

The background of the cover is a photograph of a wooden walkway or bridge. On the left, there are trees with bright yellow autumn leaves. On the right, the walkway leads towards a body of water under a hazy sky. The image is partially overlaid by large, curved purple shapes that frame the text.

Using Kanban in IT Operations

Dominica DeGrandis
and Kaimar Karu

AXELOS.com



Guidance Paper

September 2016

Contents

The point in brief	2
What is broken in IT Operations	3
Concepts defined	5
How to do Kanban for IT Operations	10
Kanban and IT Service Management	37
Summary	44
Glossary	45
About the authors	48
Acknowledgements	49

Trade marks and Statements

AXELOS, the AXELOS logo, the AXELOS swirl logo, ITIL®, MoP®, M_o_R®, MoV®, MSP®, P3M3®, P3O®, PRINCE2® and PRINCE2 Agile® are registered trade marks of AXELOS Limited. RESILIA is a trade mark of AXELOS Limited.

© Copyright AXELOS Limited 2016 and Leankit Inc.

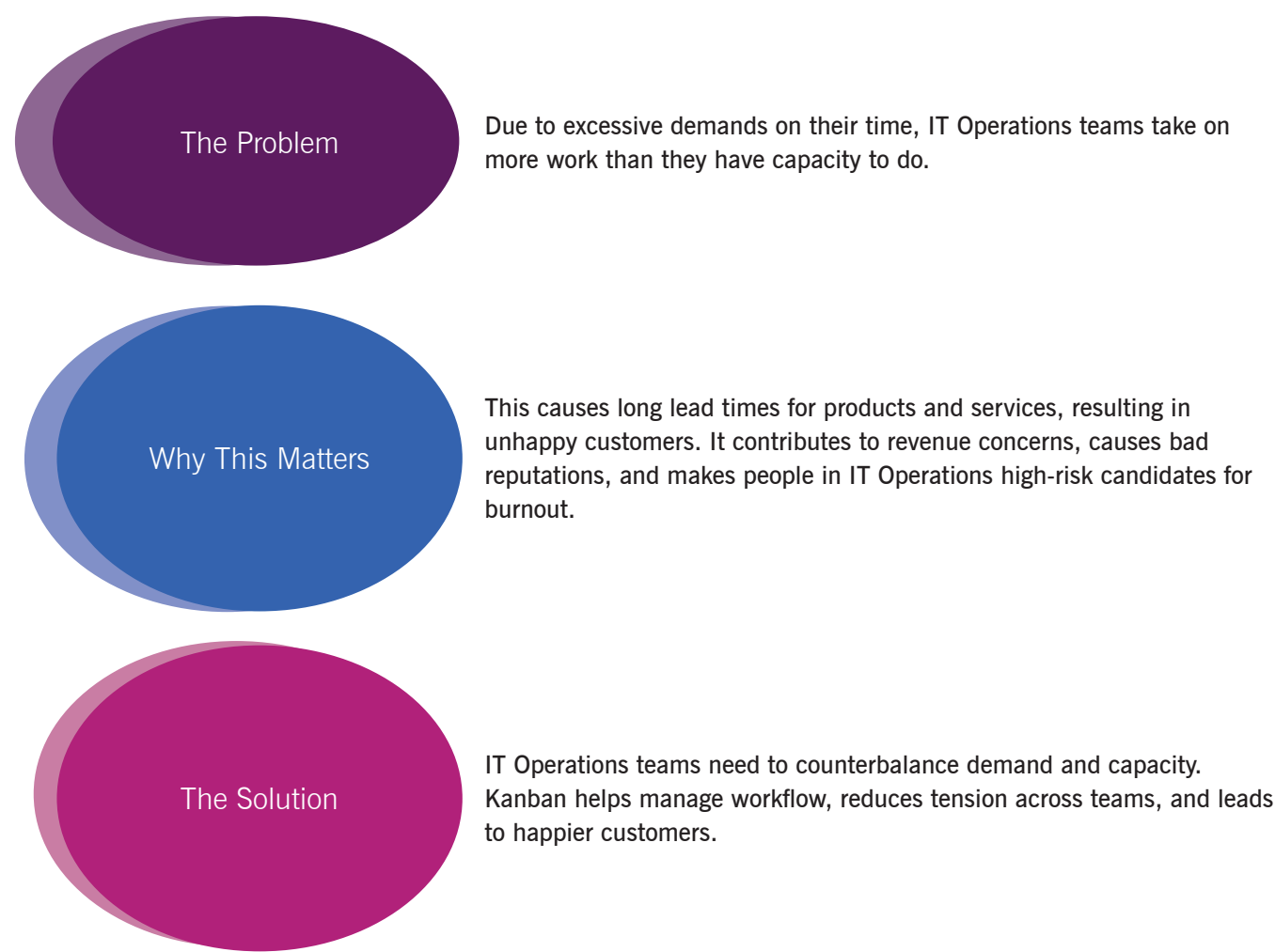
Reuse of any content in this case study is permitted solely in accordance with the permission terms at <https://www.axelos.com/policies/legal/permittted-use-of-white-papers-and-case-studies>.

A copy of these terms can be provided on application to AXELOS at Licensing@AXELOS.com.

Our Guidance Paper series should be taken as advice and no liability is accepted for any loss resulting from use of or reliance on its content. While every effort is made to ensure the accuracy and reliability of the information, AXELOS cannot accept responsibility for errors, omissions or inaccuracies. Content, diagrams, logos, and jackets are correct at time of going to press but may be subject to change without notice.

Figures 1,5,6,8,10,11 and 12 are ©LeanKit Inc. Figures 2,3,4,7 & 9 are ©AXELOS Ltd.

1 The Point in Brief



2 What's broken in IT Operations

Imagine working as fast, as smart, and as hard as you can, only to land yourself with a reputation across your company for poor performance. That's the plight of many IT Operations teams. A bad reputation is a consequence of the length of time the team takes to process requests. Requests come in from all corners of the enterprise faster than they can be handled. There can be requests to help build and deliver new features, to provide new tools, requests to roll out a new cloud-computing platform, etc.

But sometimes the team can't get to any of those requests because they are too busy preventing the production environment from falling over. Fixing an erroneous security certificate, for example, so their customers can authenticate their website before transmitting credit card information, will always be a higher priority than delivering an enhanced feature for Sales. When it comes to conflicting priorities, fixing critical production issues - such as Distributed Denial of Service (DDoS) attacks, outages, and security breaches - wins every time.

Ensuring production systems are running reliably is especially challenging when the work competes with revenue-generating requests, e.g. new feature requests. It's not uncommon to see IT Operations teams juggling a multitude of work types at any one time: implementation of new features; enhancements to existing features; fixing production issues; addressing technical debt; fulfilling service requests; performing maintenance work; responding to security and compliance questions. It's no wonder that teams feel they are stretched too thin, especially when unplanned work is added to the mix. It's this situation that contributes to the team's poor reputation.

It's hard to say "no" to every request. No one likes being *that* IT Operations person who tells the boss, "Hey, I know that Marketing has already announced this feature will be available next month, but we can't make that timeline." That's not going to go over so well.

But here's the problem: it's difficult to quantify just how much capacity teams really have. We neglect to factor in delays that occur due to interruptions and context switching. And it's hard to plan for unplanned work. So much so, that there's a higher probability of a magnitude 9.0 earthquake happening in the US Pacific Northwest within the next 49 years than there is of many software projects hitting their planned release date.

What we are good at, however – because we get it all the time – is receiving negative customer feedback. When we talk about customers here, we mean both external customers of the product or service, and internal customers within our company. Either way, you know you have a problem when your customers keep asking if the thing they requested is ready. When the answer is consistently "no," take a hard look at the workload your team is trying to cope with.

People take on more work than they have capacity to do. People tend to start work instead of finishing work. It's refreshing to start something new. And the sooner something is started, the sooner it's finished – right? Wrong! This may sound logical but, in reality, juggling too many requests brings unanticipated delays.

The term *value stream* refers to the activities, from beginning to end, that need to happen for a product or service to provide business value. This is where Lean and Kanban come to the rescue. They help overburdened teams to prioritize and complete their work sooner, and therefore deliver value to their customers sooner.

The need to balance production environment stability with product and service delivery pressurizes IT Operations teams into taking on more work than their capacity allows.

3 Concepts defined

Before we begin to describe how Kanban can be utilized for IT Operations, we need to provide a little background about IT Operations, and discuss the problem that Kanban can help to address.

In today's world of IT Operations, managing the tension between shifting business requirements and fallible computer platforms is a challenging goal. The tension endured by many teams who attempt to counterbalance project requests while keeping the production environment reliable is simply too high. Like tuning a guitar, the correct amount of string tension is needed in order for the instrument to play well. If there's too much tension, the guitar strings snap. Today, many IT Operations professionals are close to snapping, or are showing worrying signs of burnout.

Easing the tension caused by conflicting priorities permits teams to balance their work requests. This allows the work to flow more smoothly. Using Kanban enables you to discover the right amount of tension needed to optimize a team's workflow.

3.1 Kanban

Kanban is Japanese for a *visual signal* or card. Toyota line-workers used a kanban (i.e., an actual card) to represent steps in their manufacturing process. The system's highly visual nature encouraged teams to communicate smarter on what work needed to be done and when. It also has standardized queues and refined processes, which helped to reduce waste and maximize value.

Born in the era of mid-century automobile manufacturing, Kanban became the just-in-time inventory control system for the Lean manufacturing model. Lean is defined here as a Socratic business philosophy which can help improve workflow efficiency by utilizing a just-in-time pull system and visual workflow management. At its heart, Kanban is a visual pull system, working with Lean to facilitate a harmonious workflow.

Kanban revolutionized the way Toyota controlled their supply chain, which allowed them to keep up with the ever-increasing demand for vehicles.

