Sensemaking in the Agile Forest

Kanban in a nutshell [1]

Henny Portman

Introduction[2]

Kanban is originally a system from the manufacturing industry to reduce inventory and deliver exactly on time (just in time, or JIT). It was developed by Japanese Taiichi Ohno at car manufacturer Toyota (first version in 1953). This system keeps production process overloads in check by working with upper limits for the number of pieces of work in progress.

Kanban method

Based on Taiicho Ohno's Kanban system, a Kanban method has been developed to support knowledge workers. One of the first to write about it was David J. Anderson on its application in software development. Today, we see more and more disciplines using Kanban systems to manage their work, such as marketing, product development, human resources, legal, auditing, et cetera. However, many implementations go no deeper than a board and the visualization of work in progress. Few reach the depths that are behind Kanban, including measuring numbers and lead times and actively managing work in progress.

The Kanban philosophy

Kanban literally means 'signal card' in Japanese. The main features of a Kanban system are:

- The process to be managed is divided into several different steps.
- The workflow is visualized.
- Work in progress is limited.
- Lead times are controlled and measured.
- Performance agreements, such as the number of units of work in progress or lead times, are recorded.
- Feedback loops are frequently used (with the team and with the customer/client).
- The workflow can be subdivided by service classes (incidents, maintenance, projects).
- The team makes joint agreements on how the work is managed.
- The team improves the way of working together.

[1] How to cite this article: Portman, H. (2023). Kanban in a nutshell, Sensemaking in the Agile Forest series, PM World Journal, Vol. XII, Issue IV, April.

[2] This article is based on a paragraph of my Dutch book Scaling agile in organisaties

The Kanban system

Introducing Kanban is an evolutionary process. Start by focusing on quality. Then reduce the work in progress. Having too much work in progress in the system means frequent multitasking. To implement a Kanban system, the following steps can be followed (see example):

- Determine the start and end point of the process, including the intermediate steps, and record this work process on a visual board so that the status of work in progress and where a specific work unit is located can be seen immediately.
- Add queues and buffers to control the flow in the process.
- Define the design of signal cards. Define what information will be recorded on them (ID number, name, description, recording date, any hard completion date, etc.).
- Define the boundaries at the beginning and at the end of the process. Include a corresponding queue at the start and end and define how and when prioritization of the input queue takes place and how products are transferred to the receiving party.
- For each process step, determine the maximum number of work units in progress (the WIP limit: see the number at the top of each column in the example). If the system can handle more, this is immediately visible and new work can be requested ('pull' system).
- Determine which types of services are provided. Each service is characterized by its own agreements on, for instance, expected lead time of realization and how many of these services can be in the system at the same time. Agree how the different services are visible on the Kanban board. For example, by using different colored post-its.
- Make visible if work units expire during the process. This provides insight to applicants and increases the quality of the application process.
- Record all agreements in service agreements and policy guidelines and regularly evaluate whether policy guidelines should be added or adjusted.
- Measure the turnaround time per service type and optimize the process. If necessary, adjust the number of units of work in progress to reduce the lead time and increase predictability.

