

# Kanban

from the  
Inside

Understand the Kanban Method,  
connect it to what you already know,  
introduce it with impact

**Mike Burrows**

Foreword by Luke Hohmann

# Kanban from the Inside

*Understand the Kanban Method,  
connect it to what you already know,  
introduce it with impact*

*Mike Burrows*



Sequim, Washington



Blue Hole Press  
72 Buckhorn Road  
Sequim, WA 98382  
[www.blueholepress.com](http://www.blueholepress.com)

Copyright © 2014 by Michael Burrows  
All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage or retrieval system, without permission in writing from the publisher.

Printed in the United States of America.

Library of Congress Cataloging-in-Publication Data  
applied for; available at [www.loc.gov](http://www.loc.gov)  
ISBN: 978-0-9853051-9-2 (paperback)

Cover design by Jane Pruitt  
Interior design by Vicki L. Rowland

10 9 8 7 6 5 4 3 2 1 0

# ❖ CONTENTS ❖

Foreword	ix
Preface	xi
<b>Part I Kanban Through Its Values</b>	<b>1</b>
<b>1 Transparency</b>	<b>3</b>
Core Practice 1: Visualize	4
Core Practice 4: Make Policies Explicit	7
Core Practice 5: Implement Feedback Loops	9
Transparency as a Value	13
<b>2 Balance</b>	<b>15</b>
Core Practice 2: Limit Work-in-Progress (WIP)	15
Pull Systems for Knowledge Work	15
Balance Workload versus Capacity	17
Other Ways to Limit WIP	18
Balance Urgency-Driven versus Date-Driven Work	19
Risk-Based Categorization and Classes of Service	21
Balance Demand versus Capability	22
Stakeholder Balance	22
Seeking Balance	23
<b>3 Collaboration</b>	<b>25</b>
Core Practice 6: Improve Collaboratively, Evolve Experimentally	25
Collaboration Is a Thing	25
Improve Collaboratively	26
Encouraging Collaboration	26
Focus on Collaboration	27
Evolve Experimentally	28
Them and Us	29
<b>Reflection: Transparency, Balance, and Collaboration</b>	<b>30</b>
<b>4 Customer Focus</b>	<b>31</b>
Core Practice 3: Manage Flow	31

Why Customer Focus? _____	31
Satisfaction Assured _____	32
Right Across the Board _____	33
Upstream Kanban _____	34
Anticipating Needs _____	36
<b>5 Flow</b> _____	<b>37</b>
Core Practice 3: Manage Flow (Again) _____	37
Smoothness _____	37
Right Across the Board (Again) _____	39
There Is Always a Bigger Context _____	40
Some Low-Level Examples _____	41
Flow Across the Larger Organization _____	42
Managing Flow for Timeliness _____	45
<b>6 Leadership</b> _____	<b>47</b>
Foundational Principle 4: Leadership at Every Level _____	48
Opportunity Everywhere _____	48
Leadership Begets Leadership _____	49
Why Value Leadership? _____	50
What Does Leadership Look Like in Kanban? _____	51
<b>Reflection: Customer Focus, Flow, and Leadership</b> _____	<b>52</b>
<b>7 Understanding</b> _____	<b>53</b>
Change without Understanding—	
Three Anti-Patterns _____	54
Introducing the J Curve _____	57
A Pattern for Purposeful Change _____	59
<b>8 Agreement</b> _____	<b>61</b>
Pursue Evolutionary Change _____	61
Agree to Pursue _____	63
Change Management _____	64
<b>9 Respect</b> _____	<b>69</b>
Don't Start with Roles _____	69
“Be Like Water” _____	70
Respect for People _____	71
The Humane, <i>Start with What You Do Now</i> Method _____	74
<b>10 Patterns and Agendas</b> _____	<b>75</b>
“Noble Patterns” _____	75

Agendas for Change	76
The Sustainability Agenda	77
The Service Orientation Agenda	78
The Survivability Agenda	79
Your Organization's Agenda	79
How Did We Do?	80
Five Years On	81
<b>Part II Models</b>	<b>83</b>
<b>11 Systems Thinking, Complexity, and the Learning Organization</b>	<b>85</b>
Systems Thinking	85
Complexity	88
Knowledge, Learning, and the Learning Organization	91
Systems Thinking and Kanban	95
Further Reading	97
<b>12 Theory of Constraints (TOC)</b>	<b>99</b>
The Five Focusing Steps and the Process of Ongoing Improvement (POOGI)	100
Drum-Buffer-Rope (DBR)	101
The Thinking Processes	102
Critical Chain Project Management (CCPM)	103
Throughput Accounting	105
TOC's Relationship to Kanban	106
But Still . . .	108
Further Reading	108
<b>13 Agile</b>	<b>109</b>
Three Agile Methods	110
Kanban and Agile	116
The Agile Model	118
Further Reading	119
<b>14 TPS and Lean</b>	<b>121</b>
Three Lean Tools	122
TPS and Lean in Perspective	124
Lean Improvement	125
Lean Product Development	126
Lean Startup	128
Lean/Agile Hybrids	129
Kanban and Lean	130
Further Reading	131

<b>15 Economic Approaches to Flow</b>	<b>133</b>
ROI and the Pareto Diet	134
Cost of Delay	138
Cost of Carry	142
Options	143
Putting It All Together	146
Further Reading	147
<b>16 The Kanban Method</b>	<b>149</b>
A Very Brief Timeline	149
Foundational Principles	151
Core Practices	151
Contextualized Kanban	152
Enabling Concepts and Tools	155
Implementation Guidance: STATIK	156
Further Reading	157
<b>17 Smaller Models</b>	<b>159</b>
Two That Got Away	159
Thinking Tools and Coaching Models	163
Group Facilitation and Games	167
Models of Collaborative Leadership: Triads and T-Shapes	170
Further Reading	171
<b>Part III Implementation</b>	<b>173</b>
<b>18 Understand Sources of Dissatisfaction</b>	<b>175</b>
Two Perspectives	176
Two Questions	176
Formats	177
Organize and Explore	178
Follow Up	179
<b>19 Analyze Demand and Capability</b>	<b>181</b>
Know What You're Delivering, to Whom, and Why	182
Quantitative Analysis	185
How Work Arrives	188
Does It All Add Up?	189
<b>20 Model Workflow</b>	<b>191</b>
Sketching It Out	191

Top-Down Decomposition _____	193
Bottom-Up Organization _____	194
Review _____	195
<b>21 Discover Classes of Service _____</b>	<b>197</b>
Discover, Check _____	198
Examples _____	199
Toward a Healthy Mix of Work _____	199
<b>22 Design Kanban Systems _____</b>	<b>203</b>
Scope, Work Item Granularity, Work Item States _____	203
Other Dimensions _____	207
Multi-Level Designs and the Expand/Collapse Pattern _____	208
Limiting WIP _____	210
Review _____	212
<b>23 Roll Out _____</b>	<b>215</b>
Planning the Engagement _____	215
Shaping the Agenda: The Three P's _____	217
Pulling Change Through the System _____	219
Closing Thoughts _____	228
Further Reading _____	229
<b>Glossary _____</b>	<b>231</b>
<b>Appendix A _____</b>	<b>239</b>
Deming's 14 Points for Management _____	239
<b>Acknowledgments _____</b>	<b>241</b>
<b>About the Author _____</b>	<b>243</b>
<b>Index _____</b>	<b>245</b>

## ❖ CHAPTER 4 ❖

# Customer Focus

### Core Practice 3: Manage Flow

Is this a mistake? How do we get from *Manage Flow* to **customer focus**? Indulge me for a moment—let me cheat a little, expanding the wording of this core practice to express more fully what this practice really means:

**CP3** (*expanded*): Manage flow, seeking smoothness, timeliness, and good economic outcomes, anticipating customer needs.

