STATIK

SYSTEMS THINKING APPROACH TO IMPLEMENTING KANBAN

STEP BY STEP INSTRUCTION BY
DAVID ANDERSON

Systems Thinking is a way of understanding how a system behaves as a whole rather than through analysis of component parts in isolation. It is a key influence in the definition of the steps needed to introduce Kanban in an organization. The steps in this process are not necessarily sequential, but iterative, using learning from one step to inform and influence others in a collaborative environment. The steps are:

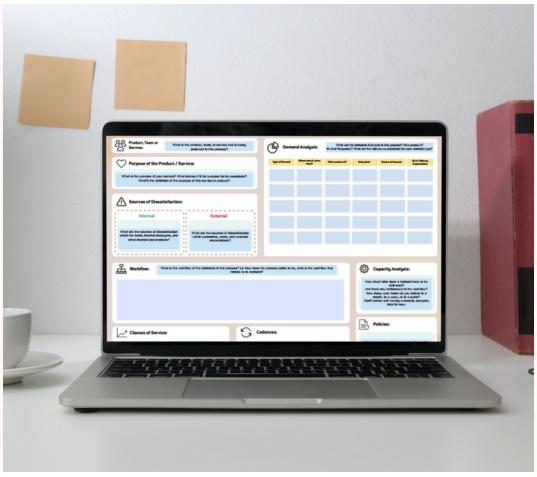
David Anderson is the author and pioneer of the Kanban Method and has more than 30 years of experience working in the high-technology industry.

Step 0: Identify Services

For each service...

- Step 1: Understand what makes the service fit for purpose for the customer
- Step 2: Understand sources of dissatisfaction with the current system
- Step 3: Analyze demand
- Step 4: Analyze capability
- Step 5: Model workflow
- Step 6: Discover classes of service
- Step 7: Design the kanban system
- Step 8: Socialize the design and negotiate implementation

STATIK



STATIK is applicable to just one service. When more than one service has been set up, Kanban practices and cadences are applied to balance demand and flow across the multiple services, and continually improve. The ordering of the steps may vary in practice and it is normal to revisit steps in the pursuit of further improvement.

Step 1: Understand what makes the service fit for purpose for the customer

This first step is considered a more advanced topic and is often skipped with novice or low-maturity implementations.

Explore the criteria that define customer satisfaction with the service delivery. These are usually related to but not limited to lead time, quality, predictability, and safety or regulatory concerns. These criteria are known as the fitness criteria because they determine how the customer evaluates whether the service delivery is acceptable, or "fit for purpose."
Explore and establish expectation levels for each criterion. These are known as fitness criteria thresholds. They represent the "good enough" or the point where performance is satisfactory.

These metrics should become **Key Performance Indicators (KPIs)** and be used to establish Service Level Expectations (SLEs) or inform negotiations on Service Level Agreements (SLAs) where appropriate. The Fitness Criteria Thresholds will be used to drive improvements and evolutionary change.
Where this step is skipped initially, it will typically be revisited as the Kanban implementation and the organizational maturity improve and some quantitative rigor is required for further improvements.

Step 2: Understand sources of dissatisfaction with the current system

This is done in two steps: ask the customers what they are unhappy about; ask the service delivery organization if they have any internal sources of dissatisfaction – things that are preventing them from doing a good and professional job and delivering on expectations. Often the sources of unhappiness on each side, external and internal, can be matched – fix one and you fix the other.

For example, a customer might complain of unpredictable, late delivery, while internally, workers may complain of being interrupted and disrupted with unplanned or additional requests taking a higher class of service. If we can address the sources of unplanned, disruptive demand, we can eliminate the interruptions and the service delivery becomes more predictable. Fixing one problem can make both sides happier – the workers are not interrupted and can focus on doing a good professional job, and the customer receives delivery within a reasonable tolerance of their original expectation.

Sources of dissatisfaction provide input for the kanban system design. We will try to design the kanban system, its capacity allocations, and its classes of service to eliminate as many of the problems as possible. Judging how much can be achieved without invoking significant resistance to change is considered an advanced coaching topic and outwith the scope of Essential Kanban.