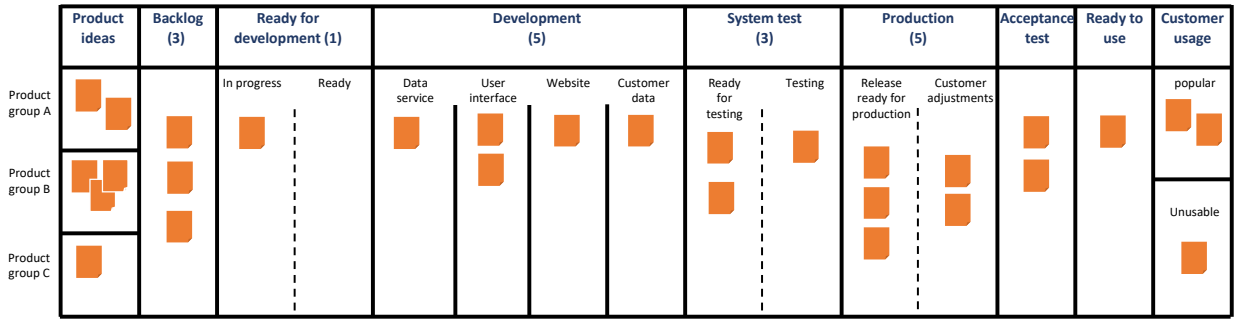

Figure 1 Example Kanban board

Differences between Kanban and Scrum

Scrum approach	**Kanban approach**
Introducing Scrum implies a revolution. The specific approach, techniques, roles and responsibilities must be embraced by all stakeholders.	Introducing Kanban is an evolutionary approach. The existing way of working, techniques, roles and responsibilities initially remain unchanged.
No changes can be made during the execution of a Sprint. This can only be done prior to a subsequent Sprint.	It is possible to address a highest priority change immediately.
Scrum is an iterative approach with a set "heartbeat" (the sprint) and set times to go into production.	Kanban involves a continuous flow of work (no sprint) and changes can be put into production on demand.
Scrum has conventions for setting up sprints, making estimates, and expecting everyone to participate in daily scrums and sprints.	Within Kanban, demand agreements (such as prioritization, lead time and limited work in progress) are defined in advance. In addition, there are no specific agreements. Everyone can continue to perform their own work, in their own way. The team can decide to have a daily stand-up, but Kanban does not prescribe this.
The Sprint Backlog shows similar units.	Tasks in a Kanban system often vary in size and scope.
Within Scrum, different options can be worked out and many processes can be used to arrive at a solution.	Within a Kanban system, there is a firmly defined business process. The Kanban system merely follows this process.
Within Scrum, the focus is on effectiveness.	Within Kanban, the focus is primarily on efficiency and then on effectiveness.
Scrum uses a multidisciplinary team.	The process steps in the workflow determine who is on the team.
Scrum has three standard roles: Scrum Master, Product Owner and developers.	Kanban has no standard roles.
Above all, scrum developers must use their knowledge, insights and expertise and work together. The Scrum guide does not give prescriptions at the level of an individual. It helps if members are generalists (T- or π- or M-shaped employees) so they can take over work from team members.	Kanban team members are mostly specialists (I-shaped employees).

Scrumban

Scrumban is the combination of Scrum and Kanban. Initially, it was intended as a transitional model to move from Scrum to Kanban and let the team experience Lean and Kanban concepts. Today, it is an approach where the team has chosen to work according to Scrum, i.e. with Sprints, but to use the Kanban system to continuously review and improve its working methods in order to thereby optimize the flow of units of work (e.g. User Stories).

The main features of a Scrumban system are:

- The team works in short iterations.
- The flow of work is visualized on a board with columns: to do, doing, and done.
- Work in progress is limited. A person works on only one task
- A team member starts by himself on the next task to be performed.
- Planning is performed the moment the number of User Stories on the Backlog is below a certain level.
- During planning, the priority of the User Stories is also determined.
- Within Scrumban, bucket size planning can be used for long-term planning. In this, there are three containers: the one-year container containing possible ideas, the six-month container in which the ideas are turned into goals, and the three-month container in which the goals are developed into user stories.
- There are no specific requirements for the team in terms of specific roles or a maximum number of team members.
- Just before the deadline of a specific project, Scrumban applies the concept of feature freeze. No new features/user stories may be picked up from that point on (see Figure 2).
- After the feature (or user story) freeze, triage (selection) by the product owner can take place. The product owner determines which pending features/user stories will and/or will not be completed (see also Figure 2).

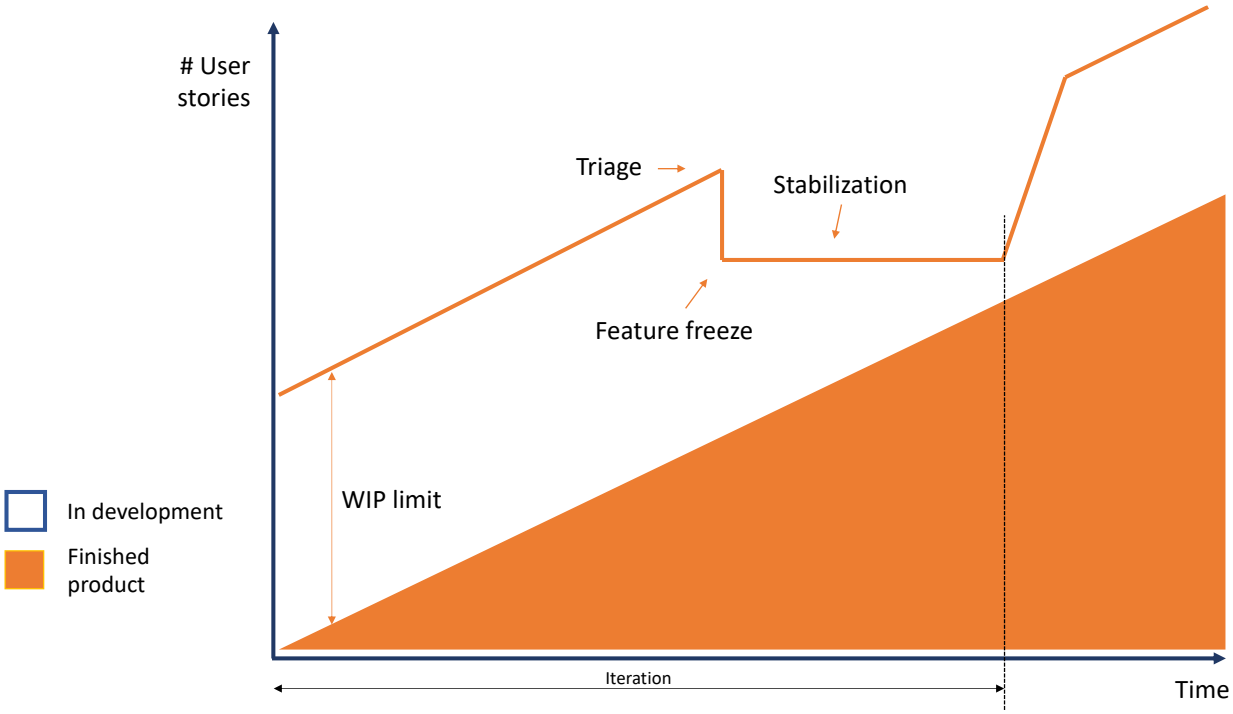

Figure 2 Ideal trajectory of a Scrumban release (source: www.continuousagile.com)

