

Like this Knowledge Brief? Find more like it at the Rapid Learning Cycles Resource Center:

<http://rapidlearningcycles.com>

Here are a few other Knowledge Briefs that might interest you:

- [The Structure to Grow Ideas into Solutions: The Rapid Learning Cycles Framework as the Engine of Innovation](#)
- [Rapid Learning Cycles Are More Fun: Eleven Reasons Why Teams Like the Rapid Learning Cycles Framework](#)
- [Flexibility for World Class Performance: Five Ways that Rapid Learning Cycles Adapts to Fit the Needs of Your PD Programs](#)



About the Author

Katherine Radeka has a rare combination of business acumen, scientific depth and ability to untangle the organizational knots to remove the barriers to change. Since 2005, Whittier Consulting Group, Inc. has helped some of the world's leading companies get their products to market faster.

She has a global reach with clients in Europe, North and South America, Asia, and Australia/New Zealand. She has worked with companies in pharma, biotech, medical device, high tech, consumer electronics, food and beverage, and consumer packaged goods, among others. She currently supports more than 75 implementations of the Rapid Learning Cycles framework through the Rapid Learning Cycles Certified™ Professionals Community.

Katherine is the author of two books. Her first book, *The Mastery of Innovation: A Field Guide to Lean Product Development* won the Shingo Research Award in 2014. This book contains 19 case studies of companies, including Steelcase, Ford, Novo Nordisk and Phillips Electronics, who have used lean ideas in product development to get their ideas to market faster.

Katherine's second book is *The Shortest Distance Between You and Your New Product: How Innovators Use Rapid Learning Cycles to Get Their Best Ideas to Market Faster*. This book summarizes Katherine's ground-breaking work to integrate Agile Development with her work on Knowledge Capitalization into a proven method for accelerating innovation.

Katherine has climbed seven of the tallest peaks in the Cascade Mountains and spent ten days alone on the Pacific Crest Trail until an encounter with a bear convinced her that she needed a change in strategic direction.



Elements of the Rapid Learning Cycles Framework

A Glossary of Key Concepts and Useful Terms

By Katherine Radeka



Elements of the Rapid Learning Cycles Framework

A Glossary of Key Concepts and Useful Terms

A - C

Activities: Tasks that need to be performed in order to finish a Product Development Program. It includes both learning activities to support closing knowledge gaps, such as experiments, and other activities to write documentation, finalize drawings, etc.

Activity Plan: A plan for tracking a team's progress at the Activities level. It is the lowest level plan that a team uses to manage its work.

Agile Software Development: A movement within the software development community to transform how teams managed software development to dramatically improve the final product. It is named for the Agile Manifesto, published in 2001, and it emphasizes ongoing customer engagement, iterative development practices, continuous delivery of working software and respect for the developers.

Balanced Scorecard: A performance management tool that seeks to reduce the negative side effects of measuring performance strictly by financial measures, first developed by Robert Kaplan and David Norton. The Scorecard consists of four types of metrics: Financial, Customer, Operations, and Learning & Growth.

Cadence: Rhythm. In the context of the Rapid Learning Cycles framework, a regular time interval in between Learning Cycle Events that establishes the "heartbeat" of the Rapid Learning Cycles framework, or the time interval between regular Status Events.

Cadenced Release Plan: A strategy for releasing products within a product family that begins with the release of a Minimum Viable Product and then times successive product enhancements to regular market pull events, such as seasonal windows or annual trade shows.

Convergent Decision Making: A process for making a complex decision that converges to the final decision in a series of stages that reduce the number of options or scenarios in a set by probing the set for weaknesses. It is a generalization of Set-Based Concurrent Engineering to apply to more types of decisions in more problem domains.

Core Hypothesis: A short description of the vision for a product that summarizes the specific customer, technical and business objectives that the product would achieve.

E - K

Extensible Knowledge: Knowledge that has value outside of the specific product for which it was developed, and that is captured and shared so that future product development teams can leverage it.

Innovation Master: The person responsible for developing the vision for a new product and then driving the product development program to completion. This is not a project or program manager who is primarily concerned with schedules and budgets - the Innovation Master is primarily concerned with driving the vision, coaching the team, aligning with stakeholders and addressing risks.

Integration Event: A pull event focused on Key Decisions that includes the team plus the stakeholders needed to make the Key Decisions that are on the agenda for the event. It usually triggers a major update to the team's Learning Cycles Plan.

Key Decision: A major decision that must be made for the product development program to go forward, and that the team does not currently have the knowledge to make.

Key Decision Flow: A visual model showing the sequence of Key Decisions that need to be made up to the next major program milestone (such as a gate review). It is best shown as a PERT chart or a simple diagram rather than a GANTT chart.

