

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: Rapid Learning Cycles as the Engine of Innovation](#)
- [The Power of Meaningful Progress: How Rapid Learning Cycles Keep Teams Motivated, Creative and Productive](#)
- [From Idea Generation to Final Decision: Divergence and Convergence in the Rapid Learning Cycles Framework](#)



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 over 50 implementations of Rapid Learning Cycles 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.



Simple, Flexible and Forgiving

Why the Rapid Learning Cycles Framework Has Staying Power

By Katherine Radeka



Simple, Flexible and Forgiving

Why the Rapid Learning Cycles Framework Has Staying Power



Key Takeaways

- **It's easier to stick with a new idea that is simple, flexible and forgiving, because teams have a lot of room for error and the ability to adapt the idea to their specific challenges.**
- **Rapid Learning Cycles are simple and flexible: the number of elements that must be done exactly right are few, and there is a lot of room for adaptation.**
- **Rapid Learning Cycles are forgiving: it just takes a replanning session with the team to get them back on track if they've made a mistake or encountered a surprise.**

Several years ago, I started using the Body by You workouts developed by Mark Lauren as my primary form of strength training. It's the only strength training exercise I've ever stuck with for this long, and the only one where I've managed anything like consistency after the initial excitement wears off. I've come to believe that the program is sticky because it's simple, flexible and forgiving. I've long had the short routines memorized. The program consists of body weight exercises, so I need no special clothes or equipment. I can literally do it anywhere and if I miss a few days, I can pick it right back up at the next opportunity.

Rapid Learning Cycles is sticky - once it gets into an organization that has given it an honest try, teams continue to do it. Teams continue to do it even if they learned an early version of the process that didn't work as well as the framework does now. They continue even if there is some management resistance, as long as they're not explicitly told to stop. They may eventually try it in a few places where it doesn't work well, and they may need to make some heavy adaptations in some areas of their business. But they tend to keep at it. That's because it is simple, flexible and forgiving.

Rapid Learning Cycles are Based on a Simple Concept and Simple Tools

The Rapid Learning Cycles framework is built on the simple idea that pulling learning forward leads to better decisions, and pushing decisions later gives teams more time to learn. Better decisions lead to a much easier time in mid to late development, when teams begin to realize the benefits of good decisions and the drawbacks of the ones that need to be revisited.

When decisions get made at the right time with the right knowledge, teams stay on schedule, they deliver more good news to their business partners and their downstream partners find them a lot easier to work with. They deliver better products. They get to spend more time on the "fun stuff" of experimentation, exploration and pushing the boundaries of the company's technology. They have to spend less time fixing problems, dealing with grumpy customers or manufacturing partners, or defending the decisions that led to cost overruns. I

It's not a cure-all; organizations still need to learn how to balance their portfolios to capacity so that they avoid overloading their engineers with too many projects. Stakeholders will have to take more responsibility for the ways in which their decision-making has helped or hindered their teams. PDP Process Owners will need to figure out how all of this integrates with product development standards in regulated industries. But since the underlying concepts are simple, they're also useful as guideposts for dealing with the inherent complexity of product development.

You don't need to purchase any software to use Rapid Learning Cycles. You just need a pile of sticky notes, a whiteboard and a willing team. You don't need to send your entire team off to training camp; it's helpful to send a Program Manager to a workshop, but I know of lots of implementations from people who have read my book and the resources on the web site, which is fine with me. Once a team has been through a kickoff event once, they know what to do after that.

Rapid Learning Cycles are Flexible

The Rapid Learning Cycles framework is a framework - not a methodology, a system or a process. The framework supports the team's product development process without dictating it. There are a few things that have to be there for the framework as a whole to do its job of upholding the team, but we've tried to keep those to an absolute minimum, so that we can focus teams' attention on the parts of the framework that contribute to structural integrity.

We have teams using the Rapid Learning Cycles framework to develop products with biology, chemistry and pure physics, as well as myriad bundles of mechanical, electrical, software and services. If you purchase a car, a new washing machine, a hearing aid, a heart monitor or a new office chair in the next two years, chances are that some components of that product will have been developed using the Rapid Learning Cycles framework. In the next five years, you may fly in an airplane or drive a truck that has components that were developed using this framework. In the next ten years, your doctor may prescribe something to you that was originally investigated using the framework, before entering clinical trials.

We have teams using it at the very beginning, to investigate opportunities for innovation, and we have teams using it post-launch to quickly resolve customer issues and do fast customization work for customers. We are starting to get our first released products where the Rapid Learning Cycles framework has been in use from the moment the product was first an idea.

The simplicity of the core concepts and the flexibility of the structures that comprise the framework give teams what they need to understand how to adapt the Rapid Learning Cycles framework to meet their specific needs - even when those needs dramatically change, or the team gets sidetracked.

Rapid Learning Cycles are Forgiving

It's great that the basic framework is simple and easy to do, once a team knows how. It's great that it can be adapted into new applications. But I think the most important reason why the Rapid Learning Cycles framework sticks is that it is forgiving.

Get pulled off on another project for three weeks or three months? Pick up the Learning Plan where you left off. The team gets sidetracked getting ready for a major prototype build or gate review and "forgets" to write up their reports? They can just write the reports - it only takes a couple of hours to write a report since we're not asking for a lengthy technical review. Is the team too optimistic about how much they can do in a Learning Cycle? Rebalance the Learning Cycles Plan to the true capacity of the team.

If a team acquiesces to a stakeholder demand to delay a Key Decision, then at least everyone has a lot more clarity about the source of the delay, and the consequences. If a major stakeholder can't attend an Integration Event and wants to re-hash all the decisions made without him, the program manager has the knowledge pulled together to review the decision, and the consequences of changing it now (the reason why this was the Last Responsible Moment).

Once a team has been through some basic training and a Kickoff Event, then they have the skills they need to get back on track. They just need to update their Key Decisions, Knowledge Gaps and Learning Cycles Plan based on where they are now. They may need to do some ruthless prioritization if deadlines are now tight, but the framework gives them the tools to make sure that they are focused on the most important things.

I even had a team recently that got a little bogged down by the complexity of their Learning Cycles plan when the program manager got a little too busy on a second project, and he didn't have time to maintain it. The program manager just assigned another member of the team to take responsibility for the Learning Cycle Plan, and she held a re-planning session to bring it back up to date. The team was back on track right after the planning meeting, and they made their next gate review on time.

As long as the team has Key Decisions to make, Knowledge Gaps to close and Knowledge Debt to eliminate, then they have the raw materials they need to get back on track, and figure out how to make the most of the time they now have left.

Finally, Rapid Learning Cycles are Probably More Fun

My partner wrote an entire Knowledge Brief on this recently, and it may seem like a strange way to describe something that's intended to improve productivity and efficiency. But it's true: I can tell when a team has embraced the framework and is using it as well as they can, from the energy in the room. It's much more fun to be learning, experimenting, exploring, pushing the limits of technology and getting closer to our customers' real needs - and getting the product out the door, seeing it for sale online or in a store, and seeing it out in the world. That's a lot more fun than fixing warranty problems, delivering bad news to partners and customers, or trying desperately to get a manufacturing line running again.

Teams that use the Rapid Learning Cycles framework get to experience a lot more of the good stuff with much less of the bad stuff, and they get to share those experiences with their teams, sponsors, business partners and ultimately their customers.

Sometimes, male colleagues ask if they can help me with my travel backpack. They are usually astonished at how heavy it is and how easily I carry it, especially because I don't look very athletic. Thanks to the stickiness of my Body by You workouts, I can throw that bag into the overhead bin as if it was stuffed with feathers. 