This chapter focuses on customer needs and how to anticipate them better. Smoothness and timeliness are covered in the next chapter, on **flow**. Keep in mind “good economic outcomes” as you read both chapters; economic decision-making is covered in Chapter 15.

### Why Customer Focus?

Task focus, role focus, team focus, project focus, product focus, company focus, technology focus . . . the list goes on. So many ways to lose sight of what we’re in business for!

In my classes I offer this advice:

*Know what you’re delivering, to whom, and why.*

You might think that this could go unsaid, but it really seems to hit home. Unsolicited, students tell me that they’d never really thought about it before. They mention it in feedback sheets as a key takeaway. It’s not that those other focuses are bad, but that customer focus helps to put them all into proper perspective.

In this chapter, we explore some practical ways in which customer focus can improve the flow of work, sometimes profoundly. It's not so surprising when you think about it: If you look at what you do from a someone else's perspective, you are likely to learn something new about how it works.

## Satisfaction Assured

Recall this *policy* from the scenario that opened Chapter 1:

- ♦ *Developers retain responsibility for work items until they have obtained customer confirmation that the item is proving its worth.*

This policy was a relatively late addition. We had evolved a development process that seemed effective enough. We'd gather requirements, build new features, test them, and release them. After a while, we got a little more sophisticated: We added a column on our board that let us track features that were released but still required further implementation steps before they could be considered complete.

Too often, though, when we checked, we found that we'd delivered features that would never be used. Features that had been asked for! How does that happen?

Our new policy was added to address what we assumed at the time to be bad customer behavior. Why ask for stuff you don't need? How about letting us know when you change your mind? However, it soon became apparent that this new policy was changing behavior on both sides. Closing a *feedback loop* was the catalyst for a level of *customer collaboration* (a value straight out of the *Agile Manifesto*<sup>10</sup>) not previously seen.

Knowing that the process was going to end in what could turn out to be a difficult conversation, developers and our internal customers alike made sure to nail those final implementation steps (clarifying timetables, keeping people suitably informed and trained, getting static data cleaned up, and so on). When necessary, these steps would be tested beforehand, often collaboratively. That, in turn, influenced the way development and specification were done. All the way back at the start of the process, it changed even the way work got prioritized, now that it was apparent that success depended on shared commitment.

---

10. See Chapter 13 for more on the *Agile Manifesto*.

I'm not exaggerating when I say that the impact of this policy change went way beyond my expectations. Some humility is in order too: We didn't have bad customers, just relationships that weren't effective enough.

## Right Across the Board

Our catalyst was a policy attached (figuratively speaking) to the right-hand column of our kanban board, customer focus somehow infecting the whole process. To understand how a transformation like ours might be repeatable, it is helpful to review the board with some specific questions. Working right-to-left, column-by-column:

- ◆ Whose needs are explored in this stage of the process, and how? Whose aren't, and what risks does that pose?<sup>11</sup>
- ◆ What do we learn in this stage that we don't (or can't) know earlier? In what ways do the activities of this stage help us home in on what will be needed?
- ◆ What is still to be learned? Are outstanding uncertainties best dealt with by pressing on or by going back?

Working that logic all the way back to the start of the process, no longer are we building to meet given requirements, but building to meet needs that are still to be discovered and explored. Not forever looking backward, justifying ourselves, proving that we are building “correctly to spec,” but looking forward, working toward meeting needs that are still unfolding. Neither do we put undue trust in a supposedly watertight process to do our work for us; instead we seek ways to capture learning more effectively.

*Creative knowledge work* isn't just about what we already know. It is (and I use a piece of technical jargon quite deliberately) a *process of knowledge discovery*. Use your kanban board to keep reminding you: “What don't we know?”

---

11. Don't forget internal stakeholders here—audit, security, finance, support, and so on. If they hold a veto, they should in some sense be treated as customers. See Middleton, Peter and James Sutton. 2005. *Lean Software Strategies: Proven Techniques for Managers and Developers*. New York: Productivity Press.

## Upstream Kanban

Let me give you the flavor of one of the more ambitious board designs that I have used to manage my personal workload. The key feature of this design involves the three columns under “IDEAS.” Notice how their descending WIP limits seem to suggest a funnel.

IDEAS			COMMITTED		
Priority 3 15	Priority 2 8	Priority 1 5	Ready 4	In Progress 3	Complete

**Figure 4.1** An example Personal Kanban board design

This kind of design is helpful when you’re as concerned with organizing the ideas and tasks you could be doing in the future as you are with managing your current workload. There are two key tricks to operating it effectively:

1. You train yourself to mentally blank out the leftmost columns until it’s time to replenish or reprioritize columns farther to the right. Soon this becomes an unconscious habit, and there’s absolutely no doubt that you’re operating a pull system. When you are ready to replenish the “Ready” queue, you’ll be able to do that entirely from the relatively small number of items (just five) currently in “Priority 1,” which is much easier than looking at the entire pool of ideas.
2. You balance your eagerness to add new ideas to the board with a willingness to remove items that are never likely to make it to the “Ready” queue. Removed items can be archived for later review or completely destroyed (I like to move individual items quickly to a holding area “below the line” and deal with them periodically en masse).

WIP limits are, of course, the reminders to practice these two personal disciplines. Once the board starts to fill up, you will find yourself deprioritizing work much more often. This is healthy!

My inspiration for this design is the *priority sieve*, a *Personal Kanban*<sup>12</sup> technique. It might seem odd to digress into Personal Kanban now (we return to it briefly in Part II), but I use this design as a model for *Upstream Kanban*, a name coined for the practice of operating a kanban system upstream of the delivery process. Upstream Kanban is about organizing needs and developing ideas so that there are always good choices on offer when delivery capacity becomes available.

This design reinforces two concepts that the portfolio managers of some of our largest corporations seem to forget:

- ◆ We generate more ideas than we can possibly use—in fact, we’d have reason to worry if it were otherwise. Over time, we will accumulate more ideas than we can usefully manage, let alone implement.
- ◆ Ideas cannot proceed on their own merits alone—they are in competition with others. Moreover, new ideas can enter the competition at any time.

Something special happens when items move into the “Ready” column: This system has a very unambiguous *commitment point*. To the left of it, we’d be happy to see items moved backward or discarded—that’s the system fulfilling an important part of its purpose. To its right, we’d think that something had gone wrong if items failed to progress with good speed, and we’d regret the waste of effort if items were abandoned before completion.

