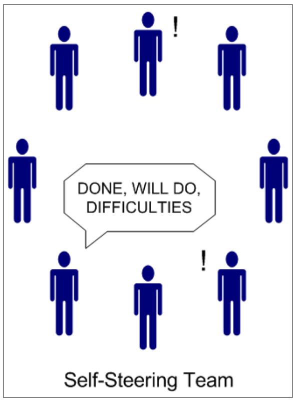
1. What have I done since the previous status report?
2. What will I do before the next status report?
3. What difficulties are in the way of me doing my work?



This distinction is depicted in the figures on the left and the right.

In agile teams, this concept of self-organization is taken quite far. Team members collaborate to get work done. No one orders a team or an individual to do specific work. The team members volunteer for work that they see needs doing, even if it is not something that is in their area of expertise. An agile team is constantly promoting learning in its people. Agile teams are also cross-functional so that the team can get work done without relying on external services. The team therefore represents a complete work unit capable of taking a function valuable to customers from start to finish, from idea to deployment.

One aspect of lean systems that is not commonly practiced in an agile environment is the idea of stopping the production line when a flaw or defect or error is discovered. In lean manufacturing, every person working on the line has the authority to stop the whole line if they notice something wrong. Then, everyone works on correcting the defect all the way to the root cause of that defect before starting the line again. This is a very powerful mechanism for making certain that there is constant improvement in the production process. Giving people the power and authority to stop the line takes a great deal of trust.



Conclusion

Both Agile Work and Lean Thinking have a great deal to offer organizations looking to improve their effectiveness. However, to truly reap the rewards possible from these approaches requires a deep cultural transformation where management, customers and employees develop a high degree of trust in each other so that all involved can safely pursue the intense and ongoing learning necessary to become Agile, to become Lean.

About Mishkin Berteig

Mishkin Berteig leads, mentors, trains and coachs teams and organizations using agile methodologies including Agile Work, Scrum and Extreme Programming. Mishkin has served as a project manager, a senior consultant, a mentor, a methodology consultant, an instructor, a senior software architect and a team lead on various projects, mostly in the financial services industry. Mishkin has 15 years of professional experience. Mishkin publishs articles and thoughts about Agile Work, Lean thinking and team dynamics on Agile Advice – Thoughts and Experiences for Practitioners of the Middle Way to Excellence. For more detail, please view Mishkin's online resume [pdf].

About Berteig Consulting Inc.

Berteig Consulting Inc. specialize in helping organizations adopt agile practices and disciplines in project management, product development, business management and human resources development. Berteig Consulting Inc. offers Agile Work speaking, training, consulting, coaching and strategic assessment services around the globe including New York, San Francisco, Toronto, Beijing, and London. Berteig Consulting Inc. is the corporate sponsor for the Agile Advice web site. For more information please visit our web site at http://www.berteigconsulting.com/ or call us at +1 416 559 1919 or email us at sales@berteigconsulting.com.

Resources

Books

The Goal - Eli Goldratt

Lean Software Development - Mary and Tom Poppendieck

Good to Great - Jim Collins

Agile Software Development with Scrum - Ken Schwaber and Mike Beedle

Web Sites

The "Thinking" Production System - http://www.toyota.co.jp/en/special/tps/tps.html

Agile Alliance – http://www.agilealliance.org/

Agile Advice – http://www.agileadvice.com/

Queuing Theory Resources - http://www2.uwindsor.ca/~hlynka/queue.html