Around the mid-2000's, a community formed around the leadership of David J. Anderson, Jim Benson, Corey Ladas, and others, who were exploring ways that Kanban could be used to benefit knowledge work. Their thinking was influenced not only by the Toyota Production System but also by the work of W. Edwards Deming, Eliyahu Goldratt, Donald Reinertsen, and others.

3.2 DevOps and Theory of Constraints

The DevOps movement came about as a means to reduce the tension between agile software development and stressed IT Operations. The term was coined in Belgium during a data centre migration project in 2009. The term DevOps sprang from an attempt to apply agile techniques to IT Operations activities and align IT Operations teams with the work of software development teams. With IT Operations receiving increasingly frequent deliveries from their development teams, the pressure to keep up necessitated faster and more efficient practices.

For IT Operations specialists, whose previous focus was on maintaining a fragile infrastructure just to keep the lights on, the accelerated pace of delivery from software development required a shift in thinking. Although the change aligned with what IT Operations had wanted to do for a long time, their contemporaneous work practices and tools couldn't support the expedited delivery schedule.

There's a famous story in Eliyahu Goldratt's book, *'The Goal'*, where 15 boy scouts hike 10 miles to an overnight camp. They discover along the way that the troop can only hike as fast as Herbie, the slowest-moving boy scout. The troop leader puts Herbie at the front of the pack to regulate the speed of the faster scouts, but Herbie can only hike one mile per hour, and they have four more miles to go to reach camp. With night drawing in, they realise they are destined to arrive after dark.

Humans are an optimistic bunch.
We often presume a task will take
less time to complete than it actually
does.

To apply this analogy to IT Operations, the organization can only move as fast as the slowest-moving piece of the system. No disrespect to Herbie, but this is known as a **constraint**. (More about Herbie later in the paper).

Kanban can assist you to clear constraints because it helps to identify bottlenecks. When using Kanban, people are less likely to overcommit or slow down the flow, as they pull work into their queue only when they have capacity to do more work, based on real availability, rather than an attempt to please others.

In addition to constraints in the system, IT Operations teams face the challenge of adjusting their work practices to the vagaries of the cloud infrastructure. The inefficiencies of the past were no longer liveable with. With a need to address the growing tension in the software build and deploy space, DevOps caught on swiftly as software delivery professionals assembled a community to support their collective endeavours.

As with Lean, there are many levels of abstraction when defining DevOps. There's a greater likelihood of an article defining DevOps by what it isn't, rather than by what it is. Many of the explanations see DevOps through a Lean lens of improving business value and regenerating culture.

DevOps is the result of applying Lean principles to the technology value stream. Like Lean, DevOps embraces a Socratic philosophy void of autocratic, command and control tendencies. Central to its philosophy are changes in the way progress is measured, feedback obtained, culture changed, and deployment pipelines automated, which work together to improve the health of the organization, which in turn improves business outcomes.

The secret to optimizing the flow of work across the value stream is to focus on finishing requests before starting new ones.

3.3 IT Service Management and ITIL®

People familiar with ITIL may recognize some of the concepts described above. ITIL co-opts good practice to leverage IT capabilities in order to deliver value to the organisation and its customers through well-managed services. Covering the whole service lifecycle from strategy through design and transition to operation, it takes a holistic end-to-end view of what is required to deliver and support a service, and how the value is realized for the customer and the service provider. Being non-prescriptive, ITIL leaves it to the organisation itself to decide which detailed philosophies and methodologies are most relevant and applicable.

The initial focus of ITIL, when the guidance was first introduced in 1989, was on infrastructure management and IT Operations. While the field of IT Service Management (ITSM) and the framework itself have evolved significantly since then, IT Operations remains one of the most popular areas for ITIL adoption, with incident management, problem management, change management, and request fulfilment processes, plus the service desk function, frequently among the first to be adopted.

ITIL recommends keeping track of both planned and unplanned work to ensure better visibility and manageability, and most organizations have solved this by utilizing specialized ITSM software platforms and introducing queues for what are often referred to as *tickets*. Items are added to these queues either manually (e.g. by service desk staff or the user themselves for service requests) or automatically (e.g. by a monitoring tool for incidents). Depending on the specific implementation of the software, sometimes the ticket is automatically assigned to a team or a person, and sometimes it remains in a backlog or shared pool until someone pulls it into one of the work queues.

Organizations generally have differing timelines for resolving incidents or completing changes for specific customers, which are usually defined by their Service Level Agreements (SLAs). This frequently means that IT Operations professionals are working on more than one ticket or task at a time, and their day is a mix of planned and unplanned work.

Kanban can be leveraged to manage work in any category, from planned maintenance to service requests unplanned work resulting from incident. Even tasks related to changes, which often require the contribution of more than one team and can be a major

Socrates is best remembered for his method of teaching by asking increasingly difficult questions, the so-called Socratic method. This generally involves the use of Socratic induction, a way of gradually arriving at generalizations through a process of questions and answers, and Socratic irony, in which the teacher pretends ignorance while questioning his students skillfully to make them aware of their errors in understanding.

Merriam-Webster:
<http://www.merriam-webster.com/dictionary/Socratic>

to
an

source of stress, can be visualized and managed using Kanban, helping to reduce the amount of *dark matter* type work, which is often invisible but tends to consume all working hours.

Chances are you are using some variety of work management tool already and, as an IT Operations professional, you are keeping your eye on a specific work queue on a daily basis. But does that queue contain all the work you need to complete? Does it contain the work you know *needs* to be addressed but which you invariably postpone because higher-priority tasks exhaust all your available capacity? Kanban can bring this work together and, over time, create opportunities to address the backburner work which will come back to haunt you if you ignore it.

We're going to discuss how Kanban can be used for knowledge work – with a focus on how to execute Kanban in the IT Operations part of ITSM – so you can avoid the high-risk tension that causes IT engineers to snap and burn out and improve effectiveness and efficiency.

At its heart,
Kanban is a visual pull
system, working with Lean
to facilitate a harmonious
workflow

4 How to do Kanban for IT Operations

Now we've established the reasons why Kanban can benefit IT Operations teams, it's time to get started with the 'how'.

Designing a basic Kanban system begins with four major tenets:

- 1. Make the work visible
- 2. Limit the work in progress
- 3. Measure the progress of the work
- 4. Communicate the work state

We'll cover each of these steps in the following pages.

4.1 Make the work visible

Twenty percent of the 100 billion neurons in the brain are devoted to analysing visual information.¹ Humans are experienced at processing information visually. We can't help it, we do it naturally. It makes sense that visualizing our work will enhance its progress. The better the visual, the greater value it brings. Author David McCandless² demonstrates that a visual, void of utility, is useless. If there's no relevance in relation to the workflow then the visual can't help but be boring. And if the visual is unappealing (like the spreadsheet in Figure 1), people will tend to zone out. However, visualizing workflow using a relevant, useful, and intuitive approach is compelling. That's the power of Kanban.