Just as it does inside the delivery process, effectiveness upstream depends on the values we’ve explored so far:

- ◆ **Transparency:** The system must make visible the difficult choices that need to be made. The decision-making rationale should itself be explicit. Decisions are the focus of feedback loops (prioritization meetings, for example).
- ◆ **Balance:** The amount of WIP in the system is controlled, both to maintain a reliable supply of high-quality ideas and to force timely decision making. If needed, additional control can be gained by allocating WIP by customer, budget line, risk category, strategic initiative, and so on.
- ◆ **Collaboration:** The work of qualifying items for further development is shared among the originators of those items and the peo-

---

12. Benson, Jim and Tonia DeMaria Barry. 2011. *Personal Kanban: Mapping Work | Navigating Life*. Seattle: Modus Cooperandi.

ple who will service them. Instead of sucking risk into the system prematurely, all parties (and there may be several involved) keep their options open until commitment is timely.

Let's see what **customer focus** adds:

- ◆ Whose needs do we think are met by these ideas?
- ◆ Are we meeting needs fast enough?
- ◆ What is the data telling us? What are people telling us?
- ◆ What might lie behind those needs?
- ◆ What needs might be going unmet?
- ◆ How can we test that?

In short: Can we develop a better sense for what will be needed?

## Anticipating Needs

If there's a single idea that I'd like you to take away from this chapter, it's making the mental shift away from doing what is asked, taking orders, fulfilling requests, meeting requirements, and so on, and reorienting the process toward discovering and meeting needs. It's a shift from an internal perspective (what we think we know) to an external one (what's still out there to be discovered). It's also a shift from the past (what we've been told) to the future (when the customer's need will be met).

That emphasis on the future is captured very nicely in the closing words of the Toyota Customer Promise. I found this displayed on a plaque behind the customer service desk at my local Toyota dealership:

*... anticipating the mobility needs of people and society ahead of time*

Think of a service on which you personally rely. Wouldn't you be delighted if the provider anticipated your needs ahead of time? What innovations might they need to introduce in order for that to happen? Can you translate that kind of thinking into your workplace?

## ❖ CHAPTER 14 ❖

# TPS and Lean

In 1978, the year of his retirement from Toyota, executive vice president Taiichi Ohno published a book describing the *Toyota Production System* (TPS). His book shared a remarkable number of concepts and tools that were barely known outside of Japan, such as:

- ◆ *Just-in-time (JIT)*—the radical idea that the right materials, parts, and assemblies should arrive where they’re needed only as they’re needed, and in the smallest possible quantities
- ◆ *Autonomation*—“automation with a human touch,” the production line’s early warning system
- ◆ The *andon* system—a visual indication of trouble combined with the means for ordinary shop-floor workers to “stop the line”
- ◆ The *Five Whys (5W)*—a technique for root cause analysis
- ◆ *Kanban*—the card-based system by which just-in-time production is managed

Over the following decade, interest in Toyota grew as its position in the global car market strengthened. In 1998 an English translation of this seminal book at last became available. Two things strike you as you read it: the brilliant simplicity of the system and its tools, and Ohno’s remarkable strength of purpose as he strove to develop a system capable of realizing the vision of the company’s founders, the Toyoda family.

Not that most of us were reading Ohno back then. The book that generated popular interest in Toyota’s unusual methods and brought the term *Lean Production* into the mainstream was *The Machine That Changed the World* by James P. Womack, Daniel T. Jones, and Daniel Roos, published

in 1991. Two years later Womack and Jones followed this up with *Lean Thinking*. Japan had Toyota; the world had Lean.

Another raft of Japanese words entered the lexicon:

- ◆ *Kaizen*—continuous improvement through incremental change
- ◆ *Kaikaku*—radical change
- ◆ *Heijunka*—production leveling; that is, deliberately mixing work on the production line rather than producing similar work in batches
- ◆ *Poka-yoke* (or *Baka Yoke*)—mistake-proofing
- ◆ *Gemba, gembatsu, genjitsu*—the “three reals,” the real place where work is done, the real thing, and real facts, respectively
- ◆ *Hoshin Kanri*—strategic planning and policy deployment

And the list goes on. In retrospect, the early years of Lean did a great job of packaging up Toyota’s tools and techniques for western consumption but somehow failed to give much insight into the thinking that generated them. This began to change as writers such as John Shook, Mike Rother, and Steven J. Spear focused on Toyota’s approach to management.

### Three Lean Tools

A detailed explanation of how a production line works at Toyota is beyond the scope of this book. With just enough detail to allow comparisons to be made with the kinds of kanban systems described in Part I, here is a simple setup that uses *kanban*, *heijunka*, and *andon*:

- ◆ No parts or materials get supplied, and nothing—neither subassemblies nor the finished product—gets built without an appropriate *kanban*.
- ◆ At the delivery end of the production line, *kanban* are<sup>47</sup> pulled one at a time from the *heijunka* box. Slots in this box (or rack) organize the *kanban* in two dimensions, with time of day along one axis and product type along the other. By choice and by physical constraint, production of each product type must be spread across the day.
- ◆ The workstation (perhaps a *cell* with multiple workers) at the front of the production line assembles and packs the order to the specifi-

---

47. “Kanban are like sheep.” One kanban, two kanban, . . .

cation given on the *kanban*. As the supply of parts here drops below a replenishment level, *kanban* are sent upstream.

- ◆ And so the process continues, subassemblies moving down the production line, *kanban* moving upstream all the way up to Goods Inwards, where parts and materials must be ordered from outside suppliers . . .
- ◆ . . . until someone pulls the *andon* cord because they've spotted a problem; alarm lights come on and the line comes to a halt. After investigating and rectifying the problem, production starts up again.

This description is rather simplistic, but there's enough here for some striking characteristics of the system to be noted:

All three tools (*kanban*, *heijunka*, and *andon*) are examples of *visual management*.

- ◆ Inventory of all kinds is limited. Neither basic supplies nor WIP (the subassemblies, or partially-built products) will be replenished until equivalent amounts have been pulled from downstream.
- ◆ Even though, perhaps, it might seem more efficient to do so, the production line doesn't work in large batches of similar items. Instead, it produces a variety of products spread over the course of the day.
- ◆ Workers on the production line would rather *stop the line* (for everyone) than allow work of inferior quality to proceed.
- ◆ This system and the *kanban* boards from Part I work very differently. On the production line, the *kanban* are sent upstream to signal that there is demand to be fulfilled. On our boards, the cards represent work items as they flow downstream; signals are implied by the gaps between the actual amount of work in progress in each state and the corresponding WIP limits.
- ◆ The *heijunka* box and our *kanban* boards both allow the mix of work to be managed.

It seems perverse, not only setting things up to work in this deliberately difficult and seemingly inefficient manner, but empowering workers to bring it all to a halt at any time! Clearly there must something special about

the company's culture for this to work at all, but why would they choose to do things this way?

## TPS and Lean in Perspective

To answer that question you must understand TPS as a magnificent example of Systems Thinking.

It starts with a vision, a *true north* that gives the direction for change:

- ♦ *Single-piece flow, in sequence, on demand, with zero defects; 100% value-adding activities and security for the people performing them*

The technology does not yet exist to make it economical to run the entire production line in batches of one (which is what single-piece flow means), but the pursuit of this perhaps impossible vision is what propelled Toyota from its struggles in postwar Japan—where land, factory space, plant, and materials were all in short supply—to the global market leadership position that it now occupies.

