JOIN OVER 100,000 PROS! Join Log In | Member Benefits

STICKYMINDS

Agile Testing DevOps Security AI

Book Review: Kanban: Successful Evolutionary Change for Your Technology Business

ARTICLE

By Steve Berczuk | February 26, 2012

SUMMARY

I've worked with Scrum for a while, having gotten my CSM certification in 2005, and I've spent time both before and after that trying to learn what I could about Scrum, agile, and Lean, both in the context of software and out side of it. After absorbing bits of information on Kanban informally, I decided that to was time to read a book on it. I read David Anderson's book Kanban: Successful Evolutionary Change for Your Technology Business.

I've worked with Scrum for a while, having gotten my CSM certification in 2005, and I've spent time both before and after that trying to learn what I could about Scrum, agile, and Lean, both in the context of software and out side of it. After absorbing bits of information on Kanban informally, I decided that to was time to read a book on it. I read David Anderson's book **Kanban: Successful Evolutionary Change for Your Technology Business**.

Anderson's book was a good introduction to Kanban in practice. It had a mix of stories, theory and guidelines that gave a good sense that the content of the book was based on practice. My only reservation about the book was that while it was well written in many spots, there were places where it was a bit less focused that I would have liked. But if you are willing to skim through the less tight parts of the book, it is worth a look. Here are some of the things I learned while reading *Kanban*.

Before I read about Kanban in detail my initial impression was that people new to agile, who didn't find Scrum appropriate for their project, were enthusiastic about Kanban because was that they were unwilling to be disciplined enough to manage their work queue. It's hard to move from one's current approach (either a long term rigid planning approach, or chaos) to an approach that required the discipline to work in the context of

We use cookies to ensure you get the best experience on our website.

Accept

Read More

Much of what I learned in Anderson's book about Kanban sounded familiar (not surprisingly) based on what I know about other agile methods. You can use an entirely Kanban-based approach to a project. But, there are elements of Kanban that you can overlay, on say, a Scrum project. The concepts such as limiting work in process (and making that limit explicit) seems line with that Scrum teams I have worked on have done. Scrum teams can certainly benefit from a deep understanding of Kanban principles much as a Scrum project is likely to fail without applying technical practices such as those of XP. (For more a more in depth comparison you can have a look at the article by Henrik Kniberg and Mattias Skarin
on infoQ,)

Kanban is less structured than Scrum in terms of boundaries and commitments. While in some sense this sounds promising for teams that wrestle with "interrupts" like "critical production issues" it also means that there is less of a framework to identify and address the underlying problems, and the path to improving your process might be slower.
Kanban can be useful. If you use your imagination, you might even think that Kanban is *the limit of Scrum as iteration length approaches 0.* If teams have trouble managing 2 week iterations, they are likely to have problems with no iterations. At least until teams internalize agile practices and values fully, Scum is probably still a better fit for teams wanting to adopt agile. (See **Ken Schwaber's comments** for a more strongly worded view on this).

While there are aspects of Kanban that will inform my Scrum practice, I think that for those trying to adopt agile, Scrum is a better approach. Kanban might seem to work well, and present less of a barrier to managing a hard to manage backlog, but keeping work in line and maintaining the sustainable pace and slack that both Scrum and Kanban advise can be much harder. Rather than being an easier way to agility, Kanban seems to require more discipline, and starting with Kanban might not the the right answer for many teams.

Topics: lean,

ABOUT THE AUTHOR

Steve Berczuk

Steve Berczuk is a Principal Software Engineer with experience as a manager, Scrum Master and technical lead in Boston, MA. The author of *Software Configuration Management Patterns: Effective Teamwork, Practical Integration*, he is a recognized expert in software configuration management and agile software development. Steve is passionate about helping teams work effectively to produce quality software. He has an M.S. in operations research from Stanford University and an S.B. in Electrical Engineering from MIT, and is a Certified ScrumMaster. Contact

We use cookies to ensure you get the best experience on our website.

UPCOMING WEBINAR

LETS HANG!

FEATURED RESOURCES

SECURITY

Shifting Security Left: DevSecOps eGuide

White Paper | StickyMinds

Log In or Register to post comments

USER COMMENTS

1 comments

| February 28, 2012 - 12:08pm EST

Your comment:

"they were unwilling to be disciplined enough to manage their work queue"
is a generalization and more of an indictment of the teams in question than it is the Kanban process.

We use cookies to ensure you get the best experience on our website.

YOU MAY ALSO LIKE

SECURITY

Agentic AI Opens Up New Security Flaws

Nahla Davies | August 27, 2025

SECURITY

Countering the Threat of Ransomware-as-a-Service (RaaS)

Nahla Davies | August 22, 2025

SECURITY

Building Trust in Blockchain: The Role of Smart Contract Testing

Ekaterina Egorova | November 18, 2024

Join the crew for FREE! Access exclusive content, contribute, and more.

See All Member Benefits

We use cookies to ensure you get the best experience on our website.