	C	F	G	L	Q	R	S	W
1	Card Title	Lane Title	Tags	Card Is Blocked	Card Type	Card Description	Card Priority	Last Activity
2	αντί να εμφανίζω την προσαρμογή	den éxouneta - to méllon wishlist	download-	FALSE	όλη εργασία	<>Regenerate όλα τα στιγμιότυπα ώστε	Κανονική	05/05 / 2015 1:41:54 PM
3	Ανάλυση Γνωστοποίηση Edition	den éxouneta - μελλοντικές εργασίες: wishlist		Αδύος	Βελτίωση App	<> Αν αυτό το κομμάτι είναι πορτοκαλί, <	Low	12/5/2015 8:14:26 πμ
4	Ρέννη Εκκαθάριση Σενάριο	den éxouneta - μελλοντικές εργασίες: wishlist		Αδύος	Εργαλεία & Wor	<> Το έργο είναι διδακτικό πλαίσιο εκπαιδευ	Low	4/6/2015 10:42:34
5	Επέκταση Disallowed Αρχείο - επέκτα	den éxouneta - μελλοντικές εργασίες: ". Wishlist		ψευδή	Βελτίωση App	<> Όταν ο χρήστης προσθέτει ".dat " σ	Κανονική	28/5/2015 3:45:39 μμ
6	Νέα Προσκόληση Κάροτα Χρώματα	den éxouneta - μελλοντικές εργασίες: wishlist		ψευδή	Νέο χαρακτηρισ	<> Ως επιπλέον βελτίωση, θα ήταν καλ	Κανονική	4/6/2015 11:15:51
7	Διαχειριστής ενημερώσεων το όνομα	den éxouneta - μελλοντικές εργασίες: wishlist		ψευδή	Εκδόττωμα	<> <a href = "http://screencast.com/l/ o	Low	05 / 28/2015 5:32:37 PM
8	Προσκή Χρήστη - Εφαρμογές & Συσκ	den éxouneta - μελλοντικές εργασίες: wishlist						6/2/2015 10:30:1
9	Εφαρμογές & Συσκευές μου - den δεί	den éxouneta - μελλοντικές εργασίες: wishlist			Κανονική	App Βελτίωση		
10	Mass Επεξεργασία - Επεξεργασία Hμ	den éxouneta - μελλοντικές εργασίες: wishlist			<> http://screen6/12/2015 10:41:			6/12/2015 10:41:49
11	Σωρευκή Αναζήτηση - Τύπος Κάροτας	den éxouneta - μελλοντικές εργασίες: wishlist		ψευδή	Βελτίωση App	<> <a href="http://screencast.com/1kbtb	Κανονική	06 / 18/2015 10:41:31 AM
12	qñtpro den λειτουργεί όταν στα Πρώτ	den éxouneta - μελλοντικές εργασίες: wishlist		False	Βελτίωση App	<> Αυτό το χαρακτηριστικό λειτουργεί <	Κανονική	18/06/2015 10:41:24 AM
13	Lane - Μαζική Επεξεργασία - Ετικέτες	den éxouneta - μελλοντικές εργασίες: wishlist		ψευδή	Βελτίωση App	<> Δίνει μικτή αντίληψη ότι οι ετικέτες μ	Κανονική	18/6/2015 2:26:28 μμ
14	Όταν η λωρίδα είναι γεμάτη αφήνουν	den éxouneta - μελλοντικές εργασίες: wishlist		ψευδή	Βελτίωση App		Κανονική	6/23/2015 11:01:43
15	Έγγραφο ΑΠΟΤΕΡΟ / CSS setup στο den	den éxouneta - μελλοντικές εργασίες: νέα απήματα		ψευδή	Εργαλεία & Wor	<u> <u> <u> μεταβλητών, κλπ. </i> <	Low	4/6/2015 10:42:33
16	Τεκμηρίωση: πώς να ρυθμίσετε κανόν	den éxouneta - μελλοντικές εργασίες: νέα απήματα		ψευδή	Εργαλεία & ροής	<> Προσθήκη στο README (bankit.an	Low	4/6/2015 10:42:34
17	Totango Συνομιλία με Shawn	den éxouneta - μελλοντικές εργασίες: νέα απήματα			Low	Έρευνα	ταμπλό	
18	Κοιτάζετε τη γραμμάττα για να ενσωμα	den éxouneta - μελλοντικές εργασίες: νέα απήματα", ψευδή			4/6/2015 11:15:3			6/4/2015 11:15:31
19	Καταγραφή Review - tableaux	den éxouneta - μελλοντικές εργασίες: νέα απήματα	tableaux	True	App Βελτίωση	<> μονοπάτια σφάλμα κρηκή και βεβα	Κανονική	6/4/2015 10:52:33
20	Συντήρηση Page για το Analytics	den éxouneta - μελλοντικές εργασίες: νέα απήματα	analytics	True	Νέο χαρακτηρισ	<> Αυτή θα είναι η έρευνα αρχική - θα	Κανονική	04/06/2015 11:15: 20 πμ
21	Growler Χρώματα	den éxouneta - μελλοντικές εργασίες: νέα απήματα		ψευδή	Νέο χαρακτηρισ	<> Τα χρώματα γεννήματος είναι ακόμα	Κανονική	4/6/2015 10:51:55
22	προσαρμοσμένες σελίδες ασφαλώνων	den éxouneta - μελλοντικές εργασίες: νέα απήματα	analytics	True	Νέο χαρακτηρισ	<> Πρέπει να ξέρω αν μπορούμε να π	Κανονική	
23	Analytics / μένα το καταληκτικό σημεί	den éxouneta - μελλοντικές εργασίες: νέα απήματα	analytics	True	Κίνδυνος / Τεύχ	<> Βλέπε συννημένο screenshot. Από	Low	06.04.2015 11 : 15: δέκα
24	In App γραμμή οδοπορήσεων για τις	den éxouneta - μελλοντικές εργασίες: νέα απήματα", ψευδή			<> Δείτε το με: 9/5/2015 0:36:30			
25	ZD # 52720: Πολλαπλή επιλογή + Del	den éxouneta - μελλοντικές εργασίες: νέα απήματα		ψευδή	App Βελτίωση	Andy Hoover (andy.hoover@bankit.com	Κανονική	
26	παλιά εικόνα στην καρτέλα όταν διαχ	den éxouneta - μελλοντικές εργασίες: νέα απήματα			Κανονική	Νέο χαρακτηριστικό γνώρισμα	Branding	

Figure 1 Spreadsheet

- 1. Ware, Colin. Information Visualization: Perception for Design. San Francisco: Morgan Kaufman, 2000.
- 2. "What Makes A Good Data Visualization? — Information Is Beautiful." Information Is Beautiful. Accessed June 17, 2016. <http://www.informationisbeautiful.net/visualizations/what-makes-a-good-data-visualization/>.

4.2 How to begin to use Kanban to visualize workflow

The first step when rolling out Kanban is to make your team’s work visible. Teams manage their work better when it is clearly laid out. We recommend starting at team level, rather than with programmes or portfolios, as the work that completes the strategic vision happens at team level. (We discuss programme and portfolio level boards in section 4.9).

Sometimes, the freedom to design your own Kanban board can be debilitating, especially if you’re accustomed to prescriptive methodologies. Kanban’s personalized approach can spark all kinds of questions

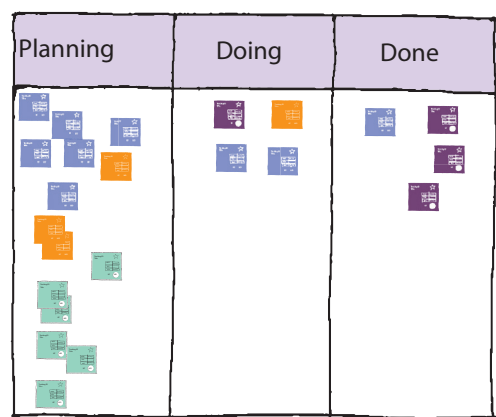


Figure 2 A basic Kanban board with three work states – planning, doing, and done

“Where do we start?
What are we supposed to show?
What about sizing work item types?
Do we need to put everything on the board?”

“Other than making work visible and revealing idle or blocked work, there is no right or wrong board design.”

Kanban starts with the reflection of your workflow, thereby showing your reality. You are only as constrained as the tools you use.

When designing your first board, it is simpler to start with a physical board, rather than an electronic tool. A physical board has less constraints, is easy to modify, and allows for greater trial and error through which you can discover what works for your team and what doesn't. The only materials you need are sticky notes, markers, and space for a board. It's easier to move to an electronic tool after your team has played around with a physical board design and has had time to reflect on aspects like the layout, how your work should flow, or when to pull in work. (We discuss electronic boards in section 4.10).

The following four-step approach is a simple way to start with Kanban:

- 1. Identify your team's work categories. For each category, choose a uniquely-coloured sticky note.
- 2. Define the work states (e.g. Planned, Doing, Done)
- 3. Identify each work item in your workload and jot it on a sticky note of the appropriate colour.
- 4. Place each work item sticky note on the board in their corresponding work states.

See section 4.3 for General Guidelines for Designing Kanban Boards.

All too often, people want to design their rows and columns on a board before they have determined their work categories. The problem is that not all work is the same. Different work types frequently have different process flows. For example, expedites aren't born in the backlog like other requests. They can be born in doing and, more often than not, they are created after a change has been carried out and something went wrong. Therefore, you need to understand the nature of your work before you are able to build a board that the work can flow across.

There is a chicken-and-egg situation that occurs when first designing a Kanban board. While it's tempting to start with a ruler and lay out the rows and columns at the very beginning, there is a benefit to categorizing work categories first, as it drives decisions for how different categories of work will be prioritized and processed.

A work state refers to where the work item is in the pipeline on its journey from start to finish. Each work item flows through several states on its way to completion. For example, if a work item in your workflow begins its life in backlog, moves from there to investigate, then to implement, to validate, and finally to done, then you have five work states: backlog, investigate, implement, validate, and done. We suggest starting with just three work states – Planning, Doing, and Done, as demonstrated in Figure 2, a basic Kanban board.

It is important to note you might require strict criteria to ensure that cards do not progress from one work state to the next before the work has been completed to the defined level of quality; e.g. you do not move from testing to ready to deploy unless the agreed quality criteria have been met.

When your team starts to design its own board, add a legend to help the eye instantly recognize the different categories of work.

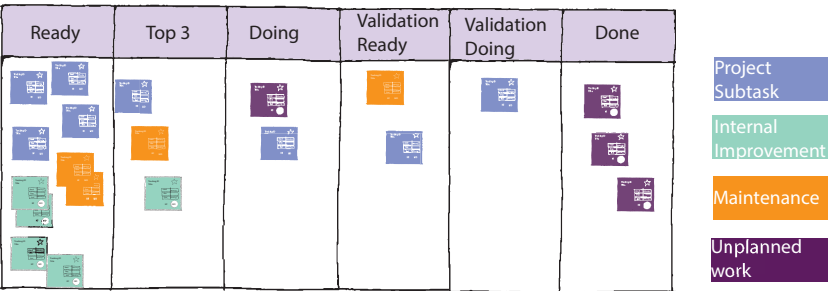


Figure 3 Example IT Operations team board structure showing work item category legend and work state columns

“
A glossary of terms can be found
at the back of the paper
”

The category legend in Figure 3 shows the following colour-coded work categories:

1. Project subtask (e.g. revenue-generating work, such as feature requests)
2. Internal improvement (e.g. deployment automation)
3. Maintenance (“keeping the lights on”)
4. Unplanned work (e.g. troubleshooting production issues)

You’ll discover not only how the team’s work moves through the system, but you’ll also gain insight for discussing prioritization, capacity, and unplanned work.

Prioritization

In the example in Figure 3, we have included a Top 3 state to indicate priority work items. Not all boards need a Top 3 column. If you include one, it will certainly spark conversations about priorities. The emphasis on prioritizing the most important work items before doing the work is significant because it indicates that the team has reached agreement on what work to commit to doing and what work not to do. Prioritizing is not always an easy task for teams. But pretending to have more capacity than we do doesn’t help deliver business value.

Capacity

Cards in the Validation Ready column queue up until there is capacity to do the validation. In our example, because the Validation Ready column shows two work items waiting for validation, it might be reasonable to ask whether validating work has the potential to be bottleneck in this scenario.