The tools support one or both of two purposes:

1. Satisfying customer demand as quickly and as smoothly as (currently) possible with the minimum amount of inventory
2. Evolving the company to take it closer to its vision, harnessing the abilities of its entire workforce to smooth flow, reduce inventories, prevent defects, eliminate other forms of waste, and (not least) design new products that customers really want and that can be produced both profitably and sustainably

The two pillars of *just-in-time* and *respect for people* are shorthand for those sub-goals.

Often missed is this crucial point: The pillars and the tools can be seen in their proper perspective only once it is grasped that Toyota's pursuit of perfection is a multi-generational challenge. Toyota works not only to build cars, but also to build the company capable of delivering on its vision.

Divorced from that kind of thinking, the tools of Lean can seem shallow. Without the tools, it can be even worse—too often we hear Lean reduced simply to a short-term focus on waste (perhaps to dress up exercises in cost cutting), or to continuous improvement (important, but very hard to sustain in isolation). The challenge of the Lean movement is to make

sure that the thinking is packaged up with the tools so that people can apply them appropriately in context.

## Lean Improvement

Much of that Lean thinking is built into and around these five improvement steps:

1. **Identify value** from the customer's standpoint.
2. **Identify the value stream**—the value-creating steps in the process—and seek to eliminate what is non-value-adding.
3. **Create flow**, removing delays between those value-creating steps, seeking smoothness.
4. **Establish pull**, where work is taken from upstream only in response to downstream demand, ultimately from the customer.
5. **Identify waste**, removing impediments to smooth flow, reducing delays, reducing inventories, eliminating defects at the source, and so on.

These steps are often referred to as *Lean principles* (a slight misnomer perhaps, but never mind). Applied repeatedly, we have an improvement cycle analogous to the POOGI loop of Theory of Constraints (Chapter 12), replacing POOGI's specific focus on constraints (usually understood to be constraints on throughput) with attention to delays, smoothness, and WIP.

Inside that improvement cycle, Lean (and Toyota before it) embraces PDCA (Chapter 3) as the way to frame each incremental change. Larger changes may be documented and planned in an *A3* (named after the paper size), and developed over a period of time in the context of a mentor/mentee relationship of some kind. Smaller changes might be managed through various styles of structured dialog known as *katas* (another Japanese word, imported this time not from Toyota but from the martial arts; it means a choreographed pattern or routine).

Drilling down into steps 2 and 5 of the Lean principles, Ohno identified *seven wastes*, or non-value-adding activities:

1. **Transportation**—a source of delay, cost, and risk of loss or damage

2. **Inventory**—materials, work-in-progress (WIP), finished but undelivered goods—wasteful both for the delays incurred and for the cost of financing it
3. **Motion**—damage to people and equipment caused by the production process
4. **Waiting**—time spent by work items in inactive states
5. **Over-processing**—doing more work than is necessary to meet specifications
6. **Over-production**—producing work in excess of immediate demand
7. **Defects**—effective capacity wasted on bringing inferior work up to the required standard

Taken as an integrated whole, the Lean principles, the wastes, the visual management tools (*kanban*, *heijunka*, *andon*, etc.), and the management practices (*kaizen*, *A3*, *hoshin kanri*, etc.), are far from shallow. Not only are there few systems as well documented as this, it takes time and effort to fully appreciate the thinking that went into it and the theory that underpins it. In short, Lean is a significant body of knowledge.

## Lean Product Development

The Lean thinking described here so far still leaves a significant gap. Bluntly, Toyota's true north will never be reached on a diet of improvement alone. "On demand" and "security for people" need more than just production excellence—they need eager customers, a pipeline of good products, and sustainable profits.

*Lean Product Development* (LPD) seeks to address this gap. Relative to Lean manufacturing, it is quite young, expanding at a rate that allows for significant diversity. Some of the different approaches to LPD are represented here by a selection of its authors. This is by no means an exhaustive survey, more a list of some of LPD's key thinkers who have influenced Kanban's development:

- ◆ Donald G. Reinertsen takes Lean back to its first principles in queuing theory and economics, allowing the ideas and techniques of both Lean manufacturing and Lean Product Development to be applied outside their original domains much more effectively. The

influence of Don's book *The Principles of Product Development Flow: Second Generation Lean Product Development* extends well beyond the Lean community. The Kanban Method might not exist were it not for Don's personal interest in David Anderson's work; *classes of service* and *cost of delay* are two Kanban concepts directly attributable to him.

- ◆ Michael N. Kennedy is the author of *Product Development for the Lean Enterprise* and *Ready, Set, Dominate*. The “Set” in the latter title refers to *set-based learning*, highlighting a crucial difference between how Toyota manages manufacturing and product development (with TPS and TPD respectively). Instead of the one-thing-at-a-time approach of single-piece flow, set-based approaches organize the parallel search for solutions that capture the appropriate combination of technical feasibility, cost-effectiveness, and attractiveness. In software development, where physical limitations are usually unimportant, we instead like to *defer commitment*, not so much exploring the solution space as allowing for multiple possibilities, buying flexibility for little or no cost.
- ◆ Peter Middleton and James Sutton weren't the first writers to consider the applicability of Lean principles to software development; neither were they the first to call out the inappropriateness of mass production as a metaphor for creative knowledge work. However, their book *Lean Software Strategies* is commendable for its commitment to customer needs, asserting that a deep understanding of the most lasting customer needs (which they refer to as *values*) leads not just to more fruitful relationships but to more enduring products and product architectures as well.
- ◆ Last but by no means least, Mary and Tom Poppendieck took the Lean principles, the wastes, and other concepts from Lean manufacturing and mapped them to a set of principles more appropriate to software development. There's no doubt that their books<sup>48</sup> had a significant impact on both Agile and more traditional thinking long before LPD fully established itself as a discipline.

---

48. Adding to the previous chapter's personal note on influential books and former line managers, Dr. Peter Lappo, who holds the unique distinction of having been both my line manager and my wife's (not simultaneously), lent me a copy of the Poppendiecks' first book soon after its publication in 2003.

It seems safe to assume that LPD will continue to develop along pragmatic lines, integrating the following:

1. Practices extracted from Toyota and other leading companies, made sufficiently separable from their host companies that they can be adopted elsewhere with some confidence
2. Practices developed and found to work well in particular contexts, some of which may with time turn out to be more broadly applicable, some not
3. A strong theoretical underpinning, some of it extracted from the previously mentioned practices, some of it imported from other fields, the rest derived from first principles and tested in the field

This process benefits not just product development but Lean manufacturing, Agile, and Kanban, too.

## Lean Startup

Eric Reis's Lean Startup model takes product development into areas of extreme uncertainty, where basic things like customers, business models, and even the basic shape of the product are largely unknown. Here, failures are hard to avoid, but they should be painful, not catastrophic (I have in mind the widespread economic damage inflicted by the dot-com crash of the late '90s). Like Lean manufacturing, Lean Startup has two principal approaches to change.

Its continuous, incremental mode (analogous to *kaizen*) is organized around an experimental improvement loop called *Build/Measure/Learn* that bears strong similarities with PDCA:

- ♦ **Build** the smallest possible product increment that can test a hypothesis.
- ♦ **Measure** the real-world impact of this increment.
- ♦ **Learn** from the results, changing assumptions or building on them.