Key Decision Report: A single page report that summarizes the knowledge the team has developed in order to make a Key Decision, and the team's decision and/or recommendation; it is presented at the Integration Event where the Key Decision will be made.

Kickoff Event: An event at the start of a program, or at the program's major milestones to establish the Key Decision Flow and Learning Cycle Plan for the next phase of the program. It is a face-to-face meeting with the entire team, plus any stakeholders who may have Knowledge Gaps of their own to close in the upcoming phase.

Knowledge Gap: Something that the team needs to learn in order to make a Key Decision or move the program forward in some other way. By definition, each Key Decision has one or more Knowledge Gaps.

Knowledge Gap Report: A single page report that summarizes the knowledge the team has developed in order to close a Knowledge Gap, and their interpretation of how this knowledge impacts the related Key Decision.

L - O

LAMDA: LOOK-ASK-MODEL-DISCUSS-ACT - a version of the Scientific Method that is tailored for the work of engineering. It was developed in the early 2000's by Dr. Allen Ward of the University of Michigan to foster systematic problem solving among engineers.

Last Responsible Moment: The last point in time when a decision must be made to avoid major impacts on downstream partners.

Lean Startup: A strategy for organizing the work to build a product in the context of a startup company working on Web 2.0 or other software. It was popularized by Eric Ries, and based in the work of Steve Blank at Stanford. It needs heavy adaptation to work for tangible products that have manufacturing, distribution and/or regulatory certification challenges to overcome.

Learning Cycle: A fixed amount of time for learning that begins and ends with a Learning Cycle Event to capture and share the knowledge gained in the last Learning Cycle, and plan the next Learning Cycle.

Learning Cycle Event: A pull event focused on Knowledge Gaps that is primarily for the team to share knowledge and build plans for learning in the next Learning Cycle. It usually triggers a minor update to the Learning Cycles Plan.

Learning Cycles Plan: A plan that maps out the Knowledge Gaps and Key Decisions that will be the team's focus for the next several Learning Cycles. It works best as a Visual Plan with color-coded sticky notes to represent Key Decisions and Knowledge Gaps.

Minimum Viable Product: A product that meets the absolute minimum requirements for features, quality, safety and reliability to deliver significant value to a real paying customer. This concept was developed by Steve Blank and popularized by Eric Ries.

P - S

Phase Gate Life Cycle: A product development process that organizes development work into phases with "gates" in between to give management the opportunity to review the team's progress and the viability of the business case before committing to the next phase of work. It was popularized in 1995 by Robert Cooper, who trademarked his variant of the process.

Product Development Process: A standardized process for organizing the work of product development that is usually supported with checklists and templates to coordinate across functional groups, ensure that certain best practices are actively used, and assist teams in preparing for major management reviews.

Pull Event: A meeting that is planned in advance for a specific date - that does not move unless there is a national emergency. The fact that the Pull Event cannot be delayed means that problems, new information and knowledge does not stay hidden, and ensures that the team stays in stronger alignment throughout the product development program.

Rapid Learning Cycles Framework: A framework for product development that organizes work into a series of Learning Cycles that focus the team on building knowledge. It pulls learning forward and pushes decisions later to reduce risk and preserve flexibility.

Scientific Method: A systematic approach to building knowledge that develops and tests hypotheses with experimentation. It is the root source for LAMDA, PDCA, Six Sigma's DMAIC and every other systematic problem solving method.

Scrum: A specific form of Agile Software Development popularized in the early 2000s by Jeff Sutherland and Ken Schwaber. This method was distinct for its 28 day "sprints" to plan work and daily "scrums" to maintain coordination within the team.

Set Based Concurrent Engineering: A problem solving method developed by Dr. Allen Ward in the early 1990s, and that he claimed to find inside Toyota in the late 1990s. In this method, teams work concurrently to narrow down on a final system design in a series of stages that evaluate sets of alternatives for each subsystem.

Stand Up Meeting: A meeting conducted in front of a Visual Plan with no chairs, and fast reports from each member of the team: what got done, what will get done, what's in the way. The goal for time is fifteen minutes or less.

Status Event: A pull event for tracking Activities, conducted on a daily to weekly basis, to minimize the time spent on project management and ensure that problems come to the surface quickly. Usually performed as a Stand Up Meeting.

T - Z

Timebox: A concept from Agile Software Development that limits the amount of time available to complete a task. If a task cannot be completed in the allotted time, more time may - or may not - be granted to complete it. The Learning Cycle is a timebox for learning - teams bring the knowledge they have built to the Learning Cycle Event even if that knowledge is incomplete.

Visual Plan: A form of lightweight project plan that has its roots in Lean Manufacturing and in Agile Software Development. Teams prepare their plans on simple whiteboards or large sheets of paper with sticky notes, markers and color-coded dots. These plans are much faster to update and build greater team alignment than project plans kept on Gantt charts in project management software. 