Unplanned Work

Because the Done column consists of just purple cards (unplanned work) in our example, we can hypothesize that other work item types are being sidelined by unplanned work (such as access requests and production issues/incidents). This is a common challenge in IT Operations and it causes frustration and disappointment for teams struggling to make time to work on their own internal improvements. (For example, note the lack of progress on the green cards for internal improvements). Add to that those troublesome blue project subtasks that project managers complain about, and you have a major source of tension for this team.

Calling attention to frustrations that prevent the team from getting their work done is an important initial step toward addressing those frustrations. The Estonian proverb, “The work will teach the doer” is a valid way to discover what to do next. We learn by observing and doing.

4.3 General Guidelines for Designing Kanban Boards

The initial team board design in Figure 3 is unique to the context of the problems being addressed by that specific team. The chances are your team’s board design will differ from every other design. It is good advice to avoid duplicating other boards.

Experience has shown us that the following guidelines are useful when designing a Kanban board:

- People who do the work should participate in the design of the board. A Socratic approach, where inquiry and discussion occurs between impacted individuals, provides the best chance of creating a board design that will be valued by those who use it. The board should not be created for others without their involvement.
- The board should accurately reflect how your team works. Basing your improvement plans upon a false design is useless. The goal of Kanban is not to have a board, but to identify ineffective and inefficient conditions.
- Design your board in such a way that idle and blocked work becomes self-evident.
- Keep the design simple and intuitive. Don’t over-engineer it, as that may bring confusion. You can always add details later on once you’ve used the board for some time and you’ve identified ways it might be improved.
- Limit the beginning and the end states (the first and last columns) of the board to the workflow within your team’s control. If you want to expand the scope, invite those impacted to contribute to the design.

Once you can see the different types of work and how much work is in which state, you can stand back and learn how your team works by observing the board.

Attribute	Purpose	Example
Vertical Column	Indicates the state the work is in	To do/prioritized/ doing/review/done
Horizontal Swimlane	Allows work categorization based on chosen criteria (e.g. work type or assigned team)	Planned work/ unplanned work or tightly coupled/loosely coupled work
Work Item Type	Categorizes different types of work	Planned project tasks, service requests, internal improvements, problem management, unplanned work
Policy	Explicitly identifies a rule	Definition of done
Tag	Highlights something special about a work item	Blocked work, dependency, child/ parent relationship

Table 1 Common attributes of Kanban board designs

Planning what work to do and when to do it consumes a significant amount of time before the doing of the work itself begins. Planning continues as thoughts and ideas creep in and evolve while figuring out how to do the work. While thinking time for planning what and when in technology work is relatively straightforward (once priorities are set), thinking time for figuring out how to actually do the work requires a special level of uninterrupted concentration. This is why limiting Work-in-Progress (WIP) is so crucial.

4.4 Limit the work in progress

It's easy to cast aside an aging request when a shiny new request arrives in the backlog. But the cost of starting new tasks before finishing the old tasks can be high. When too much work is in progress, negative issues occur: context switching increases, bottlenecks develop, dependencies rise, windows of opportunity slip, and holidays arrive. And, because progress is delayed, people pile more work on. This causes the work to take longer than it should and delays the delivery of value to the business.

An IT Operations team at a company we worked with had a workflow where the final step in the process (before Done) was Validate. There was no process at the time that allowed for timely feedback so, after they were delivered, work items stayed in Validate. The norm was to assume that everything was okay provided no one complained. While work sat idle in Validate, people pulled more work out of the backlog into Implement. The Validate queue grew and grew until it had close to 100 work items. Figure 4 shows what this looked like, with the bottleneck stacking up in the Validate queue.

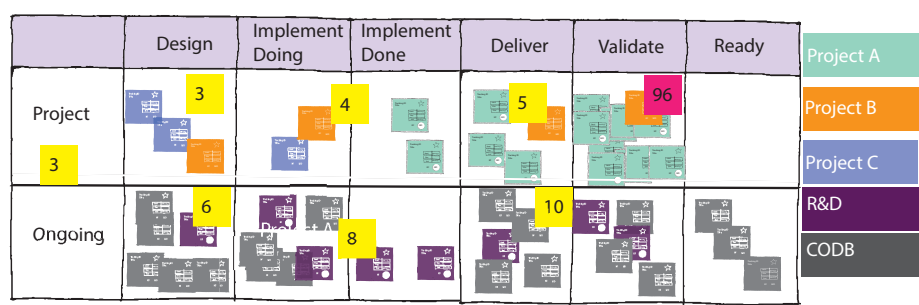


Figure 4: Kanban board with bottleneck in Validate queue

Multitasking is an
effective way to get
less done.

Many of the items delivered were indeed satisfactory, but some were not, which made the downstream customers unhappy. When they mentioned the work hadn't been delivered as they expected, the IT Operations team took a long time to respond because they were already on to the next task. As time grew between customer feedback and IT Operations response, the unhappy customers became very unhappy and eventually gave up trying to communicate. They complained amongst themselves and to other departments: "Ops never respond," "They're like a big black hole" and "What a worthless team." Watching the IT Operations team's reputation slide down the drain was depressing. When you try to do too many things at one time, you won't do anything well. Steve Uzzell phrased it nicely – "Multitasking is merely the opportunity to screw up more than one thing at a time."

Here's an exercise to try: Query your ticketing system (ITSM tool or similar) to reveal the work items that haven't been touched for a period of time (e.g., 60 days). Compare that outcome with the work that is getting finalised. Is the result discouraging? If yes, have a good, old-fashioned conversation with the team about finishing work before starting new work. Try doing it by ticket type so you can see which tickets goes smoothly and which get stuck on the way.

In an effective pull system, WIP-limits facilitate conversations that can shift the organization from an authority-focused must-do-all-the-things approach to a dialogue around what-is-the-best-thing-to-do-right-now, based on a collective agreement of the real business and economic goals and the team's capacity to do the work.

In addition to creating delays in the delivery of business value, too much work in progress masks two other major problems:

- busyness is mistaken for productivity
- fast feedback is thwarted.

When busyness is mistaken for productivity, everyone looks like they are busy, but tangible outcomes for delivering value to customers is lacking. Not a good place to be if you're serious about being successful. Too much work in progress equates to saying "yes" to too many things. When you have a tendency to take on more work than you have the capacity to do, setting and adhering to WIP-limits is a way to stop the pandemonium and to give yourself (and your people) permission to say, "Sorry; not today."

Busy people say
"yes" haphazardly.
Productive people say
"yes" deliberately.

When people have to wait too long before they receive feedback, the value of the feedback decreases. They move on to the next thing, and they no longer have the time to take the feedback on board. Consequently, they lose the opportunity to improve their work. The Lean philosophy is big on removing waste, but it's hard to see waste unless you have delivered something and received feedback on it. In the Kanban world, value trumps waste reduction, and fast feedback is a great way to provide value.

When people have to wait too long before they receive feedback, the value of the feedback decreases. They move on to the next thing, and they no longer have the time to take the feedback on board. Consequently, they lose the opportunity to improve their work. The Lean philosophy is big on removing waste, but it's hard to see waste unless you have delivered something and received feedback on it. In the Kanban world, value trumps waste reduction, and fast feedback is a great way to provide value.

Two elements can be relied upon to ensure the work gets done:

WIP Limits WIP limits create the necessary tension in the system to incentivise people to complete their work. If you can't start project B until project A is done, the pressure to finish project A influences behaviour. People will take action to avoid pulling in another work item until the current work item is completed.

Team pressure when your team stares at the same old work item day after day in the doing column on your Kanban board, you risk losing respect. Over time, your credibility is questioned. Are you spending your time doing what you should do? Or are you letting email derail you?

But what about expedites? Surely, Priority 1 incidents receive immediate attention and take priority over work that you've previously committed to finishing. This is true – production comes first! If your team is plagued by expedites, then make sure enough capacity is allocated to them. One way to do this is to investigate how many expedites have occurred over the last 60 days and how long they took to resolve. Query your work tracking system – a sample set of, say, eleven random expedites could be statistically sufficient to determine the probability of how many expedites will appear over the next 60 days. Adjust your WIP limits to accommodate the likelihood of these future expedites.

For example, if 30 expedites occurred over the last 60 days, and your Mean Time to Repair (MTTR) was 47 minutes, then you could consider reserving space for one expedite at all times, knowing that there is a 50%

probability that the expedite will be resolved in an hour.

However there's also a 50% probability that the expedite will take longer than an hour. Given the law of averages, using higher percentiles such as 75% to 85% (135 to 175 minutes, for example) may be a safer approach. If your organization is risk averse, you might consider reserving space for two expedites at any one time, given 85% of your expedites have a probability of being resolved in three hours or less.

This depends on several factors, such as the number of people on the team and their level of skill and specialization. If only one person on the team knows how to troubleshoot a particular toolset, and that toolset commonly has issues, then WIP limits may need to be reduced even further in order to avoid a serious bottleneck.

Our Recommendations for Setting WIP Limits

There are multiple ways to set WIP limits. Here are a few examples:

- WIP limit by person: for example, two work items per person.
- WIP limit by team: for example, twice the number of people on a team. A team of five people might have a WIP limit of ten for their Kanban board.
- WIP limit by work states: for example, three in Design, four in Implement, and five in Deliver.
- WIP limit by work item type: for example, one big project, two smaller projects, three maintenance tasks, and one improvement task in play, anywhere on the board.

Our favourite WIP limit setting strategy is to count up what you have right now – today. Make that number visible on your board and ask: is the work flowing smoothly to production, or are you stumbling over each other trying to get stuff done? Does it make sense to have this much work in play right now? The answer is almost always that there is too much WIP.

For example, if the team is working on 13 initiatives at the same time, and a whole lot of work is sitting in inventory, then voila – you have your answer – reduce your WIP. In addition, WIP is an indicator of Lead Time (LT). With most work items, we don't know how long they will take until they have been completed.

We suggest that you experiment with a lower amount of WIP, measure progress (see 4.5) and then have a conversation with the team to see if you should adjust again.