This approach is highly suited to web-based services, where techniques such as *continuous delivery* and *A/B testing* allow products to evolve very rapidly.

When continuous evolution runs out of steam, something more radical is needed. Lean startup's *kaikaku* is the *pivot*. This rather abused term often

gets taken to mean “throwing the original idea away and starting again,” but it is meant to indicate a significant but disciplined “course correction” that results from the invalidation of specific fundamental assumptions.

My special interest in Lean Startup and the allied discipline of *Lean UX* (the “UX” standing for *User Experience*) stems from what they do to feedback loops. This isn’t just a philosophical point—from current first-hand experience inside one of the UK government’s “digital exemplars,”<sup>49</sup> I know that you don’t need to be working for a startup business to benefit from this kind of approach.

Recall from the previous chapter how XP moved the focus of feedback loops “rightward,” so that they are centered not on requirements or design but on the code. Lean Startup adds feedback loops whose focus is even further to the right, keeping under constant examination the interactions between the product and its customers, looking to validate (or otherwise) the assumptions about customers and their needs that drive the development of the product.

Lean Startup continues a trend away from *requirements*, the basic raw material of traditional projects. XP introduced *user stories*, “a placeholder for a conversation.” Lean Startup replaces these with *hypotheses* that need to be validated. When you’re that committed to putting your assumptions to the test, your relationship to work-in-progress changes quite fundamentally.

## Lean/Agile Hybrids

There’s no doubt that Lean has much to teach Agile, and vice versa. But things can get a bit clunky when one has to keep saying, “I get *X* from Lean, and *Y* from Agile.” So why not create a hybrid of the two, combining Agile practices with Lean’s scale, say?

This is not without risk. In the absence of the values, Agile practices are hardly Agile at all. Lean tools without Lean thinking? Same problem. And the bigger and more rigid the hybrid’s process, the less sure we can be that it will suit the organization into which it is introduced.

As I write, it is far too early to tell whether off-the-shelf Lean/Agile hybrids (of which SAFe is the best-known example) will achieve anything

---

49. See <https://www.gov.uk/transformation>

like the cultural impact of their antecedents. In the meantime, I would much rather give Lean and Agile the separate credit they each fully deserve, hoping that their intelligent and respectful application in context is what is generally meant by “Lean/Agile.” I would hope to see a healthy dose of Kanban in there, too, of course!

## Kanban and Lean

Let’s turn the tables and look at the Kanban Method from Lean’s perspective.

Kanban has, quite explicitly:

- ◆ Visual pull systems, represented in Part I by the values of **transparency** and **balance**
- ◆ Respect for people (including much of what goes with that concept inside Toyota) in the form of **collaboration**, **leadership**, **understanding**, **agreement**, and **respect**

Kanban doesn’t incorporate Toyota’s True North, leaving each organization to determine its own fitness criteria. However, the values of **customer focus**, **flow**, and **respect** map to Toyota’s True North very well, and they each suggest some good things to measure.

From Lean’s perspective, Kanban could be described as the fruit of the following process:

1. Taking two of Lean’s tools—*kanban* and *heijunka*—and radically “reimagining” them for use in creative knowledge work
2. Observing that the implementation of these tools can have organizational impact
3. Identifying from successful implementations a set of principles and practices
4. Growing a new body of knowledge around those principles and practices

This version of history rather ignores the roles played by TOC and Agile in Kanban’s development, but still, it is the case that Kanban is highly aligned with Lean, is built on part of it, and, through the practice of *using models*, continues to refer back to it with some enthusiasm.

As a method and as a community, Kanban maintains some distinctiveness, however:

- ◆ We tend not to adopt Lean jargon, especially—and with the irony duly noted—the Japanese terms. There is enormous respect for and interest in the Lean and TPS heritage, but we take some effort to use plain language where possible.
- ◆ We are uncomfortable with the use of manufacturing as a metaphor for creative knowledge work. Neither do we wish to be remembered mainly for *waste* and *kaizen*; important as these are, creative knowledge work is as much about maximizing future opportunity as it is about optimizing current process.

All that said, we're still far from done with Lean. The next chapter owes much to Don Reinertsen; Part III (Implementation) has Lean influences also.

## Further Reading

- Middleton, Peter and James Sutton. 2005. *Lean Software Strategies: Proven Techniques for Managers and Developers*. New York: Productivity Press.
- Ohno, Taiichi. 1998. *Toyota Production System: Beyond Large-Scale Production*. New York: Productivity Press.
- Poppendieck, Mary and Tom Poppendieck. 2006. *Implementing Lean Software Development: From Concept to Cash*. Upper Saddle River, NJ: Addison-Wesley.
- Reinertsen, Donald G. 2009. *The Principles of Product Development Flow: Second Generation Lean Product Development*. Redondo Beach, CA: Celeritas.
- Ries, Eric. 2011. *The Lean Startup*. New York: Crown Business.
- Rother, Mike. 2009. *Toyota Kata: Managing People for Improvement, Adaptiveness and Superior Results*. New York: McGraw-Hill.
- Spear, Steven J. 2009. *The High-Velocity Edge: How Market Leaders Leverage Operational Excellence to Beat the Competition*, 2nd ed. New York: McGraw-Hill.

## ❖ CHAPTER 23 ❖

# Roll Out

I find it useful to think of Kanban implementation as a three-stage process:

1. **Planning the engagement:** preparation in terms of participants, venues, tools, supporting material, and so on
2. **Shaping the agenda:** approaching STATIK with the explicit aim of producing a compelling set of agreed upon priorities, goals, and actions
3. **Pulling change through the system:** maintaining momentum into the future, ensuring that progress will continue to be both visible and meaningful

This structure can be applied regardless of whether your aim is to build a stand-alone kanban system, to introduce the Kanban Method for the first time, or to reinvigorate fresh cycles of change. You can even use it retrospectively, helping you to think constructively about an implementation that needs a stronger connection with its host organization.

I hope to show that there is no contradiction between introducing Kanban impactfully and remaining true to its humane ethos. Toward the end of this chapter we review the role that the values can play in motivating an implementation.

## Planning the Engagement

When executing any facilitated process, it really pays to prepare properly ahead of time. As facilitator, these are the kinds of questions you need to ask yourself when planning to use STATIK:

1. Understand sources of dissatisfaction (Chapter 18).
  - ◆ Do you have at least a rough idea of the scope of the exercise? Does it have (or need) sponsorship?
  - ◆ Who should participate? Who will represent the people who work inside the presumed system boundary? Who will represent the system's customers? Does the wider organization need representation also?
  - ◆ What tools will you use to solicit and organize a good range of responses (see Chapter 17)?
2. Analyze demand and capability (Chapter 19).
  - ◆ Are you ready to help capture the “what, to whom, and why”?
  - ◆ Will you ask for data in advance (and perform some initial quantitative analysis yourself, perhaps), or will you wait to see what other participants want to do? What support can you provide?
  - ◆ What do you already know about how work arrives? Will you have the people you will need to explore this properly?
3. Model workflow (Chapter 20).
  - ◆ What is your preferred approach (sketch, top down, bottom up)?
  - ◆ Are you armed with searching questions for reviewing the output?
4. Discover classes of service (Chapter 21).
  - ◆ Will you get more traction approaching classes of service as an internal tool for organizing and scheduling work, or as a way to explore customer expectations? How will you bring the two aspects together?
  - ◆ At which organizational levels will this exercise be able to influence the overall mix of work in the system?
5. Design kanban systems (Chapter 22).
  - ◆ How will you introduce the concepts and share what has worked elsewhere? (This question answers itself when the workshop is also a training class.)
  - ◆ How much time will you want to spend refining designs before allowing them to be tried in the field?
  - ◆ Physical or electronic? What limitations (physical, geography, organization, privacy and security, or feature-wise) will need to be accommodated?