The team delivers user stories in a continuous stream. Since the team commits to an agreed upon iteration length, somewhere around two-thirds of the iteration the team will no longer pick up new user stories (Feature freeze in Figure 2) and some user stories may even be placed back on the backlog (Triage in Figure 2).

Sensemaking in the Agile Forest series

This article is part of a series of articles called *Sensemaking in the Agile Forest*. This series[3] consists of the following parts:

- Portman, H. (2022). What is Agile? Sensemaking in the Agile Forest series, PM World Journal, Vol. XI, Issue I, January.
- Portman, H. (2022). What is Scrum? Sensemaking in the Agile Forest, series article 2, *PM World Journal*, Vol. XI, Issue II, February
- Portman, H. (2022). Is agile always better? Sensemaking in the Agile Forest series, *PM World Journal*, Vol. XI, Issue III, March
- Portman, H. (2022). The ideal Product Owner, Sensemaking in the Agile Forest series, *PM World Journal*, Vol. IX, Issue IV, April

[3] This series is based on a number of short blogs I made for Forsa Advies, a project management training organization in the Netherlands (https://www.forsa-advies.nl).

- Portman, H. (2022). The Ideal Scrum Master, Sensemaking in the Agile Forest series, PM World Journal, Vol. XI, Issue V, May
- Portman, H. (2022). Is an agile team always autonomous? Sensemaking in the Agile Forest series, PM World Journal, Vol. XI, Issue VI, June
- Portman, H. (2022). What do iterative and incremental mean in Agile? Sensemaking in the Agile Forest series, *PM World Journal*, Vol. XI, Issue VII, July
- Portman, H. (2022). The Minimum Viable Product (MVP) unraveled; Sensemaking in the Agile Forest, series article, *PM World Journal,* Vol. XI, Issue VIII, August
- Portman, H. (2022). Prioritizing in an agile team, Sensemaking in the Agile Forest, series article, *PM World Journal*, Vol. XI, Issue IX, September
- Portman, H. (2022). Multitasking, task-switching or monotasking; Sensemaking in the Agile Forest series, *PM World Journal*, Vol. XI, Issue X, October
- Portman, H. (2022). Being predictable as an agile team; Sensemaking in the Agile Forest series, *PM World Journal*, Vol. XI, Issue XI, November
- Portman, H. (2022). Self-managing or self-organizing agile teams, Sensemaking in the Agile Forest series article, *PM World Journal*, Vol. XI, Issue XII, December
- Portman, H. (2023). Slicing user stories, Sensemaking in the Agile Forest series, PM World Journal, Vol. VII, Issue I, January
- Portman, H. (2023). Agile management products, Sensemaking in the Agile Forest series article, PM World Journal, Vol. VII, Issue II, February
- Portman, H. (2023). Agile and testing, Sensemaking in the Agile Forest, series article, PM World Journal, Vol. XII, Issue III, March
- What is Kanban?
- Culture makes or breaks your agile transformation
- Why agility?
- Towards a more agile organization
- Getting started as an agile team (a pilot)
- Agile team of teams structures
- Agile centers of excellence (CoE)
- Knowledge sharing within and between agile teams
- The evolution of agile frameworks
- ?

Please let me know if you would like to add specific agile topics to this series.

About the Author

Henny Portman

The Netherlands

Henny Portman, owner of Portman PM[O] Consultancy and was partner of HWP Consulting, has 40 years of experience in the project management domain. He was the project management office (PMO) thought leader within NN Group and responsible for the introduction and application of the PMO methodologies (portfolio, program, and project management) across Europe and Asia. He trains, coaches, and directs (senior) programme, project and portfolio managers and project sponsors at all levels, and has built several professional (PM(O)) communities.

Henny Portman is/was accredited in a variety of qualifications, including P3O, PRINCE2, MSP, MoP, PRINCE2 Agile, AgilePM, AgilePgM and AgileSHIFT trainer and an SPC4 SAFe consultant and trainer. He is a P3M3 trainer and assessor and PMO Value Ring Certified Consultant (PMO Global Alliance). On behalf of IPMA, he assesses mega and large projects for the IPMA Project Excellence Award. In addition to this, he is an international speaker, author of many articles and books in the PM(O) field, and an active blogger (hennyportman.wordpress.com/).

Henny can be contacted at henny.portman@gmail.com.