The secret to optimizing the flow of work across the value stream is to emphasize a request has to be completed before a new request can be begun. “Stop starting and start finishing”³ is a wise Kanban adage, (and it's also the title of a great guide that covers the core concepts of Kanban for knowledge work).

Summary of section 4.4

The upfront planning work that teams do should be considered work-in-progress. Set the organizational protocols so the requests coming into your team are balanced against the capacity the team has to do the work – and to get feedback on the work. WIP is the necessary tension in the system that can be adjusted over time to help teams achieve extraordinary success. The right amount of WIP tension is critical in allowing both creativity and innovation. Ensure the WIP is not so overwhelming as to cause workers to snap and burn out. Timely feedback can assist you to make beneficial changes.

4.5 Measure progress

Everything takes too long.

As a result, we often neglect the work that matters most today so we can work on yesterday's priorities, which we thought would already be done. Expectations about when things will be finished are outrageously unmet, and customers are unhappy. Very unhappy.

The solution is to clear the way for the most important work to get done fast and for it to be good enough from the customer's perspective. The first step in achieving this goal is to measure just how long work actually takes to do from an end-to-end system view.

Like with the boy scout troop described in section 3.2, it doesn't do a company any good for one team to deliver work faster than the teams downstream can consume it. The end product or service doesn't get delivered to the customer any sooner. The company, as a whole, only moves as fast as the slowest-moving part of the system – the constraint.

³. Rook, Arne. Stop Starting, Start Finishing! Place of Publication Not Identified: Lean-Kanban University, 2012.

When the boy scout troop needed to go faster, they had to figure out what was holding Herbie back. It turned out that his backpack held a six-pack of soda, four cans of food, and a large iron skillet. The troop divvied up the load from Herbie's backpack and carried on their journey. Relieved of most of the weight, Herbie walked faster than he previously had, meaning the whole troop could move at twice their previous speed. Herbie was no longer the bottleneck. The constraint moved on to the next thing preventing the troop from moving as fast as they'd like. Could be an injured scout. Could be a fallen tree.

Every system has a constraint. And once it's resolved, the constraint "moves on" to the next slowest part of the system. A focus on finding the bottleneck enables work to be unblocked, which allows it to move forward, thereby delivering value faster.

Some people unfamiliar with the theory of constraints may feel bad about being the bottleneck. Don't do that to yourself. The benefit of being the bottleneck is that you get the attention you need. Organizations following DevOps, Lean and Kanban practices don't punish the bottleneck. Instead, they work out how they can help.

The theory of constraints teaches us to identify constraints and how to exploit them. For software development and IT Operations teams, that means divvying up the load, automating what can be automated, and removing unnecessary activity.

4.6 How to Measure Progress

When it comes to metrics, it's essential to measure what the customer cares about. People ask questions about the things that matters to them. What do your customers ask you? Our guess is some version of "Can you go faster?" or "When will it be done?"

Given this, here are three questions you might ask yourself:

1. What is the arrival rate of work compared to the closure rate?
2. How long does it actually take to do the work?
3. Where is the work getting stuck?

4.6.1. What is the arrival rate of work compared to the closure rate?

It's optimistic to think that we can achieve everything on our to-do list, but let's quell that myth right now. It's not possible. There are always more things to do than there is time to do them. Always.

When you measure the arrival rate of work and compare it with the closure rate, the data will invariably show how much the team struggles to keep up with demand. Visualizing all requests emphasizes the reality of the situation and shines a light on the need to clarify goals, so the team can work on the most important stuff and not be completely overwhelmed.

To compare arriving requests with completed work, use Cumulative Flow Diagrams (CFDs), which map arrival rate onto closure rate data. The CFD in Figure 5: Example CFD Showing Increase in Demand shows an increase in demand over the last 30 days. The dark blue ribbon of colour represents incoming work item requests. The huge influx of work requests between Aug 9 and 15 (165, to be exact) indicates that demand has quadrupled over the previous 30 days.

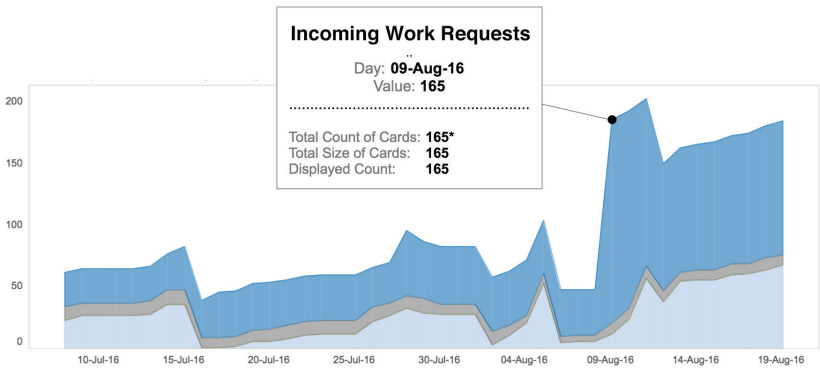


Figure 5 Example CFD Showing Increase in Demand

Determining what to do and what not to do is an essential decision.

The dark blue ribbon represents the number of incoming work requests on the Kanban board. The grey ribbon represents the work items in the Implement (or Doing) state on the board, and the light blue ribbon represents the complete work in the Done column.

We can look at this data and decide what to do. Is it possible to do only a certain percentage of it? What are the most important things to do?

If a mentality of do everything exists, people will take on more work than they have capacity to do. Creating conditions where it is acceptable for team members to question requests and say, “No, we have too much WIP already,” is a good counterbalance approach to expediting the completion of the most important work and letting go of the notion that it’s possible to match the demand with the team’s capacity to complete every request.

Additionally, the CFD shows how much WIP is in your system. The vertical distance (shown by the black arrow in Figure 6) between the top line (arrivals) and bottom line (closures) is an accurate count of the number of items in progress.

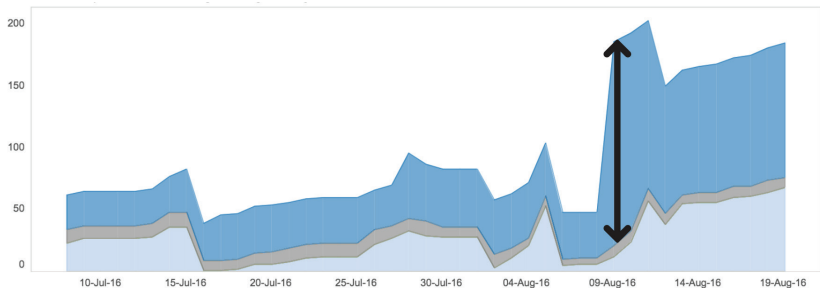


Figure 6 Vertical distance between top and bottom line of CFD is the amount of work in progress

The CFD is a powerful graph to compare demand against completed work items and to see how much WIP is in the system.

4.6.2. How long does it actually take to do the work?

Lead Time measures the time it takes work to flow end-to-end, from the time it was first requested to the time it is considered done. LT includes process time plus wait time. LT is important to measure because it allows us to quantify the probability of completing X% of work in n days.

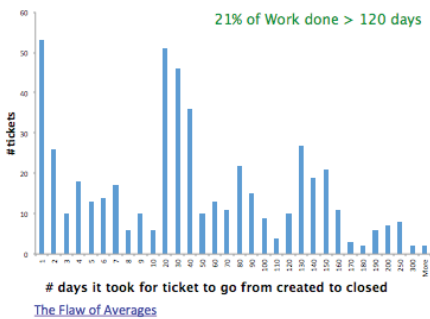


Figure 7 Histogram showing Lead Time tracked over time

Measuring LT is straightforward. Start the clock when a work item becomes a live work-in-progress item and stop the clock when it moves into the column where it is considered done. Most lead time is actually wait time and not work time. It is arguable that only 5% of LT is spent on actual work. Work sits idle, waiting, for 95% of the time.

A good way to visualize LT is using a histogram. The histogram in Figure 7 shows the distribution of LT for the work items from Figure 3.

Here, we see that more work items (53) were completed in less than one day than within any other timeframe. At 10 days, the histogram x-axis changes to 10-day brackets, and shows 50 work items were completed between 10-20 days and approximately 28 work items were completed between 120-130 days. Histograms are great for understanding the amount of variability in LT data.

This distribution, however, is not normal. Note the long tail. This is an example of why it's risky to use the average number of days as an estimate. If we are measuring LT to set expectations, and we used the average here, we'd be wrong most of the time because there's so much variability. Although it's a very small percentage, some of these work items took >200 days to complete. Statistician John Tukey, perhaps best known for coining the term "bit," said it well: "The idea is to use this data to be approximately right, instead of exactly wrong."

A scatterplot view of Cycle Time (CT) with percentile lines (as shown in Figure 8) shows the variability in the time it takes to complete work. Here, we see that CT ranges from less than one day to 14 days. The horizontal 95 Percentile marker reveals that 95% of the work is completed in less than 7 days. This helps us to be more predictable. When people ask, "When will it be done?" the CT (and/or LT) chart comes to the rescue. Given this data, we can say with a high level of confidence that 95% of these types of work requests will be completed within 7 days. Probabilistic data is incredibly useful and powerful to improve decisions. The capability to forecast how long tasks will take helps with prioritization.