Lastly, don't neglect mundane things like the choice and availability of venues, participants' schedules, stationery, and equipment. These things matter even when you don't plan to hold a big workshop.

## Shaping the Agenda: The Three P's

We've planned the details; now we need to step back and think about how the exercise is framed. Positioning, purpose, and priority give some high-level shape to the engagement that potential attendees can respond to.

### Positioning

How you choose to engage with the organization and its people will depend both on context and on your own preferences. Again, it pays to put some thought into this. You need to think about how you position the Kanban Method with respect to the needs of the organization (to the extent that you are aware of them).

To get you started, here is a positioning based on some familiar elements:

- ◆ The Kanban Method has been described as the humane, *start with what you do now* approach to change.
- ◆ We will briefly explore its principles, practices, and values, thinking about how they apply in our situation.
- ◆ We will take a look at *what we do now*—warts and all—using what is known as the *Kanban Lens*. This encourages us to recognize that:
  - ◆ What we do—our flavor of *creative knowledge work*—is service oriented.
  - ◆ Service delivery involves workflow.
  - ◆ Workflow involves a series of knowledge-discovery activities.
- ◆ Through a series of exercises, we will do the following:
  - ◆ Map our knowledge-discovery workflow.
  - ◆ Pay attention to how and why work arrives.
  - ◆ Equip ourselves to track work as it flows across and between services.
- ◆ To ensure this exercise's success, we will take time to:
  - ◆ Agree on the scope and purpose of the system under review.

- ♦ Identify sources of dissatisfaction, which we will do from multiple perspectives.
- ♦ Prioritize actions that begin to address those dissatisfactions and better align the design and operation of the system to its purpose.

## Purpose

Do you remember STATIK-0 from Chapter 16? Plan to incorporate its step 0:

0. Understand the purpose of the system.

It might seem that the obvious place to address this is before step 1 (Understand sources of dissatisfaction), but I recommend that you dwell on the “what, to whom, and why” of step 2 (Analyze demand and capability) and use that as the springboard for exploring how effectively the system serves its purpose.

This sets you up nicely for a discussion on *fitness for purpose* and *fitness criteria*. This should be much more than just a philosophical aside; identifying gaps and measures of success will help focus the subsequent design activities and give additional impetus to the rollout.

## Priorities

In a similar vein, I have found it very powerful to prioritize the values and identify a top three or four around which a compelling call to action can be built. I do this through the rather unimaginatively named “Kanban values exercise,” materials for which I have published under an open-source license at SlideShare.<sup>64</sup>

Participants receive a list of the nine values; they map them first to the Foundational Principles and then to the Core Practices. Experience has shown that this order works best, and it’s much easier if the principles and practices are taken in reverse order.

I like to explain that there are no wrong answers—but there is a canonical answer! If you follow my recommended mapping and work

---

64. You can download a PDF from <http://www.slideshare.net/asplake/kanban-values-exercise> and I’m happy to provide the source PowerPoint file on request. Both are licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

bottom-to-top, you'll be looking for **leadership**, **respect**, **agreement**, and **understanding** for the principles, then **collaboration**, **transparency** (twice), **customer focus** and **flow** (both for CP4), **balance**, and (for a third time) **transparency** for the practices. Doing the principles first allows four values to be crossed off before proceeding to the trickier set.

We then use two rounds of dot voting, allowing three votes per person per round, to identify

- ◆ The values that—for whatever reason—resonate with the most participants at a personal level
- ◆ The values that should be emphasized in the implementation initially

Discuss the results of each round. Differences between the two can be striking; in your role as facilitator, take care not to jump prematurely to negative conclusions if this is the case (I have made that mistake).

Finally, take the top three or four values identified in the second voting round and try to construct some kind of narrative around them. The Kanban agendas can be useful guides here, but the results should be specific to the context and “owned” by the group.

Groups of between five and eight people work best. If you have more than eight people, you can consolidate multiple outputs in your debrief.

Between the two rounds of voting you can host a Kanban Knowsy group game (Chapter 17). Pitch this as a fun way to find out how well people have been listening to each other as well as to explore how well they are aligned. You can feed results of the second round into Kanban Knowsy’s “Discover Play” for comparison with the three predefined agendas.

## Pulling Change Through the System

You’ve done the planning and the framing; you’ve held your workshop or done the rounds; priorities for change have been agreed upon; teams are settling into their new kanban systems. How do you ensure that it doesn’t stop there?

We are fortunate now to have a good number of well-documented Kanban implementations of significant size out of which some great advice has been abstracted—see, for example, *The Kanban Kick Start Field Guide* by Christophe Achouiantz and Johan Nordin, Yuval Yeret’s *Pull-based*

*Change*, and Jeff Anderson's *The Lean Change Method* (as mentioned in Chapter 17). These and countless smaller implementations agree on the importance of maintaining an auxiliary pull system in parallel with the main delivery system. These allow small increments of change to be pulled from some kind of backlog (perhaps represented on a *change canvas* or *story map*) and managed visually through their implementation.

There are some interesting differences in emphasis among authors that should be noted. Some like to use the *Kanban Depth Assessment* tool (or similar tool) to prioritize practices that should be implemented or refined; others focus more on dissatisfactions or problems, using these as the driver for change.

The former, practice-driven approach makes good sense when the goal is to achieve a rollout in limited time. For two reasons, though, I lean toward the latter, more problem-driven system: I'm leery of changes that might be seen to be implemented for their own sake rather than to address pre-existing dissatisfactions, and a problem management system is just too useful to be ignored.

## Identifying Increments of Change

The nine values represent practices and describe benefits; by implication we can use them to organize the problems they address also (so the apparent dichotomy between practice-driven and problem-driven change needn't be a big deal after all). Let's take one final pass through them and identify key features of effective Kanban implementations.

You can use them to assess where you are, and to prioritize increments of change that make sense in your situation. As you go through this section, give your system a score of 1 through 4 on each numbered feature according to this scale:

1. Our system exhibits this aspect barely, if at all.
2. Our system is somewhat capable of exhibiting this aspect.
3. Our system exhibits this aspect convincingly, for the most part.
4. Our system departs from this only very exceptionally; we manage the consequence when it does so.

## Transparency

1. Work items are organized visually by type, state (in some knowledge-discovery activity, waiting in a queue, or some other state), parallel work stream, and class of service.
2. It is clear which items are blocked, and for what reason.
3. To the extent that it matters, it is clear who is working on what.
4. Explicit policies capture shared expectations on work item selection, quality criteria, and so on.
5. The progress of the work and the overall effectiveness of the system are subject to review at a range of cadences, from at least daily (at the standup meeting, for example) up to quarterly and longer.
6. Attention is paid to how progress, demand, and capability are reported externally, both to customers and to the wider organization.
7. Metrics have a clear relationship to the system's purpose.