Step 3: Analyze demand

In order to design a suitable kanban system, we need to know the nature of the demand: who are the customers; what do they ask for; what is the arrival rate and pattern of arrival of requests; what are their expectations – a softer more qualitative version of step 1, if step 1 has been skipped, for example, "we expect more predictable delivery for the majority of requests".

The first task is to identify the work item types. *For example*, if we operate a café, we are in the cup of coffee delivery service business. What is the equivalent of a request for a cup of coffee in the environment under analysis? How many different types of request exist?

Work item type discovery can be a non-trivial problem and practitioners often struggle to reach a balance between sufficient detail and a suitable level of abstraction. *For example*, a cup of coffee may be too abstract, but every item on the menu as a separate work item type is too much detail. A balance can be struck if we separate out espresso drinks from a filter or drip coffee drinks, from tea and other infusions.

Having **several types** facilitates better risk management and indicates workflow and resources required such as the espresso machine. **Too many types** does not provide sufficient aggregation of similar requests to facilitate a probabilistic approach to the management of the non-deterministic nature of the demand.

For each identified work item type, we want to **understand the arrival rate** – the volume of demand, and the pattern of arrival – the rate of demand over time and at different times of day, days, or weeks of the month, months, or quarters of the year. We want to understand the ebb and flow of demand. *For example,* a café may recognize a spike in demand for cups of coffee between 8 am and 9 am, or immediately after lunch.

Understanding the volume of demand, the nature of arrival, and the expectations or business risks associated with types of request will enable us to design an elaborate kanban system with capacity allocation for different types and varying classes of service to cope with demand carrying specific risk profiles.

For example, some demand comes with hard immovable delivery dates such as seasonal events, holidays, trade shows, major world events like the Olympic Games, or regulatory dates which are not negotiable. How much of this demand exists? What is its arrival rate and when does it arrive? How much notice do we get? This type of work may require its own class of service to insure it gets priority handling and is always delivered on schedule.

In mature and advanced implementations, the demand is analyzed into categories such as value versus failure demand, deliverable versus information discovery demand; refutable versus irrefutable demand, and planned versus unplanned demand. This additional analysis informs the kanban system design but will also inform further improvement actions that may be taken later.

Step 4: Analyze capability

Analyzing capability is a step that is often skipped in novice or immature implementations or when creating a greenfield system with no existing service delivery capability as a precedent. Analyzing capability involves **studying historical data for service delivery**: lead time; quality both functional and non-functional; predictability; conformance with regulatory requirements or standards.

The current capability can be compared to the service level expectations of the customers. Given that there are sources of dissatisfaction, there will likely be a gap, perhaps a significant one, between current capability and existing customer expectations. **It is common for a historical cumulative flow diagram** to be prepared as part of capability analysis and some calculation of flow efficiency may be attempted if working time (effort expended) data is also available in addition to duration (elapsed calendar time).

Step 5: Model workflow

Workflow modeling should be performed for each work item type.
It is important not to confuse workflow modeling with Value Stream Mapping or Gemba Walk techniques from Lean Manufacturing and Toyota. Again, we are working in a professional services, intangible goods environment. Kanban has adapted to this by adopting the workflow mapping techniques from Lean Product Development. Both Reinertsen and Kennedy have described similar workflow modeling techniques but using slightly different language. In general, we prefer Kennedy's language but Reinertsen's approach to the outcome.

You map the series, or sequence, of dominant steps to discover new knowledge. **We are modeling knowledge work.** What activity gives us the most knowledge? At some point, there will be diminishing returns from that activity and we move on to the next dominant activity. In a software development process, we discover knowledge from analyzing a problem or a business domain. Later we may look to design a system to solve the problem or simulate functionality in the business domain. Later we develop code and tests. Yet later we run the tests. At each stage, we have more knowledge and more detail about the final deliverable. It is an iterative approach where more knowledge is layered on top of existing less specific knowledge.