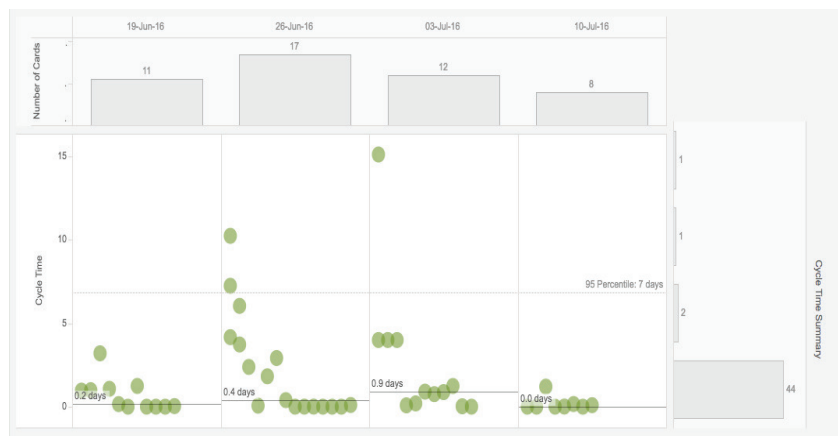


Figure 8 LT scatter plot showing probability of completing work

Another useful metric is Mean Time to Repair (MTTR). In the example from section 5.2, where 30 expedites occurred over a period of 60 days, the MTTR metric was 47 minutes. Looking at how MTTR is trending allows us to quantify the frequency of expedites and the team's capability to fix them.

Time metrics are a great way to view trends. Tracking LT over time helps us evaluate changes and forecast future work. It can drive conversations on process improvements, automation and bottlenecks.

4.6.3. Where is the work getting stuck?

If we're going to estimate anything, we should estimate how long things will sit waiting. For example, measuring the LT of work represented in Figure 6 allowed us to see that 21% of all the work took more than 120 days to complete. Much of that LT time was due to work sitting idle in the *Validate* queue.

For this team, seeing the age of each work item resulted in the discovery that some of their requests had been sitting in the backlog, inactive, for more than 12 months. As a result, the team adopted a new policy: delete any work item in the backlog that hasn't been touched for six months. Their thinking was that if it's important, it will reappear. This approach might not work in every situation, but it worked for the team in question.

The flow of work is what we're concerned with here. Your team's ability to disassemble work into small batches impacts on the speed of the work on its way to production. When you find out where work gets stuck, you can work to unblock it, and restore flow.

A good Kanban design identifies where work gets stuck – where the bottleneck is.

Summary of section 4.6

Work takes too long. Because we overbook ourselves, we don't have the capacity to work on the things that matter the most today. We are still working on tasks that mattered last week. Setting better expectations helps reduce the level of unhappiness that frequently occurs from falling behind. Use data to understand constraints and drive change. Put conditions in place that allow work to flow faster from a big picture end-to-end systems perspective. Over time, setting and adhering to appropriate WIP limits will grow a team's capability to take on and finish work faster. Setting realistic expectations based on relevant metrics arms you with credibility. Faster flow can be achievable when you reduce interruptions and waste, and avoid context switching. Remember, multitasking is a cruel joke attempted by busy, but unproductive, people. Avoid it at all costs.

4.7 Communicate the Work State

In an attempt to prevent a PR disaster and a significant loss in market share, the executive team of a multi-billion dollar manufacturing company mandates project X. Project X's goal is to discover which customer accounts still use an older version of a product that is nearing the end of its life and is no longer supported, but which is still used by 50% of the customer base! At the same time, a related sister product must be replaced for customers who have the new version of the first product, because the release of new software made it incompatible with the older sister product.

Two separate product groups are responsible for each product. Each Product team has their own product roadmap, but the plans aren't shared with each other. The different teams don't talk to each other, so they don't understand the dependencies and compatibility issues. Meanwhile, the company is losing significant market share to its competitors.

In more ways than one, invisible decisions destroy organizations.

This true story is one of many in the technology sector. It describes the disastrous effects of siloed teams not sharing mutually critical information. In this case, the cost of "I didn't get the memo" isn't just delayed or missed communication, it's incompatible components, unsupported software, sluggish LT, and delayed time-to-market, all of which affect the company's ability to remain competitive. Neither team had a transparent view of the big picture. When the big picture is invisible, people are unintentionally blindsided by the impacts of obscured information.

In more ways than one, invisible decisions destroy organizations. Invisible decisions enable companies (like the one described above) to miss critical dependencies between products. Invisible decisions also have a habit of ignoring people who should have been invited to weigh in on the impacts of those decisions.

The solution to this avoidable problem is to provide a mechanism for people to see the big picture, allowing workers to visualize connections between related items, to point people to decisions that have already been made and, if appropriate, to point them to wherever they may contribute to decisions that concern them.

4.8 Making Big-Picture Risks Visible

In the same way that a team-level Kanban board makes teamwork visible, programme-level risk boards can make cross-functional teamwork and multiple product team risks visible. Here, a programme-level board provides a big-picture view of all the work in progress. The high-level work items are connected to individual team boards. Teams flag the work items and connect them to the programme-level risk board, not only to achieve greater visibility, but also to escalate issues to management and bring awareness to the larger organization.

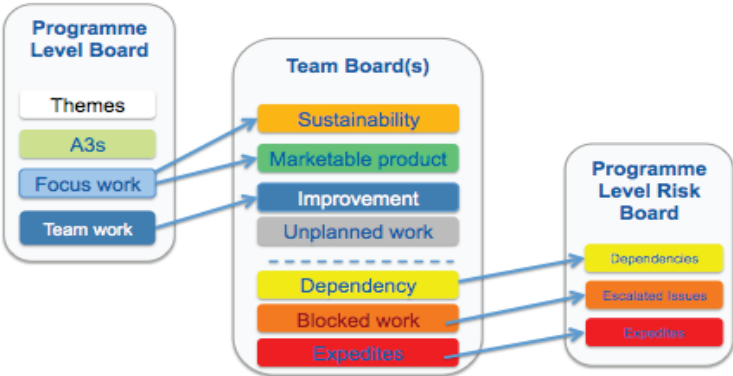
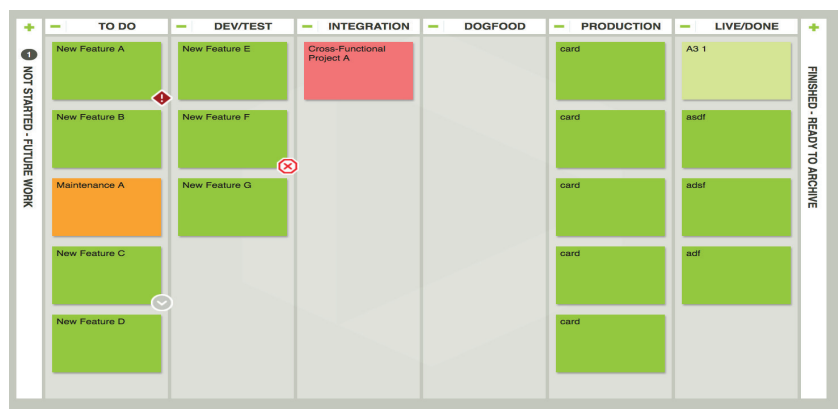


Figure 9 Show Big Picture Risks

4.9 A Look at a Programme-level Kanban Design

While larger in scope, a programme-level board has the same characteristics as a team-level board. Visualizing the work of an enterprise organization doesn't have to be daunting. In addition to being a more manageable task, starting with the high-level view has the advantage of showing you how your work fits into a larger value stream (i.e., how your work travels from its source of demand to delivery). Seeing the big picture helps your team consider the whole organization, rather than just their team.

The Kanban board in Figure 10: Product Development Roadmap represents all of the work that a product development team (including IT Operations) is currently working on. This one-page, high-level view reveals a lot of information, including the type of work, subtasks, status, and related work.



The beauty of Kanban is that it allows for different work item types – after all, not all work is the same.

Figure 10 Product Development Roadmap

Type of Work

The different coloured work items signal different kinds of work. The beauty of Kanban is that it allows for different work item types – after all, not all work is the same. New features, for example, often require different specialities to maintenance work. In this Kanban design:

- Light green cards represent A3s, a communication tool borrowed from Lean manufacturing to help business and technology teams clarify approvals and prioritizations.
- Dark green cards represent revenue-generating work; i.e., new features and enhancements initiated by sales and marketing.
- Orange cards represent revenue-protection type of work, such as keeping production stable and reliable, and also maintenance work
- Red cards represent major cross-functional team issues.

Since different types of work may require different actions – and with variable urgency – visually calling attention to the different work item types helps people quickly understand the nature of the demands of each work item.

Status of Work

The entire team can see which work items are in progress, as well as their current state. Four of the work items in Live/Done have recently been completed. Five work items have been delivered to Production, one work item is in Integration and three work items are in Dev/Test. There are five work items in the To Do column, signalling the incoming demand for this organization.

Delayed Work Items

On the second card in the Dev/Test column, the red “X” signals a blocked work item. Seeing where work is stuck in the pipeline is one of the best features of Kanban. Intentionally highlighting blocked work lets others know where a problem is holding up progress and compels them to decide what to do about it.

Section summary

Failed communication can undermine workers who are working with outdated information. Cooperation across siloed functional teams and regular teams alike requires a big picture view – not only one that’s able to be viewed by everyone who’s impacted, but also one that shows work in progress, when work is happening, who is involved, and where the important decisions are being made. Everyone benefits when they can see and understand the rationale behind a decision.

Invisible decisions destroy organizations for at least two reasons:

- People boycott decisions if they have zero opportunity to contribute.
- Workers march in different directions (with the best intent) when they are unaware of mutually critical information.

Organizations capable of negotiating hand-offs and visualizing connections between teams place themselves and the organization in a safer and more effective position. Teams who haphazardly ignore each other set themselves up for risky incompatibilities.

4.11 An Electronic Version of a Team-Level Kanban Design

As members of geographically separated teams, we dial into meetings to discuss priorities, issues, and ideas for how to implement changes. Even though we can sort-of-sometimes see one another in the meeting app, we unintentionally interrupt each other. We don't hear each other well when the meeting room echoes, or the headphones break, or someone's dog barks. Online discussions are never as good as face-to-face discussions. A natural affinity develops during face-to-face encounters, much stronger than from the other side of the continent, or even across the street. The benefit of co-located teams is well documented, but it's a rarity in today's world. More often than not, teams are distributed across the globe.