Even before process changes have been implemented, the introduction of visual management tends to deliver an immediate benefit by making visible the need to make decisions (big decisions, sometimes). The other aspects of **transparency** identified here may take more time to bed down; but they are no less important with regard to the organizational impact they can bring.

## Balance

1. Work-in-progress (WIP) is limited such that no individual, activity or work stream is overburdened or is consuming a bigger share of the available effort or shared resources than is appropriate.
2. Work is pulled into and across the system only when capacity is available.
3. WIP limits apply to all work started but not completed; this includes work waiting between activities and between services.
4. The system comfortably accommodates a variety of schedule risk profiles (distinguishing, for example, between date-driven and urgency-driven work) and classes of service.
5. In allocating capacity between competing sources of demand, consideration is given to the needs of all stakeholders and to the overall capability of the system over a broad range of timespans.

Inexperienced practitioners of Kanban seem to worry a lot about the change management aspects of **balance**. I offer two pieces of advice:

- ◆ Remember that to reduce WIP, more work must be finished than is started. If it's not obvious to people what should be completed first, focus on that issue first. Organize the work visually and with policies, and ensure that it will stay organized through effective feedback loops.
- ◆ Remember that the level of WIP is both a lever and a symptom. Visualize it; bring its root causes to the surface; expect it to reduce as the process improves.

### Collaboration

1. Improvements are framed and structured as experiments and managed visually.
2. Other bodies of knowledge are used as models for improvement (ways of looking at systems, ways to structure change, technical and management practices, and so on).
3. Collaboration is embraced as a source of performance, a driver of improvement, and an antidote to system-generated frustration.
4. The system is open to change from inside (it is self-organizing) as it pursues fitness for purpose.

Remember:

- ◆ **Collaboration** isn't just "being nice"; neither is it limited to problem solving.
- ◆ A good response to disappointment is to consider the role collaboration might have played in preempting it.

### Customer Focus

1. The delivery workflow is understood to be a process of knowledge discovery, in which needs, possibilities, and capabilities are explored.
2. Upstream of a defined commitment point, work items are managed as options.
3. Downstream of delivery, work items continue to be managed until their utility in the hands of the customer has been validated.

From experience, that last point appears to be crucial. Paradoxically, validation takes place at the end of the delivery process, and yet it's the aspect most likely to bring about collaborative **customer focus** right through the process. At the Lean Startup extreme, it is the process's engine.

## Flow

1. Work items are sized and selected to achieve a strong and reliable flow of value.
2. Batches are sized and releases scheduled to maximize overall economic outcomes (not just to minimize delivery costs).
3. Work items of exceptional value or risk are managed appropriately.
4. The system reliably delivers non-exceptional work items with appropriate predictability.
5. Measured end-to-end, time spent in active knowledge discovery dominates time lost to delays (queuing, multi-tasking, blocking) and other kinds of work.
6. Dependencies between work items and on other services are identified and visualized in good time.
7. Work items can be scheduled for release independently of their commitment into the system.

We set high standards for **flow**. Implementations are unlikely to demonstrate many of these aspects very convincingly until they have first focused their improvement efforts on sources of unpredictability and delay and given some serious thought to the true economics of their work.

## Leadership and the Leadership Disciplines

1. Leadership is open to all; acts of leadership that bring about change are especially worthy of celebration.
2. There is a shared and ongoing commitment to change, based on an evolving understanding of what we do now and its alignment to purpose from the perspective of all its stakeholders.
3. Evidence of the need for change is kept close to the surface.
4. Change is safe; its downside risks are identified and mitigated (including the risks of changing too slowly or not at all).
5. The potential benefits of change (upside risks) are watched for and nurtured.

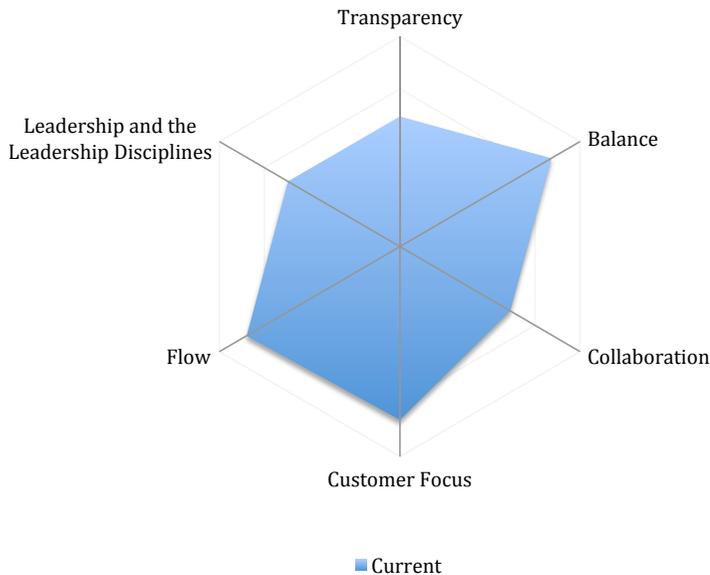
6. Change is implemented through agreement; the practices of change and the capability to change are themselves focuses for improvement.
7. Respect is always a given; at times of change, people’s attachment to their current roles, organization, and practices is never underestimated.

In your self-assessment, do not exclude the management of changes that originate from outside the system—from senior management or HR, for example. Dealing with these with **understanding, agreement, and respect** has the potential to demonstrate **leadership** to other parts of the organization.

## Visualizing change

### Visualizing the Assessment

In Figure 23.1, the geometric mean<sup>65</sup> of the scores in each category are visualized by plotting them in a radar chart.<sup>66</sup> This gives an alternative, values-based realization of the Kanban Depth Assessment tool, one that doesn’t stop at Kanban’s practices.

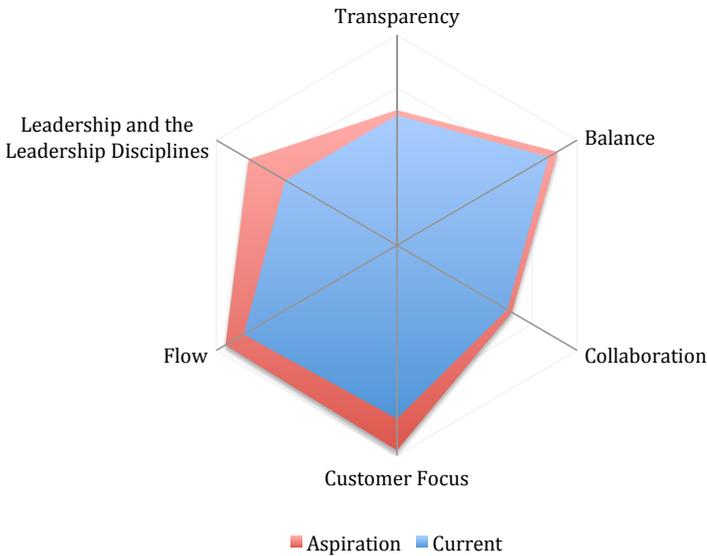


**Figure 23.1** Kanban Depth, by value

65. Relative to the simple arithmetic mean, the geometric mean amplifies the effect of the weaker scores.

66. “Radar chart” is what Excel calls them; you may know them as spider charts or Kiviati diagrams.

