The Kanban Guide for Scrum Teams

January 2021

Table of Contents

Purpose

The flow-based perspective of Kanban can enhance and complement the Scrum framework and its implementation. Teams can add complementary Kanban practices whether they are just starting to use Scrum or have been using it all along.

The Kanban Guide for Scrum Teams is the result of a collaboration between members of the Scrum.org community and leaders of the Kanban community. Together, they stand behind *The Kanban Guide for Scrum Teams*. It is their shared belief that professional product development practitioners can benefit from the application of Kanban together with Scrum.

Relation to the Scrum Guide

This guide does not replace or discount any part of *The Scrum Guide*. It is designed to enhance and expand the practices of Scrum. This guide assumes the reader is operating a process using the Scrum framework. Therefore, *The Scrum Guide* applies in its entirety.

Definition of Kanban

Kanban (n): a strategy for optimizing the flow of value through a process that uses a visual, work-in-progress limited pull system.

Kanban with Scrum Theory

Flow and Empiricism

Central to the definition of Kanban is the concept of *flow.* Flow is the movement of value throughout the product development system. Kanban optimizes flow by improving the overall efficiency, effectiveness, and predictability of a process.

Optimizing flow in a Scrum context requires defining what flow means in Scrum. Scrum is founded on empirical process control theory, or empiricism. Key to empirical process control is the frequency of the transparency, inspection, and adaptation cycle - which we can also describe as the Cycle Time through the feedback loop.

When Kanban practices are applied to Scrum, they provide a focus on improving the flow through the feedback loop; optimizing transparency and the frequency of inspection and adaptation for both the product and the process.

The Basic Metrics of Flow

The four basic metrics of flow that Scrum Teams using Kanban need to track are as follows:

- **Work in Progress (WIP):** The number of work items started but not finished. The team can use the WIP metric to provide transparency about their progress towards reducing their WIP and improving their flow. Note that there is a difference between the WIP metric and the policies a Scrum Team uses to limit WIP.
- **Cycle Time:** The amount of elapsed time between when a work item starts and when a work item finishes.
- **Work Item Age:** The amount of time between when a work item started and the current time. This applies only to items that are still in progress.
- **Throughput:** The number of work items finished per unit of time.

Little's Law – The Key to Governing Flow

A key tenet governing flow theory is Little's Law, which is a guideline that establishes the following relationship:

$$average\ cycle\ time = \frac{average\ work\ in\ progress}{average\ throughput}$$

Little's Law reveals that in general, for a given process with a given throughput, the more things that you work on at any given time (on average), the longer it is going to take to finish those things (on average).

If cycle times are too long, the first action Scrum Teams should consider is lowering WIP. Most of the other elements of Kanban are built upon the relationship between WIP and Cycle Time.

Little's Law also shows us how flow theory relies on empiricism by using flow metrics and data to gain transparency into historical flow and then using that data to inform flow inspection and adaptation experiments.

Kanban Practices

Scrum Teams can achieve flow optimization by using the following four practices:

- Visualization of the Workflow
- Limiting Work in Progress (WIP)
- Active management of work items in progress
- Inspecting and adapting the team's Definition of Workflow

Definition of Workflow

The four Kanban practices are enabled by the Scrum Team's Definition of Workflow. This definition represents the Scrum Team members' explicit understanding of what their policies are for following the Kanban practices. This shared understanding improves transparency and enables self-management.

Note that the scope of the Definition of Workflow may span beyond the Sprint and the Sprint Backlog. For instance, a Scrum Team's Definition of Workflow may encompass flow inside and/or outside of the Sprint.

Creating and adapting the Definition of Workflow is the accountability of the relevant roles on the Scrum Team as described in the Scrum Guide. No one outside of the Scrum Team should tell the Scrum Team how to define their Workflow.

Visualization of the Workflow – the Kanban Board

Visualization using the Kanban board is the way the Scrum Team makes its Workflow transparent. The board's configuration should prompt the right conversations at the right time and proactively suggest opportunities for improvement.

Visualization should include the following:

- Defined points at which the Scrum Team considers work to have started and to have finished.
- A definition of the work items - the individual units of value (stakeholder value, knowledge value, process improvement value) that are flowing through the Scrum Team's system (most likely Product Backlog items (PBIs)).
- A definition of the workflow states that the work items flow through from start to finish (of which there must be at least one active state).
- Explicit policies about how work flows through each state (which may include items from a Scrum Team's Definition of Done and pull policies between stages).
- Policies for limiting Work in Progress (WIP).

Limiting Work in Progress (WIP)

Work in Progress (WIP) refers to the work items the Scrum Team has started but has not yet finished.

Scrum Teams using Kanban must explicitly limit the number of these work items in progress. A Scrum Team can explicitly limit WIP however they see fit but should stick to that limit once established.

The primary effect of limiting WIP is that it creates a pull system. It is called a pull system because the team starts work (i.e. pulls) on an item only when it is clear that it has the capacity to do so. When the WIP drops below the defined limit, that is the signal to start new work. Note this is

different from a push system, which demands that work starts on an item whenever it is requested.

Limiting WIP helps flow and improves the Scrum Team's self-management, focus, commitment, and collaboration.

Active Management of Work Items in Progress

Limiting WIP is necessary to achieve flow, but it alone is not sufficient. The third practice to establish flow is the active management of work items in progress. Within the Sprint, this management by the Scrum Team can take several forms, including but not limited to the following:

- Making sure that work items are only pulled into the Workflow at about the same rate that they leave the Workflow.
- Ensuring work items aren't left to age unnecessarily.
- Responding quickly to blocked or queued work items as well those that are exceeding the team's expected Cycle Time levels (See Service Level Expectation - SLE).