Reinertsen uses the word "information" instead of knowledge but it amounts to the same thing – *define the sequence of dominant activities used to discover information about the finished product.*

It is important we map the activities and not the handoffs between people. Unlike value stream mapping or Gemba walking in a physical environment, we do not want a kanban system (and board) that

maps handoffs between people, instead we want a focus on the work and what happens to it. We want our kanban systems (and boards) to encourage collaboration rather than ossify existing functional divisions or specializations. **The word "dominant" is also important.** At any given time there may be multiple activities happening but one of them is the dominant activity for discovering new knowledge.

For example, when a software function is being tested, it is said to be "in test". It is the testing activity that is the dominant activity for producing new knowledge. Tests may fail and result in a rework of the code, or the design, or even the analysis. However, it is the test failing that produced new knowledge. In this example, Reinertsen's use of "information" is more enlightening. It is the test failing that provides us with the most, new information about the finished product.

Step 6: Discover classes of service

A **class of service is** a set of policies that describe how something should be treated. Typically with kanban systems, the classes of service describe the queuing discipline or priority of tickets. Classes of service can also describe a policy that affects workflow such as whether an item should receive treatment from a specialist or be tested to a specific level of quality. Classes of service might also provide information regarding scheduling or whether an item is allowed to exceed the WIP limit or not.

If there are existing classes of service and these are declared explicitly then they should be captured as the kanban system design will need to accommodate them.

However, it is more common that we are looking for hidden or implicit classes of service. *For example*, if there is a policy that one type of work is allowed to disrupt another type, effectively interrupting the worker and hence giving priority to one type over another then such policies describe implicit classes of service. We want to expose these explicitly because it allows us to ask the question, "Did you really mean for work items of type X to be given priority over items of type Y?"

Hidden classes of service exist where there is no policy but some items are treated differently from others. Often this happens because of the source of demand or the destination for delivery. It is common to find requests from a senior executive that are not passing through the formal governance and selection process and are also given top priority. We want to capture these currently hidden classes of service and **make them explicit**.

It will be important that either our kanban system is designed to cope with them, or that the exposure and transparency facilitate a discussion that enables us to design them out.

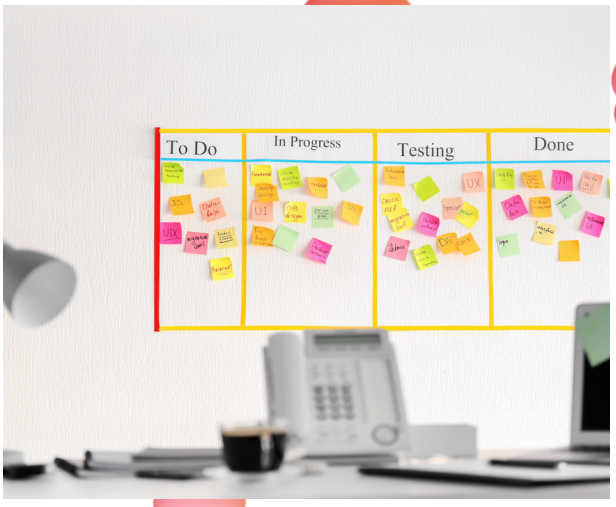
Step 7: Design the kanban system

A **kanban system consists of four core elements**: the kanban system and its kanban; the ticket design; the board design; adjustments to existing meetings and the introduction of some new ones to accommodate the feedback loops known as the Kanban Cadences.

To design the kanban system, we need the workflow model for each type of work, the states of work based on dominant activities for discovering new knowledge, and the classes of service. We will want to devise kanban (or WIP) limits for each state and possibly allocations of kanban across work item types and/or classes of service.

To design the ticket, we will need to understand what information is required at each state in the workflow, in order to make a selection decision to pull an item to the next activity state. Typically, we need the work item type, class of service, start date, due date (if applicable), specialist workflow or processing requirements, and potentially space to record elapsed time since starting, time blocked, elapsed time in a given state, business, technical, or delivery risks associated with a given item that would affect its selection, sequencing or scheduling.

To design the board, we need to understand the workflow for each type of work. We need to make a decision to have a single board for all work types and classes of service, or to have two or more boards. We need to decide how to allocate columns – usually, the states in the workflow – rows – usually work types or collections of work types – and color of tickets – usually for the class of service.