Although a physical board meets many needs of co-located teams, it lacks the ability to communicate well outside of the physical office.

The need to communicate well is critical. Electronic Kanban tools enhance communication that would otherwise have been lost in email or spreadsheets.

Similar to a phone call where nuances are difficult to hear, the metrics necessary for continuous improvement are difficult to see in a spreadsheet. Capturing key metrics such as cycle time, work-in-progress, and flow efficiency is not impossible to do with a physical Kanban board, but it's less satisfactory when compared to a Kanban tool that automatically generates those reports.

The details available from electronic boards allow for fast and accurate communication on the work item itself. The anatomy of a work item card typically includes headers, title, description, assignees, comments, sizing, and if needed, a due date. See Figure 11. A physical board may include this metadata, but it has to be added long-hand which might be hard to sustain over a long period of time.

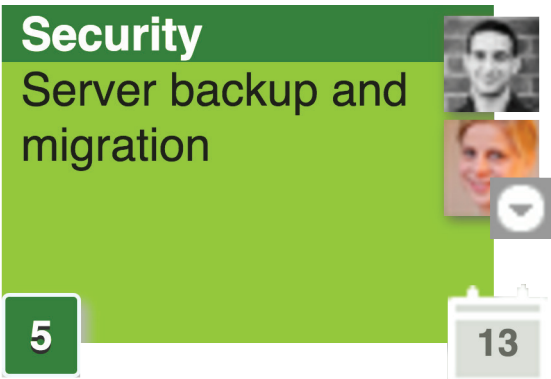


Figure 11 Card details

The board in Figure 12 represents an embellished electronic view of the design of the board in Figure 3.

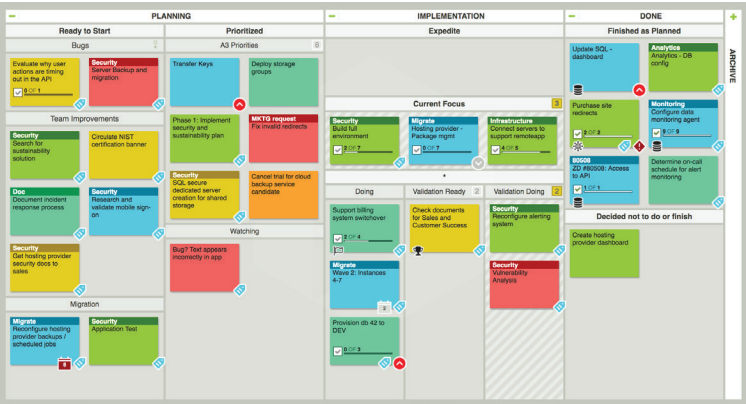


Figure 12 Electronic version of IT Operations team board

Here, we see the Ready to Start area, where a good amount of planning occurs. Discussions on dependencies, scope, and capacity occur in this phase of the workflow. Effective planning of the work allows the team to reach agreement on priorities and logistics of what, who, when and where. When work items/cards are moved to the Prioritized column, the team is crystal clear on what work will be done next and what work will need to wait, because they know they can't work on everything at once. In this board design, planning discussions ensure teams are aligned before work is pulled into the Implementation area, which includes Doing and Validation. Note the increased visibility of blocked work, bugs, expedites, related work, and assignees.

5 Kanban and IT Service Management

Now that we have explored what a good approach to Kanban looks like, let's take a look at how to get started when moving to Kanban from established but ad hoc work management, or from more standardised traditional ITSM work management models.

As in the rest of the document, we are focusing on the most widely adopted part of ITSM, IT Operations, rather than trying to cover work resulting from all lifecycle stages and processes. We believe the current challenges in IT Operations warrant narrowing the scope like this.

Many IT Operations professionals have grown up in an environment where having more work than capacity was the norm, and finding creative solutions to keep the fires under control was part of daily work. While widely acknowledged as an unsustainable model, it has prevailed for a long time, and has, among other things, been partially responsible for the emergence of the 'hero culture', where the IT Operations specialist saves the day. Again.

To start solving this puzzle, we must first acknowledge that there indeed is a problem with the current way of working. In addition to the undeniable stress in the IT Operations team and the destructive nature of this model for IT Operations professionals' health, it is important to link the proposed improvement to organizational goals.

IT Operations have often been good at hiding the deficiencies of the existing work models in the past, and concerns raised today are frequently not taken seriously.

Our experience has shown that, in most organizations, the success of any change initiative depends on strong and explicit management support.

To secure this support, the sense of urgency needs to be described in the language of the organization, not just in the language of an individual or team. So the benefits of moving to Kanban should be explained through the value it brings to the whole organisation and its customers.

5.1 Concept of work in ITSM

Earlier, we discussed different types of work that IT Operations professionals perform on a daily basis. While ITIL recommends tracking various work items through their lifecycle (from creation to resolve/close), improvement initiatives that lead from ad-hoc work management to organized work management are often performed as process-based or team-based initiatives, and rarely is work management addressed across the lifecycle.

The result of this more traditional approach is a set of individual workflows, one per process. While specifying the type of work item that is needed for better planning and reporting, performing the work required to handle any work item is still just that – work. Having to follow different and often disconnected workflows adds to the cognitive load of IT Operations professionals, and creates significant confusion over work priorities, as well as challenges with work visibility.

The varied nature of IT Operations work makes it much more difficult to estimate and plan than e.g. software development. This is why applying Scrum principles to IT Operations work is rarely successful. Agreeing on the scope of and committing to the work for the next Sprint (be it two weeks or longer) is not realistic, as a significant amount of work is unplanned. IT Operations specialists can, of course, be included in Sprints as contributors but, for them, the work coming from this stream is only one type of work out of many. And, while many organizations are undergoing major improvements in aligning workstreams and business processes, in many organizations, IT Operations still receives work over-the-wall, rather than being involved in planning from day one.

5.2 Start where you are and progress iteratively

If you are moving to Kanban from an ad hoc or an isolated multiple queue model, as the first step, you might want to start by making your work visible, and hold off setting hard WIP limits – it would not be wise to start setting arbitrary constraints before gathering information about your team's true capacity. Your first board should reflect your real situation as closely as possible, rather than attempting some idealised model. Otherwise, the board would be unusable from day one, and the whole initiative is likely to gather opposition, rather than a momentum. This is not to say WIP limits should be ignored - definitely not! A discussion about WIP limits is essential to the successful adoption of Kanban, and this should be kept in mind from day one.

Our experience has shown that, in most organizations, the success of any change initiative depends on strong and explicit management support.

Another pragmatic reason to potentially delay the implementation of WIP limits is the lack of tooling support. Most ITSM software on the market does not yet support Kanban boards and, while there has been significant progress in the past few years, Kanban capability is usually limited to visualising the work and not on setting WIP limits. We expect this to change in the near future. Most of the tools with full support for Kanban boards currently come from outside the traditional ITSM market.

In any case, we suggest you begin with a physical board, if possible. Only later consider moving to an electronic board once you've begun to understand the real dynamics of your work. For many teams, a low-tech option is fine when flow or efficiency metrics are not required. There is no mandate for an electronic board – a physical board in the room where your IT Operations professionals work can be sufficient.

For distributed teams, a physical board can be problematic, but not impossible. We have seen physical boards live-streamed via a webcam to remote teams, where the physical cards are moved whenever a remote worker pulls a card into their queue. The overhead of such arrangements is negligible compared to the value received.

5.3 Collaborate and work holistically

As noted earlier, while IT Operations professionals need to complete tasks of a variety of types, it is all part of their normal daily work. The ability to track different types of work items by category is useful for planning and reporting, but it is usually irrelevant for the person performing the work. It matters little whether the task at hand came from the problem management or the change management queue. Also in most cases, the metadata that needs to be captured for each task is similar. For example information like start time, dependencies, deadline, etc.

In some organizations adopting DevOps practices (e.g. those of Continuous Integration and Continuous Deployment), it is common to have a shared Kanban system for the whole product team, either on the same board in the case of smaller teams, or a combination of individual detailed team boards linked to a high level common board for larger teams. This allows both the development and operations tasks to be more easily tracked as part of one flow.

In most organizations, development work and operations work is still separate, and creating a unified queue across these two work-streams is not yet realistic. Therefore development can be expected to be one of the sources of the work coming to the operations team, alongside other planned and unplanned work. Regardless of whether the development team uses agile practices or follows a waterfall model for their planning and delivery, IT Operations professionals should always be included in the planning phase. It is not wise for the development team to commit to work that depends on other teams for finalization without receiving their input/feedback and checking the capacity of those teams beforehand.

As a word of caution - we have seen organizations creating workarounds where the definition of done for the development team is redefined from successfully released to production to handed over to Operations. This is an anti-pattern for successful collaboration. A model like this breaks the feedback loops that are so important for successful collaboration and is likely to result in a local optimization exercise that has questionable value for the whole organization. The improved velocity, or capacity, of the development team is irrelevant if there are bottlenecks further down the workstream.

5.4 Keep it simple and design for experience

The fact that work comes from different processes does not mean there is a need for different boards. But, to improve the visual representation of work across a variety of processes, different swimlanes can be created for different types of work. So, for instance, there could be separate swimlanes for incident-related, problem-related, and change-related work, as well as a swimlane for expedites. This should not necessarily be seen as a substitute for assigning different coloured cards based on the type of work, but as an additional method for improving the visualization.