It can be useful to visualize historical progress or the desired trajectory also. My friend Ruben Olsen has conducted multiple assessments for the same teams and charted the results over time. Alternatively, you can answer the assessment questions aspirationally, describing where you hope to be in (say) three months' time. Figure 23.2 shows how this approach might be visualized.



**Figure 23.2** Kanban Depth, with trajectory

### Managing Change Visually

The challenge is to see some concrete actions brought to fruition in such a way that some momentum is generated. This can feel very much like a delivery process, and indeed it is amendable to being managed with similar tools.

It is unusual to find a significant Kanban implementation these days that doesn't maintain some kind of visual management system in parallel with its main delivery system. These auxiliary systems are devoted to change, problems, out-of-the-ordinary dependencies, and so on.

I co-developed the design of the Problem Board in Figure 23.3 with Kevin Murray of Valtech in 2012. Together and separately we have used several variants of it since.

Problem	Being Sorted	Sorted 😊	
	Daily		Closed
	Weekly		

**Figure 23.3** The “Problem Board”

We operate this board as follows:

- ◆ Anyone is free to add new problems to the input column on the left.
- ◆ After daily triage and ownership assignment, in-progress problems move vertically between the daily and weekly areas under “Being Sorted” according to the amount of time we wish to devote to discussing them.
- ◆ Some time after they have been deemed to be “Sorted” and we are sure that they will not resurface, decisions have been logged, and so on, problems move to “Closed.”

A more change-focused design is the one shown in Figure 23.4, from Jeff Anderson; it complements his *Lean Change Canvas* (Chapter 17), but could easily be used independently.

I like Jeff’s design very much. Its first two columns emphasize **agreement**; the last two emphasize validation (and by extension, **customer focus**). Separating qualitative validation from quantitative verification seems quite smart, too; typically, teams will be happy to confirm behavior changes long before it is possible to confirm any significant performance improvement.

Agree on Urgency	Negotiate the Change	Validate Adoption	Verify Performance

**Figure 23.4** A kanban system for “Validated Change”

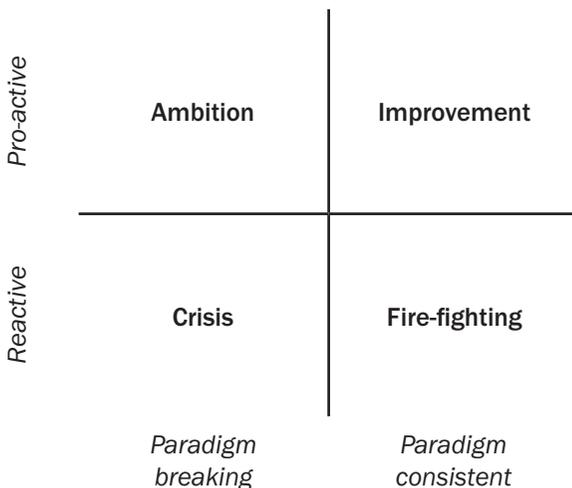
## Not All Change Is Alike

Way back in 2010 or 2011 (which seems aeons ago in Kanban terms) my friend and long-time collaborator Patrick Steyaert provoked me into “*not all work is alike*,” a phrase I have used many times since, including in this book.

With “*not all change is alike*,” Patrick reminds us of another useful parallel between managing customer deliveries on the one hand and system changes on the other. Referencing Thomas Kuhn’s book *The Structure of Scientific Revolutions*,<sup>67</sup> Patrick identifies two axes by which change can be analyzed:

1. Does the change involve reconstructing organizational commitments as it challenges prevailing thinking (revolutionary, *paradigm breaking* change), or does it build on what is there already (cumulative, *paradigm consistent* change)?
2. Is the change imposed from outside (*reactive*) or initiated internally (*proactive*)?

Those two axes suggest a quadrant, shown in Figure 23.5, which Patrick has labelled helpfully, too.



**Figure 23.5** Not all change is alike.

67. See Patrick’s post <http://lean-adaptive.com/2014/06/03/not-all-change-is-alike/>; his reference is to Thomas Kuhn’s *The Structure of Scientific Revolutions* (University of Chicago Press, 50th anniversary ed., 2012)

A few observations:

- ◆ Kanban is not about fire-fighting or managing existential crises, but if you find yourself needing to react in such a situation, it is possible that Kanban will help.
- ◆ Introducing Kanban is usually a proactive change, and it is designed to generate further proactive change.
- ◆ Whether Kanban as a whole, its individual principles, practices, and techniques, or the changes it provokes are paradigm consistent or paradigm breaking depends greatly on where you are now. For example, your organization may already be operating pull systems, or it may find the idea alien.
- ◆ In the terms of Patrick's quadrant, Kanban is not only about improvement; it is (very much) about ambition.

Neither does Kanban exclude radical change. What it asks is that you understand where you are now before you move forward on the basis of agreement and respect. If you can't or won't do that, then you are—at least as far as Kanban is concerned—on your own.

## Closing Thoughts

It seems fitting that I come to the end of the final chapter celebrating other people's work. This is very much a collaborative effort, the work of a community that is open to ideas, embraces change, and cultivates leadership in others.

I'm very glad also that I could finish the book as it started, with values. There were quite a few skeptics when I first laid out the nine; not unreasonably, they worried that it wasn't the method's place (or mine) to dictate the values of their organizations. Some warned me that values are fragile things—touch them, and they disappear!

Used carefully as proxies for practices and benefits, and as tools for organizing thoughts, stories, priorities, and so on, values have proved to be non-threatening, usefully thought provoking, and actually rather robust. To all who recognized these possibilities early on, and to you for sharing in them now, I thank you.

## Further Reading

- Achouiantz, Christophe and Johan Nordin. 2014. *The Kanban Kick-start Field Guide: Create the Capability to Evolve*. <http://leanagileprojects.blogspot.co.uk/2013/11/the-kanban-kick-start-field-guide-v11.html>
- Anderson, Jeff. 2013. *The Lean Change Method: Managing Agile Transformation Using Kanban, Kotter, and Lean Startup Thinking*. <https://leanpub.com/leanchangemethod>
- Yeret, Yuval. 2013–14. *Pull-based Change Management*. <http://yuvalyeret.com/2014/05/27/pull-based-change-management/>

## ❖ ABOUT THE AUTHOR ❖

**Mike Burrows** is UK Director and Principal Consultant at David J Anderson and Associates ([djaa.com](http://djaa.com)). In a career spanning the aerospace, banking, energy, and government sectors, Mike has been an IT director, global development manager, and software developer. He sits on the management board of Lean Kanban University (LKU), with whom he is an Accredited Kanban Trainer (AKT) and Kanban Coaching Professional (KCP). He speaks at Kanban-related events around the world, blogs at [positiveincline.com](http://positiveincline.com) and tweets as [@asplake](https://twitter.com/asplake) and [@KanbanInside](https://twitter.com/KanbanInside).