Service Level Expectation (SLE)

A service level expectation (SLE) forecasts how long it should take a given item to flow from start to finish within the Scrum Team's Workflow. The Scrum Team uses its SLE to find active flow issues and to inspect and adapt in cases of falling below those expectations.

The SLE itself has two parts: a range of elapsed days and a probability associated with that period (e.g., 85% of work items should be finished in eight days or less). The SLE should be based on the Scrum Team's historical Cycle Time, and once calculated, the Scrum Team should make it transparent. If no historical Cycle Time data exists, the Scrum Team should make its best guess and then inspect and adapt once there is enough historical data to do a proper SLE calculation.

Inspect and Adapt the Definition of Workflow

The Scrum Team uses the existing Scrum events to inspect and adapt its Definition of Workflow, thereby helping to improve empiricism and optimizing the value the Scrum Team delivers.

The following are aspects of the Definition of Workflow the Scrum Team might adopt:

- Visualization policies - for example, Workflow states - either changing the actual Workflow or bringing more transparency to an area in which the team wants to inspect and adapt.
- How-we-work policies - these can directly address an impediment. For example, adjusting WIP limits and SLEs or changing the batch size (how often items are pulled between states) can have a dramatic impact.

Flow-Based Events

Kanban in a Scrum context does not require any additional events to those outlined in *The Scrum Guide*. However, using a flow-based perspective and metrics in Scrum's events strengthens Scrum's empirical approach.

The Sprint

The Kanban complementary practices don't invalidate the need for Scrum's Sprint. The Sprint and its events provide opportunities for inspection and adaptation of both product and process. It's a common misconception that teams can only deliver value once per Sprint. In fact, they must deliver value at least once per Sprint. Teams using Scrum with Kanban use the Sprint and its events as a feedback improvement loop by collaboratively inspecting and adapting their Definition of Workflow and flow metrics.

Kanban practices can help Scrum Teams improve flow and create an environment where decisions are made just-in-time throughout the Sprint based on inspection and adaptation. In this environment, Scrum Teams rely on the Sprint Goal and close collaboration within the Scrum Team to optimize the value delivered in the Sprint

Sprint Planning

A flow-based Sprint Planning meeting uses flow metrics as an aid for developing the Sprint Backlog. Reviewing historical throughput can help a Scrum Team understand their capacity for the next Sprint.

Daily Scrum

A flow-based Daily Scrum focuses the Developers on doing everything they can to maintain consistent flow. While the goal of the Daily Scrum remains the same as outlined in *The Scrum Guide*, the meeting itself takes place around the Kanban board and focuses on where flow is lacking and on what actions the Developers can take to get it back.

Additional things to consider during a flow-based Daily Scrum include the following:

- What work items are blocked and what can be done to get them unblocked?
- What work is flowing slower than expected? What is the Work Item Age of each item in progress? What work items have violated or are about to violate their SLE and what can the Scrum Team do to get that work completed?
- Are there any factors not represented on the board that may impact our ability to complete work today?
- Have we learned anything new that might change what the Scrum Team has planned to work on next?

- Have we broken our WIP limit? And what can we do to ensure we can complete the work in progress?

Sprint Review

The Scrum Guide provides an outline of the Sprint Review. Inspecting Kanban flow metrics as part of the review can create opportunities for new conversations about monitoring progress towards the Product Goal. Reviewing Throughput can provide additional information when the Product Owner discusses likely delivery dates.

Sprint Retrospective

A flow-based Sprint Retrospective adds the inspection of flow metrics and analytics to help determine what improvements the Scrum Team can make to its processes. The Scrum Team using Kanban also inspects and adapts the Definition of Workflow to optimize the flow in the next Sprint. Using a cumulative flow diagram to visualize a Scrum Team's WIP, approximate average Cycle Time and average Throughput can be valuable.

In addition to the Sprint Retrospective, the Scrum Team should consider taking advantage of process inspection and adaptation opportunities as they emerge throughout the Sprint.

Similarly, changes to a Scrum Team's Definition of Workflow may happen at any time. Because these changes will have a material impact on how the Scrum Team performs, changes made during the regular cadence provided by the Sprint Retrospective event will reduce complexity and improve focus, commitment and transparency.

Increment

Scrum requires the team to create (at minimum) a valuable, useful Increment every Sprint. Scrum's empiricism encourages the creation of multiple valuable increments during the Sprint to enable fast inspect and adapt feedback loops. Kanban helps manage the flow of these feedback loops more explicitly and allows the Scrum Team to identify bottlenecks, constraints, and impediments to enable this faster, more continuous delivery of value.

Endnote

Scrum is not a process or technique. It is a framework within which people can address complex adaptive problems while productively and creatively delivering products of the highest possible value. As *The Scrum Guide* points out, it functions well as a container for other techniques, methodologies, and practices.

The flow optimization practices of Kanban provide Scrum Teams with additional opportunities to inspect the right thing, at the right time, and then based on that inspection, adapt as needed. Kanban's hyperfocus on transparency, visualization, and flow maximizes feedback, empiricism, and ultimately the delivery of value.

History and Acknowledgments

Kanban's use in the context of creative knowledge work mostly originated in 2006 on a team at Corbis, a media licensing company in Seattle. Those practices quickly spread to encompass a large and diverse international community that over the years continued to enhance and evolve the approach.

This guide was developed collaboratively by Scrum.org, its Professional Scrum Trainer Community, Steve Porter, Yuval Yeret, and Daniel Vacanti.

A special thank you to Glaudia Califano, Louis-Philippe Carignan, Charles Bradley, Jose Casal, Andy Hiles, Jesse Houwing, and Julia Wester for their contributions. We also owe a debt of gratitude to all those practitioners who have in the past contributed to make Kanban a viable and successful lean-agile strategy.