It is possible to make tradeoffs between the board design, the kanban system design and workflow, and the ticket design. *For example*, if we have a collection of optional activities that can happen in any sequence and even concurrently then we may choose to have a simple board design with a single column, a simple workflow state, and kanban system with just one

set of kanban for all of these activities, and use a set of paired "required"-"done" checkboxes on the ticket in order to cope with the non-deterministic sequencing and potential concurrency of activities.

As a general rule, the more deterministic our system, the more complicated we would expect workflow and board designs, while the less deterministic (in other words, the more non-deterministic) our workflow process then we would expect a simpler board design, simpler workflow model and kanban system, with a more complicated ticket design.

There are a total of **7 Kanban Cadences meetings** in the Kanban Method. It is common for new implementations to start with a subset of these. It is also common to repurpose existing meetings. Decide which of the Kanban Cadences you will be implementing. Decide the cadence of the meetings and reviews. Decide which existing meetings will be repurposed. You can utilize the workshop exercises from the Kanban Management Professional training to design your Cadence meetings: select the facilitator, frequency, and duration of the meeting, the attendees, what information each attendee should bring, which decisions will be made in those meetings, which metrics and reports are needed for the meeting.

Step 8: Socialize the design and negotiate implementation

The STATIK method encourages collaborative workshops for the analysis and creation of kanban systems, and boards. As a consequence of the collaborative nature of the workshops and STATIK exercises, it is common for everyone involved to be bought into the changes and to hold some ownership and pride in the design for implementation. In other words, our primary approach to gaining buy-in for adoption and roll-out is **collaborative involvement in the design.**

When selecting groups for STATIK workshops it pays to select a cross-functional group involving customers, other external stakeholders – influencers, decision-makers, and governance authorities – as well as team members to form the delivery functions. **The goal is** to have both internal and external people involved in the design.

Having said that, it isn't realistic that every affected stakeholder can play a role in the STATIK workshops. It is necessary to circulate the design output from the workshop in order to get a wider group bought into the design. While this topic is an advanced topic for Kanban coaches there are **some basics**:

It makes sense to **meet** with individual stakeholders privately**.** If they are external stakeholders start with a position of humility. Explain that you understand that service delivery has not been fit for purpose and you are seeking to make changes to how you are working in order to improve it and serve customers better.

Explain that the changes are mostly internal but external stakeholders may notice changes in how you interact with them, the interface they use for submitting requests and approving or selecting work for delivery, and in some of the metrics, reporting, and visibility they have in the new system compared with the older.

Walk them through how the new system will work, explaining it from their point of view. Listen to their feedback. You may get objections that will require adjustments to classes of service, capacity allocations (kanban limits), board design, or reporting requirements. As much as possible, promise to take these changes on board. Seek to gain agreement that the proposed changes will meet the stakeholder's needs assuming the changes are properly implemented.

Rework your design to accommodate what you have learned from socializing. Revisit some stakeholders to gain their agreement that you have accommodated their concerns adequately.

Once you have done this, you are ready to **hold a kick-off meeting** to launch the Kanban initiative and the introduction of the changes. Invite all the stakeholders. Again start the meeting with the position of humility.

You understand that service delivery hasn't been good enough, and you are **seeking to make changes** in the way you are working in order to **improve and satisfy everyone** involved within the reason that resources are limited and not infinite and some compromises have to be accepted. As much as possible you have **sought to preserve that which is working** well and to **minimize the changes** and focus only on what is needed to improve service delivery.

Those present may notice changes in how they interface with the group but otherwise, their **roles & responsibilities should be unaffected**. Now present the detail of the design explaining how each stakeholder and their work item types and required classes of service have been accommodated. For those, generally internal who are affected by changes in their role or responsibilities, explain what that means for them and for those with whom they work. **Provide detail** on a revision to existing meetings to accommodate the Kanban Cadences. **Explain** where boards will be situated or software being deployed.

You should be ready to get started.

DOWNLOAD STATIK TEMPLATE

on kmm.plus

go to: Resources - Download Templates

Any questions?

Email us to:
kmm@mauvisoft.com

KMM Plus