Using different swimlanes can make it easier to decide which item to pull from the queue next. For example, one person may be working purely on incident management, and therefore will only pull from the incident management swim lane. Once the work has been pulled, however, different colours make it easier to track different types of work. Also, different swimlanes can accommodate any variance in process steps that would impact the way the columns are used – for some processes, certain steps might not apply and can be grayed-out in that swimlane. For instance, triage could apply to incidents only.

5.5 Focus on value and be transparent

Different types of work can be prioritized using different rules. For most IT Operations teams, resolving incidents almost always receives the highest priority, followed closely by service requests. Both types of work are usually linked to SLAs and have agreed timelines for completion. Add to that the requests coming from development, which are often time-bound but less urgent, and you can see this leaves little room for any non-urgent work (for example, housekeeping type tasks, as well as (proactive) problem management).

Using a Kanban board for work management does not mean that whatever is in the backlog on the board automatically gets a higher priority than incoming work. The agreements around work completion deadlines still apply as they did before introducing Kanban. The benefit of having the non-urgent work on the board is the transparency it creates. The challenge with having too much work and not enough time will be visible for all to see, rather than being a nagging feeling at the back of the mind of the IT Operations professional.

Once a specific task has been sitting in the backlog for some time – a timeframe to be decided by the team itself – an honest evaluation can be carried out of whether the task is even relevant. If it is, and the value for the organization from performing it (or the damage from not performing it) is clear, a request can be put in to increase the size of the team. Without a visible track record of where the time has been spent and why the task is still waiting to be pulled in, it is often difficult to back up that request. Showing that there is more to do than there is time is understandable, and should not itself be a source of stress. Only through transparency can the situation be improved.

5.6 Measure what matters

Whenever you use Kanban to assist with the improvement of your work management, don't forget to measure the results. Equally, it is important that the metrics you put in place are balanced, and that they answer questions on behalf of both your team and your customers. The organization is interested in opportunities for work optimization, for shortening the timelines required to deliver results, for increased transparency, and for improved quality, among other things. The data you capture can be used to answer these questions, as well providing the information the team and its members need to improve.

When it comes to defining additional metrics, it is best to progress iteratively, rather than measuring everything at once. Introducing a Kanban board can be a great opportunity to review current metrics and confirm these are still relevant and useful. Displaying a board may eventually remove the need for specific reports and status updates, which would allow the team to focus on improvement initiatives, rather than designing reports. And be careful to ensure that stakeholders are not negatively impacted when work management moves to a Kanban board. It has to work for everyone.

5.7 Keep improving

Improving the way that Kanban is used should become part of continual service improvement (CSI) – applying lessons learned to further improve the value the team and the organization get from Kanban.

Additional practices can be introduced as time progresses. Even before addressing the WIP limits, you can introduce regular meetings, similar to daily stand-ups in Scrum, based around the work visualisation. In these meetings, the IT Operations team members can spend five to fifteen minutes discussing the backlog and agreeing the priorities for which tasks should be pulled in next. This does not mean that team members have to pull the tasks in regardless – the rules around capacity still apply – but the meetings will help with the holistic view of work across the various products and processes.

It also doesn't mean that the meetings should be scheduled just to have meetings – the meetings can help the team to find the time to collectively analyse the board and agree on the priorities for the next day or few days. Regular meetings are of course not the only way to achieve this objective.

The team can also agree on the rules for escalating tasks should there be a risk of breaching an SLA. Procedures around swarming can be put in place, which work similar to the concept of the Andon cord in Lean. This would require either a predefined subset of the team, or perhaps even the whole team, to gather and help solve a specific task. This is most likely to happen with high-priority incidents. A default procedure can be agreed for all expedites – the moment a high-priority incident enters the queue, swarming can be triggered and other work currently being undertaken by this team or team subset, unless of equally high priority, will be paused until the task in question has been completed.

Additional visual cues – like indicating blocked items, including deadlines on cards, adding a name and/or a photo of the team member currently working on a specific task on its card – can also be helpful in making the most of the Kanban board.

The objective of having a Kanban board is not to have a Kanban board – it is to get better at getting work done.

6 Summary

Organizations take on more work than they have capacity to do, and much of it is invisible. Invisible work destroys organizations. In addition to causing long lead times for products and services, invisible work results in unhappy customers, frustrated business leaders, system incompatibilities and high-strung employees.

We recommend starting Kanban with a non-mission critical project which can be used to learn how Kanban works best for you. If no large white board is available, a hallway wall or flipchart can work. Even jot it on a piece of paper on your desk could serve as a starting point. Often, low-key is more effective than fancy management software programmes.

Using Kanban to continuously improve delivery and services in your organization is inexpensive and relatively easy – simply categorize the work and make it visual. Then take a long hard look at where the work stalls, and what prevents the work from moving. If planned work is always getting hijacked by unplanned work, the team's best efforts to deliver value will be sabotaged. Hijacked (or stalled) work is a risk that, when flagged appropriately, can appear on a risk board to automatically get the attention it deserves from management.

Highlighting idle work is a superb way to improve communication. These necessary conversations on all things related to tension are critical elements in the success of the organization, and are why multiple levels of boards are excellent components to include in your Kanban implementation. Good decisions stem from good communication which stems from transparency of risks. Making decisions visible enables people to anticipate what's heading their way so tension can be reduced.

A glossary of Lean and Kanban terms listed alphabetically

Change

The addition, modification or removal of anything that could have an effect on IT services. The scope should include changes to all architectures, processes, tools, metrics and ocumentation, as well as changes to IT services and other configuration items.

Constraint

A bottleneck in the system. Something constraining the throughput.

Cycle Time (CT)

The elapsed time it takes to complete a request from the time work began to the time it was completed.

Dark Matter

Invisible work, e.g. organizing email, attending company briefings, updating work tools, etc. Also, in many cases, most of the maintenance work that is not driven by tickets.

Deployment Lead Time

The time it takes to deploy a change once code is checked into source control.

Expedite

A high risk issue or incident in need of urgent attention, e.g. server utilization at full capacity.

Flow Efficiency

A calculation based on two components of lead time: work time and wait time. Flow Efficiency shows the ratio of work time compared to wait time for a work item.

Incident

An unplanned interruption to an IT service or reduction in the quality of an IT service or a failure of a Configuration Item (CI) that has not yet impacted an IT service.

Kanban

Japanese word for visual signal. Used throughout this paper to refer to a visual management pull system for knowledge work.

Lead Time (LT)

The elapsed time it takes to complete a request from the time it was first requested.

Lean

A Socratic philosophy used to manage workflow, using a pull system.

MTTR

Mean Time To Repair. The average time it takes to repair a service component.

Problem

The underlying cause of one or more incidents.

Request for change

A formal proposal for a change to be made. It includes details of the proposed change, and may be recorded on paper or electronically.

Service request

A formal request from a user for something to be provided – for example, a request for information or advice; to reset a password; or to install a workstation for a new user.

Swarming

A collaboration model in IT support where professionals from several service teams or support levels come together, physically or virtually, to solve a (critical) incident. Sometimes used as a replacement for an escalation model.

Swimlane

A method for categorizing work on a Kanban board using separated horizontal lanes based on the type of work, assigned team/individual, normal/expedite items, or any other meaningful way, as required.

Systems Thinking

A holistic view of the system where the goal is to optimize the whole system versus individual functions or silos.

System

A network of interdependent components that work together to accomplish a goal. A system includes the people doing the work, and the impacting rules and tools.

Pull System

When new work is pulled into the system based on available capacity to handle it, rather than work being pushed into the system.

Queue

A pile-up of work waiting to be worked on. Work that's in a wait state.

Value Stream

The activities done from beginning to end for a specific product or service in order to provide business value.

Work State

The state that the work is in. Work flows through different states on its way to completion. The work states show us where the work currently is in the pipeline.

Workflow

The flow of work through the pipeline (or system) from beginning to end.

Work Item

Anything being worked on. Work that encompasses effort – both large and small.

Work InProgress (WIP)

Work-in-progress equals all the work started, but not yet finished.

About the authors



Dominica DeGrandis teaches Lean, Kanban and Flow to technology and business professionals. Her passion involves helping people improve workflow and optimize throughput. Dominica’s pioneer experience using Kanban in the build, deploy and release space contributed to early support for DevOps enthusiasts. She is keen on providing visibility and transparency across teams to reveal mutually critical information and enable extraordinary change.

As Director of Training & Coaching at LeanKit, Dominica combines experience, practice and theory to help teams level up their capability. She holds a BS in Information Computer Science from the University of Hawaii. Dominica blogs at leankit.com and ddegrandis.com, and tweets at [@dominicad](https://twitter.com/dominicad).



Kaimar Karu is the Head of Product Strategy and Development at AXELOS, leading a team of international experts that look after the IT Service Management (ITSM), and Project and Programme Management (PPM) best practice portfolios, which include the products ITIL and PRINCE2. He has a diverse career background in IT, having worked in IT Operations, Software Development, Project and Programme Management, and IT Service Management. He has a passion for helping people learn and improve, and has worked as a teacher, trainer, and coach in schools, universities, and professional training organizations across Europe. Before joining

AXELOS in 2014, Kaimar spent three years immersed in the world of startups, Nordic pragmatism, and multi-billion dollar acquisitions, working with Skype in his native Estonia.

Kaimar has held the position of the president of itSMF Estonia since 2013, holds a master’s degree in Philosophy, and has won a 2nd place in the national beer sommelier competition. He’s now based in London, but is on the road most of the time, exploring good, emergent, and novel practices in organizations around the world. He tweets about good food, best practices, and continual improvement at [@kaimarkaru](https://twitter.com/kaimarkaru).

Acknowledgements

We are very grateful for the contribution the following people made to the development of this paper:

Akshay Anand, Roman Jouravlev, Lari Peltoniemi, Dave van Herpen, Adam Haylock, Stuart Rance, Duncan Watkins, Greg Brougham.